

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO AMAZONAS
PRÓ-REITORIA DE PÓS-GRADUAÇÃO, PESQUISA E INOVAÇÃO – PPGI
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO – DIPESP
MESTRADO PROFISSIONAL EM ENSINO TECNOLÓGICO – MPET

SILMAR ANTONIO BUCHNER DE OLIVEIRA

**PROGRAMAÇÃO PARA ADMINISTRAÇÃO DE REDES DE COMPUTADORES:
UMA PROPOSTA DE ENSINO-APRENDIZAGEM BASEADA NO MODELO DE
SALA DE AULA INVERTIDA**

Manaus
2018

SILMAR ANTONIO BUCHNER DE OLIVEIRA

**PROGRAMAÇÃO PARA ADMINISTRAÇÃO DE REDES DE COMPUTADORES:
UMA PROPOSTA DE ENSINO-APRENDIZAGEM BASEADA NO MODELO DE
SALA DE AULA INVERTIDA**

Dissertação de mestrado, apresentada à Coordenação do Curso de Mestrado Profissional em Ensino Tecnológico do Instituto Federal de Educação, Ciências e Tecnologia do Amazonas (IFAM), Campus Manaus Centro, como requisito para a obtenção do Título de Mestre.

Orientadora: Profa. Dra. Andréa Pereira Mendonça

Manaus
2018

-
- O48p Oliveira, Silmar Antonio Buchner de.
Programação para administração de redes de computadores: uma proposta de ensino-aprendizagem baseada no modelo de sala de aula invertida. / Silmar Antonio Buchner de Oliveira. – 2018.
183 f.: il.
- Dissertação (Mestrado Profissional em Ensino Tecnológico) – Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, *Campus* Manaus Centro, 2018.
Orientadora: Profª. Dra. Andréa Pereira Mendonça.
1. Ensino tecnológico. 2. Ensino-aprendizagem. 3. Redes de computadores. I. Mendonça, Andréa Pereira. (Orient.) II. Instituto Federal de Educação, Ciência e Tecnologia do Amazonas. III. Título.

CDD 371.33

SILMAR ANTONIO BUCHNER DE OLIVEIRA

**PROGRAMAÇÃO PARA ADMINISTRAÇÃO DE REDES DE COMPUTADORES:
UMA PROPOSTA DE ENSINO-APRENDIZAGEM BASEADA NO MODELO DE
SALA DE AULA INVERTIDA**

Dissertação de mestrado, apresentada à Coordenação do Curso de Mestrado Profissional em Ensino Tecnológico do Instituto Federal de Educação, Ciências e Tecnologia do Amazonas (IFAM), Campus Manaus Centro, como requisito para a obtenção do Título de Mestre.

Orientadora: Profa. Dra. Andréa Pereira Mendonça

BANCA EXAMINADORA

Orientadora: Prof.^a Dra. Andréa Pereira Mendonça - IFAM

Prof.^a Dra. Elloá Barreto Guedes da Costa - UEA

Prof.^a Dra. Rosa Oliveira Marins Azevedo - IFAM

Manaus
2018

*À minha esposa Clediane (meu esteio) e aos
meus filhos Alexandra Eva e Tarsilo Enzo,
com amor e carinho.*

AGRADECIMENTOS

A Deus pelo dom da vida e pela força dada consonante com os desafios enfrentados ao longo da minha jornada nesta vida, pois nunca fiquei desamparado.

À minha esposa pela dedicação em assumir TUDO enquanto eu me dedicava às tarefas de mestrado. Pela força e pelo amparo nos momentos de fraqueza. Pelas palavras de incentivo quando a escuridão parecia me tomar. Pelas palavras duras nos momentos em que foram necessárias para que eu pudesse reprocessar o mundo a minha volta. Pelo afago nos momentos de desespero.

Aos meus filhos Alexandra Eva e Tarsilo Enzo pela alegria incessante que funcionou como combustível para seguir o caminho.

À Professora Dra. Andréa P. Mendonça pela sua imensa sabedoria, por acreditar em mim e pelo direcionamento correto quando os caminhos estavam tortuosos. Pelo incentivo perene e, principalmente, pela paciência.

Ao Professor Ricardo Pereira pelo seu altruísmo.

À Professora Dra. Rosa Oliveira Marins Azevedo por ter expandido o alcance do MPET, pelas palavras de incentivo em momentos delicados, por compartilhar seu conhecimento durante suas aulas e pelas valiosas contribuições no exame de qualificação.

À Professora Dra. Elloá Barreto Guedes da Costa pelas contribuições que ajudaram a nortear o caminho a ser trilhado, no Seminário de Projetos e no exame de Qualificação.

À Professora Dra. Lucilene Paes pelo acolhimento e direcionamento dos meus primeiros passos no MPET.

A todos os professores do MPET por contribuírem para meu crescimento no Mestrado.

Aos colegas de turma que sempre deram seu apoio, principalmente ao Cristiano, que me deu suporte em todos os momentos, ao Uthant pelo incentivo, à parceria do Erlande e ao Marcus Marcelo, que, além do incentivo, acolheram-me em suas famílias quando tive de me deslocar para Rio Branco e ao Raimundo pelo incentivo e apoio em Manaus.

Ao IFAC e ao IFAM pelas oportunidades.

Ao IFRO pela oportunidade de dedicação integral ao Mestrado e pelo apoio na implementação da proposta.

Aos estudantes que participaram da implementação da proposta, pela dedicação de cada um.

A todas as pessoas que de uma forma ou de outra auxiliaram para que este mestrado se realizasse em minha carreira.

Ao meu pai pelos valores ensinados e à minha mãe, que me abençoa lá do céu.

*Tenho andado distraído, impaciente e indeciso.
Ainda estou confuso, só que agora é diferente.
Estou tão tranquilo e tão contente!
(Renato Russo)*

RESUMO

As disciplinas iniciais de redes de computadores são, em geral, lecionadas de forma essencialmente teórica e com pouca associação com a prática. Diante disso, nossa pesquisa resultou em uma proposta de ensino-aprendizagem na qual, além da parte teórica, haja um maior direcionamento prático para que o estudante seja capaz de desenvolver habilidades para resolver problemas do dia a dia em Redes de Computadores com emprego de recursos de Programação. Elaboramos esta proposta seguindo os fundamentos do Alinhamento Construtivo para desenvolvimento das atividades de ensino, aprendizagem e avaliações. Também adotamos o modelo de ensino híbrido de sala de aula invertida, na qual o estudante tem acesso ao conteúdo previamente, com o auxílio de videoaulas, *links*, tutoriais, etc., e utiliza o tempo de sala de aula com o professor, para resolver problemas práticos do cotidiano de um profissional de Redes de Computadores, utilizando a linguagem Python em uma rede simulada, em um Laboratório Virtual. Avaliamos nossa proposta de ensino-aprendizagem por meio de um estudo de caso realizado com estudantes de nível superior na área de computação do município de Porto Velho/RO. Neste estudo, avaliamos o desempenho dos estudantes com respeito ao desenvolvimento de três habilidades no domínio de Redes de Computadores: Identificação de problemas; Automatização de soluções para os problemas; Associação das soluções encontradas com outros possíveis problemas. Os resultados deste estudo evidenciaram progresso dos estudantes sobre as três habilidades na resolução de problemas envolvendo monitoramento de equipamentos de redes e ações de usuários, análise de logs, backups remotos, sincronismo de dados e emissão automática de mensagens de alertas. Como produto desta pesquisa, organizamos um Guia Didático de Programação para Administração de Redes de Computadores contendo todo o material de ensino-aprendizagem e dinâmicas utilizadas para auxiliar na aplicação da proposta por outros professores de Redes de Computadores.

Palavras-chave: Ensino de Redes de Computadores. Programação para Redes de Computadores. Python. Sala de Aula Invertida. Alinhamento Construtivo.

ABSTRACT

The initial disciplines of computer networks are, in general, taught in an essentially theoretical way and with little association with practice. Therefore, our research has resulted in a teaching-learning proposal in which, in addition to the theoretical part, there is a greater practical orientation for the student to be able to develop skills to solve day-by-day problems in Computer Networks using Programming resources. We elaborated this proposal following the fundamentals of Constructive Alignment to develop teaching, learning and evaluation activities. We also adopt the inverted classroom hybrid teaching model, in which the student has access to the content previously, with the help of videos, links, tutorials, etc., and uses classroom time with the teacher, to solve practical problems on daily life of a professional of Computer Networks, using the Python language in a simulated network, in a Virtual Laboratory. We evaluated our teaching-learning proposal through a case study carried out with students of undergraduate courses in the area of computation of Porto Velho/RO city. In this study, we evaluated the performance of students with respect to the development of three skills in the field of Computer Networks: Identification of problems; Automation of solutions to problems; Association of solutions found with other possible problems. The results of this study evidenced students' progress on the three skills in solving problems involving network equipment and user actions monitoring, log analysis, remote backups, data synchronization and automatic issuance of alert messages. As a product of this research, we organized a Didactic Guide to Programming for Computer Network Administration containing all the teaching-learning material and dynamics used to assist in the application of the proposal by other teachers of Computer Networks.

Keywords: Computer Networks Teaching. Programming for Computer Network. Python. Flipped Classroom. Constructive Alignment.

LISTA DE QUADROS

Quadro 1: Exemplo de âncora e de questão motriz.	24
Quadro 2: Características da linguagem de programação Python.	30
Quadro 3: Componentes de uma rubrica (apresentação oral).	33
Quadro 4: IFs, CST e disciplinas observadas.	36
Quadro 5: Relação livro x conteúdo	40
Quadro 6: Módulos, temas, assuntos e conteúdos.	42
Quadro 7: Estrutura geral dos Roteiros de Aprendizagem (RA).	44
Quadro 8: Rubrica utilizada: Critérios (Habilidades) x Níveis de Desempenho (Conceitos). .	50
Quadro 9: Cronograma de realização dos estudos e avaliações modulares.	60
Quadro 10: Contexto e unidades de Atividades – Avaliação Diagnóstica.	72
Quadro 11: H1-Respostas do estudante Everaldo para os Estudos Presenciais.	75
Quadro 12: H1-Respostas do estudante Everaldo para a Avaliação Modular 01.	76
Quadro 13: H1-Respostas do estudante Ivan para o Estudo Presencial 06.	77
Quadro 14: Respostas do estudante Kelvin para a Avaliação Modular 01.	79
Quadro 15: H2-Avaliação Modular 02-Contexto do problema proposto (Âncora).	84
Quadro 16: H1 e H2-Estudo Presencial 03-Contexto do <i>Problema Inicial</i>	88
Quadro 17: H3-Estudo Presencial 03-Problemas a serem analisados.	89

LISTA DE FIGURAS

Figura 1: Dinâmica do Estudo Autônomo	45
Figura 2: Dinâmica do Estudo Presencial.....	45
Figura 3: Exemplos de videoaulas com <i>links</i> e QRCodes.	49
Figura 4: Publicação do edital do curso.....	58
Figura 5: Página de controle de inscrições de eventos do IFRO.	59
Figura 6: Destaque no portal do IFRO com notícia sobre o curso.	59
Figura 7: H2-Avaliação Diagnóstica – Exemplo de resolução dos estudantes.	81
Figura 8: H2-Avaliação Modular 02-Programa desenvolvido pelo estudante Kelvin.	85

LISTA DE GRÁFICOS

Gráfico 1: H1-Desempenho geral na Avaliação Diagnóstica.....	73
Gráfico 2: H1-Desempenho nos Estudos Presenciais do Módulo 01.....	73
Gráfico 3: H1-Desempenho geral na Avaliação Modular 01.....	76
Gráfico 4: H1-Desempenho nos Estudos Presenciais do Módulo 02.....	77
Gráfico 5: H1-Desempenho geral na Avaliação Modular 02.....	78
Gráfico 6: H1-Evolução ao longo das avaliações.....	79
Gráfico 7: H2-Desempenho geral na Avaliação Diagnóstica.....	80
Gráfico 8: H2-Desempenho nos Estudos Presenciais do Módulo 01.....	82
Gráfico 9: H2-Desempenho geral na Avaliação Modular 01.....	83
Gráfico 10: H2-Desempenho nos Estudos Presenciais do Módulo 02.....	83
Gráfico 11: H2-Desempenho geral na Avaliação Modular 02.....	84
Gráfico 12: H2-Evolução ao longo das avaliações.....	86
Gráfico 13: H3-Desempenho geral na Avaliação Diagnóstica.....	87
Gráfico 14: H3-Desempenho nos Estudos Presenciais do Módulo 01.....	88
Gráfico 15: H3-Desempenho geral na Avaliação Modular 01.....	90
Gráfico 16: H3-Desempenho nos Estudos Presenciais do Módulo 02.....	90
Gráfico 17: H3-Desempenho geral na Avaliação Modular 02.....	91
Gráfico 18: H3-Evolução ao longo das avaliações.....	91

LISTA DE ABREVIATURAS E SIGLAS

AEA – Atividades de Ensino e Aprendizagem
ASCII – American Standard Code for Information Interchange
CEFET – Centro Federal de Educação Tecnológica
CST – Curso Superior de Tecnologia
DHCP – Dynamic Host Configuration Protocol
DNS – Domain Name System
EBTT – Ensino Básico Técnico e Tecnológico
EUA – Estados Unidos da América
FTP – File Transfer Protocol
H1 – Habilidade 01
H2 – Habilidade 02
H3 – Habilidade 03
HTML – Hyper Transfer Markup Language
HTTP – Hypertext Transfer Protocol
HTTPS – Secure Hypertext Transfer Protocol
ICMP – Internet Control Message Protocol
IDE – Integrated Development Environment
IES – Institutos de Ensino Superior
IF – Instituto Federal
IFAC – Instituto Federal de Educação, Ciência e Tecnologia do Acre
IFAM – Instituto Federal de Educação, Ciência e Tecnologia do Amazonas
IFRO – Instituto Federal de Educação, Ciência e Tecnologia de Rondônia
IP – Internet Protocol
MIB – Management Information Base
NAT – Network Address Translator
NFS – Network File System
NTP – Network Time Protocol
OID – Object Identifier
OSI/ISO – Open System Interconnection /International Organization for Standardization
PING – Packet Internet Network Groper
POO – Programação Orientada a Objetos
PPC – Projeto Pedagógico de Curso

Próx. Profic. – Próximo de Proficiente

RA – Roteiro de Aprendizagem

RPA – Resultados Pretendidos da Aprendizagem

S. O. – Sistema Operacional

SNMP – Simple Network Management Protocol

SSH – Secure Shell

SSL/TLS – Secure Socket Layer / Transport Layer Security

TA – Tarefas de Avaliação – TA.

TCLE – Termo de Consentimento Livre e Esclarecido

TCP/IP – Transfer Control Protocol / Internet Protocol

UDP – User Datagram Protocol

WAN – Wide Area Network

SUMÁRIO

INTRODUÇÃO	16
1 FUNDAMENTAÇÃO TEÓRICA	21
1.1 ENSINO DE REDES DE COMPUTADORES NOS CST	21
1.2 RESOLUÇÃO DE PROBLEMAS E APRENDIZAGEM “MÃO NA MASSA”	23
1.3 ENSINO HÍBRIDO	26
1.3.1 Sala de aula invertida	27
1.4 PROGRAMAÇÃO PARA REDES DE COMPUTADORES	29
1.4.1 Linguagem Python	30
1.5 ALINHAMENTO CONSTRUTIVO	31
1.6 AVALIAÇÃO POR RUBRICAS	32
1.7 TRABALHOS RELACIONADOS.....	34
2 PROPOSTA DE ENSINO-APRENDIZAGEM DE PROGRAMAÇÃO PARA ADMINISTRAÇÃO DE REDES DE COMPUTADORES	39
2.1 CONTEÚDOS, MÓDULOS, TEMAS E PROBLEMAS.....	39
2.2 ROTEIROS DE APRENDIZAGEM PARA ESTUDO AUTÔNOMO E ESTUDO PRESENCIAL	43
2.3 LABORATÓRIO VIRTUAL	46
2.4 AMBIENTE VIRTUAL E RECURSOS DIGITAIS PARA GUIAR OS ESTUDOS.....	48
2.5 INSTRUMENTOS DE AVALIAÇÃO E UTILIZAÇÃO DE RUBRICAS.....	49
3 ESTUDO DE CASO.....	52
3.1 PLANEJAMENTO	52
3.1.1 Questões de pesquisa.....	52
3.1.2 Definição dos sujeitos da pesquisa.....	53
3.1.3 Recursos utilizados.....	53
3.1.4 Planejamento da aplicação	54
3.1.5 Unidades de análise.....	56
3.1.6 Coletas de evidências	56
3.2 EXECUÇÃO.....	57
3.2.1 Semana 01 – 13/11/2017 a 18/11/2017 – Atividades iniciais.....	61
3.2.2 Semana 02 – 19/11/2017 a 25/11/2017 – Módulo 1 – Tema 01	62
3.2.3 Semana 03 – 26/11/2017 a 02/12/2017 – Módulo 1 – Tema 02	63
3.2.4 Semana 04 – 03/12/2017 a 09/12/2017 – Módulo 1 – Tema 03.....	64
3.2.5 Semana 05 – 10/12/2017 a 16/12/2017 – Módulo 1 – Preparação e aplicação da Avaliação.....	65
3.2.6 Semana 06 – 02/01/2018 a 06/01/2018 – Módulo 2 – Tema 04.....	65
3.2.7 Semana 07 – 07/01/2018 a 13/01/2018 – Módulo 2 – Tema 05.....	66
3.2.8 Semana 08 – 14/01/2018 a 27/01/2018 – Módulo 2 – Tema 06.....	66
3.2.9 Semana 09 – 28/01/2018 a 03/02/01/2018 – Módulo 2 – Preparação e aplicação da Avaliação.....	67
3.3 ANÁLISE DOS DESVIOS.....	67
3.4 AVALIAÇÃO DA VALIDADE	68
3.4.1 Validade interna	69
3.4.2 Validade externa.....	69
3.4.3 Validade de conclusão.....	70

3.4.4 Validade de construção	70
4 RESULTADOS E DISCUSSÕES	71
4.1 HABILIDADES PARA IDENTIFICAR SOLUÇÕES PARA PROBLEMAS DE REDES DE COMPUTADORES	71
4.2 HABILIDADES PARA AUTOMATIZAR SOLUÇÕES PARA PROBLEMAS DE REDES DE COMPUTADORES.....	80
4.3 HABILIDADES EM ASSOCIAR AS SOLUÇÕES ENCONTRADAS COM OUTRAS POSSÍVEIS SOLUÇÕES DE OUTROS PROBLEMAS DE REDES	86
4.4 LIMITAÇÕES VERIFICADAS NA APLICAÇÃO DA PROPOSTA	92
CONSIDERAÇÕES FINAIS.....	94
REFERÊNCIAS.....	96
APÊNDICE A – Levantamento sobre cursos do eixo Informação e Comunicação nos Institutos Federais.....	100
APÊNDICE B – Anúncios de emprego para Analistas de Suporte Técnico.....	103
APÊNDICE C – Roteiros de Aprendizagem.....	105
APÊNDICE D – Avaliações Modulares.....	145
APÊNDICE E – TCLE e Questionário SEE	150
APÊNDICE F – Avaliação Diagnóstica.....	156
APÊNDICE G – Artigos Publicados.....	161

INTRODUÇÃO

As indagações que motivaram esta pesquisa tiveram origem em experiências e reflexões sob dois aspectos: individualmente, antes do ingresso no mestrado; e em conjunto com minha orientadora, após o ingresso no mestrado. Individualmente, as indagações durante minha vida profissional se formaram durante uma experiência de quatro anos na docência para ensino superior e, concomitantemente, enquanto analista de suporte técnico no serviço público. Embora tenha me graduado em Sistemas de Informação, que tem seu foco na produção de *software*, as oportunidades profissionais que se sucederam foram direcionadas a tarefas que envolviam a administração de Redes de Computadores e seus serviços inerentes. Estas demandas profissionais eram contrastantes com a minha formação na graduação em Sistemas de Informação, pois na matriz curricular constava apenas uma disciplina de Redes de Computadores que se mostrou insuficiente para a resolução de diversos problemas do domínio de Redes de Computadores com os quais acabei me deparando na vida profissional. Além de única, tratava-se de uma disciplina aplicada exclusivamente teórica, aprofundada em fundamentos sobre os modelos de protocolos dispostos em camadas OSI/ISO (*Open System Interconnection / International Organization for Standardization*) e TCP/IP (*Transfer Control Protocol / Internet Protocol*). Como não houve a parte prática inerente ao contexto de Redes de Computadores, muita dedicação foi necessária para atender as demandas profissionais que iniciavam.

Após a graduação, já no serviço público¹, atuava em conjunto com outros profissionais cujas formações se distribuíam entre tecnólogos em Redes de Computadores e bacharéis em Sistemas de Informação, todos nós aprovados em concurso público para o cargo de Analista de Suporte Técnico. Neste cargo, a equipe tinha que prover serviços para manutenção de uma rede de computadores que convergia todas as secretarias e autarquias que somavam mais de 5.000 computadores interligados ao mesmo núcleo por diversas tecnologias e compartilhando grande quantidade de recursos. Concomitantemente, ocorria a implantação da telefonia sobre IP (*Internet Protocol*) em todos os órgãos estaduais da capital e um programa para disponibilizar internet gratuita² via rádio para a população da capital e da maioria das cidades do interior. Havia muitos desafios e os profissionais oriundos daquele primeiro concurso completavam uma equipe relativamente pequena que tinha de se desdobrar

¹ Oriundo de concurso público para o cargo de analista de suporte técnico do Governo do Estado do Acre, em 2006.

² Programa Floresta Digital: www.florestadigital.acre.gov.br.

para conseguir garantir o mínimo de funcionamento da rede e dos sistemas hospedados no *data center*³.

As tarefas eram repetitivas e muito tempo era destinado a trabalhos manuais de controle e monitoramento da rede de computadores. Consequentemente, os resultados deixavam a desejar, pois tal monitoramento não era feito ininterruptamente e dependia da visualização humana para varredura dos arquivos de registro (*logs*) em busca de anomalias. Desta forma, algo importante que não fosse visualizado poderia acarretar algum problema. Esta situação gerou a necessidade de descobrir meios de automatizar tais trabalhos, de forma que houvesse mais tempo disponível para melhor execução de outras tarefas referentes à melhoria dos serviços oferecidos pela rede.

Logo, outros colegas se propuseram a engajarem-se a esse processo de automatização que envolvia o emprego de conhecimentos de programação. Entretanto, em poucas semanas, a maioria acabou por abandoná-lo. Observando o que ocorria, percebi que, apesar de todos terem cursado disciplinas de Programação e de Redes de Computadores nas suas respectivas graduações, estes colegas apresentavam dificuldades para associá-las às resoluções dos problemas vivenciados no contexto profissional que exigissem tais conhecimentos.

Uma possível origem para estas dificuldades foi aventada quando, concomitante a este emprego, iniciei na carreira docente⁴, lecionando em dois cursos de graduação de nível superior: Curso Superior de Tecnologia (CST) em Redes de Computadores e Bacharelado em Sistemas de Informação. Na oportunidade, lecionava a disciplina de Redes de Computadores para ambos os cursos e Algoritmos e Linguagem de Programação para o curso de Sistemas de Informação. É importante informar que todas as turmas eram iniciantes, nas quais foi possível observar algumas características que despertaram atenção:

- Os estudantes do curso de Sistemas de Informação não demonstravam grande interesse pela disciplina de Redes de Computadores e reclamavam da ausência de aulas práticas;
- Os estudantes do curso de Redes de Computadores davam pouca importância para a disciplina de Programação e, em seus protestos, alegavam que iniciaram no curso porque queriam trabalhar com Redes de Computadores e

³ *Data center* é o nome dado para um ambiente fechado que converge serviços de processamento, armazenamento e transmissão de informações digitais, com energia estabilizada e controles de temperatura e umidade, adequados ao funcionamento dos equipamentos. (GENG, 2015).

⁴ Após uma seleção de currículos na União Educacional do Norte – Uninorte, em Rio Branco/AC.

que programação estaria fora do contexto⁵.

Apesar de não ter lecionado qualquer disciplina de Programação para o curso de Redes de Computadores, ao comparar a forma como a mesma era lecionada por outros professores, percebi muita semelhança com a forma como eu a lecionava para as turmas iniciais do curso de Sistemas de Informação, com exemplos que visavam à resolução de cálculos matemáticos como equações de segundo grau, sequência de Fibonacci, fatoriais, cálculos geométricos, etc. Nos cursos de Sistemas de Informação esta abordagem era mais adequada, pois ao longo das disciplinas do currículo, os exemplos de problemas evoluíam e a programação passaria a ser empregada em contextos com maior complexidade, em geral, focando no desenvolvimento de sistemas. Entretanto, para o curso de Redes de Computadores, a adoção desta abordagem não possuía relação com o que seria enfrentado na futura atuação profissional dos egressos.

Quanto às disciplinas que lecionei de Redes de Computadores para ambos os cursos, havia ementas definidas pelos respectivos PPC (Projeto Pedagógico de Curso) e o conteúdo teórico determinado era muito extenso. Apesar de concordar que a teoria é a base para a evolução de qualquer área, o cumprimento de uma ementa com conteúdos teóricos tão extensos era, de certa forma, um fator complicador para o ensino também da prática, considerando a carga horária definida para a disciplina. Ainda que conseguisse identificar que faltava algum complemento prático, devido a minha pouca experiência docente, não conseguia vislumbrar um meio de acrescentar uma abordagem mais prática às aulas. Desta forma, repeti o mesmo caminho de meus professores da época da minha graduação, lecionando somente a teoria.

Em virtude de, atualmente, atuar como docente da área de Informática do Instituto Federal de Rondônia (IFRO), dediquei minha atenção de pesquisa ao ensino de Redes de Computadores no contexto dos Cursos Superiores de Tecnologia (CST), do eixo Informação e Comunicação.

Com o ingresso no Mestrado, iniciou-se a orientação e as indagações e motivações passaram a ser em conjunto com minha orientadora e, a partir daí, trilhamos esta pesquisa. Assim, iniciamos a identificação na literatura da área por autores que fundamentavam o que até então era fruto de uma experiência solo como estudante e como docente. Identificamos, por exemplo, que Chen (2014a), Kurose et al. (2002), Wang, Blum e McCoey (2014) são

⁵ Semelhante aos comportamentos identificados por Edelman, Lowe e Oswalt (2015) por parte de alguns estudantes dos cursos de Engenharia de Redes de Computadores.

autores que tecem críticas ao ensino essencialmente teórico nos cursos de Redes de Computadores e sugerem atividades práticas que envolvam programação para resolução de problemas como forma de maximizar o interesse dos estudantes para a área e desenvolver um conhecimento “comercializável”.

Por estarmos inseridos na Rede Federal de Ensino, que abrange os Institutos Federais, e que, por sua vez, têm a maioria dos cursos superiores destinados à modalidade de CST, esta modalidade tem um foco maior nesta pesquisa. A essência dos CST é a preparação de profissionais para o mercado de trabalho, pois, ao ingressarem na vida profissional, os egressos dos CST são demandados para a resolução de problemas práticos. No caso de atuação na área de Redes de Computadores, são comuns demandas voltadas para administração de Redes de Computadores, as quais envolvem automatização de procedimentos para monitoramento, controle e gerenciamento de serviços, execução de *backups*, sincronismo de dados, verificação de consumo de recursos, envio de mensagens de alerta via *e-mail*, entre outras.

Em nossa pesquisa, identificamos três habilidades relevantes ao aprendizado, voltadas à administração de Redes de Computadores.

- 1) Identificar soluções possíveis para problemas ocorridos no contexto de Redes de Computadores.
- 2) Automatizar as soluções identificadas para resolver os problemas. A automatização, por sua vez, requer a construção de um programa ou de um conjunto de programas desenvolvidos em uma linguagem de programação.
- 3) Associar as soluções dos problemas resolvidos com possíveis soluções de outros problemas similares.

Dados estes desafios, destacamos o *problema de pesquisa*: como podemos ensinar programação para administração de Redes de Computadores considerando o contexto de estudantes de nível superior na área de computação, principalmente de Cursos Superiores de Tecnologia (CST) do eixo Informação e Comunicação?

Tendo como foco este problema de pesquisa, o nosso *objetivo* foi o de desenvolver uma proposta de ensino-aprendizagem para favorecer a aprendizagem de programação para administração de Redes de Computadores por estudantes de nível superior na área de computação, principalmente os de CST.

A proposta de ensino-aprendizagem focou na resolução de problemas práticos para que os estudantes pudessem ter contato com alguns dos prováveis desafios que permearão a profissão após seus cursos regulares. Adotamos Python como linguagem de programação

devido à sua versatilidade, usabilidade e facilidade de aprendizagem. Implementamos a proposta na modalidade de ensino híbrido, mais especificamente, no modelo de sala de aula invertida, combinando Estudo Autônomo e Presencial, por entendermos que este modelo condiz com nosso intuito de incentivar que o estudante seja protagonista do seu próprio aprendizado e por utilizar meios digitais, comuns a esta geração de estudantes, como forma de auxiliar o alcance dos objetivos pretendidos de ensino-aprendizagem. Estes estudos foram conduzidos por meio de Roteiros de Aprendizagem (RA), pelo direcionamento que o instrumento fornece e cadencia durante o trajeto dos conteúdos abordados.

A fim de avaliarmos nossa proposta de ensino-aprendizagem, realizamos uma investigação empírica, do tipo *estudo de caso* com 15 (quinze) estudantes de cursos superiores da área de Computação, no período compreendido entre novembro de 2017 e fevereiro de 2018. Com a realização deste estudo de caso, buscamos respostas para as seguintes *questões de pesquisa*: (i) Em que aspectos o planejamento da proposta melhora as habilidades dos estudantes para identificarem soluções para problemas reais de Redes de Computadores? (ii) Em que aspectos o planejamento da proposta melhora as habilidades dos estudantes para automatizarem soluções por meio de Programação para problemas de Redes de Computadores? (iii) Em que aspectos o planejamento da proposta melhora as habilidades dos estudantes em associarem as soluções encontradas com outras possíveis soluções de outros problemas de redes? (iv) Quais limitações são verificadas na aplicação da proposta?

Em termos de *produto educacional*, esta pesquisa resultou em um guia para Ensino-Aprendizagem de Programação para Administração de Redes de Computadores, contendo orientações e recursos de apoio (indicação de materiais didáticos, planejamento de ensino, roteiros de aprendizagem, videoaulas, tutoriais, laboratório virtual de redes, etc.) para orientar professores de disciplinas de Redes de Computadores que desejarem reproduzir ou adaptar nossa proposta em seus contextos educacionais.

A fim de apresentarmos os detalhes de nossa pesquisa, organizamos este documento em quatro capítulos. No Capítulo 1, apresentamos os fundamentos que dão sustentação à pesquisa. No Capítulo 2, apresentamos nossa proposta de ensino-aprendizagem de Programação para Administração de Redes de Computadores. No Capítulo 3, descrevemos o planejamento, execução e resultados obtidos com o estudo de caso realizado para avaliação da nossa proposta de ensino-aprendizagem. No Capítulo 4 apresentamos os resultados obtidos e discussões sobre os mesmos.

1 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo apresentamos as bases teóricas que fundamentam nossa pesquisa. Na primeira seção, abordamos o contexto do ensino de Redes de Computadores nos Cursos Superiores de Tecnologia (CST) do eixo Informação e Comunicação. Na segunda seção, apresentamos o conceito de problema e a relação entre resolução de problemas e aprendizagem “mão na massa”, uma perspectiva importante quando se considera a formação de estudantes de CST. Posteriormente, apresentamos o conceito de ensino híbrido, com enfoque no modelo de sala de aula invertida, em virtude de este modelo subsidiar a dinâmica das aulas em nossa proposta de ensino-aprendizagem. A quarta seção é dedicada a contextualização de programação para redes de computadores. Na sequência, apresentamos o Alinhamento Construtivo que fundamentou a proposta de ensino-aprendizagem e a elaboração dos recursos didáticos. Na sexta seção, apresentamos o conceito de rubrica como instrumento para avaliação do desempenho de estudantes e finalizamos o capítulo com a sétima seção, apresentando os trabalhos que se relacionam com nossa proposta de ensino-aprendizagem.

1.1 ENSINO DE REDES DE COMPUTADORES NOS CST

Os Cursos Superiores de Tecnologia (CST) possuem regulamentação específica e uma delas é o Catálogo Nacional de Cursos Superiores de Tecnologia, disponibilizado pela Secretaria de Educação Profissional e Tecnológica do Ministério da Educação (SETEC/MEC). Os Catálogos são organizados por eixos, sendo o eixo Informação e Comunicação aquele que integra os cursos relacionados à área de Computação/Informática, como CST em Análise e Desenvolvimento de Sistemas, CST em Sistemas para Internet, CST em Redes de Computadores, CST em Jogos digitais, entre outros (BRASIL, 2016).

Por sua característica direcionada à prática profissional, os CST têm duração mais curta (entre 2 e 3 anos), em comparação com os cursos de bacharelado. Seus egressos possuem um perfil profissional dotado de compreensão sobre o processo produtivo, fundado no saber tecnológico e no ato de fazer, com autonomia para tomada de decisões profissionais, indo além do simples domínio operacional de técnicas de trabalho. Geralmente, os CST são oferecidos pelos Institutos Federais de Educação, pertencentes à Rede Federal de Educação Profissional, Científica e Tecnológica, criada em 2008. Há também a oferta de CST em universidades públicas e Instituições de Ensino Superior particulares (BRASIL, 2006;

BRASIL, 2002).

A fim de caracterizar o perfil das disciplinas de Redes de Computadores e de Programação em CST, realizamos em 2017 uma pesquisa *ad-hoc* para identificar CST do eixo Informação e Comunicação, oferecidos pelos Institutos Federais de Educação (Conforme APÊNDICE A). Como resultado desta pesquisa, identificamos uma relação de 60 (sessenta) CST, distribuídos em 40 (quarenta) instituições, conforme informações obtidas no sistema e-Mec do Ministério da Educação (E-MEC, 2017).

A partir da identificação dos cursos, analisamos as ementas das disciplinas iniciais de Redes de Computadores e identificamos que as mesmas eram majoritariamente teóricas, baseadas em alguma edição das obras de Andrew S. Tanenbaum (Redes de Computadores) e de James F. Kurose (Redes de Computadores e a Internet), centradas no modelo OSI e na pilha de protocolos TCP/IP.

A qualidade das obras é indiscutível. Contudo, os resultados da pesquisa *ad-hoc* apontam um distanciamento entre teoria e prática na forma como a disciplina de Redes de Computadores é ensinada, no tocante à resolução de problemas de redes de computadores, quando considerado o contexto profissional no qual atuam os egressos dos CST.

Na pesquisa, identificamos 9 (nove) CST que possuíam em sua matriz curricular uma disciplina cuja denominação (por exemplo, Programação de Scripts e Programação Aplicada à Gerência de Redes de Computadores) indicava a junção de redes de computadores e programação. Contudo, não identificamos informações sobre como as mesmas são aplicadas, de forma que se outro professor quiser repetir a experiência, terá dificuldades, pois não encontrará planejamento, metodologias detalhadas ou problemas práticos abordados nestas disciplinas.

Destacamos ainda que, nos CST do Eixo Informação e Comunicação há pelo menos uma disciplina de Programação no currículo, ensinada no primeiro ano do curso, antes da disciplina de Redes de Computadores. Tal organização do currículo poderia resultar no aproveitamento dos conhecimentos adquiridos em Programação, dentro da disciplina de Redes de Computadores, possibilitando aos estudantes a resolução de problemas práticos com o auxílio de Programação.

Corroborando os resultados de nossa pesquisa *ad-hoc*, Chen (2014a) aponta que os cursos de redes de computadores possuem muito conteúdo teórico, abstrato e de difícil entendimento, muitas vezes prejudicando o interesse na aprendizagem e resultando em uma capacidade prática deficitária. Aponta como solução a adoção de atividades práticas para a resolução de problemas reais. Kurose et al. (2002) citam laboratórios com aplicações práticas

e levantam questões sobre quando e quanto programar em redes de computadores. Enfatizam ainda que estudantes de redes de computadores que aprendem a codificar programas podem ter mais facilidade para internalizar e entender conceitos avançados, além de deixá-los com um conhecimento “comercializável”. Wang, Blum e McCoe (2014), ao discutirem o ensino de Redes de Computadores para iniciantes, sugerem que, além da introdução geral e tradicional, devem proporcionar atividades práticas que envolvam Programação para resolução de problemas.

Neste sentido, nossa proposta de ensino-aprendizagem é baseada em atividades práticas, no estilo “mão na massa” para a resolução de problemas que reflitam os desafios profissionais reais, conforme apresentamos na próxima seção.

1.2 RESOLUÇÃO DE PROBLEMAS E APRENDIZAGEM “MÃO NA MASSA”

Um problema pode ser definido como “uma situação que apresenta dificuldades a um indivíduo para as quais não há solução imediata” (MAZARÍO, 1999 *apud* MENDONÇA, 2010, p. 9). Em um problema, diferente de um exercício, há sempre algo novo, cuja definição não está claramente exposta, exigindo maior reflexão sobre como resolvê-lo. Ainda que exista uma linha tênue entre exercício e problema, em função da experiência do estudante⁶, no exercício as atividades objetivam a fixação de conceitos ou procedimentos contendo instruções ou exemplos semelhantes a exercícios realizados anteriormente. Um problema, por sua vez, inclui alguma “novidade”, que torna a questão mais desafiadora, forçando indagações e caminhos que o levem além do conteúdo e dos exemplos propostos (MENDONÇA, 2010).

Quando se trabalha com resolução de problemas, um aspecto importante diz respeito a sua proposição aos estudantes, pois é desejável que os problemas propostos levem os estudantes a vivenciarem experiências mais próximas daquelas que enfrentarão em seu contexto profissional. Como geralmente os problemas são apresentados aos estudantes de forma escrita, na sua elaboração deve-se ter o cuidado de torná-los o mais contextualizados possível à realidade.

Segundo Bender (2014), a elaboração de um problema deve levar em consideração duas partes importantes: *âncora* e *questão motriz*. O termo *âncora* é utilizado para descrever a base sobre a qual os questionamentos serão construídos, podendo envolver alguma situação

⁶ Um estudante experiente pode encarar como exercício o que seria problema para um estudante iniciante (MENDONÇA, 2010).

ou alguma mídia (vídeos, notícias, artigos, etc.) que melhor contextualize o problema. O termo *questão motriz* configura a questão principal, que deve ser clara, motivadora e significativa para os estudantes. Apresentamos no Quadro 1 um exemplo de emprego de *âncora* e de *questão motriz* no contexto de um problema de redes de computadores.

Quadro 1: Exemplo de âncora e de questão motriz.

Âncora
O seu chefe não é especialista em Redes de Computadores, entretanto, sempre lê <i>blogs</i> e conversa com outros “curiosos”. Ao abrir seus <i>e-mails</i> hoje, você viu que ele lhe enviou um texto com seguinte conjunto de questionamentos e determinações:
De: boss@minhaempresa.com.br Para: tic@minhaempresa.com.br Assunto: Recomendações importantes Nessa minha última viagem, consegui umas informações com um amigo que é fera em informática. Estava com uns problemas no Excel e no Word e ele resolveu num tapa! Então começamos a conversar e ele me disse que em nossa WAN local devemos utilizar frame-relay, por ser a tecnologia que eles utilizam para comunicação entre as filiais e a matriz. É muito segura, sabe? Acredito que poderíamos substituir o FTP, por exemplo, por frame-relay, para melhorar a segurança na disponibilização dos nossos arquivos. Falou também sobre a utilização de portas na comunicação entre os equipamentos e que os servidores deveriam utilizar números de portas aleatórias para melhorar a segurança. Também falou que o protocolo IP não dá garantia de entrega das informações. Sei que é isso que usamos para transmitir dados na nossa rede. Conforme ele me disse, o programa ideal é o TCP, que tem controle da conexão e, se houver queda, a conexão é retomada e as informações que não chegaram são retransmitidas. Deus me livre de usar o tal do UDP, que não nos dá garantia nenhuma, nem de entrega, nem de retransmissão. Se estivermos utilizando esse programa aí, pode parar! Ele disse que utilizamos um endereço que fica na placa de rede e outro que colocamos no computador que depende de outro número. Vamos economizar trabalho. Se já tem um endereço fixo, vamos utilizar este para que não haja confusão. Outra coisa, lembra daquele dia em que a operadora disse que estava passando por uma manutenção na rede física deles? Então, dê uma olhada se não tem a ver com o uso de https que está aparecendo agora no site do Banco do Brasil. Antes aparecia http e agora está aparecendo https. Confio em você. Se falei algo errado, me corrija. Mas o cara é fera! Abraços.
Questão motriz
Fundamentado nos seus conhecimentos sobre Redes de Computadores, envolvendo tipos de redes, protocolos, estruturas em camadas e endereçamento, como responder ao <i>e-mail</i> do seu chefe, de forma a corrigir os conceitos equivocados que lhe foram apresentados?

Fonte: Autoria própria.

Como pode ser observado no Quadro 1, o estudante, assumindo o papel de um profissional da área, é confrontado pelo seu chefe por meio de um *e-mail* cujo conteúdo apresenta uma série de incorreções e confusões sobre alguns conceitos básicos de redes de computadores. Este *e-mail* funciona como *âncora*, pois traz uma situação que pode fazer parte

do cotidiano de um profissional de redes, funcionando como base para questionamentos sobre os equívocos em seu conteúdo. Na sequência, é apresentada a *questão motriz*, que instiga o estudante a resolver o problema proposto, e, embora esteja claro o questionamento, não traz muitos detalhes sobre como o problema poderá ser resolvido, semelhante à forma real que os problemas são apresentados no dia a dia.

A resolução de problemas é um dos recursos utilizados para motivar o estudante a colocar a “mão na massa”, um termo utilizado para definir um tipo de aprendizagem que leva o estudante a aprender através de experiências com problemas que envolvem *fazer* coisas do mundo real. Neste sentido, aprendizagem “mão na massa” busca proporcionar este ambiente de aprendizado, onde os estudantes são encorajados a resolverem problemas que podem surgir no dia a dia profissional, compostos de características que, por sua natureza instigadora, motivam o estudante a *aprender fazendo* e estimulando a sua inventividade (BLIKSTEIN 2015; 2017. PORVIR, 2017).

A expressão “mão na massa” tem estreita ligação com o movimento *Maker*, que estimula a construção de soluções que possam resolver problemas cotidianos. O movimento *Maker* teve seu início marcado pela primeira publicação da revista *Make*, em 2005, nos Estados Unidos, evoluindo para a feira *Maker*, em 2006, onde pessoas comuns que construíam suas soluções em suas garagens se reuniam para expô-las. A essência deste movimento chegou às instituições de ensino como uma forma de aprendizado que envolve projetos, ideias e protótipos, convergindo para um aprendizado crítico, através de constructos do mundo real onde o estudante aprende fazendo (DOUGHERTY, 2012; HATCH, 2014, BIANCONI, 2013).

A aprendizagem “mão na massa” busca fazer com que o estudante se adapte às mudanças tecnológicas, preparando-o para as variações no mercado de trabalho, pois as profissões podem sofrer mutações, novas profissões podem surgir e outras podem ser extintas, mas se o estudante aprendeu a *fazer*, enfrentando problemas reais e se adaptando a estes, estará mais preparado para as mudanças (BLIKSTEIN 2015; 2017).

Penido (2016), ao falar sobre aprendizagem “mão na massa”, sugere um melhor aprendizado desta geração íntima dos recursos digitais, a partir de atividades práticas.

Novos modelos de escola também partem do princípio de que a geração atual aprende melhor quando se envolve em atividades práticas ou de **experimentação**. Ou seja, quando os estudantes ampliam seus conhecimentos e habilidades a partir da resolução de problemas reais ou da elaboração de projetos concretos, que dão sentido e aplicabilidade ao que aprendem (Grifo da autora).

Nossa proposta de ensino-aprendizagem é inspirada na aprendizagem por problemas de forma direcionada ao domínio de redes de computadores na perspectiva de aprendizagem

“mão na massa” para gerar soluções com emprego de programação. Em geral, quando se fala em aprendizagem “mão na massa” é muito comum associá-la à construção de produto material como, por exemplo, um robô ou algum artefato produzido com impressora 3D. No contexto deste trabalho, a construção do estudante resulta em um programa que é um produto imaterial, isto é, não tem uma existência palpável.

Com isto pretendemos que o estudante se prepare para os desafios vindouros, pois a experiência de identificar um problema, buscar sua solução, construí-la por meio de programação e poder associá-la a outras soluções possíveis através do mesmo processo está em consonância com a forma *Maker* de resolver problemas autonomamente.

Para ajudar o estudante a estudar de forma mais autônoma e ainda maximizar o tempo de sala de aula para a resolução dos problemas, adotamos o modelo de ensino híbrido de sala de aula invertida, conforme apresentamos na seção seguinte.

1.3 ENSINO HÍBRIDO

A palavra *híbrido* significa mesclado, misturado e, trazendo para a educação, remete a um modelo de ensino que aproveita recursos informatizados para sustentar parte do aprendizado de forma *online*, buscando fazer com que o estudante explore, de forma autônoma, seu próprio potencial de aprendizagem. A outra parte do aprendizado é fundamentada no ensino presencial, com a assistência de um professor. Assim, o ensino híbrido busca extrair o melhor dos dois *mundos* – *online* e presencial (CHRISTENSEN; HORN; STAKER, 2013).

Há diferentes modelos de implementação do ensino híbrido, uns categorizados como sustentados e outros como disruptivos. Os modelos sustentados baseiam-se mais no ensino presencial. Os modelos disruptivos baseiam-se mais fortemente no ensino *online*, buscam explorar espaços diferenciados de aprendizagem, prover maior flexibilização do currículo e atender a um público que tem mais dificuldade para estudar em um modelo presencial (CHRISTENSEN; HORN; STAKER, 2013).

Neste sentido, modelos de ensino híbrido como *Flex*, *A la carte*, *Virtual enriquecido* e *Rotação individual* seguem uma tendência mais disruptiva, pois tem o ensino *online* como eixo norteador (CHRISTENSEN; HORN; STAKER, 2013).

- *Flex*: o ensino é praticamente todo *online* e o estudante avança conforme seu próprio ritmo, recorrendo a um professor ou tutor presencial em caso de

dúvidas;

- *A la carte*: os estudantes participam de um ou mais cursos inteiramente *online*, com um professor responsável, também *online* e, ao mesmo tempo, continuam a ter experiências educacionais em escolas tradicionais;
- *Virtual enriquecido*: Direcionado a determinada disciplina sobre a qual o estudante possua mais necessidade de aprofundamento. Esta disciplina tem conteúdos *online* disponíveis para a melhoria do aprendizado;
- *Rotação individual*: Diferente do modelo rotacional no sentido de que, neste, os estudantes não passarão, necessariamente, por todas as estações, mas somente pelas que foram previamente customizadas para o seu aprendizado.

Conforme descrito, os modelos sustentados possuem como eixo norteador o ensino presencial, são eles: *Rotação por estações*, *Laboratório rotacional* e *Sala de aula invertida* (CHRISTENSEN; HORN; STAKER, 2013):

- *Rotação por estações*: Os estudantes se revezam em sala de aula ou turmas, em estações com temas ou atividades distintas. Para que a rotação seja completa, os estudantes passam por todas as estações;
- *Laboratório rotacional*: Parte do aprendizado ocorre dentro da sala de aula e parte ocorre em laboratório com ensino *online*;
- *Sala de aula invertida*: Parte das atividades é organizada para ser efetuada *online*, geralmente através de vídeos, antes da aula presencial, fora da sala de aula. O espaço da sala de aula, por sua vez, destina-se ao desenvolvimento de atividades com auxílio ou orientação do professor, como por exemplo, resolução de problemas.

Adotamos o modelo de sala de aula invertida, principalmente, pelo maior tempo que o professor pode dedicar ao auxílio das atividades práticas do estudante. A seguir, detalhamos este modelo e como isto ocorre.

1.3.1 Sala de aula invertida

O modelo de sala de aula invertida foi idealizado por Bergmann e Sams (2012) no ano de 2006, quando lecionavam Química em uma escola de ensino médio, relativamente rural, no interior do estado do Colorado (EUA). Por motivos particulares, era comum que

alguns estudantes faltassem às aulas, o que acarretava retrabalho dos professores e perdas de conteúdos pelos estudantes. No intuito de sanar o problema, os dois professores resolveram gravar suas aulas em vídeos e postá-las *online*. Isto resolveu o problema dos faltosos e, colateralmente, beneficiou os outros estudantes, pois ainda que não faltassem, aderiram aos vídeos. Os estudantes relataram que isto melhorou seu aprendizado, pois poderiam reposicionar a execução dos vídeos aos pontos em que não haviam compreendido e rever o conceito sempre que necessário. Com isto, os professores resolveram gravar todas as suas aulas em vídeo (ao que chamaram de pré-aula) e propuseram que os estudantes os assistissem antes das aulas presenciais, que seriam destinadas à resolução de problemas envolvendo a teoria estudada. Com o sucesso da experiência, os professores resolveram aprimorar o modelo de ensino de sala de aula invertida e divulgá-lo para que outros professores pudessem aplicá-lo.

Segundo o modelo, a pré-aula concentra-se em vídeos construídos pelos professores ou cuidadosamente buscados *online*, de acordo com os conteúdos abordados. Durante a produção das aulas autorais é proposto que os professores busquem deixar seus vídeos intuitivos e atraentes, com uso de textos, anotações, inflexões de voz, etc. Os vídeos devem ser objetivos, não devendo ultrapassar quinze minutos de duração, onde cada tópico deve corresponder a um vídeo (BERGMANN; SAMS, 2012).

É muito importante que os professores se preocupem com a adoção de tecnologias em benefício do ensino-aprendizagem, visto que esta geração de estudantes normalmente é bastante ambientada com vídeos, mensagens instantâneas e ambientes virtuais para diversos fins. Assim, diversos recursos podem ser utilizados na sala de aula invertida a fim de aproveitar esta afinidade, como Youtube, ambientes de aprendizagem como Moodle, Google Classroom, simuladores virtuais de máquinas e ambiente computacionais como VirtualBox ou VMWare⁷ e também comunicação via Whatsapp, Telegram e *e-mails*.

Ao contrário do que possam parecer, os recursos tecnológicos desenvolvidos e/ou adotados antes da aula presencial não são o maior benefício da sala de aula invertida, mas sim, o tempo liberado para atividades práticas presenciais com maior qualidade e mais envolvimento dos estudantes em estratégias auxiliares como a aprendizagem baseada em projetos e/ou problemas. Além disso, com mais tempo livre, o professor pode auxiliar melhor a aplicação do conteúdo, tirar dúvidas e direcionar maior atenção aos estudantes com dificuldades nas aulas presenciais (BERGMANN; SAMS, 2012).

⁷ Oracle VM VirtualBox e VMWare são *softwares* para virtualização de máquinas e sistemas operacionais.

Considerando que em nossa proposta de ensino-aprendizagem as atividades dos estudantes envolvem o desenvolvimento de programas, destinamos a próxima seção para tratar sobre programação no contexto de redes de computadores.

1.4 PROGRAMAÇÃO PARA REDES DE COMPUTADORES

De forma simples, podemos compreender programação de computadores como uma atividade que resulta na construção de um programa e que um programa é um conjunto de instruções, escrita em uma linguagem de programação, e que “diz” ao computador o que deve executar, por exemplo, resolver um cálculo matemático, criar um jogo, plotar um gráfico, etc. Segundo Resnick (2013), atualmente, programar é uma habilidade tão necessária quanto a escrita. Aprender a ler e escrever são uma necessidade de todo cidadão, pois complementa sua comunicação e permite que este interaja com outras pessoas, que crie textos onde possa se expressar e crie novas maneiras de pensar e de agir. Ensinar uma criança a escrever é um dos primeiros caminhos para a construção da sua autonomia. Contudo, o fato de uma criança ter que aprender a ler e escrever não significa que, no futuro, se torne um escritor ou jornalista, mas é algo que facilitará a forma como absorve e produz conhecimento. Da mesma forma, o fato de uma pessoa aprender a programar não significa que se tornará um programador profissional ou um Analista de Sistemas, mas que esta habilidade, tanto quanto ler ou escrever, lhe fornecerá outra forma de expressão, em que pode criar ou construir conhecimento através do domínio de recursos computacionais. Saber programar possibilita maior criatividade, raciocínio lógico e sistemático mais apurado, além de facilidade para trabalho colaborativo através das trocas de experiências (RESNICK, 2013; RESNICK et al. 2009).

No contexto profissional, sobretudo dos profissionais egressos de CST do eixo de Informação e Comunicação, programar é uma necessidade para automatizar soluções de problemas e no caso de redes de computadores não é diferente. Por exemplo, uma varredura em um arquivo de registro de sistemas (*logs*) pode revelar incidentes ou outras anomalias, entretanto, fazer isto sem o auxílio de estruturas de programação é repetitivo, enfadonho e ineficaz (COSTA, 2007).

Para programar é necessário algum domínio sobre alguma das diversas linguagens de programação e, neste trabalho, adotamos a linguagem Python, cujas características são explicadas na próxima seção.

1.4.1 Linguagem Python

Python é uma linguagem de programação de alto nível, de código aberto, com suporte aos paradigmas de programação estruturada e orientada a objetos, interpretada e em *scripts*, criada no início dos anos 1990, pelo holandês Guido Van Rossum (ROSSUM, 2000). Algumas das principais características de Python estão sumarizadas no Quadro 2.

Quadro 2: Características da linguagem de programação Python.

Característica	Definição
Código aberto (<i>open source</i>)	Permite que o código fonte de um <i>software</i> esteja disponível para qualquer interessado, podendo ser alterado gratuitamente, desde que a redistribuição siga os mesmos moldes do código original. (OPENSOURCE, 2007).
Alto nível	Linguagens de programação de alto nível são as mais próximas da linguagem humana, em um comparativo com a linguagem de máquina, que é classificada como o mais baixo nível e que interpreta apenas <i>zeros e uns</i> (TANENBAUM, 2007).
Suporte a POO diferentes paradigmas de programação	Suporta múltiplos paradigmas de programação, como POO (Programação Orientada a Objetos) e programação estruturada (CRUZ, 2015).
Interpretada	Enquanto as linguagens compiladas devem passar por um processo de conversão de seu código fonte para linguagem binária, para depois serem executadas pelo computador, a linguagem interpretada pode ser lida e executada diretamente pelo sistema operacional (COSTA, 2007).
Disposta em <i>scripts</i>	<i>Scripts</i> são programas compostos por sequências de instruções ou comandos, interpretados pelo computador (COSTA, 2007).

Fonte: Autoria própria, a partir das obras consultadas.

A linguagem Python é adotada em diversas aplicações como desenvolvimento de software, artes, negócios, educação, aplicações governamentais, ciências e engenharia. Destaca-se por possuir uma sintaxe simplificada; segurança na construção de códigos, em que qualquer violação produz mensagens de erro; interatividade, em que testes de codificação podem ser executados diretamente na interface de linha de comando, com verificação imediata de resultados; pode ser utilizada para propósitos gerais; gratuito e difundido em diversos segmentos, como indústria, comércio, educação, ciências, etc.; possui ampla comunidade de desenvolvedores; portátil para a maioria dos sistemas operacionais de diversos dispositivos, como computadores pessoais, celulares, sistemas embarcados em hardware, etc.; ampla biblioteca de módulos de comandos; admite a construção de programas mais curtos e legíveis em comparação com linguagem Java, C ou C++, pois permite que algumas operações complexas sejam escritas em menos sentenças; indentação substituindo *begin*, *end* e chaves, tornando o código mais legível e menos suscetível a erros de

estruturação; sem necessidade de declaração de argumentos nas variáveis; e, considerada de aprendizado fácil por diversos autores (BRIGGS, 2013; LAMBERT, 2012; GIFT; JONES, 2008; ROSSUM, 2000).

Python também pode ser aplicada em soluções para Redes de Computadores, como automação de rotinas integradas ao sistema operacional e monitoramento de redes (RHODES; GOERZEN, 2014; COSTA, 2007; SARKER; WASHINGTON, 2015; GIFT; JONES, 2008).

Além dos benefícios apresentados, a aprendizagem da linguagem Python por estudantes de redes de computadores traz benefícios agregados, pois pode ser utilizada para outros fins, tais como, plotar gráficos e desenvolver jogos. Todas estas características e diferenciais motivaram a escolha desta linguagem para uso em nossa proposta de ensino-aprendizagem, cujo planejamento foi orientado pelo Alinhamento Construtivo, conforme apresentado na seção seguinte.

1.5 ALINHAMENTO CONSTRUTIVO

O Alinhamento Construtivo é um conjunto de orientações sobre como planejar o ensino, centralizando-se nas atividades do estudante, fundamentado na Teoria do Currículo e no Construtivismo. O primeiro termo faz referência ao *alinhamento* que deve existir entre o que se objetiva como resultado da aprendizagem, as atividades de ensino e as avaliações. O termo *construtivo* remete ao construtivismo que enfatiza a participação ativa do estudante na construção do seu próprio conhecimento (BIGGS; TANG, 2011; MENDONÇA, 2015).

Para que haja alinhamento entre os resultados pretendidos da aprendizagem, atividades de ensino, atividades de aprendizagem e avaliações é necessário que os estudantes se engajem em atividades que os permitam atingir os resultados pretendidos da aprendizagem. O professor, por sua vez, deve proporcionar por meio das atividades de ensino, as condições para que os estudantes possam ser autônomos na construção de seu próprio conhecimento.

O Alinhamento Construtivo apoia-se em três pilares principais: Resultados Pretendidos da Aprendizagem – RPA; Atividades de Ensino e Aprendizagem - AEA; e, Tarefas de Avaliação – TA.

De acordo com Biggs e Tang (2011), os Resultados Pretendidos da Aprendizagem (RPA) devem expressar aquilo que pretendemos que os estudantes alcancem após ter participado de um processo de ensino. Devem ser expressos por meio de *verbos* que esclareçam a ação que é requerida dos estudantes, como por exemplo, “explicar”, “aplicar”,

“comparar”. Desta forma, os RPA são declarados observando a perspectiva do estudante.

As Atividades de Ensino e Aprendizagem (AEA), por sua vez, especificam o que o professor e os estudantes devem fazer. Nesta etapa, o professor deve concentrar-se em *o quê* o estudante deve fazer para alcançar os RPAs, previamente definidos.

As tarefas de avaliação (TA) devem ser planejadas de modo que permitam informar quão bem os estudantes alcançaram os RPA. Há vários instrumentos de avaliação possíveis, como por exemplo, testes, provas e exercícios e é fundamental que as avaliações sejam alinhadas, isto é, mantenham coerência com os RPA e as AEA. Segundo Brabrand e Dahl (2007), um planejamento de ensino está alinhado construtivamente quando:

- Os RPA são explicitamente formulados como ações;
- Os RPA são explicitamente comunicados aos estudantes;
- As AEA são consistentes/compatíveis com os RPA;
- As TA mensuram o alcance dos RPA.

Com respeito a este último item, Biggs e Tang (2011) chamam atenção para a necessidade de estabelecer, para as avaliações, critérios capazes de prover *feedback* para os estudantes sobre quais aspectos devem melhorar. Para a definição desses critérios, adotamos neste trabalho, rubricas, conforme apresentado a seguir.

1.6 AVALIAÇÃO POR RUBRICAS

Segundo Brookhart (2013), uma rubrica (do inglês, *rubrics*) é um conjunto coerente de critérios para o trabalho dos estudantes que inclui nestes critérios descrições de níveis de qualidade de desempenho. As rubricas são importantes porque elas deixam claro para os estudantes a qualidade que seus trabalhos devem ter, permitindo a eles entendimento sobre o objetivo da aprendizagem e critérios para o sucesso.

Conforme Mendonça e Coelho (2018), rubricas funcionam bem quando o objeto a ser avaliado compreende algo que o estudante tenha de produzir, refletindo mais claramente seu desempenho. Neste sentido, como nosso objeto de avaliação engloba a capacidade de análise de um problema, sua resolução e, posteriormente, associação da solução encontrada com soluções de outros problemas, a adoção de rubricas é bastante coerente.

As rubricas devem ser condizentes com os RPA, estabelecidos previamente. Para a formação de uma rubrica devem ser considerados três componentes: *critérios*, *níveis de desempenho* e *descrições*. Estes componentes são apresentados no Quadro 3 em um exemplo

de rubricas para análise de uma tarefa de apresentação oral.

Quadro 3: Componentes de uma rubrica (apresentação oral).

Níveis de Desempenho			
Critérios	Nível alto	Nível médio	Nível iniciante
	A apresentação é criativa e eficaz na transmissão das ideias para o público. O apresentador responde eficazmente às reações e perguntas do público.	As técnicas de apresentação utilizadas são eficazes na transmissão das ideias principais, mas possuem pouca ou nenhuma criatividade. Algumas perguntas do público permanecem sem resposta.	A apresentação falha em capturar o interesse do público e/ou é confusa quanto ao que deve ser comunicado.
Uso de recursos Visuais	A apresentação inclui recursos visuais apropriados e de fácil compreensão. Estes recursos são utilizados pelo apresentador para explicar informações em momentos apropriados na apresentação.	A apresentação inclui recursos visuais adequados, mas estes são muito poucos, estão em um formato que os torna difíceis de usar ou entender, ou o apresentador não se refere a eles ou os explica na apresentação.	A apresentação não inclui recursos visuais ou inclui recursos visuais inadequados ou muito pequenos ou difíceis para serem compreendidos. O apresentador não faz nenhuma menção a eles na apresentação.

Fonte: Adaptado de Stevens e Levi (2005 *apud* MENDONÇA; COELHO, 2018).

Os *critérios* devem levar em conta as características do trabalho executado pelo estudante, de forma a evidenciar o seu aprendizado, o seu conhecimento e/ou as habilidades envolvidas. Ex.: os critérios “Comunicação” e “Uso de recursos visuais” para avaliação de uma apresentação oral. Os *níveis de desempenho* utilizados na formação da rubrica devem ser dispostos de forma que possam demonstrar pontos significativos que diferenciam o enquadramento das tarefas realizadas pelo estudante. Esta diferenciação é determinante para mensuração da quantidade de níveis utilizados em cada critério. Ex.1: “Nível alto”, “Nível médio”, “Nível iniciante”; Ex.2: “Avançado”, “Proficiente”, “Iniciante”, “Insuficiente”. A determinação da *descrição* se dá em função da determinação dos critérios e dos níveis de desempenho, pois para cada conjunto de critério/nível haverá uma descrição distinta que refletirá as características de enquadramento. Ex.: Para uma apresentação oral, considerando o critério “Comunicação” e o nível de desempenho “Nível Alto”, a descrição utilizada poderia ser “A apresentação é criativa e eficaz na transmissão das ideias para o público. O apresentador responde eficazmente às reações e perguntas do público.” Já descrição do “Nível Médio” para o mesmo critério seria “As técnicas de apresentação utilizadas são eficazes na

transmissão das ideias principais, mas possuem pouca ou nenhuma criatividade. Algumas perguntas do público permanecem sem resposta.” É possível observar que entre os dois níveis exemplificados há uma ordem decrescente na qualidade de desempenho anotada na descrição. Este decréscimo se sucederá até o nível mais inferior (MENDONÇA; COELHO, 2018).

Segundo Bender (2014), rubricas podem ser do tipo *holística* ou *analítica*, que se diferenciam pela forma de avaliação dos critérios adotados. Enquanto que na rubrica do tipo holística a avaliação é feita de forma mais geral e abrange o conjunto dos critérios, na rubrica do tipo analítica são avaliados vários critérios separadamente. O exemplo de rubrica apresentado no Quadro 3 é do tipo analítica e este é o tipo de rubrica que adotamos para nossa proposta de ensino-aprendizagem, pois além de sua clareza em diferenciar os níveis de desempenho em função dos critérios para cada habilidade analisada, também facilita o *feedback* aos estudantes, possibilitando que o estudante reflita de forma individualizada sobre seu desempenho em cada atividade.

Na próxima seção apresentamos os trabalhos relacionados ao ensino de Programação para administração de Redes de Computadores.

1.7 TRABALHOS RELACIONADOS

Nesta seção apresentamos os trabalhos que se relacionam com nossa pesquisa envolvendo Programação de Computadores na perspectiva de Redes de Computadores. Também apresentamos trabalhos que se assemelham quanto a alguns métodos de ensino que adotamos.

Para buscarmos propostas em cursos de graduação, fizemos uma pesquisa *ad-hoc* por cursos de CST em Tecnologia do eixo Informação e Comunicação dos Institutos Federais de Educação (conforme APÊNDICE A). Esta delimitação se deu, primeiramente, por ser este Mestrado Profissional em Ensino Tecnológico ministrado por um Instituto Federal de Educação (IFAM), por estarmos inseridos como professores do Ensino Básico, Técnico e Tecnológico (EBTT) e por nossa proposta de ensino aprendizagem ser baseada na prática como forma de impulsionar o aprendizado, sem se distanciar de uma visão crítica sobre sua atuação, assim, se relacionando com os propósitos dos CST e, conseqüentemente, com os Institutos Federais de Educação.

O período de levantamento foi entre os meses de março e abril de 2017 e nossa estratégia de pesquisa foi: Consulta no site e-Mec, do Ministério da Educação, por Institutos

de Ensino Superior (IES), do Sistema Federal de Ensino, que resultou em uma lista de Institutos Federais (IF) e Centro Federal de Educação Tecnológica (CEFET), sobre a qual realizamos buscas em seus sites por Cursos Superiores em Tecnologias do eixo Informação e Comunicação. Nestes sites fizemos buscas por detalhes de disciplinas de Redes de Computadores e de Programação em 40 (quarenta) Institutos Federais de Educação, com 60 (sessenta) CST.

Destes cursos, identificamos apenas nove que, pela nomenclatura encontrada, poderiam se assemelhar com nossa proposta de ensino-aprendizagem, sendo que seis destes são de CST em Redes de Computadores e os outros três se distribuem entre CST em Análise e Desenvolvimento de Sistemas, CST em Gestão da Informação e CST em Sistemas para Internet. Dada a natureza do CST em Redes de Computadores, não seria incomum encontrar tais disciplinas com estas características. Por outro lado, apenas três dos outros cinquenta e três CST observados apresentaram disciplinas com a abordagem relacionada à nossa proposta de ensino-aprendizagem. Os nove cursos e seus respectivos Institutos Federais são descritos no Quadro 4.

Neste levantamento, por conta da forma como são disponibilizadas as informações, o que pudemos identificar como semelhante foi apenas a iniciativa de unir as habilidades de programação às habilidades relacionadas às Redes de Computadores, pois o que encontramos foram apenas as grades curriculares contendo os nomes das disciplinas e alguns planos de cursos. Por encontrarmos pouco detalhamento nos poucos planos de cursos ou poucas informações disponibilizadas, não nos é possível afirmar como estas disciplinas ou os conteúdos são aplicados. De uma forma geral, de acordo com os planos de ensino encontrados, as disciplinas seguem o modelo tradicional de ensino, pois são mencionadas apenas aulas expositivas e poucas referências a exercícios em laboratórios, sem qualquer alusão a estudos envolvendo ensino híbrido, omitindo a presença de conteúdos práticos sobre Programação para resolução de problemas relacionados ao domínio de Redes de Computadores.

Estas poucas disciplinas identificadas, apesar de aparentarem valorizar a utilização de recursos de Programação para resolução de problemas do domínio de Redes de Computadores, pelo nome que lhes é atribuído, não fornecem informações detalhadas sobre suas metodologias, organização, dinâmicas e instrumentos utilizados, dificultando a reprodução por terceiros destas mesmas experiências.

Neste aspecto, há um diferencial em nosso trabalho: disponibilizaremos toda a metodologia e os materiais utilizados, na forma de um produto (guia para professores) para

auxiliar professores de Redes de Computadores que atuam em CST que queiram reproduzir nossa aplicação possam fazê-lo.

Quadro 4: IFs, CST e disciplinas observadas.

Insti-tuição	CST	Disciplina/semestre	Observações
IFCa-tari-nense	Redes de Comp.	Programação de <i>Scripts</i> . 6º sem.	Traz uma breve descrição do curso, sem muitos detalhes, mas pela bibliografia, denota-se que se direciona mais a <i>scripts</i> em <i>shell</i> Linux do que para outros tipos de programação, entretanto, não há maiores informações sobre a metodologia da disciplina.
IFPB	Redes de Comp.	Programação de <i>Scripts</i> . 4º sem.	Os <i>links</i> presentes no site estavam indisponíveis durante o período de tempo pesquisado.
IFMT	Redes de Comp.	Programação aplicada à gerência de redes. 3º sem.	O único documento disponível sobre o curso é a matriz curricular, pelo qual não é possível obter maiores informações sobre a disciplina.
IFMS	An. e Desenv. de Sistemas	Laboratório de Redes de Computadores. 4º sem.	Na disciplina há um componente chamado Programação em <i>scripts</i> para servidores de redes, entretanto, não há nenhuma informação sobre a aplicação deste componente. Na bibliografia não há obras que envolvam programação.
IFCE	Redes de Comp.	Programação em ambiente de redes. 3º sem.	A ementa da disciplina descreve programação com <i>scripts</i> para Linux e Windows, inclusive com proposta de automatização de tarefas. Não há maiores detalhes. Na bibliografia há obras que tratam apenas de programação em C, programação com <i>sockets</i> e programação com <i>scripts</i> XML e WMI para Windows, direcionados a bancos de dados.
IFRN	Redes de Comp.	Programação para redes. 2º sem.	A disciplina trata de quatro assuntos: Aplicações com <i>sockets</i> ; Bancos de dados; <i>Webservices</i> ; e Automatização de tarefas via <i>scripts</i> . Não há detalhes sobre como os assuntos serão tratados dentro da disciplina. Na bibliografia há obras que tratam apenas de programação em C, programação com <i>sockets</i> e programação com <i>scripts</i> XML e WMI para Windows, direcionados a bancos de dados.
IFRN	Sistemas para Internet	Programação em ambientes de redes. 3º sem.	A ementa da disciplina menciona “Técnicas de programação em redes”, entretanto, não há informações sobre a aplicação deste conteúdo. Pela ementa, parece tratar de programação com <i>sockets</i> . A bibliografia traz obras sobre programação em C e <i>sockets</i> .
IFSul de Minas	Redes de Comp.	Programação de <i>Scripts</i> . 3º sem.	A disciplina é direcionada ao desenvolvimento de programas com <i>scripts</i> em <i>shell</i> para Windows e Linux, entretanto não há maiores detalhes sobre sua aplicação. A bibliografia se concentra em programação em <i>scripts shell</i> Linux.
IFGoi-ano	Gestão da Tecn. da Informação	Gestão de sistemas operacionais. 3º sem.	A disciplina possui um componente de Programação em <i>shell script</i> , entretanto não oferece detalhes sobre o desenvolvimento do conteúdo. A bibliografia também não menciona obras que tratem de programação.

Fonte: Autoria própria, a partir de pesquisas nos sites das instituições pesquisadas.

Em nossa proposta de ensino-aprendizagem defendemos que a prática da resolução de problemas de Redes de Computadores deva ser inserida logo nos primeiros semestres e, se possível, inserida na primeira (ou única, em alguns casos) disciplina de Redes de Computadores, independente do foco do CST na área de computação. Isto proporcionará ao estudante um conhecimento diferenciado em relação a Redes de Computadores. Em uma busca no site de anúncios de empregos Catho⁸, com filtro entre os dias 11/09/2018 e

⁸ <https://www.catho.com.br/>

17/09/2018 pela vaga de “Analista de Suporte Técnico”, 35% das vagas exigiam conhecimentos de administração de Redes de Computadores cujos problemas inerentes pressupõem serem solucionados ou facilitados através de recursos de Programação (conforme APÊNDICE B).

Nas bibliografias estudadas, destacamos três obras. A primeira é uma publicação estadunidense intitulada *Python for Unix and Linux System Administration*, de Gift e Jones (2008). Esta obra é direcionada à administração de sistemas e aborda vários conteúdos que envolvem recursos de Programação resolução de problemas inerentes às Redes de Computadores, como automatização de tarefas envolvendo serviços em rede como: monitoramento via SNMP, execução e controle de *backups* em rede, compactação e empacotamento de arquivos e diretórios, manipulação de arquivos, compartilhamento de dados, sincronismo de dados, agendamento de processos, envio de *e-mails*, entre outros. Também utiliza a linguagem Python, com a variante IPython, que é um interpretador interativo para o Python. Apesar de concordarmos com a eficiência do IPython, os seus códigos são gerados a partir de interface *web* e resultam em uma mistura de scripts HTML com *scripts* Python. Inicialmente esta variante foi cogitada para utilização em nossa proposta de ensino-aprendizagem, mas devido ao seu código ficar muito extenso em virtude do entrelaçamento dos *scripts* HTML e Python, optamos por não utilizá-la, pois nosso intuito é de que o estudante possa criar facilmente códigos em Python para resolver problemas de redes de computadores. Como nossa proposta valoriza a criação manual, via linha de comando, a construção do código em IPython seria bem mais complexa, comparado à codificação em Python “convencional”.

A segunda obra em destaque também é estadunidense, intitulada *Essential System Administration*, de Frisch (2002). Nesta obra a programação é direcionada para Unix Shell Script, uma linguagem de programação nativa dos sistemas operacionais Linux. Envolve a automatização de: monitoramento de redes e de componentes remotos com SNMP, controle de acessos de usuários, envio de *e-mails*, *backups* em rede, entre outros serviços de rede.

A linguagem Linux Shell Scripts também foi cogitada em nossos primeiros testes, entretanto, trata-se de uma linguagem específica para o ambiente Unix, não agregando tanto valor quando o estudante fosse defrontado com problemas em outros sistemas operacionais ou outros problemas que exigissem a construção de um programa para outros propósitos.

A terceira obra é brasileira, intitulada *Administração de Redes com Scripts: bash script, python e vbscript*, de Costa (2007) e, como o próprio título apresenta, utiliza linguagens em scripts para administração de redes em Linux Shell Scripts (bash), Python e

VBScript (Linguagem de *scripts* da Microsoft). Apesar de apresentar poucos exemplos, utiliza problemas que simulam contextos reais de Redes de Computadores que podem ser automatizados com recursos de Programação. Trata da automatização de: monitoramento de disponibilidade de equipamentos através do PING, manipulação de *strings*, envio de *e-mails*, *backups* em rede, compactação e empacotamento de arquivos e diretórios, compartilhamento de dados em rede, controle de usuários.

Em nossa busca por outros trabalhos que adotassem recursos de Programação para resolução de problemas de Redes de Computadores logamos pouco êxito, entretanto relatamos alguns trabalhos que utilizam algumas outras abordagens que também adotamos, como sala de aula invertida, alinhamento construtivo e aprendizagem utilizando problemas em disciplinas de cursos da área de computação.

Chen (2014b) aplicou o ensino híbrido de sala de aula invertida, apoiado pelos preceitos do Alinhamento Construtivo, em um curso superior para uma disciplina de programação com linguagem C, intitulado *Flipped Classroom Model and Its Implementation in a Computer Programming Course*, em Lärarlärdom, Suécia. As avaliações aplicadas foram utilizadas para fornecimento de *feedback* imediato aos estudantes para melhoria dos RPA. Na discussão dos resultados obtidos, o autor informou que obteve aprovação de 71% dos estudantes sobre o método de sala de aula invertida. 100% dos estudantes foram aprovados ao longo das avaliações aplicadas. Também foram comprovados indícios de que o método melhorou o engajamento dos estudantes na construção do seu próprio aprendizado.

Leite, Hoji e Abdalla Junior (2018) propuseram a modificação da disciplina de Comunicação em Redes de Computadores (do Curso Técnico de Nível Médio de Manutenção e Suporte à Informática oferecido pelo Instituto Federal de Brasília) através da adoção do modelo de Sala de Aula Invertida associado à Aprendizagem Baseada em Problemas, por conta dos altos índices de evasões. Utilizaram o Moodle como ambiente virtual para compartilhamento, repositório de conteúdos e de recursos didáticos. Também utilizaram o Whatsapp para comunicação entre os envolvidos, Youtube para disponibilização dos vídeos e Google Apps para apoio à construção e disponibilização de materiais. A adoção da Sala de Aula Invertida e da Aprendizagem Baseada em Problemas foram implantadas desde 2013 e a observação dos resultados se deu até 2016. O índice de aprovação que era de 47% atingiu os 81% entre 2013 e 2014 e 84% entre 2014 e 2016. A média geral também ascendeu de 4,9 em 2013 para 7,1 e 7,7 nos respectivos períodos citados, demonstrando a eficácia da aplicação.

O próximo capítulo apresenta os detalhes sobre nossa proposta de ensino-aprendizagem de Programação para Administração de Redes de Computadores.

2 PROPOSTA DE ENSINO-APRENDIZAGEM DE PROGRAMAÇÃO PARA ADMINISTRAÇÃO DE REDES DE COMPUTADORES

Neste capítulo apresentamos nossa proposta de ensino-aprendizagem de Programação para Administração de Redes de Computadores que seguiu a metodologia do ensino híbrido de Sala de Aula Invertida, auxiliada pela resolução de problemas com atividades “mão na massa”.

A composição desta proposta está distribuída nas seções que se seguem. Na primeira seção apresentamos a forma como foram concebidos os conteúdos, módulos, temas e problemas propostos. Na segunda seção apresentamos os Roteiros de Aprendizagem e destacamos sua importância para a evolução da proposta de ensino-aprendizagem. A terceira seção é destinada a apresentar o Laboratório Virtual, essencial para a resolução das tarefas propostas. A quarta seção é destinada à apresentação do ambiente virtual de aprendizagem e demais recursos digitais que auxiliaram na disponibilização dos materiais didáticos e na comunicação com os estudantes. Na quinta seção apresentamos a forma que avaliamos as atividades de aprendizagem e a definição e aplicação das rubricas.

2.1 CONTEÚDOS, MÓDULOS, TEMAS E PROBLEMAS

Para a composição dos módulos, temas e problemas que integram nossa proposta precisamos definir aquilo que queríamos que os estudantes fossem capazes de fazer ao concluírem todos os estudos propostos. Assim, definimos os RPA em consonância com as habilidades que pretendíamos que os estudantes alcancem. Embora possuam variações ao longo dos instrumentos didáticos utilizados em cada etapa da aplicação da proposta, de maneira geral, concentram-se em três RPA:

- i. *Identificar uma solução possível para um problema* - consiste, primeiramente, em compreender o problema encontrado e quais suas características. Depois deste entendimento o estudante deve descrever como resolverá o problema, elencando quais recursos de Redes de Computadores e de Programação serão utilizados;
- ii. *Automatizar a solução identificada* - é uma habilidade dependente da anterior, pois refletirá a sua compreensão assertiva. Consiste em construir uma solução na linguagem Python baseada na descrição produzida;

- iii. *Associar a solução do problema com possíveis soluções de outros problemas no domínio de Redes de Computadores* - é uma habilidade que objetiva a independência do indivíduo quanto à resolução de problemas, pois com esta capacidade o estudante expande as possibilidades de resolução de problemas diversos e, em uma ação em cadeia, a cada nova solução haverá uma nova associação e, conseqüentemente, o aprimoramento desta habilidade com projeções de soluções virtualmente ilimitadas.

Estando definidos os RPA, definimos então os conteúdos referentes a Redes de Computadores e, para tanto, realizamos uma pesquisa em livros nacionais e internacionais que versam sobre administração de redes de computadores e que envolvem alguma linguagem de programação como Perl, Linux Shell Scripts, Windows PowerShell e Python, conforme apresentado no Quadro 5.

Quadro 5: Relação livro x conteúdo

Relação Livro x Conteúdo		Conteúdo														
Livro	Autoria	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1 Administração de redes com scripts	(COSTA, 2007)															
2 Automating Linux and Unix System Administration	(CAMPI; BAUER, 2009)															
3 Automating System Administration with Perl	(BLANK-EDELMAN, 2009)															
4 Automating Windows Administration	(BORGE, 2004)															
5 Automating Windows Server 2008 Administration with Windows PowerShell	(MICROSOFT, 2008)															
6 Core Python Applications Programming	(CHUN, 2012)															
7 Essential System Administration-Tools and Techniques for Linux and Unix	(FRISCH, 2002)															
8 Foundations of Python Network Programming	(RHODES; GOERZEN, 2014)															
9 Learning Python Network Programming	(SARKER; WASHINGTON, 2015)															
10 Linux Command Line and Shell Scripting Bible	(BLUM; BRESNAHAN, 2015)															
11 Linux System Administration Recipes	(KEMP, 2009)															
12 Pro Python System Administration	(SILEIKA, 2014)															
13 Python for Unix and Linux System Administration	(GIFT; JONES, 2008)															
14 Shell Scripting Recipes	(JOHNSON; VARMA, 2015)															
15 Unix and Linux System Administration Handbook	(NEMETH et al., 2011)															
16 Wicked Cool Shell Scripts – 101 scripts	(TAYLOR, 2004)															
Total de ocorrências		9	8	8	8	8	7	7	7	6	6	4	4	3	3	3

Legenda dos conteúdos: 1-Monitoramento de Redes; 2-Compartilhamento de dados; 3-Envio de e-mails/relatório; 4-Gerenciamento de sistemas de arquivos; 5-Gerenciamento de usuários/acesso; 6-Login; 7-Monitoramento de sistemas; 8-Gerenciamento de Arquivos e Diretórios; 9-Backup; 10-SSH/acesso remoto; 11-Análise de logs; 12-Compressão de dados; 13-Authenticação segura; 14-Expressões regulares; 15-Manipulação de strings.

Fonte: Autoria Própria.

Após esta seleção de conteúdos, iniciamos a composição dos problemas que seriam utilizados ao longo da aplicação da proposta que foram selecionados e organizados de forma que os estudantes pudessem acumular os conhecimentos ao longo de suas resoluções e que proporcionassem uma evolução gradativa em função da consecução dos temas. Antes de adicionar os problemas aos Roteiros de Aprendizagem desenvolvidos, os mesmos foram cuidadosamente resolvidos para que pudessem ser identificadas todas as estruturas de programação e requisitos de sistema operacional.

Para determinação dos conteúdos de programação, foram priorizados problemas cujas soluções fossem mais simples e que envolvessem estruturas de programação menos complexas. Em decorrência disto, foram elencadas os seguintes conteúdos: bibliotecas e seus métodos; variáveis; tipos de dados; conversão de tipos de dados; comentários como forma de documentação; entrada e saída de dados; coletas de estados de execução de comandos; operações sobre arquivos; manipulação de datas; manipulação de *strings*; laços de repetição; estruturas de decisão.

Como a administração de Redes de Computadores envolve conhecimentos do sistema operacional utilizado, também houve a necessidade de determinarmos seus conteúdos envolvidos. Para esta determinação testamos as soluções de cada um dos problemas previamente no laboratório virtual, de forma que tivéssemos certeza dos recursos necessários. Este procedimento também auxiliou na determinação das configurações de cada máquina que compuseram o laboratório virtual. Desta forma, os conteúdos referentes ao sistema operacional foram: estruturas de arquivos e diretórios; comandos do sistema operacional; monitoramento de CPU; discos e partições; compartilhamento de arquivos e diretórios; editores de textos; acesso remoto; arquivos de configuração; hierarquia de usuários; histórico de ações, data e hora; empacotamento e compactação de dados; montagem de partições remotas⁹; agendamento de execução de programas ou comandos; arquivos de registros (*logs*); sincronismo de data e hora; sincronismo de dados; envio de *e-mails*, envio de mensagens instantâneas, configuração de aplicativos.

Determinados os conteúdos e com um dimensionamento mais claro sobre o que seria abordado na proposta de ensino-aprendizagem, procedemos à composição dos módulos e dos temas a serem abordados.

Para uma melhor estruturação da proposta de ensino-aprendizagem, devido às características a serem agrupadas, dividimo-la em dois módulos: o primeiro, abrangendo assuntos referentes ao *monitoramento de redes e de ações de usuários*; o segundo, integrando assuntos sobre a *preservação das informações e comunicação de eventos*. Cada assunto dentro dos módulos está associado a um tema, cujo desdobramento se dá pela integração de conteúdos sobre Redes de Computadores, sistemas operacionais Linux e Programação, conforme pode ser observado no Quadro 6.

⁹ Um servidor pode escrever em outro, desde que haja permissões para isso. Por exemplo, durante a execução de um backup com gravação de dados em outro computador, uma partição remota pode ser montada, o backup é executado e os dados transferidos para o computador destino. Em seguida, esta partição é desmontada no servidor de origem, de forma que o vínculo entre eles seja interrompido, por razões de segurança.

Além disso, para cada assunto há um conjunto de problemas propostos, nos quais os estudantes devem integrar teoria e prática, identificando e automatizando soluções para, posteriormente, associá-las com a resolução de outros problemas.

Quadro 6: Módulos, temas, assuntos e conteúdos.

Módulo 1 - Monitoramento de Redes de Computadores
Tema 01: Identificando a movimentação da vizinhança Assunto: Monitoramento de disponibilidade de equipamentos e serviços em rede. Conteúdos principais: <u>Redes:</u> Comando “ping”; Protocolo SNMP. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados. <u>S.O. Linux:</u> Estruturas de arquivos e diretórios; - Monitoramento de carga e de estados de componentes. Os problemas propostos envolvem: Detecções de problemas de lentidão ou perda de acesso, através de monitoramentos via SNMP e “ping”, relacionados ao estado de máquinas e componentes.
Tema 02: Cuidando do Habitat Assunto: Monitoramento de estado de discos e partições, em rede. Conteúdos principais: <u>Redes:</u> Protocolo NFS; Protocolo SNMP. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados. <u>S.O. Linux:</u> Compartilhamento de dados; Estruturas de arquivos e diretórios; Os problemas propostos envolvem: Monitoramentos remotos de discos e partições como prevenção de estouros de espaços de armazenamento.
Tema 03: De olho nas intenções Assunto: Monitoramento de ações de usuários. Conteúdos principais: <u>Redes:</u> Protocolo SSH, Acesso remoto; <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; Operações sobre arquivos; Operações com <i>strings</i> e datas; Laços; Decisão. <u>S.O. Linux:</u> Editores de textos; Acesso remoto; Arquivos de configuração; Hierarquia de usuários; Históricos de ações; Data e hora; Os problemas propostos envolvem: Monitoramentos de ações de usuários através de verificação automatizada de <i>logs</i> do sistema operacional.
Módulo 2 - Preservação de informações e comunicação de eventos
Tema 04: Construir é difícil. Reconstruir pode ser impossível Assunto: Compactação de dados, cópias de segurança em rede e registros de atividades (<i>logs</i>). Conteúdos principais: <u>Redes:</u> <i>Backup</i> ; Compartilhamento de arquivos; Protocolo NTP; Protocolo NFS. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; Operações com <i>strings</i> e datas; Laços; Decisão. <u>S.O. Linux:</u> Empacotamento e compactação de dados; Agendamento de tarefas; Sincronismo de data e hora; Arquivos de <i>logs</i> . Os problemas propostos envolvem: Agendamentos de execuções de <i>backup</i> em rede, com verificação de etapas de execução e gravação em arquivos de <i>log</i> .
Tema 05: Um é pouco. Então tenha dois (ou mais!) Assunto: Sincronismo de dados em rede. Conteúdos principais: <u>Redes:</u> <i>Backup</i> ; Políticas de <i>backup</i> ; Sincronismo de dados remotos; Compartilhamento de arquivos; Protocolo SSH; Protocolos NFS. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; Operações com <i>strings</i> e datas; Decisão. <u>S.O. Linux:</u> Sincronismo de dados; Agendamento de tarefas; Arquivos de <i>logs</i> . Os problemas propostos envolvem: Agendamentos de execuções de sincronismo de dados em rede, com verificação de etapas de execução e gravação em arquivos de <i>log</i> .
Tema 06: Seja o primeiro a saber Assunto: Automatização de mensagens de eventos e de alertas via <i>e-mail</i> e mensageria instantânea (Telegram). Conteúdos principais: <u>Redes:</u> Serviços de <i>e-mail</i> ; Protocolo SMTP; Protocolos SSL/TLS e StartTLS. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; Envio de <i>e-mails</i> ; Operações com <i>strings</i> . <u>S.O. Linux:</u> Envios de <i>e-mails</i> ; Envio de mensageria instantânea; Sincronismo de dados; Agendamento de tarefas; Configuração de serviços. Os problemas propostos envolvem: Automatizações de procedimentos com verificações de etapas de execuções e, em função dos resultados, envio de avisos e alertas via <i>e-mail</i> e/ou Telegram.

Fonte: Autoria Própria.

Cada um destes problemas foi elaborado em função dos conteúdos e temas de cada

módulo. Por exemplo, o Módulo 1 configura o ingresso (primeiro contato) aos conhecimentos abordados e, portanto, detêm a introdução aos conteúdos. Assim, o primeiro tema apresenta ao estudante uma fase inicial, abordando a base para tudo o que será desenvolvido dali por diante. Inicia com as primeiras noções sobre monitoramento de redes, envolvendo os protocolos SNMP (Protocolo para gerenciamento de dispositivos em redes IP) e ICMP (Protocolo para controle de mensagens entre dispositivos em Redes IP). Da mesma forma, são introduzidas as principais bibliotecas que serão utilizadas ao longo da aplicação da proposta de ensino-aprendizagem. Também configura o primeiro contato com o laboratório virtual e introduz o entendimento sobre a comunicação entre as máquinas que o compõem. O segundo tema complementa o primeiro, pois envolve, além dos mesmos conteúdos trabalhados no tema anterior (Redes de Computadores, Programação e Sistema Operacional), desafios adicionais envolvendo outros objetos de monitoramento. Esta evolução ocorre em todos os temas que se seguem, inclusive no Módulo 2, conduzida pelos problemas apresentados.

Os problemas são propostos apenas nos Estudos Presenciais, entretanto, a preparação para suas resoluções ocorrem nos Estudos Autônomos.

Como forma de direcionar os esforços do estudante em cada momento autônomo ou presencial de seus estudos, após definidos os módulos, temas e problemas, foram criados os Roteiros de Aprendizagem para os Estudos Autônomos e Estudos Presenciais, os quais apresentamos na próxima seção.

2.2 ROTEIROS DE APRENDIZAGEM PARA ESTUDO AUTÔNOMO E ESTUDO PRESENCIAL

Para guiar o estudante tanto no Estudo Presencial quanto no Estudo Autônomo, foram criados Roteiros de Aprendizagem (RA) que são instrumentos que o conduzem por um itinerário de estudos a fim de que ele atinja os RPA.

Conforme visto no Quadro 6, p. 42, foram elaborados 6 (seis) temas. Para cada um destes temas foram destinados dois RA, sendo um para Estudo Autônomo e outro para Estudo Presencial, totalizando 12 (doze) RA.

Os RA são semelhantes em sua estrutura geral por possuírem as mesmas seções, entretanto, com finalidades diferentes. A principal diferença reside na seção de Tarefas. Nos RA para Estudos Autônomos, as tarefas são compostas por questões envolvendo conceitos e também pequenas aplicações práticas em um laboratório virtual, criado especificamente para

dar suporte à proposta de ensino-aprendizagem. Para apoiar os estudantes na resolução das tarefas são indicados, nos próprios RA, videoaulas, tutoriais *online* e *links* de páginas web. Nos RA para Estudo Presencial, as tarefas foram inspiradas em Bender (2014) e em Biggs e Tang (2011) e são constituídas da seguinte dinâmica: (i) *aquecimento*: um tempo é destinado para que os estudantes exponham suas dúvidas sobre o Estudo Autônomo, em uma discussão mediada pelo professor; (ii) *apresentação do contexto*: é apresentado um contexto envolvendo um problema real na área de redes de computadores; (iii) *discussão*: são apresentadas questões que instigam o estudante a compreender melhor o problema e encontrar meios para solucioná-lo; (iv) *resolução*: é solicitado dos estudantes a implementação de um programa, em linguagem Python, para solucionar o problema proposto, executando-o no laboratório virtual.

Um resumo sobre a estrutura geral dos RA é descrito no Quadro 7 e a íntegra de todos os RA está disposta no APÊNDICE C.

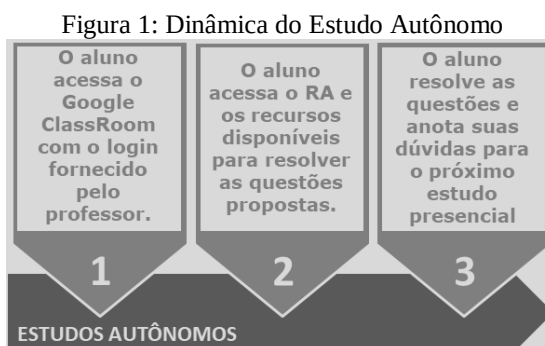
Quadro 7: Estrutura geral dos Roteiros de Aprendizagem (RA).

Cabeçalho
Identifica o módulo corrente, o número do roteiro, o tema abordado, o tipo de estudo (presencial ou autônomo) e a carga horária prevista.
Descrição geral
Apresenta orientações sobre como seguir o RA.
Resultados pretendidos da aprendizagem
Descreve os objetivos da aprendizagem a serem atingidos pelos estudantes.
Conteúdos envolvidos
Apresenta os conteúdos abordados, referentes a Redes de Computadores, Programação e Sistemas Operacionais (Linux).
Tarefas
Nos Estudos Autônomos , orientam a evolução do estudante sobre os conteúdos preparatórios para a resolução dos problemas que serão apresentados nos Estudos Presenciais. Tais tarefas são dotadas de questões cujas resoluções são guiadas por videoaulas, tutoriais <i>online</i> e <i>links</i> de páginas <i>web</i> para a aplicação prática no laboratório virtual. Nos Estudos presenciais são compostas de dinâmicas para resolução de problemas, inspiradas em Bender (2014) e em Biggs e Tang (2011), divididas em: <u>Aquecimento</u> - Durante os primeiros 30 minutos de aula, os estudantes expõem suas dúvidas sobre o Estudo Autônomo, em uma discussão mediada pelo professor. <u>Apresentação do contexto</u> - É apresentado um contexto envolvendo um problema com características reais de Redes de Computadores. <u>Discussão</u> - São dispostas questões que instiguem o estudante a tentar entender melhor a solução do problema e fomentem uma discussão sobre formas de solucioná-lo. <u>Resolução</u> - O estudante descreve como será a solução para o problema e quais recursos de Programação utilizará. Depois constrói um programa em Python consonante com a descrição e o executa no laboratório virtual. <u>Reflexão e observação (Associação)</u> - São apresentados três novos contextos de problemas, dos quais apenas um deverá ser associado como possuindo mais semelhanças em sua resolução com o problema resolvido anteriormente. Em seguida o estudante descreve como será a resolução para o problema e quais recurso de Programação utilizará.

Fonte: Autoria Própria.

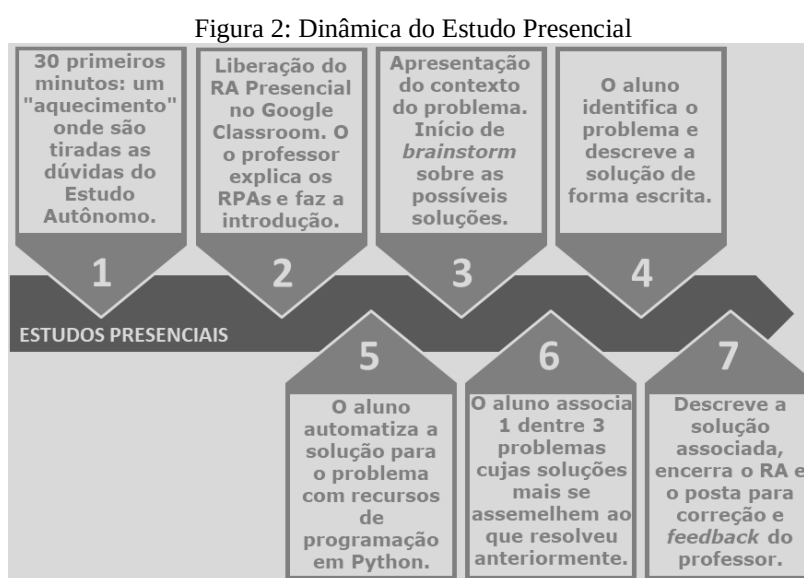
As tarefas também possuem dinâmicas distintas. Durante o Estudo Autônomo, o

estudante acessa os recursos disponibilizados (vídeos, *links*, etc.) que fornecem o conhecimento necessário para a resolução das questões propostas e faz anotações sobre suas dúvidas que deverão ser compartilhadas nos momentos iniciais (aquecimento) do próximo Estudo Presencial. A dinâmica utilizada para o Estudo Autônomo é ilustrada na Figura 1.



Fonte: Autoria própria.

O Estudo Presencial exige a aplicação dos conhecimentos praticados durante o Estudo Autônomo e será realizado em sala de aula, com a presença física de um professor. Sua dinâmica é descrita na Figura 2.



Fonte: Autoria Própria.

Inicia-se com a exposição das dúvidas anotadas pelos estudantes, compartilhamento e correção das respostas das questões resolvidas no Estudo Autônomo. Este momento ocorrerá durante os 30 (trinta) minutos iniciais do Estudo Presencial. Passado este *aquecimento*, inicia-se o momento “mão na massa”, no qual será apresentado aos estudantes o RA relativo ao Estudo Presencial, que conterà problemas relevantes que simulam alguns desafios cotidianos de um profissional que trabalha na área de Redes de Computadores. Cada problema proposto trará um contexto prático de redes para o qual o estudante deverá

identificar uma solução e desenvolver um programa na linguagem Python. A última tarefa traz três problemas cujas soluções devem ser analisadas pelo estudante no intuito de associar aquela cuja solução mais se assemelha ao programa que acabou de resolver.

Para que obtenha um bom desempenho no Estudo Presencial, o estudante deve executar as tarefas dispostas no respectivo Estudo Autônomo. Como isto depende da existência de um mesmo ambiente de Redes de Computadores tanto no Estudo Presencial quanto no Estudo Autônomo, foi necessário compor um laboratório virtual, sobre o qual discutiremos na seção a seguir.

2.3 LABORATÓRIO VIRTUAL

O laboratório virtual é um instrumento fundamental para o desenvolvimento dos estudos em nossa aplicação, pois é necessário para que o estudante execute as tarefas tanto nos Estudos Autônomos quanto nos Estudos Presenciais. Trata-se de uma composição com três máquinas virtuais dispostas em uma rede local, com características típicas de uma pequena rede de computadores, sobre as quais foram elaborados e testados os problemas e as questões que compõem os RA.

Para as aulas dos Estudos Presenciais, o estudante encontrava o ambiente virtual montado em cada máquina do laboratório físico, onde poderia praticar seus estudos e executar as tarefas. Para estudar em casa e resolver as questões dos Estudos Autônomos, o estudante deveria reproduzir o mesmo ambiente encontrado no laboratório físico. Dadas as características deste ambiente virtual e a necessidade de processamento, o estudante deveria ter disponível um computador de mesa (*desktop*) ou um *notebook*.

Inicialmente, estudamos a possibilidade de disponibilizarmos as máquinas virtuais aos estudantes a partir de um servidor centralizado, entretanto, as características que queríamos trabalhar em uma rede particular seriam afetadas caso criássemos uma máquina virtual para cada estudante e as dispuséssemos todas em uma rede. Isolar grupos com três máquinas em ambientes de rede seccionados logicamente seria possível, entretanto, inviável quanto à disposição de hardware, pois exigiria um servidor ou um conjunto de servidores que não nos eram disponíveis no momento. Desta forma, concluímos que o melhor modelo seria o que envolve a reprodução do laboratório virtual pelos estudantes em suas máquinas particulares.

Para facilitar a portabilidade deste ambiente aos estudantes, elaboramos um tutorial

para ajudá-los a reproduzir o laboratório virtual. Um estudo prévio foi feito para que este laboratório pudesse ser facilmente replicado pelo estudante em seu computador, de forma que fosse executado sem consumir muitos recursos como memória e processamento. Desta forma, optamos pelo Ubuntu Minimal por ser uma versão básica do Linux¹⁰ que possibilitava facilmente a instalação de quaisquer funcionalidades posteriores.

Para simplificar a reprodução do laboratório virtual, arquivos automatizados de configuração foram disponibilizados em uma máquina virtual “base” que disponibilizamos na página que criamos para o curso, no Google Classroom. O estudante faria o download da “base” e a importaria para o seu computador pessoal, dentro da plataforma de virtualização Oracle VM Virtualbox¹¹. A partir desta importação, o estudante executaria a clonagem da máquina base, obtendo os três servidores. Um destes servidores possuía interface gráfica e tinha instalado o ambiente Thonny¹² para desenvolver suas soluções na linguagem Python.

A composição deste laboratório virtual mereceu uma dedicação especial, pois era necessário compor um ambiente leve e eficiente e reproduzível. A ideia inicial foi de configurar as três máquinas virtuais e disponibilizar cópia dos seus arquivos no Google Drive para que os estudantes pudessem baixá-las e importá-las em seu ambiente virtualizado. Em um primeiro momento, pensamos em utilizar o Ubuntu em sua versão *desktop* ou *server* com interface gráfica, entretanto os testes resultaram em três máquinas com aproximadamente seis *gigabytes*, cada, que totalizavam 18 (dezoito) *gigabytes*, tornando-as inviáveis para *download*, principalmente em conexões lentas. Diante disto e depois de vários testes com alternativas que pudessem diminuir este tamanho, chegamos à composição atual que utiliza o Ubuntu Minimal em modo texto (linha de comando) em duas das máquinas virtuais, cada uma com um tamanho de 900 (novecentos) *megabytes*. Na terceira e principal máquina virtual que compõe o laboratório também instalamos o Ubuntu Minimal, entretanto, sobre este, instalamos a interface gráfica Lubuntu (Ligth Ubuntu). Isto resultou em uma máquina com aproximadamente três *gigabytes*. Ao todo o estudante teria de baixar quase cinco *gigabytes*. O tamanho estava menor, mas ainda grande para *download* em certas conexões.

¹⁰ O Linux é um sistema operacional bastante difundido, executado em mais de dois terços dos servidores *web*. Além de ser *software* livre (com código aberto, gratuito), possui vasto material *online* e comunidades ativas. A escolha pela distribuição Linux Ubuntu se deveu a sua popularidade, pois aproximadamente metade dos servidores Linux identificados na Internet utilizam esta distribuição (NETCRAFT, 2017).

¹¹ A plataforma de virtualização Oracle VM Virtualbox foi escolhida por ser um *software* livre, dotado de interface intuitiva e por possuir características como clonagem, importação e exportação de máquinas virtuais, *snapshots* de estados de máquina.

¹² A IDE Thonny é descrita pelos seus idealizadores como “IDE para iniciantes”. Foi escolhida por ter uma interface amigável, por ser “leve”, multiplataforma, de fácil integração de novas bibliotecas, com console integrado com ferramenta de *debug*. (THONNY.ORG, 2018).

Para diminuirmos ainda mais o volume a ser baixado, criamos uma máquina virtual Ubuntu Minimal (a qual nomeamos como “base”) somente texto, contendo arquivos executáveis que possibilitassem construir as três máquinas com todas as configurações necessárias e com tamanho um tamanho razoável de aproximadamente 900 (novecentos) *megabytes*. Assim, o estudante teve de fazer apenas um *download*, importar o arquivo para seu ambiente de virtualização (Oracle VM Virtual Box) e, localmente, clonar mais duas cópias, totalizando três máquinas virtuais. Para que cada máquina tivesse as características necessárias ao curso conforme sua função planejada na rede virtual, havia na máquina “base” três arquivos de configuração automática que possibilitavam que cada clone se tornasse uma máquina da rede, gerando uma máquina principal (Servidor1), com interface gráfica e ambiente de programação; e mais duas máquinas secundárias (Servidor2 e Servidor3).

As instruções para a reprodução do laboratório virtual, o *link* para baixar a máquina “base” e todos os materiais didáticos foram postados em um ambiente virtual sobre o qual tratamos na próxima seção.

2.4 AMBIENTE VIRTUAL E RECURSOS DIGITAIS PARA GUIAR OS ESTUDOS

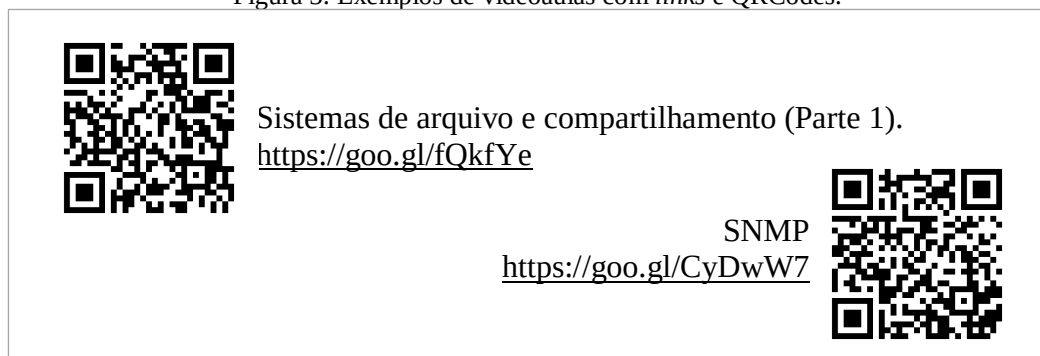
O Google Classroom (no Brasil, Google Sala de Aula) é uma plataforma *web* da Google que aproveita a disponibilização de um conjunto de serviços (*e-mail*, *drive*, gerenciador de tarefas, entre outros) e foi utilizado para disponibilizar os materiais didáticos (RA, videoaulas, tutoriais, páginas *web*, etc.) e promover interações virtuais. De forma auxiliar, para complementar a interação com os estudantes, utilizamos o Whatsapp, no qual criamos um grupo específico para a turma.

Para dar suporte ao Estudo Autônomo foram produzidos vídeos autorais e também utilizados vídeos de terceiros, especialmente selecionados para apoiar o estudo de cada tema. Cada vídeo tinha em média 7 (sete) minutos e foram produzidas 27 (vinte e sete) videoaulas autorais (totalizando, 3 horas e 28 minutos) e selecionados 17 (dezessete) vídeos de terceiros (totalizando, 1 hora e 48 minutos). Na Figura 3 são disponibilizados dois *links* com exemplos de videoaulas autorais (Compartilhamento de Arquivos; e SNMP) publicadas no Youtube e utilizadas em nossa proposta de ensino-aprendizagem. Todos os vídeos podem ser acessados seguindo os RA para Estudos Autônomos no APÊNDICE C.

A produção e a seleção dos vídeos foram realizadas após a determinação dos temas e conteúdos. No caso das videoaulas autorais, antes de cada gravação, um roteiro foi escrito

para que o vídeo fosse objetivo e relativamente curto. Para gravação, inserção de textos explicativos, anotações, edição e acréscimo de efeitos, foi utilizado o *software* Camtasia Studio por agregar todas estas funções. A publicação dos vídeos autorais foi feita no Youtube. Além dos vídeos, foram disponibilizados outros recursos como tutoriais *online*, conteúdos de páginas *web* e indicação de leituras.

Figura 3: Exemplos de videoaulas com *links* e QRCodes.



Fonte: Autoria Própria

Todos os instrumentos didáticos foram disponibilizados na plataforma Google Classroom, inclusive as avaliações, sobre as quais discutimos na próxima seção.

2.5 INSTRUMENTOS DE AVALIAÇÃO E UTILIZAÇÃO DE RUBRICAS

Em nossa proposta de ensino-aprendizagem utilizamos como instrumentos de avaliação, além das Avaliações Modulares (APÊNDICE D), os RA presenciais (APÊNDICE C), cujas resoluções nos permitem ter *feedback* sobre a aprendizagem dos estudantes e são utilizados para entendimento das suas evoluções. Apesar de “não valerem nota”, os RA presenciais são avaliativos e sobre eles são aplicadas as rubricas como se “valessem nota” com seu resultado apresentado ao estudante para autoavaliação. Este procedimento é importante para que possamos, em função das respostas, analisarmos o desempenho de cada estudante em cada habilidade trabalhada e então devolver-lhes os *feedbacks* para que possam corrigir seus erros e melhorar seus desempenhos antes dos próximos RA ou das Avaliações Modulares. Desta maneira, o estudante estará ciente, de forma antecipada, sobre quais critérios serão utilizados para avaliar suas tarefas e sobre qual será o formato das Avaliações Modulares.

Para atribuição das rubricas, que possibilitam determinar o nível de desempenho em que cada estudante está posicionado, aplicamos três avaliações, sendo uma Avaliação

Diagnóstica (antes do início do primeiro módulo) e duas Avaliações Modulares (aplicadas ao final de cada um dos dois módulos). Nestas avaliações, assim como nos RA para Estudos Presenciais, os estudantes tiveram de resolver problemas do domínio de Redes de Computadores, envolvendo as três habilidades (Identificação de uma solução, Automatização da solução e Associação da solução para resolução de outros problemas).

Dadas as características da nossa proposta de ensino-aprendizagem, utilizamos rubricas analíticas, pois nossa pretensão é a de obter os resultados sobre os critérios definidos para cada uma das três habilidades, a fim de construirmos um diagnóstico sobre cada instrumento didático avaliado (RA e Avaliações).

Nas rubricas que definimos, os níveis de desempenho são dispostos de forma gradativa, da seguinte forma: Insuficiente, Próximo de Proficiente, Proficiente e Avançado. Para que atinja bons resultados em cada habilidade, o estudante deve atingir os conceitos Proficiente ou Avançado. Esta disposição é apresentada no Quadro 8.

Quadro 8: Rubrica utilizada: Critérios (Habilidades) x Níveis de Desempenho (Conceitos).

Habilidades Conceitos	<i>Identificar uma solução possível para um problema</i>	<i>Automatizar a solução identificada;</i>	<i>Associar a solução do problema com possíveis soluções de outros problemas.</i>
Avançado	Descreve uma estratégia correta para a solução do problema, apresentando com riquezas de detalhes os elementos que constituem a solução e como os mesmos estão organizados, demonstrando compreensão dos conceitos de redes e programação.	Implementa uma solução completa e correta para o problema. O código resultante atende a especificação para a solução, possui lógica de programação clara e bem estruturada. Emprega as boas práticas de programação ¹³ .	Associa de forma correta uma dada solução a outro problema com características similares a problemas anteriormente resolvidos. A associação é apoiada por uma justificativa e uma descrição claras e completas quanto aos princípios e características que conduziram a tomada de decisão sobre o reaproveitamento das ideias e soluções para o novo contexto apresentado.
Proficiente	Descreve uma estratégia correta para a solução do problema, entretanto há algumas omissões e/ou pequenos erros no detalhamento dos elementos que constituem a solução e como os mesmos estão organizados. Estas omissões/erros não trazem prejuízo à compreensão dos conceitos de redes e programação.	Implementa uma solução completa e correta para o problema ou com pequenas imperfeições facilmente sanáveis. O código resultante atende a especificação para a solução, mas contém pequenas falhas que não prejudicam a lógica (direcionamento de saída, por exemplo), possui lógica de programação clara e bem estruturada. Falha ao empregar alguma das boas práticas de programação ¹³ .	Associa de forma correta uma dada solução a outro problema com características similares a problemas anteriormente resolvidos. A associação é apoiada por uma justificativa e uma descrição com pequenos equívocos e/ou omissões quanto aos princípios e características que conduziram a tomada de decisão sobre o reaproveitamento das ideias e soluções para o novo contexto apresentado.

¹³ Comentários relevantes, nomes de variáveis e funções significativas, ausência de código morto.

Quadro8: Rubrica utilizada: Critérios (Habilidades) x Níveis de Desempenho (Conceitos) (Continuação).

Habilidades Conceitos	Identificar uma solução possível para um problema	Automatizar a solução identificada;	Associar a solução do problema com possíveis soluções de outros problemas.
Próximo de Proficiente	Descreve uma estratégia aceitável para a solução do problema, ainda que omita muitos detalhes ou erre no detalhamento dos elementos que constituem a solução e como os mesmos estão organizados. Estas omissões/erros dificultam a demonstração de compreensão dos conceitos de redes e programação.	Implementa uma solução incompleta. O código resultante atende parcialmente as especificações para a solução e/ou contém pequenas falhas quanto à lógica e estruturação da programação. Falha ao empregar boas práticas de programação ¹³ .	Associa de forma correta uma dada solução a outro problema com características similares a problemas anteriormente resolvidos. A associação não possui justificativa e/ou a descrição (ou estão mal formuladas), não demonstra os princípios e características que conduziram a tomada de decisão sobre o reaproveitamento das ideias e soluções para o novo contexto apresentado.
Insuficiente	Apresenta uma estratégia de solução incorreta para o problema ou inadequada, demonstrando não compreender o domínio do problema.	Demonstra muita deficiência para automatizar a solução. O programa não executa, pois há erros de sintaxe e/ou lógica de programação e o programa precisa ser refeito.	Em sua tentativa, não associa de forma correta uma dada solução a outro problema com características similares a problemas anteriormente resolvidos.

Fonte: Autoria própria, a partir de Brookhart (2013).

No próximo capítulo apresentamos o Estudo de Caso utilizado para analisar a aplicação da nossa proposta de ensino-aprendizagem, no formato de um curso de extensão.

3 ESTUDO DE CASO

Neste capítulo descrevemos o planejamento e execução de um estudo de caso realizado para avaliar nossa proposta de ensino-aprendizagem. O estudo de caso foi realizado no Instituto Federal de Rondônia, campus Zona Norte, no período de 18 de novembro de 2017 a 3 de fevereiro de 2018, na forma de um curso de extensão, com 15 (quinze) estudantes de cursos de nível superior da área de computação.

Estudo de caso é uma forma empírica de investigar um fenômeno contemporâneo dentro de seu contexto da vida real, seguindo um conjunto de procedimentos pré-especificados (YIN, 2010). Segundo este autor, o estudo de caso é adequado quando: (i) as questões são do tipo “como” e “por que”; (ii) o pesquisador possui pouco controle sobre os acontecimentos; e, (iii) o foco se concentra em fenômenos contemporâneos inseridos em algum contexto da vida real. Tais características justificam a sua adoção nesta pesquisa.

A lógica do planejamento do estudo de caso, das técnicas de coletadas de dados e recomendações para tabulação e análise dos dados seguiram as orientações de Yin (2010).

Este capítulo está distribuído em seções, organizadas da seguinte forma: na primeira seção apresentamos o planejamento deste estudo de caso, com as questões de pesquisa e as etapas inerentes a este planejamento. A segunda seção é destinada à apresentação da execução deste estudo de caso. Na terceira seção são analisados os desvios e, na quarta seção é apresentada a avaliação sobre a validade do estudo de caso.

3.1 PLANEJAMENTO

Nesta seção apresentamos a descrição das questões de pesquisa, a caracterização dos sujeitos que participaram do estudo de caso, os recursos utilizados, o planejamento da aplicação da proposta, as unidades de análise e os procedimentos para coleta de evidências.

3.1.1 Questões de pesquisa

A realização do estudo de caso visa responder às seguintes questões de pesquisa:

- 1) Em que aspectos a proposta de ensino-aprendizagem melhora as habilidades dos estudantes para identificar soluções para problemas reais de Redes de Computadores?

- 2) Em que aspectos a proposta de ensino-aprendizagem melhora as habilidades dos estudantes para automatizar soluções por meio de Programação para problemas de Redes de Computadores?
- 3) Em que aspectos a proposta de ensino-aprendizagem melhora as habilidades dos estudantes em associar a solução encontrada com outras possíveis soluções de outros problemas de redes?
- 4) Quais limitações são verificadas na implementação da proposta de ensino-aprendizagem?

3.1.2 Definição dos sujeitos da pesquisa

Os sujeitos da nossa pesquisa são estudantes de cursos superiores de computação na cidade de Porto Velho-RO, que tenham cursado previamente alguma disciplina de Programação¹⁴.

Como precisamos classificar os estudantes para ordenar o preenchimento das vagas escolhemos como método de classificação o coeficiente de rendimento dos estudantes (enviado digitalmente) pela sua praticidade e simplicidade. Para participação na pesquisa cada estudante assinou o Termo de Consentimento Livre e Esclarecido (TCLE) cujo modelo se encontra no APÊNDICE E.

3.1.3 Recursos utilizados

Para a realização do estudo de caso foram planejados os seguintes recursos:

- Laboratório de informática: contendo 33 (trinta e três) computadores com acesso à Internet, sistema operacional Windows com Oracle VM Virtualbox instalado;
- Roteiros de Aprendizagem: 12 (doze) Roteiros de Aprendizagem, sendo: 6 (seis) para Estudos Autônomos, contendo os conjuntos de questões, bem como *links* para outros recursos com conteúdo teórico (vídeo aulas e indicações de leituras auxiliares e/ou complementares); e, 6 (seis) para Estudos Presenciais, contendo os problemas propostos baseados em situações

¹⁴ Não exigimos conhecimentos prévios de Redes de Computadores por entendermos que os conceitos necessários seriam ministrados ao longo do curso.

do domínio de Redes de Computadores;

- Videoaulas autorais: 27 (vinte e sete) videoaulas autorais com tempo médio de aproximadamente 7 (sete) minutos cada, num total de 3 (três) horas e 28 (vinte e oito) minutos, postadas no Youtube.
- Videoaulas de terceiros: 17 (dezessete) vídeos de terceiros com tempo médio de aproximadamente 5 (cinco) minutos cada, totalizando 1 (uma) hora e 28 (vinte e oito) minutos que tratavam sobre Python, programação e Redes de Computadores;
- Ambiente virtual de aprendizagem: Google Sala de Aula.
- Laboratório virtual de redes¹⁵: montado sobre o Oracle VM Virtualbox, com três máquinas com sistema operacional Linux, sendo uma com interface gráfica (com a IDE Thonny e Python 3 instalados) e duas com interfaces de linha de comando (com Python 3 instalado).
- Apostila autoral¹⁶: sobre os conceitos básicos de Redes de Computadores, Programação e Sistema Operacional Linux.
- Tutoriais autorais¹⁷: para instruir os estudantes sobre: como montar o laboratório virtual de redes; como resolver problemas pontuais; como configurar o serviço de mensagens Telegram.
- Páginas *web*: auxílio a assuntos específicos (*backup*, estrutura de diretórios do Linux, etc.).

3.1.4 Planejamento da aplicação

A aplicação foi planejada para ser implementada em etapas, conforme disposto a seguir:

Preparação:

- Composição da turma
- Coleta de endereços de *e-mails* dos estudantes
- Coleta de contatos de Whatsapp dos estudantes

¹⁵ O laboratório virtual foi montado no laboratório de informática do IFRO e cada estudante reproduziu de forma idêntica em seu computador pessoal. Sua finalidade é a de possibilitar ambientes idênticos tanto nos Estudos Autônomos quanto nos Estudos Presenciais.

¹⁶ Apostila Básica disponível no link: <https://goo.gl/ukBtgo>.

¹⁷ Este material foi disponibilizado de forma *online* e seus *links* estão dispostos no Produto (<http://mpet.ifam.edu.br/dissertacoesdefendidas/>).

- Criação de grupo de Whatsapp para a turma
- Postagem do material introdutório no Google Classroom: Videoaulas, dicas de Python, instruções sobre o Laboratório Virtual, Apostila básica sobre Redes de Computadores, Linux e Python.

Encontro presencial para apresentação do curso:

- Apresentação do curso (Formato, organização, conteúdos, temas, estudos, roteiros, tarefas, rubricas, duração, canais de dúvidas, ambiente virtual, laboratório virtual, certificação.)
- Assinatura do Termo de Consentimento Livre e Esclarecido
- Preenchimento do Questionário Sócio-Econômico-Educativo
- Avaliação diagnóstica

Módulo I - Monitoramento de Redes de Computadores

- Execução de três temas: Identificando a movimentação da vizinhança (Monitoramento de disponibilidade de equipamentos e serviços em rede); Cuidando do Habitat (Monitoramento de estado de discos e partições, em rede); De olho nas intenções (Monitoramento de ações de usuários).
- Disponibilização de 6 (seis) Roteiros de Aprendizagem, distribuídos em três temas, sendo um RA Autônomo e um RA Presencial por tema.
- Avaliação Modular I.

Módulo II - Preservação de informações e comunicação de eventos

- Execução de três temas: Construir é difícil. Reconstruir pode ser impossível (Compactação de dados, cópias de segurança em rede e registros de atividades); Um é pouco. Então tenha dois (ou mais!) (Sincronismo de dados em rede); Seja o primeiro a saber (Automatização de mensagens de eventos e de alertas via *e-mail* e mensageria instantânea).
- Disponibilização de 6 (seis) Roteiros de Aprendizagem, distribuídos em três temas, sendo um RA Autônomo e um RA Presencial por tema.
- Avaliação Modular II.

3.1.5 Unidades de análise

Em nosso estudo de caso avaliamos o desempenho individual dos estudantes ao trabalharem as habilidades para identificar soluções para problemas reais de Redes de Computadores; construir soluções por meio de programação para problemas de Redes de Computadores; associar as soluções encontradas com outras possíveis soluções de outros problemas de redes. Também fazemos observações qualitativas sobre as percepções dos estudantes, suas dificuldades, assim como do professor na condução da proposta de ensino.

3.1.6 Coletas de evidências

Para a coleta das evidências, adotamos instrumentos criados em função das habilidades a serem trabalhadas e dos RPA. Para critérios de aprovação individual ao final da implementação da proposta de ensino-aprendizagem, consideramos apenas a Avaliação Modular 1 e a Avaliação Modular 2. Entretanto, para verificarmos os desempenhos individuais de cada estudante ao longo do percurso desta implementação e fornecermos aos estudantes os *feedbacks* necessários, de forma a auxiliá-los em suas reflexões e ajustes em suas evoluções, além das Avaliações Modulares, consideramos a Avaliação Diagnóstica e cada um dos Estudos Presenciais. Conforme o desempenho, cada solução referente a cada habilidade é nivelada dentro das rubricas em cada um destes instrumentos e um *feedback* é fornecido ao estudante. Utilizamos também instrumentos não dependentes de rubricas, utilizados para auxílio em análises comportamentais. Elencamos a seguir, cada um destes instrumentos:

- **Questionário Sócio-Econômico-Educacional:** Aplicado no primeiro dia de aula para coletar informações que caracterizam os sujeitos da pesquisa;
- **Avaliação Diagnóstica:** Foi planejada para ser aplicada antes do primeiro Estudo Autônomo com a função de verificar o nível em que os estudantes se encontram em função de seus conhecimentos prévios;
- **Avaliação Modular:** São duas, cada uma aplicada ao final de cada módulo para verificar o desempenho individual de cada estudante em cada habilidade e para formar sua “nota final”;
- **Estudo Presencial:** Ao todo são 6 (seis), divididos em 3 (três) por módulo. Após cada Estudo Presencial os mesmos critérios das rubricas aplicados nas

Avaliações Modulares são aplicados nestes instrumentos, cujos resultados são reportados aos estudantes, apresentando os níveis nos quais suas soluções se encontram em relação às rubricas e a cada habilidade trabalhada. Durante suas execuções observamos os questionamentos, discussões, comportamentos e dificuldades dos estudantes como fonte de análise suplementar;

- **Estudo Autônomo:** As questões empregadas nos Estudos Autônomos exigiam dedicação para construção das respostas e suas resoluções refletiam isto, permitindo intervenções caso suas execuções fossem falhas ou ausentes;
- **Questionamentos dos estudantes via e-mail, telefone e Whatsapp:** Durante o percurso do estudo de caso vários questionamentos foram feitos por intermédio destes meios;
- **Discussões ao final dos Estudos Presenciais:** Ao final de cada Estudo Presencial, havia uma “roda” de discussão para coleta de opiniões sobre os materiais e recursos empregados;
- **Questionários inseridos no final dos Roteiros de Aprendizagem dos Estudos Autônomos:** Ao final de cada Estudo Autônomo os estudantes respondiam sobre o tempo dedicado aos estudos, necessidades de consultas a fontes extras e observações, sugestões ou críticas sobre o instrumento;
- **Observações diversas:** realizadas durante a execução do estudo de caso.

Na próxima seção apresentamos como realizamos a execução do estudo de caso.

3.2 EXECUÇÃO

Para implementação da proposta de ensino-aprendizagem em formato de curso de extensão, foram abertas 30 (trinta) vagas para estudantes de cursos de nível superior na área de Computação de instituições da cidade de Porto Velho/RO. O pré-requisito era tão somente o estudante ter cursado previamente alguma disciplina de Programação. Não exigimos conhecimentos prévios de Redes de Computadores por entendermos que os conceitos necessários seriam ministrados ao longo do curso. Para este curso, houve 53 (cinquenta e três) inscritos, sendo 30 (trinta) estudantes classificados conforme o coeficiente de rendimento individual.

A quantidade de vagas foi definida em função do tamanho do laboratório disponibilizado pelo IFRO que possui 33 computadores. A cidade de Porto Velho foi definida

por deter o Campus Porto Velho Zona Norte no qual está vinculado como Professor EBTB este mestrand, facilitando o deslocamento e aproveitando o apoio logístico do IFRO.

Como o período de realização do curso coincidiu com o encerramento do semestre, concorrendo com as avaliações finais dos cursos regulares nos quais os estudantes estavam matriculados e com as férias de final de ano, além de alegações de problemas particulares, 15 (quinze) estudantes desistiram. Desta forma, consideramos, para efeitos de análise, os 15 (quinze) estudantes que concluíram o curso.

Dentre os estudantes concluintes, 80% eram do sexo masculino e 20% do sexo feminino. Conforme respostas ao Questionário Sócio-Econômico-Educativo, a média de idade foi de 24,2 anos. Na ocasião do curso, 40% cursavam de faculdades públicas e 60%, privadas. Também 60% dos estudantes cursavam o segundo período de seus cursos superiores, 27% cursavam o quarto período e 13% cursavam o sétimo ou oitavo período. Quanto aos cursos superiores, 53% estavam matriculados e cursando CST em Redes de Computadores, 20% CST em Análise e Desenvolvimento de Sistemas, 20% Bacharelado em Sistemas de Informação e 7% Engenharia da Computação. Todos afirmaram ter acesso a computador e Internet banda larga em suas próprias residências. Nenhum dos estudantes havia trabalhado profissionalmente com Programação e 27% afirmaram terem trabalhado profissionalmente com Redes de Computadores. A maioria declarou se identificar mais com disciplinas de Redes de Computadores do que com as de Programação. Quanto ao conhecimento sobre linguagens de programação, 87% afirmaram conhecer duas ou mais e 40% conheciam Python.

Antes do início efetivo do curso houve um trabalho de divulgação com visita aos *campi* Zona Norte e Calama, além de uma visita na Instituição de Ensino Superior Uniron, todos em Porto Velho e divulgação em grupos de Whatsapp e no Facebook.

O Edital de Seleção nº 94 foi publicado dia 6 de novembro¹⁸ contendo as regras para inscrições no curso e demais informações (Figura 4).

Figura 4: Publicação do edital do curso.

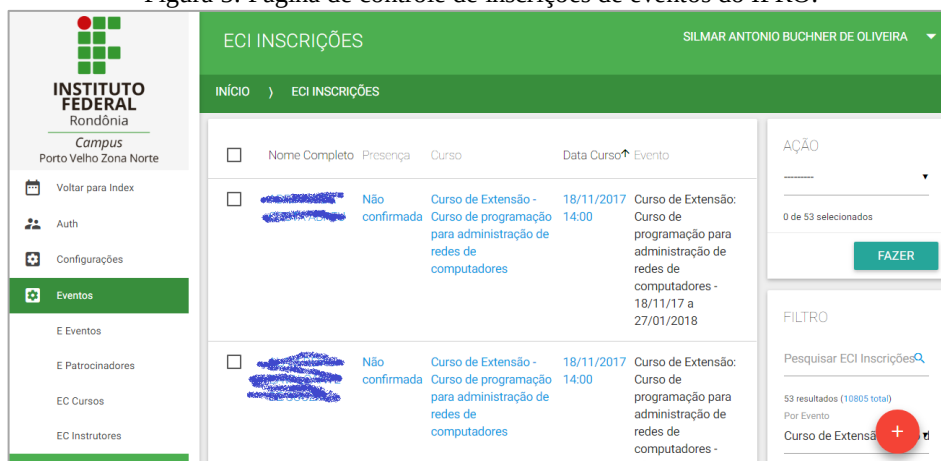


Fonte: Autoria própria.

¹⁸ Para acessar o edital na íntegra: <https://goo.gl/3mLv3k>

As inscrições foram realizadas em uma página específica disponibilizada por um sistema de controle de eventos do IFRO, conforme Figura 5.

Figura 5: Página de controle de inscrições de eventos do IFRO.



Fonte: Autoria própria.

Como auxílio para a divulgação, houve também a publicação no portal do IFRO, contendo alguns detalhes do curso e o *link* para inscrições (Figura 6).

Figura 6: Destaque no portal do IFRO com notícia sobre o curso.



Fonte: Autoria própria.

Os Estudos Autônomos foram liberados entre os sábados e as segundas-feiras de cada semana, de forma que o estudante tivesse, no mínimo, 5 (cinco) dias para cumprir as tarefas, tempo que julgamos suficiente, entretanto, dependente da dedicação de tempo estudante, visto que a média prevista para execução era de 2 (duas) horas semanais.

A parte presencial da execução da proposta de ensino-aprendizagem (Estudos Presenciais e Tutorias Presenciais) foi realizada no Laboratório de Informática nº 04,

disponibilizado pelo IFRO para a execução do curso. Os encontros para os Estudos Presenciais foram realizados aos sábados, das 14:00 às 18:00. As Tutorias Presenciais tinham caráter facultativo aos estudantes e foram realizadas sábados pela manhã, antecedendo aos Estudos Presenciais, para elucidação de dúvidas, das 9:00 às 11:30 e, conforme previsto, também em outras oportunidades solicitadas pelos estudantes.

Algumas tarefas foram realizadas durante todo o decorrer do curso. Por exemplo, os *feedbacks* sobre os Estudos Presenciais eram fornecidos aos estudantes durante a semana seguintes com o intuito de retratar seus desempenhos e fornecer instruções e dicas que pudessem melhorar aspectos que possuíam deficiências. Outra tarefa que se repetiu ao longo do curso foi o contato com os estudantes faltosos nos Estudos Presenciais, para entendermos os motivos das suas faltas e incentivá-los a continuarem no curso.

A disponibilização de cada etapa e suas datas de realização são descritas no Quadro 9.

Quadro 9: Cronograma de realização dos estudos e avaliações modulares.

Preparação (2017)			Módulo 1 (2017)			Módulo 2 (2018)			
Dia	Mês	Atividade/Tarefa	Dia	Mês	Atividade/Tarefa	Dia	Mês	Atividade/Tarefa	
Atividades iniciais Semana 01	13-17	11	Tema 01 Semana 02	19-24	11	Tema 04 Semana 06	02-05	01	Estudo autônomo 04
				25	11		06	01	09:00 - 11:30 Tutoria presencial facultativa
									14:00 - 18:00 Estudo presencial 01
	18	11	Tema 02 Semana 03	26-01	11-12	Tema 05 Semana 07	07-12	01	Estudo autônomo 05
				02	12		13	01	09:00 - 11:30 Tutoria presencial facultativa
									14:00 - 18:00 Estudo presencial 02
	18	11	Tema 03 Semana 04	03-08	12	Tema 06 Semana 08	14-26*	01	Estudo autônomo 06
				09	12		27	01	09:00 - 11:30 Tutoria presencial facultativa
									14:00 - 18:00 Estudo presencial 03
	18	11	Tema 05 Semana 09	10-15	12	Tema 09 Semana 09	28-02	01-02	Preparação para avaliação do módulo 02
16				12	03		02	14:00 - 18:00 Avaliação módulo 01	
								17/12/2017 a 01/01/2018 - RECESSO	* Houve uma prorrogação por conta de

* Houve uma prorrogação por conta de um problema com o Telegram

Fonte: Autoria própria.

As semanas que se seguiram são descritas a seguir, bem como detalhes de suas ocorrências. Todas as atividades e tarefas foram testadas antes de serem postadas.

3.2.1 Semana 01 – 13/11/2017 a 18/11/2017 – Atividades iniciais

A preparação do ambiente é muito importante e foi elaborada de forma a proporcionar todo o conjunto inicial de instrumentos necessários para que o estudante desenvolva suas atividades com eficiência. Nesta semana realizamos as atividades iniciais para a realização do curso. Iniciamos no dia 13 de novembro com a postagem do material introdutório no Google Classroom:

- Mensagem de boas vindas;
- Dicas sobre a linguagem Python e indicação de *links* de videoaulas de terceiros sobre Python básico;
- Videoaula autoral com exemplo de programação em Python, estipulando um padrão de boas práticas de programação adotado no curso;
- Disponibilização de instruções sobre a montagem do Laboratório Virtual através de videoaulas autorais sobre: Instalação do ambiente de virtualização; Clonagem da máquina base; Instalação e configuração dos servidores que compõem o Laboratório Virtual;
- Disponibilização de apostila básica autoral sobre Redes de Computadores, sistema operacional Linux e Programação, desenvolvida em função do curso.

A classificação dos estudantes aptos a participarem do curso foi concluída no dia 14 de novembro e a composição da turma foi realizada no dia 15 de novembro com a comunicação dos classificados via *e-mail* e Whatsapp, para o qual criamos um grupo. Neste comunicado também foi informado sobre como acessar o ambiente virtual do Google Classroom e sobre os materiais e tarefas iniciais nele postados. O resultado oficial foi publicado pelo IFRO no dia 16 de novembro.

O primeiro encontro presencial com a turma, inicialmente com 30 (trinta) estudantes, ocorreu no dia 18 de novembro de 2017. Neste encontro apresentamos:

- Objetivos do curso;
- O formato de estudos utilizando o modelo de sala de aula invertida;
- A organização do curso em módulos e os respectivos temas e conteúdos;
- Os RA para Estudos Presenciais e Estudos Autônomos, bem como as tarefas que os compunham;
- Explicações sobre as avaliações por rubricas e critérios de avaliações de desempenho;

- Duração do curso;
- Informações sobre o suporte a eventuais dúvidas e os canais utilizados (*e-mail* e Whatsapp);
- Frequência e importância dos *feedbacks* do professor;
- Utilização do Google Classroom;
- Importância e finalidade do Laboratório Virtual;
- Imprescindibilidade da execução de todas as tarefas e respeito aos prazos;
- Critérios de emissões de certificados ao final do curso.

Neste mesmo encontro apresentamos o Termo de Consentimento Livre e Esclarecido (TCLE), explicamos seu propósito e todos optaram por assiná-lo. Em seguida explicamos o propósito e a importância do Questionário Sócio-Econômico-Educacional (APÊNDICE E) para a pesquisa em que eles estavam sendo inseridos. Submetemos e todos os presentes concordaram em preencher. Em seguida aplicamos a Avaliação Diagnóstica (APÊNDICE F). Ao final foram liberados os acessos ao RA para Estudo Autônomo 01, envolvendo o primeiro tema a ser estudando durante a semana seguinte.

3.2.2 Semana 02 – 19/11/2017 a 25/11/2017 – Módulo 1 – Tema 01

- Tema 01: Identificando a movimentação da vizinhança

Este primeiro tema marcou o início do Módulo 1 (Monitoramento de Redes de Computadores) e tratava da automatização de procedimentos de monitoramento de estado de equipamentos e de serviços em rede, atividades que compõem a base do curso. No Estudo Autônomo os estudantes conheceram a dinâmica dos RA e iniciaram a ambientação com as exigências e os padrões estabelecidos no curso.

As tarefas que os estudantes resolveram no Estudo Autônomo 01 continham questões que envolviam monitoramento de Redes de Computadores através do comando PING e do protocolo ICMP. Os estudantes também resolveram os primeiros problemas e tiveram contato com as primeiras estruturas de programação em Python direcionadas à administração de Redes de Computadores.

Houve alguns atrasos na entrega do Estudo Autônomo 01, pois alguns estudantes tiveram dificuldades para montar o Laboratório Virtual. O principal motivo foi que, inicialmente, a arquitetura do sistema operacional Linux utilizada era a de 64 bits, entretanto,

alguns estudantes informaram que seus computadores pessoais possuíam arquitetura física de 32 bits e não conseguiam importar a máquina virtual “base” para o ambiente virtual do Oracle VM Virtualbox. Para resolver este problema de conflito de arquiteturas, refizemos a máquina virtual “base” com 32 bits e todos conseguiram montar o Laboratório Virtual. Por conta desta alteração tivemos de refazer o vídeo que instruía sobre a reprodução do Laboratório Virtual.

Outro problema identificado foi quanto à configuração das interfaces de rede das máquinas virtuais, pois os testes iniciais foram efetuados com Internet fornecida via provedor com distribuição automática de números IP (DHCP). A configuração das interfaces de rede de cada máquina funcionava no modo *bridge*, ou seja, cada máquina virtual receberia um número IP na mesma faixa da máquina hospedeira (computador físico com ambiente de virtualização instalado). Contudo, nem todos recebiam Internet desta forma, pois havia situações em que não havia servidor de DHCP ou havia a disponibilização apenas do IP da máquina hospedeira. Para sanar isto, houve a necessidade da reconfiguração. Foi inserida mais uma interface de rede em cada máquina virtual no modo NAT para comunicação com a máquina hospedeira através de uma determinada faixa de IP. Esta rede proporcionava a comunicação com a Internet. Para a comunicação entre as máquinas virtuais que compunham o Laboratório Virtual foi atribuída uma faixa de IP distinta da utilizada na Interface NAT, com a configuração de “Rede Interna”. Isto resolveu o problema em qualquer ambiente em que a máquina hospedeira estivesse inserida, desde que esta tivesse acesso à Internet.

À tarde, no Estudo Presencial, a fase de “aquecimento” que estava prevista para 30 (trinta) minutos acabou se estendendo por mais uma hora por conta dos problemas de arquitetura, da configuração das interfaces de redes e de alguns estudantes não terem completado seus Estudos Autônomos. Apesar do atraso inicial, o restante do Estudo Presencial transcorreu normalmente. Os estudantes resolveram problemas (identificar, automatizar e associar soluções) envolvendo lentidão de computadores e o monitoramento remoto destas situações.

3.2.3 Semana 03 – 26/11/2017 a 02/12/2017 – Módulo 1 – Tema 02

- Tema 02: Cuidando do *Habitat*

Neste segundo tema os estudantes seguiram com as atividades sobre monitoramento de Redes de Computadores, entretanto, sob um aspecto mais interno de cada computador que compunha a rede, com o objetivo de monitorar estados de componentes diversos, como discos

rígidos e ocupação de suas partições.

As tarefas resolvidas pelos estudantes nos Estudos Autônomos envolveram monitoramento das partições dos discos rígidos dos computadores da rede a fim de prevenir possíveis estados de superlotação que poderiam acarretar indisponibilidade de serviços. Para o sucesso deste monitoramento é necessário o conhecimento das estruturas de diretórios do Linux. Para tanto, os estudantes também resolveram questões envolvendo estruturas de diretórios.

Nos Estudos Presenciais os estudantes resolveram problemas envolvendo monitoramento remoto de discos e partições de servidores da rede, abrangendo programação em Python e o protocolo SNMP.

3.2.4 Semana 04 – 03/12/2017 a 09/12/2017 – Módulo 1 – Tema 03

- Tema 03: De olho nas intenções

O terceiro tema introduziu modos de varredura de documentos de registro de atividades (*logs*), em busca por padrões (palavras ou expressões) dentro dos textos, como forma de verificação e de prevenção de problemas. Para resolução das tarefas envolvidas neste tema foram exigidos conteúdos de programação que costumam representar maiores desafios para os estudantes, como manipulação de *strings*, laços de repetição (*for*) e estruturas de decisão (*if – else*). Também foi necessário que o estudante conhecesse um pouco mais sobre o Linux e como são empregados os arquivos de registros de atividades neste sistema operacional. Como havia a necessidade de manipulação de arquivos, os estudantes também tiveram de utilizar meios de criação e de edição de arquivos na linguagem Python.

As tarefas resolvidas pelos estudantes no Estudo Autônomo envolveram questões que abrangiam edição de arquivos de configuração do Linux para facilitar a coleta de informações e buscas por palavras específicas dentro dos arquivos de registros de atividades.

Nos Estudos Presenciais os estudantes resolveram problemas sobre possíveis tentativas de acesso indevido de usuários através de varredura por palavras-chave em arquivos de registros de atividades de autenticação de usuários.

Os estudantes demonstraram dificuldades para trabalharem com estas tarefas, principalmente quanto à manipulação de *strings*, tanto nos Estudos Autônomos quanto nos Estudos Presenciais (apesar de todos terem cursado ao menos uma disciplina de Programação). Estas dificuldades foram percebidas durante a semana do Estudo Autônomo,

pelas diversas solicitações de auxílio via Whatsapp e durante as execuções das tarefas do Estudo Presencial. Atendemos a todos os pedidos de auxílio e indicamos revisão dos vídeos sobre manipulação de *strings* disponibilizados na fase preparatória do curso. Estas ações amenizaram o problema, pois os questionamentos ao longo do curso decresceram.

Em virtude de a Avaliação Modular 1, agendada para dia 16/12/2017 e por alguns estudantes não terem concluído alguns dos seus Estudos Autônomos e/ou Estudos Presenciais, os estudantes solicitaram uma Tutoria Presencial Facultativa extra que foi agendada para a terça-feira seguinte, das 19:00 às 22:00.

3.2.5 Semana 05 – 10/12/2017 a 16/12/2017 – Módulo 1 – Preparação e aplicação da Avaliação

Durante esta semana incentivamos a revisão de todos os Estudos passados, como forma de preparação para a Avaliação Modular. Conforme agendado, prestamos a Tutoria Presencial Facultativa. Também demos continuidade às elucidações de dúvidas via Whatsapp e *e-mail*.

Conforme agendamento, aplicamos a Avaliação Modular 1 em 16/12/2017) cuja composição seguiu a mesma estrutura dos RA dos Estudos Presenciais, com exceção do compartilhamento de ideias e de resultados, dado o caráter individual da avaliação.

Após esta avaliação foi iniciado o período de recesso, entre 17/12/2017 e 01/02/2018. O *feedback* sobre a Avaliação Modular 1 foi prestado até o dia 20/12/2017.

3.2.6 Semana 06 – 02/01/2018 a 06/01/2018 – Módulo 2 – Tema 04

- Tema 04: Construir é difícil. Reconstruir pode ser impossível.

Os estudos deste quarto tema marcaram o início do Módulo 2 (Preservação de informações e comunicação de eventos). Este tema trata da compactação de dados, *backup* remoto automatizado e registro de *logs*.

No Estudo Autônomo os estudantes resolveram questões que envolviam empacotamento, compactação e compartilhamento de dados, como parte das tarefas de *backup*. Também resolveram questões sobre verificação da execução de fases do *backup*.

Os problemas resolvidos pelos estudantes no Estudo Presencial envolveram a

conscientização da necessidade de *backups*, além da automatização da execução de *backups* em máquinas remotas através de programação em Python, com verificação de etapas da execução e registro em arquivos de *log*.

3.2.7 Semana 07 – 07/01/2018 a 13/01/2018 – Módulo 2 – Tema 05

- Tema 05: Um é pouco. Então tenha dois (ou mais!)

Neste quinto tema, abordamos a importância da elaboração de políticas de *backup* como medida de segurança de dados. Complementar ao tema anterior, também apresentamos, como forma de ampliação da disponibilidade dos dados, a automatização do sincronismo de dados entre servidores.

No Estudo Autônomo os estudantes resolveram questões que envolviam sincronismo de dados, aplicação de políticas de segurança e agendamento de tarefas no Linux. Por requisitarem mais conhecimentos sobre o sistema operacional Linux, os estudantes foram incentivados a reverem a apostila disponibilizada no início do curso que tratava de Redes de Computadores, Linux e de Programação em Python.

No Estudo Presencial os estudantes resolveram problemas que envolviam a automatização do sincronismo de dados entre dois servidores da rede, agendados de forma periódica e registrando em um arquivo de *log* todas as ocorrências decorrentes do sincronismo.

3.2.8 Semana 08 – 14/01/2018 a 27/01/2018 – Módulo 2 – Tema 06

- Tema 06: Seja o primeiro a saber!

Até este momento no curso trabalhamos formas de resolver problemas diversos relacionados à Redes de Computadores, inclusive agendamento de procedimentos automatizados. Entretanto é muito importante que os responsáveis pela rede sejam avisados sobre a realização dos procedimentos e, em casos mais graves que estes avisos sejam imediatos. Assim, este tema trata da automatização de informações sobre eventos via *e-mail* e Telegram.

No Estudo Autônomo os estudantes resolveram questões envolvendo o envio de *e-mails* através do Gmail e comunicações instantâneas por mensagens via Telegram.

Entretanto, no início desta semana, percebemos uma anomalia que envolvia o funcionamento do aplicativo de envio de mensagens via Telegram. O modelo testado no início do curso não funcionava em arquitetura de 32 bits. Assim, alguns estudantes tiveram dificuldades em dar continuidade aos estudos. Para superarmos este percalço tivemos de pesquisar uma solução para o problema, o que acabou por atrasar a realização do Estudo Presencial 06.

No Estudo Presencial os estudantes resolveram problemas que envolviam a comunicação de eventos por *e-mail* e mensageria instantânea. Também envolviam a maioria dos temas estudados até então: Monitoramento de disponibilidade de equipamentos, manipulação de *strings*, compartilhamento de arquivos, *backups* em rede e agendamento de procedimentos.

A Avaliação Modular 2 inicialmente seria aplicada dia 27/01/2018, mas com o atraso, foi aplicada em 03/02/2018.

3.2.9 Semana 09 – 28/01/2018 a 03/02/01/2018 – Módulo 2 – Preparação e aplicação da Avaliação

Durante esta semana incentivamos a revisão dos Estudos passados, como forma de preparação para a Avaliação Modular.

Após a aplicação da Avaliação Modular 2, aplicada em 03/02/2018, encerramos o curso. O *feedback* sobre a Avaliação Modular 2 foi prestado até o dia 06/02/2018, mesma data em que foram liberados os certificados *online* do curso.

Ao longo do percurso ocorreram alguns desvios envolvendo conhecimentos prévios, dificuldades com o domínio de alguns conteúdos, composição do Laboratório Virtual e evasões. Discutiremos estes desvios na próxima seção.

3.3 ANÁLISE DOS DESVIOS

Entre o planejamento e a execução da nossa proposta de ensino-aprendizagem ocorreram alguns imprevistos que necessitaram de intervenção para que não houvesse grandes prejuízos.

Como um dos pré-requisitos para ingresso no curso era de o estudante já ter cursado alguma disciplina de programação, concluímos que isto seria suficiente para que os estudos

transitassem sem maiores dificuldades quanto a este quesito. Entretanto, mesmo os estudantes que declararam já conhecer a linguagem Python, tiveram problemas. Esta dificuldade gerou uma atenção maior à programação e uma Tutoria presencial foi solicitada com foco exclusivo em estruturas básicas para programação em Python, amenizando o problema.

Na concepção do laboratório virtual, as máquinas que o compunham foram construídas sobre a plataforma 64 (sessenta e quatro) bits do Ubuntu Minimal, sobre o qual foram feitos todos os testes e resolução prévia das tarefas dos Estudos Presenciais e Autônomos. Ao disponibilizarmos a imagem (arquivo “base” para instalação do sistema operacional) para que os estudantes reproduzissem os laboratórios em seus computadores, muitos reclamaram que não conseguiam importar a imagem no Oracle VM Virtualbox porque seus computadores reais que funcionariam com hospedeiros das máquinas virtuais eram de 32 (trinta e dois) bits. Isto gerou a necessidade de reconstrução da imagem para 32 (trinta e dois) bits e redistribuição para *download*, atrasando o desenvolvimento do Estudo Autônomo 01 por parte de alguns estudantes.

Ao alterarmos a plataforma do sistema operacional Linux não previmos que poderia haver algum problema com algum dos aplicativos instalados. Esta situação veio à tona no Estudo Autônomo 6, ocasião em que percebemos que o aplicativo utilizado para envio de mensagens via Telegram funcionava apenas na plataforma de 64 (sessenta e quatro) bits. Por conta deste problema, o Estudo Autônomo 06 foi liberado com uma semana de atraso, pois precisamos encontrar uma solução que cumprisse os mesmos objetivos do antigo aplicativo, o que demandou vários testes.

Alguns estudantes, por conta de compromissos particulares, alegaram não conseguir concluir as tarefas referentes aos Estudos Autônomos nos prazos especificados. Para mitigar o problema, em algumas ocasiões, aumentamos o tempo de “Aquecimento” (que era de trinta minutos) durante alguns Estudos Presenciais para até uma hora, para dar tempo de estes estudantes concluírem suas tarefas atrasadas. Isto fez com que estes Estudos Presenciais se estendessem além do horário previsto.

Na próxima seção apresentamos a análise da validade e os critérios de validação do estudo de caso.

3.4 AVALIAÇÃO DA VALIDADE

Para o estudo de caso proposto, consideramos quatro tipos de validações como

critérios de julgamento sobre a qualidade do projeto de pesquisa: Validade interna, externa, de conclusão e de construção.

3.4.1 Validade interna

Objetiva analisar se os resultados obtidos ocorreram somente em função da atuação de fatores do caso em questão. Caso haja outros fatores atuantes não esperados ou não especificados, a validade interna pode ser comprometida.

No intuito de garantir a validade interna para nossa proposta de ensino-aprendizagem, cujo objetivo é observar se houve melhorias nas habilidades envolvidas no curso, algumas medidas foram tomadas:

- Como os conhecimentos sobre programação que utilizamos no curso envolviam estruturas básicas de programação, exigimos como pré-requisito que os estudantes já tivessem cursado ao menos uma disciplina de programação, pois isso evitaria estudantes sem quaisquer conhecimentos sobre programação e, possivelmente, um desempenho disforme por conta da dificuldade;
- Procuramos instruir cada conteúdo de Redes de Computadores, fornecendo as mesmas oportunidades de aprendizagem a todos, com detalhamentos suficientes para a resolução das tarefas, de forma que os que tivessem menos experiência com redes ou que fossem oriundos de cursos em que esta disciplina não é o foco, pudessem ter oportunidades para desempenhos igualitários;
- Garantimos o acesso igualitário às orientações durante o decorrer do curso, de forma que as dúvidas levantadas por algum estudante em algum momento eram disseminadas para todos os outros estudantes nas Tutorias e Estudos Presenciais.

3.4.2 Validade externa

Para este estudo de caso, nossa intenção é a de realizarmos uma avaliação da proposta de ensino-aprendizagem em um contexto real de sala de aula sem pretensões de generalizações dos dados, o que seria inviável para o contexto da pesquisa. Ainda assim,

acreditamos que outros professores podem beneficiar-se positivamente ao reproduzirem este cenário.

3.4.3 Validade de conclusão

Como neste estudo não consideremos um grupo de controle, não é possível estabelecer relacionamentos estatísticos. Assim, a pesquisa se utilizou dos dados qualitativos para fornecer uma avaliação da proposta de ensino-aprendizagem em um ambiente real de sala de aula.

3.4.4 Validade de construção

Considerando que pode haver mudanças que envolvam os sujeitos da pesquisa e que isso poderá influenciar na replicação dos estudos, algumas medidas foram tomadas para que pudessem ser previstos eventuais desvios. Assim, consideraremos as seguintes medidas para preservar a validade de construção:

- Utilização de várias fontes de evidências durante a coleta de dados;
- Utilização de rubricas que estabelecem os mesmos critérios para avaliação do desempenho dos estudantes;
- Avaliação sobre os conhecimentos prévios dos estudantes quanto à Redes de Computadores e Programação, de forma a garantir o controle sobre possíveis situações de interferência no estudo.

No próximo capítulo, apresentamos os resultados e discussões oriundos da aplicação desta proposta de ensino-aprendizagem.

4 RESULTADOS E DISCUSSÕES

Neste capítulo apresentamos as respostas para nossas quatro questões de pesquisa. As três primeiras seções (seções 4.1, 4.2 e 4.3) são destinadas à análise das habilidades que trabalhamos durante a implementação da proposta de ensino-aprendizagem, com respostas e discussões baseadas nas ações e resultados dos estudantes quando das aplicações dos instrumentos de ensino-aprendizagem, quais sejam: Estudo Presencial, Avaliação Diagnóstica, Avaliação Modular 1 e Avaliação Modular 2. As estruturas destes instrumentos são compostas por questões que funcionam como fonte de análise de desempenho dos estudantes. Para que não haja confusão com as nomenclaturas, chamaremos estas questões de *unidades de atividades*. Os resultados e discussões referentes às habilidades avaliadas nestas três primeiras seções se deram pela aplicação de rubricas, cujos desempenhos foram divididos em quatro níveis: *Avançado*, *Proficiente*, *Próximo de Proficiente* e *Insuficiente* (vide Seção 2.5). Para que um desempenho fosse considerado satisfatório deveria estar nivelado em *Avançado* ou *Proficiente*.

A quarta seção (seção 4.4) é destinada à resposta da quarta questão de pesquisa, que aborda as limitações verificadas durante o processo de concepção e aplicação da proposta de ensino-aprendizagem e foi baseada em uma visão holística sobre sua implementação.

4.1 HABILIDADES PARA IDENTIFICAR SOLUÇÕES PARA PROBLEMAS DE REDES DE COMPUTADORES

Para análise da questão de pesquisa “*Em que aspectos a proposta de ensino-aprendizagem melhora as habilidades dos estudantes para identificar soluções para problemas reais de Redes de Computadores?*” (Habilidade 01), foram reservadas duas unidades de atividades que deveriam ser expressas de forma textual, com a finalidade de avaliar o desempenho dos estudantes em cada um dos instrumentos: Estudos Presenciais, Avaliação Diagnóstica, Avaliação Modular 01 e Avaliação Modular 02.

Dado determinado contexto de um problema do domínio de Redes de Computadores, seguindo uma sequência de etapas, o estudante deveria, para a primeira unidade de atividade, analisar o contexto e identificar o problema de forma assertiva, demonstrando entendimento sobre o mesmo. Para a segunda unidade de atividade, o estudante deveria descrever uma estratégia para solucionar o problema, com detalhamento suficiente sobre a resolução que

denotasse compreensão dos conceitos envolvidos. Para exemplificar esta estrutura, no Quadro 10 apresentamos o contexto do problema e unidades de atividades utilizadas na Avaliação Diagnóstica.

Quadro 10: Contexto e unidades de Atividades – Avaliação Diagnóstica.

Contexto do problema (Âncora):
A empresa onde você trabalha possui 25 (vinte e cinco) Pontos de Acesso sem fio (Rádios Wireless), que se comunicam com um rádio central, via IP. Estes equipamentos são responsáveis por distribuir internet em praças e espaços culturais da cidade. Para minimizar possíveis problemas com energia elétrica que possam indisponibilizar a comunicação nos locais atendidos pelo serviço, cada rádio wireless é conectado a um nobreak. Estes nobreaks são capazes de mantê-los em funcionamento por até 1 (uma) hora, em caso de falta de energia elétrica. Ocorre que muitos usuários estão reclamando nas redes sociais que está havendo muitos problemas de indisponibilidade. O primeiro passo é identificar a frequência de quedas para que os problemas possam ser investigados. Desta forma, seu chefe solicitou um relatório onde constem os equipamentos que estiveram fora do ar, de hora em hora, a cada dia. Este relatório deve ser enviado a você e a ele, por e-mail, ao final de cada dia, de forma que vocês possam lê-lo a cada manhã. Se, em algum momento, mais de 5 (cinco) equipamentos estiverem sem comunicação com a central, uma mensagem instantânea ou um e-mail deve ser enviado imediatamente para você e seu chefe.
Unidade de Atividade 01
Descreva qual(is) o(s) provável(is) problema(s) que você identificou quanto às providências a serem tomadas.
Unidade de Atividade 02
Descreva detalhadamente como você construirá a solução para o monitoramento que auxiliará na solução do(s) problema(s). Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Fonte: Autoria própria.

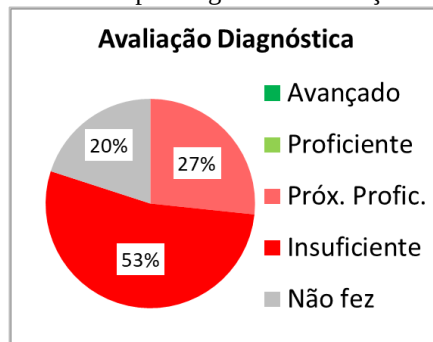
Nosso primeiro instrumento de análise foi a Avaliação Diagnóstica, cujos resultados observados demonstraram que nenhum estudante teve desempenho satisfatório para a Habilidade 01. Os estudantes demonstraram dificuldades quanto ao entendimento do contexto do problema apresentado e, conseqüentemente, na descrição detalhada da solução. O contexto envolvia uma situação em que o estudante era o administrador da rede e a situação indicava que havia quedas de energia em uma rede composta por vários rádios (*wireless access points*) que se comunicavam com um rádio central. Assim, solicitava-se que fosse identificada a frequência com que cada rádio sofria quedas de energia e que fossem enviados relatórios por *e-mail* aos administradores da rede (APÊNDICE F). O problema a ser identificado (unidade de atividade 01) era a ausência de monitoramento dos equipamentos e a descrição da solução (unidade de atividade 02) envolvia a automatização do monitoramento dos rádios em períodos específicos com o emprego do comando PING (protocolo ICMP) ou utilizando o protocolo SNMP e, por fim, envio de mensagens via *e-mail*.

Entretanto, a maioria dos estudantes não conseguiu identificar corretamente o problema, tampouco realizar a descrição da solução. As respostas envolviam suposições alheias ao problema, como sobrecarga de acessos aos rádios, queda de sinal de Internet,

downloads excessivos na rede, má distribuição da rede lógica, interferência de sinal, conflitos de IP, etc.

Na análise dos resultados desta Avaliação Diagnóstica, com a aplicação das rubricas, verificamos que nenhum estudante atingiu resultados satisfatórios, tendo suas respostas classificados nos níveis *Insuficiente* ou *Próximo de Proficiente* (conforme Gráfico 1).

Gráfico 1: H1-Desempenho geral na Avaliação Diagnóstica.

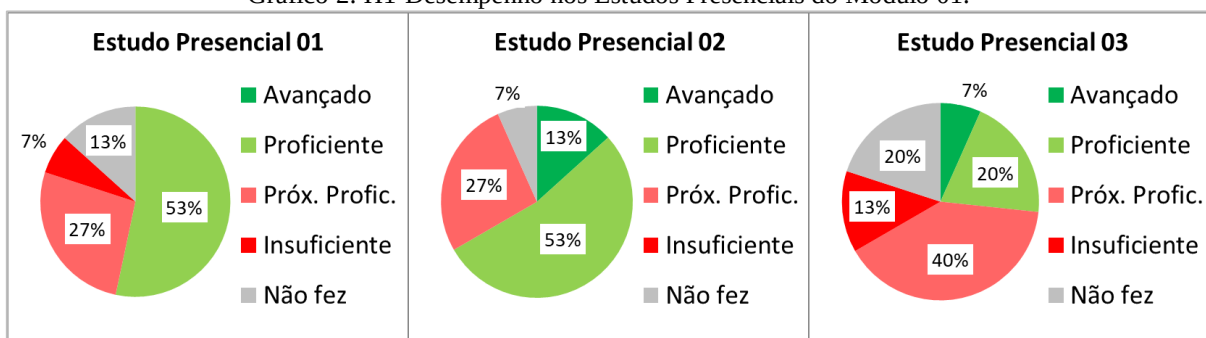


Fonte: Autoria própria.

Pelo conjunto dos resultados obtidos nesta Avaliação Diagnóstica, percebemos que a maioria dos estudantes não conseguia analisar os problemas pensando em uma solução conectada à programação, resumindo-se a aspectos técnicos de redes de computadores.

Com a aplicação do Estudo Presencial 01, notamos uma melhora em comparação com a Avaliação Diagnóstica, pois 53% dos estudantes atingiram desempenho satisfatório (Avançado ou Proficiente). No Estudo Presencial 02 esta evolução também foi percebida, pois o desempenho satisfatório subiu para 63%. Entretanto, com a aplicação do Estudo Presencial 03, houve uma queda de desempenhos satisfatórios para 27%, conforme Gráfico 2.

Gráfico 2: H1-Desempenho nos Estudos Presenciais do Módulo 01.



Fonte: Autoria própria.

Entendemos que esta diferença nos desempenhos ocorreu porque os dois primeiros temas funcionavam como introdução ao monitoramento envolvendo os protocolos SNMP e ICMP, sendo complementares. Já, o terceiro tema possuía um grau de complexidade maior, pois exigia conhecimentos de programação envolvendo operações com *strings* e manipulação

de arquivos, ao que os estudantes demonstraram dificuldades.

Outro problema identificado desde o Estudo Presencial 01, referente à Habilidade 01, foi a forma resumida com que os estudantes responderam às unidades de atividades, o que prejudicou seus desempenhos. Ao longo das tarefas nos Estudos Autônomos e nos Estudos Presenciais, conforme exercitavam a Habilidade 01, os estudantes recebiam os *feedbacks* individuais e em grupo, por *e-mail* e Whatsapp, onde enfatizávamos a necessidade de melhorias nos argumentos utilizados nas identificações e descrições, sem fuga dos contextos dos problemas apresentados. Conforme observado no desempenho da Habilidade 01 ao longo da aplicação dos Estudos Presenciais (Gráfico 2), esta intervenção surtiu efeito positivo. O estudante Everaldo¹⁹ é um bom exemplo desta evolução. Fazemos uma comparação entre suas respostas para os Estudos Presenciais do Módulo 01 no Quadro 11.

No Estudo Presencial 01, unidade de atividade 01, a identificação envolvia problemas com o desempenho do servidor que necessita ser monitorado, entretanto Everaldo entendeu que havia problemas de roteamento. Para a unidade de atividade 02, além de sugerir uma solução fora do contexto, a descrição ficou muito curta. O desempenho, neste caso, para a Habilidade 01 foi considerado *Insuficiente*.

No Estudo Presencial 02, unidade de atividade 01, a resposta evoluiu em comparação com o Estudo Presencial 01, pois o contexto é respeitado e a identificação é mais assertiva. Entretanto, na unidade de atividade 02 há uma dispersão no detalhamento e o estudante Everaldo acaba tentando relatar (de forma confusa) o funcionamento do protocolo SNMP, fugindo do contexto. O desempenho foi considerado *Próximo de Proficiente*.

No Estudo Presencial 03, unidade de atividade 01, a resposta é genérica e acaba prejudicada por conta disso. Na unidade de atividade 02 há uma melhora em comparação com os Estudos Presenciais anteriores, pois o estudante consegue visualizar um cenário mais complexo e objetivo dentro do contexto apresentado. Faltou detalhar como seria feita a programação. Neste caso o desempenho foi considerado *Proficiente*.

Ao longo destes primeiros três temas a maioria dos estudantes já era capaz de identificar soluções envolvendo: monitoramento de redes com os protocolos ICMP e SNMP; monitoramento de estados de componentes físicos remotos como discos, partições e memórias; monitoramento de ações de usuários; manipulação de arquivos; criação automática de arquivos de *logs*. A ambientação com a linguagem Python para resolução de problemas de redes progredia, de forma que a utilização de estruturas de programação e chamadas de

¹⁹ Os nomes dos estudantes citados são fictícios.

bibliotecas se tornavam mais naturais, proporcionando descrições mais robustas, assertivas e contendo cada vez mais detalhamento sobre a construção das soluções.

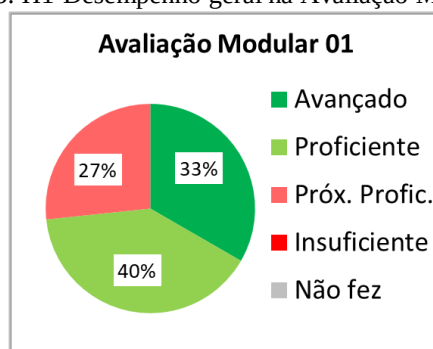
Quadro 11: H1-Respostas do estudante Everaldo para os Estudos Presenciais.

Contexto do problema (Âncora): Estudo Presencial 01: Na empresa em que você trabalha, os usuários estão reclamando que, durante seus expedientes, em alguns momentos, têm dificuldades para acessar a máquina Servidor2, devido à lentidão. Relatam que, geralmente, ocorrem em momentos de maior pico de trabalho, quando todos os computadores estão em operação. O Servidor2 funciona como servidor de arquivos e está localizado fisicamente no mesmo prédio, atrás de um firewall. A rede não apresenta gargalos em nenhum momento. Observando a ocupação dos discos, verificou-se que não há problemas de espaço nem de lentidão nos acessos. Estudo Presencial 02: Por medida de segurança, você verificou que é necessário que as máquinas Servidor1 e Servidor03 tenham suas partições e discos monitorados para prevenir estouros de espaços de armazenamento, pois um servidor parado pode significar prejuízos financeiros e de imagem para a empresa. Estudo Presencial 03: No Servidor1, como é padrão nas máquinas Ubuntu Linux, há um arquivo que guarda os registros ou logs de autenticação. O arquivo fica no seguinte caminho: /var/log/auth.log. Neste servidor há um usuário chamado user2, que não possui a senha de root. Você desconfia que usuário user2 efetuou tentativas para descobrir a senha de root. Há também a desconfiança de que o usuário apagou seus comandos do history. É necessário averiguar no arquivo auth.log se há registros de tentativas de escalada de privilégios do user2. Caso existam tentativas registradas, será necessário apresentá-las aos seus superiores, através de relatório. (Para mudar para user2, no console, digite: su user2. A senha do user2 é user2@. Simule tentativas de acesso ao root com senhas incorretas)
Unidade de Atividade 01 - Descreva qual(is) o(s) provável(is) problema(s) que você identificou e que podem ser monitorados. Estudo Presencial 01: Os problemas são a lentidão da rede em horários de pico. Temos que monitorar a rede para saber se existe algum problema de roteamento (<i>sic</i>). Estudo Presencial 02: Pode ocorrer perda de disponibilidade do sistema. Caso aja indisponibilidade com a superlotação do disco pode haver perda de dinheiro para uma empresa, podendo ocasionar até em uma falência (<i>sic</i>). Estudo Presencial 03: Existe um problema de desconfiança pela falta de monitoramento (<i>sic</i>).
Unidade de Atividade 02 - Descreva detalhadamente como você construirá a solução para o monitoramento que auxiliará na solução do(s) problema(s). Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajude a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível. Estudo Presencial 01: Para resolver isso podemos utilizar um script utilizando o comando ping. Esse script vai mostra se o servidor de arquivo está acessível ou não (<i>sic</i>). Estudo Presencial 02: Bom, para solucionar esse problema podemos utilizar o protocolo SNMP para fazer o monitoramento dos Servidores. Cada máquina possui uma tabela com os respectivos caminhos para chegar a um determinado recurso de hardware, chamamos essa tabela de MIB, dentro dessas tabelas temos os endereços de cada dispositivo do hardware a ser monitorado, esses endereços denominamos de OIDs. Dentro do script, que será feito para fazer todo esse monitoramento, usaremos cada Caminho das OIDs referente ao disco rígido para fazer todo esse monitoramento (<i>sic</i>). Estudo Presencial 03: Para resolver esse problema é necessário fazer o monitoramento em tempo real do arquivo .bash_history (já que existe a suspeita de que o user2 esteja apagando o seu histórico). para isso seria necessário acessar o user2 via protocolo SSH, para podermos executar o comando "tail -f /home/user2/.bash_history" . também é necessário que se armazene as tentativas de acesso do root através de um script, para servir de prova (<i>sic</i>).

Fonte: Autoria própria.

Na Avaliação Modular 01 a evolução da Habilidade 01 seguiu nesta crescente e os desempenhos observados chegaram a 73% de respostas classificadas como satisfatórias, conforme demonstrado no Gráfico 3.

Gráfico 3: H1-Desempenho geral na Avaliação Modular 01.



Fonte: Autoria própria.

Como exemplo de desempenho na Avaliação Modular 01, novamente apresentamos as respostas do estudante Everaldo, que teve um desempenho satisfatório, conforme disposto no Quadro 12.

Quadro 12: H1-Respostas do estudante Everaldo para a Avaliação Modular 01.

Contexto do problema (Âncora): Você tem receio de que, futuramente, possa haver problemas com estouro de partições nos servidores. Por medidas de segurança, você decidiu monitorar o percentual de utilização de disco em cada um deles e colocar os resultados em um relatório, criado no Servidor1. Em seguida o arquivo deve ser copiado para ao Servidor3, com uso do protocolo NFS. Para o monitoramento do percentual de uso de discos, considere a seguinte OID: “1.3.6.1.4.1.2021.9.1.9.1”.
Unidade de Atividade 01 - Descreva qual(is) o(s) provável(is) problema(s) que você identificou e que podem ser monitorados. Pode-se notar que existe um problema pela falta de monitoramento de disco, se os discos não forem monitorados pode haver um estouro de partição, ocasionando indisponibilidade do próprio sistema (sic).
Unidade de Atividade 02 - Descreva detalhadamente como você construirá a solução para o monitoramento que auxiliará na solução do(s) problema(s). Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajude a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível. Para resolver essa problemática utilizaremos um script para buscar as informações de armazenamento de disco, depois que obtivermos as informações atribuiremos elas a um relatório para ser analisado, posteriormente iremos montar uma partição compartilhada entre os servidores e copiar o relatório para a partição compartilhada, por fim iremos desmontar a partição (sic).

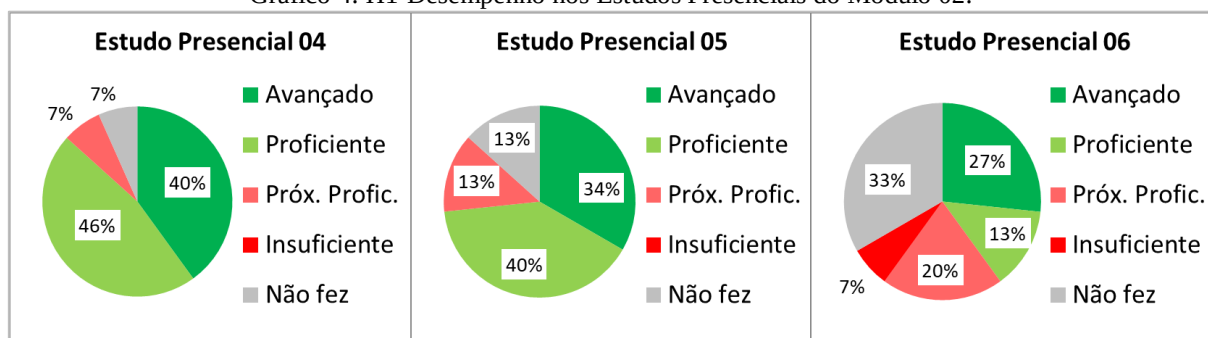
Fonte: Autoria própria.

Nas respostas do estudante Everaldo para a unidade de atividade 01, percebemos uma evolução na qualidade da resposta, em comparação com o apresentado no Estudo Presencial 03, pois o contexto foi bastante respeitado e o problema foi identificado com assertividade. O desempenho na unidade de atividade 02 também merece destaque, ainda que pudesse ter acrescentado mais detalhes. O desempenho foi classificado como *Proficiente*.

No Módulo 02 observamos continuidade na evolução da Habilidade 01. Nos três Estudos Presenciais que se seguiram observamos que os estudantes adquiriam mais confiança na resolução dos problemas. O desempenho considerado satisfatório no Estudo Presencial 04

foi de 86% e no Estudo Presencial 05 foi de 74%, conforme Gráfico 4.

Gráfico 4: H1-Desempenho nos Estudos Presenciais do Módulo 02.



Fonte: Autoria própria.

No Estudo Presencial 06 observamos uma dissonância com os Estudos Presenciais anteriores. Houve um terço de estudantes ausentes e apenas 40% de desempenhos considerados satisfatórios. Apesar desta anomalia, a maioria dos estudantes foi capaz de identificar problemas envolvendo serviços de rede, como: automatização de procedimentos de *backup* com geração de relatório de execução; envios de *e-mails* com informações sobre a execução de procedimentos; sincronismo de dados entre servidores; envio de alertas imediatos utilizando o aplicativo Telegram. Como exemplo do desempenho da turma apresentamos as respostas para do estudante Ivan no Estudo Presencial 06.

Quadro 13: H1-Respostas do estudante Ivan para o Estudo Presencial 06.

Contexto do problema (Âncora): Você entrou recentemente na empresa e imediatamente verificou que não são feitos <i>backups</i> dos servidores. Preocupado com a situação, decidiu iniciar pelas configurações (/etc), do Servidor1. Os <i>backups</i> devem ser feitos todos os dias, de madrugada, compactados e armazenados no próprio Servidor1, por enquanto. Também é necessário verificar todas as manhãs o resultado do <i>backup</i> executado na madrugada: se o <i>backup</i> foi executado; em que horário começou e em que horário foi finalizado; além da verificação da execução do procedimento de empacotamento e compactação, se foram ou não executados com sucesso.
Unidade de Atividade 01 - Descreva qual(is) o(s) provável(is) problema(s) que você identificou quanto à verificação das partições e envio de mensagens. A verificação do percentual de memória dos servidores é de suma importância, pois, caso atinja a capacidade máxima dos discos os servidores irão parar os serviços (sic).
Unidade de Atividade 02 - Descreva detalhadamente como você construirá a solução para o monitoramento que auxiliará na solução do(s) problema(s). Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajude a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível. Parte 1: Serão utilizados as bibliotecas subprocess, smtplib e os recursos que o smtp tem disponível para a verificação dos host. Parte 2: com a criação de duas variáveis que irão receber o percentual dos dois servidores e será inserido esses resultados em outra variável da qual se tornará um arquivo.log. Parte 3: usando as condicionais IF/ELSE caso o percentual seja superior a 61% será enviado uma mensagem via telegram no celular do administrador, isso será possível usando os recursos do "telegram-notify". Parte 4: O arquivo.log que receberá os resultados das variáveis dos servidores 1 e 3, será listado usando o comando "cat" e posteriormente será enviado esse conteúdo listado via <i>e-mail</i> do administrador. Parte 5: O final do código conterá as configurações de <i>e-mail</i>. Parte 6: Com o uso do contrab será programado para ser executado o programa em tela todo dia as 23hrs (sic).

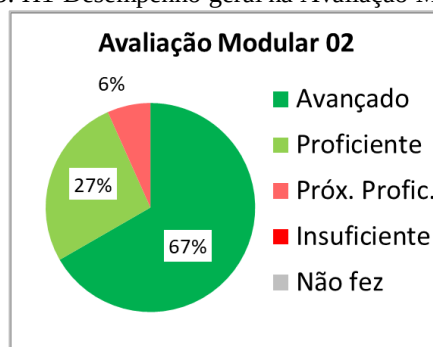
Fonte: Autoria própria.

É possível perceber uma melhor consistência na resposta do estudante Ivan para a

unidade de atividade 01, devido à objetividade e respeito ao contexto do problema apresentado. Este perfil foi verificado na maioria das respostas dos outros estudantes em todos os Estudos Presenciais do Módulo 02. Da mesma forma, também percebemos certa consistência na maioria das respostas da unidade de atividade 02. O modelo de resposta adotado pelo estudante Ivan e por outros colegas consistia em dividir a resposta em partes. Este modelo foi sugerido durante os *feedbacks* fornecidos aos estudantes ao longo do Módulo 02 como uma alternativa para melhor detalhar a descrição da solução. Observamos na resposta do estudante Ivan um detalhamento que envolve conteúdos de Redes de Computadores, de Programação e de sistema operacional Linux. Este foi um padrão observado na maioria das respostas dos estudantes para a unidade de atividade 02 neste segundo módulo.

O melhor desempenho para a Habilidade 01 foi observado na Avaliação Modular 02, em que 94% dos estudantes obtiveram desempenho considerado satisfatório para suas respostas, conforme disposto no Gráfico 5.

Gráfico 5: H1-Desempenho geral na Avaliação Modular 02.



Fonte: Autoria própria.

Como exemplo de desempenhos obtidos pelos estudantes na Avaliação Modular 02, apresentamos as respostas construídas pelo estudante Kelvin, conforme disposto no Quadro 14.

O bom desempenho apresentado nos resultados do estudante Kelvin foi observado na maioria das unidades de atividades 1 e 2 resolvidas pelos estudantes da turma. As respostas foram concisas, permanecendo dentro do contexto apresentado e com detalhamento suficiente para orientar o posterior desenvolvimento da solução na linguagem Python. Nesta avaliação a maioria os estudantes demonstrou habilidade para analisar e entender o contexto de um problema do domínio de Redes de Computadores e também para descrever com riqueza de detalhes uma solução, de acordo com as exigências, evidenciando o domínio da Habilidade 01.

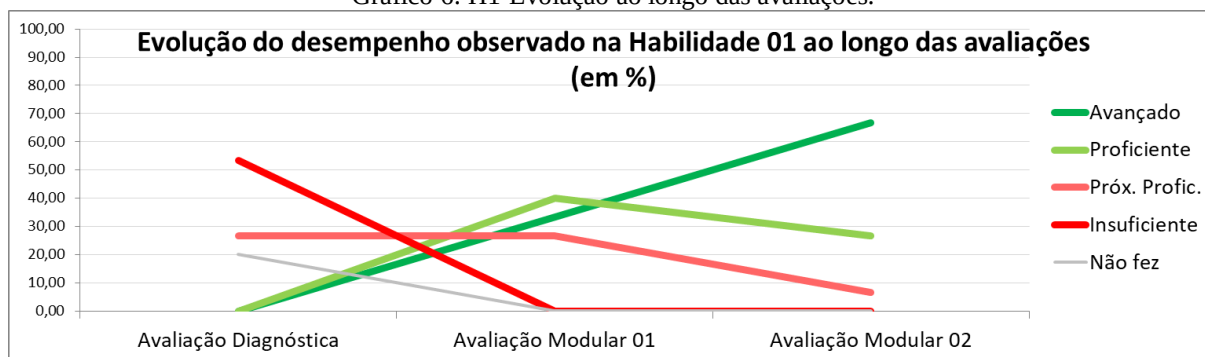
Quadro 14: Respostas do estudante Kelvin para a Avaliação Modular 01.

Contexto do problema (Âncora): Os usuários da empresa em que você trabalha utilizam o diretório /home/user1/documentos/ para armazenar documentos confidenciais da empresa. Desta forma, há a necessidade da execução de um <i>backup</i> diário, às 23 horas, com compactação de arquivos. Também é necessário enviar este <i>backup</i> para o Servidor3. Caso a montagem no Servidor3 e o <i>backup</i> ocorram com sucesso, deve ser enviada uma mensagem para seu <i>e-mail</i>. Caso haja problemas na montagem no Servidor3 ou no <i>backup</i>, além da mensagem por <i>e-mail</i>, é necessário que seja enviado um alerta via Telegram para seu celular.
Unidade de Atividade 01 - Descreva qual(is) o(s) provável(is) problema(s) que você identificou quanto à necessidade de <i>backup</i> apresentado no contexto. Como informado, os dados que estão no caminho /home/user1/documentos/ do servidor1 são sensíveis, portanto, a perda ou o vazamento desses dados causaria enormes problemas para a empresa, como percas financeiras, a credibilidade da empresa. Portanto, a política de <i>backup</i> sempre é necessária para evitar casos como os citados agora, então é de extrema importância salvar essas informações no servidor 3. (sic).
Unidade de Atividade 02 - Descreva detalhadamente como você construirá a solução para o monitoramento que auxiliará na solução do(s) problema(s). Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajude a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível. ETAPA 1: importaria as seguintes bibliotecas: subprocess (para utilizar o comando call a fim de receber um resultado da execução), time (para informar a hora da execução do programa) e smtplib (para automatizar o envio de alertas ao meu <i>e-mail</i>). ETAPA 2: criaria três variáveis para receber codificação ASCII e carga de informações. ETAPA 3: criaria mais variáveis a fim de receber os comandos do processo de ponto de montagem e desmontagem entre o servidor 1 e 3, além de armazenar também, os comandos para compactação e gravação do diretório /home/user1/documentos no ponto de montagem entre o servidor 1 e 3. ETAPA 4: criaria uma estrutura de condição if else, para elaborar as condições de envio das mensagens, ou seja, caso a montagem e compactação funcione, caíra numa condição, ou caso não haja falha, caíra em outra condição. ETAPA 5: utilizaria os comandos pertencentes a biblioteca smtplib para o envio das mensagens produzidas na ETAPA 4. ETAPA 6: abriria o crontab para fazer a execução desse programa a cada 23 horas, diariamente. A linha de comando ficaria assim: 0 23 * * * /usr/bin/python3 /home/user1/provafinal.py (sic).

Fonte: Autoria própria.

De forma complementar, no Gráfico 6, apresentamos uma visão resumida sobre a evolução dos estudantes ao longo das avaliações da Habilidade 01.

Gráfico 6: H1-Evolução ao longo das avaliações.



Fonte: Autoria própria.

Um bom desempenho na Habilidade 01 é muito importante para o desenvolvimento da Habilidade 02, sobre a qual discutimos na próxima seção.

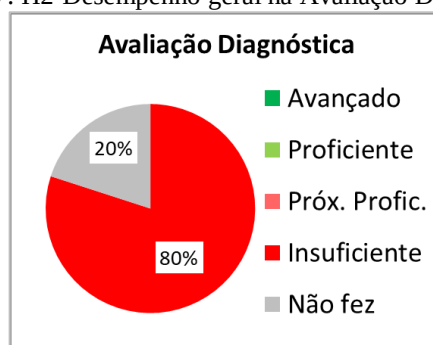
4.2 HABILIDADES PARA AUTOMATIZAR SOLUÇÕES PARA PROBLEMAS DE REDES DE COMPUTADORES

Para análise da questão de pesquisa “*Em que aspectos a proposta de ensino-aprendizagem melhora as habilidades dos estudantes para automatizar soluções por meio de programação para problemas de Redes de Computadores?*” (Habilidade 02), foi destinada uma unidade de atividade que deveria ser expressa no formato de *script* de programação em Python. Os instrumentos analisados foram: Estudo Presencial, Avaliação Diagnóstica, Avaliação Modular 01 e Avaliação Modular 02. Para responder a esta única unidade de atividade o estudante deveria levar em consideração o mesmo contexto de problema utilizado para responder as duas unidades de atividades anteriores. Inclusive, o *script* de programação desenvolvido deveria seguir a descrição da solução feita anteriormente, implementando uma solução completa e correta para o problema, com lógica de programação clara e bem estruturada, empregando boas práticas de programação, como comentários relevantes, nomes de variáveis e funções significativas, ausência de código morto, etc.

Na Avaliação diagnóstica nenhum dos estudantes que realizou a unidade de atividade relativa à Habilidade 02 alcançou desempenho satisfatório em suas respostas, ou seja, nenhum atingiu os níveis *Proficiente* ou *Avançado*, conforme Gráfico 7.

A fim de aproveitar os conhecimentos prévios dos estudantes em programação, deixando-os livres para utilizarem suas lógicas de programação, excepcionalmente, na Avaliação Diagnóstica os estudantes poderiam utilizar qualquer linguagem de programação. Entretanto, apesar de todos já terem cursado ao menos uma disciplina de programação, as respostas não demonstravam domínio sobre a linguagem escolhida e indicavam dificuldades com lógica de programação. Poucos estudantes conseguiram desenvolver alguma estrutura do início ao fim, pois a maioria apenas iniciou uma tentativa e a abandonou sem conclusão.

Gráfico 7: H2-Desempenho geral na Avaliação Diagnóstica.



Fonte: Autoria própria.

Conforme observamos na seção anterior (vide Quadro 10), o contexto do problema apresentado na Avaliação Diagnóstica predizia que houvesse automatização do monitoramento dos rádios em períodos específicos com o emprego do comando PING (protocolo ICMP) ou utilizando o protocolo SNMP e, por fim, envio de mensagens via *e-mail*. Entretanto, os programas que os estudantes apresentaram como resposta nesta avaliação para a Habilidade 02 não condiziam com o contexto, possuindo estruturas confusas. Um exemplo de resposta é disponibilizado na Figura 7.

Figura 7: H2-Avaliação Diagnóstica – Exemplo de resolução dos estudantes.

```

int[] relatorio = new int[24];
int iP, iPcon, hora;
String envia Mensagem;
if (iPcon >= 5)
{
    envia Mensagem relatorio = Console.WriteLine("Ata");
}
else { relatorio = new int[24]; while (iP = iPcon) {
    if (hora <= 00h)
    {
        for (int i=0; i <= 24; i++)
        {
            relatorio[i] = iP;
        }
    }
    else { envia mensagem = relatorio[i]; }
}
}

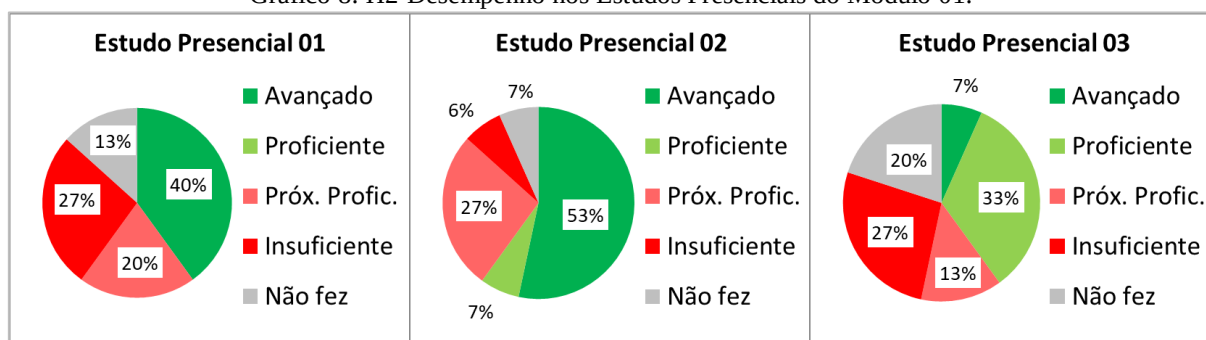
```

Fonte: Autoria própria.

No programa apresentado pelo estudante não há qualquer menção a monitoramento de equipamentos ou envio de mensagens. A lógica utilizada possui equívocos, como a inserção de um laço *for* para fazer a leitura de um vetor chamado *relatório*, entre outras estruturas confusas. Construções com estas características foram identificadas em todos os programas escritos pelos estudantes na Avaliação Diagnósticas.

Este cenário mudou com o início do curso e com a aplicação dos RA e os resultados verificados nos Estudos Presenciais mostraram boa evolução na Habilidade 02, conforme Gráfico 8.

Gráfico 8: H2-Desempenho nos Estudos Presenciais do Módulo 01.



Fonte: Autoria própria.

No Estudo Presencial 01 os estudantes tiveram o primeiro contato com a linguagem Python e com o Laboratório Virtual. Também introduzimos estruturas básicas de programação em Python e os protocolos SNMP e ICMP para monitoramento simples de Redes de Computadores. Apesar de percebermos uma considerável melhora no desempenho da Habilidade 02 em comparação com os resultados da Avaliação Diagnóstica, menos da metade (40%) dos estudantes atingiram resultados satisfatórios. Estes estudantes eram capazes de desenvolver programas envolvendo importação de bibliotecas, conversões de tipos de dados, utilizavam corretamente as funções das bibliotecas importadas e exploravam as informações fornecidas pelo sistema operacional Linux. Tais programas eram apresentados com estruturação correta e a lógica de programação utilizada se apresentava concisa e refletia a solução esperada para o contexto do problema apresentado. Em comparação com a Avaliação Diagnóstica, percebemos melhora no desenvolvimento dos programas apresentados pelos estudantes, mesmo nos daqueles que não alcançaram resultados satisfatórios.

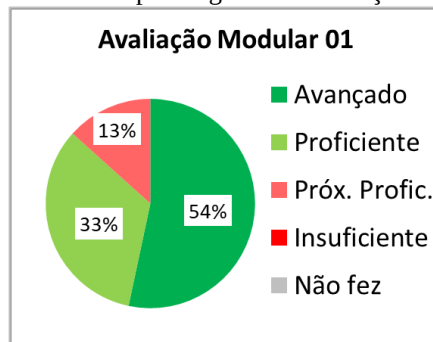
No Estudo Presencial 02 os resultados considerados satisfatórios alcançaram 60%. Atribuímos esta melhora à continuidade dos estudos que naturalmente facilitou a ambientação com nossa proposta de ensino-aprendizagem. Como a intenção neste Estudo Presencial 02 era justamente esta ambientação, o mesmo foi elaborado como uma sequência direta do Estudo Presencial 01. Como os assuntos abordados também envolviam monitoramento com o protocolo SNMP, os estudantes conseguiam desenvolver com mais destreza e segurança seus programas.

No Estudo Presencial 03 acrescentamos manipulação de *strings* e de arquivos em Python, pois além de monitorar elementos dentro da rede, os estudantes deveriam automatizar a criação de relatórios em função da resolução dos problemas. Estes assuntos foram recebidos com dificuldade pelos estudantes, ao ponto de, ao final do Estudo Presencial 03, solicitarem uma Tutoria Presencial extra para poderem assimilá-los melhor. Esta dificuldade se refletiu

nos resultados observados e somente 40% dos estudantes obtiveram níveis considerados satisfatórios.

Na Avaliação Modular 01 os resultados para a Habilidade 02 se mostraram melhores, pois 87% dos estudantes obtiveram desempenhos considerados satisfatórios, conforme Gráfico 9.

Gráfico 9: H2-Desempenho geral na Avaliação Modular 01.

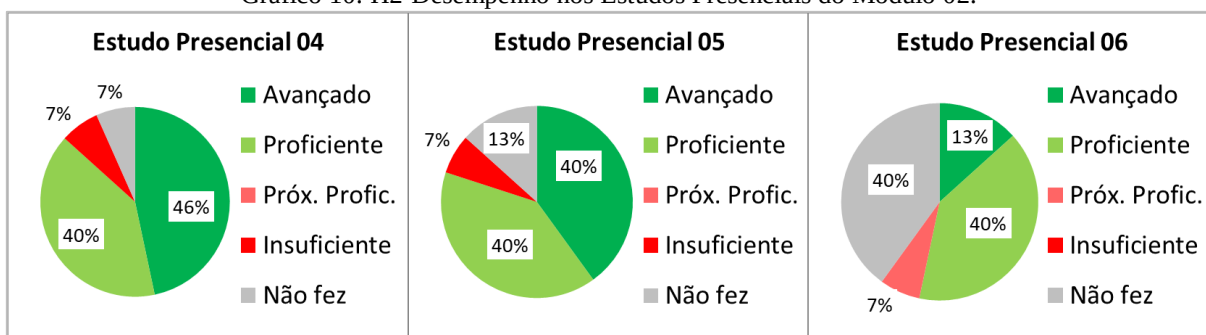


Fonte: Autoria própria.

Na Avaliação Modular 01 os estudantes foram capazes de desenvolver soluções em Python para automatizar o monitoramento de desempenho remoto de componentes físicos de servidores da rede, com criação de relatório de execução e envio do arquivo para outro servidor da rede via compartilhamento remoto.

No Módulo 02 percebemos certa constância no desempenho para a Habilidade 02, pois no Estudo Presencial 04 constatamos que 86% dos estudantes tiveram suas unidades de atividades classificadas em níveis satisfatórios. No Estudo Presencial 05 este percentual atingiu 80%. Contudo, no Estudo Presencial 06 houve uma anomalia, pois 40% dos estudantes deixaram de comparecer. Ainda assim, 53% dos estudantes conseguiram resultados satisfatórios, conforme Gráfico 10.

Gráfico 10: H2-Desempenho nos Estudos Presenciais do Módulo 02.



Fonte: Autoria própria.

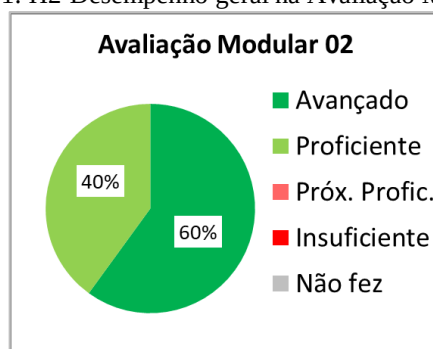
Ao questionarmos os estudantes faltosos, recebemos justificativas variadas como: problemas de saúde seu ou familiar, viagens e trabalho. Segundo os relatos, nenhuma das

justificativas foi ocasionada por qualquer insatisfação com o curso. Assim tratamos o fato como coincidência. Nos *feedbacks* aos estudantes durante o período que separou o último Estudo Presencial da Avaliação Modular 02 incentivamos e cobramos que os estudantes trabalhassem não só o Estudo Presencial 06, mas todos os anteriores. Pedimos que refizessem as atividades e que reportassem quaisquer dificuldades.

Durante os Estudos Presenciais do Módulo 02 os estudantes eram capazes de criar *scripts* na linguagem Python de forma muito mais fluida, pois, diferentemente do que observamos no Módulo 01, o tempo era dedicado muito mais a entender o problema de Redes de Computadores e encontrar sua solução do que com os recursos de Programação como sintaxe e bibliotecas.

Na Avaliação Modular 02 todos os estudantes (100%) tiveram seus desempenhos na unidade de atividade da Habilidade 02 classificados como satisfatórios, com destaque para 60% de classificações no nível *Avançado*, conforme Gráfico 11.

Gráfico 11: H2-Desempenho geral na Avaliação Modular 02.



Fonte: Autoria própria.

Os programas que os estudantes foram capazes de desenvolver para a unidade de atividade da Avaliação Modular 02 se mostraram bem mais concisos, utilizando com boa assertividade os recursos de programação para resolverem os problemas de Redes de Computadores propostos. O contexto do problema apresentado na Avaliação modular 02 é descrito no Quadro 15.

Quadro 15: H2-Avaliação Modular 02-Contexto do problema proposto (Âncora).

Os usuários da empresa em que você trabalha utilizam o diretório `/home/user1/documentos/` para armazenar documentos confidenciais da empresa. Desta forma, há a necessidade da execução de um *backup* diário, às 23 horas, com compactação de arquivos. Também é necessário enviar este *backup* para o Servidor3. Caso a montagem no Servidor3 e o *backup* ocorram com sucesso, deve ser enviada uma mensagem para seu *e-mail*. Caso haja problemas na montagem no Servidor3 ou no *backup*, além da mensagem por *e-mail*, é necessário que seja enviado um alerta via Telegram para seu celular.

Fonte: Autoria própria.

Considerando este contexto, apresentamos na Figura 8 o script desenvolvido pelo

estudante Kelvin para a unidade de atividade referente à Habilidade 02 na Avaliação Modular 02.

Figura 8: H2-Avaliação Modular 02-Programa desenvolvido pelo estudante Kelvin.

```

1 #coding=utf-8
2 #27/01/2018
3 #PROGRAMA PARA COMPACTAR OS ARQUIVOS DO SERVIDOR 1 E ENVIÁ-LOS AO SERVIDOR 3, ALÉM DE ENVIAR ALERTAS AO E-MAIL E TELEGRAM
4 #O PROGRAMA SERÁ EXECUTADO DIARIAMENTE ÀS 23 HORAS ATRÁVES DE UMA LINHA DE COMANDO QUE SERÁ INCLuíDA NO CRONTAB.
5 #O COMANDO A SER EXECUTADO NO CRONTAB É: 0 23 * * * /usr/bin/python3 /home/user1/████████.py
6 #chama as bibliotecas
7 import subprocess, time, smtplib
8 #abre uma variável que inicia a carga de informações com a codificação ascii
9 mensagem = '\r\n'
10 #abre duas variáveis para carga de informações
11 celularMSG1 = ''
12 celularMSG3 = ''
13 #concatena com a mensagem anterior e armazena a data de início do programa
14 mensagem = mensagem + "Data e Horário da checagem: " + time.strftime('%d/%m/%Y %H:%M:%S') + '\r\n'
15 #variável que recebe o caminho do diretório remoto.
16 diretorioRemoto = "10.0.10.13:/home/user1/compartilhado/"
17 #variável que recebe o ponto de montagem na máquina local.
18 pontoMontagem = "/mnt/"
19 #monta o diretório remoto no ponto de montagem local.
20 resultado1 = subprocess.call(["mount", diretorioRemoto, pontoMontagem])
21 #guarda o caminho e o nome do arquivo .tar, com data e hora
22 arquivoTar = "/home/user1/compartilhado/mnt" + time.strftime("%Y-%m-%d_%H:%M:%S") + "-backup-documentos.tar"
23 #guarda o caminho do diretório que será compactado.
24 diretorioRemoto = "/home/user1/documentos/"
25 #compacta e junta os arquivos que estiverem no diretorioRemoto
26 #e os grava no diretório /mnt como arquivo tar e guarda o comando
27 resultado2 = subprocess.call(["tar", "cvfz", arquivoTar, diretorioRemoto])
28 #desmonta o ponto de montagem na máquina local.
29 resultado3 = subprocess.call(["umount", pontoMontagem])
30 #Verifica se o estado da execução é igual a zero O comando funcionou
31 #ou senão, indica que houve um problema no comando.
32 if resultado1 == 0:
33     print ("Montagem foi bem sucedida \r\n")
34     mensagem = mensagem + "Montagem foi bem sucedida \r\n"
35 else:
36     print ("Houve algum problema na montagem \r\n")
37     celularMSG1 = "Houve um problema na montagem \r\n"
38     #Envia a mensagem ao telegram
39     subprocess.check_output(["telegram-notify","--success","--text", celularMSG1])
40 if resultado2 == 0:
41     print ("Compactou com sucesso \r\n")
42     mensagem = mensagem + "Compactou com sucesso \r\n"
43 else:
44     print ("Houve algum problema na compactagem \r\n")
45     celularMSG3 = "Houve um problema na compactagem \r\n"
46     #Envia a mensagem ao telegram
47     subprocess.check_output(["telegram-notify","--success","--text", celularMSG3])
48 #indica qual servidor SMTP será usado
49 servidorSMTP = smtplib.SMTP('smtp.gmail.com', 587)
50 #invoca a criptografia TLS
51 servidorSMTP.starttls()
52 #indica os usuário e senha do email
53 servidorSMTP.login('login', 'senha')
54 #envia a mensagem
55 ##### Acrescentei .encode('utf-8' na variável mensagem, pois estava dando erro de tipo
56 servidorSMTP.sendmail('login', 'login@gmail.com', mensagem.encode('utf-8'))
57 #encerra a conexão entre o servidor SMTP
58 servidorSMTP.quit()

```

Fonte: Autoria própria.

Neste *script* podemos perceber a preocupação com as boas práticas, pois as variáveis possuem nomes significativos, os comentários são suficientemente informativos e todo o código gerado é importante para a efetiva execução do programa e atendimento dos requisitos do contexto do problema. A codificação é bem cadenciada e as instruções denotam domínio da linguagem de programação Python, compreensão do contexto do problema envolvendo Redes de Computadores e conhecimento sobre o sistema operacional Linux.

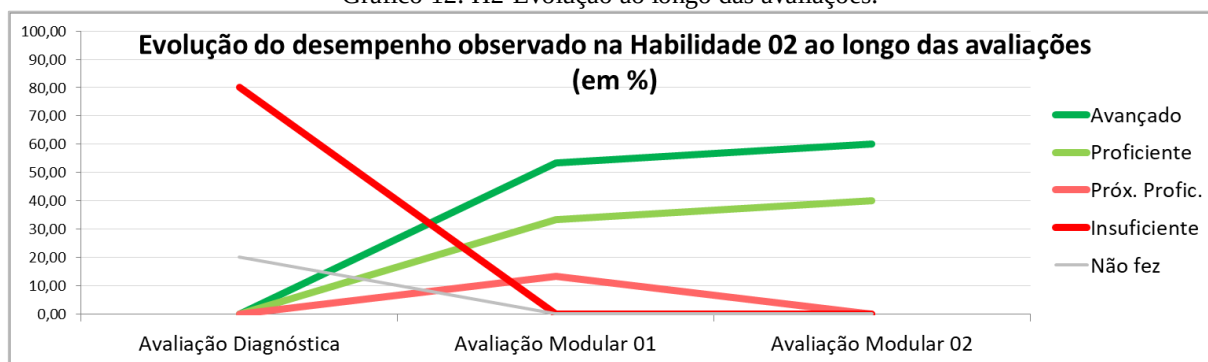
Por estas características, este *script* desenvolvido pelo estudante Kelvin foi classificado como *Avançado* e representa a maturidade que esperávamos que os estudantes alcançassem no decorrer do curso. A maioria dos estudantes foi capaz de produzir códigos

com qualidade semelhante e aqueles que apresentaram codificações com problemas em algum dos requisitos foram classificados no nível *Proficiente*, ainda dentro do desempenho considerado satisfatório.

Ao compararmos este padrão qualitativo de programação observado nesta Avaliação Modular 02 com o padrão que encontramos na Avaliação Diagnóstica, percebemos muita diferença e podemos aferir a evolução abrangida ao longo dos Estudos Presenciais. Esta evolução não se configura apenas na estrutura, mas na compreensão do problema e na capacidade de visualização de soluções para problemas do domínio de Redes de Computadores envolvendo Programação.

Como uma visão resumida sobre a evolução dos estudantes ao longo das avaliações da Habilidade 02, apresentamos o Gráfico 12.

Gráfico 12: H2-Evolução ao longo das avaliações.



Fonte: Autoria própria.

É muito importante que o estudante domine as Habilidades 01 e 02, entretanto é necessário exceder o perímetro dos problemas apresentados e ser capaz de associar as soluções encontradas com soluções de outros possíveis problemas que possam surgir. Isto envolve a Habilidade 03, sobre a qual discutimos na próxima seção.

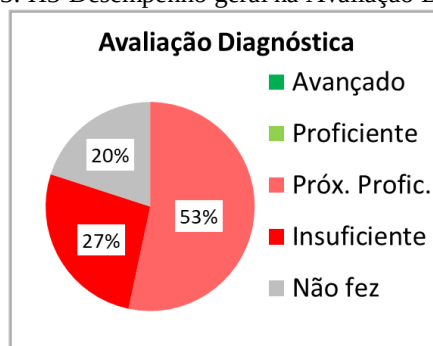
4.3 HABILIDADES EM ASSOCIAR AS SOLUÇÕES ENCONTRADAS COM OUTRAS POSSÍVEIS SOLUÇÕES DE OUTROS PROBLEMAS DE REDES

Para análise da questão de pesquisa “*Em que aspectos a proposta de ensino-aprendizagem melhora as habilidades dos estudantes em associar a solução encontrada com outras possíveis soluções de outros problemas de redes?*” (Habilidade 03), foram destinadas duas unidades de atividades que deveriam ser expressas no formato textual. Na primeira unidade de atividade reservada para análise da Habilidade 03 o estudante era confrontado com

três novos contextos de problemas e deveria analisá-los a fim de identificar qual deles poderia ser solucionado com recursos mais semelhantes aos utilizados para resolver o contexto do problema apresentado anteriormente. Na segunda unidade de atividade o estudante deveria descrever uma solução para o problema associado, nos mesmos moldes que já havia feito para a Habilidade 01, seguindo os mesmos critérios, ou seja, desenvolver uma estratégia para solucionar o problema, com detalhamento suficiente sobre a resolução e que denotasse compreensão dos conceitos envolvidos sobre Programação, Redes de Computadores e sistema operacional Linux. Como havia dependência entre estas duas unidades de atividades, se o estudante se equivocasse na associação, consequentemente prejudicaria a descrição. Os instrumentos analisados foram: Estudo Presencial, Avaliação Diagnóstica, Avaliação Modular 01 e Avaliação Modular 02.

Na Avaliação Diagnóstica, da mesma forma que na Habilidade 01 e na Habilidade 02, nesta Habilidade 03 também constatamos resultados abaixo do satisfatório, pois nenhum conjunto de unidades de atividades foi classificado nos níveis *Proficiente* ou *Avançado*. É importante observar que aproximadamente metade dos estudantes conseguiu associar as soluções dos problemas de forma correta, ainda que não tenham conseguido justificar o porquê desta associação nem descrever uma solução para o problema associado. Para as respostas destes estudantes que conseguiram este acerto mínimo atribuímos o nível de conceito *Próximo de Proficiente* e para o restante da turma, por não terem nenhum acerto considerado, atribuímos o nível de desempenho *Insuficiente*. Esta situação pode ser observada no Gráfico 13.

Gráfico 13: H3-Desempenho geral na Avaliação Diagnóstica.



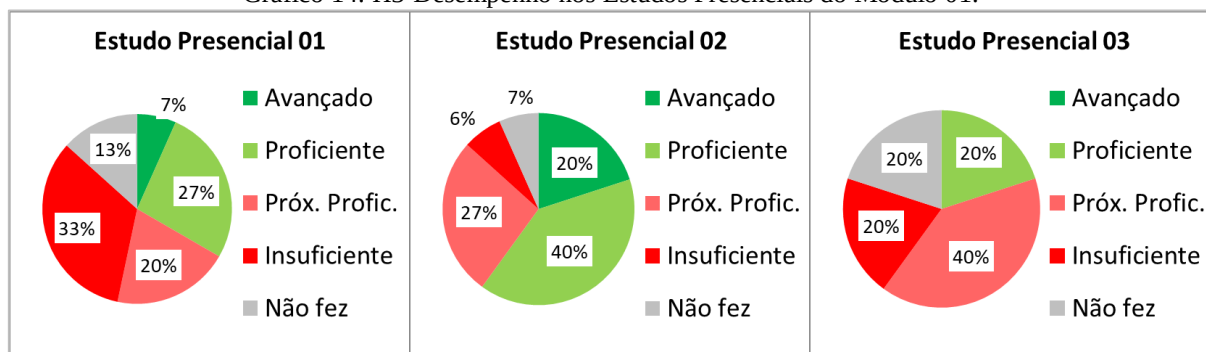
Fonte: Autoria própria.

Durante os Estudos Presenciais constatamos avanços na assimilação necessária para a associação do contexto de problema e também nas descrições, conforme apresentamos no Gráfico 14.

Apesar de constataremos evolução nas respostas às unidades de atividades da

Habilidade 03, os percentuais atingiram patamares menores em comparação com os desempenhos observados nas Habilidades 01 e 02 para o Módulo 01. No Estudo Presencial 01 somente 34% dos estudantes atingiram os níveis satisfatórios (*Avançado* ou *Proficiente*). No Estudo Presencial 02 foram 60% de classificações satisfatórias e no Estudo presencial 03 houve uma queda para apenas 20%.

Gráfico 14: H3-Desempenho nos Estudos Presenciais do Módulo 01.



Fonte: Autoria própria.

Para uma melhor análise, tomamos como exemplo o Estudo Presencial 03 onde percebemos maiores dificuldades dos estudantes nas três habilidades, sendo alcançados os níveis satisfatórios da seguinte forma: Habilidade 01- 44%; Habilidade 02 - 40%; Habilidade 03 - 20%. Com estes percentuais podemos concluir que, pelo menos 40% dos estudantes obtiveram êxito em identificar o problema e desenvolver uma solução automatizada na linguagem de programação Python, entretanto, metade destes tiveram dificuldades em associar corretamente as soluções com outros problemas de Redes de Computadores. Para fazer uma ilustração melhor das dificuldades dos estudantes é necessário trazer o contexto do problema apresentado para análise das Habilidades 01 e 02 (Quadro 16). Para evitar confusões, chamaremos este problema de *Problema Inicial*.

Quadro 16: H1 e H2-Estudo Presencial 03-Contexto do *Problema Inicial*.

Contexto do problema (Âncora):

No Servidor1, como é padrão nas máquinas Ubuntu Linux, há um arquivo que guarda os registros ou *logs* de autenticação. O arquivo fica no seguinte caminho: /var/log/auth.log. Neste servidor há um usuário chamado user2, que não possui a senha de root. Você desconfia que usuário user2 efetuou tentativas para descobrir a senha de root. Há também a desconfiança de que o usuário apagou seus comandos do *history*. É necessário averiguar no arquivo auth.log se há registros de tentativas de escalada de privilégios do user2. Caso existam tentativas registradas, será necessário apresentá-las aos seus superiores, através de relatório. (Para mudar para user2, no console, digite: su user2. A senha do user2 é user2@. Simule tentativas de acesso ao root com senhas incorretas.)

Fonte: Autoria própria.

Para resolver este problema, primeiramente o estudante teve de concluir que se tratava de necessidade de monitoramento de arquivos de registros de atividades (*logs*) (Habilidade 01). Em seguida teve de automatizar a análise dos arquivos de registros de

atividades referentes às atividades do usuário suspeito, em busca de expressões que indicassem tentativas de acesso não permitido (Habilidade 02). Para a associação do *Problema Inicial* com outro possível problema foram colocados os contextos de problemas constantes no Quadro 17.

Quadro 17: H3-Estudo Presencial 03-Problemas a serem analisados.

Problema 2 – Reflexão e observação - Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede. Situação A: Na empresa em que você trabalha há um servidor de banco de dados que está com problemas. Para resolver, foi contratado um especialista que fará acesso remoto via SSH. Assim, será necessário criar um usuário específico para tal acesso. O seu chefe solicitou que tudo o que fosse executado durante a resolução fosse registrado em um arquivo para eventuais consultas caso o problema se repita. Situação B: Na lista de discussão sobre Linux que você participa, foi divulgada uma falha de segurança envolvendo o aplicativo de DNS. Imediatamente você acessou seu servidor para verificar a versão e verificou que o aplicativo está desatualizado. Situação C: Alguns funcionários antigos, do setor de tecnologia da informação, deixaram a empresa antes de você entrar. Fazendo uma varredura nos servidores, você verificou que estes ex-funcionários ainda possuem usuários e permissões de acesso. Verificou também que os usuários têm permissão de acesso remoto via SSH.
--

Fonte: Autoria própria.

Analisando as três *situações*, esperava-se que o estudante associasse as soluções construídas para o *Problema Inicial* com o problema proposto na Situação A, pois exigia um monitoramento e transcrição do arquivo *history*²⁰ para um novo arquivo contendo as ações do usuário criado para a manutenção remota, em moldes semelhantes ao utilizado para resolver o *Problema Inicial*.

Como exemplo de resposta correta e justificada, apresentamos a resposta do estudante Jarbas: “*Como será necessário manipular e criar relatórios de monitoramento das ações dos usuários, a situação A é a mais similar.*” Os contextos dos problemas apresentados nas *situações B e C* não possuem tanta similaridade quanto a *situação A*, pois na *B* o problema se refere à verificação de aplicativos, sem necessidade explícita de manipulação de arquivos. Já a *situação C* indica que ex-funcionários ainda possuem contas ativas para acesso remoto, mas não pressupõe utilização de monitoramento ou manipulação de arquivos na resolução do problema, mas possível exclusão de usuários desligados da empresa.

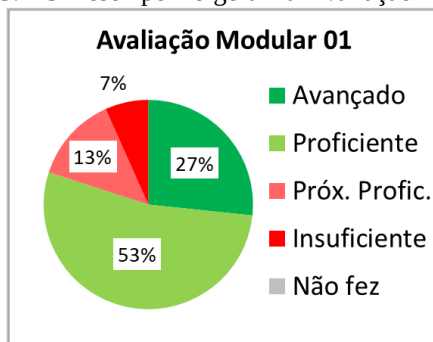
Muitos estudantes apresentaram dificuldades para interpretar os problemas apresentados nas unidades de atividades da Habilidade 03 no Módulo 01, o que fez com que buscássemos conversar mais sobre isso nos momentos de discussão nos Estudos Presenciais e nos *feedbacks*, incentivando-os a dedicarem mais atenção na interpretação dos problemas e refletirem mais sobre os recursos utilizados no Problema Inicial para efeitos de comparação.

Esta ação se refletiu na Avaliação Modular 01, visto que as respostas para a

²⁰ O arquivo *history* é utilizado nos sistemas Unix/Linux para armazenar todos os comandos executados pelo usuário.

Habilidade 03 atingiram 80% de avaliações classificadas como satisfatórias, conforme apresentado no Gráfico 15.

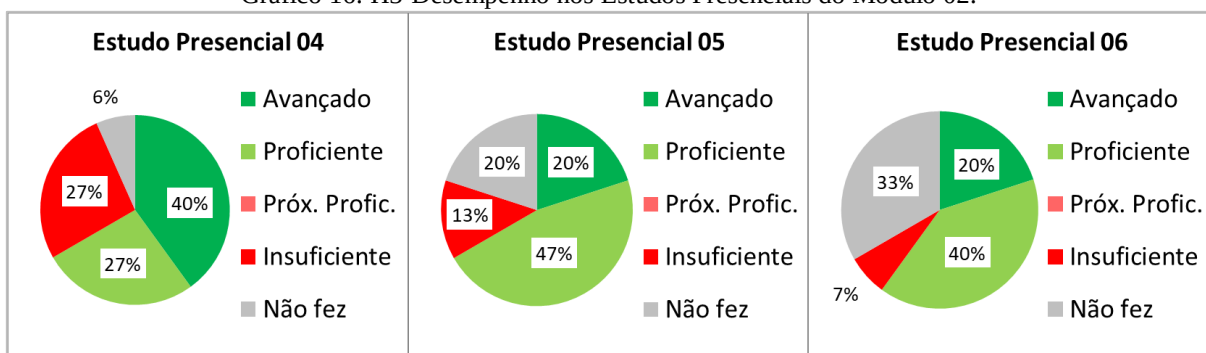
Gráfico 15: H3-Desempenho geral na Avaliação Modular 01.



Fonte: Autoria própria.

Nos Estudos Presenciais do Módulo 02 continuamos cobrando maior atenção e os resultados para a Habilidade 03 se estabilizaram entre 60% e 67% de respostas classificadas como satisfatórias, conforme Gráfico 16.

Gráfico 16: H3-Desempenho nos Estudos Presenciais do Módulo 02.



Fonte: Autoria própria.

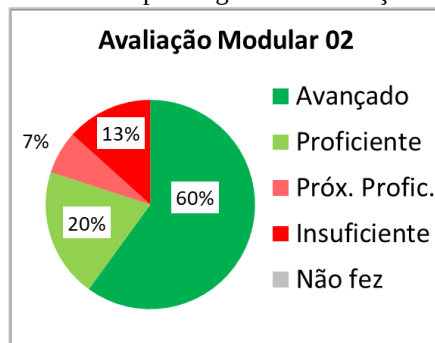
Consideramos que a Habilidade 03 nos representou desafios um pouco maiores, pois os resultados do seu desempenho não ocorriam simplesmente em função de acréscimo de complexidade ou acúmulo de conteúdos, diferente do que ocorreu com as outras duas habilidades. Necessitava, além da prática, de dedicação à interpretação dos problemas e um olhar voltado para as características das soluções dos problemas apresentados. Portanto, nossa intervenção era no sentido de solicitar maior dedicação à atenção para a resolução das unidades de atividades da Habilidade 03, dada sua importância para a vida prática-profissional do estudante. Acreditamos que esta intervenção tenha melhorado a postura dos estudantes referente à Habilidade 03, pois todos os estudantes que realizaram o Estudo Presencial 06 associaram com exatidão os problemas apresentados.

O que fez com que algumas avaliações ficassem aquém do satisfatório foi a forma

com que justificaram a associação na primeira unidade de atividade ou pela ausência ou pouco detalhamento na segunda unidade de atividade.

Na Avaliação Modular 02 os resultados classificados como satisfatórios atingiram os 80%, conforme Gráfico 17.

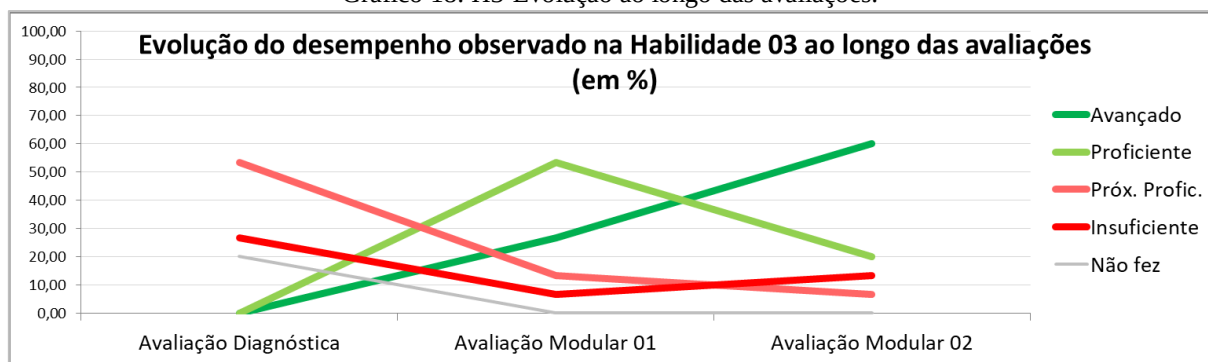
Gráfico 17: H3-Desempenho geral na Avaliação Modular 02.



Fonte: Autoria própria.

Nesta Avaliação Modular 02 apenas 13% dos estudantes associaram os problemas de forma equivocada e apenas 7% deixaram de detalhar a solução. Isto representa um avanço significativo, principalmente quando comparamos com os resultados obtidos na Avaliação Diagnóstica. Naquela avaliação, apesar de 53% dos estudantes terem conseguido acertar qual o problema que mais se associava, nenhum soube justificar ou detalhar como o problema seria resolvido. Esta comparação pode ser observada no Gráfico 18, bem como a evolução dos estudantes ao longo das três avaliações da Habilidade 03.

Gráfico 18: H3-Evolução ao longo das avaliações.



Fonte: Autoria própria.

Ao analisarmos as três Habilidades, podemos afirmar que houve evolução significativa desde o início da aplicação da nossa proposta de ensino-aprendizagem. Entretanto, identificamos algumas limitações, apresentadas na próxima seção.

4.4 LIMITAÇÕES VERIFICADAS NA APLICAÇÃO DA PROPOSTA

Para responder à questão de pesquisa “*Quais limitações são verificadas na implementação da proposta de ensino-aprendizagem?*”, levamos em consideração: a abrangência das disciplinas de Redes de Computadores, de Programação e do sistema operacional Linux; o dimensionamento da previsão de tempo destinado à realização dos Estudos Autônomos; as dificuldades dos estudantes para reprodução do Laboratório Virtual; e, a quantidade de estudantes concluintes nesta implementação.

Ainda que tenhamos dedicado um tempo considerável para concepção e organização da proposta e, posteriormente, aproximadamente seis meses de planejamento do curso e desenvolvimento e seleção de recursos didáticos, não é possível desenvolver um trabalho que abranja todo o domínio que envolva programação e Redes de Computadores (tampouco é nossa pretensão), mesmo que tenhamos delimitado o escopo em administração de Redes de Computadores. Assim, uma das limitações desse trabalho se dá pela própria necessidade de delimitação dos domínios abordados.

Quanto à disciplina de Programação, ainda que todos tivessem cursado ao menos uma disciplina e, presumidamente, possuíssem conhecimentos básicos, a maioria dos estudantes teve problemas quando foram confrontados com problemas que envolviam estruturas de repetição (*for*), de decisão (*if-else*) e operações com *strings*, mesmo os que eram oriundos de cursos de Bacharelado em Sistemas de Informação ou CST em Análise e Desenvolvimento de Sistemas. Observamos que não era apenas um problema com a linguagem Python, mas com lógica de programação. Tentamos contornar o problema com *feedbacks* e indicação de vídeos de terceiros sobre Python Básico e vídeos autorais sobre a aplicação da linguagem em problemas reais.

Quanto à disciplina de Redes de Computadores percebemos as seguintes dificuldades dos estudantes: desconhecimento sobre o funcionamento básico de protocolos como ICMP e SNMP; ausência de noções de *backup* em rede; desconhecimento sobre consultas básicas de endereços IP e de outros dados básicos. Diante disso, parte das Tutorias Presenciais Facultativas e dos Estudos Presenciais teve de ser dedicada à familiarização com estes temas. Era previsto que os estudantes teriam dificuldades com determinados conceitos, pois não era pré-requisito ter conhecimentos em Redes de Computadores, entretanto, apenas um dos estudantes afirmou nunca ter cursado disciplina de Redes de Computadores, o que nos induziu a prever que o restante da turma dominaria ao menos os conceitos básicos. Assim, com o objetivo de mitigar o problema, tivemos que dedicar um tempo extra durante os Estudos

Presenciais e Tutorias Presenciais Facultativas, além de indicação de materiais complementares para a execução dos Estudos Autônomos.

Sobre o sistema operacional Linux, entendemos que se tratava de falta de costume com o sistema operacional em si e com algumas tarefas que envolviam o console de linha de comandos, que era desconhecido para muitos. Na tentativa de mitigar esta limitação, fornecemos uma apostila básica e uma explanação básica sobre estruturas e principais comandos em uma Tutoria Presencial Facultativa. Outras dúvidas eram tiradas durante os Estudos e *feedbacks*.

Sobre o tempo que seria destinado aos Estudos Autônomos, havíamos previsto que seriam suficientes duas horas semanais, entretanto, de acordo com o *feedback* dos estudantes, foram necessárias aproximadamente três horas em média por semana para resolver as questões nos respectivos Roteiros de Aprendizagem. Relacionado a isto também houve uma dificuldade, ou mesmo, resistência dos estudantes em estudarem autonomamente, pois a maioria informou não ter costume com o modelo.

A reprodução do Laboratório Virtual tomou um tempo maior do que o previsto, o que prejudicou a realização mais fluida do Estudo Autônomo 01 por alguns estudantes que tiveram problemas com a versão da máquina virtual disponibilizada, causando atrasos. Estes atrasos tiveram de ser administrados e os estudantes auxiliados presencial e remotamente para compensação do tempo perdido. Na segunda semana o atraso já havia sido superado pelos envolvidos e as atividades puderam transcorrer normalmente.

A quantidade de estudantes que concluíram o curso também é um fator limitante. Entendemos que alguns fatores podem ter contribuído para a evasão, como: o formato de extensão, em que alguns estudantes não se viam comprometidos com o curso, colocando-o em segundo plano, frente aos seus cursos superiores regulares; o período do ano em que o curso foi aplicado, que envolveu concorrência com as avaliações finais dos cursos regulares e, em seguida o período de férias dos estudantes.

Apesar destes desvios, consideramos os resultados positivos. Esta e outras percepções são apresentadas no próximo capítulo, onde constam nossas considerações finais.

CONSIDERAÇÕES FINAIS

Nesta pesquisa objetivamos desenvolver uma proposta de ensino-aprendizagem de Programação pra administração de Redes de Computadores que pudesse servir como alternativa para que professores que ministram disciplinas de Redes de Computadores possam implementar atividades práticas com emprego de recursos de Programação para resolução de problemas reais do cotidiano de um profissional de Redes de Computadores.

Esta proposta de ensino-aprendizagem norteia-se pelo alinhamento construtivo e é baseada no modelo de ensino híbrido de sala de aula invertida, combinando estudos autônomos e presenciais conduzidos por Roteiros de Aprendizagem, suportados por um conjunto de problemas reais, *links* de páginas Internet e, principalmente videoaulas, para resolução de problemas reais do domínio de Redes de Computadores com programação na linguagem Python.

A nossa proposta de ensino-aprendizagem foi aplicada no formato de curso de extensão no Instituto Federal de Rondônia com 15 (quinze) estudantes de cursos superiores da área de computação. Sobre esta aplicação realizamos um estudo de caso para respondermos nossas questões de pesquisa e avaliarmos como a proposta poderia contribuir para melhoria de três habilidades específicas: *identificar* problemas de Redes de Computadores, *automatizar* soluções para estes problemas e *associar* as soluções encontradas com outros problemas de Redes de Computadores.

Os resultados obtidos foram positivos, pois, considerando a análise das respostas nos instrumentos de avaliação, constatamos evolução nas três habilidades analisadas: para *identificar* problemas, o aproveitamento satisfatório final foi de 94%; para *automatizar* soluções, o aproveitamento satisfatório final foi de 100%; e, para *associar* as soluções, o aproveitamento satisfatório final foi de 80%. Estes percentis indicam que os estudantes tiveram maior facilidade para construir programas do que para identificar com precisão os problemas de Redes de Computadores. Também indicam que a associação das soluções encontradas é uma tarefa mais complexa do que identificar o problema e automatizar sua solução. Também foi positiva a adaptação dos estudantes ao modelo de sala de aula invertida, onde pudemos constatar um aumento de 50% no tempo de estudos fora de sala de aula, pelos estudantes.

A partir deste trabalho elaboramos um produto educacional²¹ no formato de um guia

²¹ Disponível em: <http://mpet.ifam.edu.br/dissertacoes-defendidas/>

didático com instruções para que outros professores possam reproduzir esta experiência, podendo adaptá-la com uma disciplina regular, disciplina optativa, parte de uma disciplina de Redes de Computadores, curso de extensão ou curso livre. Neste guia disponibilizamos todo o material didático que utilizamos: Roteiros de Aprendizagem; conjunto de problemas; videoaulas; *links* para ampliação dos conhecimentos sobre os conteúdos e métodos abordados; instruções sobre a implementação do Laboratório Virtual.

A partir deste trabalho também realizamos uma publicação de um artigo científico intitulado “Sala de Aula Invertida - Uma Experiência no Ensino-Aprendizagem de Programação para Administração de Redes de Computadores” na Revista Renote – Novas Tecnologias na Educação (RENOTE, 2018), Volume 16, nº 1, julho/2018 (OLIVEIRA; MENDONÇA, 2018a). Também publicamos outro artigo intitulado “Programação para Administração de Redes de Computadores – Uma Experiência com Estudantes de Computação”, elaborado para participação no 26º WEI – Workshop sobre Educação em Computação, do XXXVIII Congresso da Sociedade Brasileira de Computação realizado em Natal-RN (OLIVEIRA; MENDONÇA, 2018b) (APÊNDICE G).

Este trabalho também teve um impacto na atuação profissional deste mestrando, enquanto professor do Instituto Federal de Rondônia, pois nas disciplinas lecionadas buscamos implantar metodologias ativas, com inclusão de vídeos para auxiliarem as tarefas dos Roteiros de Aprendizagem como instrumentos de ensino-aprendizagem. Em novos cursos a serem implantados no IFRO pretendemos sugerir a adoção das estratégias didáticas utilizadas neste curso, como o ensino híbrido e maior adoção de atividades práticas.

Como trabalhos futuros, pretendemos reaplicar esta experiência, adaptada como parte de uma disciplina intitulada “Programação Aplicada em Redes de Computadores”, no quinto período do CST em Redes de Computadores e, conforme a experiência, pretendemos propor uma mudança definitiva para a disciplina. Pretendemos também aplicar parte da metodologia em uma disciplina de Lógica de Programação para o primeiro período do mesmo curso, de forma que os problemas elaborados reflitam situações práticas de Redes de Computadores.

REFERÊNCIAS

- BENDER, W. N. **Aprendizagem baseada em projetos** – Educação diferenciada para o Século XXI. Porto Alegre: Penso, 2014.
- BERGMANN, J.; SAMS, A. **Flip YOUR Classroom** – Reach Every Student in Every Class Every Day. Alexandria: ISTE, 2012.
- BIANCONI, B. Errar, descobrir e reinventar: faça você mesmo. **Porvir**. 2013. Disponível em: <<http://porvir.org/na-cena-maker-prototipos-convergencia-para-aprendizado-critico/>>. Acesso em: 09 mar. 2017.
- BIGGS, J.; TANG, C. **Teaching for Quality Learning at University**. 4. ed. Berkshire, England: Society for Research into Higher Education & Open University Press, 2011.
- BLANK-EDELMAN, D. N. **Automating System Administration with Perl**. 2. ed. Sebastopol: O'Reilly, 2009.
- BLIKSTEIN, P. Palestra sobre aprendizagem mão na massa. **Transformar 2015**. São Paulo: 2015. Disponível em: <<http://transformareducacao.org.br/video/aprendizagem-mao-na-massa-paulo-blikstein/>>. Acesso em: 09 mar. 2017.
- BLIKSTEIN, P. **Como a educação mão na massa chega até a sala de aula**. Entrevista concedida ao site Porvir. Disponível em: <<https://soundcloud.com/porvir/como-a-educacao-mao-na-massa-chega-ate-a-sala-de-aula>>. Acesso em: 09 mar. 2017.
- BLUM, R.; BRESNAHAN, C. **Linux Command Line and Shell Scripting Bible**. 3.ed. Indianápolis: Wiley, 2015.
- BORGE, S. **Automating Windows Administration**. Nova Iorque: Apress, 2004.
- BRABRAND, C.; DAHL, B. Constructive alignment and the solo taxonomy: a comparative study of university competences in computer science vs. mathematics. In: AUSTRALIAN COMPUTER SOCIETY, INC. **Proceedings** of the Seventh Baltic Sea Conference on Computing Education Research –Volume 88, p. 3-17. Darlinghurst, Austrália, 2007.
- BRASIL. Ministério da Educação. **Catálogo Nacional de Cursos Superiores de Tecnologia**. 3 ed. Brasília: jun. 2016. Disponível em: <http://portal.mec.gov.br/index.php?option=com_docman&view=download&alias=44501-cncst-2016-3edc-pdf&category_slug=junho-2016-pdf&Itemid=30192>. Acesso em: 01 set. 2016.
- BRASIL. Ministério da Educação. **Decreto n.º 5.773, de 9 de maio de 2006**. Dispõe sobre o exercício das funções de regulação, supervisão e avaliação de instituições de educação superior e cursos superiores de graduação e sequenciais no sistema federal de ensino. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2006/decreto/d5773.htm>. Acesso em: 01 set. 2016.
- BRASIL. Ministério da Educação. **Parecer Conselho Nacional de Educação – Conselho Pleno nº 29/2002**. Institui as Diretrizes Curriculares Nacionais Gerais para a organização e o

funcionamento dos cursos superiores de tecnologia. Disponível em: <<http://portal.mec.gov.br/cne/arquivos/pdf/cp29.pdf>>. Acesso em: 23 ago. 2016.

BRIGGS, J. R. **Python for kids - A Playful Introduction to Programming**. San Francisco: No Starch Press, 2013.

BROOKHART, S. M. **How to create and use rubrics for formative assessment and grading**. Alexandria: ASCD, 2013.

CAMPI, N.; BAUER, K. **Automating Linux and Unix System Administration**. 2. ed. Nova Iorque: Apress, 2009.

CHEN H. Discussion on the Teaching Methods of Applied Undergraduate Computer Network Courses. 3rd International Conference on Science and Social Research (ICSSR 2014), 2014, Paris. **Proceedings...** Paris: Atlantis Press, 2014a, p. 127-130.

CHEN, E. Z. Flipped Classroom Model and Its Implementation in a Computer Programming Course. **Lärarlärdom**, vol. 2014, p. 180-200, 2014b. Disponível em: <<http://www.diva-portal.org/smash/get/diva2:777869/FULLTEXT01.pdf>>. Acesso em: 28 abr. 2018.

CHRISTENSEN, C.; HORN, M.; STAKER, H. Ensino Híbrido: uma Inovação Disruptiva?. Uma introdução à teoria dos híbridos. Clayton Christensen Institute for disruptive innovation, 2013. Disponível em: <https://s3.amazonaws.com/porvir/wp-content/uploads/2014/08/PT_Is-K-12-blended-learning-disruptive-Final.pdf>. Acesso em: 22 fev. 2017.

CHUN, W. J. **Core Python Applications Programming**. 3.ed. Nova Jersey: Prentice Hall, 2012.

COSTA, D. G. **Administração de redes com scripts** – Bash script, python e VBScript. Rio de Janeiro: Brasport, 2007.

CRUZ, F.. **Python: Escreva seus primeiros programas**. São Paulo: Casa do Código, 2015.

DOUGHERTY, D. The Maker Movement. **Innovations – Technology, Governance & Globalization**. Cambridge, MIT Press Journals, v. 7, n. 3, p. 11-14, verão 2012. Disponível em: <http://www.mitpressjournals.org/doi/abs/10.1162/INOVA_00135#.WMGFETsrK02>. Acesso em: 09 mar. 2017.

EDELMAN, J.; LOWE, S. S.; OSWALT, M. **Network Programmability Network Automation. Skills for the Next-Generation Network Engineer**. Early release. Sebastopol: O'Reilly, 2015.

E-MEC. **Cadastro e-MEC de Instituições e Cursos de Educação Superior** - Ministério da Educação. Disponível em: <<http://emec.mec.gov.br/>>. Acesso em: 9 mar. 2017.

FRISCH, A. **Essential System Administration**. Tools and techniques for Linux and Unix. 3 ed. Sebastopol: O'Reilly, 2002.

GENG, H. **Data Center Handbook**. Hoboken: Wiley, 2015.

GIFT, N.; JONES, J. M. **Python for Unix and Linux System Administration**. Sebastopol: O'Reilly, 2008.

HATCH, M. **The Maker Movement Manifesto**. Nova Yorque: McGraw-Hill, 2014.

JOHNSON, C. F. A.; VARMA, J. **Shell Scripting Recipes**. 2. ed. . Nova Iorque: Apress, 2015.

KEMP, J. **Linux System Administration Recipes**. Nova Iorque: Apress, 2009.

KUROSE, J. *et al.* Workshop on computer networking: curriculum designs and educational challenges. **ACM SIGCOMM Computer Communications Review**, v. 32, n. 5, p. 1–9, Nov. 2002.

LAMBERT, K. A. **Python: first programs**. Boston: Cengage Learning, 2012.

LEITE, F. N.; HOJI, E. S.; ABDALA JUNIOR, H., A Blended Learning Method Applied in Data Communication and Computer Networks Subject. **IEEE Latin America Transactions**, v. 16, n. 1, jan. 2018. p. 163-171. Disponível em: <http://www.ewh.ieee.org/reg/9/etrans/ieee/issues/vol16/vol16issue01Jan.2018/16TLA1_25NLeite.pdf>. Acesso em: 28 abr. 2018.

MENDONÇA, A. P. Alinhamento Construtivo: fundamentos e aplicações. In: GONZAGA, A. P. **Formação de professores no Ensino Tecnológico**: fundamentos e desafios. Curitiba: CRV, 2015. p. 109-130.

MENDONÇA, A. P. **Programação Orientada ao Problema**: Uma Metodologia para Entendimento de Problemas e Especificação no Contexto de Ensino de Programação para Iniciantes. 2010. Tese (Doutorado em Ciências da Computação) - Universidade Federal de Campina Grande, Campina Grande, 2010.

MENDONÇA, A. P.; COELHO, I. M. W. da S. Rubricas e suas Contribuições para a Avaliação de Desempenho de Estudantes. In: Souza, Ana Cláudia R. de; Coelho, Iandra M. W. da S.; Neto, João dos S. C.; Azevedo, Rosa O. M (Org.). **Formação de Professores e Estratégias de Ensino**: perspectivas teórico-práticas. 1a. ed. Curitiba: Appris, 2018.

MICROSOFT. **Automating Windows Server 2008 Administration with Windows PowerShell**. Microsoft: 2008.

NEMETH, E. et al. **Unix and Linux System Administration Handbook**. 4. ed. Nova Jersey : Prentice Hall, 2011.

NETCRAFT. **September 2017 Web Server Survey** - Web-facing operating systems. Disponível em: <<https://news.netcraft.com/archives/2017/09/11/september-2017-web-server-survey.html>>. Acesso em: 28 abr. 2018.

OLIVEIRA, S. A. B. de; MENDONÇA, A. P. Sala de Aula Invertida – Uma Experiência no Ensino-Aprendizagem de Programação para Administração de Redes de Computadores. **RENOTE – Novas Tecnologias na Educação**. v.16, n. 1, jul. 2018a. Disponível em <<https://seer.ufrgs.br/renote/article/view/86020>>. Acesso em 30. nov. 2018.

OLIVEIRA, S. A. B. de; MENDONÇA, A. P. Programação para Administração de Redes de Computadores – Uma experiência com estudantes de computação. XXXVIII Congresso da Sociedade Brasileira de Computação, 26º WEI – Workshop sobre Educação em Computação, 2018, Natal. **Anais...** Natal, 2018b, p. 246-255. Disponível em:

<<http://natal.uern.br/eventos/csbc2018/anais/Anais%20WEI%202018.pdf>>. Acesso em: 30 nov. 2018.

OPENSOURCE.org. **Open source initiative**. 2007. Disponível em: <<https://opensource.org/definition>>. Acesso em: 28 mar. 2017.

PENIDO, A. Escolas em (re)construção. In: FUTURA. **Destino: Educação – Escolas Inovadoras**. Moderna, 2016. Disponível em: <<http://www.moderna.com.br/lumis/portal/file/fileDownload.jsp?fileId=8A808A82569F88700156D71DB78437F5>>/. Acesso em: 21 mar. 2017.

PORVIR.org. **Mão na massa** – O aprendizado que passa pelas mãos. Disponível em: <<http://porvir.org/especiais/maonamassa/>>. Acesso em: 09 mar. 2017.

RESNICK, M. et al. Scratch: programming for all. **Communications of the ACM**. Vol. 52, n.11. 2009. p. 60-67. Disponível em: <<http://cacm.acm.org/magazines/2009/11/48421-scratch-programming-for-all/fulltext>>. Acesso em: 13 mar. 2017.

RESNICK, M. **Learn to code, code to learn**. Edsurge News, 2013. Disponível em: <<https://www.edsurge.com/news/2013-05-08-learn-to-code-code-to-learn>>. Acesso em: 27 mar. 2017.

RHODES, B.; GOERZEN, J. **Foundations of Python Network Programming**. 3.ed. Nova Iorque: Apress, 2014.

ROSSUM, G. V. **Guia de aprendizagem de Python**. Release 2.0. Santa Clara: BeOpen, 2000.

SARKER, M. O. F.; WASHINGTON, S. **Learning Python Network Programming**. Birmingham: Packt, 2015.

SILEIKA, R. **Pro Python System Administration**. 2. ed. Nova Iorque: Apress, 2014.

TANENBAUM, A. S. **Organização Estruturada de Computadores**. 5. ed. Pearson, 2007.

TAYLOR, D. **Wicked Cool Shell Scripts**. San Francisco: No Starch Press. 2004.

THONNY.COM. **Python IDE for beginners**. Disponível em <<http://thonny.org>>. Acesso em: 28 abr. 2018.

WANG, Y.; BLUM, T.; MCCOEY, M. Teaching a networking class for freshmen: course design and lessons learned. In: Proceedings of the 15th Annual Conference on Information technology education, 2014, Atlanta. **Proceedings...** Atlanta: 2014, p. 9-14.

YIN, R. K. **Estudo de caso: Planejamento e Métodos**. 4. ed. Porto Alegre: Bookman, 2010.

APÊNDICE A – Levantamento sobre cursos do eixo Informação e Comunicação nos Institutos Federais

- **Período do levantamento:** entre março e abril de 2017.
- **Foco:** CST em Tecnologia do eixo Informação e Comunicação dos Institutos Federais.
- **Estratégia de pesquisa:** Consulta no site e-Mec, do Ministério da Educação, por Institutos de Ensino Superior (IES), do Sistema Federal de Ensino, que resultou em uma lista de Institutos Federais (IF) e Centro Federal de Educação Tecnológica (CEFET), sobre a qual realizamos buscas em seus sites por Cursos Superiores em Tecnologias do eixo Informação e Comunicação (Conforme relação abaixo). Nestes sites fizemos buscas por detalhes de disciplinas de Redes de Computadores e de programação.

CST do eixo Informação e Comunicação da Rede Federal de Ensino.

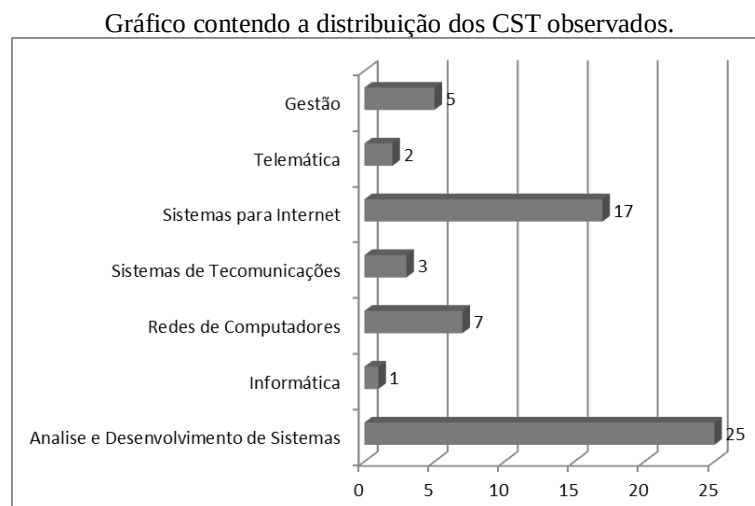
	RELAÇÃO DE INSTITUTOS FEDERAIS, CONFORME E-MEC (HTTP://EMEC.MEC.GOV.BR/):
1	CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA (CEFET/RJ)
2	CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS (CEFET/MG)
3	INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA DE RONDÔNIA (IFRO)
4	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA BAIANO (IFBAIANO)
5	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA CATARINENSE (IF CATARINENSE)
6	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA BAHIA (IFBA)
7	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA (IFPB)
8	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE ALAGOAS (IFAL)
9	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE BRASÍLIA (IFB)
10	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS (IFG)
11	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MATO GROSSO (IFMT)
12	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MATO GROSSO DO SUL (IFMS)
13	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE MINAS GERAIS (IFMG)
14	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE PERNAMBUCO (IFPE)
15	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE RORAIMA (IFRR)
16	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA CATARINA (IFSC)
17	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO (IFSP)
18	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SERGIPE (IFS)
19	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO ACRE (IFAC)
20	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO AMAPÁ (IFAP)
21	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO AMAZONAS (IFAM)
22	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO CEARÁ (IFCE)
23	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO ESPÍRITO SANTO (IFES)
24	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO MARANHÃO (IFMA)
25	INSTITUTO FED. DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO NORTE DE MINAS GERAIS (IFNMG)
26	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PARÁ (IFPA)
27	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PARANÁ (IFPR)
28	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ (IFPI)
29	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO DE JANEIRO (IFRJ)
30	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE DO NORTE (IFRN)
31	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE DO SUL (IFRS)
32	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO SERTÃO PERNAMBUCANO (IF SERTÃO)
33	INSTITUTO FED. DE EDUC., CIÊNCIA E TECNOLOGIA DO SUDESTE DE MINAS GERAIS (IFSEMG)
34	INSTITUTO FED. DE EDUC., CIÊNCIA E TECNOLOGIA DO SUL DE MINAS GERAIS (IF SUL DE MINAS)
35	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO TOCANTINS (IFTO)
36	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO TRIÂNGULO MINEIRO (IFTM)
37	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA FARROUPILHA (IFFARROUPILHA)
38	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA FLUMINENSE (IF FLUMINENSE)
39	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO (IF GOIANO)
40	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SUL-RIO-GRANDENSE (IFSUL)

Fonte: Nossa, a partir de consulta ao site do e-Mec (E-MEC, 2017).

- Total de IES observadas: 40

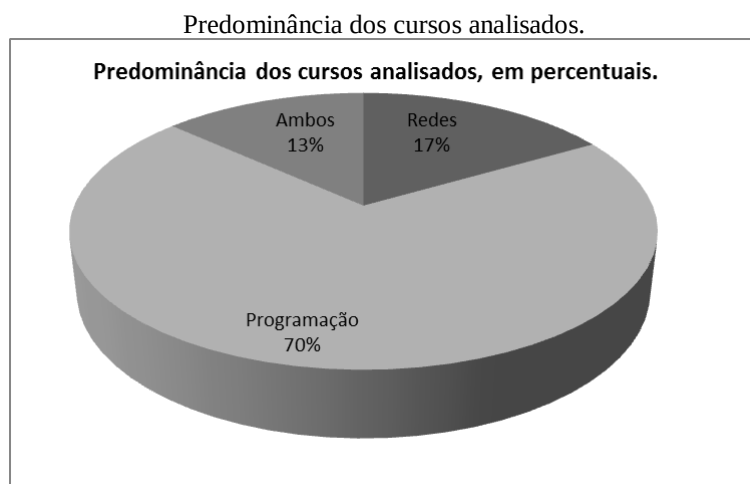
- Total de CST observados: 60
- Total de IES observados que não possuem CST do eixo Informação e Comunicação: 6 ²²

A distribuição entre os cursos pode ser observada conforme o gráfico a seguir.



Fonte: Autoria própria.

Cada curso tem um foco específico e esta distribuição pode ser observada no gráfico abaixo.



Fonte: Nossa

As principais referências bibliográficas utilizadas em disciplinas iniciais de Redes de Computadores tinham a presença de várias obras, entretanto, notamos a presença maior dos seguintes autores: Andrew S. Tanenbaum, Douglas E. Comer, e James F. Kurose; Keith W. Ross, distribuídas da seguinte forma:

²² O IFBA estava com os *links* indisponíveis. Os outros não possuem cursos no eixo Informação e Comunicação.

- Quantidade de CST com informações sobre referências bibliográficas: 34/60
- Tanenbaum: 33/60
- Kurose: 31/60
- Comer; Ross: 24/60
- As três na mesma ementa: 22/60

Observamos também as ocorrências das disciplinas iniciais de Redes de Computadores e de programação por períodos (semestres).

- Para os cursos em que Redes de Computadores são o foco, as disciplinas de programação costumam ser lecionadas, em sua maioria, a partir do primeiro período. Dos 14 cursos, 11 traziam informações sobre o período em que as disciplinas são ministradas e todos oferecem disciplinas de programação a partir do primeiro período.
- Para os cursos em que programação é o foco, as disciplinas iniciais de Redes de Computadores costumam ser lecionadas com uma variação maior, entre o primeiro e o quarto período, havendo uma predominância média a partir do terceiro período.

APÊNDICE B – Anúncios de emprego para Analistas de Suporte Técnico

Este levantamento foi feito no site de anúncios de vagas de emprego www.catho.com.br no dia 17/09/2018, compreendendo anúncios entre 11/09/2018 a 17/09/2018.

Foram encontradas 48 ofertas de emprego com a chave de busca “Analista de Suporte Técnico”. Destas, 35,4% (17 ofertas) exigiam conhecimentos que pressupõem a necessidade de emprego de programação para administração de redes de computadores. Nas figuras abaixo são apresentados alguns exemplos encontrados nesta busca.

Anúncio de vaga para Analista de Suporte Técnico (I).

Analista de Suporte Técnico

hoje

A Combinar

1 vaga: São Paulo - SP (1)

Suporte ao cliente, gerenciamento de rede, suporte a clientes, monitoramento de servidores, desenvolvimento de novas ferramentas para melhoria do serviço, instalação, manutenção e configuração de servidores. Conhecimentos em Windows Server e Workstation, linguagem SQL, análise de Logs. Conhecimentos em telefonia e em redes IP, conexão VPN, telefonia Voip (plataformas SIP, asterisk, ATA'S).

[enviar currículo](#) | 30 dias grátis

Autoria própria a partir de busca no site Catho – Grifamos.

Anúncio de vaga para Analista de Suporte Técnico (II).

Analista de Suporte Técnico

terça, 11/09

De R\$ 2.001,00 a R\$ 3.000,00

1 vaga: Curitiba - PR (1)

Configuração e manutenção da segurança de rede. Instalação, manutenção e ampliação da rede local e externa. Acompanhar processo de compra de equipamentos para manutenção da rede local e externa. Instalar e configurar novos equipamentos na rede. Realizar verificação diária nos servidores: controle de backups, estado de discos rígido, parametrização de rotinas automática, atualização de versão e segurança. Monitorar a performance da rede e seus equipamentos. Garantir a integridade e confidencialidade das informações, registrando ocorrências de infrações. Configurar e manter a comunicação digital (correio eletrônico, Webmail, Outlook). Definir controles de acesso de banda LAN e WAN a cada grupo de usuários. Instalar e configurar computadores, isolados ou em redes, periféricos e softwares pré-definidos. Configurar os serviços e/ou funções de sistemas operacionais. Realizar backup e restauração de computadores. Selecionar softwares a partir das necessidades do usuário. Realizar manutenção de hardware dos desktop e notebooks do grupo. Testar equipamentos periféricos (placas de rede, portas de comunicação, monitores, teclados, modems, cabos placas de vídeo). Elaborar manuais e melhorar procedimentos de TI. Conhecimento em montagem e manutenção de computadores, instalação e preparação em sistema operacional. Conhecimentos em Switch Cisco, Active Directory, VMWare, Storage EMC e ArcServer. Ensino Superior em Tecnologia de Redes.

[enviar currículo](#) | 30 dias grátis

Autoria própria a partir de busca no site Catho – Grifamos.

Anúncio de vaga para Analista de Suporte Técnico (III).

Analista de Suporte Técnico I

R\$ 1.700,00

1 vaga: Porto Alegre - RS (1)

sexta, 14/09

Prestar suporte ao cliente instalando e realizando manutenção de software e hardware de propriedade da empresa em topologias diversificadas. **Controlar e monitorar ativamente ambiente operacional da rede de computadores.** Executar implantação física e lógica de projetos de rede de computadores. Prestar assistência técnica na instalação e utilização em diferentes topologias há equipamentos da empresa.

Cursando Técnico ou Superior na área de TI. **Conhecimento de sistemas operacionais abertos; ferramentas de monitoramento; ambientes virtualizados;** protocolo de roteamento estático e dinâmicos; Conhecimento avançado de plano de redes de computadores e tipologia TCP/IP; Infraestrutura de redes físicas e lógicas (camada 1, 2, 3).

[enviar currículo](#) | 30 dias grátis



Autoria própria a partir de busca no site Catho – Grifamos.

APÊNDICE C – Roteiros de Aprendizagem

Nas páginas que se seguem (da p. 106 à p. 144), são apresentados os 12 (doze) RA, agrupados por 6 (seis) temas, cada um composto por um RA para Estudo Autônomo e um RA para Estudo Presencial.

Curso de Programação para Administração de Redes de Computadores

Módulo 01

Roteiro de aprendizagem para Estudo Autônomo 01

1.1 Identificando a movimentação da vizinhança



Esforço previsto estudando autonomamente: 2 horas



Descrição Geral

Caro estudante,

Este Roteiro de Aprendizagem foi preparado para seu Estudo Autônomo e fornecerá a base teórica para a resolução dos problemas que serão apresentados em nossa aula presencial.

O monitoramento de *hosts*²³ é muito importante na administração de redes e pode prevenir, minimizar ou evitar problemas que possam indisponibilizar recursos.

Neste Roteiro veremos algumas formas de monitorar a comunicação entre hosts de uma rede de computadores que precisa ser administrada. Para tratarmos deste assunto, inicialmente, é necessário conhecer (ou relembrar) os protocolos ICMP (Internet Control Message Protocol) e SNMP (Simple Network Management Protocol). Para tanto, assista aos vídeos, trabalhe nos laboratórios virtuais e acesse os *links* disponibilizados ao longo das tarefas.

Os conteúdos são disponibilizados em uma ordem que facilita um melhor aproveitamento e sequência de raciocínio. Portanto, siga a sequência e não pule etapas. Caso necessite, complemente seus estudos através de suas próprias pesquisas em outras fontes (Internet, livros, etc.).

Ao longo dos nossos estudos, talvez você se depare com certos termos empregados em programação ou em redes de computadores que possivelmente gerem alguma indagação. Procuramos contemplá-los nos vídeos ou em notas em nossas tarefas. Se ainda assim permanecerem dúvidas, faça suas pesquisas, suas anotações e traga para discutirmos no Estudo Presencial.

Compartilhe suas pesquisas com seus colegas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas relacionados a monitoramento de comunicação entre computadores;
- Automatizar o monitoramento de comunicação entre computadores;
- Identificar a importância e aplicação dos protocolos ICMP e SNMP;
- Identificar e aplicar bibliotecas específicas da linguagem Python para coleta de resultados.

²³ Um *host* é qualquer máquina com recursos computacionais, que possua um número IP e que faça parte de uma rede de computadores.

Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

1. Monitoramento de disponibilidade de equipamentos em redes
 - 1.1. Redes de Computadores
 - 1.1.1. Protocolos TCP/IP
 - 1.1.2. Protocolos ICMP e SNMP
 - 1.1.3. Gerenciamento de Redes
 - 1.1.4. Monitoramento de *hosts*
 - 1.1.5. Métricas de comunicação
 - 1.2. Programação
 - 1.2.1. Variáveis
 - 1.2.2. Comentários como forma de documentação
 - 1.2.3. Importação de bibliotecas
 - 1.2.4. Utilização de métodos de bibliotecas
 - 1.2.5. Entrada e saída de dados
 - 1.2.6. Coletas de estados de execuções de comandos
 - 1.2.7. Verificação de tipos de dados
 - 1.2.8. Conversão de tipos de dados
 - 1.3. Sistemas Operacionais (Linux)
 - 1.3.1. Estruturas de arquivos e diretórios
 - 1.3.2. Comandos de listagem de arquivos e diretórios
 - 1.3.3. Monitoramento de carga e de estados do processador

Tarefas

Abaixo são propostas questões que, para serem resolvidas, necessitam de conhecimentos teóricos, cujos conteúdos seguem disponibilizados nos *links* que acompanham cada questão.

É importante que as respostas ocorram na sequência em que são dispostas as questões.

Questão 01

Para resolver os itens abaixo, assista ao vídeo “Introdução ao Gerenciamento de Redes – parte 6 – ICMP” (elaborado pelo Nic.br). Este vídeo pode ser acessado através do *link*: <https://www.youtube.com/watch?v=q9O59TI9-pA>.

Observe as afirmações feitas nos itens abaixo: 1.1, 1.2 e 1.3. Caso identifique afirmações equivocadas, corrija-as:

1.1 Você disparou um comando PING (Packet Internet Groper) através da sua máquina para o endereço www.uol.com.br e, através de um programa que monitora e analisa pacotes na rede²⁴, observou que sua máquina enviou uma mensagem *echo request*, recebendo um *echo reply*. Isto

²⁴ São programas que capturam tráfego de rede para análise, visando prevenir ou descobrir anomalias na rede. Existem vários exemplos destes programas na Internet. Por exemplo, o Wireshark e o TCPDump.

significa que o PING entrou em *loop* infinito, resultando em um erro.

1.2 A latência é medida através do cálculo da média obtida no campo “time” e é um resultado sempre confiável, devido à prioridade do protocolo ICMP sobre os outros protocolos em suas passagens pelos roteadores.

1.3 Sempre que, após o disparo de um comando PING, um *echo request* obtiver como resposta um *request timeout*, significa que o host destino está sem conectividade.

Questão 02

Para resolver os itens abaixo, assista ao vídeo “Comando PING”, Este vídeo pode ser acessado através do *link*: <https://www.youtube.com/watch?v=LsMFYX9BEC8>.

Responda os itens abaixo:

Exemplo de PING 1

```
root@ubuntu:~# ping -c 3 www.registro.br
PING registro.br (200.160.2.3) 56(84) bytes of data.
64 bytes from registro.br (200.160.2.3): icmp_seq=1 ttl=128 time=102 ms
64 bytes from registro.br (200.160.2.3): icmp_seq=2 ttl=128 time=111 ms
64 bytes from registro.br (200.160.2.3): icmp_seq=3 ttl=128 time=109 ms

--- registro.br ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 102.950/107.950/111.755/3.702 ms
```

Fonte: Autoria própria

2.1 Observando a Figura 1, por que o comando resultou em apenas 3 linhas?

2.2 Ainda quanto à Figura 1, quantos pacotes foram perdidos neste teste? Como isto é identificado?

2.3 A partir de um de nossos servidores, execute os comandos PING a seguir, para os seguintes endereços:

```
ping -c 3 localhost
```

```
ping -c 3 www.registro.br.br
```

A média de latência entre os testes teve grande diferença. Por quê?

Questão 03

Para resolver os itens a seguir, assista ao vídeo “SNMP”. Este vídeo pode ser acessado através do *link*: <https://youtube.com/watch?v=hyRg6DcS9hs>.

Observe a seguinte situação:

Um colega está com dificuldades para gerenciar seus dispositivos via SNMP. Seu cenário é o seguinte: Possui um *datacenter* com 15 máquinas conectadas, entre servidores *Web*, servidores de bancos de dados, *Firewalls*, *proxies*, entre outros.

A fim de utilizar o protocolo SNMP para gerenciar estas máquinas, ele configurou um servidor exclusivo para monitoramento (NMS – *Network Management System* – ou Estação de Gerenciamento), no qual instalou o SNMP no modo *gerente*, com todos seus pacotes e suas MIBS (*Management Information Base*). Nesta mesma máquina também acrescentou as linhas de configuração do *snmpd.conf*, liberando acesso às OIDs (*Object Identifier*). Após configurar o SNMP nesta máquina, ele a plugou na rede. As outras máquinas da rede permaneceram inalteradas.

Entretanto, ao executar os comandos *snmpwalk* ou *snmpget*, desta estação de gerenciamento para as outras máquinas, não obteve sucesso. Ele consegue “pingar” todas as máquinas a partir da máquina de monitoramento. Também verificou no *Firewall* que as portas²⁵ 161 e 162 (utilizadas pelo SNMP) estão liberadas.

Por exemplo, se ele executa:

```
snmpwalk -v2c -c public 192.168.0.110 1.3.6.1.2.1.1.3
```

A resposta é sempre a mesma:

```
Timeout: No Response from 192.168.0.110.
```

Ou seja: excede o tempo esperado para a resposta da solicitação

Após analisado o cenário, responda:

3.1 Qual solução podemos indicar para que nosso colega possa resolver o problema?

Questão 04

Para resolver os itens a seguir, assista aos dois vídeos que tratam da biblioteca do Python chamada subprocess. Estes vídeos podem ser acessados através dos *links* citados a seguir:

Para resolver o item 4.1, assista ao vídeo “BibliotecaSubprocess1”:
https://www.youtube.com/watch?v=73_3zZ7wlhY.

Para o item 4.2, assista ao vídeo “BibliotecaSubprocess2”:
https://www.youtube.com/watch?v=nZpPj_xsEAc.

Também para o item 2.4, utilize as seguintes OIDs:

% de utilização por programas do usuário: 1.3.6.1.4.1.2021.11.9

% de utilização por programas do sistema: 1.3.6.1.4.1.2021.11.10

% de tempo ocioso da CPU: 1.3.6.1.4.1.2021.11.11

Carga da CPU há um minuto: 1.3.6.1.4.1.2021.10.1.3.1

Carga da CPU há cinco minutos: 1.3.6.1.4.1.2021.10.1.3.2

Carga da CPU há quinze minutos: 1.3.6.1.4.1.2021.10.1.3.3

Carga da CPU (Quantidade de processos em fila para processamento – O ideal é que não seja maior que a quantidade de núcleos de um processador ou de um conjunto de processadores. Por exemplo, se o computador é um *quad-core*, o ideal é que não haja mais de 4 processos em espera, ou seja, 1 processo por núcleo. Se este número for alto, significa que há sobrecarga na CPU)

Para outras OIDs, veja o link: <https://www.sysadmin.md/snmp-most-useful-linux-oids.html>

Os exemplos utilizados no vídeo (*testaSubprocess1.py*, *testaSubprocess2.py*, *testaSubprocess3.py*, *testaSubprocess4.py* e *testaSubprocess5.py*) também estão disponibilizados para consulta no Módulo I, Estudo Autônomo 01, da página do nosso curso (Google Sala de Aula).

Construa os seguintes programas: (Lembre-se das boas práticas exemplificadas em nosso primeiro

²⁵ Na camada de transporte do TCP/IP, cada serviço é identificado por um número chamado “porta”. Por exemplo: HTTP utiliza a porta 80; o serviço de transporte de *e-mail*, SMTP, utiliza a porta 25; o SNMP utiliza as portas 161 e 162; etc.

programa: <https://goo.gl/rvnzWy>).

4.1 Conforme o vídeo “BibliotecaSubprocess1”, seguindo o modelo `testaSubprocess2.py`, faça um programa que mostre na tela o resultado da busca pela OID 1.3.6.1.2.1.1.5 na máquina local (*localhost*).

4.2 Conforme o vídeo “BibliotecaSubprocess2”, seguindo o modelo do programa `testaSubprocess5.py` faça um programa que solicita que o usuário digite a OID a ser buscada na máquina local (*localhost*) e que retorne o conteúdo desta OID.

Abraços e bons estudos!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 01

Roteiro de Aprendizagem para Estudo Presencial 01

1.1 Identificando a movimentação da vizinhança



Carga horária prevista em sala de aula: 4 horas



Descrição Geral

Caro estudante,

No dia-a-dia de um profissional de redes, ele se depara com vários problemas técnicos relacionados com o monitoramento de redes e equipamentos e, como vimos no roteiro para Estudo Autônomo, esses problemas podem ser monitorados com o auxílio do ICMP e do SNMP.

Neste roteiro você resolverá alguns problemas que podem surgir na vida profissional e que envolvem o conteúdo já estudado.

É importante seguir a sequência em que são dispostos os problemas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas relacionados a monitoramento de comunicação entre computadores;
- Automatizar o monitoramento de comunicação entre computadores;
- Associar a resolução encontrada com a resolução de outros problemas de redes.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

2. Monitoramento de disponibilidade de equipamentos em redes
 - 2.1. Redes de Computadores
 - 2.1.1. Protocolos TCP/IP
 - 2.1.2. Protocolos ICMP e SNMP
 - 2.1.3. Gerenciamento de Redes
 - 2.1.4. Monitoramento de *hosts*
 - 2.1.5. Métricas de comunicação
 - 2.2. Programação
 - 2.2.1. Variáveis
 - 2.2.2. Comentários como forma de documentação
 - 2.2.3. Importação de bibliotecas
 - 2.2.4. Utilização de métodos de bibliotecas
 - 2.2.5. Entrada e saída de dados

- 2.2.6. Coletas de estados de execuções de comandos
- 2.2.7. Verificação de tipos de dados
- 2.2.8. Conversão de tipos de dados
- 2.3. Sistemas Operacionais (Linux)
 - 2.3.1. Estruturas de arquivos e diretórios
 - 2.3.2. Comandos de listagem de arquivos e diretórios
 - 2.3.3. Monitoramento de carga e de estados do processador



Tarefas

Abaixo, propomos problemas que buscam reproduzir contextos do cotidiano de um profissional que tenha de resolver problemas relacionados a redes de computadores. Cada problema é dividido em três partes: Apresentação do contexto, discussão e resolução. Utilize os conhecimentos acumulados até agora para auxiliar nas resoluções.

É **importante** que as resoluções ocorram na sequência em que os problemas são dispostos.

Problema 1

Obs.: Para resolução deste problema, utilize as seguintes máquinas virtuais: Servidor1, Servidor2.

1.1 Apresentação do contexto (Âncora):

Na empresa em que você trabalha, os usuários estão reclamando que, durante seus expedientes, em alguns momentos, têm dificuldades para acessar a máquina Servidor2, devido à lentidão. Relatam que, geralmente, estas dificuldades ocorrem em momentos de maior pico de trabalho, quando todos os computadores estão em operação.

O Servidor2 funciona como servidor de arquivos e está localizado fisicamente no mesmo prédio, atrás de um *firewall*. A rede não apresenta gargalos em nenhum momento. Observando a ocupação dos discos, verificou-se que não há problemas de espaço nem de lentidão nos acessos.

1.2 Discussão (Questões Motrizes): (Caso julgue necessário, pode tomar notas sobre esta discussão para construir sua solução.)

1.2.1 Fundamentado nos seus conhecimentos adquiridos/aprimorados no Estudo Autônomo, como esta situação pode ser monitorada?

1.2.2 Qual(is) protocolo(s) poderá(ão) ser utilizado(s) para que possamos executar tais monitoramentos? Por quê?

1.2.3 De que forma isto pode ser implementado?

1.3 Resolução:

1.3.1 Descreva qual(is) o(s) provável(is) problema(s) que você identificou e que podem ser monitorados.

1.3.2 Descreva detalhadamente como você construirá a solução para o monitoramento que auxiliará na solução do(s) problema(s). Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajude a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3.3 Construa um programa que possa monitorar as possíveis causas dos problemas.

Conforme a descrição feita no Item 1.3.2, converta a solução em um programa e anexe o arquivo .py. Este programa deverá atender aos seguintes requisitos:

- Executar (“rodar”);
- Resolver os problemas de acordo com o contexto proposto;
- Estar em conformidade com a descrição da construção da solução para o problema;
- A codificação deve ser bem estruturada e eficiente (sem excessos);
- Deve conter comentários que instruem sobre a construção do programa;

Problema 2 – Reflexão e observação

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Um colega do setor de TI está fora da empresa e tentando acessar o Servidor1 remotamente via SSH, sem sucesso. A mensagem é de tempo excedido e a conexão não se completa. O Servidor1 está atrás de um *Firewall* que bloqueia a porta SSH (22) para acessos de fora da rede interna.

Situação B: Seu chefe ligou reclamando que não está conseguindo acessar o site www.cursosparaexecutivos.com, que acessava normalmente até então. Informou que consegue acessar outros sites.

Situação C: Os funcionários do setor de armazenamento de imagens por satélite estão reclamando que não conseguem mais gravar seus arquivos no Servidor3, recebendo a mensagem de que não há espaço em disco.

2.1 Qual das situações (A, B ou C) apresenta um problema similar ao problema 1 e aos recursos trabalhados nele? Justifique.

2.2 Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Abraços e bons estudos.

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 01

Roteiro de aprendizagem para Estudo Autônomo 02

1.2 Cuidando do *habitat*



Esforço previsto estudando autonomamente: 2 horas



Descrição Geral

Car@ estudante,

Este Roteiro de Aprendizagem foi preparado para seu Estudo Autônomo e fornecerá a base teórica para a resolução dos problemas que serão apresentados em nossa aula presencial.

Conforme vimos anteriormente, nos Roteiros de Aprendizagem para estudos autônomos e presenciais, é possível monitorar o estado das redes e dos equipamentos que as compõem. Neste roteiro, veremos que também é necessário o monitoramento da situação interna de cada equipamento que contenha serviços de redes (servidores²⁶ Web, DNS, DHCP, FTP, VPN, VoIP, Firewall, etc.), como ocupação de discos e partições, que podem causar indisponibilidade a qualquer momento. Veremos também como compartilhar arquivos de forma automática para acesso centralizado, através do protocolo NFS (Network File System).

Os conteúdos são disponibilizados em uma ordem que facilita um melhor aproveitamento e sequência de raciocínio. Portanto, siga a sequência e não pule etapas. Caso necessite, complemente seus estudos através de suas próprias pesquisas na Internet.

Compartilhe suas pesquisas com seus colegas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas relacionados a controle de discos e partições que necessitem monitoramento;
- Automatizar o monitoramento do controle de ocupação de discos e partições;

²⁶ Servidor: Servidor é uma máquina que provê serviços específicos, de forma dedicada. Um servidor pode prover um ou mais serviços. Alguns tipos de servidores: Servidor Web: provê a hospedagem de páginas da internet; Servidor DNS (Domain Name System): Servidor de nomes para uma rede ou Internet; Servidor DHCP (Dynamic Host Configuration Protocol): Fornece endereços IP e configurações pertinentes a hosts de uma mesma rede; Servidor FTP (File Transfer Protocol): Oferece uma interface que pode enviar e/ou receber arquivos em uma rede; VPN (Virtual Private Network): Disponibiliza conexões seguras para acessos externos; VoIP (Voice Over IP): Disponibiliza serviço de voz, semelhante à telefonia convencional, entretanto, sobre uma rede TCP/IP; Firewall: Baseado em regras definidas pelo administrador de redes, filtra os pacotes que entram ou que saem da rede, podendo negar ou permitir o tráfego.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

3. Monitoramento de estado de discos e partições, em rede
 - 3.1. Redes de Computadores
 - 3.1.1. Monitoramento de partições
 - 3.1.2. Compartilhamento de arquivos e diretórios em rede
 - 3.1.3. Protocolos de compartilhamento
 - 3.1.4. Protocolos SNMP e NFS
 - 3.2. Programação
 - 3.2.1. Variáveis
 - 3.2.2. Comentários como forma de documentação
 - 3.2.3. Importação de bibliotecas
 - 3.2.4. Utilização de métodos de bibliotecas
 - 3.2.5. Entrada e saída de dados
 - 3.2.6. Conversão de tipos de dados
 - 3.3. Sistemas Operacionais (Linux)
 - 3.3.1. Ocupação de discos e partições
 - 3.3.2. Estrutura de diretórios
 - 3.3.3. Estouro de partições
 - 3.3.4. Compartilhamento de arquivos e diretórios
 - 3.3.5. Compartilhamento via NFS
 - 3.3.6. Estruturas de arquivos e diretórios
 - 3.3.7. Comandos para listagem de arquivos e diretórios



Tarefas

Abaixo são propostas questões que, para serem resolvidas, necessitam de conhecimentos teóricos, cujos conteúdos seguem disponibilizados nos *links* que acompanham cada questão.

É importante que as respostas ocorram na sequência em que são dispostas as questões.

Questão 01

Para resolver os itens abaixo, assista ao vídeo “Monitoramento de partições”. Este vídeo pode ser acessado através do *link*: <https://youtu.be/rvzFzpGdbxU>.

Utilizando o comando “df -h”, (hipoteticamente) você percebeu que a partição / (barra) de um dos seus servidores Linux está com 90% de ocupação. Esta máquina concentra o banco de dados do programa de vendas e caixa da empresa.

Responda os itens abaixo:

- 1.1 Se esta partição chegar a 100%, quais prejuízos você entende que podem ocorrer?
- 1.2 Se esta partição chegar a 100%, como o problema gerado pode ser resolvido?
- 1.3 Como posso prevenir a ocorrência de estouro de partições?
- 1.4 Utilizando o Servidor2 ou o Servidor3, execute, com *snmpwalk*, em um único arquivo.py, cada uma das OIDs abaixo. Utilize o comando df (sem o -h) e deduza para que servem cada uma das seguintes OIDs, semelhante ao que foi visto no vídeo. Anote o que faz cada uma delas:

- a) 1.3.6.1.4.1.2021.9.1.6;
- b) 1.3.6.1.4.1.2021.9.1.7;
- c) 1.3.6.1.4.1.2021.9.1.8;
- d) 1.3.6.1.4.1.2021.9.1.9.

Questão 2

Para resolver os itens abaixo, assista aos vídeos “Sistemas de arquivos e compartilhamento_1” e “Sistemas de arquivos e compartilhamento_2”. Estes vídeos podem ser acessados através dos *links*: https://www.youtube.com/watch?v=f_8vHUvSevA e <https://youtu.be/AucMUMcl1Eg>.

Vídeo complementar recomendado (Opcional):

- Estrutura de Diretórios do Linux: <https://youtu.be/zeX9MbEZSmw> (Curso de Linux - Professor Nathan Brito)

Sites complementares recomendados (Opcionais):

- Instalação do NFS no Ubuntu: <https://tutoriais.wordpress.com/2014/09/26/instalacao-do-servidor-nfs-no-ubuntu-14-04-lts/>. (T. I. Tutoriais).
- Configuração do NFS no Linux: <https://www.vivaolinux.com.br/artigo/Configurando-um-servidor-NFS-em-4-passos>. (Viva o Linux. Fábio José de Matos).

Responda os itens abaixo:

2.1 Conforme observado nos vídeos, os arquivos são compartilhados através da montagem de uma partição da máquina cliente na máquina servidor. Entretanto, após a desmontagem da partição na máquina cliente, esta partição deixará de existir na máquina cliente. Como posso acessar estes arquivos novamente?

- () Os arquivos não poderão mais ser acessados.
- () A partição deverá ser montada novamente.
- () As máquinas servidor e cliente devem ser reinicializadas.
- () A máquina servidor deve ser reinicializada.

2.2 Conforme observado nestes vídeos e nos vídeos dos estudos anteriores (lembra da biblioteca subprocess?), crie um programa capaz de automatizar os subitens abaixo (Lembre-se das boas práticas exemplificadas em nosso primeiro programa: <https://goo.gl/rvnzWy>). Depois faça o *upload* do arquivo .py no local indicado:

- I. a montagem da partição do Servidor3 (que será o servidor de NFS) no Servidor1 (que será o cliente NFS);
- II. a inserção (cópia ou criação) de algum arquivo na partição;
- III. a desmontagem da partição.

Abraços e bons estudos!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 01

Roteiro de aprendizagem para Estudo Presencial 02

1.2 Cuidando do Habitat



Carga horária prevista em sala de aula: 4 horas



Descrição Geral

Caro estudante,

O gerenciamento de partições é um conceito único, entretanto, varia conforme os sistemas operacionais adotados, devido ao sistema de arquivo intrínseco de cada um. A falta de controle sobre partições e discos pode gerar problemas em servidores de redes de computadores afetando a disponibilidade de serviços e informações. Neste roteiro será trabalhada a automatização deste tipo de monitoramento

É importante seguir a sequência em que são dispostos os problemas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas relacionados a controle de discos e partições que necessitem monitoramento;
- Automatizar o monitoramento do controle de ocupação de discos e partições;
- Associar a resolução encontrada com a resolução de outros problemas de redes.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

4. Monitoramento de estado de discos e partições, em rede
 - 4.1. Redes de Computadores
 - 4.1.1. Monitoramento de partições
 - 4.1.2. Compartilhamento de arquivos e diretórios em rede
 - 4.1.3. Protocolos de compartilhamento
 - 4.1.4. Protocolos SNMP e NFS
 - 4.2. Programação
 - 4.2.1. Variáveis
 - 4.2.2. Comentários como forma de documentação
 - 4.2.3. Importação de bibliotecas
 - 4.2.4. Utilização de métodos de bibliotecas
 - 4.2.5. Entrada e saída de dados
 - 4.2.6. Conversão de tipos de dados

4.3. Sistemas Operacionais (Linux)

4.3.1. Ocupação de discos e partições

4.3.2. Estrutura de diretórios

4.3.3. Estouro de partições

4.3.4. Compartilhamento de arquivos e diretórios

4.3.5. Compartilhamento via NFS

4.3.6. Estruturas de arquivos e diretórios

4.3.7. Comandos para listagem de arquivos e diretórios



Tarefas

Abaixo propomos problemas que buscam reproduzir contextos do cotidiano de um profissional que tenha de resolver problemas relacionados a redes de computadores. Cada problema é dividido em três partes: Apresentação do contexto, discussão e resolução. Utilize os conhecimentos acumulados até agora para auxiliar nas resoluções.

É **importante** que as resoluções ocorram na sequência em que os problemas são dispostos.

Problema 1

Obs.: Para resolução deste problema, utilize as máquinas virtuais: Servidor1 e Servidor3.

1.1 Apresentação do contexto (Âncora):

Por medida de segurança, você verificou que é necessário que as máquinas Servidor1 e Servidor03 tenham suas partições e discos monitorados para prevenir estouros de espaços de armazenamento, pois um servidor parado pode significar prejuízos financeiros e de imagem para a empresa.

1.2 Discussão (Questões Motrizes): (Caso julgue necessário, pode tomar notas sobre esta discussão para construir sua solução.)

1.2.1 Qual a melhor estratégia para evitar uma eventual parada por problemas com estouro de espaços de discos ou de partições?

1.2.2 Como podem ser geradas informações para que providências sejam tomadas sempre que um patamar pré-estabelecido de ocupação for atingido?

1.2.3 Isto pode ser feito a partir de um servidor central? Como seria este procedimento?

1.3 Resolução:

1.3.1 Descreva qual(is) o(s) provável(is) problema(s) que você identificou.

1.3.2 Descreva detalhadamente como será construída a solução para o problema. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3.3 Conforme a descrição feita no Item 3.2, converta a solução em um programa. Este programa deverá atender aos seguintes requisitos:

- Executar (“rodar”);
- Resolver os problemas de acordo com o contexto proposto;

- Estar em conformidade com a descrição da construção da solução para o problema;
- A codificação deve ser bem estruturada e eficiente (sem excessos);
- Deve conter comentários que instruem sobre a construção do programa;
- Não deve necessitar de ajustes posteriores para seu correto funcionamento.

Problema 2 – Reflexão e observação

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Houve uma queda de energia durante a madrugada que durou mais tempo que os *no-breaks* suportariam. Todo o *Data Center* ficou sem energia. Pela manhã, quando a energia elétrica foi restabelecida, um dos servidores não religou.

Situação B: Você precisa fazer um levantamento nos servidores sobre itens de *upgrade*. Um destes itens se refere à memória. Após um levantamento sobre o crescimento da demanda, foi concluído que o parâmetro para upgrade seria: sempre que houvesse picos de memória que ocupassem mais de 80% na maioria das medições, deveria ser acrescento mais memória ao servidor.

Situação C: Os funcionários do setor de Recursos Humanos estão reclamando que não conseguem ter acesso à rede. O acesso é dado através de um *Switch*²⁷ não gerenciável que está com todas as luzes apagadas.

2.1 Qual das situações (A, B ou C) apresenta um problema similar ao problema 1 e aos recursos trabalhados no problema 1? Justifique.

2.2 Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Abraços e bons estudos.

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

²⁷ Um *switch* utiliza o conceito de *bridge*, que pertence à camada 2 (camada de enlace do modelo OSI – a segunda, de um total de sete camadas). Entretanto, ao invés de replicar toda a comunicação em todas as portas (*broadcast*), um *switch* possui tecnologia que permite que a comunicação seja enviada somente à porta física de destino. Existem *switches* gerenciáveis e não gerenciáveis. Os gerenciáveis possuem formas de configuração que permitem várias possibilidades de envio e monitoramento da comunicação, inclusive SNMP. Veja mais em: <https://goo.gl/YvSEKJ>.

Curso de Programação para Administração de Redes de Computadores

Módulo 01

Roteiro de atividades para Estudo Autônomo 03

1.3 De olho nas intenções



Esforço previsto estudando autonomamente: 2 horas



Descrição Geral

Caro estudante,

Este Roteiro de Aprendizagem foi preparado para seu Estudo Autônomo e fornecerá a base teórica para a resolução dos problemas que serão apresentados em nossa aula presencial.

Em uma empresa, provavelmente os servidores são acessados por mais de uma pessoa. Intencionalmente ou não, podem ocorrer problemas operacionais, como, por exemplo, a execução equivocada de comandos que podem colocar em risco ou eliminar permanentemente dados de um servidor. Portanto, é importante monitorar o que é executado e por quem é executado, para que se possam eliminar ou minimizar problemas oriundos de acessos indevidos ou do mau uso de prerrogativas, além de facilitar eventuais auditorias sobre anomalias detectadas. Neste contexto, uma das formas de se precaver é monitorando o que cada usuário está fazendo nos servidores. Neste Estudo Autônomo trataremos deste tema e veremos alguns conceitos inerentes.

Os conteúdos são disponibilizados em uma ordem que facilita um melhor aproveitamento e sequência de raciocínio. Portanto, siga a sequência e não pule etapas. Caso necessite, complemente seus estudos através de suas próprias pesquisas na Internet.

Compartilhe suas pesquisas com seus colegas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas relacionados à necessidade de monitorar os acessos e ações de usuários a serviços de redes;
- Automatizar o monitoramento de acessos e ações de usuários a serviços de redes;
- Identificar bibliotecas e métodos para manipulação de datas em Python;
- Conhecer formas de trabalho com arquivos em Python.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

5. Monitoramento de ações de usuários
 - 5.1. Redes de Computadores
 - 5.1.1. Registros de atividades de usuários

- 5.1.2. Acesso remoto via SSH
- 5.1.3. Identificação de endereços IP
- 5.2. Programação
 - 5.2.1. Variáveis
 - 5.2.2. Comentários como forma de documentação
 - 5.2.3. Importação de bibliotecas
 - 5.2.4. Utilização de métodos de bibliotecas
 - 5.2.5. Entrada e saída de dados
 - 5.2.6. Operações sobre arquivos
 - 5.2.7. Manipulação de *strings*
 - 5.2.8. Manipulação de datas
 - 5.2.9. Conversão de tipos de dados
 - 5.2.10. Laço de repetição *for*
 - 5.2.11. Estrutura de decisão *if - else*
- 5.3. Sistemas Operacionais (Linux)
 - 5.3.1. Editores de texto
 - 5.3.2. Acesso remoto
 - 5.3.3. Arquivos de configuração
 - 5.3.4. Hierarquia de usuários
 - 5.3.5. Histórico de ações
 - 5.3.6. Data e hora



Abaixo são propostas questões que, para serem resolvidas, necessitam de conhecimentos teóricos, cujos conteúdos seguem disponibilizados nos *links* que acompanham cada questão.

É importante que as respostas ocorram na sequência em que são dispostas as questões.

Questão 01

Você precisará utilizar um editor de textos em Linux. Existem vários e um dos mais amigáveis e intuitivos é o Nano.

Se for utilizar o editor de textos “Nano” e tiver dúvidas, leia o conteúdo deste *link*: <https://www.vivaolinux.com.br/artigo/Introducao-ao-Linux-O-editor-de-texto-Nano>.

Para resolver os itens abaixo, assista ao vídeo “Monitoramento de ações com history”. Este vídeo pode ser acessado através do *link*: <https://youtu.be/XFndlx7R2MI>.

Conforme você observou, as modificações realizadas no vídeo possibilitaram facilitar o monitoramento de usuários específicos, quanto ao que executam, em modo texto, em uma máquina Linux.

Resolva:

1.1 Seguindo o exemplo do vídeo, faça as mesmas modificações para o “user1” do servidor1, de forma que possam ser monitoradas as ações deste usuário. Lembrando que as linhas a serem acrescentadas no arquivo `.bashrc` são:

```
HISTTIMEFORMAT="%d/%m/%y %T " e
```

```
PROMPT_COMMAND="history -a; history -c; history -r; "
```

OBS. É muito importante que você se certifique de que as alterações foram efetivas. Caso contrário,

pode não conseguir concluir as próximas questões. Portanto, ao inserir as linhas acima, verifique cuidadosamente a grafia, preservando maiúsculas, minúsculas e, principalmente, os espaços em branco. Execute um SSH a partir de outro servidor, para testar.

Questão 2

Para responder o item seguinte, assista aos vídeos abaixo. É importante que você reproduza os exemplos apresentados durante os vídeos para uma melhor assimilação dos conteúdos. Estes vídeos podem ser acessados através dos *links* abaixo:

Sobre manipulação de arquivos, assista aos dois vídeos seguintes:

https://youtu.be/C9_DTR1ICIs?list=PLUukMN0DTKctbzhbYe2jdF4cr8MOWCIXc – Arquivos – Parte1 – Python para Zumbis

<https://youtu.be/5z6zqyeLxjg?list=PLUukMN0DTKctbzhbYe2jdF4cr8MOWCIXc> – Arquivos – Parte2 – Python para Zumbis

Sobre métodos de manipulação de strings (se quiser, pode avançar o vídeo para 4 minutos e 14 segundos, onde trata dos métodos “split” e “join”):

<https://youtu.be/OInhmFfmNXE?list=PLUukMN0DTKctbzhbYe2jdF4cr8MOWCIXc> – Métodos para manipulação de strings – Python para Zumbis

Conversão de data, do formato Linux para um formato legível, com Python.

<https://youtu.be/x6P1gNN2CLs> – Conversão de data Linux com Python.

Resolva:

2.1 Você desconfia que o usuário user1 está acessando o servidor1 em horários e dias não convencionais e acessando a máquina como “root”. Como primeiro passo, você decidiu ver todos os comandos “sudo” executados por ele. Assim, construa o programa para esta verificação.

Questão 3

Para responder o item seguinte, assista aos vídeos abaixo. É importante que você reproduza os exemplos apresentados durante os vídeos para uma melhor assimilação dos conteúdos. Estes vídeos podem ser acessados através dos *links* abaixo:

Sobre manipulação de arquivos, assista aos dois vídeos seguintes:

https://youtu.be/C9_DTR1ICIs?list=PLUukMN0DTKctbzhbYe2jdF4cr8MOWCIXc – Arquivos – Parte1 – Python para Zumbis

<https://youtu.be/5z6zqyeLxjg?list=PLUukMN0DTKctbzhbYe2jdF4cr8MOWCIXc> – Arquivos – Parte2 – Python para Zumbis

Assista também ao vídeo “Gravando dados em arquivo”. Este vídeo pode ser acessado através do *link*: <https://youtu.be/Os8l-vwnyas>

Resolva:

3.1 Agora, modifique o programa que você construiu no item 2.1 da questão 2, de forma que o mesmo grave os resultados em um arquivo.

Abraços e bons estudos!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 01

Roteiro de aprendizagem para Estudo Presencial 03

1.3 De olho nas intenções



Carga horária prevista em sala de aula: 4 horas



Descrição Geral

Caro estudante,

Conforme visto no Estudo Autônomo, há maneiras de monitorar algumas ações de usuários sobre servidores que comportam serviços de redes, para fins de auditoria ou para recuperação de problemas devido a erros ou intencionalidades.

Neste Roteiro de Aprendizagem são apresentados problemas que envolvem o monitoramento destas ações.

É importante seguir a sequência em que são dispostos os problemas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas relacionados à necessidade de monitorar os acessos e ações de usuários a serviços de redes;
- Automatizar o monitoramento de acessos e ações de usuários a serviços de redes;
- Associar a resolução encontrada com a resolução de outros problemas de redes.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

6. Monitoramento de ações de usuários
 - 6.1. Redes de Computadores
 - 6.1.1.Registros de atividades de usuários
 - 6.1.2.Acesso remoto via SSH
 - 6.1.3.Identificação de endereços IP
 - 6.2. Programação
 - 6.2.1.Variáveis
 - 6.2.2.Comentários como forma de documentação
 - 6.2.3.Importação de bibliotecas
 - 6.2.4.Utilização de métodos de bibliotecas
 - 6.2.5.Entrada e saída de dados

- 6.2.6. Operações sobre arquivos
- 6.2.7. Manipulação de *strings*
- 6.2.8. Manipulação de datas
- 6.2.9. Conversão de tipos de dados
- 6.2.10. Laço de repetição *for*
- 6.2.11. Estrutura de decisão *if - else*
- 6.3. Sistemas Operacionais (Linux)
 - 6.3.1. Editores de texto
 - 6.3.2. Acesso remoto
 - 6.3.3. Arquivos de configuração
 - 6.3.4. Hierarquia de usuários
 - 6.3.5. Histórico de ações
 - 6.3.6. Data e hora



Tarefas

Abaixo, propomos problemas que buscam reproduzir contextos do cotidiano de um profissional que tenha de resolver problemas relacionados a redes de computadores. Cada problema é dividido em três partes: Apresentação do contexto, discussão e resolução. Utilize os conhecimentos acumulados até agora para auxiliar nas resoluções.

É importante que as resoluções ocorram na sequência em que os problemas são dispostos.

Problema 1

Obs.: Para resolução deste problema, utilize as máquinas virtuais: Servidor1 e Servidor3.

1.1 Apresentação do contexto (Âncora):

No Servidor1, como é padrão nas máquinas Ubuntu Linux, há um arquivo que guarda os registros ou logs de autenticação.

O arquivo fica no seguinte caminho: `/var/log/auth.log`

Neste servidor há um usuário chamado `user2`, que não possui a senha de root.

Você desconfia que usuário `user2` efetuou tentativas para descobrir a senha de root. Há também a desconfiança de que o usuário apagou seus comandos do history.

É necessário averiguar no arquivo `auth.log` se há registros de tentativas de escalada de privilégios do `user2`. Caso existam tentativas registradas, será necessário apresentá-las aos seus superiores, através de relatório.

(Para mudar para `user2`, no console, digite: `su user2`. A senha do `user2` é `user2@`. Simule tentativas de acesso ao root com senhas incorretas.)

1.2 Discussão (Questões Motrizes): (Caso julgue necessário, pode tomar notas sobre esta discussão para construir sua solução.)

1.2.1 Qual o problema central?

1.2.2 Como analisar o arquivo `auth.log`, em busca das entradas específicas?

1.2.3 Como descobrir padrões gravados no arquivo `auth.log` quando há erros de tentativa de acesso como superusuário?

1.2.4 Quais recursos vistos no Estudo Autônomo podem ser aplicados nesta solução?

1.3 Resolução:

1.3.1 Descreva qual(is) o(s) provável(is) problema(s) que você identificou.

1.3.2 Descreva detalhadamente como será construída a solução para o problema. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3.3 Conforme a descrição feita no Item 3.2, converta a solução em um programa. Este programa deverá atender aos seguintes requisitos:

- Executar (“rodar”);
- Resolver os problemas de acordo com o contexto proposto;
- Estar em conformidade com a descrição da construção da solução para o problema;
- A codificação deve ser bem estruturada e eficiente (sem excessos);
- Deve conter comentários que instruem sobre a construção do programa;

Problema 2 – Reflexão e observação

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Na empresa em que você trabalha há um servidor de banco de dados que está com problemas. Para resolver, foi contratado um especialista que fará acesso remoto via SSH. Assim, será necessário criar um usuário específico para tal acesso. O seu chefe solicitou que tudo o que fosse executado durante a resolução fosse registrado em um arquivo para eventuais consultas caso o problema se repita.

Situação B: Na lista de discussão sobre Linux que você participa, foi divulgada uma falha de segurança envolvendo o aplicativo de DNS. Imediatamente você acessou seu servidor para verificar a versão e verificou que o aplicativo está desatualizado.

Situação C: Alguns funcionários antigos, do setor de tecnologia da informação, deixaram a empresa antes de você entrar. Fazendo uma varredura nos servidores, você verificou que estes ex-funcionários ainda possuem usuários e permissões de acesso. Verificou também que os usuários têm permissão de acesso remoto via SSH.

2.1 Qual das situações (A, B ou C) apresenta um problema similar ao problema 1 e aos recursos trabalhados no problema 1? Justifique.

2.2 Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Abraços e bons estudos.

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 02

Roteiro de aprendizagem para Estudo Autônomo 04

2.1 Construir é difícil. Reconstruir pode ser impossível



Esforço previsto estudando autonomamente: 1 hora e meia



Descrição Geral

Car@ estudante,

Neste estudo, iniciamos o Módulo 02 do nosso curso!

Este Roteiro de Aprendizagem foi preparado para seu Estudo Autônomo e fornecerá a base teórica para a resolução dos problemas que serão apresentados em nossa aula presencial.

Os problemas que ocasionam indisponibilidade de dados são diversos e podem envolver hardware, software, erros humanos ou intenções maléficas. Um problema que gere perdas irreversíveis de dados pode significar o fim de uma empresa. Para a prevenção destes problemas, a cópia de segurança ou *backup* é imprescindível. Neste Roteiro veremos como automatizar *backups*.

Os conteúdos são disponibilizados em uma ordem que facilita um melhor aproveitamento e sequência de raciocínio. Portanto, siga a sequência e não pule etapas. Caso necessite, complemente seus estudos através de suas próprias pesquisas na Internet.

Compartilhe suas pesquisas com seus colegas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas relacionados à execução de cópias de segurança (*backups*) em rede;
- Automatizar e monitorar a execução de *backups* em rede.
- Conhecer formas de empacotamento e compactação de arquivos.
- Identificar necessidades e meios de agendar execução de programas em Python, no Linux.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

7. Compactação de dados, cópias de segurança em rede e registros de atividades (logs).
 - 7.1. Redes de computadores
 - 7.1.1. Cópia de segurança (*Backups*) remotas
 - 7.1.2. Compartilhamento de arquivos e diretórios
 - 7.1.3. Identificação de endereços IP
 - 7.1.4. Protocolo NFS

- 7.1.5. Protocolo NTP
- 7.2. Programação
 - 7.2.1. Variáveis
 - 7.2.2. Comentários como forma de documentação
 - 7.2.3. Importação de bibliotecas
 - 7.2.4. Utilização de métodos de bibliotecas
 - 7.2.5. Entrada e saída de dados
 - 7.2.6. Manipulação de *strings*
 - 7.2.7. Monitoramento de estado da execução de comandos do S. O.
 - 7.2.8. Estrutura de decisão *if – else*
 - 7.2.9. Manipulação de arquivos
 - 7.2.10. Manipulação de datas
 - 7.2.11. Conversão de tipos de dados
- 7.3. Sistemas Operacionais (Linux)
 - 7.3.1. Empacotamento de dados
 - 7.3.2. Compactação de dados
 - 7.3.3. Montagem de partições remotas
 - 7.3.4. Conferência de estado de execução de comandos
 - 7.3.5. Agendamento de execução de programas ou comandos
 - 7.3.6. Arquivos de registros (*logs*)
 - 7.3.7. Sincronismo de data e hora



Abaixo são propostas questões que, para serem resolvidas, necessitam de conhecimentos teóricos, cujos conteúdos seguem disponibilizados nos *links* que acompanham cada questão.

É importante que as respostas ocorram na sequência em que são dispostas as questões.

Questão 1

Para resolver o item abaixo, assista ao vídeo “Empacotando arquivos com tar”. Este vídeo pode ser acessado através do *link*: https://youtu.be/6_HfPU8D8Aw

Para a questão 1.1, analise cada exemplo de utilização do comando tar e associe com sua respectiva função:

- A) tar cvfz /tmp/contaCorrente.tar /contabilidade/contaCorrente*
- B) tar cvfz /tmp/contaCorrente.tar /contabilidade/
- C) tar tvf /tmp/contaCorrente.tar
- D) tar xvfz /tmp/contaCorrente.tar -C /mnt/
- E) tar xvfz /tmp/contaCorrente.tar

- () Cria um arquivo chamado contaCorrente.tar, empacotando e compactando todos os arquivos do diretório chamado contabilidade.
- () Descompacta o conteúdo de contaCorrente.tar no diretório atual.
- () Visualiza o conteúdo do arquivo contaCorrente.tar.

() Cria um arquivo chamado `contaCorrente.tar`, empacotando e compactando os arquivos do diretório chamado `contabilidade` que iniciem com `contaCorrente` no nome.

() Descompacta o conteúdo de `contaCorrente.tar` em outro diretório.

1.1 A Associação correta teria a seguinte sequência (Assinale a correta):

A, C, D, B, E

D, A, B, C, E

B, E, C, A, D

Questão 2

Antes de resolver esta questão, é necessário adicionar o diretório `/backup` no Servidor1. Como root, digite:

```
mkdir /backup
```

Para resolver o item abaixo, assista ao vídeo “Backup em rede no Linux com Python”. Este vídeo pode ser acessado através do *link*: <https://youtu.be/LWGkEM7loEs>

Conforme observado no vídeo, é possível criar um *backup* em rede, empacotando e compactando arquivos.

No próximo vídeo, veja como testar a execução de cada comando do programa através do vídeo “Teste de estado da execução do comando”. Este vídeo pode ser acessado através do *link*: <https://youtu.be/JGTqAwz2eek>

Resolva:

2.1 O programa “*backup01.py*”, conforme visto, está funcionando perfeitamente, entretanto, existe a necessidade de modificá-lo, de forma que a execução de cada comando do programa possa ser testado no momento em que é executado. Você precisa resolver este problema.

Questão 3

Para resolver o item abaixo, assista ao vídeo “Agendamento de execuções com o Crontab”. Este vídeo pode ser acessado através do *link*: <https://youtu.be/OtxvUigIM1w>

Como visto, o aplicativo Crontab possibilita o agendamento da execução de programas no Linux.

Resolva:

3.1 Existe um comando no Linux chamado *ntpdate*, que é utilizado para atualizar a data e hora do sistema, sincronizando-o a algum provedor de serviços de data e hora na Internet. Para usar o relógio da RNP (Rede Nacional de Ensino e Pesquisa), o comando é `ntpdate -u ntp.caais.rnp.br`. O executável do *ntpdate* fica em `/usr/sbin/ntpdate`.

Analisando, você concluiu que é necessário que o sincronismo de horário com a RNP seja executado uma vez por semana, todo domingo, às 2 horas da manhã.

Escreva abaixo como ficaria esta entrada no crontab:

Abraços e bons estudos!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 02

Roteiro de aprendizagem para Estudo Presencial 04

2.1 Construir é difícil. Reconstruir pode ser impossível



Carga horária prevista em sala de aula: 4 horas



Descrição Geral

Caro estudante,

Conforme visto no Estudo Autônomo, as cópias de segurança podem significar a continuidade de negócios de uma empresa após um desastre. Seguindo com este estudo, serão apresentados problemas que envolvam a necessidade cópias de segurança em rede. É importante seguir a sequência em que são dispostos os problemas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas relacionados à execução de cópias de segurança (*backups*) em rede;
- Automatizar e monitorar a execução de *backups* em rede.
- Associar a resolução encontrada com a resolução de outros problemas de redes.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

8. Compactação de dados, cópias de segurança em rede e registros de atividades (logs).
 - 8.1. Redes de computadores
 - 8.1.1. Cópias de segurança (*Backups*) remotas
 - 8.1.2. Compartilhamento de arquivos e diretórios
 - 8.1.3. Identificação de endereços IP
 - 8.1.4. Protocolo NFS
 - 8.1.5. Protocolo NTP
 - 8.2. Programação
 - 8.2.1. Variáveis
 - 8.2.2. Comentários como forma de documentação
 - 8.2.3. Importação de bibliotecas
 - 8.2.4. Utilização de métodos de bibliotecas
 - 8.2.5. Entrada e saída de dados
 - 8.2.6. Manipulação de *strings*

- 8.2.7. Monitoramento de estado da execução de comandos do S. O.
- 8.2.8. Estrutura de decisão *if – else*
- 8.2.9. Manipulação de arquivos
- 8.2.10. Manipulação de datas
- 8.2.11. Conversão de tipos de dados
- 8.3. Sistemas Operacionais (Linux)
 - 8.3.1. Empacotamento de dados
 - 8.3.2. Compactação de dados
 - 8.3.3. Montagem de partições remotas
 - 8.3.4. Conferência de estado de execução de comandos
 - 8.3.5. Agendamento de execução de programas ou comandos
 - 8.3.6. Arquivos de registros (*logs*)
 - 8.3.7. Sincronismo de data e hora



Abaixo, propomos problemas que buscam reproduzir contextos do cotidiano de um profissional que tenha de resolver problemas relacionados a redes de computadores. Cada problema é dividido em três partes: Apresentação do contexto, discussão e resolução. Utilize os conhecimentos acumulados até agora para auxiliar nas resoluções.

É importante que as resoluções ocorram na sequência em que os problemas são dispostos.

Problema 1

Obs.: Para resolução deste problema, utilize as máquinas virtuais: Servidor1 (Thonny) e Servidor3.

1.1 Apresentação do contexto (Âncora):

Você entrou recentemente na empresa e imediatamente verificou que não são feitos *backups* dos servidores. Preocupado com a situação, decidiu iniciar pelas configurações (/etc), do Servidor1. Os *backups* devem ser feitos todos os dias, de madrugada, compactados e armazenados no próprio Servidor1, por enquanto. Também é necessário verificar todas as manhãs o resultado do *backup* executado na madrugada: se o *backup* foi executado; em que horário começou e em que horário foi finalizado; além da verificação da execução do procedimento de empacotamento e compactação, se foram ou não executados com sucesso.

1.2 Discussão (Questões Motrizes): (Caso julgue necessário, pode tomar notas sobre esta discussão para construir sua solução.)

1.2.1 Por que é necessário fazer *backups*?

1.2.2 De que forma os recursos vistos nos estudos autônomos podem ser aplicados nesta solução? Explique.

1.2.3 Este procedimento poderia ser utilizado para fazer os *backups* de outros servidores? Se sim, de que forma? Se não, por quê?

1.3 Resolução:

1.3.1 Descreva qual(is) o(s) provável(is) problema(s) que devem ser solucionados, conforme o contexto apresentado acima.

1.3.2 Descreva detalhadamente como será construída a solução para o problema. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3.3 Conforme a descrição feita no Item 1.3.2, converta a solução em um programa e anexe o arquivo .py. Este programa deverá atender aos seguintes requisitos:

- Executar (“rodar”);
- Resolver os problemas de acordo com o contexto proposto;
- Estar em conformidade com a descrição da construção da solução para o problema;
- A codificação deve ser bem estruturada e eficiente (sem excessos);
- Deve conter comentários que instruem sobre a construção do programa.

Problema 2 – Reflexão e observação

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Uma vez por mês, deve ser realizado, de forma automatizada, uma atualização de pacotes (update do apt-get) no servidor de bancos de dados, entretanto, antes que isto aconteça, você deve se certificar de que, em caso de alguma falha que possa interromper o seu funcionamento, estejam sendo executadas, diariamente, cópias de segurança dos arquivos de bancos de dados, de forma segura em outro servidor, de maneira que possam ser recuperadas em uma eventual reconstrução. Para verificar se as cópias foram efetivamente realizadas, deve ser possível analisar em arquivo de log se o procedimento foi realizado com sucesso e ver a relação de arquivos copiados.

Situação B: Os arquivos dos usuários de um dos seus servidores estão ocupando 75% da partição a eles destinada. O percentual máximo de alerta, de 70%, foi ultrapassado. Caso a partição atinja 100% de ocupação, problemas podem ocorrer. Como é uma máquina antiga, não compensa, financeiramente, executar um upgrade de discos. Entretanto há uma máquina nova, com espaço em disco, memória e processamento muito superiores, à disposição.

Situação C: Um funcionário do setor de RH está reclamando que seu aplicativo de mensagens instantâneas, onde faz parte de um grupo de discussão da empresa, que roda em seu celular empresarial, teve todos os dados perdidos, após uma restauração de configurações de fábrica. Segundo o mesmo, seus dados não eram sincronizados na nuvem. Por conta deste problema, o funcionário solicita auxílio para recuperar seus dados.

2.1 Qual das situações (A, B ou C) apresenta um problema similar ao problema 1 e aos recursos trabalhados no problema 1? Justifique.

2.2 Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Abraços e bons estudos.

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 02

Roteiro de aprendizagem para Estudo Autônomo 05

2.2 Um é pouco. Então tenha dois (ou mais!)



Esforço previsto estudando autonomamente: 2 horas



Descrição Geral

Caro estudante,

A cópia de segurança é uma estratégia fundamental para garantir a sobrevivência de qualquer dado importante, entretanto, alguns serviços exigem disponibilidade ininterrupta (ou o mais próximo disso...). Nestes casos, recuperar uma cópia de segurança pode representar um custo de tempo que você não possui. Outro risco é um problema físico que pode deixar inoperante o equipamento até que você o substitua. Durante este tempo de substituição, deverá haver a instalação e configuração do servidor, para que, só então, possa ser recuperada a cópia de segurança. A depender do ramo da empresa, esta indisponibilidade pode representar prejuízos. Neste ponto entra o sincronismo de dados que pode ser construído e controlado com auxílio da programação. Por exemplo, em um ambiente em que existam dois servidores funcionando com dados importantes sincronizados entre si: em caso de falha do servidor principal, o servidor auxiliar pode ser ativado rapidamente, resultando em um tempo menor de serviços inoperantes.

Os conteúdos são disponibilizados em uma ordem que facilita um melhor aproveitamento e sequência de raciocínio. Portanto, siga a sequência e não pule etapas. Caso necessite, complemente seus estudos através de suas próprias pesquisas na Internet.

Compartilhe suas pesquisas com seus colegas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas que possam ser resolvidos com a aplicação de sincronismo de dados;
- Entender as vantagens do sincronismo de dados para preservar as informações mais críticas e que mereçam tal procedimento;
- Entender a importância de uma política de *backup*;
- Entender a importância do manutenção de *backups* e sua independência quanto ao sincronismo de dados;
- Aplicar o *rsync* como meio para sincronismo de dados em Linux.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

9. Sincronismo de dados em rede
 - 9.1. Redes de computadores
 - 9.1.1. Sincronismo remoto de dados
 - 9.1.2. Cópias de segurança (*backups*)
 - 9.1.3. Políticas de *backup*
 - 9.1.4. Riscos e prejuízos causados por sistemas fora de operação
 - 9.1.5. Protocolo SSH
 - 9.1.6. Protocolo NFS
 - 9.2. Programação
 - 9.2.1. Variáveis
 - 9.2.2. Comentários como forma de documentação
 - 9.2.3. Importação de bibliotecas
 - 9.2.4. Utilização de métodos de bibliotecas
 - 9.2.5. Entrada e saída de dados
 - 9.3. Sistemas Operacionais (Linux)
 - 9.3.1. Sincronismo de dados
 - 9.3.2. Agendamento de procedimentos
 - 9.3.3. Comandos do S. O.
 - 9.3.4. Montagem de partições remotas



Abaixo são propostas questões que, para serem resolvidas, necessitam de conhecimentos teóricos, cujos conteúdos seguem disponibilizados nos *links* que acompanham cada questão.

É importante que as respostas ocorram na sequência em que são dispostas as questões.

Questão 1

Para resolver os itens abaixo, assista ao vídeo “Sincronismo de dados com rsync”. Este vídeo pode ser acessado através do *link*: <https://youtu.be/bX2wZ4RMDeI>

1.1 Na empresa em que você começou a trabalhar recentemente, não há *backups* sendo executados para o servidor Web. Ao questionar os funcionários mais antigos, foi informado que foi investido em um servidor que funciona em paralelo, sincronizando todos os dados das páginas Web, através do uso do rsync. Sendo o novo responsável pelo setor de TI, você manterá os mesmos procedimentos do seu antecessor quanto a este servidor? Justifique.

1.2 Você precisa desenvolver um programa que seja capaz de sincronizar todos os dados constantes no diretório /home/user1/Documentos/, do servidor1, com o diretório /home/user1/sincronismo/, no servidor3. O sincronismo deve ocorrer automaticamente todos os dias, às 20 horas.

Questão 2

Para resolver o item abaixo, leia atentamente o que é disposto na página deste link: <https://goo.gl/ZFGWYr>. Trata-se de um modelo de política de *backup* disposto no site Bacula do Brasil. Considere que o modelo apresentado é a política de *backup* da empresa na qual você trabalha.

Na data de hoje, você, que é o responsável pelo Setor de TI, recebeu um memorando com os

seguintes termos:

Ao Setor de TI

Caro responsável,

Necessitamos, com URGÊNCIA, que seja recuperado o arquivo "reunião com importadores.docx" contidos em cópia de segurança (*backup*) gravado no servidor de arquivos na data de 04 de fevereiro de 2015. O arquivo foi apagado acidentalmente em 08 de março de 2015 e necessitamos destes dados para a preparação de nova reunião que ocorrerá na próxima segunda-feira.

Informamos que o não atendimento desta solicitação causará prejuízos à empresa, podendo V. Sa. ser responsabilizado.

Atenciosamente,

A diretoria.

2.1 Responda ao memorando, considerando a política de *backup* da empresa.

2.2 Diante do contexto desta questão, escreva sobre a importância da política de *backup* para os responsáveis pela execução dos *backups*:

Abraços e bons estudos!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 02

Roteiro de aprendizagem para Estudo Presencial 05

2.2 Um é pouco. Então tenha dois! (ou mais)



Carga horária prevista em sala de aula: 4 horas



Descrição Geral

Car@ estudante,

Conforme visto no Estudo Autônomo, o sincronismo de dados pode ser muito útil em aplicações que devem estar disponíveis o máximo de tempo possível. É importante frisar que o sincronismo de dados não substitui a construção e retenção dos *backups*.

Neste Estudo Presencial, trabalharemos problemas que exigem a aplicação de sincronismo de dados.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Identificar problemas que possam ser resolvidos com a aplicação de sincronismo de dados;
- Automatizar e monitorar a execução de sincronismo de dados em rede;
- Aplicar o rsync como meio para sincronismo de dados em Linux;
- Associar a resolução encontrada com a resolução de outros problemas de redes.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

10. Sincronismo de dados em rede

10.1. Redes de computadores

10.1.1. Sincronismo remoto de dados

10.1.2. Cópias de segurança (*backups*)

10.1.3. Políticas de *backup*

10.1.4. Riscos e prejuízos causados por sistemas fora de operação

10.1.5. Protocolo SSH

10.1.6. Protocolo NFS

10.2. Programação

10.2.1. Variáveis

10.2.2. Comentários como forma de documentação

10.2.3. Importação de bibliotecas

10.2.4. Utilização de métodos de bibliotecas

- 10.2.5. Entrada e saída de dados
- 10.3. Sistemas Operacionais (Linux)
 - 10.3.1. Sincronismo de dados
 - 10.3.2. Agendamento de procedimentos
 - 10.3.3. Comandos do S. O.
 - 10.3.4. Montagem de partições remotas



Abaixo, propomos problemas que buscam reproduzir contextos do cotidiano de um profissional que tenha de resolver problemas relacionados a redes de computadores. Cada problema é dividido em três partes: Apresentação do contexto, discussão e resolução. Utilize os conhecimentos acumulados até agora para auxiliar nas resoluções.

É **importante** que as resoluções ocorram na sequência em que os problemas são dispostos.

Problema 1

Obs.: Para resolução deste problema, utilize as máquinas virtuais: Servidor1 e Servidor3.

1.1 Apresentação do contexto (Âncora):

Como na empresa onde você trabalha, ainda não foi definida uma política de *backup*, no intuito de melhorar a segurança e garantir menor indisponibilidade de alguns serviços constantes no Servidor1 em caso de falhas, você decidiu seguir a seguinte estratégia, até que uma política seja definida e aprovada:

Todos os dias, às 19 horas, é necessário que os arquivos do diretório `/etc/` do Servidor1 sejam sincronizados com o diretório `/home/user1/sincronismo/`, do Servidor3.

É importante que seja gerado um arquivo de texto com extensão `.log`, que ficará no próprio Servidor1, contendo a data e hora de início e fim da execução. Caso ocorra algum erro durante a tentativa de sincronismo ou de outros procedimentos, também deve constar no arquivo.

No Servidor3, às 22 horas, é necessário que todo o conteúdo existente no diretório `/home/user1/sincronismo/` seja compactado e copiado para o diretório `/backup/`.

1.2 Discussão (Questões Motrizes): (Caso julgue necessário, pode tomar notas sobre esta discussão para construir sua solução.)

1.2.1 Por que não se pode esperar até que a política de *backup* tenha sido definida e aprovada?

1.2.2 Quais os procedimentos para solucionar o problema em cada servidor?

1.2.3 O sincronismo poderia ser realizado mais de uma vez por dia? Seria necessário? Justifique.

1.3 Resolução:

1.3.1 Dentro do contexto apresentado, descreva qual(is) o(s) provável(is) problema(s) que você identificou quanto à automatização necessária.

1.3.2 Descreva detalhadamente como será construída a solução para o problema. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3.3 Conforme a descrição feita no Item 1.3.2, converta a solução em um programa. Este programa deverá atender aos seguintes requisitos:

- Executar (“rodar”);
- Resolver os problemas de acordo com o contexto proposto;
- Estar em conformidade com a descrição da construção da solução para o problema;
- A codificação deve ser bem estruturada e eficiente (sem excessos);
- Deve conter comentários que instruem sobre a construção do programa;

Problema 2 – Reflexão e observação

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Um colega do setor de TI está fora da empresa e tentando acessar o Servidor1 remotamente via SSH, sem sucesso. A mensagem é de tempo excedido e a conexão não se completa. O Servidor1 está atrás de um Firewall que bloqueia a porta SSH (22) para acessos de fora da rede interna.

Situação B: Os *backups* da empresa em que você trabalha são armazenados todas as noites em fitas. Após um problema no HD do servidor *Web*, houve a necessidade da recuperação do último *backup*. Ao tentar recuperar os dados, você verificou que a mídia estava vazia.

Situação C: A empresa RS Advogados possui vários servidores e um deles é específico para o armazenamento dos arquivos dos associados, cujo *backup* é realizado em fita todos os dias. Entretanto, recentemente, houve um problema neste servidor, (depois diagnosticado como um problema na controladora de discos) que indisponibilizou os dados por 8 horas comerciais até que pudesse ser sanado e o *backup* recuperado. Este tempo foi suficiente para que causasse transtornos quanto a processos que necessitavam tramitar no início do próximo dia.

Você foi contratado para que desenvolva uma solução em rede para que este problema seja minimizado, caso aconteça novamente. Para tanto, a empresa disponibilizou dois servidores Linux, sendo o primeiro, o servidor de arquivos. O segundo é um servidor com Linux instalado e configurado para funcionar na rede, com as mesmas características do primeiro, entretanto, sem os arquivos.

Resolução:

2.1 Qual das situações (A, B ou C) apresenta um problema similar ao problema 1 e aos recursos trabalhados no problema 1? Justifique.

2.2 Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Abraços e bons estudos.

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 02

Roteiro de aprendizagem para Estudo Autônomo 06

2.3 Seja o primeiro a saber.



Esforço previsto estudando autonomamente: 2 horas



Descrição Geral

Caro estudante,

As soluções construídas até o momento possibilitam gerar informações sobre suas execuções e são fixadas nos servidores, aguardando até que você as acesse. Assim, caso ocorram problemas, estes poderão ser conhecidos na primeira oportunidade de acesso aos servidores e, conforme a situação, tomadas as providências adequadas. Entretanto, por conta de medidas urgentes que devam ser tomadas frente a eventuais problemas específicos, algumas ocorrências exigem ações mais breves e suas informações devem estar disponíveis imediatamente. Assim, informações via correio eletrônico e mensageria instantânea são essenciais para atender este tipo de necessidade.

Os conteúdos são disponibilizados em uma ordem que facilita um melhor aproveitamento e sequência de raciocínio. Portanto, siga a sequência e não pule etapas. Caso necessite, complemente seus estudos através de suas próprias pesquisas na Internet.

Compartilhe suas pesquisas com seus colegas.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Enviar mensagens de informação e de alerta, através de *e-mail* e de mensageria instantânea.
- Identificar problemas que necessitem de monitoramento constante e de informação imediata de alertas;
- Automatizar o envio de mensagens de alerta referentes a programas executados por agendamento;



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

11. Automação de mensagens de eventos e de alertas via *e-mail* e mensageria instantânea.

11.1. Redes de computadores

11.1.1. Envio de mensagens via *e-mail*

11.1.2. Envio de mensagens via mensageria instantânea (Telegram)

11.1.3. Protocolo SMTP

11.1.4. Protocolos SSL/TLS e StartTLS

- 11.1.5. Servidores de *e-mail*
- 11.1.6. Configuração do Gmail para envio de mensagens a partir do Servidor Linux
- 11.1.7. Protocolo ICMP
- 11.2. Programação
 - 11.2.1. Variáveis
 - 11.2.2. Comentários como forma de documentação
 - 11.2.3. Importação de bibliotecas
 - 11.2.4. Utilização de métodos de bibliotecas
 - 11.2.5. Entrada e saída de dados
 - 11.2.6. Envio de *e-mails*
 - 11.2.7. Manipulação de *strings*
- 11.3. Sistemas Operacionais (Linux)
 - 11.3.1. Envio de e-mails
 - 11.3.2. Envio de mensagens instantâneas
 - 11.3.3. Agendamento de procedimentos
 - 11.3.4. Comandos do S. O.
 - 11.3.5. Procedimentos em *Shell Script*
 - 11.3.6. Configuração de aplicativos



Abaixo são propostas questões que, para serem resolvidas, necessitam de conhecimentos teóricos, cujos conteúdos seguem disponibilizados nos *links* que acompanham cada questão.

É **importante** que as respostas ocorram na sequência em que são dispostas as questões.

Questão 1

Para resolver o item abaixo, assista ao vídeo “Enviando *e-mails* com Python”. Este vídeo pode ser acessado através do *link*: <https://youtu.be/QWbieMzpNtk>

1.1 É necessário realizar um *backup* compactado, todos os dias, às 22 horas, do conteúdo do diretório `/home/user1/` do Servidor1. Deve ser composta uma mensagem contendo: 1) Data/hora de início e fim do *backup*; 2) Estado de cada procedimento executado no *backup*; 3) Lista de tudo o que foi “*backupeado*” (pode usar o `tar tvf nomedoarquivo.tar` para gerar a lista).

Questão 2

Para resolver o item abaixo, siga o “Tutorial Telegram”. Este tutorial pode ser acessado através do *link*: <https://drive.google.com/file/d/1IrIWGtLYO990PW91J-Mkfi7hPEGQclfk/view?usp=sharing>

2.1 Você desconfia de que o Servidor3 está perdendo a comunicação em determinados momentos com o Servidor1. Assim, há a necessidade de verificar, através do comando PING, se o Servidor3 está respondendo. A verificação deverá ocorrer a cada 5 minutos e cada vez que o servidor não responder, deverá ser enviada uma mensagem instantânea ao seu celular.

Abraços e bons estudos!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

Curso de Programação para Administração de Redes de Computadores

Módulo 02

Roteiro de aprendizagem para Estudo Presencial 06

2.3 Seja o primeiro a saber.



Carga horária prevista em sala de aula: 4 horas



Descrição Geral

Car@ estudante,

No Estudo Autônomo vimos que recursos de comunicação como *e-mails* ou mensageria instantânea podem ser utilizados para informar procedimentos realizados e/ou mensagens de alerta.

Neste Estudo Presencial, vamos trabalhar problemas em que tais recursos podem ser empregados.

Desejo a você um bom trabalho!



Resultados Pretendidos da Aprendizagem

- Enviar mensagens de informação e de alerta, através de *e-mail* e de mensageria instantânea;
- Identificar problemas que necessitem de monitoramento constante e de informação imediata de alertas;
- Automatizar o envio de mensagens de alerta referentes a programas executados por agendamento;
- Associar a resolução encontrada com a resolução de outros problemas de redes.



Conteúdos envolvidos

Os conteúdos abordados nestes estudos são:

12. Automatização de mensagens de eventos e de alertas via *e-mail* e mensageria instantânea.

12.1. Redes de computadores

12.1.1. Envio de mensagens via *e-mail*

12.1.2. Envio de mensagens via mensageria instantânea (Telegram)

12.1.3. Protocolo SMTP

12.1.4. Protocolos SSL/TLS e StartTLS

12.1.5. Servidores de *e-mail*

12.1.6. Configuração do Gmail para envio de mensagens a partir do Servidor Linux

12.1.7. Protocolo ICMP

12.2. Programação

12.2.1. Variáveis

12.2.2. Comentários como forma de documentação

- 12.2.3. Importação de bibliotecas
- 12.2.4. Utilização de métodos de bibliotecas
- 12.2.5. Entrada e saída de dados
- 12.2.6. Envio de *e-mails*
- 12.2.7. Manipulação de *strings*
- 12.3. Sistemas Operacionais (Linux)
 - 12.3.1. Envio de e-mails
 - 12.3.2. Envio de mensagens instantâneas
 - 12.3.3. Agendamento de procedimentos
 - 12.3.4. Comandos do S. O.
 - 12.3.5. Procedimentos em *Shell Script*
 - 12.3.6. Configuração de aplicativos



Tarefas

Abaixo, propomos problemas que buscam reproduzir contextos do cotidiano de um profissional que tenha de resolver problemas relacionados a redes de computadores. Cada problema é dividido em três partes: Apresentação do contexto, discussão e resolução. Utilize os conhecimentos acumulados até agora para auxiliar nas resoluções.

É **importante** que as resoluções ocorram na sequência em que os problemas são dispostos.

Problema 1

Obs.: Para resolução deste problema, utilize as máquinas virtuais: Servidor1 e Servidor3.

1.1 Apresentação do contexto (Âncora):

Você pretende verificar o percentual de ocupação dos diretórios / (barra) dos servidores 1 e 3, diariamente, às 23 horas. Deve ser gerado um arquivo de log, cujo conteúdo deverá ser enviado ao seu *e-mail*. Caso a execução de alguma verificação resulte em erro, deve ser enviada uma mensagem de alerta no seu celular.

1.2 Discussão (Questões Motrizes): (Caso julgue necessário, pode tomar notas sobre esta discussão para construir sua solução.)

- 1.2.1 Como aferir a ocupação em cada servidor?
- 1.2.2 Como será gerado o arquivo de log?
- 1.2.3 Como será enviado ao *e-mail*?
- 1.2.4 Qual o mecanismo para enviar a mensagem de erro ao celular?

1.3 Resolução:

1.3.1 Descreva qual(is) o(s) provável(is) problema(s) que você identificou quanto à verificação das partições e envio de mensagens.

1.3.2 Descreva detalhadamente como será construída a solução para o problema. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3.3 Conforme a descrição feita no Item 1.3.2, converta a solução em um programa. Este programa deverá atender aos seguintes requisitos:

- Executar (“rodar”);
- Resolver os problemas de acordo com o contexto proposto;
- Estar em conformidade com a descrição da construção da solução para o problema;
- A codificação deve ser bem estruturada e eficiente (sem excessos);
- Deve conter comentários que instruem sobre a construção do programa;

Problema 2 – Reflexão e observação

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Você precisa sincronizar, a cada 30 minutos, todo o diretório /home/ do Servidor1, com um diretório no Servidor3. Os logs gerados devem ser armazenados no Servidor1. Caso algo aconteça de errado durante o processo, você precisa saber imediatamente, via mensagem em seu celular, enquanto que o seu chefe precisa ser notificado por *e-mail*.

Situação B: Os *backups* do diretório /home/ do Servidor1 devem ser executados com compactação e empacotamento de dados e enviados todos os dias, às 20 horas para o Servidor3. Cada processo diário deverá gerar um arquivo de log, cujo nome deverá conter a data da criação (ano-mês-dia), semelhante ao nome gerado para o arquivo de *backup*. Exemplo: 2017-11-28_backupgeral.tar; e 2017-11-28_backupgeral.log. O arquivo de log deverá conter a data e a hora do início do *backup*, a lista dos arquivos copiados, o status da criação do arquivo .tar e sua cópia e, ao final, a data e hora do término do processo.

Situação C: Alguns funcionários antigos, do setor de tecnologia da informação, deixaram a empresa antes de você entrar. Fazendo uma varredura nos servidores, você verificou que estes ex-funcionários ainda possuem usuários e permissões de acesso. Verificou também que os usuários têm permissão de acesso remoto via SSH. Seu chefe solicitou que seja feita uma lista contendo os nomes dos funcionários e que seja enviada para ele, na próxima segunda-feira, via *e-mail*.

Resolução:

2.1 Qual das situações (A, B ou C) apresenta um problema similar ao problema 1 e aos recursos trabalhados no problema 1? Justifique.

2.2 Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Abraços e bons estudos.

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com

APÊNDICE D – Avaliações Modulares

Nas páginas que se seguem (da p. 146 à p. 149) são apresentados as avaliações modulares 1 e 2.



CURSO DE PROGRAMAÇÃO
PARA
ADMINISTRAÇÃO DE
REDES DE COMPUTADORES
PROF. SILMAR ANTONIO



INSTITUTO FEDERAL
AMAZONAS



Mestrado em
Ensino Tecnológico

Avaliação Módulo 1

Nome: _____



Descrição Geral

Caro estudante,

Nesta avaliação colocaremos em prática os estudos realizados durante o módulo 1. O objetivo desta avaliação é identificar em que nível nossas práticas e assuntos abordados foram assimilados.

Leia com atenção e resolva as tarefas propostas com calma e da forma mais completa possível.

É permitido que você acesse quaisquer dos seus materiais, reveja os vídeos (com fones de ouvidos) e acesse a internet.

Não é permitida a interação entre estudantes, mesmo que de forma virtual.

IMPORTANTE: Somente serão consideradas as avaliações dos estudantes que estiveram presentes em sala de aula.

Desejo a você uma boa avaliação do módulo 1.



Resolução de Problemas

Problema 01

Apresentação do contexto:

Por desconfiança de problemas relacionados a desempenho, reclamados por alguns setores da empresa, você decidiu monitorar a utilização das CPUs do Servidor1 e do Servidor2 via SNMP. Este monitoramento deverá ser colocado em um relatório para posterior análise. Entretanto, como você desconfia que o Servidor1 e o Servidor2 podem ter problemas mais sérios, decidiu enviar o relatório contendo os levantamentos sobre os dois servidores, para o Servidor3. Com isto, você pretende analisar o documento gerado, a partir do Servidor3.

Para o monitoramento das CPUs, considere a seguinte OID: “.1.3.6.1.4.1.2021.10.1.3.1”.

1.1 Descreva qual(is) o(s) provável(is) problema(s) que você identificou quanto ao desempenho dos servidores.

1.2 Descreva detalhadamente como você construirá o monitoramento do Servidor1 e do Servidor2, a

construção do relatório e o envio para um diretório compartilhado no Servidor3. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajude a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3 Construa um programa em Python que possa automatizar as tarefas de monitoramento, construção do relatório e o envio para o Servidor3.

Problema 2: Reflexão e observação:

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Houve uma queda de energia durante a madrugada que durou mais tempo que os no-breaks suportariam. Todo o Data Center ficou sem energia. Pela manhã, quando a energia elétrica foi restabelecida, um dos servidores não religou.

Situação B: Em dado momento do dia, os usuários reclamam de lentidão no Servidor3. Você desconfia que sejam problemas relacionados à memória RAM e gostaria de gerar um relatório de utilização de memória que seria acionado sempre que achar necessário.

Situação C: Os *backups* da empresa em que você trabalha são armazenados todas as noites em fitas. Após um problema no HD do servidor Web, houve a necessidade da recuperação do último *backup*. Ao tentar recuperar os dados, você verificou que a mídia estava vazia. É necessário enviar um *e-mail* ao seu chefe comunicando este problema.

2.1) Qual das situações (A, B ou C) apresenta maior similaridade um problema similar ao problema 1 e aos recursos trabalhados no problema 1? Justifique.

2.2) Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Desejo a você uma ótima avaliação!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com.



CURSO DE PROGRAMAÇÃO
PARA
ADMINISTRAÇÃO DE
REDES DE COMPUTADORES
PROF. SILMAR ANTONIO



INSTITUTO FEDERAL
AMAZONAS



Mestrado em
Ensino Tecnológico

Avaliação Módulo 2

Nome: _____



Descrição Geral

Caro estudante,

Nesta avaliação colocaremos em prática os estudos realizados durante o módulo 2. O objetivo desta avaliação é identificar em que nível nossas práticas e assuntos abordados foram assimilados.

Leia com atenção e resolva as tarefas propostas com calma e da forma mais completa possível.

É permitido que você acesse quaisquer dos seus materiais, reveja os vídeos (com fones de ouvidos) e acesse a internet.

Não é permitida a interação entre estudantes, mesmo que de forma virtual.

IMPORTANTE: Somente serão consideradas as avaliações dos estudantes que estiveram presentes em sala de aula.

Desejo a você uma boa avaliação do módulo 1.



Resolução de Problemas

Problema 01

Apresentação do contexto:

Os usuários da empresa em que você trabalha utilizam o diretório `/home/user1/documentos/` para armazenar documentos confidenciais da empresa. Desta forma, há a necessidade da execução de um *backup* diário, às 23 horas, com compactação de arquivos. Também é necessário enviar este *backup* para o Servidor3. Caso a montagem no Servidor3 e o *backup* ocorram com sucesso, deve ser enviada uma mensagem para seu *e-mail*. Caso haja problemas na montagem no Servidor3 ou no *backup*, além da mensagem por *e-mail*, é necessário que seja enviado um alerta via Telegram para seu celular.

1.1 Descreva qual(is) o(s) provável(is) problema(s) que você identificou quanto à necessidade de *backup* apresentado no contexto.

1.2 Descreva detalhadamente como você construirá a estrutura de *backup*, envio para o Servidor3, verificações e construção das mensagens por *e-mail* e Telegram. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajude a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3 Conforme a descrição feita no Item 1.2, converta a solução em um programa. Este programa deverá atender aos seguintes requisitos:

- Executar (“rodar”);
- Resolver os problemas de acordo com o contexto proposto;
- Estar em conformidade com a descrição da construção da solução para o problema;
- A codificação deve ser bem estruturada e eficiente (sem excessos);
- Deve conter comentários que instruem sobre a construção do programa;

Problema 2: Reflexão e observação:

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Seu chefe ligou reclamando que não está conseguindo acessar o site www.cursosparaexecutivos.com, que acessava normalmente até então. Informou que consegue acessar outros sites. Pediu então que você o comunique via *e-mail* ou Telegram sobre a origem e/ou solução do problema.

Situação B: Há a necessidade de sincronizar o diretório `/home/user1/documentos/` com o `/backup/`, ambos do Servidor1 e, em seguida, empacotar e compactar o conteúdo do diretório `/backup/` e enviar para o Servidor3. O envio deve ser monitorado onde, não ocorrendo sem erros, deve ser enviada uma mensagem por *e-mail*. Caso existam erros, a mensagem de erro deve ser enviada por *e-mail* e ao seu celular.

Situação C: Você entrou recentemente na empresa e imediatamente verificou que não são feitos *backups* dos servidores. Preocupado com a situação, decidiu iniciar pelas configurações (/etc), do Servidor1. Os *backups* devem ser feitos todos os dias, de madrugada, compactados, no Servidor3. Também é necessário verificar todas as manhãs o resultado do *backup* executado na madrugada: se o *backup* foi executado; em que horário começou e em que horário foi finalizado; além da verificação de cada um dos procedimentos, se foram executados com sucesso.

2.1) Qual das situações (A, B ou C) apresenta maior similaridade ao problema 1 e aos recursos nele trabalhados? Justifique.

2.2) Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Desejo a você uma ótima avaliação!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com.

APÊNDICE E – TCLE e Questionário SEE

TCLE – Termo de Consentimento Livre e Esclarecido

Este curso faz parte de uma pesquisa para obtenção do título de mestre em educação tecnológica, pelo Instituto Federal do Amazonas – IFAM, intitulado “Programação para Redes de Computadores: Uma proposta de ensino híbrido”.

COORDENAÇÃO: Professor Silmar Antonio Buchner de Oliveira - IFRO

ORIENTAÇÃO: Professora Dra. Andréa Pereira Mendonça – IFAM

1. NATUREZA DA PESQUISA: Você está sendo convidado a participar desta pesquisa que tem como objetivo melhorar o ensino de redes nos cursos de graduação na área de Informática, associando a resolução de problemas de Redes de Computadores com recursos de programação, em um formato de ensino híbrido de sala de aula invertida, em que o estudante estuda a parte teórica fora da sala de aula, em um tempo que antecede um encontro presencial, em sala, com presença de um professor, no qual serão dirimidas dúvidas e exercitada a prática, com foco em problemas da área de Redes de Computadores.

2. APLICAÇÃO DA PESQUISA: A presente pesquisa será realizada sob os auspícios do Instituto Federal de Educação, Ciência e Tecnologia de Rondônia – IFRO – Campus Zona Norte, envolvendo estudantes que estão cursando nível superior, tanto do IFRO quanto de outras instituições. As atividades serão desenvolvidas no formato de um Curso de Extensão, com um total de 52 horas.

3. PARTICIPANTES DA PESQUISA: Participarão desta pesquisa estudantes que estejam matriculados e cursando qualquer curso de nível superior na área de Informática e que já tenham cursado, pelo menos, uma disciplina de programação. Os participantes foram selecionados através do cálculo da média aritmética constante em seus históricos, submetidos pelos estudantes que se candidataram ao curso, dos quais foram classificados 30 (trinta) participantes, graduandos do IFRO e de outras instituições.

4. ENVOLVIMENTO NA PESQUISA: Ao participar deste estudo você será convidado a preencher um questionário sócio acadêmico, juntamente com outros estudantes que aceitaram participar da pesquisa. Ao longo da pesquisa, serão desenvolvidas atividades no contexto de ensino e aprendizagem, envolvendo resolução de problemas e construção de programas de

computadores. Estas atividades ficarão de posse do pesquisador.

5. **SOBRE O QUESTIONÁRIO:** Serão solicitadas algumas informações básicas e perguntas de múltipla escolha ou escolha simples sobre suas condições sociais e acadêmicas. Em momento algum, os participantes desta pesquisa serão identificados ou terão os dados publicados nominalmente.

6. **RISCOS E DESCONFORTO:** A participação nesta pesquisa não traz complicações legais de nenhuma ordem. Nenhum dos procedimentos utilizados oferece riscos à sua dignidade ou a sua saúde.

7. **CONFIDENCIALIDADE:** Todas as informações coletadas nesta investigação são estritamente confidenciais. Acima de tudo, interessam os dados coletivos e não aspectos particulares de cada estudante. Os dados serão analisados apenas para fins acadêmicos. O objetivo da pesquisa é o de melhorar o ensino de redes nos cursos de graduação, na área de Informática. Sua identidade não será divulgada ou publicada, exceto se devidamente autorizado.

8. **BENEFÍCIOS:** Ao participar desta pesquisa, você terá oportunidade de desenvolver habilidades de programação para a resolução de problemas do domínio de Redes de Computadores, em um modelo de ensino híbrido, que envolvem estudos autônomos e presenciais. Serão trabalhadas habilidades para i) identificação de problemas relacionados a Redes de Computadores; ii) resolução de problemas com utilização de recursos de programação para automatização das soluções; iii) associação das soluções encontradas com outras soluções de problemas de natureza semelhante. Os aspectos desta pesquisa se concentram em melhorar a formação do estudante de forma a capacitá-lo a extrair melhores benefícios das tecnologias atuais e, conseqüentemente, torná-lo um profissional mais bem preparado para o mercado de trabalho, capaz de tomar decisões autonomamente frente a problemas que envolvam Redes de Computadores, com o emprego de recursos de programação.

9. **PAGAMENTO:** Você não terá nenhum tipo de despesa por participar deste estudo, bem como não receberá nenhum tipo de pagamento por sua participação.

Após estes esclarecimentos, solicitamos sua contribuição com a referida pesquisa. Para tanto, preencha os itens que se seguem:

CONSENTIMENTO LIVRE E ESCLARECIDO

Tendo em vista os itens acima apresentados, eu, de forma livre e esclarecida, concordo com os termos supracitados e confirmo minha participação na pesquisa.

Nome do estudante (Por extenso)

Porto Velho, 18 de novembro de 2017

Silmar Antonio Buchner de Oliveira

Porto Velho, 18 de novembro de 2017

Agradecemos a sua colaboração e colocamo-nos à disposição para esclarecimentos adicionais. O pesquisador responsável por esta pesquisa é o Prof. Silmar Antonio Buchner de Oliveira, mestrando do MPET do Instituto Federal do Amazonas, sob orientação da Professora Dra. Andréa Pereira Mendonça, docente do Mestrado Profissional em Ensino Tecnológico do Instituto Federal do Amazonas IFAM.



CURSO DE PROGRAMAÇÃO
PARA
ADMINISTRAÇÃO DE
REDES DE COMPUTADORES
PROF. SILMAR ANTONIO

Curso de Programação para Administração de Redes de Computadores

Questionário Sócio-econômico-educativo

Este questionário foi desenvolvido exclusivamente para fins acadêmicos, fazendo parte de uma pesquisa intitulada “Programação para Redes de Computadores: Uma proposta de ensino híbrido”. Os dados são confidenciais e serão utilizados para fins estatísticos.

Os questionários preenchidos, bem como seus dados não serão compartilhados com terceiros. Não serão divulgados quaisquer dados que possam identificar sua identidade. O Acesso a estas informações, na íntegra, será de exclusividade e de responsabilidade do professor Silmar Antonio Buchner de Oliveira.

Nome: _____

Bairro onde mora: _____

Sexo: Masculino () Feminino ()

Idade: _____

1. Em que tipo de instituição você concluiu seu ensino médio?

() Pública () Privada

2. Em qual curso superior, na área de Informática, você está matriculado?

R.: _____

Instituição:

() IFRO Calama () IFRO Zona Norte () Uniron () UNIR () Unopar
() Porto () Unigran

3. Atualmente você estuda em qual período do seu curso?

R.: _____

4. Semanalmente, qual o tempo aproximado que você dedica aos estudos, fora da sala de aula?

() 30 minutos ou menos. () 1 hora () 2 horas () 3 horas () 4 horas ou mais
() Costumo estudar somente em períodos de provas.

5. Como você acessa computadores fora da instituição de ensino? (pode marcar mais de um resultado)

() Em casa. () No trabalho. () Lan House.

- () Biblioteca pública. () Acesso na casa de uma amigo ou parente.
 () Não tenho acesso a computadores fora do ambiente acadêmico.

6. Qual a situação do computador ao que você tem acesso, fora do ambiente acadêmico, quanto à utilização:

- () É meu e somente eu uso a qualquer momento.
 () É meu mas tenho de compartilhar o tempo de uso com outras pessoas.
 () Não é meu mas posso utilizar quando quiser.
 () Não é meu e tenho horários ou períodos determinados para utilizar.
 () Outras situações. Descreva: _____
 () Não tenho acesso a computadores fora do ambiente acadêmico.

7. Descreva a memória RAM seu computador:

- () Até 2GB () Até 4GB () Até 8GB () Mais de 8GB () Desconheço

8. Como você acessa Internet fora da instituição de ensino? (pode marcar mais de um resultado)

- () Em casa, Internet banda larga via operadora (Oi, Claro, Net, etc.)
 () Em casa, via Rádio.
 () No trabalho.
 () Internet via celular.
 () Outro _____
 () Não possuo acesso à Internet fora da instituição de ensino.

9. Qual a velocidade da Internet que você acessa? (considere a que você utilizará neste curso)

- () Entre 128 Kbps e 512 Kbps
 () Entre 512 Kbps e 1 Gbps
 () Entre 1 Gbps e 2 Gbps
 () Entre 5 Gbps e 10 Gbps
 () Maior que 10 Gbps
 () Desconheço esta informação.

10. Em uma escala de 1 a 5, qual o grau de identificação com os tipos disciplinas abaixo?

Programação

- 1() Não me identifico
 2() Me identifico pouco
 3() Me identifico razoavelmente
 4() Me identifico bastante
 5() Me identifico completamente.

Redes de Computadores

- 1() Não me identifico
 2() Me identifico pouco
 3() Me identifico razoavelmente
 4() Me identifico bastante
 5() Me identifico completamente.

11. Quantas disciplinas de Redes de Computadores você já cursou? (Considere, por exemplo: Redes de computadores; Introdução a Redes de Computadores; Redes convergentes; etc.)

- () 0 () 1 () 2 () 3 () 4 ou mais

12. Quantas disciplinas de programação você já cursou? (Considere, por exemplo: Algoritmos; Algoritmos e linguagens de programação; Lógica de programação; Linguagens de programação I, Programação Orientada a Objetos, etc.)

- () 0 () 1 () 2 () 3 () 4 ou mais

13. Em qual(is) linguagem(s) de programação você já programou? (pode marcar mais de um resultado)

- () Portugol (algoritmos/Lógica) () Pascal. () C.
() Java. () Python. () *Shell scripts* (Linux).
() Outra(s).

14. Onde você aprendeu a programar?

- () Sozinho ou através de cursos, antes da faculdade.
() Na faculdade.

15. Quanto tempo você trabalha por dia?

- () 4 horas. () 6 horas.
() 8 horas. () Não trabalho.

16. Você trabalha ou já trabalhou com programação?

- () Sim. Quanto tempo? _____
() Não.

17. Você trabalha ou já trabalhou com Redes de Computadores?

- () Sim. Quanto tempo? _____
() Não.

18. Você já resolveu problemas de Redes de Computadores utilizando recursos de programação?

- () Não.
() Sim. Descreva em que situação:

Caso ache necessário, escreva alguma observação sobre algum(s) item(ns) do questionário ou quaisquer outros assuntos pertinentes ao curso.

Obrigado por colaborar com esta pesquisa!

Professor Silmar Antonio

APÊNDICE F – Avaliação Diagnóstica



CURSO DE PROGRAMAÇÃO
PARA
ADMINISTRAÇÃO DE
REDES DE COMPUTADORES
PROF. SILMAR ANTONIO

Curso de Programação para Administração de Redes de Computadores

Avaliação diagnóstica – 18/11/2017

Nome: _____



Descrição Geral

Caro estudante,

Esta avaliação diagnóstica tem por objetivo identificar os conhecimentos sobre Redes de Computadores adquiridos antes deste curso e com base nos resultados, estudarmos melhores formas para o desenvolvimento deste curso.

Os resultados serão mantidos sob sigilo e serão utilizados somente para análise de dados, sendo preservados totalmente a sua identidade.

O resultado deste diagnóstico não contará como nota para formar suas notas ao longo deste curso. Ainda assim, peço que resolva o que for proposto, com dedicação.

Leia com atenção e resolva as tarefas propostas com calma e da forma mais completa possível.



Parte 1 – Conceitual

1) O comando PING está presente na maioria dos sistemas operacionais e auxilia no diagnóstico de problemas em Redes de Computadores e em dispositivos que a formam. Assim, assinale corretamente qual dos protocolos abaixo está associado ao comando PING?

- | | |
|-------------|-------------------------------|
| a. () SNMP | d. () DHCP |
| b. () SSH | e. () TLS |
| c. () ICMP | f. () Desconheço/Não lembro. |

2) Além do PING, há outras formas de monitorar os componentes de uma rede. Um deles utiliza uma espécie de dicionário de dados, chamada MIB (*Management Information Base*) e OID (*Object Identifier*), que armazenam informações referentes ao equipamento a ser monitorado. A qual forma de monitoramento estamos nos referindo?

- | | |
|-------------|-------------------------------|
| a. () IGMP | d. () DNS |
| b. () DHCP | e. () SNMP |
| c. () ICMP | f. () Desconheço/Não lembro. |

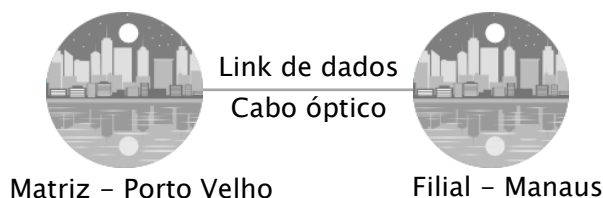
3) Em uma rede de computadores, um recurso muito útil é o que possibilita que arquivos e diretórios sejam compartilhados entre diferentes *hosts* (Um *host* é qualquer equipamento em rede, que possua um endereço IP). Assinale **DOIS** protocolos abaixo que possibilitam este tipo de compartilhamento.

- a. ☐ SMB
 b. ☐ DNS
 c. ☐ NFS
 d. ☐ DNS
 e. ☐ DHCP
 f. ☐ Desconheço/Não lembro.

4) LANs, MANs, WANs, PANs e WLans são tipos de redes classificadas conforme sua distância entre os pontos conectados. Neste contexto, observe as figuras e classifique-as conforme a distância entre os pontos conectados, assinalando a alternativa correta.

4.1)

- a. ☐ LAN
 b. ☐ MAN
 c. ☐ WAN
 d. ☐ PAN
 f. ☐ Desconheço/Não lembro.



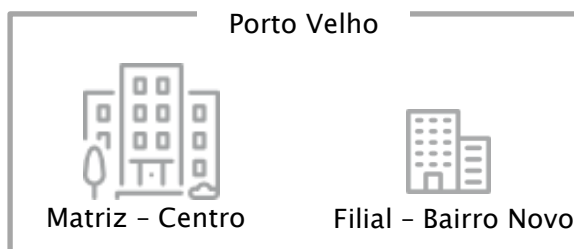
4.2)

- a. ☐ LAN
 b. ☐ MAN
 c. ☐ WAN
 d. ☐ PAN
 f. ☐ Desconheço/Não lembro.



4.3)

- a. ☐ LAN
 b. ☐ MAN
 c. ☐ WAN
 d. ☐ PAN
 f. ☐ Desconheço/Não lembro.



5) Relacione as colunas, de acordo com as características de cada protocolo:

- | | | |
|---------|--------------------------|--|
| (1) IP | ☐ | Oferece uma conexão segura, buscando garantir a entrega dos segmentos na sequência em que foram enviados pela origem. Utiliza portas para identificação de seus serviços. |
| (2) TCP | | |
| (3) UDP | | |
| (4) MAC | ☐ | não é orientado à conexão. Pode ser atribuído manualmente a um host. |
| | ☐ | Presente em dispositivos de rede, identifica-os através de uma numeração fixa. Caso o dispositivo seja posto em outra máquina, ainda assim, utilizará a mesma identificação. |
| | ☐ | Um protocolo muito utilizado para conexões de streaming, como música <i>online</i> , por exemplo. . Utiliza portas para identificação dos seus serviços. |

6) Quanto às pilhas de protocolos TCP/IP e OSI/ISO, considere:

1 – TCP/IP

2 – OSI/ISO

Numere as questões abaixo, conforme a correspondência apresentada anteriormente (pode haver mais de uma marcação para cada item correspondente).

() Possui 7 camadas.

() Viabilizou a Internet.

() É um modelo de referência sem aplicação prática difundida.

() Entre suas camadas, estão a de apresentação e de Sessão.



Parte 2 – Resolução de Problemas

Obs.: Para resolver os itens dos problemas propostos na parte 2 que não couberem no espaço reservado para cada resposta, você pode utilizar o verso da avaliação.

Problema 01

A) Apresentação do contexto:

A empresa onde você trabalha possui 25 (vinte e cinco) Pontos de Acesso sem fio (Rádios *Wireless*), que se comunicam com um rádio central, via IP. Estes equipamentos são responsáveis por distribuir internet em praças e espaços culturais da cidade. Para minimizar possíveis problemas com energia elétrica que possam indisponibilizar a comunicação nos locais atendidos pelo serviço, cada rádio wireless é conectado a um nobreak. Estes nobreaks são capazes de mantê-los em funcionamento por até 1 (uma) hora, em caso de falta de energia elétrica. Ocorre que muitos usuários estão reclamando nas redes sociais que está havendo muitos problemas de indisponibilidade. O primeiro passo é identificar a frequência de quedas para que os problemas possam ser investigados.

Desta forma, seu chefe solicitou um relatório onde constem os equipamentos que estiveram fora do ar, de hora em hora, a cada dia. Este relatório deve ser enviado a você e a ele, por *e-mail*, ao final de cada dia, de forma que vocês possam lê-lo a cada manhã. Se, em algum momento, mais de 5 (cinco) equipamentos estiverem sem comunicação com a central, uma mensagem instantânea ou um *e-mail* deve ser enviado imediatamente para você e seu chefe.

1.1 Descreva qual(is) o(s) provável(is) problema(s) que você identificou quanto às providências a serem tomadas.

1.2 Descreva detalhadamente como você construirá uma forma de geração do relatório automático para a solução do(s) problema(s). Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajude a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

1.3 Construa um programa que possa automatizar este relatório, conforme a solicitação do seu chefe. (em qualquer linguagem de programação que você domine, inclusive português).

Problema 2: Reflexão e observação:

Leia atentamente as três situações apresentadas a seguir e após a leitura faça o que se pede.

Situação A: Os *backups* da empresa em que você trabalha são armazenados todas as noites em fitas. Após um problema no HD do servidor Web, houve a necessidade da recuperação do último *backup*. Ao tentar recuperar os dados, você verificou que a mídia estava vazia. É necessário enviar um *e-mail* ao seu chefe comunicando este problema.

Situação B: Alguns funcionários antigos, do setor de tecnologia da informação, deixaram a empresa antes de você entrar. Fazendo uma varredura nos servidores, você verificou que estes ex-funcionários ainda possuem usuários e permissões de acesso. Verificou também que os usuários têm permissão de acesso remoto via SSH. Seu chefe solicitou que seja feita uma lista contendo os nomes dos funcionários que estão nesta situação e que seja enviada para ele, na próxima segunda-feira, via *e-mail*.

Situação C: Os usuários da internet estão reclamando que em vários momentos durante o dia está havendo lentidão no acesso aos sites. Consultando a operadora de Internet, eles solicitaram um monitoramento que demonstre esta lentidão, a cada 10 minutos, com testes feitos a, pelo menos 3 sites diferentes. Os testes devem ser gravados em relatórios diários e enviados à operadora, com cópia para você.

2.1) Qual das situações (A, B ou C) apresenta um problema similar ao problema 1 e aos recursos trabalhados no problema 1? Justifique.

2.2) Descreva DETALHADAMENTE como você automatizaria uma solução para a situação que você identificou como similar ao problema 1. Este detalhamento deverá conter o passo-a-passo sobre a solução e o máximo de comentários que ajudem a complementar a descrição da solução. Tente enriquecer ao máximo sua resposta, o mais detalhadamente possível.

Desejo a você uma ótima avaliação diagnóstica!

Obs.: Todas as figuras empregadas no texto foram cedidas por www.flaticon.com.

APÊNDICE G – Artigos Publicados



Sala de Aula Invertida - Uma Experiência no Ensino-Aprendizagem de Programação para Administração de Redes de Computadores

Silmar Antonio Buchner de Oliveira – MPET/IFAM – silmarantonio@gmail.com
 Andréa Pereira Mendonça – MPET/IFAM – andrea.mendonca@ifam.edu.br

Resumo: Neste artigo relatamos o planejamento e execução de uma proposta de ensino-aprendizagem de programação para administração de redes de computadores no modelo de sala de aula invertida, destinada a estudantes de Cursos Superiores de Tecnologia. A proposta pauta-se na resolução de problemas práticos, envolvendo automatização de soluções para monitoramento de equipamentos de redes e ações de usuários, análise de logs, sincronismos de dados e emissão automática de mensagens de alertas, implementados na linguagem de programação Python. Os resultados obtidos evidenciaram progresso dos estudantes, mas também a necessidade de ajustes na produção de conteúdos de nivelamento em conceitos básicos de redes e de programação. Toda a experiência com o modelo e os recursos utilizados serão compilados em um guia para utilização por outros professores.

Palavras chave: Ensino híbrido; Sala de aula invertida; Redes de computadores; Programação.

Flipped Classroom – An Experience in the Teaching-Learning of Programming for Computer Network Administration

Abstract: In this paper we present a teaching-learning proposal, following the flipped classroom model, of a course on programming for network administration aimed at students of undergraduate technological courses. This proposal is based on practical problem solving by using Python programming language and addressing automatization tasks such as for hardware and user actions monitoring, log analysis, data synchronization and automatic notifications. The students' progress was measured and revealed one of the positive outcomes of this course altogether with the need of levelling courses on fundamentals of computer networks and on programming. The whole experience with this teaching-learning proposal as well as the resources adopted are being synthesized in a guide that will be freely shared with other instructors.

Keywords: Blended-learning; Flipped classroom; Computer Networks; Programming.

1. Introdução

Considerado como uma das grandes tendências do século XXI, o ensino híbrido, do inglês *blended learning*, é uma abordagem que combina ensino presencial, em sala de aula, e *online*, no qual o estudante dispõe de tecnologias e algum controle sobre o tempo, lugar, modo e/ou ritmo do estudo (CHRISTENSEN; HORN; STAKER, 2013).

Há diferentes modelos de ensino híbrido relatados na literatura (HORN; STAKER, 2015), sendo um deles denominado sala de aula invertida (*flipped classroom*). Neste modelo, inverte-se a forma tradicional de ensino, isto é, o estudante passa a assistir a aula ou ter acesso ao conteúdo fora de sala, por meio de recursos como videoaulas, disponibilizadas previamente pelo professor. As tarefas práticas, tais como resolução de exercícios, que o estudante tradicionalmente executava em casa, passam a ser realizadas em sala de aula, com orientação do professor (BERGMANN; SAMS, 2012).

A implementação do modelo de sala de aula invertida propicia aos estudantes o desenvolvimento de autonomia e responsabilidade por sua própria aprendizagem, e eleva o papel do professor na medida em que este passa a auxiliar os estudantes na



aprendizagem mais profunda, orientando-os na resolução de problemas e esclarecimentos de dúvidas.

Estes benefícios nos motivaram a desenvolver uma proposta de ensino-aprendizagem de programação para administração de redes de computadores, destinada a estudantes de Cursos Superiores de Tecnologia (CST). A proposta foi implementada no formato de sala de aula invertida, no qual, fora de sala, os estudantes tinham acesso ao conteúdo por meio do Google Classroom e, em sala de aula, o tempo era dedicado à resolução de problemas práticos envolvendo monitoramento de equipamentos de redes e ações de usuários, análise de *logs*, sincronismos de dados e emissão automática de mensagens de alertas, sendo a programação desenvolvida na linguagem Python.

Os fundamentos desta proposta de ensino-aprendizagem, o percurso metodológico de sua elaboração, assim como os resultados obtidos com a sua implementação são apresentados nas próximas seções.

2. Sala de Aula Invertida e o Ensino de Redes de Computadores em CST

Nas próximas seções contextualizamos a dinâmica de implementação da sala de aula invertida, assim como detalhamos os desafios do ensino de programação para redes de computadores, considerando estudantes de Cursos Superiores de Tecnologia (CST), público alvo deste trabalho.

2.1 Sala de aula invertida

O modelo de sala de aula invertida foi idealizado por Bergmann e Sams (2012) no ano de 2006, quando lecionavam química em uma escola de ensino médio, relativamente rural, no interior do estado do Colorado (EUA). Por motivos particulares, não era incomum que os estudantes faltassem às aulas, o que acarretava retrabalho dos professores e perdas de conteúdos pelos estudantes. No intuito de sanar o problema, os dois professores resolveram gravar suas aulas em vídeos e postá-las *online*. Isto resolveu o problema dos faltosos e, colateralmente, beneficiou os outros estudantes, pois ainda que não faltassem, aderiram aos vídeos. Os estudantes relataram que isto melhorou seu aprendizado, pois poderiam voltar os vídeos aos pontos em que não haviam compreendido e rever o conceito sempre que necessário. Com isto, os professores resolveram gravar todas as suas aulas em vídeo (ao que chamaram de pré-aula) e propuseram que os estudantes os assistissem antes das aulas presenciais, que seriam destinadas à resolução de problemas envolvendo a teoria estudada. Com o sucesso da experiência, os professores resolveram aprimorar o modelo de ensino de sala de aula invertida e divulgá-lo para que outros professores pudessem aplicá-lo.

Segundo o modelo, a pré-aula concentra-se em vídeos construídos pelos professores ou cuidadosamente buscados *online*, de acordo com os conteúdos abordados. Durante a produção das aulas autorais é proposto que os professores busquem deixar seus vídeos intuitivos e atraentes, com uso de textos, anotações, inflexões de voz, etc. Os vídeos devem ser objetivos, não devendo ultrapassar quinze minutos de duração, onde cada tópico deve corresponder a um vídeo. (BERGMANN; SAMS, 2012).

É muito importante que os professores se preocupem com a adoção de tecnologias em benefício do ensino-aprendizagem, visto que esta geração de estudantes normalmente é bastante ambientada com vídeos, mensagens instantâneas e ambientes virtuais para diversos fins. Assim, diversos recursos podem ser utilizados na sala de aula invertida a fim de aproveitar esta afinidade, como Youtube, ambientes de aprendizagem como Moodle, Google Classroom, simuladores virtuais de máquinas e ambiente



computacionais como VirtualBox ou VMWare¹ e também comunicação via Whatsapp, Telegram e e-mails.

Ao contrário do que possam parecer, os recursos tecnológicos adotados antes da aula presencial não são o maior benefício da sala de aula invertida, mas sim, o tempo liberado para atividades práticas presenciais com maior qualidade e mais envolvimento dos estudantes em estratégias auxiliares como a aprendizagem baseada em projetos e/ou problemas. Além disso, com mais tempo livre, o professor pode auxiliar melhor a aplicação do conteúdo, tirar dúvidas e direcionar maior atenção aos estudantes com dificuldades nas aulas presenciais (BERGMANN; SAMS, 2012).

Tais características revelam-se benéficas ao contexto do ensino-aprendizagem de programação para redes de computadores, conforme descrevemos a seguir.

2.2 Ensino de redes de computadores em CST

Os Cursos Superiores de Tecnologia (CST) são uma modalidade de ensino superior com duração mais curta (entre 2 e 3 anos), se comparado a cursos de Bacharelado e, geralmente, são oferecidos pelos Institutos Federais de Educação, pertencentes à rede Federal de Educação Profissional, Científica e Tecnológica, criada em 2008. Há também a oferta de CST em universidades públicas e Instituições de Ensino Superior particulares (BRASIL, 2006; BRASIL, 2002).

O Catálogo Nacional de Cursos Superiores de Tecnologia organiza e orienta a oferta de cursos superiores de tecnologia, inspirado nas diretrizes curriculares nacionais e em sintonia com a dinâmica do setor produtivo e as expectativas da sociedade (SETEC/MEC). O Catálogo é organizado por eixos, sendo o eixo Informação e Comunicação aquele que integra os cursos relacionados à área de Computação/Informática, como CST em Análise e Desenvolvimento de Sistemas, CST em Sistemas para Internet, CST em Redes de Computadores, CST em Jogos digitais, entre outros (BRASIL, 2016).

Independente da denominação, os Cursos do eixo Informação e Comunicação possuem em seus currículos, pelo menos, uma disciplina de Redes de Computadores. De acordo com uma pesquisa *ad-hoc* que realizamos em sessenta (60) CST a partir de informações disponibilizadas nos sites de quarenta (40) Institutos Federais de Educação, observamos que as ementas são baseadas essencialmente em teorias sobre pilhas de protocolos TCP/IP e no modelo de referência OSI/ISO, não havendo menções a aplicações práticas nas disciplinas iniciais.

Em contrapartida a este cenário de formação, o mercado de trabalho requer dos egressos dos CST a resolução de problemas práticos e, no caso da área de redes de computadores, são comuns demandas voltadas para administração de redes de computadores, as quais envolvem automatização de procedimentos para monitoramento, controle e gerenciamento de serviços, como execução de *backups*, sincronismo de dados, verificação de consumo de recursos, envio de mensagens de alerta via e-mail, etc.

Tendo em vista a melhoria da formação destes estudantes, concebemos uma proposta de ensino-aprendizagem de programação para administração de redes de computadores, sendo possível implementá-la desde as disciplinas iniciais de redes de computadores. A concepção desta proposta encontra sustentação em diferentes autores, dentre eles, Kurose et al. (2002) que enfatizam que estudantes de redes de computadores que aprendem a codificar programas podem ter mais facilidade para internalizar e entender conceitos avançados, além de deixá-los com um conhecimento “comercializável”.

¹ VirtualBox e VMWare são *softwares* para virtualização de máquinas e sistemas operacionais.



Para que a proposta de ensino-aprendizagem oportunizasse aos alunos maior experiência “mão na massa”, adotamos o modelo de sala de aula invertida, combinando estudo autônomo, para apropriação dos conceitos, com estudo presencial, no qual o tempo de sala de aula era dedicado a resolução de problemas práticos, conforme metodologia descrita a seguir.

3 Metodologia

Nesta seção ressaltamos a metodologia de concepção da proposta de ensino-aprendizagem que demandou seis meses de trabalho, incluindo planejamento de conteúdos, organização dos estudos presenciais e autônomos, seleção de tecnologias e produção de videoaulas.

3.1 Organização do conteúdo

Os assuntos foram selecionados a partir de uma busca por livros nacionais e estrangeiros que envolvessem administração de redes de computadores. Dos trinta livros iniciais pesquisados, selecionamos dezesseis e, neste subconjunto, selecionamos os assuntos mais frequentes e identificamos também os tipos de problemas associados aos conteúdos, conforme ilustrado no Quadro 1.

Quadro 1 - Relação de assuntos versus ocorrência no conjunto de livros.

Assunto	Ocorrências
Monitoramento de Redes	9
Compartilhamento de dados	8
Envio de e-mails/relatório	8
Gerenciamento de sistemas de arquivos	8
Gerenciamento de usuários	8
Login (acesso de usuários)	7
Monitoramento de sistemas	7
Gerenciamento de Arquivos e diretórios	7
Backup	6
SSH/acesso remoto	6
Análise de logs (registros de atividades)	4
Compressão de dados	4
Expressões regulares	3
Manipulação de strings	3

Fonte: Autoria própria.

Organizamos os assuntos por temáticas e, para uma melhor estruturação, dividimos a proposta de ensino-aprendizagem em dois módulos: o módulo 1 para tratar sobre monitoramento de redes e de ações de usuários; e, o módulo 2 para tratar sobre preservação das informações e comunicação de eventos. Em cada módulo foram abordadas 3 temáticas, cada uma englobando um ou mais assuntos. O grau de complexidade dos assuntos, questões e problemas aumentava ao longo de cada módulo conforme evoluíam as temáticas, que, por sua vez, integravam conteúdos sobre redes de computadores, sistemas operacionais e programação, conforme pode ser observado no Quadro 2.

Seguindo os preceitos da sala de aula invertida, proposto por Bergmann e Sams (2012), cada temática foi tratada seguindo duas abordagens. A primeira chamamos de “estudo autônomo”, na qual o estudante, fora da sala de aula, apropria-se dos conceitos envolvidos; e a segunda, chamamos de “estudo presencial”, que ocorre com a presença do professor em um laboratório de informática, a fim de dirimir dúvidas e solucionar problemas práticos, articulando os conceitos aprendidos durante o estudo autônomo.



Quadro 2 – Módulos, temas, assuntos, conteúdos e problemas.

Módulo 1 - Monitoramento de redes de computadores		
Tema 01: Identificando a movimentação da vizinhança Assunto: Monitoramento de disponibilidade de equipamentos e serviços em rede. Conteúdos principais: <u>Redes:</u> Comando "ping"; Protocolo SNMP. <u>Programação:</u> Bibliotecas e métodos; variáveis; E/S de dados; <u>S.O. Linux:</u> Estruturas de arquivos e diretórios; Monitoramento de estados de componentes. Problemas propostos: Detecção de problemas de comunicação, através de monitoramentos via SNMP e "ping".	Tema 02: Cuidando do <i>Habitat</i> Assunto: Monitoramento de estado de discos e partições, em rede. Conteúdos principais: <u>Redes:</u> Protocolo NFS; Protocolo SNMP. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; <u>S.O. Linux:</u> Compartilhamento de dados; Estruturas de arquivos e diretórios; Problemas propostos: Monitoramentos remotos de discos e partições como prevenção de estouro de espaços de armazenamento.	Tema 03: De olho nas intenções Assunto: Monitoramento de ações de usuários. Conteúdos principais: <u>Redes:</u> Protocolo SSH, Acesso remoto; <u>Programação:</u> Bibliotecas e métodos; variáveis; E/S de dados; Arquivos; <i>Strings</i> e datas; Laços; Decisão. <u>S.O. Linux:</u> Editores de textos; Acesso remoto; Arquivos de configuração; Históricos de ações; Data e hora; Problemas propostos: Monitoramentos de ações de usuários através análise <i>logs</i> do sistema operacional.
Módulo 2 - Preservação de informações e comunicação de eventos		
Tema 04: Construir é difícil. Reconstruir pode ser impossível Assunto: Compactação de dados, <i>backup</i> em rede e registros de atividades (<i>logs</i>). Conteúdos principais: <u>Redes:</u> <i>Backup</i> ; Compartilhamento de arquivos; Protocolo NTP; Protocolo NFS. <u>Programação:</u> Bibliotecas e métodos; variáveis; E/S de dados; <i>Strings</i> e datas; Laços; Decisão. <u>S.O. Linux:</u> Empacotamento e compactação de dados; Agendamento de tarefas; Sincronismo de data e hora; Arquivos de <i>logs</i> . Problemas propostos: Agendamentos <i>backups</i> em rede, com verificação de etapas de execução e gravação de <i>logs</i> .	Tema 05: Um é pouco. Então tenha dois (ou mais!) Assunto: Sincronismo de dados em rede. Conteúdos principais: <u>Redes:</u> <i>Backup</i> ; Políticas de <i>backup</i> ; Sincronismo de dados remotos; Compartilhamento de arquivos; Protocolo SSH; Protocolos NFS. <u>Programação:</u> Bibliotecas e métodos; variáveis; E/S de dados; <i>Strings</i> e datas; Decisão. <u>S.O. Linux:</u> Sincronismo de dados; Agendamento de tarefas; Arquivos de <i>logs</i> . Problemas propostos: Agendamentos de sincronismo de dados em rede, com verificação de etapas de execução e gravação de <i>logs</i> .	Tema 06: Seja o primeiro a saber Assunto: Automação de mensagens de eventos e de alertas via <i>e-mail</i> e mensageria instantânea (Telegram). Conteúdos principais: <u>Redes:</u> Serviços de <i>e-mail</i> ; Protocolo SMTP; Protocolos SSL/TLS e StartTLS. <u>Programação:</u> Bibliotecas e métodos; variáveis; E/S de dados; <i>E-mails</i> ; <i>Strings</i> ; <u>S.O. Linux:</u> <i>E-mails</i> ; mensageria instantânea; Sincronismo de dados; Agendamento de tarefas. Problemas propostos: Automações de procedimentos com verificações de etapas de execuções e envio de alertas via <i>e-mail</i> e Telegram.

Fonte: Autoria própria.

3.2 Elaboração dos Roteiros de Aprendizagem para estudo Autônomo e Presencial

Os Roteiros de Aprendizagem (RA) funcionam como guias para que o estudante alcance os resultados pretendidos da aprendizagem em seus estudos autônomos e presenciais. Para cada temática foram criados dois RA, um para o estudo autônomo e outro para o estudo presencial, totalizando doze (12) RA. Os RA para estudo autônomo e presencial são semelhantes em sua estrutura geral, diferenciando-se na seção de tarefas. Nos estudos autônomos, as tarefas são compostas por questões envolvendo conceitos e também pequenas aplicações práticas em um laboratório virtual, criado especificamente para dar suporte à proposta de ensino-aprendizagem. Para apoiar os estudantes na resolução das tarefas são indicados, nos próprios RA, vídeos, tutoriais *online* e *links* de páginas *web*.

Nos RA para estudo presencial, as tarefas foram inspiradas em Bender (2014) e em Biggs e Tang (2011) e são constituídas da seguinte dinâmica: (i) *aquecimento*: um tempo é destinado para que os estudantes exponham suas dúvidas sobre o estudo autônomo, em uma discussão mediada pelo professor; (ii) *apresentação do contexto*: é apresentado um contexto envolvendo um problema real na área de redes de computadores; (iii) *discussão*: são apresentadas questões que instigam o estudante a compreender melhor o problema e encontrar meios para solucioná-lo; (iv) *resolução*: é



solicitado dos estudantes a implementação de um programa, em linguagem Python, para solucionar o problema proposto, executando-o no laboratório virtual.

3.3 Laboratório virtual

Presencialmente, no laboratório de informática o estudante encontrava um ambiente virtual de redes para testar seus programas. Contudo, para que pudesse cumprir com as tarefas do RA autônomo, foi necessária a reprodução do ambiente virtual em seus computadores pessoais.

Para auxiliar os estudantes, elaboramos um tutorial para os ajudar a reproduzir o laboratório virtual que continha 3 servidores Linux. Um estudo prévio foi feito para que este laboratório pudesse ser facilmente replicado pelo estudante em seu computador, de forma que fosse executado sem consumir muitos recursos como memória e processamento. Desta forma, optamos pelo Ubuntu Minimal por ser uma versão básica do Linux² que possibilitava facilmente a instalação de quaisquer funcionalidades posteriores.

Para facilitar a reprodução dos servidores, arquivos automatizados de configuração foram disponibilizados em uma máquina virtual “base” que o estudante baixaria e importaria, no seu computador pessoal, para a plataforma de virtualização Oracle VM Virtualbox³. A partir desta importação, o estudante executaria a clonagem da máquina base, obtendo os três servidores. Um destes servidores possuía interface gráfica e tinha instalado o ambiente Thonny⁴ para desenvolver suas soluções na linguagem Python.

3.4 Ambiente Virtual e Recursos Digitais para guiar o estudo Autônomo e Presencial

O Google Classroom (no Brasil, Google Sala de Aula) é uma plataforma *web* da Google que reúne um conjunto de serviços (*e-mail*, *drive*, gerenciador de tarefas, entre outros) e foi utilizado para disponibilizar os materiais didáticos (RA, videoaulas, tutoriais, apostilas, páginas *web*, etc.) e promover as interações virtuais, também feitas por Whatsapp.

Para dar suporte ao estudo autônomo, produzimos vídeos autorais e também utilizamos vídeos de terceiros, especialmente selecionados para apoiar o estudo das temáticas. Cada vídeo tinha em média sete (07) minutos e foram produzidas 27 videoaulas autorais (totalizando, 3 horas e 28 minutos) e selecionados 17 vídeos de terceiros (totalizando, 1 hora e 48 minutos). A produção e a seleção dos vídeos foram realizadas após a determinação dos temas e conteúdos. No caso das videoaulas autorais, antes de cada gravação, um roteiro foi escrito para que o vídeo fosse objetivo e relativamente curto. Para gravação, inserção de textos explicativos, anotações, edição e acréscimo de efeitos foi utilizado o *software* Camtasia Studio por agregar todas estas funções. A publicação dos vídeos autorais foi feita no Youtube. Além dos vídeos, foram disponibilizados outros recursos como tutoriais *online*, conteúdos de páginas *web* e indicação de leitura de capítulos de livros.

² O Linux é um sistema operacional bastante difundido, executado em mais de dois terços dos servidores *web*. Também possibilita facilmente outros serviços de rede, além de ser *software* livre (com código aberto, gratuito), com vasto material *online* e comunidades ativas. A escolha pelo Linux Ubuntu se deveu a sua popularidade, pois aproximadamente metade dos servidores Linux identificados na Internet utilizam esta distribuição (NETCRAFT, 2017).

³ A plataforma de virtualização Oracle VM Virtualbox foi escolhida por ser um *software* livre e dotado de interface intuitiva.

⁴ A IDE Thonny é descrita pelos seus idealizadores como “IDE para iniciantes”. Foi escolhida por ter uma interface amigável, por ser “leve” e multiplataforma (THONNY.ORG, 2018).



3.5 Avaliações

Planejamos a realização de três avaliações (uma diagnóstica e duas modulares), aplicadas individual e presencialmente. A avaliação diagnóstica foi estruturada em duas partes: uma, abordando teoria, na forma de questões abertas e de múltipla escolha; e outra abordando problemas práticos envolvendo a administração de redes, na qual os estudantes deveriam prover soluções na forma de programas. A avaliação diagnóstica foi realizada no primeiro encontro e tinha por objetivo capturar os conhecimentos prévios dos alunos, tanto do ponto de vista conceitual, quanto de programação.

As avaliações modulares foram aplicadas ao final de cada módulo e compreendiam apenas problemas práticos de redes, para os quais os alunos deveriam implementar programas para solucioná-los. Estas avaliações tinham por objetivo prover *feedback* sobre quão bem os estudantes alcançaram os resultados pretendidos da aprendizagem, que diziam respeito a automatização de soluções para problemas envolvendo administração de redes de computadores.

4 Implementação da Proposta

A proposta de ensino-aprendizagem foi implementada na forma de um curso de extensão, com carga horária de 60 (sessenta) horas, sendo 40 (quarenta) horas para estudos presenciais e 20 (vinte) horas para estudos autônomos. O curso foi oferecido pelo Instituto Federal de Rondônia, Campus Zona Norte, para estudantes de CST na área de Informática, sendo apenas exigido como pré-requisito o estudante ter cursado previamente alguma disciplina de programação. O curso foi realizado entre novembro de 2017 e fevereiro de 2018 e contou com a participação efetiva de 15 estudantes que concluíram o curso.

Ressaltamos que está é uma pesquisa de natureza aplicada, com abordagem predominantemente qualitativa, pois levou em conta a observação dos fenômenos durante a aplicação da proposta de ensino.

No primeiro dia do curso a proposta de ensino foi apresentada aos estudantes, assim como a dinâmica dos estudos por meio da sala de aula invertida. Neste dia também foi realizada a avaliação diagnóstica. No geral, o desenvolvimento do curso ocorreu da seguinte maneira: a cada início de semana, o professor disponibilizava o RA autônomo e os materiais didáticos no Google Classroom, ficando disponível para tirar dúvidas por *e-mail* e, principalmente, por Whatsapp em um grupo criado para este fim. No sábado pela manhã, facultativamente, em um horário das 9:00 às 11:30 havia uma tutoria com a presença do professor em que os estudantes poderiam elucidar suas dúvidas sobre os estudos autônomos e utilizar o laboratório de informática.

Nos estudos presenciais, aos sábados, das 14:00 às 18:00, os estudantes tinham acesso ao RA para estudo presencial, que os guiaria na resolução de problemas práticos. Com o conteúdo básico previamente trabalhado pelos estudantes, ao professor cabia a orientação da parte prática e elucidação de dúvidas durante o percurso de resolução de problemas, com oportunidade de atender melhor às dificuldades individuais. Após o estudo presencial, as tarefas eram corrigidas e, durante a semana, um *feedback* sobre o desempenho era dado a cada estudante, contendo apontamentos sobre quais conteúdos ou habilidades deveria dedicar mais atenção.

As avaliações modulares, nas datas marcadas, eram disponibilizadas em formulários eletrônicos no Google Classroom, realizadas individualmente pelos estudantes no laboratório de informática, com o uso de computadores e do laboratório virtual. As questões destas avaliações requeriam programação em Python e as soluções eram desenvolvidas na IDE Thonny, no ambiente Linux e as respostas, bem como os códigos resultantes, eram copiados no formulário eletrônico.



5 Resultados

Dos estudantes que participaram do curso, 80% eram de estudantes do sexo masculino e 20% do sexo feminino e a média de idade era de 24,2 anos. Do total, 60% dos alunos cursavam o segundo período de seus respectivos cursos, 27% cursavam o quarto período e 13% cursavam o sétimo ou oitavo período. Quanto aos cursos superiores, 53% estavam matriculados e cursando CST em Redes de Computadores, 20% CST em Análise e Desenvolvimento de Sistemas, 20% Bacharelado em Sistemas de Informação e 7% Engenharia da Computação. Todos informaram terem cursado disciplinas de redes de computadores e de programação em seus respectivos cursos superiores.

Os resultados da avaliação diagnóstica revelaram dificuldades dos estudantes com a resolução de problemas envolvendo programação, pois apenas seis estudantes conseguiram desenvolver alguma solução, ainda que seus aproveitamentos individuais não tenham ultrapassado 25% de acertos. Também foram evidenciadas dificuldades com questões básicas sobre teorias de redes de computadores, pois mais da metade dos estudantes erraram questões objetivas envolvendo protocolos ICMP (*Internet Control Message Protocol*) e o comando PING; SNMP (*Simple Network Management protocol*) e nomenclaturas básicas; protocolos de compartilhamento de arquivos e diretórios; e noções básicas sobre TCP/IP e OSI/ISO. Havia uma expectativa maior sobre este desempenho, visto que 73,3% dos estudantes assumiram ter maior afinidade com redes de computadores do que com programação e que todos haviam cursado ao menos uma disciplina de redes de computadores.

Nos estudos autônomos, observamos dificuldades dos estudantes para estudarem sozinhos, pois era uma prática nova para a maioria. Do total de alunos, apenas 33,3% deles afirmaram estudar mais de 2 horas por semana fora da sala de aula. Essa dificuldade se refletia nos estudos presenciais, havendo ocasiões em que a fase de aquecimento teve de ser estendida de 30 minutos para 1 hora, atrasando o início das tarefas práticas. Para ajudar os estudantes a mitigar esta dificuldade, conversamos com o grupo sobre estratégias para melhorar a gestão do tempo e a importância do estudo além do espaço de sala de aula. No decorrer do curso o início das atividades práticas nos estudos presenciais foi se normalizando e, nos últimos temas a dedicação para os estudos autônomos havia aumentado para 3 horas semanais, em média.

Foram entregues concluídos, em média, 86% dos RA para estudos autônomos e 87% dos RA para estudos presenciais, ainda que alguns estudantes tenham atrasado suas entregas. As alegações para os atrasos envolviam problemas pessoais e conflitos com as atividades dos estudantes em seus cursos regulares.

Para avaliarmos os recursos disponibilizados, ao final de cada encontro presencial os estudantes eram questionados sobre a qualidade dos vídeos e de outros materiais disponibilizados. De forma geral, os estudantes avaliaram estes recursos como adequados ao tempo e ao tema de cada estudo. Houve solicitação de acréscimo de materiais sobre Linux e redes de computadores. Em virtude disso, produzimos uma apostila básica sobre os assuntos e dedicamos quatro horas para uma aula extra presencial antes do final do primeiro módulo, o que resultou em uma queda acentuada quanto às dúvidas sobre tais assuntos no decorrer do curso.

No início do curso, 55,87% dos estudantes afirmaram recorrer a outras fontes além das indicadas nos RA autônomos, principalmente sobre conceitos de programação em Python, ainda que tivéssemos indicado vídeos sobre todos os conteúdos de programação exigidos no curso. Fizemos um trabalho individual incentivando uma melhor exploração do material e ao final do curso apenas 15,38% afirmaram ter de recorrer a materiais extras em seus estudos autônomos.



Para aferirmos a evolução dos estudantes quanto as suas habilidades para resolução de problemas de redes com recursos de programação, levamos em conta, além das avaliações, os estudos presenciais. Notamos dificuldades ao longo das três primeiras temáticas e atuamos junto a cada estudante, presencial e remotamente, para elucidar suas dúvidas. Na avaliação intermediária, ocorrida ao final do primeiro módulo, a média da turma foi 6,8 pontos (em uma escala de 0 a 10), acima da nota mínima para aprovação no curso e também acima da média aceita para aprovação nos cursos superiores regulares que os estudantes cursavam, ambos de 6,0 pontos. Na avaliação final, realizada após o término do segundo módulo, a média da turma foi de 8,7 pontos (em uma escala de 0 a 10), representando uma melhora significativa no desempenho do grupo.

Acreditamos que a implementação desta proposta de ensino como uma disciplina do currículo pode contribuir para um melhor desempenho dos estudantes. Na forma de um curso de extensão, demandou atividades extras para além das atividades desenvolvidas pelos alunos no período letivo, principalmente se considerarmos a natureza aplicada da proposta e a necessidade de estudo autônomo. Este é um aspecto que será mais bem avaliado em trabalhos futuros.

Com respeito à adoção do Google Classroom, observamos ampla aceitação por parte dos estudantes, principalmente para o compartilhamento de materiais. Contudo, esta plataforma não foi muito utilizada pelos alunos para as interações virtuais, neste caso, os alunos preferiram utilizar o Whatsapp.

6 Considerações Finais

Neste trabalho evidenciamos a trajetória de preparação e aplicação de uma experiência de ensino-aprendizagem pautada no modelo de sala de aula invertida para desenvolver em estudantes de nível superior conhecimento para a resolução de problemas práticos de redes de computadores, com recursos de programação em Python.

Em outros trabalhos encontramos relatos de aplicações de sala de aula invertida como em Santos et al. (2017), que relataram resultados positivos com a aplicação do modelo no ensino médio. Chen (2014) descreveu melhoria nas notas finais dos estudantes após aplicação do modelo no ensino de programação em C em curso superior. Oliveira et al. (2017) propuseram uma série de análises sobre a aplicação de metodologias semelhantes em disciplinas de programação em C. Medeiros e Bessa (2017) propuseram uma ferramenta para análise de avaliações em grandes turmas, utilizando o modelo de sala de aula invertida. Leite, Hoji e Abdalla Junior (2018) descrevem os desafios para a implementação do modelo de sala de aula invertida em disciplinas de Redes de Computadores em um curso técnico. Diferentemente destes trabalhos, além de implementar o modelo de sala de aula invertida, temos por objetivo conceber um guia didático contendo o planejamento, a metodologia detalhada da implementação, assim como os recursos empregados, para que a experiência possa ser reproduzida por outros professores.

Como trabalhos futuros, pretendemos também melhorar alguns aspectos da proposta, como a adição de videoaulas para nivelamento sobre redes de computadores, programação em Python e Linux. Adicionalmente, pretendemos implementar a proposta em uma disciplina regular de redes de computadores.

Referências

BENDER, W. N. **Aprendizagem baseada em projetos** – Educação diferenciada para o Século XXI. Porto Alegre: Penso, 2014.



BERGMANN, J.; SAMS, A. **Flip YOUR Classroom** – Reach Every Student in Every Class Every Day. Alexandria: ISTE, 2012.

BIGGS, J.; TANG, C. **Teaching for Quality Learning at University**. 4. ed. Berkshire, England: Society for Research into Higher Education & Open University Press, 2011.

BRASIL. Ministério da Educação (2002). **Parecer Conselho Nacional de Educação** – Conselho Pleno nº 29/2002. Institui as Diretrizes Curriculares Nacionais Gerais para a organização e o funcionamento dos cursos superiores de tecnologia. Disponível em: <<http://portal.mec.gov.br/cne/arquivos/pdf/cp29.pdf>>. Acesso em: 28 abr. 2018.

BRASIL. Ministério da Educação (2006). **Decreto n.º 5.773, de 9 de maio de 2006**. Dispõe sobre o exercício das funções de regulação, supervisão e avaliação de instituições de educação superior e cursos superiores de graduação e sequenciais no sistema federal de ensino. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2006/decreto/d5773.htm>. Acesso em: 28 abr. 2018.

BRASIL. Ministério da Educação. **Catálogo Nacional de Cursos Superiores de Tecnologia**. 3 ed. Brasília: jun. 2016. Disponível em: <http://portal.mec.gov.br/index.php?option=com_docman&view=download&alias=44501-cnecst-2016-3edc-pdf&category_slug=junho-2016-pdf&Itemid=30192>. Acesso em: 28 abr. 2018.

CHEN, E. Z., Flipped Classroom Model and Its Implementation in a Computer Programming Course. **Lärlärdom**, vol. 2014, p. 180-200, 2014. Disponível em: <<http://www.diva-portal.org/smash/get/diva2:777869/FULLTEXT01.pdf>>. Acesso em: 28 abr. 2018.

CHRISTENSEN, C.; HORN, M.; STAKER, H. Ensino Híbrido: uma Inovação Disruptiva? Uma introdução à teoria dos híbridos. **Clayton Christensen Institute for disruptive innovation**, 2013. Disponível em: <https://s3.amazonaws.com/porvir/wp-content/uploads/2014/08/PT_Is-K-12-blended-learning-disruptive-Final.pdf>. Acesso em: 28 abr. 2018.

HORN, M.; STAKER, **Blended - Using Disruptive Innovation to Improve Schools**. San Francisco: Jossey-Bass, 2015.

KUROSE, J. F.; et al. Workshop on computer networking: curriculum designs and educational challenges. **ACM SIGCOMM Computer Communications Review**, v. 32, n. 5, p. 1–9, Nov. 2002. Disponível em <Workshop on computer networking: curriculum designs and educational challenges>. Acesso em: 28 abr. 2018.

LEITE, F. N.; HOJI, E. S.; ABDALA JUNIOR, H., A Blended Learning Method Applied in Data Communication and Computer Networks Subject. **IEEE Latin America Transactions**, v. 16, n. 1, jan. 2018. p. 163-171. Disponível em: <http://www.ewh.ieee.org/reg/9/etrans/ieee/issues/vol16/vol16issue01Jan.2018/16TLA1_25NLeite.pdf>. Acesso em: 28 abr. 2018.

MEDEIROS, R. A. C.; BESSA, A. MiniTeste: uma ferramenta ágil para aplicação de avaliações personalizadas. **RENOTE** - Revista Novas Tecnologias na Educação, Porto Alegre, v. 15, n. 1, jul. 2017. Disponível em: <<http://seer.ufrgs.br/index.php/renote/article/view/75126>>. Acesso em: 13 mai. 2018.



NETCRAFT. **September 2017 Web Server Survey** - Web-facing operating systems. Disponível em: <<https://news.netcraft.com/archives/2017/09/11/september-2017-web-server-survey.html>>. Acesso em: 28 abr. 2018.

OLIVEIRA, M. G. et al. Um Curso de Programação a Distância com Metodologias Ativas e Análise de Aprendizagem por Métricas de Software. **RENOTE** - Revista Novas Tecnologias na Educação, Porto Alegre, v. 15, n. 1, jul. 2017. Disponível em: <<http://www.seer.ufrgs.br/renote/article/view/75143>>. Acesso em: 13 mai. 2018.

SANTOS, A. C. et al. Ensino Híbrido: Relato de Experiência sobre o uso de AVEA em uma proposta de Sala de Aula Invertida para o Ensino Médio. **RENOTE** - Revista Novas Tecnologias na Educação, Porto Alegre, v. 15, n. 2, dez. 2017. Disponível em: <<http://seer.ufrgs.br/index.php/renote/article/view/79186/46020>>. Acesso em: 28 abr. 2018.

THONNY.COM. **Python IDE for beginners**. Disponível em <<http://thonny.org>>. Acesso em: 28 abr. 2018.

Programação para Administração de Redes de Computadores – Uma Experiência com Estudantes de Computação

Silmar Antonio Buchner de Oliveira^{1,2}, Andréa Pereira Mendonça¹

¹Mestrado Profissional em Ensino Tecnológico. Instituto Federal do Amazonas (IFAM)
Manaus – AM – Brasil.

²Instituto Federal de Rondônia (IFRO). Porto Velho – RO – Brasil.

silmar.oliveira@ifro.edu.br, andrea.mendonca@ifam.edu.br

Abstract. *In this paper we report an experience on conception and execution of a programming course on computer network administration. This course is aimed at students of undergraduate technology courses in Computing related areas. Organized according to a flipped classroom approach, the course focused on solving practical problems by using the Python programming language. The results obtained evidence student's progress in automatising solutions for network equipment and user actions monitoring, log analysis, data synchronizing as well as automatic alerts issuing. The teaching materials and the dynamics of the course will be integrated into a guideline to aid other teachers of computer network affined courses.*

Resumo. *Neste artigo relatamos uma experiência de criação e execução de um curso de programação para administração de redes de computadores, voltado para estudantes de Cursos Superiores de Tecnologia na área de Computação. Implementado no formato de sala de aula invertida, o curso focou na resolução de problemas práticos utilizando a linguagem Python. Os resultados evidenciaram progresso dos estudantes na automatização de soluções para monitoramento de equipamentos de redes e ações de usuários, análise de logs, sincronismos de dados e emissão automática de mensagens de alertas. O material de ensino e dinâmica do curso serão integrados em um guideline para auxiliar outros professores de redes de computadores.*

1. Introdução

Os currículos dos Cursos Superiores em Computação no Brasil, independente da categoria (bacharelado, tecnologia ou licenciatura), possuem em comum a presença de duas disciplinas: (i) **programação** que integra a área de formação básica; e (ii) **redes de computadores** que integra a área de formação tecnológica (BRASIL, 2012).

Tratando especificamente sobre os Cursos Superiores de Tecnologia (CST), observamos que a disciplina de redes de computadores é lecionada priorizando aspectos teóricos, tais como: topologia de redes; protocolos de comunicação, em especial, o modelo de referência OSI (*Open System Interconnection*) e a pilha de protocolos TCP/IP (*Transfer Control Protocol/Internet Protocol*); tipos de redes e aspectos básicos de interconexão de redes (WANG; BLUM; MCCOEY, 2014). Por outro lado, a disciplina de programação é ensinada dentro de uma perspectiva prática, voltada para a resolução de problemas com a adoção de uma linguagem de programação.

Ocorre que, ao ingressarem na vida profissional, os estudantes dos Cursos Superiores de Tecnologia são demandados para a resolução de problemas práticos. No caso de atuação na área de redes de computadores, são comuns demandas voltadas para administração de redes de computadores, as quais envolvem automatização de procedimentos para monitoramento, controle e gerenciamento de serviços, como execução de *backups*, sincronismo de dados, verificação de consumo de recursos, envio de mensagens de alerta via *e-mail*, etc. Estas demandas, por sua vez, não podem ser atendidas com base em uma formação de redes pautada apenas em aspectos teóricos e dissociada dos conhecimentos de programação.

Dada esta lacuna na formação, relatamos neste artigo a proposta e execução de um curso de programação para administração de redes de computadores, voltado para estudantes de CST. O curso focou na resolução de problemas práticos, adotou Python como linguagem de programação, foi administrado na modalidade de ensino híbrido, mais especificamente, no modelo de sala de aula invertida, combinando estudo autônomo e presencial, os quais foram conduzidos por meio de roteiros de aprendizagem (RA).

2. Ensino de redes de computadores nos CSTs

Os Cursos Superiores de Tecnologia (CSTs) possuem regulamentação específica e uma delas é o Catálogo Nacional de Cursos Superiores de Tecnologia, disponibilizado pela Secretaria de Educação Profissional e Tecnológica do Ministério da Educação (SETEC/MEC). Os Catálogos são organizados por eixos, sendo o eixo Informação e Comunicação aquele que integra os cursos relacionados à área de Computação/Informática, como CST em Análise e Desenvolvimento de Sistemas, CST em Sistemas para Internet, CST em Redes de Computadores, CST em Jogos digitais, entre outros (BRASIL, 2016).

Por sua característica direcionada à prática profissional, os CSTs têm duração mais curta (entre 2 e 3 anos), em comparação com os cursos de bacharelado. Seus egressos possuem um perfil profissional dotado de compreensão sobre o processo produtivo, fundado no saber tecnológico e no ato de fazer, com autonomia para tomada de decisões profissionais, indo além do simples domínio operacional de técnicas de trabalho. Geralmente, os CST são oferecidos pelos Institutos Federais de Educação, pertencentes à rede Federal de Educação Profissional, Científica e Tecnológica, criada em 2008. Há também a oferta de CST em universidades públicas e Instituições de Ensino Superior particulares (BRASIL, 2006; BRASIL, 2002).

A fim de caracterizar o perfil das disciplinas de redes de computadores e de programação em CST, realizamos uma pesquisa *ad hoc* em cursos do eixo Informação e Comunicação, oferecidos pelos Institutos Federais de Educação. Como resultado desta pesquisa, identificamos uma relação de 60 (sessenta) cursos, distribuídos em 40 (quarenta) instituições, conforme informações obtidas no sistema e-Mec do Ministério da Educação.

A partir da identificação dos cursos, analisamos as ementas das disciplinas iniciais de redes de computadores e identificamos que as mesmas eram essencialmente teóricas, baseadas em alguma edição das obras de Andrew S. Tanenbaum (Redes de

· <http://emec.mec.gov.br>

Computadores) e de James F. Kurose (Redes de Computadores e a Internet), centradas no modelo OSI e na pilha de protocolos TCP/IP.

A qualidade das obras é indiscutível. Contudo, os resultados da pesquisa *ad hoc* apontam um distanciamento entre teoria e prática na forma como a disciplina é ensinada, no tocante à resolução de problemas de redes de computadores, quando considerado o contexto profissional no qual atuam os egressos dos CST.

Na pesquisa, identificamos 9 (nove) CST que possuíam em sua matriz curricular uma disciplina cuja denominação (por exemplo, Programação de Scripts e Programação Aplicada à Gerência de Redes de Computadores) indicava a junção de redes de computadores e programação. Entretanto, não identificamos informações sobre como as mesmas são aplicadas, de forma que se outro professor quiser repetir a experiência, terá dificuldades, pois não encontrará planejamento, metodologias detalhadas ou problemas práticos abordados nestas disciplinas.

Destacamos ainda que, nos CSTs do Eixo Informação e Comunicação há pelo menos uma disciplina de programação no currículo, ensinada no primeiro ano do curso, antes da disciplina de redes de computadores. Tal organização do currículo poderia resultar no aproveitamento dos conhecimentos adquiridos em programação, dentro da disciplina de redes de computadores, possibilitando aos alunos a resolução de problemas práticos com o auxílio de programação.

Corroborando com os resultados de nossa pesquisa *ad hoc*, Chen (2014) aponta que os cursos de redes de computadores possuem muito conteúdo teórico, abstrato e de difícil entendimento, muitas vezes prejudicando o interesse na aprendizagem e resultando em uma capacidade prática deficitária. O autor sugere como solução a adoção de atividades práticas para a resolução de problemas reais. Kurose et al. (2002) também fazem críticas ao ensino essencialmente teórico nos cursos de redes de computadores e citam a importância de utilizar laboratórios com aplicações práticas. Inclusive, levantam perspectivas sobre quando e quanto programar em redes de computadores. Enfatizam ainda que os estudantes de redes de computadores que aprendem a codificar programas podem ter mais facilidade para internalizar e entender conceitos avançados, além de desenvolver um conhecimento “comercializável”. Wang, Blum e McCoey (2014) ao discutirem o ensino de redes de computadores para iniciantes, sugerem que, além da introdução geral e tradicional, devem proporcionar atividades práticas que envolvam programação para resolução de problemas.

Frente a esta lacuna de formação, desenvolvemos a proposta de um curso de programação para administração de redes de computadores, focado na resolução de problemas práticos, que adota a linguagem de programação Python e organiza as práticas de ensino-aprendizagem de acordo com o ensino híbrido, mais especificamente, o modelo de sala de aula invertida, como será detalhado na seção seguinte.

3. Metodologia do curso

O curso está estruturado em dois módulos: o primeiro, abrangendo assuntos referentes ao monitoramento de redes e de ações de usuários; o segundo, integrando assuntos sobre a preservação das informações e comunicação de eventos via e-mail e Telegram. Cada assunto está associado a uma temática, cujo desdobramento se dá pela integração de conteúdos sobre redes, sistemas operacionais e programação, conforme pode ser

observado no Quadro 1. Além disso, para cada assunto, há um conjunto de problemas propostos, nos quais os estudantes devem integrar teoria e prática, propondo soluções automatizadas com uso de programação. Os assuntos abordados em cada temática são cumulativos e suas complexidades evoluem conforme o andamento do curso.

Quadro 1. Módulos, temas, assuntos e conteúdos.

Módulo 1 - Monitoramento de redes de computadores
Tema 01: Identificando a movimentação da vizinhança Assunto: Monitoramento de disponibilidade de equipamentos e serviços em rede. Conteúdos principais: <u>Redes:</u> Comando "ping"; Protocolo SNMP. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados. <u>S.O. Linux:</u> Estruturas de arquivos e diretórios; - Monitoramento de carga e de estados de componentes. Problemas propostos: Detecções de problemas de lentidão ou perda de acesso, através de monitoramentos via SNMP e "ping", relacionados ao estado de máquinas e componentes.
Tema 02: Cuidando do Habitat Assunto: Monitoramento de estado de discos e partições, em rede. Conteúdos principais: <u>Redes:</u> Protocolo NFS; Protocolo SNMP. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados. <u>S.O. Linux:</u> Compartilhamento de dados; Estruturas de arquivos e diretórios; Problemas propostos: Monitoramentos remotos de discos e partições como prevenção de estouros de espaços de armazenamento.
Tema 03: De olho nas intenções Assunto: Monitoramento de ações de usuários. Conteúdos principais: <u>Redes:</u> Protocolo SSH, Acesso remoto; <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; Operações sobre arquivos; Operações com <i>strings</i> e datas; Laços; Decisão. <u>S.O. Linux:</u> Editores de textos; Acesso remoto; Arquivos de configuração; Hierarquia de usuários; Históricos de ações; Data e hora; Problemas propostos: Monitoramentos de ações de usuários através de verificação automatizada de <i>logs</i> do sistema operacional.
Módulo 2 - Preservação de informações e comunicação de eventos
Tema 04: Construir é difícil. Reconstruir pode ser impossível Assunto: Compactação de dados, cópias de segurança em rede e registros de atividades (<i>logs</i>). Conteúdos principais: <u>Redes:</u> Backup; Compartilhamento de arquivos; Protocolo NTP; Protocolo NFS. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; Operações com <i>strings</i> e datas; Laços; Decisão. <u>S.O. Linux:</u> Empacotamento e compactação de dados; Agendamento de tarefas; Sincronismo de data e hora; Arquivos de <i>logs</i> . Problemas propostos: Agendamentos de execuções de backup em rede, com verificação de etapas de execução e gravação em arquivos de <i>log</i> .
Tema 05: Um é pouco. Então tenha dois (ou mais!) Assunto: Sincronismo de dados em rede. Conteúdos principais: <u>Redes:</u> Backup; Políticas de backup; Sincronismo de dados remotos; Compartilhamento de arquivos; Protocolo SSH; Protocolos NFS. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; Operações com <i>strings</i> e datas; Decisão. <u>S.O. Linux:</u> Sincronismo de dados; Agendamento de tarefas; Arquivos de <i>logs</i> . Problemas propostos: Agendamentos de execuções de sincronismo de dados em rede, com verificação de etapas de execução e gravação em arquivos de <i>log</i> .
Tema 06: Seja o primeiro a saber Assunto: Automatização de mensagens de eventos e de alertas via <i>e-mail</i> e mensageria instantânea (Telegram). Conteúdos principais: <u>Redes:</u> Serviços de e-mail; Protocolo SMTP; Protocolos SSL/TLS e StartTLS. <u>Programação:</u> Bibliotecas e métodos; variáveis; entrada e saída de dados; Envio de <i>e-mails</i> ; Operações com <i>strings</i> . <u>S.O. Linux:</u> Envios de <i>e-mails</i> ; Envio de mensageria instantânea; Sincronismo de dados; Agendamento de tarefas; Configuração de serviços. Problemas propostos: Automatizações de procedimentos com verificações de etapas de execuções e, em função dos resultados, envio de avisos e alertas via e-mail e/ou Telegram.

Fonte: Autoria Própria.

Para a definição dos conteúdos, realizamos uma pesquisa em livros nacionais e internacionais que tratam sobre redes de computadores, conforme apresentado no Quadro 2. Com base nesta pesquisa, selecionamos os assuntos mais frequentes e identificamos também os tipos de problemas associados aos conteúdos.

Quadro 2. Relação livro x conteúdo

Legenda dos conteúdos: 1-Monitoramento de Redes; 2-Compartilhamento de dados; 3-Envio de e-mails/relatório; 4-Gerenciamento de sistemas de arquivos; 5-Gerenciamento de usuários/acesso; 6-Login; 7-Monitoramento de sistemas; 8-Gerenciamento de Arquivos e Diretórios; 9-Backup; 10-SSH/acesso remoto; 11-Análise de logs; 12-Compressão de dados; 13-Autenticação segura; 14-Expressões regulares; 15-Manipulação de strings.

Relação Livro x Conteúdo		Conteúdo														
Livro	Ano	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1-Administração de redes com scripts	2007															
2-Automating Linux and Unix System Administration	2009															
3-Automating System Administration with Perl	2009															
4-Automating Windows Administration	2004															
5-Automating Windows Server 2008 Administration with PowerShell	2008															
6-Core Python Applications Programming	2012															
7-Essential System Administration-Tools and Techniques for Linux and Unix	2002															
8-Foundations of Python Network Programming	2014															
9-Learning Python Network Programming	2016															
10-Linux Command Line and Shell Scripting Bible	2015															
11-Linux System Administration Recipes	2009															
12-Pro Python System Administration	2014															
13-Python for Unix and Linux System Administration	2008															
14-Shell Scripting Recipes	2015															
15-Unix and Linux System Administration Handbook	2011															
16-Wicked Cool Shell Scripts - 101 scripts	2004															
Total de ocorrências		9	8	8	8	8	7	7	7	6	6	4	4	3	3	3

Fonte: Autoria Própria.

A dinâmica para administração dos conteúdos e resolução dos problemas foi organizada obedecendo ao modelo de ensino híbrido de sala de aula invertida, criado por Bergmann e Sams (2012). De acordo com este modelo, uma parte da aprendizagem é feita de forma autônoma pelo estudante, fora da sala de aula, utilizando recursos tecnológicos. A outra parte é realizada em sala de aula, com a presença de um professor. Assim, a carga horária do curso de 60 (sessenta) horas, foi distribuída em 40 (quarenta) horas para estudos presenciais e 20 (vinte) horas para estudos autônomos.

Para guiar o aluno tanto no estudo presencial quanto no autônomo, baseados nos conteúdos das obras elencadas no Quadro 2, definimos os Roteiros de Aprendizagem (RA) que são instrumentos que conduzem o aluno por um itinerário de estudo a fim de que ele atinja os resultados pretendidos da aprendizagem. Para cumprir os assuntos descritos no Quadro 1, doze (12) roteiros foram elaborados, sendo 6 (seis) roteiros de aprendizagem para estudo autônomo¹ e 6 (seis) roteiros para estudo presencial². Os RAs para estudo autônomo e presencial se assemelham na sua estrutura geral, a diferença reside na seção de tarefas, conforme pode ser verificado no Quadro 3.

¹ Um exemplo de RA autônomo (RA Autônomo 01) está disponível em <https://goo.gl/t6wTj5>.

² Um exemplo de RA presencial (RA Presencial 01) está disponível em <https://goo.gl/qeTGDu>.

Para auxiliar na realização dos estudos autônomos, foram criadas 27 vídeoaulas, com tempo médio de 7 minutos cada, num total de 3 horas e 28 minutos, postadas no Youtube. Também foram utilizados vídeos de terceiros como os que tratavam sobre Python e redes, tais como: Python para Zumbis¹, Curso em vídeo² e Nic.br³.

Quadro 3. Estrutura geral dos Roteiros de Aprendizagem (RAs).

Cabeçalho
Identifica o módulo corrente, o número do roteiro, o tema abordado, o tipo de estudo (presencial ou autônomo) e a carga horária prevista.
Descrição geral
Apresenta orientações sobre como seguir o RA.
Resultados pretendidos da aprendizagem
Descreve os objetivos da aprendizagem a serem atingidos pelos estudantes.
Conteúdos envolvidos
Apresenta os conteúdos abordados, referentes a redes de computadores, programação e sistemas operacionais (Linux).
Tarefas
Nos estudos autônomos , orientam a evolução do estudante sobre os conteúdos preparatórios para a resolução dos problemas que serão apresentados nos estudos presenciais. Tais tarefas são dotadas de questões cujas resoluções são guiadas por vídeos, tutoriais <i>online</i> e <i>links</i> de páginas <i>web</i> para a aplicação prática em um laboratório virtual, criado especificamente para o curso.
Os estudos presenciais são compostos de dinâmicas para resolução de problemas, inspiradas em Bender (2014) e em Biggs e Tang (2011), divididas em: <u>Aquecimento</u> - Durante os primeiros 30 minutos de aula, os estudantes expõem suas dúvidas sobre o estudo autônomo, em uma discussão mediada pelo professor. <u>Apresentação do contexto</u> - É apresentado um contexto envolvendo um problema com características reais de redes de computadores. <u>Discussão</u> - São dispostas questões que instigam o estudante a tentar entender melhor a solução do problema e fomentem uma discussão sobre formas de solucioná-lo. <u>Resolução</u> - O estudante descreve como será a solução para o problema e quais recursos de programação utilizará. Depois constrói um programa em Python consonante com a descrição e o executa no laboratório virtual.

Fonte: Autoria Própria.

O material do curso foi disponibilizado de forma virtual no Google Classroom⁴. Outros canais virtuais também foram utilizados, como os que serviam para elucidação de dúvidas sobre o estudo autônomo, como e-mails e mensagens Whatsapp. De forma complementar, uma vez por semana havia duas horas e meia dedicadas ao atendimento aos alunos, cuja presença era facultativa. Estes atendimentos tinham por objetivo dirimir dúvidas e explicar conceitos, antes de cada estudo presencial. Também foram disponibilizados horários extras no laboratório para que os estudantes pudessem realizar

¹ <https://www.pythonschool.com/python-para-zumbis/>

² <http://www.cursoemvideo.com/course/curso-python-3/>

³ <http://www.nic.br/videos/>

⁴ Google Classroom (no Brasil, Google Sala de Aula) é uma plataforma web da Google que reúne um conjunto de serviços (e-mail, drive, gerenciador de tarefas, entre outros) para auxiliar e promover atividades educacionais.

estudos autônomos e sanar dúvidas sobre os estudos presenciais, dispondo de computadores e Internet na própria instituição de ensino.

Três avaliações foram planejadas para o curso: (i) avaliação diagnóstica, realizada antes de iniciar o curso e que tinha por objetivo identificar os conhecimentos prévios dos alunos; (ii) avaliação intermediária, planejada para o final do primeiro módulo; e, (iii) avaliação final, a ser realizada após o término do segundo módulo. Cada avaliação possuía estrutura semelhante aos roteiros de aprendizagem para estudo presencial, com tarefas direcionadas à resolução de problemas de redes com uso de programação.

A primeira experiência de aplicação do curso foi realizada no período de novembro/2017 a janeiro/2018, na forma de um curso de extensão, ofertado no Instituto Federal de Educação, Ciência e Tecnologia de Rondônia - IFRO, com 30 vagas para estudantes de cursos de nível superior na área de Informática. O pré-requisito era tão somente o estudante ter cursado previamente alguma disciplina de programação. Para este curso, houve 53 (cinquenta e três) inscrições, sendo 30 (trinta) estudantes classificados, conforme o coeficiente de rendimento. Como o período de realização do curso coincidiu com o encerramento do semestre, concorrendo com as avaliações finais dos cursos regulares nos quais os alunos estavam matriculados, e com as férias de final de ano, 15 (quinze) alunos desistiram, mesmo tendo assinando termo de compromisso. Desta forma, os resultados apresentados na próxima seção serão reportados com base nos alunos que concluíram o curso (15 alunos).

4. Resultados

Dos alunos que concluíram o curso, 80% eram do sexo masculino e 20% do sexo feminino. A média de idade do grupo era de 24,2 anos. Identificamos também que 80% dos alunos haviam cursado, no mínimo, uma disciplina de redes na graduação.

Como falamos anteriormente, antes de iniciarmos o curso, os alunos realizaram uma avaliação diagnóstica contendo: (i) questões objetivas sobre a teoria básica de redes de computadores, envolvendo classificação de redes por distribuição geográfica, principais protocolos e modelos de pilhas de protocolos; e, (ii) uma questão prática envolvendo a automatização com verificação de disponibilidade de equipamentos em uma rede, com envio de alerta por e-mail.

Com respeito às questões teóricas objetivas, mais de 84% dos estudantes não acertaram questões básicas como associar o comando *ping*^{*} ao protocolo ICMP (*Internet Control Message Protocol*). Dificuldade semelhante foi percebida quando quase 70% dos estudantes tampouco demonstraram conhecimentos sobre a utilidade do protocolo SNMP^{*} (*Simple Network Management Protocol*). Com respeito à questão prática, 30% dos estudantes deixaram de resolvê-la e, dos que tentaram resolver, seus desempenhos não ultrapassaram 35% de acertos.

Considerando a característica do grupo, no qual 80% dos alunos já haviam cursado uma disciplina de redes de computadores, era esperado melhores resultados nas questões teóricas. Os resultados desta avaliação diagnóstica ratificam as dificuldades de

* O comando *ping* faz parte do protocolo ICMP e é utilizado para testes básicos de comunicação entre computadores.

* SNMP é um protocolo utilizado para gerenciamento de redes de computadores e seus componentes.

aprendizagem em redes, conforme contextualizamos na Seção 2.

Com respeito à realização dos roteiros de aprendizagem administrados ao longo do curso, os estudantes concluíram 86% dos RA para estudo autônomo e 87% dos RA para estudo presencial. Embora a evolução dos estudantes tenha sido evidenciada ao longo do curso, percebemos na maioria dos alunos uma dificuldade inicial em desenvolver seus estudos autônomos, pois a experiência de “estudar sozinho” parecia inédita e/ou desconfortável para muitos. Em virtude disso, houve entregas posteriores ao prazo estabelecido e também ausências de entregas, conforme constatado pelos números apresentados acima. Cabe destacar que, a não entrega dos RA para estudo autônomo acabava impactando na entrega dos RA para estudo presencial, pois faltava aos alunos a fundamentação conceitual necessária para a resolução dos problemas propostos.

No decorrer do curso, os estudantes foram capazes de automatizar soluções para 12 (doze) problemas, distribuídos em 6 (seis) RA para estudo presencial. Estes problemas evoluíram desde situações simples envolvendo monitoramento de conexões de redes com o comando “*ping*”, até automatização de análise remota de *logs*, monitoramento e emissão de alertas sobre a ocupação de partições de discos de máquinas remotas, sincronismo de dados entre servidores, cópias de segurança em rede, além de envio de alertas automáticos via e-mail e Telegram, orientados à criticidade de cada situação. Os alunos também tiveram que montar e administrar seus laboratórios virtuais, utilizando o sistema operacional Linux sobre a plataforma de virtualização Oracle VM Virtualbox, ferramentas até então desconhecidas pela maioria.

No processo de resolução de problemas, identificamos dificuldades dos alunos com respeito à programação, mais especificamente, com a manipulação de *strings*, arquivos e estrutura de laços de repetição. Também identificamos desconhecimento sobre o sistema operacional Linux, mitigados pela intensificação de atendimento a alunos de forma presencial e remota, indicação de vídeos de terceiros e pela produção de uma apostila envolvendo estes assuntos.

A fim de proporcionar um *feedback* contínuo aos estudantes, ao final de cada estudo presencial, os estudantes recebiam de maneira individual, por e-mail ou Whatsapp informações sobre seu desempenho, com indicação de quais aspectos teóricos e/ou práticos deveriam melhorar. Estes *feedbacks* foram muito importantes e refletiram em melhorias no desempenho individual ao longo do curso.

Na avaliação intermediária, ocorrida ao final do primeiro módulo, a média da turma foi 6,8 pontos (em uma escala de 0 a 10), acima da nota mínima para aprovação no curso e também acima da média aceita para aprovação nos CST regulares que os estudantes cursavam, ambos de 6,0 pontos. Na avaliação final, realizada após o término do segundo módulo, a média da turma foi de 8,7 pontos (em uma escala de 0 a 10), representando uma melhora significativa no desempenho do grupo.

Com base nesta aplicação do curso foi possível identificar aspectos a serem melhorados, como por exemplo: (i) ampliar o material de apoio sobre Python no que diz respeito à manipulação de *strings*, arquivos e estrutura de repetição, com exercícios específicos para consolidar a aprendizagem; (ii) incluir conteúdos básicos sobre o sistema operacional Linux, a fim de proporcionar aos alunos melhor familiaridade com seus conceitos e operações; e, (iii) melhorar as vídeoaulas para esclarecer termos e conceitos básicos de redes, que foram mencionados pelos alunos ao longo do curso.

Com respeito ao estudo autônomo, a aplicação deste curso revelou algumas demandas para professores e gestores de Instituições de Ensino Superior (IES). Primeiro, é necessário que as IES disponibilizem infraestrutura adequada para os alunos, permitindo-lhes, por exemplo, acesso aos laboratórios ou outras dependências fora do horário de aula. Para os professores, é necessário estabelecer com a turma uma “cultura de estudo autônomo” e para isso, os alunos precisam ser orientados. No caso deste trabalho, identificamos os roteiros de aprendizagem como instrumentos adequados para orientar os alunos no estudo. Porém, a preparação dos roteiros e dos materiais de suporte demandam um tempo considerável de planejamento. Em contrapartida, possuem a vantagem de poder ser reutilizados em outras ofertas do curso.

Considerando a plataforma web utilizada, o Google Classroom teve alta aceitação entre os alunos, isto porque todos eram usuários de e-mail da Google e podiam facilmente aproveitar os recursos disponibilizados, com destaque para o compartilhamento de materiais, vídeos e submissão das respostas dos alunos aos roteiros de aprendizagem. Contudo, para as comunicações virtuais e trocas de informação, o tempo de resposta era mais efetivo com a adoção do Whatsapp.

5. Considerações Finais

Neste artigo reportamos o planejamento de um curso de programação para administração de redes de computadores, assim como os resultados de sua aplicação com estudantes de Computação. O curso reuniu conteúdos de redes, sistemas operacionais e programação, em uma perspectiva prática, pautada na resolução de problemas e tendo no modelo de sala de aula invertida a dinâmica para condução do processo de ensino-aprendizagem.

O planejamento do curso demandou seis meses de trabalho e incluiu a elaboração da ementa, conteúdos, organização dos roteiros, elaboração dos problemas práticos, criação dos recursos de apoio (vídeos, textos, códigos, laboratórios virtuais, etc.), além de identificação de recursos disponíveis na Internet que pudessem ser reutilizados no curso para auxiliar os alunos no estudo autônomo.

Como trabalhos futuros, desenvolveremos um *guideline* para professores, disponibilizando todo o planejamento e recursos produzidos para o curso. Acreditamos que isso possa contribuir com outros docentes que ministram a disciplina de redes de computadores. Almejamos também desenvolver uma análise focada no nível de complexidade dos problemas propostos, a fim de identificarmos pontos de correção e redimensionamento dos problemas e dos próprios conteúdos associados.

Adicionalmente, realizaremos novos estudos com a aplicação da proposta de ensino-aprendizagem descrita neste artigo, pois compreendemos que a quantidade de estudantes envolvida nesta primeira experiência representa uma limitação do trabalho. Assim, em aplicações futuras, pretendemos oferecer o curso como uma disciplina do currículo, administrada durante um semestre acadêmico, o que permitirá uma avaliação mais criteriosa do desempenho dos estudantes, suas dificuldades, assim como das potencialidades e limitações do próprio curso, a fim de que possamos ampliar as discussões sobre Educação em Computação, mais especificamente neste trabalho, sobre o ensino de redes de computadores.

Referências

- BENDER, W. N. **Aprendizagem baseada em projetos** – Educação diferenciada para o Século XXI. Porto Alegre: Penso, 2014.
- BERGMANN, J; SAMS, A. **Flip YOUR Classroom** – Reach Every Student in Every Class Every Day. Alexandria: ISTE, 2012.
- BIGGS, J.; TANG, C. **Teaching for Quality Learning at University**. 4. ed. Berkshire, England: Society for Research into Higher Education & Open University Press, 2011.
- BRASIL. Ministério da Educação (2002). **Parecer Conselho Nacional de Educação – Conselho Pleno nº 29/2002**. Institui as Diretrizes Curriculares Nacionais Gerais para a organização e o funcionamento dos cursos superiores de tecnologia. Disponível em: <<http://portal.mec.gov.br/cne/arquivos/pdf/cp29.pdf>>. Acesso em: 23 ago. 2016.
- BRASIL. **Parecer CNE/CES No 136/2012**. Institui Diretrizes Curriculares Nacionais para os cursos de graduação em Computação.
- BRASIL. Ministério da Educação (2006). **Decreto n.º 5.773, de 9 de maio de 2006**. Dispõe sobre o exercício das funções de regulação, supervisão e avaliação de instituições de educação superior e cursos superiores de graduação e sequenciais no sistema federal de ensino. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2004-2006/2006/decreto/d5773.htm>. Acesso em: 01 set. 2016.
- BRASIL. Secretaria de Educação Profissional e Tecnológica / Ministério da Educação (2016). **Catálogo Nacional de Cursos Superiores de Tecnologia**. 3a edição. Disponível em: <http://portal.mec.gov.br/index.php?option=com_docman&view=download&alias=44_501-cncst-2016-3ede-pdf&category_slug=junho-2016-pdf&Itemid=30192>. Acesso em: 10 fev. 2018.
- CHEN H, Discussion on the Teaching Methods of Applied Undergraduate Computer Network Courses. 3rd International Conference on Science and Social Research (ICSSR 2014), 2014, Paris. **Proceedings...** Paris: Atlantis Press, 2014, p. 127-130.
- KUROSE, J. F.; et al. Workshop on computer networking: curriculum designs and educational challenges. **ACM SIGCOMM Computer Communications Review**, v. 32, n. 5, p. 1–9, Nov. 2002.
- WANG, Y; BLUM, T; MCCOEY, M. Teaching a networking class for freshmen: course design and lessons learned. In: Proceedings of the 15th Annual Conference on Information technology education, 2014, Atlanta. **Proceedings...** Atlanta: 2014, p. 9-14.