



UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Câmpus de São José do Rio Preto

GIOVANA GIARDINI BORGES

***Framework* para auxílio no desenvolvimento de *soft skills* em
estudantes de Engenharia de Software**

São José do Rio Preto
2024

Giovana Giardini Borges

***Framework para auxílio no desenvolvimento de soft skills em
estudantes de Engenharia de Software***

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Campus de São José do Rio Preto.

Orientadora: Profa. Dra. Rogéria Cristiane Gratão de Souza

São José do Rio Preto
2024

B732f	Borges, Giovana Giardini Framework para auxílio no desenvolvimento de soft skills em estudantes de Engenharia de Software / Giovana Giardini Borges. -- São José do Rio Preto, 2024 99 f. : il., tabs. Dissertação (mestrado) - Universidade Estadual Paulista (Unesp), Instituto de Biociências Letras e Ciências Exatas, São José do Rio Preto Orientadora: Rogéria Cristiane Gratão de Souza 1. Ciência da computação. 2. Engenharia de Software. 3. Soft skills. 4. Aprendizagem ativa. I. Título.
-------	---

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca do Instituto de Biociências Letras e Ciências Exatas, São José do Rio Preto. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

Giovana Giardini Borges

***Framework para auxílio no desenvolvimento de soft skills em
estudantes de Engenharia de Software***

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Campus de São José do Rio Preto.

Comissão examinadora:

Profa. Dra. Rogéria Cristiane Gratão de Souza
Universidade Estadual Paulista “Júlio de Mesquita Filho” – São José do Rio Preto
Orientadora

Prof. Dr. Frank José Affonso
Universidade Estadual Paulista “Júlio de Mesquita Filho” – Rio Claro

Profa. Dra. Valéria Farinazzo Martins
Universidade Presbiteriana Mackenzie – São Paulo

São José do Rio Preto
23 de janeiro de 2024

Para minha família.

Agradecimentos

Agradeço a todos que me apoiaram e incentivaram durante meus estudos para o mestrado. Sou muito agradecida a meus pais, irmãos e amigos que sempre torceram por mim, além de todos os mestres que me ajudaram a trilhar o caminho até aqui.

Agradeço, acima de tudo, ao meu Deus e Senhor, Jesus Cristo, sem o qual nada seria possível. Obrigada.

*Aceitai a minha correção e não a prata, e o conhecimento mais do que o ouro fino
escolhido. Porque melhor é a sabedoria do que os rubis; e de tudo o que se deseja nada se
pode comparar a ela.
- Provérbios 8: 10-11*

Resumo

O bom profissional de *software* deve possuir habilidades técnicas e não técnicas, isto é, *hard* e *soft skills* para lidar com os diferentes desafios que encontrará no decorrer de sua carreira. Para que isso seja possível, o profissional deve desenvolver tais habilidades desde a graduação. Nesse contexto, apesar de haver trabalhos relacionados na literatura, não foi encontrado resultado voltado para apoiar os docentes no desenvolvimento de *soft skills* dos estudantes em sala de aula por meio de metodologias ativas de educação, com foco na motivação dos alunos. Assim, com o intuito de proporcionar um modelo que auxilie os docentes no desafio de contemplar *hard* e *soft skills* em um processo de ensino-aprendizagem mais significativo para os estudantes matriculados em disciplinas de Engenharia de Software, o trabalho apresenta um *framework* baseado nas metodologias ativas PBL (Aprendizagem Baseada em Projetos, do inglês *Project Based Learning*) e *role-play*. Neste contexto, ênfase foi dada para as seis principais *soft skills* almejadas para engenheiros de software: comunicação, trabalho em equipe, organização, liderança, aprendizagem e criatividade. Como complemento aos elementos motivacionais, o *framework*, denominado FraSSD – *Framework for Soft Skills Development*, utiliza alguns elementos da gamificação, a saber: competição, pontuação e placar. Por fim, as atividades estabelecidas para os alunos são baseadas na Taxonomia de Bloom, a fim de proporcionar desenvolvimento progressivo do conhecimento. Como prova de conceito, o FraSSD foi aplicado em uma turma de Engenharia de Software e foi realizada uma autoavaliação, por parte dos alunos, antes e após o desenvolvimento do projeto proposto pelo *framework*, na qual os estudantes avaliaram seu nível de desenvolvimento em cada uma das seis *soft skills* trabalhadas durante a disciplina. Os resultados foram comparados por meio de teste estatístico, mostrando que houve aumento no nível de desenvolvimento de todas as *soft skills* trabalhadas, com destaque para organização, criatividade e trabalho em equipe. Na visão do docente que aplicou o *framework*, o FraSSD forneceu um modelo fácil de ser instanciado e que ajudou na motivação dos alunos em contrapartida a métodos tradicionais de ensino.

Palavras-chave: *Soft skills*, Engenharia de Software, Aprendizagem ativa, *Framework*, Educação.

Abstract

A good *software* professional must have technical and non-technical skills, that is, hard and soft skills, to deal with the different challenges encountered throughout his career. For this to be possible, he must develop such skills since graduation. In this context, although there are studies in the literature, a model has not been found that helps educators develop soft skills in students during the classroom and includes motivational aspects in addition to active educational methodologies. Thus, to provide a model that assists educators in contemplating hard and soft skills in a more meaningful teaching-learning process for students, the work presents a framework based on active PBL (Project Based Learning) and role-play methodologies. In this context, the focus was on the most relevant soft skills for software engineers: communication, teamwork, organization, leadership, learning, and creativity. As a complement to the motivational elements, the framework FraSSD - Framework for Soft Skills Development uses some elements of gamification: competition, punctuating, and scoreboard. Finally, the student's activities are based on Bloom's Taxonomy to provide progressive knowledge development. As a proof of concept, we applied the FraSSD in a Software Engineering class and carried out a self-assessment by the students before and after the development of the project proposed by the framework, in which the students evaluated their level of development in each one of the six soft skills worked on during the course. We used a statistical test to compare the results, which showed an increase in the level of development of all the soft skills worked on, emphasizing the soft skills of organization, creativity, and teamwork. For educators who applied the FraSSD, it provided an easy model to instantiate, and that helped motivate students as opposed to traditional teaching methods.

Keywords: Soft skills, Software Engineering, Active learning, Framework, Education

Lista de Ilustrações

Figuras

Figura 1: Processo de Revisão Sistemática da Literatura.....	21
Figura 2: Citação das soft skills nos trabalhos analisados (%).....	23
Figura 3: Taxonomia de Bloom Original e Revisada no Domínio dos Processos Cognitivos.....	40
Figura 4: Fluxograma para escolha do teste estatístico para duas amostras pareadas.....	42
Figura 5: Estruturação do FraSSD.....	44
Figura 6: Atividades do FraSSD na visão do aluno.....	47
Figura 7: Placar final com notas das tarefas	65
Figura 8: Placar com notas finais do projeto na disciplina.....	66
Figura 9: Gráfico do perfil dos alunos quanto às <i>soft skills</i>	67
Figura 10: Gráfico do perfil dos alunos quanto ao PBL.....	67
Figura 11: Gráfico do perfil dos alunos quanto a experiência profissional.....	68
Figura 12: Metodologia que causa maior motivação nos estudos, segundo a visão dos alunos	71

Quadros

Quadro 1: Atividades da Etapa de Contextualização	49
Quadro 2: Atividades da Etapa de Projeto.....	51
Quadro 3: Atividades da Etapa de Métricas	53
Quadro 4: Atividades da Etapa de Avaliação	55
Quadro 5: Instanciação do FraSSD	57

Lista de Tabelas

Tabela 1: <i>Strings</i> de busca RSL.....	20
Tabela 2: Filtros aplicados nas bases de dados durante a RSL	20
Tabela 3: Critérios de inclusão e exclusão	21
Tabela 4: Relação veículo de publicação X artigos selecionados na RSL	22
Tabela 5: Metodologias de ensino para desenvolver Soft Skills	33
Tabela 6: Taxonomia de Bloom nas duas dimensões.....	40
Tabela 7: Interpretação dos valores dos coeficientes de tamanho de efeito d e r	43
Tabela 8: Atividades do FraSSD na visão do aluno em relação à Taxonomia de Bloom revisada.....	48
Tabela 9: Análise do desenvolvimento das soft skills antes e após a aplicação do FraSSD segundo a percepção dos alunos.....	68

Lista de Abreviaturas e Siglas

AVW	<i>Active Video Watching</i>
CDIO	<i>Conceive-Design-Implement-Operate</i>
FraSSD	<i>Framework for Soft Skills Development</i>
GBL	<i>Game-Based Learning</i>
GDBL	<i>Game Development Based Learning</i>
PBL	<i>Project Based Learning</i>
PrBL	<i>Problem Based Learning</i>
ePrBL	<i>e-learning Problem Based Learning</i>
RSL	Revisão Sistemática da Literatura

Sumário

Capítulo 1 – Introdução	13
1.1 Considerações iniciais	13
1.2 Motivação e Justificativa	15
1.3 Objetivo	16
1.4 Metodologia.....	16
1.5 Estrutura do documento.....	17
Capítulo 2 – Levantamento Bibliográfico	18
2.1 Considerações iniciais	18
2.2 Revisão sistemática.....	18
2.3 Resultados da RSL.....	22
2.3.1 Q1: Quais são as <i>soft skills</i> almejadas para Engenheiros de Software?	22
2.3.2 Q2: Quais são as metodologias de ensino utilizadas para o desenvolvimento de <i>soft skills</i> em estudantes de graduação, mantendo-os motivados?	28
2.4 Avaliação das <i>soft skills</i>	36
2.4.1 Taxonomia de Bloom	38
2.4.2 Testes estatísticos.....	40
Capítulo 3 – <i>Framework for Soft Skills Development</i> (FraSSD)	44
3.1 Considerações iniciais	44
3.2 Premissas	44
3.3 Atores.....	45
3.4 Descrição do <i>framework</i>	46
3.4.1 Atividades do FraSSD na visão do aluno	46
3.4.2 Atividades do FraSSD na visão do professor	48
3.4.2.1 Contextualização.....	48
3.4.2.2 Projeto.....	51
3.4.2.3 Métricas	53
3.4.2.4 Avaliação	55
Capítulo 4 – Resultados e discussão	57
4.1 Considerações iniciais	57
4.2 Instanciação do <i>framework</i>	57
4.3 Análise dos resultados	66
4.4 Avaliação do FraSSD	69
Capítulo 5 – Conclusões	72

5.1	Considerações iniciais	72
5.2	Contribuições	72
5.3	Publicação	74
5.4	Limitações	74
5.5	Trabalhos futuros	74
Referências		76
Apêndice A - Trabalhos selecionados na RSL		83
Apêndice B - <i>Soft skills</i> para Engenheiros de Software		87
Apêndice C – FraSSD		94
Apêndice D – Termo de Consentimento Livre e Esclarecido		95
Apêndice E – Formulário inicial de avaliação quanto ao nível das <i>soft skills</i>		97
Apêndice F – Formulário final de avaliação quanto ao nível das <i>soft skills</i>		98
Apêndice G – Formulário de avaliação pelo professor quanto ao FraSSD		99

Capítulo 1 – Introdução

1.1 Considerações iniciais

Atualmente, o profissional de *software* deve possuir não apenas habilidades técnicas, mas também as não técnicas, denominadas *soft skills*. Estas englobam habilidades de trabalho em equipe, resolução de problemas, comunicação, pensamento crítico, liderança, iniciativa, dentre outras (CAEIRO-RODRÍGUEZ et al., 2021; DEEP et al., 2019). Além disso, as *soft skills* podem ser consideradas como um complemento das *hard skills*, isto é, as habilidades técnicas de um indivíduo, sendo, portanto, importante que um profissional envolvido em um processo de *software* possua ambas (AHMED et al., 2012).

Não existe um consenso na literatura quanto a definição formal do que são as *soft skills*. Além disso, são encontradas muitas variações de como as *soft skills* são nomeadas, como, por exemplo, habilidades do século 21, habilidades gerais, habilidades transversais, dentre outras, sendo que novas *soft skills* são continuamente sugeridas. Isso cria confusão, especialmente para comparar e conseguir separar as *soft skills* em diferentes áreas (CAEIRO-RODRÍGUEZ et al., 2021). Também são encontradas variações de nomenclatura de acordo com a região. Por exemplo, na Nova Zelândia é mais comum o termo “habilidades essenciais” (*essential skills*); na Austrália, competências-chave (*key competencies*), habilidades sociais (*soft skills*) e habilidades para empregabilidade (*employability skills*) se mostram mais frequentes; competências necessárias (*necessary skills*), competências para empregabilidade (*employability skills*) ou competências no local de trabalho (*workplace know-how*) podem ser encontradas em trabalhos de origem nos Estados Unidos; por fim, competências-chave (*key skills*) ou competências essenciais (*core skills*) aparecem no contexto do Reino Unido (DEEP et al., 2019).

Matteson et al. (2016), após análise da literatura com o intuito de definir o termo “*soft skill*”, afirmam que estas referem-se à capacidade de acessar determinado conhecimento e utilizá-lo para realizar uma ação ou tarefa.

É possível perceber, pelas nomenclaturas presentes na literatura, que as *soft skills* encontram-se atreladas ao sucesso profissional de uma pessoa. No contexto do Engenheiro de Software existe uma lacuna entre o que é trabalhado durante a graduação e as habilidades necessárias no mundo profissional (GAROUSI et al., 2019). No local de trabalho, diante das rápidas mudanças atualmente, as credenciais acadêmicas e as habilidades técnicas não são mais suficientes para o sucesso, enquanto as *soft skills* são cada vez mais consideradas

impulsionadoras do sucesso nas carreiras profissionais (TAYLOR, 2019). Diante disso, Taylor (2019) afirma que as instituições de ensino superior devem responder a essa demanda com iniciativas curriculares que incluam as *soft skills* a fim de auxiliar no sucesso profissional e pessoal dos alunos.

Diversas propostas podem ser encontradas na literatura para desenvolver *soft skills* em alunos de graduação. Um ponto em comum parece ser o foco em proporcionar um ambiente realista aos estudantes (KHAKUREL e PORRAS, 2020), isto é, fazer com que os alunos tenham contato com situações que podem realmente acontecer durante sua vida profissional. Além disso, segundo Khodeir e Nessim (2020), deve-se focar em trabalhar algumas *soft skills* específicas em alunos de graduação, escolhendo as de maior urgência ou relevância, e não tentar abranger todas as existentes.

Deep et al. (2019) afirmam que os professores que desejam ajudar os alunos a desenvolver *soft skills* devem identificar as habilidades-alvo que querem trabalhar com os alunos e adotar pedagogias e estratégias instrucionais testadas que promovam e desenvolvam essas habilidades. Em complemento, Lev Semionovitch Vigotski, importante psicólogo da área do desenvolvimento educacional, afirma que o professor universitário deve ser o agente mediador deste processo, propondo desafios aos alunos, ajudando-os a resolvê-los, ou proporcionando atividades em grupo para apoio mútuo (VIGOTSKY, 1988 apud FERNANDES et al., 2020).

Nesse âmbito, Capretz et al. (2017) afirmam que é impossível excluir as *soft skills* dos processos de *software*, pois um *software* é desenvolvido por pessoas e para pessoas e, assim, envolve fatores humanos. Ahmed et al. (2012) complementam com o fato de que o processo de desenvolvimento de *software* contém diferentes atividades, como análise do sistema, *design*, codificação e teste, o que requer não apenas habilidades técnicas diferentes, mas também personalidades distintas e, dessa forma, as *soft skills* podem ser ainda mais importantes do que as *hard skills* em algumas situações.

Diante deste cenário, com o intuito de evidenciar a relevância do presente trabalho, na seção 1.2 são apresentadas a motivação e justificativa para desenvolvimento do trabalho, enquanto o objetivo é apresentado na seção 1.3. Em seguida, a metodologia adotada é apresentada na seção 1.4 e, por fim, na seção 1.5 é explicitada a estrutura adotada no presente documento.

1.2 Motivação e Justificativa

Trabalhos têm surgido salientando a importância do desenvolvimento das *soft skills* em estudantes de graduação (STRAY et al., 2021; GROENEVELD et al., 2021; HAZZAN, 2021; BYRNE et al., 2020; FIORAVANTI et al., 2020; GAROUSI et al., 2019; VAN LAAR et al., 2017). Percebe-se, nesse contexto, o uso de metodologias ativas, isto é, com o foco no aluno (YOUNIS et al., 2021; KUPPUSWAMY e MHAKURE, 2020; CAEIRO-RODRÍGUEZ et al., 2021; KONINGS e LEGG, 2020; DEEP et al., 2019; YOUNIS et al., 2019; GONZÁLEZ-MORALES et al., 2011; PÉREZ e RUBIO, 2020; BARROS e BITTENCOURT, 2018). Uma forte razão para isso é que essa abordagem auxilia na motivação dos alunos, pois, ao buscarem por suas próprias soluções de forma ativa, os alunos ficam mais engajados e motivados a alcançar o objetivo proposto, construindo um aprendizado mais significativo.

Ademais, é possível notar um interesse crescente na Europa quanto ao desenvolvimento de *soft skills* no contexto da disciplina de Engenharia de Software, buscando inclui-las em seus currículos educacionais (CAEIRO-RODRÍGUEZ et al., 2021). Esta é uma disciplina orientada para a indústria (BASTARRICA et al., 2017), o que faz com que os alunos devam aprender a lidar não apenas com problemas técnicos, mas também serem capazes de, por exemplo, gerenciar e planejar projetos, recursos humanos e orçamentos; serem cooperativos e capazes de se comunicarem bem com a equipe e os *stakeholders*; além de possuírem mentes criativas para lidar com as intensas mudanças da tecnologia e cenário de um projeto (IBRAHIM e ABD HALIM, 2014).

No panorama do Brasil também é possível encontrar alguns trabalhos voltados para os estudantes de graduação em disciplinas de Engenharia de Software. Estes utilizam metodologias ativas, como aprendizagem baseada em times (RAMOS et al., 2018) e aprendizagem baseada em projetos (FIORAVANTI et al., 2020; BARROS e BITTENCOURT, 2018; RIBEIRO e BITTENCOURT, 2018). Entretanto, nos estudos realizados para embasar o presente trabalho, destaca-se que não foi encontrada uma proposta que pudesse auxiliar os docentes a incluírem o desenvolvimento de *soft skills* de forma efetiva nas disciplinas sob suas responsabilidades visando ajudar os alunos a se tornarem profissionais mais completos.

Portanto, considerando a necessidade de se investir, cada vez mais, no desenvolvimento das *soft skills* necessárias a futuros profissionais de *software* para que estes exerçam suas tarefas com eficiência e qualidade, esse trabalho propõe o FraSSD – *Framework for Soft Skills Development*. O FraSSD contribui para preencher algumas lacunas observadas na literatura como: auxiliar os docentes para aperfeiçoamento das *soft skills* dos alunos em sala de aula, bem

como para a aplicação de metodologias ativas e técnicas para manter os alunos motivados a desenvolver as *hard* e *soft skills*; reunir e organizar *soft skills* que apresentam diferentes termos, porém com definição equivalente, para sanar a divergência de nomenclatura presente na literatura.

1.3 Objetivo

Diante do exposto, o presente trabalho tem como objetivo geral auxiliar no desenvolvimento de *soft skills* relevantes a futuros profissionais de *software* ainda durante a graduação. O intuito é proporcionar um complemento à formação técnica recebida em disciplinas de Engenharia de Software. Para isso, como objetivo específico, tem-se a estruturação de um *framework* denominado FraSSD, o qual é embasado na metodologia ativa Aprendizagem Baseada em Projetos (PBL), alinhada à técnica do *role-play* e a elementos motivacionais da gamificação. Espera-se contribuir para uma formação mais completa dos futuros profissionais de *software*, proporcionando um modelo que auxilie os docentes no desafio de contemplar *hard* e *soft skills* em um processo de ensino-aprendizagem mais significativo para os estudantes, além de apresentar um agrupamento das *soft skills* com semânticas semelhantes e nomenclaturas distintas.

1.4 Metodologia

Como primeiro passo para a estruturação do *framework*, realizou-se uma Revisão Sistemática da Literatura (RSL). O intuito foi definir as principais *soft skills* almejadas para profissionais de *software*, bem como identificar quais métodos de ensino poderiam trazer maior motivação aos alunos e promover o processo de ensino-aprendizagem de forma efetiva.

Após isso, foi estruturado o *framework* baseado nos métodos de ensino PBL (YOUNIS et al., 2021; KUPPUSWAMY e MHAKURE, 2020; KONINGS e LEGG, 2020) e *role-play* (MAXIM et al., 2017; PÉREZ e RUBIO, 2020), os quais mostraram-se mais eficazes na literatura para desenvolver *soft skills* em estudantes matriculados em disciplina de Engenharia de Software. Estes métodos foram alinhados com aspectos da gamificação (MAXIM et al., 2017; MAXIM et al., 2019; SOUZA et al., 2018) como elementos motivacionais, a saber: competição, pontuação e placar. Além disso, a fim de auxiliar os alunos em um aprendizado gradual, as atividades propostas aos alunos no FraSSD são baseadas na Taxonomia de Bloom (SOBRAL, 2021; FOREHAND, 2010).

Em seguida, o *framework* foi instanciado e aplicado no contexto de uma disciplina semestral de Engenharia de Software ministrada para estudantes universitários vinculados a um curso de Bacharelado em Ciência da Computação. Para análise de adequação da proposta, os passos do *framework* foram seguidos por professor e alunos no decorrer da disciplina. As percepções dos alunos e do professor sobre as *soft skills* antes da intervenção do *framework* e após sua aplicação foram coletadas para avaliação.

Por fim, os dados coletados foram analisados por meio de formulários e teste estatístico, tirando-se conclusões sobre o desenvolvimento das *soft skills* com a aplicação do FraSSD. A partir das percepções coletadas, foi possível propor melhorias quanto a estruturação e aplicação do *framework*.

1.5 Estrutura do documento

Além do presente, este documento está estruturado em mais quatro capítulos, conforme explicitado a seguir.

No capítulo 2 é apresentado o levantamento bibliográfico realizado por meio de uma RSL, contemplando dois tópicos principais: *soft skills* e metodologias de ensino. O primeiro tópico dispõe as principais *soft skills* para profissionais de *software*, mostrando um *ranking* das mais citadas na literatura, bem como a definição de cada uma. O segundo apresenta as metodologias de ensino que podem ser utilizadas para desenvolver *soft skills* em estudantes de graduação e que os mantenha motivados no processo. Por fim, um tópico de discussão sobre estratégias de avaliação do desenvolvimento das *soft skills* nos alunos é apresentado.

No capítulo 3 é apresentada a definição do *framework* proposto, detalhando suas etapas de forma a auxiliar na sua efetiva aplicação por docentes.

No capítulo 4 é apresentada a instanciación do *framework*, com ênfase nas análises dos dados coletados, discutindo-se os resultados.

Por fim, no capítulo 5 são apresentadas as considerações finais, destacando as contribuições e limitações do presente trabalho, bem como propostas para projetos futuros.

Capítulo 2 – Levantamento Bibliográfico

2.1 Considerações iniciais

A identificação do estado da arte envolvendo os principais tópicos abordados no presente trabalho é necessária para garantir que esforços sejam voltados para o avanço das pesquisas em temas relevantes e atuais.

Neste contexto, a metodologia utilizada para o levantamento bibliográfico é descrita na seção 2.2, a qual foi realizada através de uma Revisão Sistemática da Literatura (RSL). Na seção 2.3 são apresentados os resultados da RSL em duas subseções, as quais respondem às duas questões de pesquisa formuladas, sendo uma delas referente às *soft skills* apresentadas na literatura como relevantes para futuros Engenheiros de Software e, a outra, referente às metodologias de ensino que podem auxiliar no desenvolvimento de *soft skills* em alunos de graduação a fim de mantê-los motivados. Por fim, na seção 2.4 é apresentada uma discussão sobre formas de avaliação do desenvolvimento das *soft skills*, detalhando o uso da Taxonomia de Bloom como desenvolvimento gradual do conhecimento e a autoavaliação das *soft skills* por meio de formulários, tendo sua verificação realizada com testes estatísticos.

2.2 Revisão sistemática

Para realizar o levantamento bibliográfico, efetuou-se uma RSL baseada no método descrito por Felizardo et al. (2017). Este consiste, como passo inicial, na definição das *strings* de busca, das bases de dados, das questões de pesquisa e dos critérios de inclusão e exclusão; após isso, as *strings* de busca são aplicadas nas bases de dados selecionadas e tem-se a primeira seleção de trabalhos, lendo-se o título e resumo destes, separando-os de acordo com os critérios de inclusão e exclusão; por fim, realiza-se uma segunda seleção, desta vez com leitura completa dos trabalhos, tendo como critério de seleção, novamente, os critérios de inclusão e exclusão definidos.

Para entender quais *soft skills* são mais citadas na literatura, buscando aquelas consideradas mais relevantes para futuros Engenheiros de Software, foi elaborada a questão de pesquisa Q1: Quais são as *soft skills* almejadas para os Engenheiros de Software?

Além disso, buscando entender quais metodologias de ensino podem ser utilizadas para auxiliar no desenvolvimento de *soft skills* em alunos de graduação, elaborou-se a questão de pesquisa Q2: Quais são as metodologias de ensino utilizadas para o desenvolvimento de *soft*

skills em estudantes de graduação, mantendo-os motivados? É válido destacar que foi enfatizada a motivação devido a diferentes autores (ANGELI et al., 2020; BRITO et al., 2020; GAROUSI et al., 2019; SOUZA et al., 2018; MAXIM et al., 2017; PINTO e SILVA, 2017) abordarem a importância de manter os alunos motivados para garantir um processo de ensino-aprendizagem efetivo.

A partir disso, com o intuito de contemplar os principais resultados da área, definiu-se seis bases de dados distintas para serem realizadas as buscas. São estas: *IEEE Xplore*; *ScienceDirect*; *SpringerLink*; *ACM Digital Library*; Portal Periódicos CAPES e *Google Scholar*.

Na sequência, a fim de identificar os trabalhos mais relevantes, foram definidas as *strings* de busca formadas com as palavras-chave “*soft skills*”, “habilidades não técnicas”, “habilidades pessoais”, “habilidades transferíveis”, “profissionais de *software*”, “engenheiros de *software*”, “ensino”, “aprendizagem”, “desenvolvimento” e “motivação”, sendo aplicadas com operadores lógicos OR e AND em português e inglês. Além disso, alguns filtros foram utilizados de acordo com a disponibilidade em cada base de dados, também em português e inglês. Na Tabela 1 é ilustrado como foram montadas as *strings* de busca para cada questão de pesquisa em português e inglês, enquanto na Tabela 2 são exibidos os filtros aplicados em cada base de dados. Destaca-se que a questão Q2 gerou duas *strings* de busca a fim de entender quais metodologias de ensino podem auxiliar no desenvolvimento de *soft skills*, bem como motivar os alunos.

Por fim, os critérios de inclusão e exclusão definidos são descritos na Tabela 3. Para obter os resultados mais recentes e relevantes, definiu-se que aqueles com data anterior a 2017 seriam analisados apenas se tivessem mais de 15 citações. Para verificar o número de citações, foram consideradas as métricas do Google Scholar.

Concluída a etapa de definições, foi executada a busca pelas *strings* nas bases de dados selecionadas. Como as bases de dados retornavam, muitas vezes, milhares de resultados, optou-se por aplicar os critérios de inclusão e exclusão apenas nos 100 (cem) resultados mais relevantes. Para tanto, foi usada a ordenação por relevância disponível nas respectivas bases. Isso foi necessário devido a disponibilidade do autor.

Assim, apenas os metadados dos artigos selecionados pelos critérios de inclusão e exclusão por meio da leitura de seus títulos e resumos foram organizados em uma planilha. Esta continha o título, autores, ano, base de dados de origem, *string* de busca e questão de pesquisa possivelmente respondida por esses artigos. Além disso, para facilitar a organização dos

resultados da revisão sistemática, foram eliminadas as duplicatas para gerar uma lista limpa, conforme aplicado por Assyne et al. (2022). Essa fase resultou em 94 artigos selecionados.

Tabela 1: Strings de busca RSL

Questão de pesquisa	String de busca		
	Sn	Inglês	Português
Q1	S1	(Soft skills OR non-technical skills OR personal skills OR transferable skills) AND (software professionals OR software engineers)	(Soft skills OR habilidades não técnicas OR habilidades pessoais OR habilidades transferíveis) AND (profissionais de software OR engenheiros de software)
Q2	S2	(Soft skills OR non-technical skills OR personal skills OR transferable skills) AND (teaching OR learning OR development)	(Soft skills OR habilidades não técnicas OR habilidade pessoais OR habilidades transferíveis) AND (ensino OR aprendizagem OR desenvolvimento)
	S3	(Soft skills OR non-technical skills OR personal skills OR transferable skills) AND (teaching OR learning OR development) AND (motivation)	(Soft skills OR habilidades não técnicas OR habilidade pessoais OR habilidades transferíveis) AND (ensino OR aprendizagem OR desenvolvimento) AND (motivação)

Fonte: Desenvolvido pelo autor.

Tabela 2: Filtros aplicados nas bases de dados durante a RSL

Basse de dados	Filtros aplicados
<i>IEEE Xplore</i>	Publication topics: educational courses; software engineering; teaching; educational institutions; computer science education; engineering education; software development management*
<i>ScienceDirect</i>	Subject areas: computer science; engineering
<i>SpringerLink</i>	Content type: article Discipline: Computer science, education
<i>ACM</i>	-
Portal Periódico CAPES	-
<i>Google Scholar</i>	-

* Aplicado somente em S1, conforme disponibilidade da base de dados.

Fonte: Desenvolvido pelo autor.

Após isso, realizou-se a leitura dos artigos com reaplicação dos critérios de inclusão e exclusão, obtendo-se os 56 artigos selecionados para o estudo, dos quais 13 responderam a Q1, 33 responderam a Q2 e 10 responderam às duas questões de pesquisa. Um esquema do processo

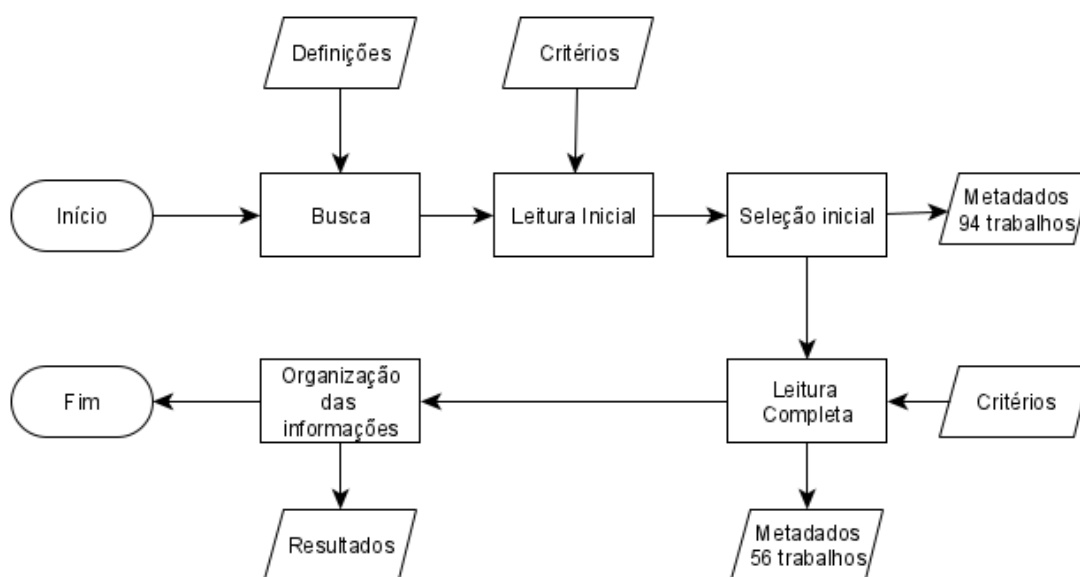
seguido na RSL pode ser visto na Figura 1, enquanto na Tabela 4 é mostrada a relação entre o número de artigos selecionados e as respectivas bases de dados. No Apêndice A é mostrada a lista completa dos artigos selecionados, detalhando quais questões de pesquisa cada um responde.

Tabela 3: Critérios de inclusão e exclusão

Critérios	
Inclusão	1 – Responde diretamente a uma das questões de pesquisa. 2 – O trabalho está escrito em inglês ou português. 3 – O trabalho apresenta metodologia e resultados bem definidos. 4 – Data de publicação posterior a 2017 ou, caso anterior, apresenta mais de 15 citações.
Exclusão	1 – Apresenta <i>soft skills</i> para outros tipos de profissionais ou contextos. 2 – O trabalho não está escrito em português ou inglês. 3 – Não foi possível ter acesso ao trabalho na íntegra.

Fonte: Desenvolvido pelo autor.

Figura 1: Processo de Revisão Sistemática da Literatura



Fonte: Desenvolvido pelo autor.

Por fim, observa-se que o processo de busca e leitura para a seleção inicial dos trabalhos foi realizada em agosto de 2021, enquanto a leitura completa dos trabalhos, bem como organização dos resultados, foi realizada entre setembro de 2021 e junho de 2022.

Tabela 4: Relação veículo de publicação X artigos selecionados na RSL

Base de dados	Artigos selecionados
IEEE Xplore	29
ScienceDirect	8
SpringerLink	2
ACM Digital Library	10
CAPES Periodicals Portal	5
Google Scholar	2
Total	56

Fonte: Desenvolvido pelo autor.

2.3 Resultados da RSL

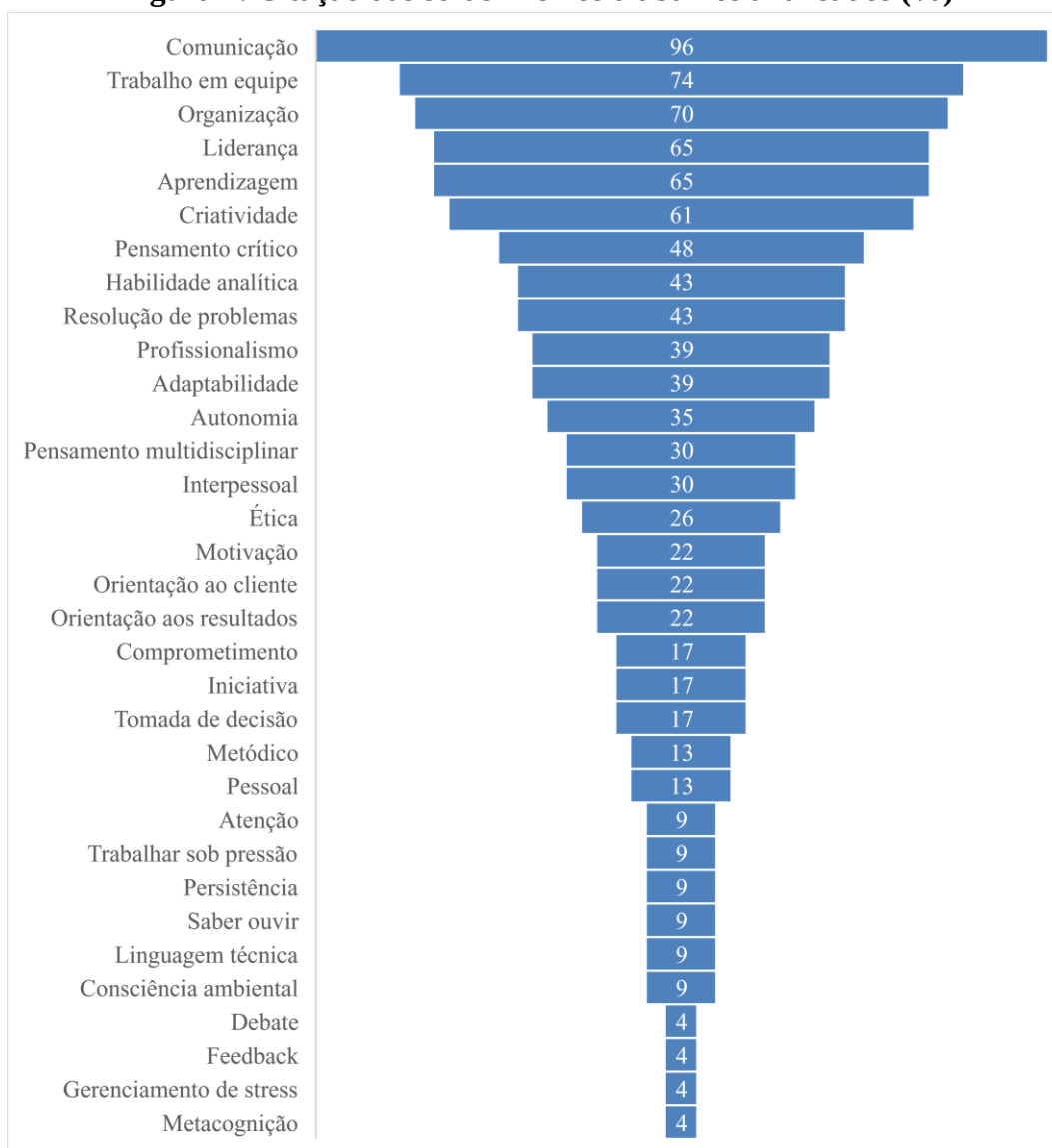
Essa seção foi organizada a partir dos trabalhos selecionados na RSL a fim de responder as duas questões de pesquisa formuladas. Assim, nas seções 2.3.1 e 2.3.2 são apresentados os resultados referentes a Q1 e Q2, respectivamente.

2.3.1 Q1: Quais são as *soft skills* almejadas para Engenheiros de Software?

Dos 56 artigos selecionados, 23 apresentaram resultados referentes às *soft skills*. Entretanto, alguns autores atribuem nomenclaturas diferentes a *soft skills* com semântica semelhante. Assim, nesse trabalho, um agrupamento das *soft skills* foi realizado, totalizando 33 *soft skills*, cuja frequência de citação na literatura foi calculada, conforme exibido na Figura 2.

Com isso, é possível perceber que a comunicação se encontra como a principal *soft skill* para um Engenheiro de Software, sendo citada em 96% dos trabalhos que tratam do tema, a qual envolve formas de comunicação oral e escrita. Em seguida, trabalho em equipe, mostra-se presente em 74% dos trabalhos. Isso mostra que a ideia de um profissional da área da tecnologia lidando apenas consigo e com as máquinas é equivocada e não se adequa mais ao contexto atual. Outras *soft skills* de destaque são organização, a qual está presente em 70% dos estudos, liderança e aprendizagem, presentes em 65% e, finalmente, criatividade, presente em 61% dos trabalhos analisados. Estas merecem atenção dos docentes para buscar desenvolvê-las em seus alunos desde a graduação.

Um maior detalhamento pode ser visto no Apêndice B, por meio dos grupos definidos no presente trabalho para as *soft skills* identificadas, ordenados pela frequência que são citados na literatura.

Figura 2: Citação das soft skills nos trabalhos analisados (%)

Fonte: Desenvolvido pelo autor.

Buscou-se também, além de identificar, definir as *soft skills* encontradas, a fim de trazer maior entendimento sobre o que cada uma delas abrange em relação às habilidades almejadas para um profissional de *software*. Dentre os 23 trabalhos em que foram encontradas citações de *soft skills*, apenas 5 deles apresentam definições. São estes: Matturro et al., 2019; Taylor, 2019; Van Laar et al., 2017; Ahmed et al., 2012 e González-Morales et al., 2011. As definições encontradas referem-se a 23 das 33 *soft skills* listadas, as quais são descritas a seguir.

- **Comunicação:** capacidade de transmitir informações para que sejam bem recebidas e compreendidas (MATTURRO et al., 2019; AHMED et al., 2012); comunicar-se oralmente e por escrito de forma simples, concisa, inequívoca e de fácil compreensão (MATTURRO et al., 2019); habilidades de fala e escrita (TAYLOR,

2019); habilidade de transmitir informações a outras pessoas, garantindo que o significado seja expresso de forma eficaz (VAN LAAR et al., 2017); capacidade de reportar as informações utilizando a escrita de forma clara (GONZÁLEZ-MORALES et al., 2011);

- **Trabalho em equipe:** habilidade de um indivíduo trabalhar próximo a outras pessoas; trabalhar efetivamente em um time e contribuir para os objetivos buscados; habilidade de cooperar com outras pessoas durante o trabalho (MATTURRO et al., 2019); capacidade de trabalhar de forma efetiva em um time (TAYLOR, 2019); habilidade para trabalhar em equipe de forma a trocar informações, negociar acordos e tomar decisões com respeito mútuo para alcançar um objetivo comum (VAN LAAR et al., 2017); alguém que pode trabalhar efetivamente em um ambiente de equipe e contribuir para a meta desejada (AHMED et al., 2012);
- **Organização:** capacidade de classificar, priorizar e controlar a execução de suas tarefas de acordo com o plano e os recursos sob sua responsabilidade; capacidade de um indivíduo de avaliar e priorizar tarefas e garantir que sejam concluídas em tempo hábil (MATTURRO et al., 2019); capacidade de manter-se dentro do cronograma (TAYLOR, 2019); capacidade de gerenciar com eficiência várias tarefas e manter o cronograma sem desperdiçar recursos (AHMED et al., 2012);
- **Liderança:** capacidade de liderar e motivar os outros (TAYLOR, 2019); ser capaz de organizar, planejar e direcionar as atividades necessárias para alcançar os objetivos, além de repassar os resultados (GONZÁLEZ-MORALES et al., 2011); saber gerenciar as tarefas do time (GONZÁLEZ-MORALES et al., 2011);
- **Aprendizagem:** capacidade de se adaptar a novas tarefas, funções ou desafios de forma eficaz e com facilidade; ter interesse em aprender e ter a capacidade de autoaprendizagem em pouco tempo (MATTURRO et al., 2019); ter vontade de aprender coisas novas e buscar por isso (TAYLOR, 2019); habilidade para explorar constantemente novas informações que possam ser integradas em um ambiente para melhorar continuamente as próprias capacidades (VAN LAAR et al., 2017); capacidade de aprender novos conceitos, metodologias e tecnologias em um período relativamente curto (AHMED et al., 2012);
- **Criatividade:** capacidade de identificar e criar ideias e oportunidades; capacidade de produzir ou propor soluções imaginativas e práticas para problemas de negócios; ter pensamento criativo para apresentar novas ideias (MATTURRO et al., 2019);

habilidade de gerar ideias novas ou anteriormente desconhecidas, ou tratar ideias familiares de uma nova maneira e transformar essas ideias em um produto, serviço ou processo que seja reconhecido como novo dentro de um domínio específico (VAN LAAR et al., 2017); a capacidade de apresentar soluções novas e criativas (AHMED et al., 2012);

- **Pensamento crítico:** habilidade de determinar, de forma cuidadosa e deliberativa, a aceitação ou rejeição de uma informação (MATTURRO et al., 2019); capacidade de alguém utilizar conhecimento e habilidade para guiar o seu comportamento (TAYLOR, 2019); habilidade para fazer escolhas baseadas em informações obtidas usando raciocínio reflexivo e evidências suficientes para apoiar as alegações (VAN LAAR et al., 2017); capacidade de analisar criticamente o trabalho e o seu próprio desempenho (GONZÁLEZ-MORALES et al., 2011);
- **Habilidade analítica:** capacidade de compreender e explicar cada parte de um todo; capacidade de decompor uma situação em pequenas partes, reconhecer o que precisa ser feito e planejar um curso de ação adequado para cada passo (MATTURRO et al., 2019); capacidade de tomar decisões sensatas com base nas informações disponíveis (AHMED et al., 2012);
- **Resolução de problemas:** capacidade de resolver conflitos de interesse em situações de trabalho; habilidade de entender, articular e resolver problemas complexos; habilidade de pensar logicamente, analisar e resolver problemas (MATTURRO et al., 2019); capacidade de resolver e lidar com problemas (TAYLOR, 2019); capacidade de processar e entender cognitivamente uma situação-problema para encontrar uma solução por meio de uso ativo do conhecimento (VAN LAAR et al., 2017); habilidade de lidar com problemas que apareçam, seja com o projeto ou com as pessoas envolvidas e, mais que isso, evitar que estes problemas ocorram (GONZÁLEZ-MORALES et al., 2011);
- **Profissionalismo:** comportar-se de forma profissional em todas as situações; capacidade de negociar com as pessoas para encontrar uma solução que beneficie ambas as partes (TAYLOR, 2019);
- **Adaptabilidade:** capacidade de se adaptar e trabalhar de forma eficaz com diferentes situações em face à mudança (MATTURRO et al., 2019); ser flexível e adaptável às circunstâncias quando necessário (TAYLOR, 2019); habilidade para adaptar o pensamento, atitude ou comportamento a ambientes em mudança (VAN

LAAR et al., 2017); capacidade de aceitar e adaptar-se às mudanças ao realizar uma tarefa sem mostrar resistência (AHMED et al., 2012; MATTURRO et al., 2019);

- **Autonomia:** capacidade de governar-se por seus próprios meios; capacidade de concluir o trabalho de forma independente (MATTURRO et al., 2019); capacidade de gerenciar a si mesmo e se responsabilizar por seus atos (TAYLOR, 2019); habilidade para definir metas para si mesmo e gerenciar a progressão para atingir essas metas (VAN LAAR et al., 2017); capacidade de lidar com as tarefas com o mínimo de supervisão possível (AHMED et al., 2012; MATTURRO et al., 2019);
- **Pensamento multidisciplinar:** ser capaz de utilizar o conhecimento de diferentes disciplinas para resolver problemas complexos; facilidade de interagir com pessoas de outras culturas (TAYLOR, 2019); habilidade para mostrar compreensão cultural e respeitar outras culturas (VAN LAAR et al., 2017); capacidade de analisar diferentes fontes, classificando e sintetizando as informações relevantes (GONZÁLEZ-MORALES et al., 2011);
- **Interpessoal:** habilidade de se comportar de forma a aumentar a probabilidade de alcançar os objetivos (MATTURRO et al., 2019); ter bom relacionamento com as pessoas ao redor (TAYLOR, 2019); capacidade de lidar com outras pessoas por meio da comunicação social e interações em condições favoráveis e desfavoráveis (AHMED et al., 2012; MATTURRO et al., 2019);
- **Ética:** habilidade de seguir as regras e preceitos de valor, ordem e moralidade (MATTURRO et al., 2019); habilidade para se comportar de forma socialmente responsável, demonstrando consciência e conhecimento dos aspectos legais e éticos (VAN LAAR et al., 2017);
- **Orientação ao cliente:** habilidade de identificar e encontrar as necessidades dos clientes (MATTURRO et al., 2019); capacidade de maximizar a satisfação do cliente no ambiente de trabalho (TAYLOR, 2019); saber gerenciar os interesses do cliente durante todo o projeto (GONZÁLEZ-MORALES et al., 2011);
- **Orientação aos resultados:** capacidade de atingir e/ou exceder as metas de venda e/ou objetivos (MATTURRO et al., 2019); saber informar aos envolvidos no projeto, com diferente nível de detalhamento, os resultados obtidos (GONZÁLEZ-MORALES et al., 2011);
- **Comprometimento:** ser responsável com o trabalho (MATTURRO et al., 2019);

- **Iniciativa:** capacidade de propor e realizar ações sem a necessidade de que outros requisitem; habilidade de ser ativo e otimista ao encontrar trabalhos desafiadores (MATTURRO et al., 2019);
- **Tomada de decisão:** habilidade de avaliar as alternativas e tomar decisões apropriadas; habilidade de tomar decisões baseadas em informações (MATTURRO et al., 2019); habilidade de escolher entre soluções de forma racional (TAYLOR, 2019); capacidade de tomar decisões importantes sobre o desenvolvimento do projeto (GONZÁLEZ-MORALES et al., 2011);
- **Metódico:** habilidade de utilizar um conjunto de passos ordenados e definidos por um método ou técnica para resolver um problema em particular (MATTURRO et al., 2019);
- **Pessoal:** capacidade de controlar as emoções, ou, ao menos, as reações a elas (TAYLOR, 2019);
- **Saber ouvir:** capacidade de considerar o que o interlocutor está reportando (MATTURRO et al., 2019).

Após reunir as definições encontradas na literatura, buscou-se também estabelecer um entendimento comum para as outras 10 *soft skills*, por meio da formulação de uma definição com base na percepção pessoal obtida com a leitura das referências, listadas no Apêndice B, que as abordavam. São elas:

- **Motivação:** habilidade de uma pessoa de permanecer interessada e engajada na tarefa que se propôs a cumprir, de forma que isso a auxilie a realizar o melhor trabalho possível;
- **Atenção:** envolve a capacidade de concentração durante a realização de determinada tarefa a fim de cumpri-la corretamente;
- **Trabalhar sob pressão:** habilidade de conseguir cumprir os objetivos definidos mesmo em situações desfavoráveis;
- **Persistência:** habilidade de permanecer envolvido em alguma atividade específica para buscar pelo resultado desejado, mesmo diante de dificuldades;
- **Linguagem técnica:** ter conhecimento da linguagem técnica apropriada de determinada área e saber utilizá-la quando for conveniente para a situação;
- **Consciência ambiental:** atentar-se para as questões ambientais, analisando quais os impactos ambientais de suas ações e escolhas;

- **Debate:** habilidade de debater determinado assunto, expondo uma ideia ou opinião de forma consistente e respeitosa;
- **Feedback:** habilidade de fornecer retornos do que foi feito em um projeto ou atividade de forma a acrescentar valor ao time e a clientes, além da capacidade de receber retornos e aplicá-los para aprimorar suas habilidades;
- **Gerenciamento de estresse:** habilidade de manter equilíbrio emocional diante de situações que possam ser estressantes no dia a dia;
- **Metacognição:** capacidade de ter consciência de seus atos e pensamentos e, assim, saber administrá-los na rotina de trabalho.

Uma vez definidas as *soft skills*, é possível perceber ainda mais o porquê estas são tão importantes para o futuro profissional de *software* e, portanto, é interessante que sejam trabalhadas ainda na graduação. Para isso, deve-se atentar para estratégias e metodologias adequadas para tal, buscando o engajamento dos alunos, motivando-os a buscarem por um aprendizado abrangente e se tornarem profissionais melhor preparados.

2.3.2 Q2: Quais são as metodologias de ensino utilizadas para o desenvolvimento de *soft skills* em estudantes de graduação, mantendo-os motivados?

Educadores têm questionado a educação e buscado por estratégias para obter a atenção dos alunos e auxiliá-los na aprendizagem. A principal delas é a Aprendizagem Ativa (CASTRO e SIQUEIRA, 2017): um método instrucional que envolve os alunos em seu próprio processo de aprendizagem e, assim, o foco da aprendizagem muda dos professores para os alunos, trazendo maior responsabilidade e autonomia aos alunos (NICOLA et al., 2018). Essa abordagem de aprendizagem envolve permitir que o aluno desenvolva *hard* e *soft skills* em um ambiente criado pelo instrutor e não que receba as informações prontas desse instrutor (PAWAR et al., 2020). Algumas metodologias que seguem a aprendizagem ativa são a Aprendizagem Baseada em Projetos (PBL), a Aprendizagem Baseada em Problemas (PrBL), a aprendizagem envolvendo jogos, dentre outras. É válido ressaltar que a sigla PBL pode ser encontrada na literatura se referindo tanto à Aprendizagem Baseada em Projetos quanto à Aprendizagem Baseada em Problemas, porém, nesse trabalho, será seguido a nomenclatura definida.

Tanto o PBL quanto o PrBL seguem uma forma de Aprender Fazendo, do inglês *Learning by Doing* (ANGELI et al., 2020). O PBL é um modelo instrucional envolvente que fornece mais experiência de aprendizado e leva ao desenvolvimento cognitivo de alto nível por

meio do envolvimento dos alunos com projetos complexos (YOUNIS et al., 2021) e pode ser, além disso, baseado em problemas ou situações da vida real. O PrBL, por outro lado, tem sua origem no ensino da medicina e envolve os alunos na resolução de um problema para que estes possam desenvolver suas habilidades (PINTO e SILVA, 2017). Além disso, o PrBL pode estar alinhado ao uso de ferramentas de *e-learning*, isto é, ferramentas de ensino eletrônicas, como e-mails, plataformas de socialização, vídeos, internet, gestão de aprendizagem, dentre outras. Esta metodologia é conhecida como ePrBL (DEEP et al., 2019).

Em contrapartida às metodologias Aprender Fazendo, a qual engloba o PBL e o PrBL, existe a Aprender Ensinando. Esta propõe que os alunos podem desenvolver habilidades quando ensinam e participam do desenvolvimento das aulas. É mais comum no ensino de línguas estrangeiras, enquanto a Aprender Fazendo mostra-se a mais presente nas disciplinas de Ciência da Computação (ANGELI et al., 2020).

Quanto à aprendizagem envolvendo jogos, existem três principais métodos utilizados no ensino da Engenharia de Software descritos na literatura: Aprendizagem Baseada em Jogos, do inglês *Game-Based Learning* (GBL); Aprendizagem Baseada no Desenvolvimento de Jogos, do inglês *Game Development Based Learning* (GDBL) e a gamificação. O primeiro deles utiliza jogos para ensinar aos alunos algum tópico da disciplina; o segundo utiliza o desenvolvimento de jogos como meio de aprendizagem; o último e mais recente deles (desde 2011), utiliza aspectos dos jogos para desenvolver alguma habilidade ou característica nos alunos, buscando motivá-los (SOUZA et al., 2018; GARCIA et al., 2020).

Na análise realizada por Souza et al. (2018), notou-se que os elementos de jogos mais utilizados para a gamificação são pontuação, níveis e placares. Entretanto, os autores afirmam que não se deve pensar na gamificação como um simples “sistema de apontamento”, revelando que a competição é um importante princípio muito eficaz no engajamento das pessoas. Dessa forma, os pontos e os placares são eficazes quando usados para promover a competição entre os alunos.

Existe, ainda, uma abordagem que afirma que vídeo é uma ferramenta poderosa para aprendizagem, proporcionando maior motivação aos alunos ao assistir a tutoriais, por exemplo. Entretanto, somente assistir não é o suficiente, sendo necessário que os alunos possam interagir por meio de comentários, questões ou discussões sobre o vídeo. Esta abordagem é conhecida como Assistir a Vídeos de Forma Ativa, do inglês *Active Video Watching* (AVW) (GALSTER et al., 2018).

Pode-se, além disso, aplicar *frameworks* para obter uma aprendizagem ativa. Um deles encontrado na literatura é o CDIO - *Conceive-Design-Implement-Operate* (Conceber-Projetar-

Implementar-Operar) (KULKARNI et al., 2020). Este tem três objetivos principais para os alunos: 1. Dominar um conhecimento de trabalho mais profundo dos fundamentos técnicos; 2. Liderar a criação e operação de novos produtos e sistemas; 3. Entender a importância e o valor estratégico de seu trabalho.

Outras metodologias ativas podem ser encontradas na literatura, sendo que, muitas vezes, duas ou mais são combinadas para atingir propósitos específicos de ensino (PINTO e SILVA, 2017). Tem-se, por exemplo, a Aprendizagem Baseada em Times, a qual consiste em formar grupos de alunos para desenvolver um projeto ou resolver um problema, às vezes atentando-se para a heterogeneidade do grupo (MILCZARSKI et al., 2021; ANDERSSON e LOGOFATU, 2018). Outro método é o *role-play*, que consiste na criação de uma história para envolver os alunos, isto é, um cenário fictício é criado para simular a vida real (MAXIM et al., 2017). Tem-se, também, a denominada educação cega, do inglês *blended education*, a qual se refere ao ensino à distância com uso de tecnologias (LYZ et al., 2020) e, por fim, a integração da pesquisa dentro do ensino em diferentes níveis, podendo receber a pesquisa dos professores, trabalhar ao lado deles de forma investigativa ou, ainda, com pesquisa tutorada, onde os alunos trabalham em grupos escrevendo e discutindo artigos (GIACAMAN e SINNEN, 2018). Como complemento, proporcionar um ambiente realista aos estudantes mostra-se um ponto muito forte no ensino (KHAKUREL e PORRAS, 2020).

No geral, a Aprendizagem Ativa consegue motivar os alunos por meio do engajamento proporcionado por ela (ANGELI et al., 2020). Portanto, é uma ferramenta que poderia promover uma aprendizagem e um envolvimento mais significativos entre os alunos, aplicando estratégias ativas de aprendizagem, como PBL, *role-play* e gamificação (MAXIM et al., 2017).

De acordo com Younis et al. (2021) e Carter (2011), existem três formas de se desenvolver *soft skills* em estudantes de graduação:

- 1) Os alunos desenvolvem *soft skills* em matérias diversas: A habilidade de trabalhar em equipe pode ser desenvolvida em aulas de ciências sociais e a habilidade de escrita em uma matéria de redação, por exemplo. Também abrange o conceito observacional, no qual os alunos observam a atitude do professor e, com isso, aprendem. O problema dessa abordagem é que profissionais precisam aprender a se comunicar com conceitos técnicos presentes em sua área de atuação, o que não acontece em matérias de formação geral;
- 2) Disciplinas específicas são criadas para apresentação e desenvolvimento de *soft skills*: Esta abordagem tem se mostrado impopular entre estudantes e professores, pois estes consideram as disciplinas mais difíceis de serem ministradas, enquanto aqueles sentem-

se desmotivados por terem a impressão de estarem aprendendo coisas das quais já possuem conhecimento prévio;

- 3) Estímulos são incorporados, intencionalmente, dentro de projetos de disciplinas para desenvolver *soft skills*: Esta é uma abordagem que se mostra a mais popular e mais encontrada na literatura. Pode-se, ainda, aplicar o conceito de *capstones* dentro de disciplinas. Estes são cursos que devem, segundo o currículo da ACM/IEEE¹, abranger um ano letivo completo, enfatizar o trabalho em grupo, enfatizar a implementação real e abordar a implementação de um conceito. Khakurel e Porras (2020), por exemplo, aplicam os *capstones*, porém adaptando-os para um período de seis meses.

No trabalho de Caeiro-Rodríguez et al. (2021), verifica-se que ao questionar os 184 estudantes participantes da pesquisa sobre a percepção deles quanto a efetividade do sistema educacional vigente para desenvolvimento de *soft skills*, 45% mostraram-se insatisfeitos, 41% ficaram indecisos e apenas 14% responderam positivamente. Assim, são apresentadas as metodologias emergentes reunidas pelo projeto ModEs (MODES Project, 2020 apud CAEIRO-RODRÍGUEZ et al., 2021), isto é, estratégias pedagógicas que podem ser aplicadas no ensino para desenvolvimento de *soft skills*. São elas as metodologias: expositiva, guiada ou ativa. A primeira utiliza leituras, seminários e demonstrações, por exemplo, para desenvolver as *soft skills*. Já a segunda utiliza discussões, *workshops*, projetos e mentorias para alcançar tal objetivo. Por fim, a metodologia ativa utiliza debate, encenação, jogos de negócios e visitas/viagens, dentre outros. Os alunos apontaram que a proposta que mais os agrada para o desenvolvimento de *soft skills* é a metodologia ativa PBL, pois permite desenvolver projetos para resolver problemas que simulem uma situação do mundo real.

Nesse âmbito, um ponto a ser considerado é a motivação dos alunos no processo de desenvolvimento. Para isso, é importante que eles entendam o que são as *soft skills*, qual o motivo de estarem buscando desenvolvê-las e para que servem, sendo isso um ponto de partida para os professores aplicarem métodos de ensino (BYRNE et al., 2020).

Brito et al. (2020) diz que o PBL ajuda na motivação e no processo de aprendizagem, uma vez que envolve os alunos na investigação com o objetivo de construir um produto, resolver um problema ou ambos. Os autores notaram, entretanto, que é preciso que os projetos sejam baseados em situações passíveis de acontecerem na vida real, além de ter prazos e atividades que possam ser concluídas no período escolar; caso contrário, os alunos podem se desmotivar ou sentir que fracassaram por não conseguirem finalizar o projeto. De fato, no

¹ Disponível em: https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf. Acessado pela última vez em: 03 de novembro de 2023.

trabalho de Garousi et al. (2019) os autores afirmam que, diante dos 35 trabalhos avaliados, é possível perceber que a aplicação de projetos baseados na vida real mostrou-se eficiente para desenvolver nos estudantes as *soft skills* de forma ativa. Por fim, grande parte das abordagens atuais propõem abordagens de aprendizagem ativa (CAEIRO-RODRÍGUEZ et al., 2021). Nesse contexto, Magano et al. (2021) afirma que o PBL é uma das formas mais envolventes de trabalhar a lacuna entre a universidade e a indústria uma vez que o PBL proporciona aprendizagem contextualizada. Os autores notaram também a preferência dos alunos por exercícios que simulam situações do mundo real, uma vez que poderiam estimular, entre outros, a autonomia e o pensamento crítico.

Outra forma para desenvolver *soft skills*, tendo o foco também na motivação dos alunos, é por meio da gamificação (SOUZA et al., 2018) ou da combinação de metodologias que seguem a Aprendizagem Ativa (PINTO e SILVA, 2017).

Nesse contexto, são encontrados diferentes exemplos na literatura de aplicação de metodologias para desenvolvimento de *soft skills* em alunos de graduação. Nestas, cada docente, de acordo com a necessidade da turma ou do foco da disciplina, adapta as metodologias ativas para alcançar o objetivo de desenvolver um conjunto de *soft skills*. De fato, Garousi et al. (2019) dizem que a maioria dos estudos foca em uma região, uma vez que as *soft skills* variam em importância, facilidade de desenvolver, dentre outros fatores, de acordo com o meio em que uma pessoa está inserida (ANDERSSON e LOGOFATU, 2018).

Assim, considerando os resultados da RSL, os exemplos de metodologias encontradas em 34 trabalhos foram agrupados. Para organização, apesar de existir complexidade nas metodologias de ensino, as quais podem ser adaptadas para diferentes contextos, buscou-se organizar por metodologias de ensino principais e auxiliares, uma vez que duas ou mais metodologias são combinadas na maioria das vezes. Na Tabela 5 são indicadas a metodologia principal e as metodologias auxiliares, bem como quais trabalhos as utilizaram em suas propostas. Além disso, as principais *soft skills* desenvolvidas, quando especificadas no trabalho, são apontadas. Por fim, indica-se o número de trabalhos (n) com a aplicação exclusiva de determinada metodologia principal de ensino ou, quando for o caso, do seu alinhamento à uma metodologia auxiliar e o número total de trabalhos (Total) que utilizam determinada metodologia principal como solução em suas propostas.

É possível perceber a predominância do PBL, isto é, o uso de projetos para auxiliar os estudantes a desenvolverem ou aprimorarem suas *soft skills*. Esta pode estar, ainda, alinhada a outros métodos, como uso de cenários reais, *role-play* ou gamificação. Isso acontece por essa abordagem trazer maior engajamento dos alunos ao proporcionar uma aprendizagem mais

significativa. Isto é, a partir do momento que os alunos entendem para que serão utilizadas suas *hard* e *soft skills*, eles buscam aprimorá-las.

Tabela 5: Metodologias de ensino para desenvolver Soft Skills

Metodologia principal de ensino: Aprendizagem baseada em projetos (PBL)				
Metodologia auxiliar de ensino	Trabalho(s)	Principais <i>soft skills</i> desenvolvidas	n	Total
-	YOUNIS et al., 2021;	Trabalho em equipe	7	20
	KUPPUSWAMY e MHAKURE, 2020;	-		
	KONINGS e LEGG, 2020;	Trabalho em equipe, liderança, pensamento multidisciplinar, ética, resolução de problemas, tomada de decisão		
	YOUNIS et al., 2019;	Trabalho em equipe		
	FIORAVANTI et al., 2020;	<i>Feedback</i> , motivação, pessoal		
	BARROS e BITTENCOURT, 2018;	Organização, interpessoal, resolução de problemas, tomada de decisão, habilidade analítica, criatividade		
	RIBEIRO e BITTENCOURT, 2018	Pessoal, interpessoal		
Problemas baseados no mundo real	ALVES and ROCHA, 2021;	Comunicação, trabalho em equipe, tomada de decisão	7	
	ZMEEV e ZMEEV, 2020;	-		
	DAVIS e REBELSKY, 2019;	Comunicação, interpessoal		
	BASTARRICA et al., 2017;	Orientação ao cliente, linguagem técnica		
	ZHENG et al., 2015;	Organização, comunicação, trabalho em equipe, motivação		
	CARTER, 2011;	Comunicação, linguagem técnica		
	GONZÁLEZ-MORALES et al., 2011	Trabalho em equipe, liderança		

Aprendizagem baseada em times, problemas baseados no mundo real	VALERO et al., 2020	-	1	
<i>Role-play</i> e gamificação	MAXIM et al., 2017	Organização, comunicação	1	
Aprendizagem baseada em times	PAWAR et al., 2020	Trabalho em equipe, comunicação, resolução de problemas, ética	1	
Problemas baseados no mundo real, times heterogêneos, <i>role-play</i>	PÉREZ e RUBIO, 2020	Trabalho em equipe, comunicação, motivação	1	
Educação cega, <i>framework</i> CDIO	LYZ et al., 2020	Organização	1	
Aprender Ensinando	ANGELI et al., 2020	-	1	
Metodologia principal de ensino: Aprendizagem baseada em times				
Metodologia auxiliar de ensino	Trabalho(s)	Principais <i>soft skills</i> desenvolvidas	n	Total
-	RAMOS et al., 2018;	Trabalho em equipe, habilidade analítica, resolução de problemas	3	6
	NICOLA et al., 2018;	Liderança, habilidade de trabalhar sob pressão, organização, profissionalismo, resolução de problema		
	CUKIERMAN e PALMIERI, 2014	Comunicação, trabalho em equipe, resolução de problema		
Times heterogêneos (culturas e/ou disciplinas)	MILCZARSKI et al., 2021	Comunicação, interpessoal, trabalho em equipe	1	
Gamificação	MAXIM et al., 2019	-	1	

Framework CDIO	KULKARNI et al., 2020	Trabalho em equipe, comunicação, organização	1	
Metodologia principal de ensino: Aprendizagem baseada em problemas (PrBL)				
Metodologia auxiliar de ensino	Trabalho(s)	Principais <i>soft skills</i> desenvolvidas	n	Total
<i>e-learning</i> Aprendizagem baseada em problemas (ePrBL)	DEEP et al., 2019	Aprendizagem, resolução de problema	1	5
Problemas do mundo real, times heterogêneos	ANDERSSON e LOGOFATU; 2018	Comunicação, resolução de problema, trabalho em equipe	1	
Problemas do mundo real	RHODES et al., 2018	Resolução de problemas, liderança, comunicação	1	
Pesquisa	GIACAMAN e SINNEN, 2018	Trabalho em equipe, resolução de problema, comunicação, organização	1	
Projetos	IBRAHIM e ABD HALIM, 2014	Habilidade analítica, resolução de problema, organização, comunicação	1	
Metodologia principal de ensino: Aprendizagem baseada no desenvolvimento de jogos (GDBL)				
Metodologia auxiliar de ensino	Trabalho(s)	Principais <i>soft skills</i> desenvolvidas	n	Total
-	PANTHALOOKARAN, 2018	Pensamento crítico, criatividade, trabalho em equipe, comunicação	1	1
Metodologia principal de ensino: Gamificação				
Metodologia auxiliar de ensino	Trabalho(s)	Principais <i>soft skills</i> desenvolvidas	n	Total
-	HUFFMAN et al., 2019	Profissionalismo, pensamento multidisciplinar,	1	1

		liderança, trabalho em equipe		
Metodologia principal de ensino: Active Video Watching (AVW)				
Metodologia auxiliar de ensino	Trabalho(s)		n	Total
-	GALSTER et al., 2018	comunicação	1	1

Fonte: Desenvolvido pelo autor.

2.4 Avaliação das *soft skills*

Nos trabalhos analisados, não foi encontrado um método específico de avaliação do desenvolvimento das *soft skills*, pois mostra-se como uma característica pessoal, difícil de ser medida. Porém, algumas abordagens são apresentadas na literatura e, assim, esta subseção apresenta uma discussão sobre métodos de avaliação do desenvolvimento das *soft skills*, a fim de obter uma visão completa para a estruturação do *framework* proposto nesse trabalho.

Nesse contexto, alguns trabalhos na literatura utilizam métricas para avaliar atividades que contemplam *soft skills*, além de formulários aplicados antes e após determinada intervenção para autoavaliação (YOUNIS et al., 2021; KONINGS e LEGG, 2020; KULKARNI et al., 2020). Adicionalmente, tem-se trabalhos que alinham tais técnicas a análise com testes estatísticos para verificar se houve desenvolvimento das *soft skills* (YOUNIS et al., 2021; PÉREZ e RUBIO, 2020). Também é possível utilizar o *framework* conhecido como Taxonomia de Bloom como um facilitador na estruturação dos objetivos educacionais e, com isso, impulsionar o desenvolvimento das *soft skills* (SOBRAL, 2021; FOREHAND, 2010; KRATHWOHL, 2002).

No trabalho de Pérez e Rubio (2020) são comparadas as notas de provas realizadas por alunos em dois grupos diferentes, um onde foi aplicado o modelo habitual de ensino e outro em que foi aplicado um projeto baseado no PBL. A partir desses dados, o teste estatístico T de Student foi aplicado para comparar as médias das duas turmas, com o qual os autores concluíram haver aumento significativo nas habilidades técnicas dos alunos quando utilizado a metodologia do PBL. Além disso, os autores aplicaram um formulário final para coletar a percepção dos alunos sobre a aplicação do PBL, sendo possível notar maior satisfação dos alunos com a utilização do PBL, além de melhorias nas *soft skills* motivação, trabalho em equipe e comunicação com a aplicação do PBL, segundo autoavaliação dos estudantes.

No mesmo segmento de ideia, no trabalho de Younis et al. (2021) são aplicados dois formulários coletando a percepção dos alunos sobre o crescimento de suas *hard* e *soft skills* em

uma escala variando de 1 (muito ruim) a 5 (muito bom). Esses formulários são aplicados no início e no fim do semestre, após a realização da intervenção com a metodologia do PBL. A partir desses dados, a fim de verificar a percepção dos alunos quanto ao desenvolvimento de suas *soft skills*, os autores compararam a média atribuída pelos alunos nos dois questionários utilizando o teste estatístico T de Student, juntamente com o coeficiente *d* de Cohen, e notaram que houve crescimento em todas as *soft skills* trabalhadas, sendo que trabalho em equipe apresentou o maior aumento. No trabalho de Dubey e Tiwari (2020), os autores também utilizam teste T de Student com o coeficiente *d* de Cohen para avaliar e comparar a importância que alunos e profissionais da área de tecnologia atribuem a determinadas *soft skills*. A partir do teste, os autores concluíram que em apenas três de quarenta e quatro *soft skills* - interpessoal, pensamento multidisciplinar e capacidade de trabalhar sob pressão - apresentaram uma diferença estatisticamente insignificante, enquanto as outras apresentaram um valor *p* estatisticamente significativo, indicando que os entrevistados não tinham uma visão semelhante em relação aos atributos de habilidades interpessoais.

Já o trabalho de Konings e Legg (2020) apresenta métricas definidas para cada etapa do projeto aplicado. Juntamente com elas, os autores associam as *hard* e *soft skills* que devem ser acessadas para receberem determinada pontuação. A partir disso, é avaliado o desempenho dos alunos. No fim, aplicam formulário para que estes respondam sobre suas percepções de desenvolvimento de *hard* e *soft skills* com a aplicação do PBL em contrapartida às metodologias tradicionais. O PBL apresenta as maiores pontuações de acordo com os alunos.

No mesmo contexto, o trabalho de Kulkarni et al. (2020) apresenta métricas definidas em uma tabela para as etapas do projeto aplicado. Esta tabela fornece pontuação de 0 a 3 para os níveis Insuficiente, Médio, Bom e Ótimo. Além disso, os autores sugerem a aplicação de um formulário para coletar a percepção dos alunos sobre o desenvolvimento das *soft skills*. Os autores levam em consideração ambas as avaliações para verificar se houve desenvolvimento das *soft skills*.

Por fim, alguns exemplos de aplicação da Taxonomia de Bloom são encontrados nos trabalhos de Prasad (2021) e Sobral (2021). O primeiro trabalho aplica um formulário e verifica o desempenho dos alunos em um curso de teste de *software*. O formulário de avaliação contém perguntas abrangendo os níveis de conhecimento descritos na Taxonomia de Bloom para medir o nível de conhecimento dos alunos com relação ao conteúdo previsto para a disciplina. No segundo, são apresentados alguns exemplos de aplicação da Taxonomia de Bloom em um contexto do ensino de fundamentos de programação, como, por exemplo, para ensinar a estrutura de dados “vetor” em uma disciplina de computação. A autora relata que existem

diferentes abordagens para o uso da taxonomia, sendo que alguns trabalhos utilizam o *framework* como uma ferramenta para definição de objetivos de aprendizagem, enquanto outros utilizam para medir o desempenho dos alunos. Além disso, diferentes variações da Taxonomia são utilizadas, sendo que as seis categorias apresentadas na Taxonomia de Bloom original e, também, na revisada, podem ser adaptadas.

Diante disso, nas seções 2.4.1 e 2.4.2 são apresentados, respectivamente, um detalhamento sobre a Taxonomia de Bloom e uma apresentação sobre testes estatísticos e como estes podem ser utilizados no contexto de avaliação do desenvolvimento das *soft skills*.

2.4.1 Taxonomia de Bloom

A Taxonomia de Bloom é um *framework*, ou modelo, apresentado em camadas que classifica o conhecimento em seis níveis de complexidade (FOREHAND, 2010). Assim, professores podem utilizá-lo para direcionar atividades ou identificar o nível de conhecimento em que os alunos se encontram. O *framework* foi apresentado, inicialmente, em 1956 (BLOOM et al., 1956) pelo professor Benjamin S. Bloom da Universidade de Chicago e, posteriormente, teve modificações, sendo a mais conhecida delas a versão apresentada em 2002 no trabalho de Krathwohl (2002), a qual ficou conhecida como Taxonomia de Bloom revisada. Uma das maiores contribuições da Taxonomia de Bloom revisada é a reorganização da estrutura hierárquica, apresentada no primeiro modelo, em duas dimensões (KRATHWOHL, 2002). Assim, o conhecimento foi apresentado nas dimensões: Domínio dos Processos Cognitivos e Domínio do Conhecimento, e teve os nomes das camadas do *framework* substituídos por verbos (SOBRAL, 2021; FOREHAND, 2010).

Na Figura 3 é possível ver a dimensão do Domínio dos Processos Cognitivos da Taxonomia de Bloom em sua versão original, à esquerda, e revisada, à direita.

A estrutura dos Processos Cognitivos apresentados na Figura 3 é definida a seguir (SOBRAL, 2021; FOREHAND, 2010).

- Lembrar: Recuperar conhecimento relevante de memória de longo prazo, seja reconhecendo ou recuperando;
- Compreender: Determinar o significado de mensagens, incluindo orais, escritas e comunicação gráfica, por meio de interpretação, sumarização, exemplificação, classificação ou explicação;
- Aplicar: Realizar ou usar um procedimento em uma situação, seja executando ou implementando;

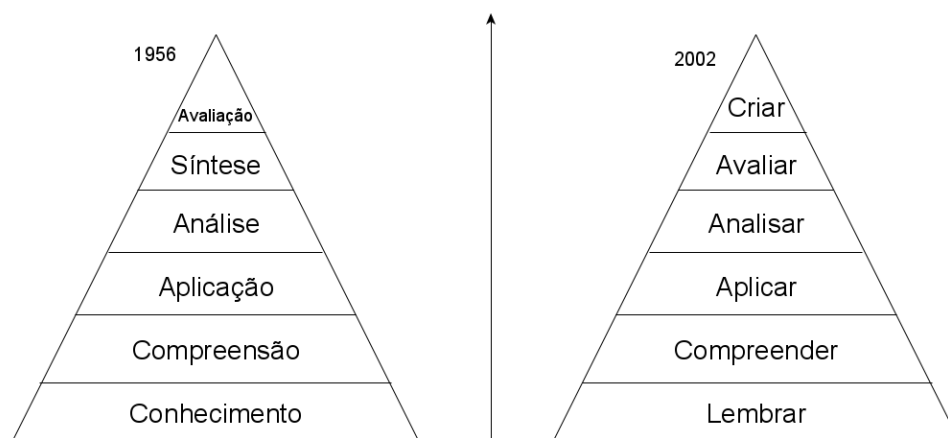
- Analisar: Quebrar o material em suas partes constituintes e detectar como as partes se relacionam umas com as outras, bem como com uma estrutura geral ou propósito, por meio de diferenciação, organização e atribuição;
- Avaliar: Fazer julgamentos com base em critérios e padrões, incluindo verificação e crítica ou julgamento de situações;
- Criar: Juntar elementos para formar algo, um todo coerente ou fazer um produto original, seja gerando, planejando e produzindo.

Quanto à dimensão do Domínio do Conhecimento, a estrutura apresenta quatro tipos de conhecimento. São estes (SOBRAL, 2021):

- Conhecimento Factual: Elementos básicos que os alunos devem saber para se familiarizar com uma disciplina ou resolver problemas nele;
- Conhecimento Conceitual: As inter-relações entre os elementos básicos dentro de uma estrutura maior que lhes permite funcionar em conjunto;
- Conhecimento Processual: Como fazer algo; métodos de inquérito e critérios para usar habilidades, técnicas e métodos;
- Conhecimento Metacognitivo: Conhecimento da cognição em geral, bem como a consciência e o conhecimento da própria cognição.

Segundo Krathwohl (2002), o modelo revisado apresenta a Tabela de Taxonomia, resumida na Tabela 6, a qual permite classificar os objetivos, atividades e avaliações com uma visão clara, concisa e visual de representação de um determinado objetivo educacional ou atividade. A partir desse planejamento, os professores podem examinar e adequar sua proposta pedagógica mais facilmente.

Figura 3: Taxonomia de Bloom Original e Revisada no Domínio dos Processos Cognitivos



Fonte: Adaptada de Sobral (2021).

Diante disso, optou-se por utilizar a Taxonomia de Bloom revisada neste trabalho, a fim de utilizar a Tabela 6 como uma facilitadora na elaboração das atividades do *framework* proposto. Segundo Sobral (2021), a tabela deve ser preenchida de acordo com os objetivos de aprendizagem, deixando em branco as linhas e colunas que não serão trabalhadas em determinada atividade. Seu objetivo é auxiliar na estruturação dos objetivos educacionais.

Tabela 6: Taxonomia de Bloom nas duas dimensões

Dimensão do	Dimensão do Processo Cognitivo					
Conhecimento	Lembrar	Compreender	Aplicar	Analisar	Avaliar	Criar
Factual						
Conceitual						
Processual						
Metacognitivo						

Fonte: Adaptado de Sobral (2021).

2.4.2 Testes estatísticos

Testes estatísticos são utilizados para observar e analisar resultados. Para isso, são estabelecidas duas hipóteses, a hipótese inicial ou nula (H_0) e a hipótese alternativa (H_1). A partir disso, busca-se saber se H_0 é aceita e H_1 rejeitada, ou o contrário. Em ambos os casos, é comparado o nível crítico, ou p-valor, ao nível de significância alfa (α) estabelecido. O p-valor

é o menor valor para o qual, ao usar determinada metodologia com os valores observados, ainda rejeitaríamos H_0 . O valor α é geralmente definido entre 0,01 e 0,05, sendo que, quanto maior a precisão requerida no teste, menor este deve ser (PINHEIRO et al., 2012; VIEIRA, 2008).

Assim, tem-se:

$p < \alpha \rightarrow H_0$ é rejeitada e H_1 é aceita

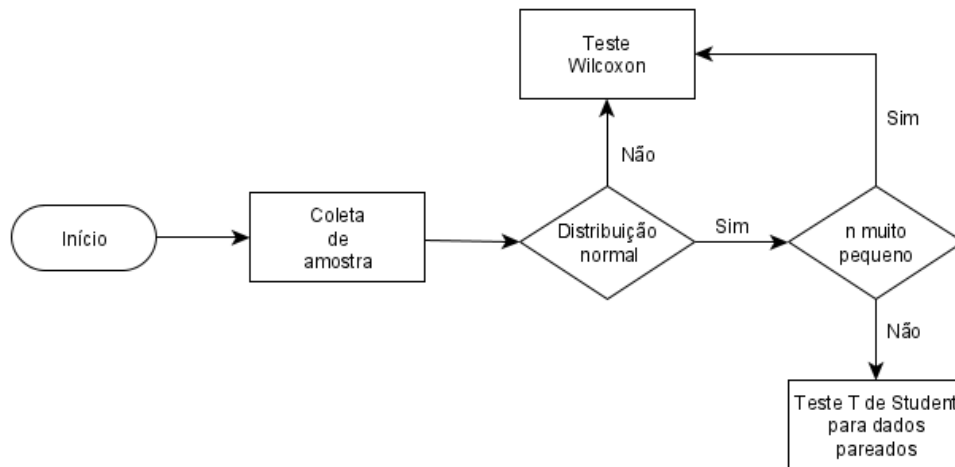
$p \geq \alpha \rightarrow H_0$ é aceita e H_1 é rejeitada

Para escolha adequada do teste estatístico, verifica-se se os dados são pareados (dependentes), ou não pareados (independentes). No caso de dados dependentes, trata-se de análise da mesma variável nas mesmas unidades antes e depois de uma intervenção, enquanto o segundo analisa grupos independentes de dados (VIEIRA, 2008). Outro ponto é que, se o pesquisador busca verificar se houve mudança na média calculada, tem-se um teste bilateral. Em contrapartida, se a intenção for verificar o sinal de diferença entre as médias analisadas, tem-se um teste unilateral, no qual dobra-se o nível de significância α utilizado (VIEIRA, 2008).

Assim, um fluxo deve ser seguido para a escolha do teste estatístico mais adequado, seguindo os critérios para cada um deles e o objetivo que se espera alcançar. No geral, existem dois tipos de teste: os paramétricos e os não-paramétricos. No primeiro, as análises são realizadas baseadas na média, enquanto no segundo, a medida utilizada é a mediana (VIEIRA, 2008). Para utilizar um teste paramétrico, algumas condições devem ser satisfeitas: (1) a amostra não pode ser muito pequena e (2) deve haver distribuição normal. Uma amostra é considerada “não muito pequena” a partir de 30 elementos por alguns autores (LEVINE et al., 2000). Quanto a verificar se uma distribuição é normal, isso pode ser feito aplicando-se testes como o Shapiro-Wilk (VIEIRA, 2008) no conjunto de dados, por exemplo. Tais verificações são importantes para garantir conclusões corretas da estatística do teste. Caso a amostra não cumpra esses requisitos, é possível comparar duas medianas ao invés de duas médias. Tem-se como exemplo de teste paramétrico para comparar dois grupos pareados, o teste T de Student, já como exemplo de seu correspondente não-paramétrico, tem-se o Teste Wilcoxon (VIEIRA, 2008).

Assim, tendo-se como objetivo comparar o desenvolvimento das *soft skills* de um mesmo grupo de estudantes antes e depois de uma intervenção, deve-se utilizar teste de amostra pareada. Portanto, um fluxograma de escolha do teste estatístico adequado entre duas amostras pareadas é apresentado na Figura 4.

Figura 4: Fluxograma para escolha do teste estatístico para duas amostras pareadas



Fonte: Desenvolvido pelo autor.

É válido ressaltar que a hipótese H_0 do teste de normalidade Shapiro-Wilk indica que a amostra possui distribuição normal. Assim, $p < \alpha$ indica que a distribuição não é normal (H_1). No mesmo contexto, os testes T de Student e Wilcoxon tem como H_0 que não há mudança nos dados. Assim, $p < \alpha$ indica que houve mudança (H_1) (VIEIRA, 2008).

Entretanto, a estatística dos testes informa apenas se uma diferença é significativa, ou não, porém não revela a magnitude desta diferença. Para isso, calcula-se o tamanho de efeito sugerido por Cohen (1977), o qual é obtido a partir do coeficiente d para testes paramétricos e com o seu correspondente não-paramétrico, coeficiente r . O primeiro é baseado na média e desvio padrão da amostra, enquanto o segundo é baseado na estatística do teste, o qual, por sua vez, é baseado na mediana da amostra (COHEN, 1977; FRITZ et al., 2012).

O coeficiente d de Cohen permite-nos interpretar a criticidade da variância de vários atributos por tamanho do efeito (COHEN, 1992 apud DUBEY e TIWARI, 2020). Para cálculo do coeficiente d de Cohen é indicada a Fórmula 1 (DUBEY e TIWARI, 2020; YOUNIS et al., 2021), sendo que $M1$ representa a média da variável 1, $M2$ a média da variável 2 e SD_{pooled} é o agrupamento do desvio padrão das duas variáveis.

$$D = \frac{(M1 - M2)}{SD_{pooled}} \quad (1)$$

Para cálculo de SD_{pooled} , por sua vez, é usada a Fórmula 2 (DUBEY e TIWARI, 2020; YOUNIS et al., 2021), sendo que $SD1$ representa o desvio padrão da variável 1 e $SD2$, da variável 2.

$$SD_{pooled} = \sqrt{\frac{SD1^2 + SD2^2}{2}} \quad (2)$$

Já o coeficiente r sugerido por Cohen (1977) e aperfeiçoado por Fritz et al. (2012), baseia-se na estatística fornecida por teste não-paramétricos, chamada de z , e no tamanho da amostra N . Fritz et al. (2012) afirma, ainda, que as estimativas do tamanho do efeito permanecem independentes do tamanho da amostra, apesar da presença de N nas fórmulas. Isso ocorre porque z é sensível ao tamanho da amostra, então, ao dividir por uma função de N , remove-se o impacto do tamanho da amostra da estimativa do tamanho do efeito resultante. Assim, para cálculo de r é aplicada a Fórmula 3.

$$r = \frac{z}{\sqrt{N}} \quad (3)$$

Por fim, para análise dos resultados, tem-se a Tabela 7 como referência para valores do tamanho do efeito de d e r (COHEN, 1977; FRITZ et al., 2012).

Tabela 7: Interpretação dos valores dos coeficientes de tamanho de efeito d e r .

	Insignificante	Pequeno	Médio	Grande
d	< 0,2	$0,2 \leq d < 0,5$	$0,5 \leq d < 0,8$	$d \geq 0,8$
r	< 0,1	$0,1 \leq r < 0,3$	$0,3 \leq r < 0,5$	$r \geq 0,5$

Fonte: Desenvolvido a partir de dados de Cohen (1977) e Fritz et al. (2012).

Capítulo 3 – *Framework for Soft Skills Development* (FraSSD)

3.1 Considerações iniciais

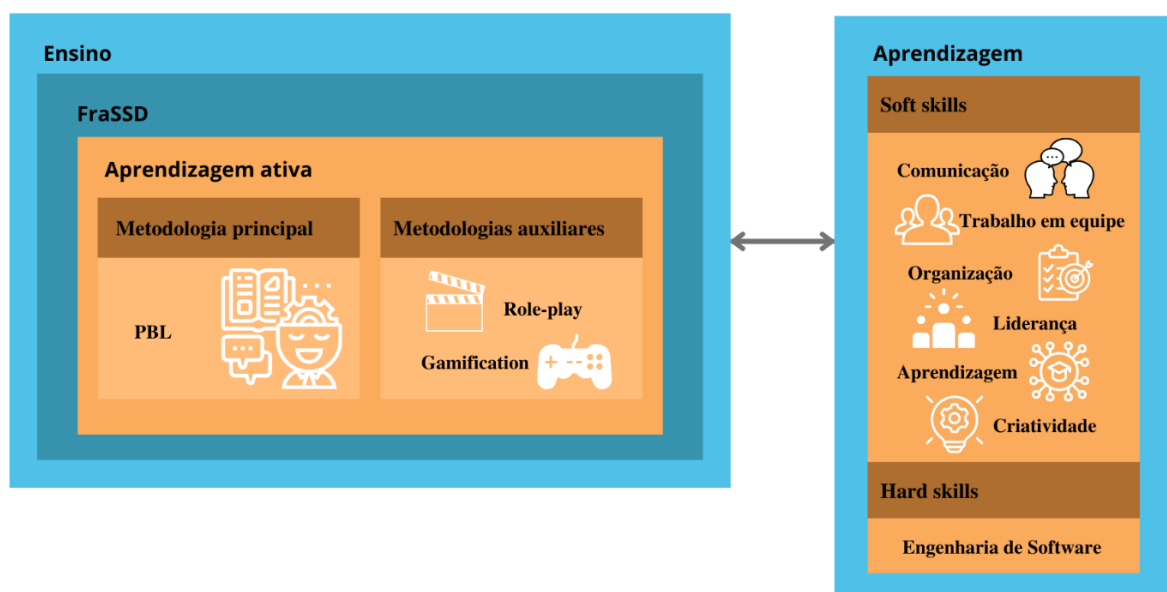
O *Framework for Soft Skills Development* (FraSSD) tem como objetivo ajudar a suprir a necessidade de se investir, cada vez mais, no desenvolvimento das *soft skills* necessárias a futuros profissionais de *software* para que estes exerçam suas tarefas com eficiência e qualidade. Assim, o FraSSD apresenta uma sequência de atividades para auxiliar o docente no desenvolvimento de *soft skills* em alunos de graduação, com foco na Engenharia de Software.

Para detalhamento do FraSSD, na seção 3.2 são apresentadas as premissas para elaboração do *framework*; na seção 3.3 são descritos os atores presentes no FraSSD; e, por fim, na seção 3.4 são detalhadas as atividades nas visões de cada ator, isto é, aluno e professor.

3.2 Premissas

O *framework* proposto - FraSSD, foi estruturado de forma que possa ser aplicado durante uma disciplina de Engenharia de Software, comumente obrigatória nas estruturas curriculares de cursos de graduação de Bacharelado em Ciência da Computação. Na Figura 5 são esquematizadas as premissas estabelecidas para definição do *framework*.

Figura 5: Estruturação do FraSSD



Fonte: Desenvolvido pelo autor.

Da análise da Figura 5, observa-se que, dado o contexto voltado para uma disciplina de Engenharia de Software, foi estabelecida como metodologia principal o PBL, uma vez que essa se mostrou mais eficiente para desenvolvimento de *soft skills* segundo os resultados obtidos a partir da RSL. Além disso, conta com fator motivacional ao colocar o foco no aluno com uma aprendizagem ativa.

De maneira complementar, foram adotadas duas metodologias auxiliares: o *role-play* e a gamificação. Para o *role-play*, é prevista a criação de uma história para envolver os alunos, estabelecendo um cenário fictício para simular a vida real (MAXIM et al., 2017). O objetivo é que os alunos se sintam motivados ao participar de uma situação que pode vir a acontecer em suas vidas profissionais, trazendo a sensação de um aprendizado útil e, assim, possibilitando uma aprendizagem mais significativa. Já para a gamificação, alguns elementos foram incorporados ao *framework*, a saber: competição, pontuação e placar (SOUZA et al., 2018), com o intuito de engajar e motivar os alunos no projeto.

Além disso, uma vez que abordar todas as *soft skills* almejadas para profissionais de software seria inviável no contexto de uma disciplina semestral, selecionou-se as seis mais frequentemente citadas na literatura e que são possíveis de serem abordadas, em conjunto com as *hard skills*, durante o desenvolvimento de um projeto: comunicação, trabalho em equipe, organização, liderança, aprendizagem e criatividade.

3.3 Atores

O *framework* prevê a participação de dois tipos de atores: aluno e professor. Ambos os atores devem desenvolver atividades específicas para que os objetivos do *framework* de auxiliar no desenvolvimento gradual de *soft skills* dos alunos, bem como fornecer um material de apoio para os professores, sejam alcançados.

Na visão do aluno, serão associadas a ele atividades com entregas parciais do projeto, para as quais ele deverá acessar *hard* e *soft skills* para conseguir completar de forma satisfatória. No quesito avaliação, o aluno é responsável por realizar uma autoavaliação do nível de desenvolvimento de suas *soft skills*. Tal avaliação acontecerá antes e após a intervenção proposta pelo *framework* por meio de formulários. O objetivo para isso é verificar se houve algum avanço ou contribuição no nível de desenvolvimento das *soft skills* dos estudantes com base na comparação das respostas. Tal avaliação é importante para estimular a análise crítica dos alunos sobre seu próprio desenvolvimento pessoal e a busca por melhorias.

Na visão do professor, ele é o responsável por atividades de organização, planejamento e apresentação da proposta de projeto para os alunos. Além disso, cabe a ele definir tarefas, prazos, critérios de avaliação, bem como coordenar como o projeto será realizado, prestando apoio aos alunos. No quesito avaliação, ele é o responsável por avaliar criticamente o desenvolvimento do projeto dos alunos, analisando se foram trabalhadas *hard* e *soft skills* de acordo com as métricas definidas. Isso é importante pois o professor pode ter uma visão externa dos alunos de como estes tem se desenvolvido com relação ao conteúdo programático da matéria, *hard skills*, e ao pessoal, *soft skills*. A partir dessa análise, o professor também é responsável por gerenciar um placar com a pontuação dos grupos seguindo a metodologia de gamificação.

3.4 Descrição do *framework*

O *framework* foi estruturado em atividades voltadas tanto para a visão do aluno quanto para a visão do professor. Na visão do aluno, apresenta atividades apoiadas na Taxonomia de Bloom revisada, a fim de promover desenvolvimento gradual das *soft skills*. Na visão do professor, apresenta atividades divididas em quatro etapas: Contextualização, Projeto, Métricas e Avaliação. Tais etapas foram pensadas com o intuito de apresentar um modelo sequencial e organizado para os professores seguirem durante a disciplina a fim de cumprir o conteúdo previsto e, ao mesmo tempo, contribuir para o desenvolvimento gradual das *soft skills* dos alunos.

Nas subseções 3.4.1 e 3.4.2 são apresentadas as atividades na visão do aluno e do professor, respectivamente.

3.4.1 Atividades do FraSSD na visão do aluno

As atividades do FraSSD na visão do aluno representam entregas a serem realizadas no decorrer da disciplina a fim de desenvolver o projeto proposto pelo professor. Tais entregas foram estabelecidas com base na Taxonomia de Bloom revisada, a fim de trazer um conhecimento progressivo das *soft skills* aos alunos, ajudando-os a desenvolvê-las.

Assim, buscando auxiliar o professor na organização das atividades para um desenvolvimento progressivo, foi feito um mapeamento das atividades do FraSSD em relação à Taxonomia de Bloom nas quais haverá participação dos alunos. Uma visão geral destas pode ser vista na Figura 6.

Tem-se como principal objetivo da atividade A1 promover a compreensão do que são as *soft skills* e sua importância – conhecimento factual (A1.1), além das inter-relações entre estas e a área de Engenharia de Software – conhecimento conceitual (A1.2). A Atividade deve ser desenvolvida logo nas primeiras aulas, antes do início do projeto, a fim de contextualizar os alunos.

Figura 6: Atividades do FraSSD na visão do aluno

A1: Contato com conceitos das <i>soft skills</i>	A2: Desenvolvimento do projeto
• A1.1 - Compreender os conceitos das <i>soft skills</i> • A1.2 - Compreender as inter-relações das <i>soft skills</i> com a Engenharia de Software	• A2.1 - Lembrar conceitos de <i>hard</i> e <i>soft skills</i> relevantes ao projeto • A2.2 - Compreender as inter-relações entre <i>hard</i> e <i>soft skills</i> e o projeto • A2.3 - Aplicar métodos, técnicas e conhecimentos para realizar o projeto • A2.4 - Avaliar criticamente as alternativas para solução do projeto • A2.5 - Criar uma solução para o problema apresentado no projeto
A3: Avaliação do nível de desenvolvimento pessoal das <i>soft skills</i>	
• A3.1 - Analisar o nível de desenvolvimento pessoal em cada <i>soft skill</i> • A3.2 - Avaliar o nível de desenvolvimento pessoal em cada <i>soft skill</i> antes e depois da intervenção	

Fonte: Desenvolvido pelo autor.

A Atividade A2 envolve o desenvolvimento do projeto proposto na disciplina conforme o *framework* orienta. Para desenvolver o projeto, os alunos deverão lembrar elementos básicos das *hard* e *soft skills* relacionadas à Engenharia de Software – conhecimento factual (A2.1), compreender quais as inter-relações desse conhecimento com o projeto a ser desenvolvido – conhecimento conceitual (A2.2), serem capazes de aplicar métodos e técnicas na criação e desenvolvimento do projeto – conhecimento processual (A2.3), avaliarem criticamente qual a melhor maneira de aplicar tais métodos e técnicas – conhecimento metacognitivo (A2.4) - e, por fim, criarem uma solução para o problema proposto no projeto – conhecimento processual (A2.5).

Por fim, a Atividade A3 tem como objetivo que os alunos avaliem o nível de desenvolvimento de suas *soft skills*. Esta deve ser aplicada em dois momentos: antes da intervenção do *framework* e depois. Tal avaliação será realizada por meio de resposta a

formulário e tem o intuito de entender se houve, ou não, desenvolvimento em alguma *soft skill* trabalhada. Os alunos precisarão refletir sobre o próprio nível de desenvolvimento em cada *soft skill* apresentada (A3.1) e depois avaliá-lo (A3.2) – conhecimento metacognitivo.

Na Tabela 8 é apresentada a relação entre as atividades descritas e a Taxonomia de Bloom revisada, utilizando, para isso, o formato ilustrado na Tabela 6 (subseção 2.4.1) como um facilitador na estruturação dos objetivos educacionais.

Tabela 8: Atividades do FraSSD na visão do aluno em relação à Taxonomia de Bloom revisada

Dimensão do Conhecimento	Dimensão do Processo Cognitivo					
	Lembrar	Compreender	Aplicar	Analisar	Avaliar	Criar
Factual	A2.1	A1.1				
Conceitual		A1.2, A2.2				
Processual			A2.3			A2.5
Metacognitivo				A3.1	A3.2, A2.4	

Fonte: Desenvolvido pelo autor.

3.4.2 Atividades do FraSSD na visão do professor

Na visão do professor, existem atividades de elaboração do projeto e sua aplicação junto aos alunos, avaliação e *feedback*, as quais são divididas em quatro etapas: Contextualização, Projeto, Métricas e Avaliação. As atividades, de maneira geral, foram pensadas para auxiliar o professor na organização e elaboração do projeto de forma a suprir as demandas das atividades estabelecidas para a visão dos alunos, com o intuito de proporcionar uma aprendizagem contínua e significativa para eles. Além disso, tomou-se o cuidado de combinar os aspectos curriculares da disciplina onde o *framework* será aplicado com a introdução das *soft skills*.

Um maior detalhamento de cada etapa, com as respectivas atividades na visão do professor e sua relação com as atividades na visão do aluno, é apresentado nas subseções 3.4.2.1 a 3.4.2.4.

3.4.2.1 Contextualização

A Etapa de Contextualização (Co) tem como objetivo levantar informações antes do início da aplicação do projeto, além de criar a história seguindo a técnica do *role-play* a fim de

contextualizar o projeto para os alunos. Esta etapa envolve as atividades Co1 a Co5, conforme apresentado no Quadro 1.

Quadro 1: Atividades da Etapa de Contextualização

Etapa de Contextualização (Co)
Co1 – Verificar tempo e recursos disponíveis
Co2 – Verificar a quantidade de grupos para o projeto
Co3 - Verificar quais tópicos serão abordados na disciplina atual
Co4 - Verificar conhecimentos técnicos prévios dos alunos
Co5 - Definir a história (<i>role-play</i>)

Fonte: Desenvolvido pelo autor.

Na Atividade Co1 busca-se verificar o tempo e recursos disponíveis para aplicar o projeto. Assim, com o intuito de balancear o conteúdo teórico da disciplina com o projeto proposto pelo *framework*, recomenda-se que o tempo destinado a aplicação do *framework* em sala de aula não exceda 25% da carga horária da disciplina, seja ela anual, semestral ou bimestral. É válido ressaltar que esse tempo de 25% se refere às atividades em sala de aula. Estas envolvem atividades como apresentação pelo professor aos alunos sobre as *soft skills*; apresentação final do projeto desenvolvido pelos alunos, bem como avaliação destes trabalhos por alunos e professor, conforme necessário; momento para o professor apresentar *feedback* aos alunos, conforme necessário; e autoavaliação por meio de formulários. Dessa forma, para realizar as atividades entregáveis do projeto, os alunos deverão utilizar período extra ao da disciplina. Quanto aos recursos, pode-se observar a possibilidade de utilizar projetores para apresentação, meios digitais de compartilhamento de vídeos e documentos, repositórios de códigos compartilhados, dentre outros. Por fim, o professor deve analisar se terá, ou não, alguém para auxiliá-lo nessa tarefa, como outro docente ou alunos de pós-graduação, monitores, dentre outros.

Quanto à Atividade Co2, voltada para verificar a quantidade de grupos para o projeto para aplicação do PBL, cabe observar que o trabalho de Al Mulhim e Eldokhny (2020) afirma que, apesar de ser mais aceito na literatura a eficácia de grupos menores (3 a 4 alunos), os resultados da sua pesquisa mostram que os alunos apresentam maior satisfação, além de obterem maior qualidade no produto, em grupos grandes (7 a 8 alunos). Por outro lado, o trabalho de Chen e Yang (2019) mostra que o desenvolvimento dos alunos não é influenciado pelo tamanho do grupo do PBL. Dessa forma, não há um consenso na literatura quanto ao tamanho ideal dos grupos. Por outro lado, Kokotsaki et al. (2016) indicam que a implementação

bem-sucedida do PBL na sala de aula depende da capacidade do professor de efetivamente apoiar a aprendizagem dos alunos, motivá-los e orientá-los ao longo do caminho. Para isso, é essencial que os professores recebam apoio da instituição escolar (ERSTAD, 2002) e de outros colegas, pois, quando estes se sentem bem apoiados por suas instituições em termos de competência e autonomia, ficam mais motivados para implementar e persistir no uso do PBL (LAM, et al., 2010). Assim, para a Atividade Co2, recomenda-se que o professor analise qual o tamanho da turma e quantos grupos conseguiria gerenciar, já tendo verificado na Atividade Co1 se terá alguma pessoa para o auxiliar na administração dos grupos. Dessa forma, o foco seria em quantos grupos o professor consegue administrar, a partir do tamanho da turma, mais do que qual o tamanho dos grupos em si.

Após isso, verifica-se na Atividade Co3 o conteúdo programático da disciplina em que o projeto será inserido. Uma vez que o FraSSD utiliza a abordagem de ensino de incorporar, intencionalmente, estímulos para desenvolver habilidades como comunicação e trabalho em equipe dentro de projetos de disciplinas (YOUNIS et al., 2021; CARTER, 2011), o projeto deve estar alinhado ao conteúdo programático.

Na Atividade Co4, tem-se como sugestão avaliar o conteúdo programático de disciplinas que são pré-requisitos dos alunos ou aplicar um teste de nivelamento logo no início da disciplina para conseguir entender o nível de conhecimento técnico dos alunos nos tópicos relacionados a Engenharia de Software até aquele momento. Esse passo é importante, pois o conhecimento prévio dos alunos pode ser um elemento auxiliador no processo de desenvolvimento de *soft skills* (ANDERSSON e LOGOFATU, 2018). Assim, observar as *hard skills* já adquiridas pelos alunos auxiliará na elaboração das etapas do projeto.

Por fim, na Atividade Co5, tem-se a criação da história, seguindo o método do *role-play*. Esta tem como objetivo criar um contexto para os alunos, aproximando o projeto de duas situações: o ambiente em que os alunos estão inseridos e uma situação da vida real. Isto é, a história deve simular algo que poderia acontecer em algum momento de suas vidas profissionais, pois, assim, trará a sensação aos alunos de estarem realizando algo útil, auxiliando-os a ter um aprendizado mais significativo. Por fim, esta deve contemplar o elemento de competição da gamificação, criando uma situação que impulse a competição entre os alunos a fim de ajudar no engajamento e motivação.

Para melhor organização e cumprimento das atividades, foi proposto um modelo para a Etapa de Contextualização, o qual é apresentado no Apêndice C, sendo a 1ª Parte do FraSSD. É válido ressaltar que essa primeira etapa é realizada apenas uma vez antes do início da

disciplina e, dessa forma, a 1ª Parte do Apêndice C será preenchida uma única vez no momento citado.

3.4.2.2 Projeto

A Etapa de Projeto (Pr) tem como objetivo planejar o projeto a ser aplicado, definindo as tarefas que serão propostas aos alunos, quais *hard* e *soft skills* serão trabalhadas em cada uma delas, *feedbacks* que serão apresentados aos alunos, prazos e indicações de ferramentas de apoio aos alunos. Por fim, consiste na aplicação e desenvolvimento do projeto com os alunos. Logo, esta etapa envolve as atividades Pr1 a Pr6, conforme apresentado no Quadro 2.

Quadro 2: Atividades da Etapa de Projeto

Etapa de Projeto (Pr)
Pr1 – Definir as tarefas
Pr2 - Descrever quais <i>soft</i> e <i>hard skills</i> serão trabalhadas em cada tarefa
Pr3 – Definir <i>feedbacks</i> para cada tarefa e prazos para o professor apresentá-los
Pr4 - Definir prazo de entrega e data de aplicação das tarefas
Pr5 - Sugerir ferramentas de apoio para as tarefas
Pr6 – Aplicação e desenvolvimento do projeto

Fonte: Desenvolvido pelo autor.

Para elaborar o projeto, as atividades Pr1 a Pr5 devem ser trabalhadas juntas, sendo possível utilizar a 2ª Parte do Apêndice C como modelo. Assim, ao definir as tarefas do projeto em Pr1, deve-se definir quais *soft* e *hard skills* serão trabalhadas (Pr2), assim como definir a maneira que os *feedbacks* serão apresentados aos alunos e quais os prazos para o professor apresentá-los (Pr3), definir quais os prazos de entrega das tarefas pelos alunos e qual a data de aplicação destas em sala de aula, caso necessário (Pr4) e, por fim, pensar em sugestões de ferramentas que podem auxiliar os alunos a desenvolvê-las (Pr5).

Porém, é importante salientar que, durante o planejamento da disciplina, deve-se reservar um momento, antes do início do projeto, para cumprir dois objetivos estabelecidos nas Atividades A1 e A3 da visão dos alunos, os quais são: (1) apresentar os conceitos de *soft skills* aos alunos, salientando sua importância para um futuro Engenheiro de Software e (2) promover autoavaliação do nível de desenvolvimento das *soft skills* dos alunos antes da intervenção do *framework*. Para alcançar o primeiro objetivo, Byrne et al. (2020) afirma que um ponto de partida para os professores aplicarem métodos de ensino é fazer com que os alunos entendam o que são as *soft skills*, qual o motivo de buscar desenvolvê-las e para que servem.

Para alcançar o segundo objetivo, os alunos devem ser consultados sobre o aceite ou não de participar da pesquisa por meio de um Termo de Consentimento Livre e Esclarecido (Apêndice D). Em caso de aceite, responderão a um formulário inicial, descrito no Apêndice E, antes do início do desenvolvimento do projeto. Isso permitirá que eles classifiquem suas percepções pessoais sobre cada *soft skill* que será trabalhada durante o projeto. Para classificação será utilizada a Escala Likert, na qual os participantes de uma pesquisa podem expressar suas respostas em uma escala com nível de concordância entre “Muito ruim” e “Muito Bom” representados em valores de 1 a 5, respectivamente (JOSHI et al., 2015). Além disso, os alunos responderão algumas perguntas sobre sua experiência prévia envolvendo *soft skills* e PBL, além de informar se têm experiência profissional na área. Tais informações auxiliarão em uma interpretação mais adequada para o estabelecimento dos resultados do projeto.

Especificamente sobre a Atividade Pr1, é preciso definir as tarefas para desenvolvimento do projeto, as quais devem apresentar aos alunos entregas incrementais, a fim de direcioná-los no desenvolvimento do projeto em sua totalidade. A partir disso, deve-se atentar para quais *soft skills* serão trabalhadas em cada tarefa, bem como alinhá-las às *hard skills*, isto é, ao conhecimento técnico previsto no conteúdo programático da disciplina em andamento, o que equivale a Atividade Pr2. Afinal, é um projeto para auxiliar os alunos a desenvolverem suas *soft skills* dentro do contexto do conteúdo que estão estudando em Engenharia de Software. Por isso, a importância de verificar o conteúdo programático da disciplina na Atividade Co2 da Etapa de Contextualização.

Outro ponto a se considerar é ter o cuidado de programar o prazo e especificar o *feedback* a ser dado aos alunos após cada tarefa prevista, conforme indica a Atividade Pr3, pois segundo Cukierman e Palmieri (2014) apontam, tal atitude contribui para o desempenho dos alunos, ajudando-os a alinhar seus objetivos caso surja alguma dificuldade.

Na Atividade Pr4, com as tarefas do projeto formuladas, deve-se definir os prazos de entrega, bem como data de aplicação de cada tarefa, se couber. O primeiro se refere aos prazos para os alunos finalizarem e entregarem cada tarefa, estando, assim, presente em todas as tarefas. Por outro lado, o segundo refere-se à data de aplicação de tarefas desenvolvidas em sala de aula, isto é, atividades como apresentação aos alunos sobre as *soft skills* e apresentação final do projeto desenvolvido por eles, dentre outras, conforme descrito na subseção 3.4.2.1, não excedendo 25% das aulas da disciplina. É válido ressaltar, por fim, que os prazos devem ser condizentes com a tarefa, de forma a proporcionar tempo suficiente para que os alunos a finalizem, pois, caso contrário, os alunos podem se sentir desmotivados com a sensação de que fracassaram (BRITO et al., 2020).

Na Atividade Pr5, o professor deve verificar se existem ferramentas de apoio para auxiliar os alunos durante o desenvolvimento do projeto. Em caso afirmativo, deve divulgá-las aos alunos como sugestão.

Por fim, a Atividade Pr6 consiste na aplicação do projeto definido para os alunos. Logo, isso envolve as Atividades A1 a A3 da visão do aluno, isto é, aula inicial expositiva sobre as *soft skills*, autoavaliação inicial pelos alunos, desenvolvimento do projeto durante a disciplina e autoavaliação final após a intervenção do FraSSD. Para verificar os conhecimentos dos alunos sobre *soft skills* após a intervenção, deve ser aplicado o formulário final descrito no Apêndice F. Tal formulário coletará, também, informações sobre a motivação dos alunos utilizando o PBL em relação a atividades de avaliação do método tradicional de ensino, como provas individuais, além de colher suas percepções e sugestões de melhorias quanto ao modelo seguido durante o semestre, baseado no FraSSD.

3.4.2.3 Métricas

A Etapa de Métricas (Me) tem como objetivo delimitar os critérios para que os alunos alcancem cada nível de pontuação definido para cada tarefa proposta. Além disso, engloba a incorporação dos aspectos da gamificação competição, pontuação e placar na atribuição das notas aos alunos. Esta etapa envolve as atividades Me1, Me2 e Me3, conforme apresentado no Quadro 3.

Quadro 3: Atividades da Etapa de Métricas

Etapa de Métricas (Me)

Me1 - Definir os critérios de pontuação para avaliação das tarefas

Me2 – Definir os pesos de cada tarefa

Me3 - Definir avaliação segundo aspectos da gamificação

Fonte: Desenvolvido pelo autor.

Para cumprir a Etapa de Métricas, pode ser utilizada a 2ª Parte do Apêndice C como modelo, a qual é desenvolvida juntamente com a Etapa de Projetos. Assim, para cada tarefa definida, identifica-se, também, os critérios para que os alunos alcancem cada nível de pontuação, conforme apontado na Atividade Me1. É válido ressaltar, porém, que nem toda tarefa terá necessariamente que ser avaliada, não tendo, portanto, a Etapa de Métricas. Como exemplo, tem-se a apresentação inicial das *soft skills* para os alunos, a qual contempla uma tarefa, porém não avaliativa.

Na Atividade Me1, portanto, são definidos os critérios de pontuação para cada nível apresentado nos trabalhos de Konings e Legg (2020) e Kulkarni et al. (2020), os quais são: insuficiente, regular, bom e muito bom. Tais critérios têm o intuito de direcionar a avaliação das tarefas do projeto, ao mesmo tempo que utiliza o elemento de gamificação pontuação, variando de 0 a 3 pontos. Para definir os critérios em Me1, sugere-se que sejam consideradas as *soft* e *hard skills* trabalhadas em cada tarefa. Dentro desse cenário, para ajudar o professor no estabelecimento de prioridades entre as diferentes *soft skills* que possam ser trabalhadas em uma mesma tarefa, pode-se tomar como base o *ranking* descrito na Figura 2 (subseção 2.3.1), a fim de verificar quais *soft skills* são vistas como mais importantes para Engenheiros de Software. Assim, se os alunos conseguiram cumprir o proposto na tarefa, acessando as *hard* e *soft skills* previstas para isso, estariam no nível ótimo, decrescendo a pontuação conforme alguma *skill* não é totalmente aplicada ou se forem encontrados erros na execução da tarefa proposta.

Além disso, devem ser definidos, em Me2, os pesos que as pontuações das tarefas receberão, os quais devem ser apresentados em porcentagens. Estes, posteriormente, serão utilizados na Etapa de Avaliação para determinar o desempenho dos alunos.

Após isso, na Atividade Me3, o professor deve incorporar os elementos de gamificação competição, pontuação e placar. No levantamento realizado no trabalho de Souza et al. (2018) é discutido que a gamificação ainda é pouco explorada em contextos de ensino na Engenharia de Software. Porém, o estudo aponta que este é um dispositivo eficaz para motivar os alunos a se adequarem aos comportamentos desejados, como o uso mais frequente de ferramentas específicas, adquirir o hábito de aplicar técnicas específicas ou ser mais participativo em sala de aula. Para isso, os trabalhos utilizam de estratégias de premiações, competição e apresentação de placares. Nesse contexto, o trabalho de Maxim et al. (2019) afirma ter aplicado a gamificação montando uma tabela de classificação. Esta classificação, por sua vez, foi alimentada com a pontuação atribuída aos grupos e aos integrantes considerando aspectos como codificação, além de contribuição e gerenciamento do projeto, por exemplo. Em outro trabalho, os autores afirmam terem atribuído, além da pontuação, medalhas aos alunos com melhor desempenho em determinadas áreas analisadas (MAXIM et al., 2017). Porém, salientam que a gamificação não foi formalmente estruturada no projeto proposto.

Assim, no FraSSD, a competição deve ser impulsionada pela história contada com o método da *role-play*, trazendo, por exemplo, um concurso na qual os grupos possam competir. No quesito pontuação e placar, a pontuação deve ser baseada nos níveis atribuídos às tarefas (0 a 3 pontos) e utilizada na atualização do placar, o qual pode ser exibido por meio eletrônico ou

físico, de acordo com os recursos disponíveis. Por fim, podem ser estabelecidas, também, recompensas para cada tarefa ou uma recompensa geral ao final do projeto para fomentar a competição e motivação dos alunos.

3.4.2.4 Avaliação

A Etapa de Avaliação (Av) consiste em avaliar o desempenho e desenvolvimento dos alunos com a aplicação do *framework*. Esta etapa envolve as atividades Av1 e Av2, conforme apresentado no Quadro 4.

Quadro 4: Atividades da Etapa de Avaliação

Etapa de Avaliação (Av)
Av1 – Avaliação somativa
Av2 – Autoavaliação

Fonte: Desenvolvido pelo autor.

Como método avaliativo são consideradas duas abordagens: avaliação somativa realizada com as métricas definidas para cada tarefa (Av1) e percepção quanto ao nível de desenvolvimento de *soft skills* a partir de autoavaliação realizada pelos alunos e aplicação de teste estatístico para análise (Av2).

Na primeira abordagem, a avaliação com as métricas definidas será utilizada para verificar o desempenho dos alunos em cada tarefa e durante todo o projeto. Assim, considerando os pesos e tarefas definidos nas Etapas de Projeto e Métricas, sendo p_1 a p_n os pesos das tarefas t_1 a t_n , de forma que $p_1 + p_2 + \dots + p_{n-1} + p_n = 100$ e sendo t_n e p_n a tarefa atual trabalhada e seu respectivo peso atribuído, tem-se a Fórmula 4, apresentada a seguir. Esta é utilizada para a atualização das notas no placar (NP) no decorrer do projeto, sendo os pesos das tarefas considerados em porcentagem e aplicados no intervalo [0, 100].

$$NP = \frac{(p_1 \times t_1 + p_2 \times t_2 + \dots + p_{n-1} \times t_{n-1} + p_n \times t_n)}{(p_1 + p_2 + \dots + p_n)} \quad (4)$$

Além disso, considera-se a Fórmula 5 como uma média ponderada para cálculo da Nota Final (NF) do projeto proposto baseada na NP, sendo N_{max} o valor máximo dentro do domínio de notas da disciplina e P_{max} o peso que o projeto terá dentro da disciplina, com P_{max} representado como uma porcentagem no intervalo [0, 100]. É válido ressaltar que este cálculo é importante para converter as notas das tarefas (0 a 3) para o domínio da disciplina, o qual, geralmente está no intervalo de [0, 10] ou [0, 100].

$$NF = \frac{P_{max}}{100} * \left(\frac{NP * N_{max}}{3} \right) \quad (5)$$

Por fim, de forma visual, o placar ajudará, além de motivar os alunos, a verificar o desempenho deles.

Na segunda abordagem, serão utilizadas as respostas dos formulários (Apêndices E e F) fornecidas pelos alunos antes e depois da intervenção com o FraSSD para comparar e verificar, por meio de teste estatístico, se os alunos perceberam ter desenvolvido suas *soft skills* após a aplicação do FraSSD. Os resultados ajudarão, na visão do aluno, para reflexão sobre suas *soft skills*, tomando consciência de melhorias ou encontrando pontos que precisam ser trabalhados a fim de se tornarem um melhor profissional da área. Na visão do professor, as respostas serão úteis para avaliar se o FraSSD foi capaz de auxiliar no processo de conciliar o ensino aprendizagem do conteúdo técnico com o desenvolvimento de *soft skills* almejadas para Engenheiros de Software.

Capítulo 4 – Resultados e discussão

4.1 Considerações iniciais

A fim de realizar uma avaliação sobre a adequação do *framework* proposto, é importante que este seja instanciado e aplicado em um contexto real. Além disso, coletar as percepções de alunos e professor contribui para validar pontos positivos do FraSSD, bem como propor melhorias.

Neste contexto, na seção 4.2 é apresentada a instanciação seguida pelos alunos e professor responsável pela disciplina; na seção 4.3 são apresentados os resultados coletados e análise feita após a instanciação do FraSSD; e, por fim, na seção 4.4 é apresentada uma reflexão sobre as avaliações do FraSSD fornecidas tanto por alunos quanto por docentes.

4.2 Instanciação do framework

O FraSSD foi aplicado em uma disciplina semestral de Engenharia de Software ministrada para estudantes universitários vinculados a um curso de Bacharelado em Ciência da Computação. Para instanciação das etapas Contextualização, Projeto e Métricas definidas no *framework*, o modelo descrito no Apêndice C foi utilizado pelo docente responsável, sendo o resultado exibido no Quadro 5.

Quadro 5: Instanciação do FraSSD

1ª Parte - Contextualização

Tempo e recursos disponíveis: a disciplina é semestral, distribuída em aulas semanais de 4h durante 15 semanas, totalizando 60 horas. Dessa forma, reserva-se 15 horas (25%) em sala de aula para aplicação do *framework*, podendo ser distribuídas entre as 15 semanas de aula. Estas horas são reservadas para apresentação pelo professor aos alunos sobre as *soft skills*; apresentação final do projeto desenvolvido pelos alunos, bem como avaliação destes trabalhos pelos pares; momento para o professor apresentar *feedback* aos alunos, conforme necessário; e autoavaliação por meio de formulários. Já as atividades entregáveis do projeto deverão ser realizadas no contraturno da disciplina. Como recursos, a sala conta com projetor e os alunos possuem acesso a laboratórios com computadores e internet, além de conta institucional Google, cujas ferramentas podem ser utilizadas no projeto.

Quantidade de grupos para o projeto: a turma tem 41 alunos matriculados. Considerando que a disciplina tem apenas um docente responsável e esse não terá apoio de monitores, optou-se por limitar o número de grupos em no máximo 10, sendo 9 grupos compostos por 4 alunos e 1 grupo composto por 5 alunos.

Tópicos da disciplina (hard skills): segundo o conteúdo programático, a disciplina possui como objetivo, em linhas gerais, fornecer uma visão geral do processo de desenvolvimento de software por meio de técnicas e metodologias utilizadas durante o ciclo de vida do *software*. Para isso, aborda temas voltados para:

1. Fundamentos da Engenharia de Software
2. Modelos de Processo de Desenvolvimento de Software
3. Desenvolvimento Ágil
4. Gerência de Projetos
5. Processo de Engenharia de Requisitos
6. Arquitetura de Software
7. Estratégias e Técnicas de Teste de Software
8. Evolução do Software

Assim, o projeto proposto deverá ter atividades que permitam aos alunos desenvolver suas *soft skills* a partir das etapas de desenvolvimento de um *software*. Como a disciplina é apenas de um semestre, não se espera que os alunos implementem uma solução de fato, mas que pensem em uma proposta analisando os requisitos, metodologias que seriam utilizadas durante o desenvolvimento, pensamento crítico sobre arquitetura e evolução do *software*. Por fim, os alunos devem pensar em maneiras de apresentar a ideia do produto.

Conhecimento técnico prévio dos alunos: na estrutura curricular do curso, é possível notar que, apesar de existirem algumas disciplinas envolvendo conceitos de programação estruturada e orientada a objetos, a disciplina de Engenharia de Software não possui pré-requisito e representa o primeiro contato dos alunos com os fundamentos da Engenharia de Software. Dessa forma, é importante que as etapas do projeto sigam uma sequência paralela aos assuntos abordados em sala de aula, a fim de que os alunos entendam do que se trata cada atividade e possam, com isso, aplicar o conhecimento teórico estudado durante as aulas.

História (role-play): pensando no cotidiano dos alunos e buscando contextualizar o projeto, definiu-se a história **segundo** a metodologia *role-play*:

“A reitoria da universidade está buscando por propostas de melhorias tecnológicas para o campus. Durante o semestre, grupos formados por 4 a 5 participantes poderão competir

elaborando propostas de melhorias. No decorrer do semestre, os grupos terão a chance de convencer a reitoria de que sua ideia é a melhor por meio de entregas que receberão pontos de 0 a 3. Um placar atualizado pelo professor com as pontuações após cada entrega poderá ser acompanhado por todos, evidenciando a competição entre os grupos. Após isso, os alunos apresentarão seus trabalhos uns para os outros e assumirão, em grupo, o papel de avaliadores da reitoria, avaliando cada proposta dos demais grupos com pontos de 0 a 3, não podendo avaliar o grupo ao qual pertencem. O grupo com maior pontuação acumulada durante todo o projeto será o campeão da competição e terá sua proposta implementada pela reitoria.”

2ª Parte – Projeto e Métricas

Tarefa 1: Contextualização e autoavaliação inicial

Aplicação: Aula 1

Prazo: Aula 2

Descrição: Essa tarefa inicial consiste em aula de apresentação, por parte do professor, do planejamento estabelecido para a disciplina e da definição das *soft skills* que serão trabalhadas durante o desenvolvimento do projeto, salientando a importância delas para a completa formação de um Engenheiro de Software. Além disso, deve ser explicado aos alunos o projeto que será desenvolvido em grupo, explicitando as tarefas que serão realizadas, prazos definidos e ferramentas de apoio. Com isso, os alunos devem se organizar em grupos, respeitando o limite estabelecido para a quantidade de membros em cada grupo. Por fim, deve ser explicado e aplicado o formulário inicial para autoavaliação das *soft skills*. Este formulário será entregue em sala de aula e terá o prazo de uma semana para responder, até Aula 2 – mesmo prazo para indicação da composição dos grupos.

Feedback: não se aplica.

Hard skills: não se aplica.

Soft skills: não se aplica.

Ferramentas: Não se aplica.

Tarefa 2: Apresentação dos grupos

Prazo: Aula 3

Descrição: Nessa tarefa, cada grupo deverá pensar em um nome para seu grupo, montar uma logomarca e apresentar seu grupo e integrantes em um vídeo. Por fim, o grupo deve postá-lo no Google Sala de Aula.

Feedback: Cada grupo deve votar, por meio de formulário disponibilizado pelo professor, nos demais grupos nos quesitos apresentação, criatividade e <i>layout</i>, atribuindo notas de 0 a 3, conforme definido nas métricas. Essa participação dos alunos na votação busca aproximá-los de situações do mundo real, em que eles deverão analisar criticamente os grupos e projetos. O professor também deve atribuir notas de 0 a 3, com base nos mesmos quesitos e métricas estabelecidos para os alunos. Assim, o <i>feedback</i> consiste na apresentação do placar atualizado com as votações dos alunos e professor. A média final será composta pela média aritmética das votações do professor e dos alunos. Caso um grupo não entregue a avaliação dos demais, o mesmo terá descontado 20% da sua média obtida. Por fim, o professor também comentará sobre os vídeos com os alunos em sala de aula, complementando o <i>feedback</i> para os alunos.	Prazo avaliação alunos: Aula 4 Prazo feedback professor: Aula 5
Soft skills: Comunicação, trabalho em equipe, organização, criatividade.	
Hard skills: Não se aplica.	
Ferramentas: Google Sala de Aula² para postagem dos vídeos; Google Formulários³ para votação; Canva⁴ para atualização do placar por parte do professor.	
Métricas para Tarefa 2	
Peso (%): 10	
Insuficiente (0): O grupo não conseguiu passar as informações necessárias sobre os membros do grupo.	
Regular (1): O grupo conseguiu se apresentar, porém com dificuldade em transmitir as informações sobre os membros. Baixa criatividade.	
Bom (2): O grupo expôs as informações necessárias sobre os membros do grupo de maneira clara, porém com pouco entrosamento. Criatividade regular.	
Ótimo (3): O grupo conseguiu se apresentar de forma a ser entendido e passar todas as informações necessárias sobre os membros de forma entrosada. Bastante criativos.	

Tarefa 3: Proposta	Prazo: Aula 6
Descrição: Nessa tarefa, os alunos devem entregar um relatório com sua proposta de solução para a situação apresentada na história seguindo o método <i>role-play</i>. Assim, os alunos devem	

² <https://classroom.google.com/u/0/>

³ <https://docs.google.com/forms/u/0/>

⁴ <https://www.canva.com/>

descrever qual a proposta deles para melhoria do campus. O relatório deve conter contextualização da proposta e como isso contribuirá para o campus. Além disso, deve ter uma seção com uma visão geral dos requisitos almejados.	
Feedback: O <i>feedback</i> será dado pelo professor, o qual pode fornecer comentários e sugestões por meio de texto, encontros presenciais ou à distância, conforme a necessidade do grupo. Além disso, a nota atribuída, definida nas métricas, deve ser contabilizada para atualizar o placar.	Prazo: Aula 8.
Soft skills: Comunicação, trabalho em equipe, organização, liderança, aprendizagem, criatividade.	
Hard skills: Processo de software, Engenharia de requisitos, Metodologia ágil.	
Ferramentas: Google Docs ⁵ para escrever relatório; Google Sala de Aula para postagem do relatório; Canva para atualização do placar por parte do professor.	
Métricas para Tarefa 3	Peso (%): 20
Insuficiente (0): O grupo não conseguiu transmitir a ideia de sua proposta ou não apresenta a contribuição para o campus. Não apresenta análise de requisitos.	
Regular (1): O grupo expôs a proposta, porém com dificuldade de passar a informação. Não foram detalhados a contribuição da proposta para o campus e a análise de requisitos.	
Bom (2): O grupo expôs a ideia de forma satisfatória e utilizando linguagem técnica em alguns momentos necessários, porém com menor nível de detalhamento sobre a contribuição da proposta para o campus. Solução restrita, com poucos requisitos ou baixa inovação.	
Ótimo (3): O grupo conseguiu expor sua ideia de forma a ser entendida, utilizando linguagem técnica quando necessário. Houve detalhamento quanto a contribuição da proposta para o campus. Solução inovadora.	

Tarefa 4: Projeto	Prazo: Aula 10
Descrição: Nessa tarefa, os alunos devem entregar o documento com sua proposta finalizada. Este deve contar com a especificação de requisitos, plano de projeto, projeto do sistema (arquitetura e modelos do sistema) e protótipo de interface. Assim, será possível ter uma visão completa de como será implementada a solução.	
Feedback: O <i>feedback</i> será dado pelo professor da mesma maneira que na Tarefa 3, com comentários e sugestões por meio de texto, encontros	Prazo: Aula 12.

⁵ <https://www.google.com/intl/pt-BR/docs/about/>

presenciais ou a distância, conforme a necessidade do grupo. É importante que essas sugestões os auxiliem na preparação da apresentação para os “avaliadores da reitoria”, conforme a história contada segundo a técnica do <i>role-play</i> . Por fim, deve-se atualizar o placar com a nota recebida na atividade.	
Soft skills: Comunicação, trabalho em equipe, organização, liderança, aprendizagem, criatividade.	
Hard skills: Metodologias para desenvolvimento de sistemas, Engenharia de Requisitos, Gerência de Projetos, Modelos de Sistemas, Arquitetura de Software, Projeto de interface.	
Ferramentas: Google Docs para escrever relatório; Google Sala de Aula para postagem do relatório; Figma ⁶ para projeto de interface; YeD Graphic Editor ⁷ ou Canva para estruturação da arquitetura; Canva para atualização do placar por parte do professor.	
Métricas para Tarefa 4	Peso (%): 25
Insuficiente (0): O grupo não conseguiu se organizar com as etapas. Apresentação insatisfatória da solução nos quesitos especificação de requisitos, plano de projeto, projeto do sistema e protótipo de interface. Apresenta erros conceituais.	
Regular (1): O grupo mostra certa organização, porém não de forma completa. Apresentação regular nos quesitos especificação de requisitos, plano de projeto, projeto do sistema e protótipo de interface. Documentação incompleta.	
Bom (2): O grupo conseguiu planejar e organizar as etapas do projeto especificação de requisitos, plano de projeto, projeto do sistema e protótipo de interface com algumas ressalvas, podendo conter conceitos corretos e documentação correta, porém com escrita ruim ou interface de baixa qualidade.	
Ótimo (3): O grupo conseguiu planejar e organizar todas as etapas de especificação de requisitos, plano de projeto, projeto do sistema e protótipo de interface criticamente. Conseguiu apresentar todas as informações de forma compreensível.	

Tarefa 5: Apresentação e avaliação das propostas

Aplicação: Aulas 13 e 14

⁶ <https://www.figma.com/>

⁷ <https://www.yeworks.com/products/yed>

Descrição: Essa tarefa consiste na apresentação das soluções dos alunos para representantes da reitoria, a qual é composta pelos alunos dos outros grupos e pelo professor. Assim, os alunos devem apresentar suas propostas de solução por meio de slides, vídeos ou qualquer método de preferência deles para tentar convencer os representantes da reitoria de que a sua proposta é a melhor para o campus.	
Feedback: O <i>feedback</i> será fornecido após todas as apresentações, na Aula 14, por meio da votação final, na qual os grupos e professor devem atribuir nota de 0 a 3 para cada proposta. Para tentar evitar que os alunos não votem apenas com interesse próprio, estes não podem votar no próprio grupo e o professor terá seu voto com peso dois na média ponderada dos pontos. Assim, a nota final dessa tarefa para atualização do placar será: Nota final = $\frac{(2 \times N_p + M_a)}{3}$, sendo N_p a nota do professor e M_a a média aritmética das notas dos alunos. Por fim, o <i>feedback</i> do professor para os alunos, bem como apresentação do placar final será na Aula 15.	Prazo votações: Aula 14 Prazo feedback professor: Aula 15.
Soft skills: Comunicação, trabalho em equipe, organização, liderança, aprendizagem, criatividade.	
Hard skills: Metodologias para desenvolvimento de sistemas, Engenharia de Requisitos, Gerência de Projetos, Modelo de Sistemas, Arquitetura de Software, Projeto de interface.	
Ferramentas: Projetor; Google Apresentação para fazer slides; papéis para votos.	
Métricas para Tarefa 5	Peso (%): 25
Insuficiente (0): O grupo mostra falta de organização e fluidez na comunicação. Não consegue debater sobre a relevância de sua proposta.	
Regular (1): O grupo apresenta a proposta com certa falta de organização e fluidez. Apresentação visual de baixa qualidade contendo muitos erros de digitação e/ou informações com visibilidade comprometida.	
Bom (2): O grupo consegue apresentar sua proposta de forma organizada e fluída, porém, encontra dificuldade em debater a relevância da proposta. Apresentação visual com qualidade regular contendo poucos erros de digitação e/ou informações com visibilidade comprometida.	
Ótimo (3): O grupo consegue apresentar sua proposta de forma organizada e fluída. Além disso, consegue debater sobre a relevância de tal proposta para convencer os avaliadores. Apresentação de qualidade.	

Tarefa 6: Autoavaliação final	Aplicação: Aula 15
Descrição: Essa tarefa consiste na aplicação final do formulário para autoavaliação das <i>soft skills</i> , o qual será aplicado em sala de aula.	
Feedback: não se aplica.	
Hard skills: não se aplica.	
Soft skills: aprendizagem.	
Ferramentas: Não se aplica.	

Fonte: Desenvolvido pelo autor.

É válido ressaltar que as Tarefas 1 e 6 não possuem métricas, pois equivalem a contextualização das *soft skills* para os alunos, bem como aplicação inicial e final do formulário de autoavaliação dos alunos, não sendo necessário avaliação. Por outro lado, foi estabelecido pelo docente da disciplina que, além do projeto, teria uma prova individual para avaliar conteúdos não abrangidos pelo projeto proposto, e que a mesma contabilizaria 20% da nota final dos alunos.

Finalizada a instanciação, na Etapa de Avaliação foram consideradas as duas abordagens indicadas no FraSSD: Av1 e Av2. A primeira teve a intenção de avaliar os alunos por meio das pontuações recebidas nas tarefas e compor a nota final recebida com o desenvolvimento do projeto. A segunda consistiu na autoavaliação dos alunos, por meio dos formulários dos Apêndices E e F, se houve desenvolvimento nas *soft skills* trabalhadas. Com os dados coletados, foi aplicado o teste estatístico não-paramétrico de Wilcoxon, com o cálculo do coeficiente r de Cohen, conforme descrito na subseção 2.4.2, para analisar o nível do desenvolvimento das *soft skills*.

Assim, na primeira abordagem, definiu-se o placar com as logomarcas dos grupos oriundos da Tarefa 2. Este foi atualizado pelo professor com as pontuações obtidas pelos grupos a cada tarefa e apresentado à turma conforme prazo previsto para os *feedbacks* das tarefas. As notas foram atualizadas de acordo com a Fórmula 4 (subseção 3.4.2.4). Dessa forma, na Figura 7 é indicado o resultado do placar ao término de todas as tarefas, o qual foi atualizado aplicando os pesos definidos pelo docente nas notas correspondentes às quatro tarefas executadas e que tinham métricas associadas, conforme indicado na Fórmula 6.

$$NP = \frac{(10 \times t_2 + 20 \times t_3 + 25 \times t_4 + 25 \times t_5)}{(10+20+25+25)} \quad (6)$$

O grupo com maior pontuação, conforme história contada no método do *role-play*, terá sua proposta implementada pela reitoria. Além disso, as três primeiras posições do pódio receberam premiações como recompensa. O placar com todas as equipes foi disponibilizado para os alunos no Google Sala de Aula para consulta durante todo o desenvolvimento do projeto, com o intuito de promover a competição e engajamento das equipes.

Figura 7: Placar final com notas das tarefas



Fonte: Desenvolvido pelo autor.

Por fim, a Nota final de cada grupo com o projeto foi usada para compor 80% da nota total dos alunos na disciplina, considerando o domínio de notas sendo $[0, 10]$, conforme descrito anteriormente. Assim, foi montado o placar exibido na Figura 8, cujas notas são calculadas conforme valores explicitados na Fórmula 7.

$$NF = \frac{80}{100} * \frac{NP \times 10}{3} \quad (7)$$

Na segunda abordagem, foi utilizada a aplicação inicial e final dos formulários de autoavaliação descritos nos Apêndices E e F, respectivamente, evidenciando a variação de respostas dos alunos do início do semestre até o final. Esta atividade teve como objetivo fomentar aos alunos a reflexão sobre suas *soft skills*, enquanto para o professor foi capaz de auxiliar no processo de conciliar o ensino aprendizagem do conteúdo técnico com o desenvolvimento de *soft skills* almejadas para Engenheiros de Software. A análise dos dados coletados, bem como aplicação e conclusões do teste estatístico, é apresentada na seção 4.3.

Figura 8: Placar com notas finais do projeto na disciplina



Fonte: Desenvolvido pelo autor.

4.3 Análise dos resultados

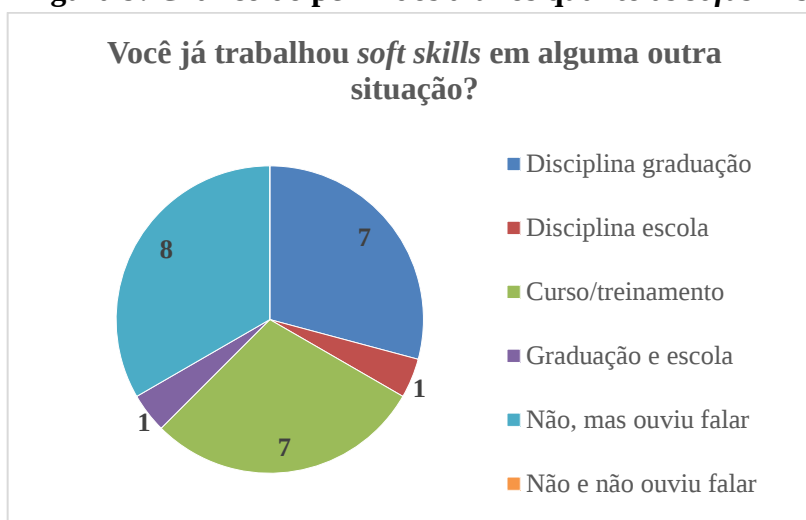
Para análise dos resultados são considerados os dados coletados com a aplicação do formulário inicial, antes da aplicação do FraSSD (Apêndice E), e final, após sua aplicação (Apêndice F). Participaram da pesquisa apenas os alunos que concordaram e assinaram o Termo de Consentimento (Apêndice D) e responderam aos dois questionários, o que totalizou 24 participantes.

Algumas perguntas do formulário inicial (Apêndice E) serviram para identificar o perfil dos alunos, uma vez que abrangeram informações quanto ao contato deles com conceitos de *soft skills* e PBL, além de experiência profissional. Os gráficos das Figuras 9, 10 e 11 exibem resultados sobre o perfil dos alunos. É válido ressaltar que, após esclarecimento adicional pelo docente do que seria o método PBL, alguns alunos reconheceram que, mesmo afirmando ter tido contato com o que pensavam ser o PBL, era apenas a aplicação de um projeto na disciplina, o que não configura o PBL como um todo. Além disso, quanto aos alunos que afirmaram ter experiência profissional na área de Ciência da Computação, um indicou ter 5 meses de experiência, enquanto o outro, 11 meses, até o momento da aplicação do formulário.

Sobre a percepção dos alunos quanto a seu nível de desenvolvimento nas *soft skills*, a primeira pergunta de ambos os formulários pedia para que avaliassem sua percepção na escala Likert de 1 (muito ruim) a 5 (muito bom), em cada uma das seis *soft skills* trabalhadas no projeto. Assim, a partir das respostas de ambos os formulários, foi possível aplicar teste estatístico para verificar se houve aumento no desenvolvimento das seis *soft skills* trabalhadas.

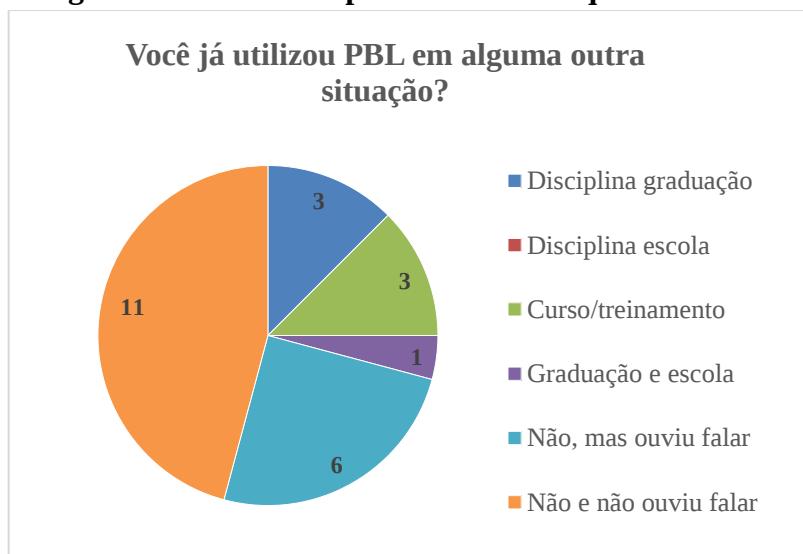
Com base no conteúdo da subseção 2.4.2, o teste estatístico utilizado foi o não-paramétrico de Wilcoxon⁸, uma vez que o tamanho da amostra não alcançou o mínimo de 30 elementos. Diante disso, na Tabela 9 é exibido o resumo estatístico dos dados para cada *soft skill*, além dos valores resultantes do teste de Wilcoxon. O resumo estatístico apresenta valor mínimo (Min), máximo (Max) e percentis, indicando os valores que delimitam 25%, 50% (mediana) e 75% dos dados, enquanto o teste de Wilcoxon apresenta a estatística do teste (z), o valor *p* e o valor do coeficiente *r* de Cohen de tamanho do efeito.

Figura 9: Gráfico do perfil dos alunos quanto às *soft skills*



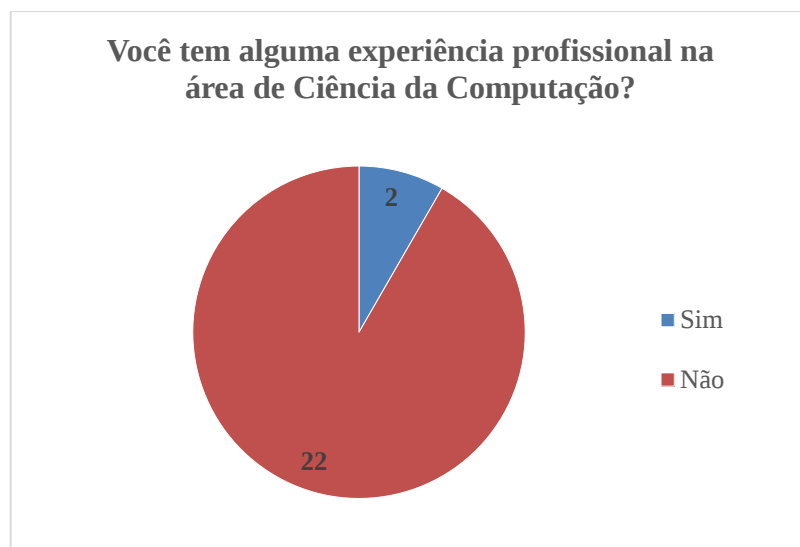
Fonte: Desenvolvido pelo autor.

Figura 10: Gráfico do perfil dos alunos quanto ao PBL



Fonte: Desenvolvido pelo autor.

⁸ Teste realizado com *Software IBM SPSS*. Disponível em: <https://www.ibm.com/br-pt/spss>. Acessado pela última vez em: 22 de outubro de 2023.

Figura 11: Gráfico do perfil dos alunos quanto a experiência profissional

Fonte: Desenvolvido pelo autor.

Tabela 9: Análise do desenvolvimento das soft skills antes e após a aplicação do FraSSD segundo a percepção dos alunos

Soft skill	Formulário	Min	Max	Percentis			Teste de Wilcoxon		
				25%	50%	75%	z	p	r
Comunicação	Inicial	2	5	3	4	4	-2,066	0,039*	0,422 ^b
	Final	1	5	3,5	4	4			
Trabalho em equipe	Inicial	1	4	3	3	4	-2,524	0,012*	0,515 ^c
	Final	1	5	3	4	4,5			
Organização	Inicial	1	5	2	3	3	-3,220	0,001*	0,657 ^c
	Final	1	5	3	4	4			
Liderança	Inicial	1	5	2,5	4	4	-2,288	0,022*	0,467 ^b
	Final	1	5	3	3	4			
Aprendizagem	Inicial	1	5	3	4	4	-2,357	0,018*	0,481 ^b
	Final	1	5	4	4	5			
Criatividade	Inicial	1	5	2,5	3	4	-2,911	0,004*	0,594 ^c
	Final	1	5	3	4	4			
^a pequeno	^b médio	^c grande	* Teste foi estatisticamente significativa ($p < 0,05$)						

Fonte: Desenvolvido pelo autor.

A partir da análise da Tabela 9, nota-se o aumento da mediana nas *soft skills* trabalho em equipe, organização e criatividade, o que mostra uma percepção de aumento no desenvolvimento dessas habilidades, segundo a visão dos alunos. Quanto às *soft skills* comunicação e aprendizagem, que mantém a mediana no mesmo valor antes e após a aplicação,

e liderança, que aparenta ter uma diminuição da mediana, ainda pode-se notar um aumento do desenvolvimento das *soft skills* a partir da observação do intervalo interquartil (25% a 75%), o qual cresce em direção à mediana, indo de 3 para 3,5 em comunicação, de 3 para 4 em aprendizagem e de 2,5 para 3 em liderança. Pode-se dizer, ainda, que esse aumento é estatisticamente significativo para todas as *soft skills* ($p < 0,05$). Dessa forma, conclui-se que todas as *soft skills* apresentaram aumento no nível de desenvolvimento após a aplicação do FraSSD, segundo a percepção dos alunos. Além disso, o coeficiente r de tamanho de efeito mostra que essa diferença estatística é grande nas *soft skills* trabalho em equipe, organização e criatividade e média nas *soft skills* comunicação, liderança e aprendizagem. Assim, elencando-se as *soft skills* pelo seu tamanho de efeito (r), tem-se que as *soft skills* em que os alunos mais conseguiram perceber crescimento pessoal foram, respectivamente: organização, criatividade, trabalho em equipe, aprendizagem, liderança e comunicação.

4.4 Avaliação do FraSSD

Para avaliar o *framework* proposto neste trabalho, o FraSSD, a fim de verificar pontos positivos e receber sugestões de melhorias futuras, consultou-se os dois atores participantes do processo: aluno e professor. No caso dos alunos, estes puderam avaliar alguns recursos incorporados ao FraSSD por meio do Formulário Final (Apêndice F), como a experiência com o método PBL durante a disciplina ou apresentar observações sobre pontos que gostaram, bem como melhorias que consideravam necessárias na maneira de desenvolver o projeto proposto. No caso do professor, contou-se com a participação do professor responsável pela instanciação do FraSSD descrita na seção 4.2, o qual realizou avaliação por meio do Formulário do Apêndice G. Além disso, contou-se com uma revisão de dois professores convidados, os quais, após analisarem o *framework* e simularem o preenchimento do modelo descrito no Apêndice C, sem aplicação real em disciplina, responderam às perguntas: “Foi fácil entender como instanciar o *framework*” e “Foi fácil desenvolver as atividades do *framework* na visão do professor”, avaliando com a Escala Likert de muito ruim (1) a muito bom (5).

Na avaliação dos alunos, houve elogios e sugestões. No caso dos elogios, percebe-se que os alunos se sentiram mais motivados ao trabalhar com o PBL do que com os métodos tradicionais. Eles afirmaram que o método abordado passa uma sensação maior de estar realizando algo que enfrentarão fora da universidade em comparação às provas, além da percepção de que o trabalho contínuo e gradual no projeto auxiliou a desenvolver tanto *hard* quanto *soft skills* de maneira consistente e dinâmica.

Quanto às sugestões, os comentários envolveram aspectos sobre o FraSSD e sugestões que podem auxiliar os docentes na aplicação do *framework* durante a disciplina. Com relação ao FraSSD, os alunos sugeriram: ter uma escala mais dinâmica para avaliação, isto é, ter notas intermediárias entre os valores de Insuficiente a Muito Bom (0 a 3); ter maior detalhamento das métricas estabelecidas, complementando com modelos de documentos a serem adotados. Assim, a partir das observações feitas por alunos, sugere-se para futuras aplicações do FraSSD: (1) que a escala de métricas de 0 a 3 seja ampliada, por exemplo, para uma escala de 1 a 10, contemplando os níveis Insuficiente, Regular, Bom e Muito Bom; (2) incluir uma nova seção em cada tarefa, semelhante a “Ferramentas”, na qual serão informados “Modelos” de documentos, quando couber à tarefa, para auxiliar os alunos.

Quanto aos comentários que podem ser úteis para os docentes considerarem durante a aplicação do FraSSD no decorrer da disciplina, estes se relacionaram, basicamente, a deixar mais explícito como o conteúdo passado em sala de aula se relaciona com as etapas do projeto proposto. Assim, sugere-se tornar a seção de “*hard skills*” do FraSSD acessível aos alunos, uma vez que indica claramente quais os tópicos da disciplina que serão trabalhados naquela tarefa específica.

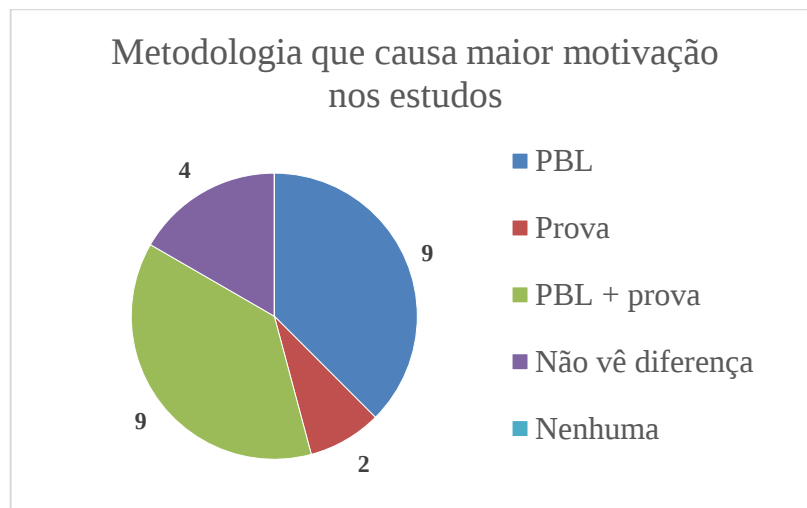
Além disso, os alunos puderam informar no Formulário Final (Apêndice F), qual metodologia de ensino causava maior motivação em seus estudos. Os resultados estão exibidos na Figura 12. Percebe-se que as metodologias PBL e PBL alinhada a provas são as que causam maior motivação nos alunos quanto a seus estudos, em consonância com o modelo adotado no FraSSD. Além disso, apenas dois alunos afirmaram que preferem o método tradicional de ensino com provas, enquanto outros quatro afirmaram não ver diferença entre uma metodologia e outra.

No caso do professor responsável pela disciplina, este respondeu ao formulário do Apêndice G com notas segundo a Escala Likert, variando de muito ruim a muito bom. No geral, o professor achou muito fácil instanciar o FraSSD e notou melhorias nas *soft skills* comunicação, trabalho em equipe, organização, aprendizagem e criatividade, porém notou que nem todos se destacaram na liderança. O professor também afirmou ter sido desafiador garantir o *feedback* para os grupos nos prazos estabelecidos, o que poderia ser melhorado com a definição de um menor número de grupos. Por fim, o professor considerou o método de avaliação eficaz e notou maior motivação na maioria dos alunos com o método empregado.

Já os professores que auxiliaram na revisão do FraSSD indicaram que acharam muito fácil entender como instanciá-lo, bem como desenvolver as atividades previstas na visão do

professor. Como sugestões de melhoria, a primeira delas foi apresentar alguns exemplos de gamificação para deixar mais claro como isso poderia ser abordado na disciplina, o que foi contemplado na subseção 3.4.2.3. Sugeriu-se, também, fornecer uma premiação para os primeiros colocados do *ranking*, o que foi aplicado no final da disciplina de Engenharia de Software. Por fim, sugeriu-se que o FraSSD fornecesse um guia com todas as Etapas de planejamento para serem preenchidas juntas, o que foi contemplado por meio do estabelecimento de uma tabela única abrangendo as etapas de Contextualização, Projeto e Métricas (Apêndice C).

Figura 12: Metodologia que causa maior motivação nos estudos, segundo a visão dos alunos



Fonte: Desenvolvido pelo autor.

Capítulo 5 – Conclusões

5.1 Considerações iniciais

Diante da importância das *soft skills* tanto quanto das *hard skills* para ser um bom profissional de *software*, ter maneiras de se desenvolver tais habilidades desde a graduação mostra-se crucial para formar bons profissionais, preparados para assumir suas funções de forma completa. Assim, ter estratégias que auxiliem os docentes no desenvolvimento de *hard* e *soft skills* junto aos alunos, de forma a mantê-los motivados, é essencial.

Diante deste cenário, nesse trabalho, foi apresentado um *framework*, o FraSSD, para auxiliar os docentes nesse processo de ensino-aprendizagem junto aos alunos. A partir da instanciação do *framework* e análises decorrentes, na seção 5.2 são apresentadas as contribuições deste trabalho; já na seção 5.3 são descritas as limitações identificadas; e, finalmente, na seção 5.4 são apresentadas propostas para trabalhos futuros.

5.2 Contribuições

A RSL realizada neste trabalho abrange conhecimentos relacionados às *soft skills* relevantes para futuros Engenheiros de Software e formas de desenvolvê-las em alunos de graduação, segundo metodologias de ensino. Por meio dos conhecimentos obtidos com este trabalho, foram destacadas a importância de:

- Desenvolver *soft skills*, em paralelo às *hard skills*, desde a graduação para a completa formação dos profissionais de *software*;
- Usar aprendizagem ativa como auxiliador na motivação dos alunos e no desenvolvimento de suas *hard* e *soft skills*.

Com isso, tem-se as seguintes contribuições:

- Identificação das *soft skills* mais relevantes para Engenheiros de Software, organizando-as por frequência de citação nos trabalhos encontrados na literatura. Essa informação é essencial para profissionais da área, a fim de incentivar a busca constante pelo aprimoramento de *soft skills* alinhadas com as *hard skills*;
- Agrupamento de termos de *soft skills*. Como não há consenso na literatura sobre as nomenclaturas das *soft skills*, existem diferentes termos associados à mesma definição. Assim, os termos foram agrupados neste trabalho em 33 *soft skills*, conforme Apêndice B, contribuindo para uma compreensão inequívoca;

- Definição das *soft skills* consideradas mais relevantes para Engenheiros de Software, incluindo dez *soft skills* almejadas para profissionais de software que ainda não estavam claramente definidas na literatura. Isto permite que aqueles que desejam desenvolver as suas *soft skills* possam empreender esforços da forma correta, sem ambiguidade nos termos;
- Levantamento das principais metodologias de ensino utilizadas para o desenvolvimento de *soft skills* em alunos de graduação, organizando-as por metodologias principais e auxiliares, apresentando quais as principais *soft skills* podem ser desenvolvidas com essas metodologias. Além disso, foi apresentada a frequência com que as metodologias são utilizadas nos artigos encontrados. Isto incentiva a incorporação intencional, pelos educadores, de estímulos capazes de desenvolver *soft skills* nos futuros Engenheiros de Software desde a graduação;
- Avaliação do desenvolvimento das *soft skill*. Foi apresentado uma forma de avaliação do desenvolvimento das *soft skills*, na qual foi combinada a autoavaliação, por meio de formulários, e a Taxonomia de Bloom revisada, utilizando testes estatísticos para medir tal desenvolvimento. Uma vez que é encontrada dificuldade para avaliar as *soft skills* na literatura, as estratégias apresentadas mostram-se relevantes para evolução do estado da arte;
- Criação de um *framework* para desenvolver *hard* e *soft skills* em alunos de graduação, com foco em Engenharia de Software. O FraSSD é baseado nas *soft skills* que se mostraram mais buscadas em Engenheiros de Software e nas metodologias encontradas na literatura para o desenvolvimento de *soft skills* em estudantes de graduação, alinhadas a elementos motivacionais. Com isso, é possível auxiliar os docentes responsáveis por disciplina de Engenharia de Software no desafio de incluir *hard* e *soft skills* em um processo de ensino-aprendizagem mais significativo para os alunos, contribuindo para uma formação mais completa dos futuros Engenheiros de Software;
- Aplicação do FraSSD em turma de Engenharia de Software. Isso possibilitou a coleta das percepções dos alunos em uma situação real e a constatação do desenvolvimento de todas as seis *soft skills* trabalhadas durante a disciplina, evidenciando a acurácia do *framework*. Além disso, foi possível coletar informações para melhorias do FraSSD.

Diante do exposto, evidencia-se a contribuição do presente trabalho para o desenvolvimento de *soft skills* relevantes a futuros profissionais de *software*, em conjunto com as *hard skills* comumente previstas para a disciplina Engenharia de Software, tornando o processo de ensino-aprendizagem mais significativo para os estudantes de graduação.

5.3 Publicação

Como resultado da Revisão Sistemática da Literatura realizada neste trabalho, tem-se a seguinte publicação em periódico internacional:

Borges, G. G.; Gratão de Souza, R. C. *Skills development for software engineers: Systematic literature review*. Information and Software Technology, v. 168, 2024. DOI: <https://doi.org/10.1016/j.infsof.2023.107395>.

5.4 Limitações

O FraSSD teve sua avaliação restrita a aplicação real em uma única turma semestral de Engenharia de Software, devido ao tempo para concluir este trabalho. Dessa forma, apesar de ter sido possível tirar conclusões importantes com a experiência vivenciada, sugere-se como melhorias nesse aspecto: (1) o uso de uma amostra maior, isto é, com mais turmas ou, então, (2) comparar o desempenho de uma turma seguindo o método tradicional e outra com a aplicação do FraSSD. Tais abordagens poderiam fornecer dados mais robustos para análise.

Além disso, o FraSSD apresenta metodologias de ensino e *soft skills* pré-definidas. No caso das metodologias de ensino, isso pode ser uma limitação para docentes que apresentem dificuldade de aplicar tais metodologias, devido ao contexto da disciplina ou qualquer outro fator. Logo, a expansão do *framework* de forma a considerar metodologias alternativas, poderia contribuir neste aspecto. Já com relação às *soft skills*, pode ser interessante considerar as individualidades dos alunos para estabelecer quais *soft skills* deveriam ser priorizadas em determinada turma. Para tanto, estudos adicionais deveriam ser feitos para auxiliar os docentes na identificação das *soft skills* almejadas, bem como sobre as metodologias de ensino que poderiam ser empregadas.

5.5 Trabalhos futuros

Como proposta para trabalhos futuros, tem-se como sugestão maior investimento em métodos de avaliação do desenvolvimento das *soft skills*. Uma vez que não há um padrão

definido, trabalhar para apresentação de um modelo de avaliação pode contribuir para outras pesquisas na área.

Além disso, pesquisas que apontem para a preocupação quanto a individualidade dos alunos, incluindo aspectos culturais, no desenvolvimento de *soft skills* mostra-se relevante. Ao considerar as características e objetivos pessoais dos estudantes nas estratégias utilizadas, contribui-se para a motivação dos alunos e, conseqüentemente, para um ensino-aprendizagem mais significativo.

Por fim, no contexto de motivação, estudar outros aspectos motivacionais para o ambiente de sala de aula para o desenvolvimento de *soft skills* pode contribuir para a qualidade de novas propostas voltadas para metodologias de ensino.

Referências

- AL MULHIM, E.; ELDOKHNY, A. **The impact of collaborative group size on students' achievement and product quality in project-based learning environments.** International Journal of Emerging Technologies in Learning (iJET), v. 15, n. 10, p. 157-174, 2020.
- AHMED, F.; CAPRETZ, L.F.; CAMPBELL, P. **Evaluating the Demand for Soft Skills in Software Development.** IEEE Computer Architecture Letters, v. 14, n. 01, p. 44-49, 2012. DOI: <https://doi.org/10.1109/MITP.2012.7>.
- ALVES, I.; ROCHA, C. **Qualifying Software Engineers Undergraduates in DevOps-Challenges of Introducing Technical and Non-technical Concepts in a Project-oriented Course.** 2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET), IEEE, p. 144-153, 2021.
- ANDERSSON, C.; LOGOFATU, D. **Using cultural heterogeneity to improve soft skills in engineering and computer science education.** 2018 IEEE Global Engineering Education Conference (EDUCON), IEEE, p. 191-195, 2018.
- ANGELI, L.; LACONICH, J. J. J.; MARCHESE, M. **A constructivist redesign of a graduate-level cs course to address content obsolescence and student motivation.** Proceedings of the 51st ACM Technical Symposium on Computer Science Education, p. 1255-1261, 2020.
- ASSYNE, N.; GHANBARI, H.; PULKKINEN, M. **The state of research on software engineering competencies: A systematic mapping study.** Journal of Systems and Software, v. 185, p. 111183, 2022.
- BARROS, F. L. F.; BITTENCOURT, R. A. **Evaluating the influence of PBL on the development of soft skills in a computer engineering undergraduate program.** 2018 IEEE Frontiers in Education Conference (FIE), IEEE, p. 1-9, 2018.
- BASTARRICA, M. C.; PEROVICH, D.; SAMARY, M. M. **What can students get from a software engineering capstone course?.** 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET), IEEE, p. 137-145, 2017.
- BRITO, I. S.; BARROS, J. P.; RODRIGUES, E. **Moving to Project-Based Learning at the Program Level: an Experience Report.** 2020 IEEE Global Engineering Education Conference (EDUCON), IEEE, p. 1614-1621, 2020.
- BYRNE, Z.S.; WESTON, J.W.; CAVE, K. **Development of a scale for measuring students' attitudes towards learning professional (ie, soft) skills.** Research in Science Education, v. 50, n. 4, p. 1417-1433, 2020. DOI: <https://doi.org/10.1007/s11165-018-9738-3>.
- CAEIRO-RODRÍGUEZ, M.; MANSO-VÁZQUEZ, M.; MIKIC-FONTE, F. A.; LLAMAS-NISTAL, M.; FERNÁNDEZ-IGLESIAS, M. J.; TSALAPATAS, H.; HEIDMANN, O.; DE CARVALHO, C. V.; JESMIN, T.; TERASMAA, J.; SØRENSEN, L. T. **Teaching soft skills in Engineering education: an European perspective,** IEEE Access, v. 9, p. 29222-29242, 2021.

CAPRETZ, L. F.; AHMED, F.; SILVA, F. Q. B. **Soft sides of software**. Information and Software Technology, v. 92, p. 92-94, 2017.

CARROLL, N. L.; MARKAUSKAITE, L.; CALVO, R. A. **E-portfolios for developing transferable skills in a freshman engineering course**. IEEE Transactions on Education, v. 50, n. 4, p. 360-366, 2007.

CARTER, L. **Ideas for adding soft skills education to service learning and capstone courses for computer science students**. Proceedings of the 42nd ACM technical symposium on Computer science education, p. 517-522, 2011.

CASTRO, R. M.; SIQUEIRA, S. W. M. **Aprendizagem Ativa em Sistemas de Informação: Novas Técnicas Propostas e Reflexões sobre as Experiências**. XIII Simpósio Brasileiro de Sistemas de Informação (SBSI), p. 535-542, 2017. DOI: <https://doi.org/10.5753/sbsi.2017.6085>.

CHEN, C. H.; YANG, Y. C. **Revisiting the effects of project-based learning on students' academic achievement: A meta-analysis investigating moderators**. Educational Research Review, v. 26, p. 71-81, 2019.

COHEN, J. **Statistical power analysis for the behavioral sciences**. Academic press, ed. rev., 1977.

CUKIERMAN, U. R.; PALMIERI, J. M. **Soft skills in engineering education: A practical experience in an undergraduate course**. 2014 International Conference on Interactive Collaborative Learning (ICL), IEEE, p. 237-242, 2014.

DAVIS, J.; REBELSKY, S.A. **Developing soft and technical skills through multi-semester, remotely mentored, community-service projects**. Proceedings of the 50th ACM Technical Symposium on Computer Science Education, p. 29-35, 2019.

DEEP, S.; SALLEH, B. M.; OTHAN, H. **Improving the soft skills of engineering undergraduates in Malaysia through problem-based approaches and e-learning applications**. Higher Education, Skills and Work-Based Learning, 2019.

DUBEY, R. S.; TIWARI, V. **Operationalisation of soft skill attributes and determining the existing gap in novice ICT professionals**. International Journal of Information Management, v. 50, p. 375-386, 2020.

ERSTAD, O. **Norwegian students using digital artifacts in project-based learning**. Journal of Computer Assisted Learning, v. 18, p. 427-437, 2002.

FELIZARDO, K. R.; NAKAGAWA, E. Y.; FABBRI, S. C. P. F.; FERRARI, F. C. **Revisão sistemática da literatura em engenharia de software: teoria e prática**. Brasil: Elsevier, 2017.

FERNANDES, F. G.; MOLLO, R. A. T.; DA COSTA, F. B. **A aplicação de um jogo para motivação do processo de ensino-aprendizagem em cursos de engenharia e ciências exatas**. Revista Temas em Educação, v. 29, n. 2, 2020.

FIORAVANTI, M. L.; SENA, B.; BARBOSA, E. F. **Assessing the Development of Soft Skills for Project Management using PBL: A Case Study**. 2020 IEEE Frontiers in Education Conference (FIE), IEEE, p. 1-8, 2020.

FOREHAND, M. **Bloom's taxonomy**. Emerging perspectives on learning, teaching, and technology, v. 41, n. 4, p. 47-56, 2010.

FREITAS, A.; GARCIA, P.; LOPES, H.; SOUZA A. **Mind the gap: bridging the transversal and transferable skills chasm in a public engineering school**. 2018 3rd International Conference of the Portuguese Society for Engineering Education (CISPEE), IEEE, p. 1-5, 2018.

FRITZ, C. O.; MORRIS, P. E.; RICHLER, J. J. **Effect size estimates: current use, calculations, and interpretation**. Journal of experimental psychology: General, v. 141, n. 1, p. 2, 2012.

GALSTER, M.; MITROVIC, A.; GORDON, M. **Toward enhancing the training of software engineering students and professionals using active video watching**. Proceedings of the 40th international conference on *software engineering*: Software engineering education and training, p. 5-8, 2018.

GARCIA, I.; PACHECO, C.; MÉNDEZ, F.; CALVO-MANZANO, J. A. **The effects of game-based learning in the acquisition of "soft skills" on undergraduate software engineering courses: A systematic literature review**. Computer Applications in Engineering Education, v.28, nº 5, p. 1327-1354, 2020.

GAROUSI, V.; GIRAY, G.; TÜZÜN, E.; CATAL C., FELDERER, M. **Aligning software engineering education with industrial needs: a meta-analysis**. Journal of Systems and Software, v. 156, p. 65-83, 2019. DOI: <https://doi.org/10.1016/j.jss.2019.06.044>.

GIACAMAN, N.; SINNEN, O. **Preparing the software engineer for a modern multi-core world**. Journal of Parallel and Distributed Computing, v. 118, p. 247-263, 2018.

GONZÁLEZ-MORALES, D.; DE ANTONIO, L. M. M.; GARCÍA, J. L. R. **Teaching "Soft" skills in software engineering**. 2011 ieee global engineering education conference (educon), IEEE, p. 630-637, 2011.

GROENEVELD, W.; LUYTEN, L.; VENNEKENS, J.; AERTS, K. **Exploring the Role of Creativity in Software Engineering**. 2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS), IEEE, p. 1-9, 2021. DOI: <https://doi.org/10.1109/ICSE-SEIS52602.2021.00009>.

HAZZAN, O. **Exponential Competence of Computer Science and Software Engineering Undergraduate Students**. 2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET), IEEE, p. 105-109, 2021. DOI: <https://doi.org/10.1109/ICSE-SEET52601.2021.00020>.

HUFFMAN, C. L.; TALLANT, A. C.; YOUNG, S. C.; CHEN, K. **Use of Marketing and Gamification to Promote Participation in Extracurricular Experiences Focused on Transferable Skill Development**. Journal of Applied Learning in Higher Education, v. 8, p. 41-58, 2019.

IBRAHIM, N.; ABD HALIM, S. **Generic framework design of project-oriented problem-based learning (POPBL) for software engineering courses.** 2014 8th. Malaysian Software Engineering Conference (MySEC), IEEE, p. 359-364, 2014.

JOSHI, A.; KALE, S.; CHANDEL, S.; PAL, D. K. **Likert scale: Explored and explained.** British Journal of Applied Science & Technology, v. 7, n. 4, p. 396, 2015.

KHAKUREL, J.; PORRAS, J. **The Effect of Real-World Capstone Project in an Acquisition of Soft Skills among Software Engineering Students.** 2020 IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T), IEEE, p. 1-9, 2020.

KHODEIR, L. M.; NESSIM, A. A. **Changing skills for architecture students employability: Analysis of job market versus architecture education in Egypt.** Ain Shams Engineering Journal, v. 11, n. 3, p. 811-821, 2020.

KOKOTSAKI, D.; MENZIES, V.; WIGGINS, A. **Project-based learning: A review of the literature.** Improving schools, v. 19, n. 3, p. 267-277, 2016.

KONINGS, D.; LEGG, M. **Delivering an Effective Balance of Soft and Technical Skills within Project-Based Engineering Courses.** 2020 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), IEEE, p. 157-164, 2020.

KRATHWOHL, D. **A revision of bloom's taxonomy: An overview.** Theory into Practice, v. 41, n. 4, p. 212-218, 2002.

KULKARNI, S.; PATIL, S.; PAWAR, R. **Adoption of the Conceive-Design-Implement-Operate approach to the Third Year Project in a team-based design-build environment.** Procedia Computer Science, v. 172, p. 559-567, 2020.

KUPPUSWAMY, R.; MHAKURE, D. **Project-based learning in an engineering-design course—developing mechanical-engineering graduates for the world of work.** Procedia CIRP, v. 91, p. 565-570, 2020.

LAM, S. F., CHENG, R. W. Y., CHOY, H. C. **School support and teacher motivation to implement project-based learning.** Learning and Instruction, v. 20, p. 487–497, 2010.

LEVINE, D. M.; BERENSON, M. L.; STEPHAN, D. **Estatística: Teorias e Aplicações usando Microsoft Excel em Português.** Editora Livro Técnicos e Científicos (LTC), 1ª Ed., 2000.

LYZ, N.; LYZ, A.; NESHCHADIM, I.; KOMPANIETS, V. **Blended Learning and Self-Reflection as Tools for Developing IT-Students' Soft Skills.** 2020 V International Conference on Information Technologies in Engineering Education (Inforino), IEEE, p. 1-4, 2020.

MAGANO, J.; SILVA, C. S.; FIGUEIREDO, C.; VITÓRIA, A.; NOGUEIRA, T. **Project management in engineering education: Providing generation Z with transferable skills.** Ieee Revista Iberoamericana De Tecnologias Del Aprendizaje, v. 16, n. 1, p. 45-57, 2021.

MATTESON, M. L., ANDERSON, L. E BOYDEN, C. **“Soft skills’: A phrase in search of meaning”**. Portal: Libraries and the Academy, v. 16, n. 1, p. 71-88, 2016. DOI: <https://doi.org/10.1353/pla.2016.0009>.

MATTURRO, G. **Soft skills in software engineering: A study of its demand by software companies in Uruguay**. 2013 6th international workshop on cooperative and human aspects of software engineering (CHASE), IEEE, p. 133-136, 2013. DOI: <https://doi.org/10.1109/CHASE.2013.6614749>.

MATTURRO, G.; RASCHETTI, F.; FONTÁN, C. **Soft skills in software development teams: A survey of the points of view of team leaders and team members**. 2015 IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering, IEEE, p. 101-104, 2015. DOI: <https://doi.org/10.1109/CHASE.2015.30>.

MATTURRO, G.; RASCHETTI, F.; FONTÁN, C. **A Systematic Mapping Study on Soft Skills in Software Engineering**. J. Univers. Comput. Sci., v. 25, n. 1, p. 16-41, 2019.

MAXIM, B. R.; BRUNVAND, S.; DECKER, A. **Use of role-play and gamification in a software project course**. 2017 IEEE frontiers in education conference (FIE), IEEE, p.1-5, 2017.

MAXIM, B. R.; DECKER, A.; YACKLEY, J. J. **Student Engagement in Active Learning Software Engineering Courses**. 2019 IEEE Frontiers in Education Conference (FIE), IEEE, p. 1-5, 2019.

MILCZARSKI, P.; PODLASKI, K.; HŁOBAŹ, A.; DOWDALL, S.; STAWSKA, Z.; O'REILLY, D. **Soft Skills Development in Computer Science Students via Multinational and Multidisciplinary GameDev Project**. Proceedings of the 52nd ACM Technical Symposium on Computer Science Education, p. 583-589, 2021.

NICOLA, S.; PINTO, C.; MENDONÇA, J. **The role of education on the acquisition of 21st century soft skills by Engineering students**. 2018 3rd International Conference of the Portuguese Society for Engineering Education (CISPEE), IEEE, p. 1-4, 2018. DOI: <https://doi.org/10.1109/CISPEE.2018.8593495>.

PANTHALOOKARAN, V. **Gamification of physics themes to nurture engineering professional and life skills**. 2018 IEEE Global Engineering Education Conference (EDUCON), IEEE, p. 931-939, 2018.

PAWAR, R.; METRI, R.; SAWANT, S.; KULKARNI, S. **Evolving Product Development Skills Through Group Based Activity Instructions in Engineering Exploration Course**. Procedia Computer Science, v. 172, p. 314-323, 2020.

PÉREZ, B.; RUBIO, A. L. **A project-based learning approach for enhancing learning skills and motivation in software engineering**. Proceedings of the 51st ACM technical symposium on computer science education, p. 309-315, 2020.

PINHEIRO, J. I. D.; CARVAJAL, S. S. R.; CUNHA, S. B.; GOMES, G. C. **Probabilidade e Estatística: Quantificando a incerteza**. Editora Elsevier, 1ª Ed, 2012.

PINTO, S. F.; SILVA, P. C. **Gamification applied for software engineering teaching-learning process**. Proceedings of the 31st Brazilian Symposium on Software Engineering, p. 299-307, 2017.

PRASAD, G. N. R. **Evaluating student performance based on bloom's taxonomy levels**. Journal of Physics: Conference Series, v. 1797, n°. 1, IOP Publishing, 2021.

RAMOS, C. S.; KOSLOSKI, R. A. D.; VENSON, E.; DA COSTA FIGUEIREDO, R. M.; DEON, V. H. A. **TBL as an active learning-teaching methodology for software engineering courses**. Proceedings of the XXXII Brazilian Symposium on Software Engineering, p. 289-297, 2018.

RHODES, A.; DANAHER, M.; KRANOV, A. A. **Concurrent direct assessment of foundation skills for general education**. On the Horizon, v. 26, n. 2, p. 79-90, 2018.

RIBEIRO, A. L.; BITTENCOURT, R. A. **A pbl-based, integrated learning experience of object-oriented programming, data structures and software design**. 2018 IEEE Frontiers in Education Conference (FIE), IEEE, p. 1-9, 2018.

SCHIPPER, M.; VAN DER STAPPEN, E. **Motivation and attitude of computer engineering students toward soft skills**. 2018 IEEE global engineering education conference (EDUCON), IEEE, p. 217-222, 2018.

SEDELMAIER, Y.; LANDES, D. **Software engineering body of skills (SWEBOS)**. 2014 IEEE Global Engineering Education Conference (EDUCON), IEEE, p. 395-401, 2014. DOI: <https://doi.org/10.1109/EDUCON.2014.6826125>.

SOBRAL, S. R. **Bloom's Taxonomy to improve teaching-learning in introduction to programming**, 2021. International Journal of Information and Education Technology, v. 11, n° 3, p. 148-153. DOI: [10.18178/ijiet.2021.11.3.1504](https://doi.org/10.18178/ijiet.2021.11.3.1504)

SOUZA, M. R. D. A.; VEADO, L.; MOREIRA, R. T.; FIGUEIREDO, E.; COSTA, H. **A systematic mapping study on game-related methods for software engineering education**. Information and software technology, v. 95, p. 201-218, 2018.

STRAY, V.; FLOREA, R.; PARUCH, L. **Exploring human factors of the agile software tester**. Software Quality Journal, p. 1-27, 2021. DOI: <https://doi.org/10.1007/s11219-021-09561-2>.

TAYLOR, E. **Investigating Soft Skills Development at a Higher Education Institution in South Africa**. Proceedings of the 2019 3rd International Conference on Education and E-Learning, p. 140-146, 2019. DOI: <https://doi.org/10.1145/3371647.3371669>.

VALERO, M. D. R.; REID, T.; DELL, G.; STACEY, D.; HATT, J.; MOORE, Y.; CLIFT, S. **Embedding employability and transferable skills in the curriculum: a practical, multidisciplinary approach**. Higher Education Pedagogies, v. 5, n. 1, p. 247-266, 2020.

VAN LAAR, E.; VAN DEURSEN, A.J.; VAN DIJK, J.A.; DE HAAN, J. **The relation between 21st-century skills and digital skills: A systematic literature review**. Computers in human behavior, v. 72, p. 577-588, 2017.

VIEIRA, S. **Introdução à Bioestatística**. Editora Elsevier, 4ª Ed, 2008.

YOUNIS, A. A.; SUNDERRAMAN, R.; METZLER, M.; BOURGEOIS, A. G. **Developing parallel programming and soft skills: A project based learning approach**. Journal of Parallel and Distributed Computing, v. 158, p. 151-163, 2021.

YOUNIS, A. A.; SUNDERRAMAN, R.; METZLER, M.; BOURGEOIS, A. G. **Case study: Using project based learning to develop parallel programming and soft skills**. 2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), p. 304-311, IEEE, 2019.

ZHENG, G.; ZHANG, C.; LI, L. **Practicing and evaluating soft skills in IT capstone projects**. Proceedings of the 16th annual conference on information technology education, p. 109-113, 2015.

ZMEEV, D. O.; ZMEEV, O. A. **Project-Oriented Course of Software Engineering Based on Essence**. 2020 IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T), IEEE, p. 1-3, 2020.

Apêndice A - Trabalhos selecionados na RSL

#	Autores	Título	Ano	QP
1	Ahmed et al.	Evaluating the Demand for Soft Skills in Software Development	2012	Q1
2	Alves e Rocha	Qualifying Software Engineers Undergraduates in DevOps - Challenges of Introducing Technical and Non-technical Concepts in a Project-oriented Course	2021	Q2
3	Andersson e Logofatu	Using cultural heterogeneity to improve soft skills in Engineering and computer science education	2018	Q2
4	Angeli et al.	A Constructivist Redesign of a Graduate-level CS Course to Address Content Obsolescence and Student Motivation	2020	Q2
5	Barros e Bittencourt	Evaluating the Influence of PBL on the Development of Soft Skills in a Computer Engineering Undergraduate Program	2018	Q2
6	Bastarrica et al.	What can students get from a software engineering capstone course?	2017	Q2
7	Brito et al.	Moving to Project-Based Learning at the Program Level: An Experience Report	2020	Q2
8	Byrne et al.	Development of a Scale for Measuring Students' Attitudes Towards Learning Professional (i.e., Soft) Skills	2020	Q1, Q2
9	Caeiro-Rodríguez et al.	Teaching Soft Skills in Engineering Education: A European Perspective	2020	Q1, Q2
10	Carroll et al.	E-portfolios for developing transferable skills in a freshman engineering course	2007	Q1
11	Carter	Ideas for Adding Soft Skills Education to Service Learning and Capstone Courses for Computer Science Students	2011	Q1, Q2
12	Cukierman e Palmieri	Soft Skills in Engineering Education - A practical experience in an undergraduate course	2014	Q1, Q2
13	Davis e Rebelsky	Developing Soft and Technical Skills Through Multi-Semester, Remotely Mentored, Community-Service Projects	2019	Q2
14	Deep et al.	Improving the soft skills of engineering undergraduates in Malaysia through problem-based approaches and e-learning applications	2019	Q2
15	Fioravanti et al.	Assessing the Development of Soft Skills for Project Management using PBL: A Case Study	2020	Q2

16	Freitas et al.	Mind the gap: bridging the transversal and transferable skills chasm in a public engineering school.	2018	Q1
17	Galster et al.	Toward Enhancing the Training of Software Engineering Students and Professionals Using Active Video Watching	2018	Q2
18	Garcia et al.	The effects of game-based learning in the acquisition of “soft skills” on undergraduate software engineering courses: A systematic literature review	2020	Q2
19	Garousi et al.	Aligning software engineering education with industrial needs: A meta-analysis	2019	Q1, Q2
20	Giacaman eSinnen	Preparing the software engineer for a modern multi-core world	2018	Q2
21	González-Morales et al.	Teaching "soft" skills in Software Engineering	2011	Q1, Q2
22	Groeneveld et al.	Exploring the Role of Creativity in Software Engineering	2021	Q1
23	Hazzan	Exponential Competence of Computer Science and Software Engineering Undergraduate Students	2021	Q1
24	Huffman et al.	Use of Marketing and Gamification to Promote Participation in Extracurricular Experiences Focused on Transferable Skill Development	2019	Q2
25	Ibrahim e Abd Halim	Generic Framework Design of Project-Oriented Problem-Based Learning (POPBL) for Software Engineering Courses	2014	Q2
26	Khakurel e Porras	The Effect of Real-World Capstone Projects in an Acquisition of Soft Skills Among Software Engineering Students	2020	Q1, Q2
27	Konings e Legg	Delivering an Effective Balance of Soft and Technical Skills within Project-Based Engineering Courses	2020	Q1, Q2
28	Kulkarni et al.	Adoption of the Conceive-Design-Implement-Operate approach to the Third Year Project in a team-based design-build environment	2020	Q2
29	Kuppuswamy e Mhakure	Project-based learning in an engineering-design course – developing mechanical- engineering graduates for the world of work	2020	Q2
30	Lyz et al.	Blended Learning and Self-Reflection as Tools for Developing IT-Students’ Soft Skills	2020	Q2
31	Magano et al.	Project management in engineering education: Providing generation Z with transferable skills.	2021	Q2
32	Matturro	Soft Skills in Software Engineering - A Study of Its Demand by Software Companies in Uruguay	2013	Q1

33	Matturro et al.	A Systematic Mapping Study on Soft Skills in Software Engineering	2019	Q1
34	Matturro et al.	Soft Skills in Software Development Teams - A Survey of the Points of View of Team Leaders and Team Members	2015	Q1
35	Maxim et al.	Use of Role-play and Gamification in a Software Project Course	2017	Q2
36	Maxim et al.	Student Engagement in Active Learning Software Engineering Courses	2019	Q2
37	Milczarski et al.	Soft Skills Development in Computer Science Students via Multinational and Multidisciplinary GameDev Project	2021	Q2
38	Nicola et al.	The role of education in the acquisition of 21st-century soft skills by Engineering students	2018	Q2
39	Panthalookaran	Gamification of Physics Themes to Nurture Engineering Professional and Life Skills	2018	Q2
40	Pawar et al.	Evolving Product Development Skills Through Group-Based Activity Instructions in Engineering Exploration Course	2020	Q2
41	Pérez e Rubio	A Project-Based Learning Approach for Enhancing Learning Skills and Motivation in Software Engineering	2020	Q2
42	Pinto e Silva	Gamification applied for Software Engineering teaching-learning process	2017	Q2
43	Ramos et al.	TBL as an active learning-teaching methodology for software engineering courses	2018	Q2
44	Rhodes et al.	Concurrent direct assessment of foundation skills for general education.	2018	Q2
45	Ribeiro e Bittencourt	A PBL-Based, Integrated Learning Experience of Object-Oriented Programming, Data Structures, and Software Design	2018	Q2
46	Schipper and Van Der Stappen	Motivation and attitude of computer engineering students toward soft skills	2018	Q1
47	Sedelmaier e Landes	Software Engineering Body of Skills (SWEBOS)	2014	Q1
48	Souza et al.	A systematic mapping study on game-related methods for software engineering education	2018	Q2
49	Stray et al.	Exploring human factors of the Agile software tester	2021	Q1
50	Taylor	Investigating Soft Skills Development at a Higher Education Institution in South Africa	2019	Q1

51	Valero et al.	Embedding employability and transferable skills in the curriculum: a practical, multidisciplinary approach.	2020	Q1, Q2
52	Van Laar et al.	The relation between 21st-century skills and digital skills: A systematic literature review	2017	Q1
53	Younis et al.	Developing Parallel Programming and Soft Skills: A Project-Based Learning Approach	2021	Q2
54	Younis et al.	Case Study: Using Project-Based Learning to Develop Parallel Programming and Soft Skills	2019	Q2
55	Zheng et al.	Practicing and Evaluating Soft Skills in IT Capstone Projects	2015	Q1, Q2
56	Zmeev e Zmeev	Project-Oriented Course of Software Engineering Based on Essence	2020	Q2

Apêndice B - *Soft skills* para Engenheiros de Software

1. Comunicação (96%)	
Nomenclatura	Referências
Comunicação	(HAZZAN, 2021; GROENEVELD et al., 2021; TAYLOR, 2019; BYRNE et al., 2020; FREITAS et al., 2018; VAN LAAR et al., 2017; SEDELMAIER e LANDES, 2014; AHMED et al., 2012)
Comunicação (oral/escrita)	(CAEIRO-RODRÍGUEZ et al., 2021; KONINGS e LEGG, 2020; GAROUSI et al., 2019; CARTER, 2011)
Apresentação	(MATTURRO et al., 2019; SCHIPPER e VAN DER STAPPEN, 2018; ZHENG et al., 2015)
Habilidades de comunicação	(STRAY et al., 2021; MATTURRO et al., 2015; MATTURRO, 2013)
Comunicação interna e externa	(KHAKUREL e PORRAS, 2020)
Comunicação efetiva	(VALERO et al., 2020; CUKIERMAN e PALMIERI, 2014)
Relatórios escritos	(GONZÁLEZ-MORALES et al., 2011)
2. Trabalho em equipe (74%)	
Nomenclatura	Referências
Trabalho em equipe	(KONINGS e LEGG, 2020; GAROUSI et al., 2019; SCHIPPER e VAN DER STAPPEN, 2018; MATTURRO et al., 2015; ZHENG et al., 2015; MATTURRO, 2013; AHMED et al., 2012; GONZÁLEZ-MORALES et al., 2011; CARTER, 2011)
Colaboração com outros	(VAN LAAR et al., 2017; SEDELMAIER e LANDES, 2014)
Trabalho em equipe e colaboração	(CAEIRO-RODRÍGUEZ et al., 2021; KHAKUREL, PORRAS, 2020)
Trabalhar em times	(VALERO et al., 2020)
Relações interpessoais e trabalho em equipe	(TAYLOR, 2019)
Gerenciamento de times	(FREITAS et al., 2018)
Trabalho efetivo em times	(CUKIERMAN e PALMIERI, 2014)
3. Organização (70%)	
Nomenclatura	Referências
Habilidades organizacionais	(HAZZAN, 2021; KHAKUREL e PORRAS, 2020; SCHIPPER e VAN DER STAPPEN, 2018; MATTURRO et al., 2015; MATTURRO, 2013; AHMED et al., 2012; CARTER, 2011)

Planejamento	(CAEIRO-RODRÍGUEZ et al., 2021; MATTURRO et al., 2019; ZHENG et al., 2015; MATTURRO et al., 2015; MATTURRO, 2013)
Gerenciamento de tempo	(KONINGS e LEGG, 2020; FREITAS et al., 2018; ZHENG et al., 2015)
Gerenciamento de tempo e trabalho	(VALERO et al., 2020)
4. Liderança (65%)	
Nomenclatura	Referências
Liderança	(CAEIRO-RODRÍGUEZ et al., 2021; KHAKUREL E PORRAS, 2020; BYRNE et al., 2020; KONINGS e LEGG, 2020; GAROUSI et al., 2019; MATTURRO et al., 2019; FREITAS et al., 2018; MATTURRO et al., 2015; MATTURRO, 2013; Carter, 2011)
Habilidade de coordenar times	(HAZZAN, 2021)
Gerenciamento de equipe	(CAEIRO-RODRÍGUEZ et al., 2021; MATTURRO et al., 2019)
Liderança e inteligência emocional	(TAYLOR, 2019)
Liderança e supervisão	(GONZÁLEZ-MORALES et al., 2011)
5. Aprendizagem (65%)	
Nomenclatura	Referências
Vontade de aprender	(CAEIRO-RODRÍGUEZ et al., 2021; KHAKUREL E PORRAS, 2020; MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013)
Aprendizado contínuo	(KONINGS e LEGG, 2020; SCHIPPER e VAN DER STAPPEN, 2018; CUKIERMAN e PALMIERI, 2014)
Rápida aprendizagem	(HAZZAN, 2021; MATTURRO et al., 2019; AHMED et al., 2012)
Curiosidade	(STRAY et al., 2021; GROENEVELD et al., 2021)
Aprender	(VAN LAAR et al., 2017)
Aprendizagem	(ZHENG et al., 2015)
6. Criatividade (61%)	
Nomenclatura	Referências
Criatividade	(STRAY et al., 2021; GROENEVELD et al., 2021; CAEIRO-RODRÍGUEZ et al., 2021; KONINGS e LEGG, 2020; VALERO et al., 2020; MATTURRO et al., 2019; VAN LAAR et al., 2017; Carter, 2011; CARROLL et al., 2007)

Inovação	(HAZZAN, 2021; BYRNE et al., 2020; MATTURRO et al., 2019; VAN LAAR et al., 2017)
Criatividade e inovação	(SCHIPPER e VAN DER STAPPEN, 2018)
7. Pensamento crítico (48%)	
Nomenclatura	Referências
Pensamento crítico	(CAEIRO-RODRÍGUEZ et al., 2021; KHAKUREL E PORRAS, 2020; GAROUSI et al., 2019; HAZZAN, 2021; GROENEVELD et al., 2021; TAYLOR, 2019; VAN LAAR et al., 2017; MATTURRO et al., 2019; GONZÁLEZ-MORALES et al., 2011)
Criticidade e reflexão	(VALERO et al., 2020)
Pensamento reflexivo	(CARROLL et al., 2007)
8. Habilidade analítica (43%)	
Nomenclatura	Referências
Análítico e capacidade de resolver problemas	(SCHIPPER e VAN DER STAPPEN, 2018; MATTURRO et al., 2015; MATTURRO, 2013; AHMED et al., 2012)
Análise de problemas	(HAZZAN, 2021; STRAY et al., 2021; SEDELMAIER e LANDES, 2014)
Habilidade analítica	(VALERO et al., 2020; MATTURRO et al., 2019)
Pensamento lógico	(SCHIPPER e VAN DER STAPPEN, 2018)
9. Resolução de problemas (43%)	
Nomenclatura	Referências
Resolução de problemas	(CAEIRO-RODRÍGUEZ et al., 2021; SCHIPPER e VAN DER STAPPEN, 2018; VAN LAAR et al., 2017; Carter, 2011)
Gerenciamento de conflitos	(KHAKUREL E PORRAS, 2020; KONINGS e LEGG, 2020; TAYLOR, 2019; GONZÁLEZ-MORALES et al., 2011)
Resolver problemas complexos	(VALERO et al., 2020)
Resolução de problemas e gerenciamento de conflitos	(MATTURRO et al., 2019)
10. Profissionalismo (39%)	
Nomenclatura	Referências
Profissionalismo	(CARTER, 2011)
Engajamento público e cívico	(BYRNE et al., 2020)
Negociação	(MATTURRO et al., 2019)
Profissionalismo e trabalho ético	(TAYLOR, 2019)

Encontros de negócios	(SCHIPPER e VAN DER STAPPEN, 2018)
Criação de negócios	(FREITAS et al., 2018)
Responsabilidade pessoal e social	(VAN LAAR et al., 2017)
Empreendedorismo	(CUKIERMAN e PALMIERI, 2014)
Empreendimento	(CARROLL et al., 2007)
11. Adaptabilidade (39%)	
Nomenclatura	Referências
Flexibilidade	(KONINGS e LEGG, 2020; VALERO et al., 2020; MATTURRO et al., 2019)
Aberto a mudanças	(HAZZAN, 2021)
Adaptável	(STRAY et al., 2021; CARROLL et al., 2007)
Flexibilidade e adaptabilidade	(CAEIRO-RODRÍGUEZ et al., 2021)
Aberto a mudanças e adaptabilidade	(KHAKUREL E PORRAS, 2020)
Mudança de gerenciamento	(MATTURRO et al., 2019)
12. Autonomia (35%)	
Nomenclatura	Referências
Autonomia	(KONINGS e LEGG, 2020; MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013; CARROLL et al., 2007)
Independência	(HAZZAN, 2021; KHAKUREL E PORRAS, 2020)
Autogerenciamento	(TAYLOR, 2019)
13. Pensamento multidisciplinar (30%)	
Nomenclatura	Referências
Pensamento multidisciplinar	(KONINGS e LEGG, 2020; TAYLOR, 2019; CARTER, 2011)
Adaptabilidade cultural	(BYRNE et al., 2020)
Diversidade de conhecimento e cultura	(VALERO et al., 2020)
Buscar informações de diferentes fontes	(CAEIRO-RODRÍGUEZ et al., 2021; GONZÁLEZ-MORALES et al., 2011)
14. Interpessoal (30%)	
Nomenclatura	Referências

Interpessoal	(KHAKUREL e PORRAS, 2020; MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013; AHMED et al., 2012; CARROLL et al., 2007)
Social	(VALERO et al., 2020)
15. Ética (26%)	
Nomenclatura	Referências
Ética	(CAEIRO-RODRÍGUEZ et al., 2021; KONINGS e LEGG, 2020; VALERO et al., 2020; SCHIPPER e VAN DER STAPPEN, 2018; MATTURRO et al., 2019; CUKIERMAN e PALMIERI, 2014)
16. Motivação (22%)	
Nomenclatura	Referências
Motivação	(KHAKUREL e PORRAS, 2020; MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013; CARTER, 2011)
17. Orientação ao cliente (22%)	
Nomenclatura	Referências
Orientação ao cliente	(KHAKUREL e PORRAS, 2020; MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013)
Gerenciamento com clientes e suas expectativas	(GONZÁLEZ-MORALES et al., 2011)
18. Orientação aos resultados (22%)	
Nomenclatura	Referências
Orientação aos resultados	(KONINGS e LEGG, 2020; MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013)
Evolução/comunicação dos resultados	(GONZÁLEZ-MORALES et al., 2011)
19. Comprometimento (17%)	
Nomenclatura	Referências
Comprometimento	(MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013)
Responsabilidade	(KHAKUREL e PORRAS, 2020)
20. Iniciativa (17%)	
Nomenclatura	Referências
Iniciativa/ proatividade	(CAEIRO-RODRÍGUEZ et al., 2021; MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013)
21. Tomada de decisão (17%)	

Nomenclatura	Referências
Tomada de decisão	(KONINGS e LEGG, 2020; MATTURRO et al., 2019; VAN LAAR et al., 2017; GONZÁLEZ-MORALES et al., 2011)
22. Metódico (13%)	
Nomenclatura	Referências
Metódico	(MATTURRO et al., 2019; MATTURRO et al., 2015; MATTURRO, 2013)
23. Pessoal (13%)	
Nomenclatura	Referências
Pessoal	(SEDELMAIER e LANDES, 2014)
Habilidades pessoais	(GAROUSI et al., 2019)
Inteligência emocional	(TAYLOR, 2019)
24. Atenção (9%)	
Nomenclatura	Referências
Atenção aos detalhes	(STRAY et al., 2021; KONINGS e LEGG, 2020)
25. Trabalhar sob pressão (9%)	
Nomenclatura	Referências
Trabalhar sob pressão	(KHAKUREL E PORRAS, 2020)
<i>Deadline</i> com desafios	(ZHENG et al., 2015)
26. Persistência (9%)	
Nomenclatura	Referências
Persistência	(HAZZAN, 2021)
Capacidade de trabalhar com restrições	(GROENEVELD et al., 2021)
27. Saber ouvir (9%)	
Nomenclatura	Referências
Ouvir	(MATTURRO et al., 2019)
Bom ouvinte	(CAEIRO-RODRÍGUEZ et al., 2021)
28. Linguagem técnica (9%)	
Nomenclatura	Referências
Alfabetização na informação, tecnologia e comunicação técnicas	(VAN LAAR et al., 2017)
Habilidades de pesquisa e comunicação científica	(FREITAS et al., 2018)

29. Consciência Ambiental (9%)	
Nomenclatura	Referências
Consideração ao meio ambiente	(KONINGS e LEGG, 2020; VALERO et al., 2020)
30. Debate (4%)	
Nomenclatura	Referências
Habilidade de debate	(SCHIPPER e VAN DER STAPPEN, 2018)
31. Feedback (4%)	
Nomenclatura	Referências
<i>Feedback</i>	(SCHIPPER e VAN DER STAPPEN, 2018)
32. Gerenciamento do estresse (4%)	
Nomenclatura	Referências
Capacidade de gerenciar o estresse	(MATTURRO et al., 2019)
33. Metacognição (4%)	
Nomenclatura	Referências
Metacognição	(VAN LAAR et al., 2017)

Apêndice C – FraSSD

1ª Parte - Contextualização
Tempo e recursos disponíveis:
Quantidade de grupos para o projeto:
Tópicos da disciplina (<i>hard skills</i>):
Conhecimento técnico prévio dos alunos:
História (<i>role-play</i>):

2ª Parte – Projeto e Métricas		
Tarefa 1	Prazo: Aula x1	Aplicação: Aula y1
Descrição:		
Feedback:		Prazo:
Hard skills:		
Soft skills:		
Ferramentas:		
Métricas para tarefa 1		Peso: (%)
Insuficiente (0):		
Regular (1):		
Bom (2):		
Ótimo (3):		
:	:	:
:	:	:
Tarefa n	Prazo: Aula xn	Aplicação: Aula yn
Descrição:		
Feedback:		Prazo:
Hard skills:		
Soft skills:		
Ferramentas:		
Métricas para tarefa n		Peso: (%)
Insuficiente (0):		
Regular (1):		
Bom (2):		
Ótimo (3):		

Apêndice D – Termo de Consentimento Livre e Esclarecido

Esclarecimentos

Este é um convite para você participar da pesquisa: “**Framework para auxílio no desenvolvimento de soft skills em estudantes de Engenharia de Software**”, que tem como pesquisadoras responsáveis a mestranda do Programa de Pós-Graduação em Ciência da Computação da Universidade Estadual Paulista “Júlio de Mesquita Filho” (UNESP), Giovana Giardini Borges, e a professora Rogéria Cristiane Gratão de Souza. Esta pesquisa será realizada durante o primeiro semestre de 2023, no contexto da disciplina de Engenharia de Software I da UNESP, campus de São José do Rio Preto.

Para tanto, pretende-se coletar, por meio de um questionário, as informações referentes à autoavaliação quanto ao nível de *soft skills*, bem como informações sobre interação prévia do participante com *soft skills* e com a metodologia ativa de ensino denominada PBL (*Project Based Learning*), além de experiência no mercado de trabalho na área de Engenharia de Software. Tem-se como objetivo avaliar a eficácia do *Framework* proposto em projeto de mestrado em andamento quanto ao desenvolvimento das *soft skills*, almejadas para Engenheiros de Software, durante a graduação.

Caso você decida participar, serão fornecidos dois questionários para serem respondidos, um antes do início da aplicação do *framework* e um após o término. Estes poderão ser respondidos durante horário reservado em sala de aula, no período da disciplina, ou em momento oportuno ao participante dentro de um período estipulado pelas pesquisadoras responsáveis.

Durante a realização do teste e preenchimento do questionário, a previsão de riscos é mínima, e correspondem ao cansaço inerente de se responder a um questionário. Embora o tempo estimado para resposta ao questionário não deva ultrapassar 10 minutos. Caso você se sinta cansado e deseje continuar em outro momento, será possível continuar posteriormente, dentro do período estipulado; caso não queira continuar, é possível interromper o preenchimento do questionário, sem prejuízo a você.

O benefício de participar desta pesquisa é o autoconhecimento quanto ao seu nível de *soft skills*. Além disso, contribui-se para melhorias no *framework* sendo proposto como uma solução para uma lacuna encontrada na literatura sobre o tema.

Para você participar da pesquisa, você deverá ser maior de 18 anos e estar matriculado na disciplina de Engenharia de Software I, disciplina de graduação do 1º semestre da UNESP, campus de São José do Rio Preto. Não haverá nenhum gasto por sua parte para a realização desta pesquisa e você também não receberá nenhum valor ou nota para a participação.

Caso sofra algum dano comprovadamente decorrente desta pesquisa, você poderá ser indenizado, também cabendo-lhe atendimento imediato e a posterior, nos termos da lei.

Durante todo o período da pesquisa você poderá tirar suas dúvidas com as pesquisadoras responsáveis:

- Giovana Giardini Borges (giovana.giardini@unesp.br)
- Rogéria Cristiane Gratão de Souza (rogeria.souza@unesp.br)

Você tem o direito de se recusar a participar ou retirar seu consentimento, em qualquer momento da fase de coleta de informações para a pesquisa, sem nenhum prejuízo para você.

Os dados que você irá fornecer serão confidenciais e serão divulgados apenas em congressos ou publicações científicas, não havendo divulgação de nenhum dado que possa lhe identificar. Esses dados serão guardados pela pesquisadora Giovana Giardini Borges em local seguro e por um período de 5 anos.

Este documento estará em duas vias. Uma ficará com você e a outra com a pesquisadora Giovana Giardini Borges.

Caso queira ter acesso às informações desta pesquisa, após a análise dos resultados, ou queira algum esclarecimento sobre os resultados da pesquisa, você poderá entrar em contato com as pesquisadoras responsáveis.

Consentimento Livre e Esclarecido

Após ter sido esclarecido sobre os objetivos, importância e o modo como os dados serão coletados nessa pesquisa, além de conhecer os riscos e benefícios que ela poderá trazer para mim e ter ficado ciente de todos os meus direitos, concordo em participar da pesquisa **“Framework para auxílio no desenvolvimento de soft skills em estudantes de Engenharia de Software”**, e autorizo a divulgação das informações por mim fornecidas em congressos e/ou publicações científicas desde que nenhum dado possa me identificar.

(Assinatura do participante)

Declaração do pesquisador responsável

Como pesquisador responsável pelo estudo **“Framework para auxílio no desenvolvimento de soft skills em estudantes de Engenharia de Software”**, declaro que assumo a inteira responsabilidade de cumprir fielmente os procedimentos e direitos que foram esclarecidos e assegurados ao participante desse estudo, assim como manter sigilo e confidencialidade sobre a identidade do mesmo.

Declaro ainda estar ciente que na inobservância do compromisso ora assumido estarei infringindo as normas federais de ética, que regulamenta as pesquisas envolvendo o ser humano.

(Assinatura pesquisadora responsável)

(Assinatura pesquisadora responsável)

Apêndice E – Formulário inicial de avaliação quanto ao nível das *soft skills*

1 - Tendo a escala de 1 a 5 definida, respectivamente, como “Muito ruim” a “Muito bom”, assinale com um X qual a sua percepção pessoal sobre o seu nível atual de desenvolvimento envolvendo cada uma das *soft skills* listadas a seguir.

	1	2	3	4	5
Comunicação					
Trabalho em equipe					
Organização					
Liderança					
Aprendizagem					
Criatividade					

2 - Você já trabalhou *soft skills* em alguma outra situação?

- () Sim, em outra disciplina de graduação. Descreva qual(is): _____
- () Sim, em uma disciplina da escola. Descreva qual(is): _____
- () Sim, durante curso/treinamento.
- () Nunca trabalhei, mas já tinha ouvido falar.
- () Nunca trabalhei *soft skills* nem tinha ouvido falar sobre.

3 - Você já utilizou PBL (*Project Based Learning* - Aprendizagem Baseada em Projetos) em alguma outra situação?

- () Sim, em outra disciplina de graduação. Descreva qual(is): _____
- () Sim, em uma disciplina da escola. Descreva qual(is): _____
- () Sim, durante curso/treinamento.
- () Nunca utilizei, mas já tinha ouvido falar.
- () Nunca utilizei PBL nem tinha ouvido falar sobre.

4 - Você tem alguma experiência profissional na área de Ciência da Computação?

- () Sim. Quanto tempo? _____
- () Não

Apêndice F – Formulário final de avaliação quanto ao nível das *soft skills*

1 - Tendo a escala de 1 a 5 definida, respectivamente, como “Muito ruim” a “Muito bom”, assinale com um X qual a sua percepção pessoal sobre o seu nível atual de desenvolvimento envolvendo cada uma das *soft skills* listadas a seguir.

	1	2	3	4	5
Comunicação					
Trabalho em equipe					
Organização					
Liderança					
Aprendizagem					
Criatividade					

2 - Quanto a sua motivação aos estudos, quando comparando o uso do PBL (*Project Based Learning* - Aprendizagem Baseada em Projetos) com a aplicação de avaliação tradicional, como as provas individuais, assinale a alternativa que melhor reflete sua opinião.

- ☐ () Sinto-me mais motivado utilizando o PBL.
- ☐ () Sinto-me mais motivado quando preciso fazer provas.
- ☐ () Sinto-me mais motivado quando a disciplina utiliza ambas as abordagens: PBL e prova.
- ☐ () Não vejo diferença entre as duas abordagens (PBL e prova) quanto a minha motivação aos estudos.
- ☐ () Não me sinto motivado com nenhuma das abordagens (PBL e prova).

3 - Se tiver algum comentário sobre sua experiência utilizando o PBL durante a disciplina Engenharia de Software I ou alguma sugestão de melhoria que queira compartilhar, descreva a seguir.

Apêndice G – Formulário de avaliação pelo professor quanto ao FraSSD

Para as questões de I a VI, responda em uma escala Likert de concordância, variando de 1 a 5, sendo 1 “discordo plenamente” e 5, “concordo plenamente”.

I – Foi fácil entender como instanciar o *framework*:

II – Foi fácil desenvolver as atividades do *framework* na visão do professor:

III – Foi fácil desenvolver as atividades do *framework* na visão do aluno:

IV – O método de avaliação do desempenho dos alunos foi eficaz:

V – Notei que os alunos se mantiveram mais motivados durante o processo, comparado aos métodos tradicionais de ensino:

VI – Notei melhoria em alguma, ou várias, *soft skills* dos alunos:

Descreva se tem alguma observação ou sugestão de melhoria:
