



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA BAHIA
Campus Porto Seguro
Licenciatura em Computação

RAYANA RIBEIRO BONFANTI

LEARCOD: DESIGN DE UM SOFTWARE EDUCATIVO PARA O APRENDIZADO E AVALIAÇÃO PROCESSUAL NO ENSINO DE LINGUAGEM DE PROGRAMAÇÃO

Porto Seguro – BA
2022



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA BAHIA
Campus Porto Seguro
Licenciatura em Computação

RAYANA RIBEIRO BONFANTI

LEARCOD: DESIGN DE UM SOFTWARE EDUCATIVO PARA O APRENDIZADO E AVALIAÇÃO PROCESSUAL NO ENSINO DE LINGUAGEM DE PROGRAMAÇÃO

Trabalho de Conclusão de Curso apresentado ao Instituto Federal de Educação, Ciência e Tecnologia da Bahia (IFBA), Campus Porto Seguro, como requisito parcial à obtenção do título de Licenciado em Computação.

Orientador: Prof. Me. Josualdo Junior Dias da Silva

Porto Seguro – BA
2022

FOLHA DE APROVAÇÃO

RAYANA RIBEIRO BONFANTI

LEARCOD: DESIGN DE UM SOFTWARE EDUCATIVO PARA O APRENDIZADO E AVALIAÇÃO PROCESSUAL NO ENSINO DE LINGUAGEM DE PROGRAMAÇÃO

Trabalho de Conclusão de Curso aprovado como requisito parcial à obtenção do título de Licenciada em Computação do Instituto Federal de Educação, Ciência e Tecnologia da Bahia, Campus Porto Seguro, pela seguinte Banca examinadora:

Prof. Me. Josualdo Junior Dias da Silva (IFBA/Campus Porto Seguro)
Orientador

Profª. Dra. Roberta Gondim Britto (IFBA/Campus Porto Seguro)
Membro Examinador

Profª. Me. Kênia Xavier Teodoro Oliveira (IFBA/Campus Porto Seguro)
Membro Examinador

Porto Seguro – BA, 27 de junho
2022

RESUMO

O modelo tradicional de ensino, normalmente mais presente até hoje, aquele em que o professor baseia a ideia de ensinar e aprender na pedagogia da transmissão, em que há pouca interação para o estudante. O uso das Novas Tecnologias Digitais de Informação e Comunicação (NTDIC's) em salas de aulas e/ou fora dela tem sido pouco desenvolvida, para apoiar e auxiliar no ensino e aprendizado. No que se refere as áreas como de Exatas, Computação e afins, o ensino e aprendizagem apresenta altos índices de reprovação e evasão, como também dificuldades no processo de avaliação. Com isso, existe a necessidade de repensar métodos para o ensino e avaliação, visando integrar as NTDIC's no aprendizado e avaliação para os discentes e docentes. O período da pandemia de COVID-19 devido o coronavírus SARS-CoV-2, impossibilitou as aulas presenciais, assim tornando o ensino remoto presente, e, de certa forma dificultando o desenvolvimento de habilidades ligadas ao Ensino de Linguagem de Programação. Esta monografia apresenta a proposta de desenvolver o design de um *software* educativo, a partir da prototipação de telas, para o aprendizado e avaliação processual no ensino de linguagem de programação. O trabalho foi articulado em Fase Exploratória e Estruturação e Desenvolvimento, por meio de características da pesquisa-ação. Partindo de um referencial bibliográfico, que permitiu pensar sobre um aprendizado e processo de avaliação prático por meio de um aplicativo, seguido por Observação Documental em pesquisas, produção de instrumentos de coletas de dados, prototipação das telas e avaliação do *design* do *software* educativo *learcod* por usuários discentes e docentes. Os resultados de viabilidade mostraram serem viáveis e aceitáveis para o direcionamento da continuação do *design* do *software* a partir de um fluxo de atividades e, por conseguinte a prototipação de telas, bem como a coleta e análise dos dados de usabilidade demonstraram boa experiência de usabilidade pelos usuários, apesar de precisar de mais ajustes na prototipação, como por exemplo a acessibilidade. Por fim, no geral as características do aplicativo trazem grandes interesses para os docentes e discentes em seu processo de avaliação e aprendizado para ser possivelmente usado com frequência em sala de aula ou fora dela após a sua implementação.

Palavras-chave: Aplicativo Educativo; *Design Thinking*; Programação; Avaliação; Ensino e Aprendizagem.

ABSTRACT

The traditional teaching model, usually more present today, is one in which the teacher is based on an idea of teaching and learning in the pedagogy of transmission, in which there is little interaction for the student. The use of New Digital Information and Communication (NTDI's) in classes and/or technologies has been little developed, for operations and to assist in teaching and learning rooms. With regard to the areas of Exact Sciences, Computing and the like, teaching and learning has high rates of failure and evasion, as well as difficulties in the evaluation process. With this, there is a need to rethink methods for teaching and assessment, integrating as NTDIC's in learning and assessment for students and teachers. The period of the COVID-19 pandemic due to the SARS-CoV-2 coronavirus due to the SARS-CoV-2 coronavirus as face-to-face classes, as well as remote teaching, making it impossible to develop skills combined with Programming Language Teaching. This monograph presents a proposal to develop the design of an educational software, from the prototyping of screens, for learning and procedural evaluation in the teaching of programming language. The work was articulated in Exploratory Phase and Structuring and Development, through action characteristics. Starting from the bibliography and the evaluation process of practical instruments for data collection and evaluation of the data collection software, prototype of the data collection software and evaluation of the project design. students and faculty users. The results of analysis of the viable and acceptable flow of studies for directing the continuation of the design and, from a data prototyping and, as well as a prototyping of data and usage screens, a good usability experience for users, regardless of needing more settings in prototyping, such as accessibility. Finally, there is no way the app's features and great ratings are of general interest to its teachers and learning to be used possibly frequently in the classroom or outside of it after its implementation.

Palavras-chave: Educational Application; *Design Thinking*; Programming; Evaluation; Teaching and Learning.

LISTA DE ABREVIATURAS E SIGLAS

PC	Pensamento Computacional
NTDIC's	Novas Tecnologias Digitais de Informação e Comunicação
DT	<i>Design Thinking</i>
CT	<i>Computational Thinking</i>
PBL	<i>Problem Based Learning</i>
ABP	Aprendizagem Baseada em Problemas
STI	Sistema Tutor Inteligente
POO	Programação Orientada a Objetos
QUIS	<i>Questionnaire for User Interaction Satisfaction</i> - Questionário de Satisfação de Interação do Usuário
SUMI	<i>Software Usability Measurement Inventory</i> - Inventário de Medição de Usabilidade de Software
WAMMI	<i>Website Analysis and MeasurMent Inventory</i> - Análise de Website e Inventário de Mensuração
SUS	<i>System Usability Scale</i> - Escala de Usabilidade do Sistema

LISTA DE FIGURAS

Figura 1. Ciclo Adaptado da Pesquisa-Ação (Feitor, Filho e Souza, 2013).	35
Figura 2. Fluxo de Atividade da Explicação de Algoritmo.....	45
Figura 3. Fluxo de Atividade do Exercício de Assinalar de Algoritmo.....	45
Figura 4. Fluxo de Atividade do Exercício de Codificação de Algoritmo.....	46
Figura 5. Fluxo de Atividade do Exercício de Codificação de Banco de Dados.....	46
Figura 6. Tela Inicial do Aplicativo Learcod para os Usuários.	49
Figura 7. Tela de Estatística de Dados do Usuário.	51
Figura 8. Tela de Configurações de Perfil do Usuário.	52
Figura 9. Tela de Explicação Antes do Exercício para o Usuário.	53
<i>Figura 10. Telas dos Exercícios Existentes de Algoritmo para o Usuário.....</i>	<i>55</i>
Figura 11. Tela de Criação de Turma pelo Docente.....	57
Figura 12. Tela de Cadastro de Estudante por Turma pelo Docente.....	57
Figura 13. Tela de Adicionar Desafios pelo Docente.....	58
Figura 14. Tela de Descrição dos Campos de Adicionar Desafio pelo Docente.	59
Figura 15. Funcionalidade de Acessibilidade na Prototipação do Learcod.....	66
Figura 16. Facilidade de Uso da Prototipação do Learcod.....	66
Figura 17. Experiência Prévia para uso da Prototipação do Learcod.	67
Figura 18. Compreensão da Informação para uso da Prototipação do Learcod.....	67
Figura 19. Aprendizado de Linguagens de Programação pelo uso da Prototipação do Learcod.....	68
Figura 20. Satisfação Geral no uso da Prototipação do Learcod.	69

LISTA DE QUADROS

Quadro 1. Análise do Levantamento de Viabilidade dos Docentes.	40
Quadro 2. Análise do Levantamento de Viabilidade dos Discentes.....	41
Quadro 3. Análise da Avaliação da Usabilidade dos Docentes.	62

TRAJETÓRIA ACADÊMICA

Estudante do Ensino Médio e Técnico em Informática do Instituto Federal da Bahia do *Campus* Porto Seguro, ingressei em 2012, em que pude além de ter a oportunidade de concluir meu ensino médio, ter diversos componentes curriculares, como Linguagens de Programação Estruturadas, Orientadas a Objetos, Modelagem de Dados, Banco de Dados, Redes de Computadores, dentre outros. Nesse período eu pude vivenciar a interdisciplinaridade da Física que era uma disciplina bem atrativa para mim, com a programação, fazendo linhas de código que pudessem criar fórmulas de Física em formas de código para gerar resultados.

Com esse interesse em programação me fez ingressar no curso de Licenciatura em Computação no próprio *Campus* Porto Seguro do Instituto em 2017.1 que além dos conhecimentos técnicos, pude vivenciar conhecimentos da área Pedagógica. Logo nos primeiros semestres foram diversos componentes relacionados a pedagogia, o que me proporcionou várias reflexões sobre além dos conhecimentos técnicos, e sim o pensar, construir e ensinar.

Além dos componentes curriculares que tive muito conhecimento, os projetos que participei, estágios/componentes/projetos extracurriculares, esses puderam trazer a prática que desenvolveu toda a minha formação. Logo no primeiro semestre iniciei no Programa Institucional de Bolsas de Iniciação à Docência (PIBID), que pude vivenciar aulas presenciais no Colégio Modelo Luís Eduardo Magalhães, tendo algumas dificuldades, pois não tínhamos muitos computadores e a nossa proposta era trazer a tecnologia para as salas de aulas, mas trouxemos ideias que pudessem ajudar em todo o processo.

Outra experiência que tive foi no componente curricular de Metodologias e Práticas de Ensino III, na qual nossa turma pudemos construir uma oficina com acessibilidade com jogos lúcidos que contemple o ensino de Computação por meio do Pensamento Computacional, tivemos a presença de um colega de turma que possui deficiência visual para essa dinâmica e toda interação foi feita entre os estudantes de forma que a inclusão fosse realizada. Essa produção da oficina pode trazer uma publicação de artigo na SBIE em 2020 no contexto de Educação em

Computação, trazendo assim mais conhecimentos e experiências da parte acadêmica.

Também tive a oportunidade de participar como bolsista no Residência Pedagógica em 2021 no período da pandemia de COVID-19 devido o coronavírus SARS-CoV-2, em que as atividades foram realizadas remotamente. Dessa forma, as formações foram introduzidas de forma que pudessem feitas por meio de cursos, oficinas, palestras, também passamos por várias reuniões de reflexões para repensar metodologias que agora teriam que ser produzidas via online, e, todas essas práticas foram realizadas me fizeram perceber a grande dificuldade que tive nesse caminho, mas que sempre pensava novas ideias para melhorar posteriormente algum ponto da aula.

A minha experiência também como monitora em Linguagem de Programação I no Ensino Médio Técnico em Informática, mas também realizei monitorias para o primeiro semestre do curso de Licenciatura em Computação, me fizeram pensar algumas dificuldades de aprendizagem em programação, principalmente na Graduação, como por exemplo a estrutura base de matemática, percebia que eles não conseguiam fazer contas simples de regra de três ou algum tipo de proporção lógica matemática, cálculos que envolvem porcentagem, ou seja, cálculos simples de matemática que são do ensino básico se tornava uma dificuldade para eles, e isso eu tinha que para explicar a matemática antes de explicar a programação. Outro ponto também que percebi foi a interpretação textual, eles não conseguiam entender o que o texto pedia ou até mesmo estruturar em palavras o que eu dizia em palavras só que em ordem, assim também dificultando a parte de leitura e escrita antes mesmo do entendimento da programação.

Além de todas essas questões, tinha também um estudante que possui esquizofrenia e com isso também se tornou mais complexo a forma com que eu tinha que explicar o conteúdo, pois a programação para ele também era complexa, pois além das habilidades matemáticas e interpretação textual precária, a programação se tornava confusa, então eu tinha que voltar e explicar em português em forma de lógica textual ou matemática para que pudesse compreender melhor, para que assim pudesse transpor para a programação, então dessa forma ele conseguiu compreender um pouco melhor o conteúdo, assim percebendo que os conteúdos da matemática e

a interpretação textual foi fundamental para que pudesse repassar para o entendimento da programação.

Nesse contexto, foi que percebendo todas essas variáveis, tentei pensar em alguma ideia de tecnologia que pudesse auxiliar esse caminho dos estudantes no seu aprendizado, para que não seja tão árduo e que também para os professores possam avaliar de forma contínua e formativa esse processo de ensino dos estudantes, para que seja um conjunto. Portanto, esse meu trabalho se baseia através de toda minha vivência que tenho tido e visto das dificuldades dos estudantes e tentado perceber algumas variáveis que poderiam ser agregadas em alguma ferramenta para auxiliar nesse processo.

SUMÁRIO

1	Introdução	14
1.1	Problema e Justificativa	15
1.2	Objetivos	16
1.3	Organização do Trabalho	16
2	Fundamentação Teórica	17
2.1	Ensino Básico: antes de iniciar os estudos de programação	17
2.2	Aprendizagem e Avaliação	22
2.3	<i>Design de Software</i>	25
2.3.1	<i>Software de Prototipagem</i>	27
2.3.2	Acessibilidade	29
2.4	Trabalhos Relacionados	30
2.4.1	Considerações acerca dos Trabalhos Relacionados	32
3	Percurso Metodológico	34
3.1	Caracterização da Pesquisa	34
3.2	Fase Exploratória	36
3.3	Estruturação e Desenvolvimento	37
3.3.1	Observação Documental	37
3.3.2	Questionário de Viabilidade	39
3.3.3	Avaliação de Viabilidade com Docentes	39
3.3.4	Avaliação de Viabilidade com Discentes	41
3.3.5	Fluxo de Atividade do Aplicativo Educativo	42
3.3.6	Considerações da escolha do Software de Prototipagem	47
3.3.7	Prototipação	48
3.3.7.1	Prototipação do Perfil de Usuário Discentes	48
3.3.7.2	Prototipação do Perfil de Usuário Docentes	56
3.4	Aplicação do Questionário de Usabilidade	59
4	Resultados e Discussões	61
4.1	Avaliação da Experiência de Usuário com Docentes	61
4.2	Avaliação da Experiência de Usuário com Discentes	65
5	Considerações Finais	71
5.1	Trabalhos Futuros	72
	Referências Bibliográficas	73

Apêndice A – Questionário do Levantamento sobre Viabilidade Tecnológica para os Discentes.....	77
Apêndice B – Questionário do Levantamento sobre Viabilidade Tecnológica para os Docentes	82
Apêndice C – Questionário de Usabilidade da Prototipagem do Aplicativo <i>Learcod</i> para Discente	86
Apêndice D – Questionário de Usabilidade da Prototipagem do Aplicativo <i>Learcod</i> para Docente	89
Apêndice E – Protótipo da Sequência Didática de Algoritmo	92
Apêndice F – Protótipo da Sequência Didática de Banco de Dados.....	93

1 Introdução

No processo histórico do ensino, há propostas de metodologias ativas, que induz o discente a estar mais presente nas atividades de forma autônoma, engajada, protagonista e praticante da ação do assunto abordado, e, momentos de reflexão do docente, que contribui para ajustes dessas atividades aos discentes. O uso constante de redes sociais, possibilitou que a tecnologia aos poucos se tornasse mais normal na vida cotidiana das pessoas, ditas como nativas digitais. Com isso, é clara a percepção da necessidade de utilizar técnicas, metodologias e métodos com a tecnologia computacional, como o uso de equipamentos, *softwares* que auxiliem o desenvolvimento da avaliação e aprendizagem ativa do discente pelo docente, que de acordo com Pinto (2005), tecnologia seria,

[...] a teoria, a ciência, o estudo, a discussão da técnica, abrangidas nesta última noção as artes, as habilidades do fazer, as profissões e, generalizadamente, os modos de produzir alguma coisa (Pinto, 2005, p. 219).

Um destes segmentos de tecnologia pode ser analisado, o aplicativo de celular, sendo ele acessado pela internet ou não, mas seu uso pode ser auxiliado pelo componente curricular da classe e da turma que está sendo inserida, em que é possível ter um processo avaliativo dentro e fora de sala de aula. Além de ser um dos principais meios de uso atualmente dos jovens constantemente, como a popularização com o acesso aos dispositivos eletrônicos, como smartphones, tablets, notebooks, esse uso faz com que os jovens cada vez mais tenham contato com a tecnologia e tenham mais facilidade com esse meio, sendo considerados como os nativos digitais, que são pessoas que nasceram com as tecnologias digitais ao seu redor e usufruindo da mesma. A proposta de softwares educativos em sala de aula é uma idealização fundamental para que possa auxiliar o ensino e aprendizagem, pois os softwares educativos são programas que podem ajudar nas habilidades e desenvolver o aprendizado dos estudantes em seu processo em sala de aula ou fora.

Em vista disso, este trabalho busca desenvolver o *design* de um *software* educativo para o aprendizado e avaliação processual no ensino de linguagem de programação. Visto o ciclo de um projeto de desenvolvimento de um software, foram executadas avaliações de experiências de usuário para analisar a usabilidade anteriormente à implementação do código.

1.1 Problema e Justificativa

A quantidade de discentes com dificuldades de aprendizado em linguagem de programação é crescente (Silva et al., 2018), (Viana et al., 2019), principalmente de forma autônoma, engajada e protagonista, e, pouca utilização de tecnologias tanto para a aprendizagem quanto para o processo de avaliação. Ademais, em 2020 vivenciamos uma pandemia de COVID-19 devido o coronavírus SARS-CoV-2, em que transmite por vias nasais e pela boca e o mundo inteiro paralisou pelas mortes ocorridas, nesse período começou a quarentena. Nessa situação, as escolas e universidades tiveram que se adaptar para o ensino remoto, com isso, o ensino e aprendizado também teve mudanças de acordo com suas metodologias. Para tanto, novas ferramentas de tecnologia computacional foram inseridas nesse contexto, como o *meet*, *google class*, *moodle*, *pacote office*, dentre outros. o uso de um aplicativo educativo seria ideal para esse contexto como facilitador tanto para o estudante, como para o professor.

Além desses fatores, é possível inserir que, a retenção e evasão de estudantes em componentes curriculares de programação/linguagem de programação pode ser um agravante que influencie o aprendizado do estudante, bem como a avaliação do docente como formativa e contínua. Assim, repensando metodologias que demonstrem melhor desenvolvimento das habilidades de aprendizado e avaliação. Com isso, a Tecnologia Educacional pode contribuir para esse direcionamento para Educação em Computação auxiliando no processo de aprendizagem e avaliação processual dos discentes e docentes. Partindo desse contexto, o uso de um *software* educativo contribui no processo de avaliação e aprendizagem processual ativa do discente?

Sendo assim, o aplicativo educativo oferece diversos recursos para sanar os impasses encontrados no ensino e aprendizado em sala de aula, principalmente relacionados a Tecnologia. Porém a utilização de um aplicativo nesse tipo de contexto acarreta vários desafios, pois é necessário o equipamento, caso não tenha o equipamento, para ser equânime às aulas, será apenas feito em sala de aula as atividades e tarefas do app, entre outras propostas devidas. Além de considerar estes problemas, também devem ser levadas em consideração questões de usabilidade para que esse aplicativo educativo possa proporcionar uma boa experiência aos

usuários durante o processo de ensino e aprendizado, que são a atratividade visual, uma boa utilidade para o usuário, que tenha acessibilidade, que traga credibilidade acima de tudo, a interface mostra ser intuitiva ao ser manuseada, dentre outros pontos que são característicos de uma boa experiência ao usuário nesse processo de prototipagem.

1.2 Objetivos

Tendo em vista o problema analisado, este trabalho tem como objetivo geral:

- Desenvolver *design* de um *software* educativo para o aprendizado e avaliação processual no ensino de linguagem de programação.

E para atingir o objetivo geral, este trabalho tem como objetivos específicos:

- Explorar metodologias ativas e avaliações de aprendizagem, atreladas à *software* educativo na linguagem de programação;
- Desenvolver o *design* do *software* educativo para o ensino e aprendizagem processual a partir de metodologias ativas;
- Avaliar o *design* do *software* educativo processual pelos discentes e docentes.

1.3 Organização do Trabalho

Este trabalho está organizado da seguinte forma: uma fundamentação teórica é apresentada no Capítulo 2, um levantamento de trabalhos relacionados é apresentado no Capítulo 3, a descrição do protótipo do aplicativo educativo é apresentada no Capítulo 4, os resultados e discussões realizados para a validação do protótipo proposto são apresentados no Capítulo 5 e as considerações finais alcançadas são apresentadas no Capítulo 6.

2 Fundamentação Teórica

O desenvolvimento de um aplicativo educativo para o aprendizado de linguagem de programação envolve a aplicação de diversos conceitos da pedagogia, Ensino Básico: antes de iniciar os estudos de programação, como português, matemática, raciocínio lógico; Aprendizagem e Avaliação, temas como autonomia, protagonismo, motivação, jogos, divisão de etapas, entre outros; Acessibilidade, com especificidades como deficiência visual e auditiva. Ademais, também contempla temas da área de computação, como *Design de Software*, contemplando conteúdo de prototipação das telas (mockups), arquitetura da informação, princípios do *design*, usabilidade, entre outros. Além disso, algumas ferramentas são necessárias, como *JustinMind*¹, que foi analisado como o software de prototipação de telas para essa monografia no tópico 4.2.4 de Protótipo dentro da fase exploratória que contempla *Software de Prototipagem*. A seguir, cada um desses conceitos e ferramentas listados serão explicados abaixo.

2.1 Ensino Básico: antes de iniciar os estudos de programação

Em cursos da área de Computação, seja o Bacharelado ou Licenciatura, há um alto índice de reprovação nos componentes de Programação, podendo ser justificado pelos conteúdos envolvendo interpretação de texto, raciocínio lógico e matemática, na qual seria a Educação Básica necessária para o desenvolvimento de programação. (Franzen, Hemming e Bercht, 2018). Essa evasão dos estudantes nos componentes de programação persiste também em outras áreas afins de exatas, como as Engenharias e Matemática. Segundo Krzyzanowski et al. (2019), um dos principais motivos para essa evasão e descontinuação é o despreparo do ensino básico antes de entrar na graduação, como o desenvolvimento do raciocínio lógico, resolução de problemas, pensamento computacional, as habilidades matemáticas e interpretação de texto.

Para evidenciar os motivos que podem causar a dificuldade no aprendizado de programação, Giraffa e Mora (2013), identificam como: a escrita, interpretação de textos, hábito de estudo e habilidades matemáticas. Além disso, aspectos que ligam a fatores motivacionais, emocionais, estruturais e socioeconômicos, são considerados pertinentes para um declínio da continuação no estudo de programação, como

também, questão de gênero, pois predominantemente é composto por sua maioria do gênero masculino.

Uma pesquisa de levantamento bibliográfico e pesquisa de campo feita pelos autores Giraffa e Mora (2013), foi realizada com estudantes de graduação, que abandonaram ou cancelaram a disciplina antes do término do período letivo, do curso de Sistemas de Informação e de Ciência da Computação da PUCRS, no período de 2012 a 2013. Essa pesquisa fez diversos questionários, alguns deles são, em relação à falta de tempo para estudar, em que 74% afirmaram que é um fator influenciador para o desempenho no componente de programação, pois muitos dos estudantes tem uma jornada dupla de atividade, ou seja, trabalham e estudo ao mesmo tempo, assim tendo pouco tempo para estudo extra fora da sala de aula e do trabalho, prejudicando seu desempenho.

Outros fatores relacionados à desistência de programação, é a falta de base das habilidades matemáticas, raciocínio lógico e interpretação de texto. Sabe-se que, a proporção maior dos cursos da área de Computação é dominante pelo gênero masculino. Assim, de acordo com Alves e Ferrão (2019), trazem resultados da educação básica no Brasil baseada nos dados da Prova Brasil, em que no caso das meninas tem melhor desempenho em leitura e interpretação textual, já no caso dos meninos acontece o contrário, nas habilidades matemáticas acabam normalmente se destacando mais, algo padronizado no sistema educacional no Brasil. Com isso, percebe-se a inserção maior deles nos cursos de Computação, e, também de Exatas, dificultando cada vez mais a equidade social no país, bem como o aprimoramento e avanço tecnológico, pois a demanda de mão de obra tem aumentado, porém a qualificação não tem tido tanta visibilidade. Ademais, os processos de desigualdades não ocorrem apenas no que se refere ao gênero, mas também a raça/cor, região/localidade, que são variáveis fundamentais para serem consideradas como precursoras de um mau desempenho em programação.

No trabalho de Alves e Ferrão (2019), conclui-se que o sistema educacional melhorou ao longo do período de 2007 até 2017, mas que foram passos largos para que alcançasse a expectativa inicial do governo, dos componentes de programação das universidades ou do mercado de trabalho exigida. Portanto, medidas de inserção como exemplo *Computational Thinking* (CT), no português, Pensamento

Computacional (PC) no ensino básico pode acarretar melhorias no desempenho dos estudantes que cursarem Computação em sua graduação, como também quaisquer outros cursos, pois o PC auxilia o desenvolvimento de habilidades do aprendizado do estudante como em leitura, na matemática e outras áreas. Desse modo tendo uma qualificação educacional para ser inserido no mundo de trabalho, de forma que traga aplicação do mundo real para os conteúdos de sala de aula.

Por meio do raciocínio matemático ou computacional que se pode avançar nos estudos de programação, pois é um elemento fundamental para sua continuação. Mediante a isso, com as habilidades matemáticas é indiscutível a ideia de resolver problemas que não envolvam restritamente a aritmética, e sim o foco no raciocínio lógico, pois é uma forma de estimular o cérebro para desenvolver suas habilidades e não ficar fechado em uma conta aritmética, mas também para outros caminhos do mundo, de conteúdos e de disciplinas.

Diante disso, de acordo com Vickery (2016), aponta que as habilidades do pensamento da matemática devem ser estimuladas por meio de problemas que relacione com o mundo real, assim dando sentido para o estudante sobre o que está fazendo, também do uso do PC, do objetivo no raciocínio lógico/algoritmo que envolva ações colaborativas, que possa ter debates em grupos ou duplas para articular o pensamento para resolução do problema, pois com isso, aumenta a motivação, a confiança e a autoestima.

O aprendizado das habilidades de matemática pode ser visto como exemplo em outros países mais desenvolvidos, como Inglaterra, Austrália, Canadá, entre outros, que desenvolvem as habilidades matemáticas computacionais dos estudantes desde muito cedo de uma forma autônoma que traga confiança e aprendizado. Como exemplo no artigo de Rodrigues e Cordeiro (2013), na Austrália é promovido de acordo com a idade e capacidade dos estudantes, assim podendo individualizar o aprendizado de forma que ainda continuem em grupos, na qual não demandaria tanto do professor, mas sim de uma gestão escolar. No ensino de habilidades de matemática, eles desenvolvem diversos aspectos, como entendimento, fluência, raciocínio lógico, pensamento, análise, críticas, conceitos, resolução de problemas e habilidades (Rodrigues e Cordeiro, 2013). Com isso, os estudantes desde criança se

tornam mais aptos na vida escolar, profissional e pessoal, portanto, as dificuldades que provavelmente teriam em aprender linguagem de programação seriam baixas.

A retenção de estudantes nos componentes de programação pode afetar também a língua materna (escrita), como interpretação textual, enunciados, além do hábito de estudo e pesquisas. No trabalho de Gomes e Melo (2012), perceberam que as maiores dificuldades encontradas no ensino e aprendizado de linguagem de programação no que se refere à língua escrita, são: interpretação do enunciado e dificuldade no entendimento da língua estrangeira utilizada que foi o inglês. Diante disso, há outra lacuna no ensino de programação para pessoas surdas, pensando acerca de Acessibilidade e Inclusão, pois para ouvintes já existe a dificuldade de interpretação textual e a língua materna, para os surdos, essa dificuldade se torna maior ainda, pelo fato que a língua materna deles é a língua de sinais e não o português, assim se torna mais complexo a situação, quando é utilizada língua estrangeira, como o inglês (Oliveira et al., 2019). Portanto, ferramentas que possam traduzir os códigos para a língua mais adequada é imprescindível para melhor desempenho.

De acordo com Jesus et al. (2018), um dos erros mais comuns no ensino e aprendizado de linguagem de programação está voltado mais para a parte de sintaxe, semântica e lexical, a menor parte de erros é para a parte de lógica. Portanto, percebe-se que os estudantes normalmente erram mais em aspectos de linguagem textual ou interpretação desses conjuntos, assim sendo necessário o reforço no idioma português no que se refere a sua interpretação para o desenvolvimento do ensino e aprendizado de linguagem de programação.

No que se refere ao contexto do português, tanto como seu idioma, mas também a parte gramatical, interpretação textual, entre outras características, há uma possível relação de padrão textual de uma linguagem para outra, como um reconhecimento de padrão de uma das habilidades do Pensamento Computacional, traz a concepção de padrões existentes em algo, nesse caso o português pode ser traduzido para outros idiomas, como inglês, francês, mas também, pode ser traduzido para linguagens computacionais, como *java*, *python*, *c*, *php*. Assim, o padrão pode ser identificado por meio de pontos-chaves, como por exemplo: uma frase em português “meu nome é rayana”, cada palavra será traduzida para o inglês ficando “my name is

rayana”, portanto em alguns casos da linguagem não necessariamente será traduzido palavra por palavra, sendo reconhecimento no contexto geral, o mesmo acontece com a linguagem de programação, por exemplo em um enunciado “escreva um algoritmo que leia três notas, faça a média e exiba o retorno”, identificando palavra por palavra podemos traduzir os termos para a linguagem computacional, como escreva um algoritmo será traduzido para qual linguagem será usada? precisa de um corpo inicial da linguagem? Como exemplo na linguagem Portugol Studio, é usada com o idioma em português e seu corpo inicial é programa { funcao inicio() { } }. Dando sequência ao enunciado, pode-se analisar o leia três notas, para ler uma nota é necessário criar uma variável, nesse caso são três, normalmente nota é considerado como um número com casas decimais, assim sendo usado o tipo da variável como real, posteriormente feito a criação das variáveis, é possível a leitura dele. Na sequência do enunciado é dito para fazer a média, do mesmo jeito é necessária uma variável do tipo real para a atribuição da soma das médias e dividido por três pois é a quantidade final de notas lidas. Por fim, é preciso exibir o retorno da média, utilizando a sintaxe escreva já da própria da linguagem é possível exibir a média e algum texto relacionado ao mesmo. Com isso, percebemos alguns padrões de tradução, em que é possível ajudar nesse processo de hábitos de lógica de programação para iniciantes de um curso de graduação ou até mesmo ensino médio. Também pode ser reconhecido um processo como redação, quando fazemos uma redação de concurso ou vestibulares, lemos um enunciado e traduzimos ele para um texto, que será dividido em introdução, desenvolvimento e conclusão, tendo todo seu passo a passo e estrutura correta de acordo com o enunciado para seguir os parâmetros exigidos.

No contexto de conteúdos e exercícios, a utilização de divisão de etapas nesses pontos é importante para um aprendizado mais eficaz, com foco em um tempo pequeno, mas dedicado especificamente a uma parte de estudo, assim não forçando ou até mesmo estendendo o pensamento do cérebro a ser desgastado. Assim, além do foco, hábito e direcionamentos melhores para habilidades do Pensamento Computacional são abordados para um melhor desenvolvimento no ensino e aprendizado de Linguagem de Programação, como também para o professor avaliar esse processo é importante verificar todos os passos obtidos pelos estudantes em um espaço de tempo que normalmente não há para eles avaliarem de forma tão minuciosa.

Nesse contexto de inserir diversos tipos de linguagens diferentes no aplicativo educativo *learcod*, pode ser considerado como uma proposta de padronizar o software para que possa inserir os exercícios independente da linguagem de programação, assim sendo um aplicativo mais universal, pois nesse app quaisquer linguagens podem ser abordadas, não apenas uma específica, a ideia é que a linguagem seja adaptada ao tipo da tarefa que é do aplicativo e assim poder seguir o fluxo dos tópicos de cada assunto. Dessa forma, possibilita novas versões desse aplicativo e não ficando restrito em apenas um conteúdo ou apenas em uma linguagem, em que ele possa desenvolver e construir um leque de mais versões com mais opções de linguagens tanto para que os estudantes possam desenvolver suas habilidades e conhecimentos, como para os professores possam usufruir no auxílio em seus componentes curriculares.

2.2 Aprendizagem e Avaliação

Nesse seguimento, aborda-se diversos aspectos da aprendizagem e avaliação, que serão tratadas em partes e divididas em mais subtópicos. Quando se fala em avaliação, logo se pensa em provas, exames, aqueles que irão dar uma nota final a todo o seu aprendizado. Esse processo de avaliação, na qual chamamos de tradicional não propõem oportunidades para apresentar *feedbacks*, retornar ao conteúdo aprendido, inserir o estudante como protagonista e autônomo do seu aprendizado, dentre outros pontos. Por isso, a avaliação formativa é uma ideia muito nova e muito difícil de ser inserida, mas com intuito de quebrar com essa abordagem da avaliação tradicional, em que o estudante é o centro da avaliação, exemplo da autoavaliação, a via de mão dupla é fundamental nesse desenvolvimento (Perrenoud, 1999).

Nesse sentido, Perrenoud (1999), continua que a avaliação formativa pretende melhorar a aprendizagem em curso, sendo uma diferenciação do ensino.

A avaliação formativa é um componente quase obrigatório de toda avaliação contínua [...] nem toda avaliação contínua pretende ser formativa. Em uma classe [...] não é somente ensinar, mas também manter a ordem, animar trocas, por para trabalhar, garantia uma coexistência pacífica. (Perrenoud, 1999, p. 79)

Com isso, pode-se entender que não é só trabalho intelectual, mas também a comunicação e social, diversos outros aspectos que são trabalhados, como também

o processo é algo centrado no estudante e contínuo para que desenvolva os conhecimentos e as habilidades do estudante.

Em conjunto com a avaliação tem a aprendizagem, que nos faz repensar os modos de intervir em sala de aula, novas metodologias, técnicas que permeiam esse processo, desenvolvimento do conhecimento do estudante. Segundo Mendes (2019), o uso das tecnologias, recursos da internet, *games*, que façam essa interação com o estudante é importante para o desenvolvimento das aulas e o aprendizado, pois possibilita outros leques de opções, impacta a vida dos alunos, traz o ensino acessível, propicia um ensino e aprendizado com qualidade.

São diversas formas, estratégias, técnicas, metodologias que podem ser abordadas, repensadas para serem usadas em sala de aula, para que seja mais engajada, motivante, para que o estudante seja o atuante do seu aprendizado, dentre elas podemos citar as metodologias ativas, que se categoriza em subtópicos como a PBL (*Problem Based Learning*, ou em português, Aprendizagem Baseada em Problemas - ABP), Sala de Aula Invertida, *Coding Dojo*, *Gamificação*, *Peer Instruction* (Aprendizado por Pares), dentre outras (Ribeiro, 2008), (Villa e Delgado, 2019), (Bergmann e Sams, 2016).

O contexto de questões de programação em menores partes ou subdivididas pode ser correlacionada com Pensamento Computacional (PC), que são habilidades e competências que podem ser desenvolvidas para desenvolver melhor nas linguagens de programação. Segundo Wing (2006), afirma que:

o pensamento computacional é uma habilidade fundamental para todos, não apenas para os cientistas da computação. Para leitura, escrita e aritmética, devemos adicionar a habilidade destacada à capacidade analítica de cada criança. (Wing, 2006, p. 33)

No PC traz a possibilidade de se resolver problemas, existem algumas competências que se classificam como abstração, reconhecimento de padrão, algoritmo e decomposição e composição. Dessa forma, os estudantes conseguem ter uma possível aptidão melhor por querer praticar os exercícios pela forma decomposta em que se está em quando estiver em sala de aula já estar compreendo o conteúdo.

Essa abordagem relaciona com o tema de como nosso cérebro trabalha de dois jeitos diferentes a depender do caso. Segundo Oakley (2019), relata que:

nosso cérebro trabalha de dois jeitos, um no modo focado e outro no modo difuso. [...] No modo focado significa estar prestando atenção. [...] No modo difuso sua mente está relaxada e desocupada, sem pensar em nada em particular. (Oakley, 2019, p. 25)

Com a pandemia pelo covid-19 causou muitos impactos a sociedade, um deles foi no ensino, que passou do presencial para o ensino remoto, assim formas de repensar o ensino foram necessárias para que as aulas tornassem mais atrativas. De acordo com Ferreira, Branchi e Sugahara (2020), relatam que o uso das tecnologias digitais são fundamentais nas aulas para os professores na transformação que ocorre na aprendizagem do estudante. Desse modo, percebe-se a importância do uso da tecnologia como auxiliador em sala de aula em todos os âmbitos.

Para adentrar no âmbito das correções das questões em códigos de programação, se toma como base o Sistema Tutor Inteligente pelos autores Britto, Filho e Barros (2017, p. 1), que “são softwares capazes de complementar o processo de ensino-aprendizagem dos alunos, de forma personalizada, em diversas áreas do conhecimento”. Esses programas indagam um ambiente mais interativo e dinâmico no ensino e aprendizagem e que possa ser adaptado para o estilo de cada estudante. A ideia que é dada pelos autores Britto, Filho e Barros (2017) é que quando a questão de eletrônica está correta irá passar para o próximo, porém quando erra não irá mostrar o resultado da questão certa e sim o Sistema Tutor Inteligente compreende que é necessário aprender mais sobre esse conteúdo específico, então ele retorna ao tópico para que possa estudar e então possa seguir a diante e refazer a questão para ter uma outra chance em tentar resolvê-la, e sucessivamente.

Seguindo essa abordagem da autonomia, podemos perceber que a autonomia pode ser considerada um ponto no ensino e aprendizado, pois é um dos processos não só de ensino, mas também de avaliação formativa e contínua. Segundo Kamii (1989), relata que a autonomia significa o ato de governar a si mesmo, isso faz com que desenvolva habilidades de pensar a criticidade, criatividade, lógica, entre outros aspectos que são importantes para o estudante. Essas habilidades são fundamentais de serem desenvolvidas em salas de aulas ou fora dela para que desenvolva o estudante em seu processo de ensino e aprendizado.

O desenvolvimento da autonomia ocorre desde criança, isso podemos destacar segundo Kamii (1989), que

A criança torna-se progressivamente mais autônoma à medida em que cresce, ou seja, torna-se apta a governa-se, ela é menos governada por outras pessoas. Na realidade, a maioria dos adultos não se desenvolve desta forma ideal. A grande maioria interrompe seu desenvolvimento num nível baixo. (Kamii, 1989, p. 106)

Uma abordagem que pode ser analisada são as estratégias motivacionais, que nada mais é do que a forma como se motiva o estudante em sala de aula, Fernández e Callegari (2010), comentam

Motivar os alunos em sala de aula significa valer-se apenas de recursos pontuais, como oferecer esporadicamente músicas e jogos [...] Contudo, o processo de motivação vai muito além disso, trata-se de uma abordagem de ensino que se pauta na criação e manutenção de um ambiente totalmente propício à aprendizagem, objetivando manter o interesse dos alunos. (Fernández e Callegari, 2010, p. 73)

Segundo Fernández e Callegari (2010), algumas estratégias para a motivação do estudante podem ser levadas em consideração, como: desenvolver a autoconfiança; promover o sentimento da autoeficácia; ajudar a reconhecer a relação entre esforço e resultados; encorajar a estabelecer objetivos específicos durante o componente; apresentar o conteúdo criativo, deixar mais relevante/curioso; combinar as habilidades dos estudantes com o nível de dificuldades das tarefas; incentivar a formação de grupos com objetivos específicos; utilizar *feedbacks* motivacionais; dentre outros.

2.3 Design de Software

Para tratar do tema de prototipação, antes é necessário compreender o que é Design Thinking (DT), que significa um modo de pensar e abordar problemas que colocar as pessoas no centro da solução desse problema. Muito se é tratado esse tema na área de design de projetos, estratégias que possam amenizar erros de implantação de códigos posteriormente. O DT também pode correlacionar com a educação, sugerindo modelos de abordagens de pensar em habilidades de autoconhecimento, empatia, resolução de problemas, comunicação, inteligência emocional, tomada de decisão, pensamento criativo e crítico, relacionamentos interpessoais, dentre outros (Gonsales, 2018).

A Engenharia de Software pode ter várias ramificações, como por exemplo a DT, assim a Engenharia de Software pode ser explicada como um processo de desenvolver um software de quaisquer tipos, seguindo alguns passos, nesse primeiro

passo foi elaborado a pesquisa direcionada (ou de referências) que é o estudo de diversas fontes para embasar todo o trabalho, posteriormente existe a prototipação é a parte que terá as telas do software, mostrando todo o fluxo de interação e apresentando todas as ideias possíveis, sendo o mais objetivo possível. É nesse momento da prototipação que se utilizará a tipografia, paleta de cores, ícones adequados, acessibilidade, componentes, para que possa interagir adequadamente ao usuário. Essa prototipagem não tem apenas um significado visual, tem também aspecto das funcionalidades serem úteis para os usuários finais (Rocha et al. 2021).

A avaliação dessa prototipagem é fundamental para ter como parâmetro além da usabilidade, refazer alguns ajustes do protótipo caso algum ícone, cor, tipografia ou qualquer outra funcionalidade precise colocar ou retirar. Essas alterações ajudam no processo de desenvolvimento do software para que posteriormente na implementação do código não ocorra tantos erros. Há diversos tipos de questionários de usabilidades para essas avaliações, como: QUIS (Questionnaire for User Interaction Satisfaction, em português, Questionário de Satisfação de Interação do Usuário), SUMI (Software Usability Measurement Inventory, em português, Inventário de Medição de Usabilidade de Software), WAMMI (Website Analysis and MeasurMent Inventory, em português, Análise de Website e Inventário de Mensuração), SUS (System Usability Scale, em português, Escala de Usabilidade do Sistema), o mais usual é o SUS, pois tem maior escalabilidade de questões que envolvem a usabilidade voltada para aspectos tecnológicos (Filardi e Traina, 2008).

Para adentrar no âmbito mais técnico, esse aplicativo se caracteriza como um software educativo pelo intuito dele que já teve como objetivo e função relacionada a formação e direta para educação, tratando de seu ensino e aprendizado, como também avaliação. Segundo Tavares e Silva (2017), comentam os tipos de Softwares que se dividem em dois: educativo e educacional, o primeiro tem como objetivo ser direcionado pedagogicamente para a educação sendo usado por docentes e discentes, já o segundo não tem esse intuito, porém ele pode ser abordado em ambas as partes de quaisquer usuários. Portanto, o aplicativo se caracteriza como um software educativo por ter um direcionamento para educação, em que pode ser usado por professores, estudantes e futuramente por quaisquer usuários, assim sendo uma ferramenta mais pedagógica em seu ponto de vista.

Segundo Gomes e Melo (2012), “O conteúdo deve ser potencialmente significativo. [...] O conteúdo deve possuir um significado lógico e psicológico, ou seja, além de fazer sentido, precisa ter importância para o estudante”. Por fim, considera-se que, não é apenas colocar conteúdos de programação em um aplicativo e colocar uma ideia de Tecnologia em sala de aula ou fora dela em um componente curricular, precisa de um contexto metodológico e pedagógico que faça sentido de estudo e aprendizagem para o estudante, caso contrário não será atrativo para que ele participe ativamente dessas tarefas frequentemente.

2.3.1 Software de Prototipagem

Os *Softwares* de Prototipagem são um meio facilitador para o entendimento dos requisitos iniciais do desenvolvimento de um software, que se pode visualizar conceitos e funcionalidades de modo simples para compreensão. Utilizando a prototipagem aumenta a percepção de valor da solução colocada para o problema do cliente, com isso, o usuário pode avaliar os recursos e a organização do layout que estão na prototipagem de forma que irá impactar a experiência dos usuários. Logo abaixo será analisado algumas vantagens e desvantagens de utilizar alguma dessa ferramenta para prototipagem:

- **Vantagens**

- Velocidade – possível fazer cópias, interações com apenas alguns cliques.
- Bibliotecas de Elementos – elementos já prontos de interações simples, bibliotecas de elementos embutidos, pode criar seu próprio elemento para uso repetido.
- Fluxograma e Fluxo de Usuários Avançados – permite gerar *sitemaps* à medida que você cria novas telas para criação de protótipos, lado a lado para que você possa navegar neles.
- Colaboração Integrada – compartilhamento de edição e por meio de links, tanto para a equipe de desenvolvimento, como para os clientes.
- Apresentação Simplificada – exportar para PDF ou para um aplicativo da web/mobile.

- **Desvantagens**

- Falta de familiaridade – caso não seja um designer ou afins, o uso das ferramentas pode-se tornar complexo, mas nada que seja impossível de ser aprendido.
- Fidelidade e funcionalidade limitadas – boa parte das ferramentas de prototipagem é de baixa fidelidade ou de baixa funcionalidade.
- Custos-Benefícios – fator predominante em caso de pesquisas acadêmicas, até mesmo empresas que esperam não gastar muito com determinadas ferramentas.

- **Exemplos**

- *JustinMind*: há painéis dinâmicos e eventos, podendo simular botões, mudanças de tabulação, os eventos permitem indicar ações, sendo de interação. Gratuito, mas com uma versão paga.
- *Balsamiq*: similar com a experiência de um quadro branco, com intuito ser rápido da sua criação de protótipos e *wireframes*, esse tipo de processo normalmente permite que você tenha mais ideias em um curto período. Gratuito, mas com uma versão paga.
- *Pencil*: possível criar interfaces com base em sistemas operacionais, como o Android e iOS, como também permite exportar o projeto em diversos formatos para pré-visualização de sua navegação. Apenas gratuito.
- *Invision*: possível compartilhar e comentar um projeto em tempo real com o cliente, assim o *feedback* chega mais rápido e mais claro, e os processos são simplificados. Gratuito, mas com uma versão paga.
- *Axure*: bem complexa, mas oferece o máximo de controle aos projetistas, normalmente não muito usual para iniciantes. Gratuito, mas com uma versão paga.
- *UXPin*: colaboração e comunicação com a equipe durante o processo em tempo real, integrar versões de documentos dos arquivos, demonstrar o processo aos clientes e coletar sugestões e comentários. Gratuito, mas com uma versão paga.
- *Figma*: possível coletar comentários ou anotações sobre os projetos, é uma ferramenta de interface baseada em navegador, o que facilita

a criação de *software* pelas equipes. Gratuito, mas com uma versão paga.

- *Flinto*: bastantes funcionalidades relacionadas às transições, permite manipular ou ajustar cada elemento de transição, por isso é ideal para projetos que necessitem passagem de usuários entre várias páginas. Gratuito, mas com uma versão paga.
- *AdobeXD*: criado para competir com o mercado do Sketch e Figma, com compatibilidade com aplicativos do Adobe, como Photoshop, o que facilita essa integração. Gratuito, mas com uma versão paga.
- *WebFlow*: inclui recursos de hospedagem com várias opções de design, além de ser capaz de editar em tempo real através de uma interface semelhante ao Photoshop. Gratuito, mas com uma versão paga.
- *Fluid*: possível criar o layout das principais visualizações do seu aplicativo, mas também pode vincular cada uma dessas visualizações a outros controles, podendo assim realizar uma demonstração interativa muito próxima do que seria o resultado. Gratuito, mas com uma versão paga.
- *Origami*: apenas idealizado para criação de protótipos do tipo iOS e apenas para o sistema operacional Mac OS. Apenas gratuito.
- *Proto.io*: possui um mecanismo que permite criar animações para aplicativos móveis por meio de uma linha do tempo para definir quanto tempo a animação durará. Gratuito, mas com uma versão paga.
- *FramerJS*: muitos designers de UX usa para mostrar as interações e as animações específicas que podem criar, nessa ferramenta, quando carrega um gráfico, o código é gerado no lado esquerdo da tela, e ele pode ser reutilizado ou ajustado para ilustrar variações. Gratuito, mas com uma versão paga.

2.3.2 Acessibilidade

Na educação há diversas metodologias que são importantes para serem repensadas no campo da acessibilidade, como inserir mais o estudante em sua turma no contexto da aula para que o seu desenvolvimento e habilidades sejam equânimes. Como também, o uso da tecnologia pode auxiliar nesse ambiente, pois trazendo um

aplicativo com interpretação em libras ou um software de desktop que irá auxiliar na no áudio descritivo para deficientes visuais, todas essas ferramentas poderão auxiliar o desempenho do estudante em sala de aula em seus estudos.

Segundo o trabalho de Cardoso et al. (2021), que propõe um aplicativo para deficientes visuais usarem em mercado para saberem os preços, informações dos produtos, o que para eles é muito inacessível e isso tornou mais fácil com a tecnologia. Por meio da tecnologia além de facilitar que é o celular que sempre está em sua mão que pode utilizar o tempo todo contigo em qualquer situação, pessoas assim podem criar aplicativos mais assistivos dessa forma.

Além da acessibilidade auditiva, foi levada em consideração aos surdos para uma inclusão maior também, pois as vezes a interpretação de termos da língua e de termos técnicos são incompreendidos, assim necessitando de determinados glossários com vídeo para mostrar animação daquela frase ou palavra que deseja saber ou reajuste do texto para interpretação. Tendo em frente esse problema, no trabalho de Borszcz et al. (2021), realizaram um projeto Libras Play com intuito de tecnologia assistiva para surdos para oferecer inclusão e facilitar o ensino e a educação entre professores e estudantes, bem como a comunicação social, essa é uma das formas que traz além do desenvolver o ensino e aprendizado, a comunicação social entre os colegas e professores, pois interage melhor entre si. Além dessas, pode-se destacar diversos outros pontos que podem ser considerados pontos de acessibilidade ou âmbito de inclusão, como deficiência intelectual, transtorno de déficit de atenção, autismo transtorno do espectro do autismo, dentre outros que podem ser repensados para incluir um aprendizado mais eficaz e que desenvolva as habilidades desse indivíduo.

2.4 Trabalhos Relacionados

Para considerar uma avaliação mediada por um programa, alguns parâmetros são mostrados por Mittal e Mandalika (2015), como correção, estilo, eficiência e complexidade. Ademais, o *feedback* é algo importante a ser considerado, por exemplo o SQLify é um programa para facilitar o aprendizado da linguagem SQL, tendo como principal foco o *feedback* dos resultados. Por fim, para a avaliação desse programa também é abordado os testes, como o de caixa preta e branca, assim precisando de um contexto mais holístico para ele (Mittal e Mandalika, 2015).

No trabalho de Kogure et. al (2019), desenvolveram o programa TEDVit, um sistema que visualiza o comportamento da linguagem de programação. Com isso, eles propuseram também um Modelo Conceitual Orientado a Objetos (Modelo OOC), que apresenta uma interação do professor com a linguagem de programação, que nesse caso seria com os estudantes. Portanto, fizeram uma avaliação experimental que investigaram hipóteses como "o TEDViT aumenta o nível de compreensão dos conceitos de programação?", "o modelo OOC ajuda os estudantes no ensino e aprendizado de programação?", entre outras indagações. Os resultados foram positivos para a compreensão da aprendizagem de programação, porém algumas melhorias podem ser feitas, como a interface do sistema com usuário e simplificar algumas regras do professor-estudante.

A ferramenta PROALG que Franzen, Hemming e Bercht (2018) desenvolveram em seu artigo teve como apoio no ensino de programação de forma virtual, essa proposta vai com base em divisão de etapas de um mesmo exercício, como ideia de fragmentar para poder compreender melhor o problema e poder resolvê-lo, assim tendo mais motivação e concentração no exercício.

No trabalho de Gomes, D'emery e Filho (2020), AAPW é um ambiente de aprendizagem de web para facilitar o contexto pedagógico da linguagem, bem como a relação com a IDE utilizada e o idioma trabalhado. Esse ambiente visualiza tanto na parte de cliente, como de servidor, além de alguns códigos em português para auxiliar no processo de desenvolvimento e aprendizado. Com isso, é possível analisar que houve um bom desempenho dos estudantes nas resoluções do problema nesse AAPW, como também aceitação.

No artigo de Eles utilizaram a estratégia de prototipação, criando as telas de como ficariam o aplicativo educativo e questionaram os usuários sobre seus rearranjos e ajustes finais para alterações, assim tendo menos erros no futuro, no caso da implantação dos códigos. Pois o que normalmente os analistas de sistema realizam é um processo de questionários e documentos de requisitos, conversa com cliente para prosseguir e desenvolver o aplicativo, mas essa estratégia de já interagir com os usuários finais com a prototipação das telas dá grande potencial de pequenos erros futuramente e garantia de usabilidade para o usuário, como também a participação ativa dos usuários no desenvolvimento do aplicativo educativo. Os resultados obtidos

foram que essa proposta tem potencial para um recurso didático como análise antecipada, pois após os ajustes da prototipação teve resultado positivo dos participantes.

Segundo Britto, Filho e Barros (2017) em seu trabalho apresentam, um modelo de Sistema Tutor Inteligente (STI) aplicado ao Ensino de Sistema Básico Eletrônico, os STI são programas que tem como intuito usar técnicas de inteligências artificial para auxiliar estudantes no seu processo de ensino e aprendizado, eles tem diversas técnicas diferentes e métodos que podem ser abordados para diferentes ambientes de interação em que será atuado, proporcionado um ambiente dinâmico e interativo, sendo um conteúdo adaptativo ao estilo de cada estudante.

A ideia central desse sistema dos autores é propor um modelo que irá desenvolver o aprendizado de eletrônica básica para os estudantes, ao estudar um determinado conteúdo, irá repassar para algumas questões, caso essas estiverem certas, seguirá o fluxo para os próximos passos e níveis, caso alguma dessas questões o estudante erre, o STI irá perceber que o estudante precisa estudar mais o conteúdo específico dessa questão na qual errou, pois ficou alguma lacuna, assim voltando para a tela de explicação e não dizendo qual foi a questão correta, quando o estudante terminar seu estudo, irá avançar, ao seguir será voltado para a mesma questão na qual ele errou tendo a oportunidade de refazer novamente ou refazê-la em outro momento caso quiser. Dessa forma, é possível verificar que, o modelo dá a possibilidade de rever o conteúdo a ser estudado antes mesmo de mostrar qual a questão correta e, também dando outra chance de refazer a questão, isso possivelmente desenvolve o ensino e aprendizado do estudante ao longo do processo de educação.

2.4.1 Considerações acerca dos Trabalhos Relacionados

Percebe-se uma lacuna em ferramentas/metodologias que auxiliem no processo contínuo de avaliação para os professores em conjunto com o aprendizado do estudante na mesma tecnologia. Com isso, percebe-se a necessidade de uma aplicação que contemple todas essas lacunas e outros aspectos necessários para o aprendizado e avaliação processual para estudantes e professores. Além disso, há poucos estudos que realizam o uso de prototipação com participação ativa para análise da experiência do usuário anteriormente à implementação do software.

Portanto, considera-se que, com os trabalhos relacionados citados acima, o uso de prototipagem antes da implementação é fundamental para evitar possíveis erros futuros, o que seria prejudicial tanto em tempo, custo, entre outros aspectos.

3 Percurso Metodológico

Este percurso metodológico está organizado da seguinte forma: a caracterização da pesquisa no 3.1, a fase exploratória no 3.2, a estruturação e desenvolvimento no 3.3, como subtópico tem a observação documentação no 3.3.1, o questionário de viabilidade no 3.3.2, a avaliação de viabilidade com os docentes no 3.3.3 e questionário de viabilidade com os discentes no 3.3.4, o fluxo de atividade do aplicativo educativo no 3.3.5, as considerações da escolha do software educativo no 3.3.6, a prototipação no 3.3.7, como subtópico da prototipação tem-se a prototipação do perfil de usuário discentes no 3.3.7.1, prototipação do perfil de usuário docentes no 3.3.7.2, e por fim, a avaliação de usabilidade no 3.4.

3.1 Caracterização da Pesquisa

Esta pesquisa utiliza elementos da pesquisa-ação (Feitor, Filho e Souza, 2013), de natureza quantitativa - traduz em números opiniões dos entrevistados, que compõe docentes e discentes (Nascimento e Cavalcante, 2018). Na pesquisa-ação, também pode ser chamada de prática-exploratória, tem como objetivo primordial intervir na realidade social, mas alguns dos aspectos da pesquisa-ação nesse trabalho foram introduzidas, como planejar/identificar, levantar e refletir sobre os dados, investigar soluções, agir e observar, no que se refere a implementar a melhor solução e de intervir na realidade social ficou como trabalhos futuros, além da reflexão das coletas dos dados para transpor da prototipação para implementação e aplicação em sala de aula para que possa ser realizada o uso das atividades do aplicativo para novos - ajustes e atualizações . Desta forma, com base em seus objetivos, a pesquisa é de caráter prático, se aproximando da pesquisa aplicada, ao utilizar das etapas da pesquisa-ação para sistematizar os procedimentos de desenvolvimento experimental a partir de saberes preexistentes para a comprovação ou demonstração da viabilidade técnica ou funcional de novos produtos, processos, sistemas e serviços ou, ainda, aperfeiçoamento do que já está produzido ou estabelecido (Tumelero, 2019).

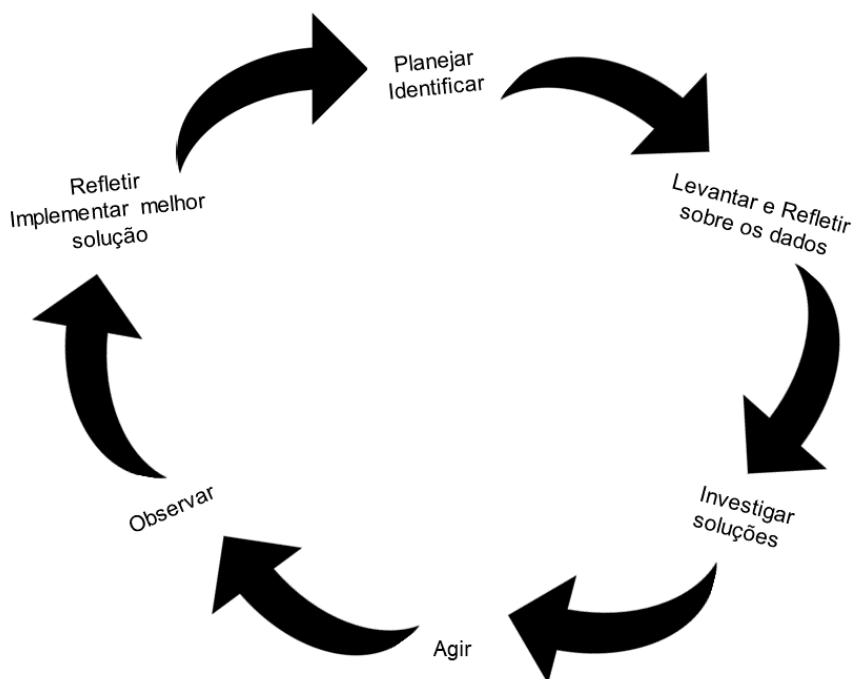


Figura 1. Ciclo Adaptado da Pesquisa-Ação (Feitor, Filho e Souza, 2013).

No processo da pesquisa-ação tem a possibilidade de pesquisa em formato circular como mostrado na figura 1, esse processo é investigativo e de mudanças que provoca reflexões, com abordagens educacionais, organizacionais, empresariais, dentre outros (Feitor, Filho e Souza, 2013).

Nessa fase inicial de planejar é a parte que irá identificar o problema, nesse pesquisa temos o problema como: o uso de um software educativo contribui no processo de avaliação e aprendizagem processual ativa do discente? Na introdução desse trabalho foi discutido o contexto desse problema, assim seguindo para o próximo ciclo que é o levantamento e reflexão dos dados que será o referencial teórico que também foi abordado no capítulo 2, para que se possa embasar todos os pontos e contrapontos desse trabalho. A seguir houve a investigação das soluções, a partir de reflexões de todos os dados de pesquisa de estudo já obtidos, pela pesquisa bibliográfica e questionário de viabilidade, e ideias com algumas soluções que pudessem sanar alguns impasses no aplicativo educativo. Por conseguinte, a ação tomada foi a produção de uma prototipação de telas de um aplicativo educativo, consequentemente foi feita uma oficina para obter resultados de coleta de dados de usabilidade tanto de aspectos tecnológicos quanto da parte pedagógica dessa prototipação. A observação é uma parte do ciclo da pesquisa-ação que segue a coleta de dados também como ponto de reflexão a ser tomada de ação, para posteriormente

seguir no próximo ponto que é refletir e implementar a melhor solução, assim repensando e refletindo os resultados de acordo com a ação do produto que poderá ter alguma modificação ou ajuste de melhora, discussões que podem ser analisadas para que sejam refeitas para que estejam mais adequadas ao nível do cliente ou usuário final. Assim, esse ciclo pode auxiliar em um processo de usabilidade de software e implantação de código sem tantos erros futuros.

3.2 Fase Exploratória

A fase exploratória serviu para proporcionar uma nova visão a respeito do fenômeno estudado, como a pesquisa bibliográfica ou levantamento bibliográfico de trabalhos científicos já publicados, usado para entender o objeto da pesquisa. Essa fase segue todo o fluxo do trabalho a partir da Estruturação e Desenvolvimento, que teve como outros pontos o processo de observação documental, que foi feito uma busca de leituras para dar início ao *design* do *software* educativo, buscando em livros e artigos científicos sendo selecionados de acordo com a categoria de conteúdos previamente elaborados no cronograma da pesquisa, como metodologias ativas, acessibilidade, *design thinking*, entre outros, como forma de estruturar toda ideiação do aplicativo *learcod*.

Para que possa seguir com todo o processo do trabalho, foi feito um questionário de viabilidade para análise e discussões, que foi discutida em reuniões de orientação do trabalho para que essa viabilidade pudesse dar seguimento as próximas atividades, foi produzido nesse formulário diversas sessões para que pudesse traçar dados nas análises, como por exemplo a identificação, como faixa etária e questões de viabilidade. A partir da análise considerada foi possível produzir o fluxo de atividade do aplicativo educativo, que foi feito uma ideia anterior da prototipação de como ficaria os exercícios e o ciclo de funcionamento do aprendizado de linguagem de programação. Posteriormente foram feitas as considerações acerca da escolha do software de prototipagem, com base nos critérios de colaboração integrada, biblioteca de elementos, entre outros, a seguir foi iniciado a prototipação que foi dividida em prototipação do perfil de usuário discentes e docentes, com intuito de ter os dois campos de espaços no aplicativo para o docente e discente de cadastro e telas específicas para ambos.

Por fim foi realizada o questionário de avaliação de usabilidade para ter a experiência do usuário com a prototipação, com isso obteve a análise e discussão dos resultados para conclusão do trabalho, que pode perceber que é preciso de mais ajustes para próximos etapas de trabalhos para que o aplicativo seja implementado, pois o interesse de acordo com os dados foi presente.

Nesse processo da fase exploratória foi também possível perceber como o desenvolvimento do *design* de um aplicativo educativo pode ser ajustado de acordo com os dados coletados do público-alvo que irá usá-lo, assim podendo seguir um fluxo de prévia de erros antes mesmo da implementação de códigos, bem como também pode ser usado no seu processo de implementação e posteriormente com o público-alvo.

3.3 Estruturação e Desenvolvimento

Nesse processo de desenvolvimento do aplicativo educativo, a estruturação se embasa da seguinte forma, a primeira etapa começa com a observação documental que é parte da fundamentação teórica, onde que se busca todo o material de artigos, livros para fundamentar e estruturar todo o conhecimento teórico desse trabalho; posteriormente na segunda etapa foi feito um questionário de viabilidade do aplicativo *learcod*, para verificar se é viável a criação desse *software*, bem como algumas questões relacionadas ao ensino e aprendizagem; a seguir na terceira etapa foi produzido um fluxo de atividades chamado de sequência didática que é uma idealização de como vai ficar a prototipação em forma de fluxo de atividade, como um tipo de *storyboard* (Lucema et. al, 2014), na quarta etapa foi decidido qual software a ser usado para a prototipagem, assim foi descrito as diversas características de vários programas, bem como as vantagens e desvantagens, então foi feito as considerações para que fizesse as comparações de acordo com as questões que atendessem aos requisitos e ao escolhido; por fim tem-se a prototipagem que mostra todo o processo das telas, as interações e explicações de algumas delas de acordo com o processo pedagógico e design.

3.3.1 Observação Documental

A Observação Documental tem como parte o planejamento e identificação como características da pesquisa-ação, assim levando para uma base de leitura que tem uma estrutura para dar o desenvolvimento a ideia do aplicativo educativo. Essa

observação documental é o processo de estudo que se baseou na pesquisa da fundamentação teórica, que se estruturou em partes como da Educação, aprendizagem e avaliação, da área técnica que seria o *design thinking* que envolve a prototipagem de telas e usabilidade, como também envolvimento de trabalhos relacionados para que possa embasar com ideias e soluções para o aplicativo educativo e estudos de acessibilidade. Com isso, é possível observar por meio das referências bibliográficas estudos para embasar toda a estrutura do trabalho e desenvolver o software educativo com mais propriedade e embasamento.

Uma observação que foi analisada foi a estrutura da linguagem de programação usada para fazer parte inicialmente da prototipagem das telas, sendo escolhida a linguagem de programação Portugol, pois se assemelha com a língua portuguesa, como forma de iniciação, além de ser uma das linguagens mais iniciais no ensino de Algoritmo. Posteriormente, podem ser analisadas linguagens como C, Java, Python, entre outros no que se refere a transição para o Inglês, ainda sendo de forma estruturada essa segunda fase. Adiante, é considerado Linguagens mais robustas para Programação Orientada a Objetos, como Java, C++, Python, Php. Para envolvimento de um projeto como todo, pode ser analisado Linguagem de Banco de Dados, como SQL. Ademais, o uso de um servidor local e/ou online para estruturar um desenvolvimento de um sistema web. Para trabalhos futuros, podem ser colocados como estrutura desse aplicativo o desenvolvimento de Linguagens de Programação para Mobile, como exemplo o uso do Flutter que é um framework feito pela *Google* com a linguagem Dart que pode ser feito aplicações tanto Android como iOS. Em um escopo mais específico, no uso de uma linguagem, como exemplo o Portugol Studio, nessa aplicação pode ser usada diversos conteúdos dele, como estrutura de condição, repetição, vetores, entre outros.

Dessa forma, a observação documental viabilizou a primeira parte para iniciar o processo de desenvolvimento do software do aplicativo educativo, como forma de embasamento teórico, como formas de estruturar todas as pesquisas relacionadas as partes de ensino, linguagem de programação e design para que possa desenvolver um projeto com bases e estruturas que tenham fundamentos.

3.3.2 Questionário de Viabilidade

Foi levantado um questionário de viabilidade de característica da pesquisa-ação de levantamento e reflexão dos dados iniciais para esta pesquisa porque esse procedimento caracteriza-se pelas análises de ferramentas e metodologias já existentes para melhorias ou correlações que se deseja implementar nesse trabalho. Desta forma, a pesquisa fez uso de um questionário quantitativo, que apresenta dados em números, para a coleta de dados junto aos estudantes e professores, tanto do campus Porto Seguro, como de pessoas externas, em sua maioria voltadas para área de Computação. Com esse procedimento, mediante a análise quantitativa, obtive as conclusões correspondentes aos dados coletados de uma proposta de melhoria para implementar um aplicativo educativo para aprendizado e avaliação processual no ensino de linguagem de programação.

3.3.3 Avaliação de Viabilidade com Docentes

Esse levantamento sobre viabilidade tecnológica que tem como parte observar e refletir os dados, tem como objetivo avaliar a viabilidade do uso do aplicativo educativo no ambiente escolar como recurso auxiliar em componentes curriculares em quaisquer linguagens de programação tanto para estudantes no seu desenvolvimento de ensino e aprendizado, como para os professores no processo de avaliação formativa e contínua. Com isso, o formulário foi destinado para professores e estudantes, tanto do *campus* Porto Seguro, como externo, sendo também de qualquer área, mas em sua maioria de Computação e Informática, em que no total de discentes e docentes tiveram 36 (trinta e seis) respostas para essa análise, para a análise específica dos docentes obtive 4 (quatro) resultados, que mesmo sendo insuficiente será analisado com intuito para novas coletas de dados futuramente com professores.

O formulário foi dividido em algumas seções, na identificação do docente em que há campos opcionais e outros obrigatórios para serem preenchidos como visto no apêndice B, em que nesse caso obtivemos 3 (três) resultados dos docentes e os demais foram 4 (quatro) resultados, as outras seções são gerais tanto para docente como para discente que é a identificação, habilidades e o próprio questionário de viabilidade.

Quadro 1. Análise do Levantamento de Viabilidade dos Docentes.

Categorias	Análises dos Dados
Faixa Etária	Há 2 (dois) docentes que estão entre 30 e 40 anos, já um docente está entre 40 e 60 anos.
Área de Estudo	Os 3 (três) docentes são da área de Computação.
Plataformas de Ensino	Os 4 (quatro) docentes usam as plataformas de ensino até 5 vezes na semana.
Proveito das NTDIC's	Os 3 (três) docentes afirmaram que há um proveito das NTDIC's nas avaliações.
Viabilidade das NTDIC's em programação	Os 4 (quatro) docentes consideraram que é viável o uso das NTDIC's para as avaliações.
Motivação no uso de plataformas para programação	Os 3 (três) docentes concordaram que se sentiriam motivados em usar plataformas para avaliação de atividades de programação.
Softwares similares	“BOCA Online Contest Administrator (https://www.ime.usp.br/~cassio/boca/) - Juiz para competições de programação online com correção automática, mas também já utilizei para avaliações em disciplina que envolvem” – uma sugestão que pode trazer como complemento de idealização para o aplicativo educativo <i>learcod</i> .
Manuseio de uma nova ferramenta	Os 4 (quatro) docentes concordaram que conseguem aprender a manusear algum tipo de software/aplicativo ou plataforma.

Os dados apresentados no quadro 1 de análise do levantamento de viabilidade dos docentes, demonstram que independente da faixa etária dos professores, para essa pesquisa eles iriam conseguir manusear bem o uso do aplicativo como auxiliar em seu componente curricular, pela concordância em conseguir aprender a usar um software, isso é uma experiência que ajuda no processo do envolvimento do professor com os estudantes, pois como todos os professores que responderam ao questionário são da área de Computação, isso também provavelmente facilita esse meio de comunicação e experiência.

São consideradas pelos docentes como resultado positivo que as NTDIC's são viáveis tanto para o uso auxiliar de avaliação como para o aprendizado de programação em componentes curriculares, como também os docentes afirmaram que seria proveitoso o uso das NTDIC's em avaliações. Em conjunto com isso vem a motivação do uso de algum aplicativo para auxiliar as atividades em sala de aula, em decorrência disso, pode ser que se tenha essa motivação pelo uso de cinco dias por semana da ferramenta de lançamento de notas, assim podendo ter esse uso frequente com o aplicativo *learcod* também.

Na categoria que indica *softwares* similares foram poucas sugestões, uma delas é o *BOCA Online Contest Administrator*, o intuito dessa questão no formulário que foi aberta era para dar algumas ideias para que possa inserir no aplicativo *learcod*, mas nesse caso analisando o aplicativo BOCA foi o que não colocaria, há um demora no *feedback* para os discentes quando enviam seus códigos para os docentes e isso era algo a tentar ser sanado no aplicativo *learcod*, como também quando enviava um código que funcionava corretamente, mas de forma diferente como o docente colocou no padrão, isso deixaria o discente ficar restrito a algum tipo de código e não poder fazer diversas formas, essas análises foram anotadas e estão em construção para serem feitas em trabalhos futuros.

3.3.4 Avaliação de Viabilidade com Discentes

Nesse questionário obteve cerca de 32 (trinta e dois) resultados dos discentes para análise no todo de 36 (trinta e seis) que compõe tanto docentes e discentes no levantamento de viabilidade, podendo ser analisado abaixo o quadro 2.

Quadro 2. Análise do Levantamento de Viabilidade dos Discentes.

Categorias	Análises dos Dados
Plataformas de Ensino	Com 40,6% dos discentes usam as plataformas de ensino até 5 vezes na semana, já 25% usam todos os dias da semana e 28,1% usam entre de 1 até 2 dias na semana.
Proveito das NTDIC's	Com 93,8% os discentes afirmaram que há um proveito das NTDIC's nas avaliações.
Viabilidade das NTDIC's em programação	Com 84,4% dos dados é viável o uso das NTDIC's para as avaliações.
Experiência em NTDICs	Cerca de 93,8% dos resultados, afirmaram que tem algum tipo de experiência com as NTDIC's.
Motivação a estudar programação	Com 96,9% dos discentes disseram que se sentiriam motivados a estudar mais programação com o uso de um aplicativo.
Uso do controle de frequência	Com 56,3% dos discentes disseram que afirma que gostaria que fosse aplicado um tipo controle de frequência no aplicativo, já 18,8% não gostariam e 25% responderam que não sabem.

Por meio da coleta e análise dos dados foi possível perceber um retorno positivo em relação a viabilidade do aplicativo educativo, no quesito de ter experiência em NTDIC's confirmaram que possui aptidão para manusear as Novas Tecnologias Digitais de Informação e Comunicação, já se um aplicativo seria motivante para estudar programação, também respondeu que isso poderia os motivar a estudar, pois

acredita-se que pela área de Computação que estão inseridos, o uso de uma ferramenta como um software pode ser interessante para o auxílio do aprendizado. Como também concordam que há um proveito e é viável o uso das NTDIC's no contexto de avaliações em componentes curriculares de programação, assim trazendo mais uma avaliação perto das atividades que eles estão familiarizados a fazer e de forma processual, podendo ser analisado pelo software esses detalhes de progresso de desenvolvimento de aprendizado e avaliação.

No cenário que diz se os estudantes gostariam que tivesse um controle de frequência como se fosse a presença marcada em sala de aula no aplicativo, das 32 respostas 56,6% responderam que sim, isso pode significar que de acordo com a quantidade de pessoas que frequentam a plataforma de ensino na categoria discutida no quadro 1 que frequentam até 5 (cinco) vezes na semana, seria bom para eles esse controle de frequência, já para os estudantes que normalmente não frequentam com tantas vezes por semana, esse controle de frequência no aplicativo não seria uma abordagem que faria ele gostar tanto assim, provavelmente por questões de falta de tempo por ter o emprego corrido semanalmente e ter que focar nas atividades e não conseguir focar nessas ferramentas/software, dentre outros motivos que podem ser analisados em novas pesquisas, pois essa questão é preciso analisar com mais detalhes nas próximas etapas, caso for iniciar novas coletas em trabalhos futuros.

A partir dos resultados coletados do questionário de viabilidade tanto dos discentes como dos docentes, é possível perceber que, a ideia como solução de proposta do aplicativo educativo *learcod* se mostra viável para auxiliar o aprendizado e avaliação processual no ensino de linguagem de programação. Com isso, foi possível seguir com a produção do fluxo de atividade do aplicativo educativo para dar progresso ao desenvolvimento do *design* de *software* educativo que se mostra a seguir.

3.3.5 Fluxo de Atividade do Aplicativo Educativo

Nesse fluxo de atividade que tem que como característica investigar soluções e parte da ação do trabalho, foram feitos dois exemplos como na figura 4 com a linguagem de algoritmo e na figura 5 com a linguagem de banco de dados, mostrando essa possível forma de fazer exercícios fixos para linguagens diferentes. Como também apresentados por completo o fluxo de atividade no Apêndice E – uma

sequência didática de algoritmo e do Apêndice F – uma sequência didática de banco de dados.

O fluxo de atividade tem algumas descrições como: ícone de dúvidas que pode ser visto na figura 4: será possível entender como funciona determinado botão por meio de leitura de um texto ou áudio descritivo; no tópico de linguagem, pode ser o idioma definido em nível de linguagem, posteriormente por dificuldade, assim sucessivamente, exemplo de linguagem Portugal - idioma em português; em todo o aplicativo é utilizado de ferramentas tecnológicas que contemple a acessibilidade apresentado na figura 3 para identificação de áudio em todos os componentes, entender bem os detalhes por sua escrita. Em relação algumas necessidades do aplicativo, é preciso habilitar o áudio descritivo para todo o aplicativo, que será logo ao abrir o aplicativo pela primeira vez instalado; dentro do conceito de variável por exemplo, tem minietapas para fechar o círculo, dentro de cada mini etapa tem mais subetapas que são os outros exercícios; caso houver imagem, o botão de acessibilidade pode ser acionado para áudio descrição como na figura 3. Repetição Espaçada que pode ser vista nas figuras 3, 4 e 5: no exercício que o usuário responder, caso erre, será repetido a explicação para lembrar os conceitos. Posteriormente, será continuado os outros exercícios que ainda não fez, quando tiver perto do fim, será mostrado novamente o exercício que errou, caso acerte passe a diante, senão refazer novamente todo o passo até acertar.

No fluxo de atividade organizado para a prototipação do aplicativo *learcod*, também é possível perceber dois modos de funcionamento do usuário, o modo focado e difuso. No modo focado, o foco de prestar atenção em algo que estamos fazendo, pode ser um exemplo como, a minitarefa no aplicativo *learcod*. Já no modo difuso seria basicamente você “moscar” metaforicamente dizendo, nesse caso pode fazer qualquer outra atividade, como tomar uma água, conversar com alguém, fazer uma caminhada, tirar um cochilo, comer um sanduíche, apenas uma pausa e depois voltaria ao modo focado novamente, esse tempo pode variar de pessoa para pessoa.

Para se referir ao aplicativo *learcod* no aspecto motivacional, podemos citar o que envolve a idealização de possibilidade de colocar estilos de aprendizagem que adequem para grupos ou até mesmo cada estudante; também pode visualizar a inserção de atividades em grupos, colocar tarefas que possam ser feitas em grupos e

que cada estudante possa ir fazendo sua parte nesse caminho, cada um contribuindo com seu conhecimento e trocando de ideia com o outro colega; os feedbacks são vistos imediatamente, após o resultado de uma questão sendo correta ou incorreta, e quando correta é merecido de um prêmio em formas de medalhas ou outros tipos; quando você erra a questão não irá tirar um ponto, só irá para a parte da explicação apresentada na figura 2, para que possa rever o conteúdo, isso seria uma ideia de autoconfiança, para que não se desestimele daquele processo e ao acertar terá a recompensa de prêmios ou medalhas para dar essa sensação de autoeficácia.

É possível destacar que, todos esses conceitos, também são fundamentais a autonomia e são levados para o *design* do aplicativo learcod e posteriormente para implementados, pois é possível governar o seu próprio aprendizado, ir de acordo com seu ritmo, possibilitando o desenvolvimento dessas habilidades da autonomia, que possa fazer por si quando quiser e a hora que puder, assim podendo fluir melhor no processo de desenvolvimento de suas atividades.

Ademais, como mostrados nos apêndices E e F, o fluxo de atividade pode ser feito com quaisquer tipos de linguagens de programação, pois no apêndice E foi produzido com a linguagem de algoritmo e no apêndice F com banco de dados, assim pode ser criado um novo fluxo de atividade com outra linguagem de programação qualquer que contemple essa dinâmica de exercícios propostas do fluxo de atividade. Assim, nesses fluxos de atividades produzidos terão seus subtópicos de conceitos e aprendizados, podendo representar o aplicativo com as diversas questões específicas para esse conceito e a explicação adequada. Para cada exercício há um tipo fixo de forma, como por exemplo, uma pergunta e imagens a serem definidas abaixo – para acessibilidade terá áudio descrição – nos exercícios 1.A, 2. B que pode ser visto na figura 3, 3.C 5.E terão questões de assinalar tendo suas particularidades de questões individuais, no 4.D será de combinar pares, no exercício 6.F que pode ser visto na figura 4, irá ter um código pronto e terá que saber qual o resultado desse código de acordo com a pergunta feita, ou terá que fazer um código de acordo com as palavras que estão acima e terá que encaixar na ordem correta para que execute de acordo com a pergunta, também é possível ter exercício como no 7.G com perguntas que possam relacionar caixas de imagens com textos dos códigos, outro como o 8.H que os códigos estão embaralhados e precise colocar em ordem correta para que esteja

de acordo com a questão correta, no exercício 9.I é possível que arraste o campo na ordem correta da linha do código abaixo para que execute e passe adiante.



Figura 2. Fluxo de Atividade da Explicação dos Exercícios de Algoritmo.

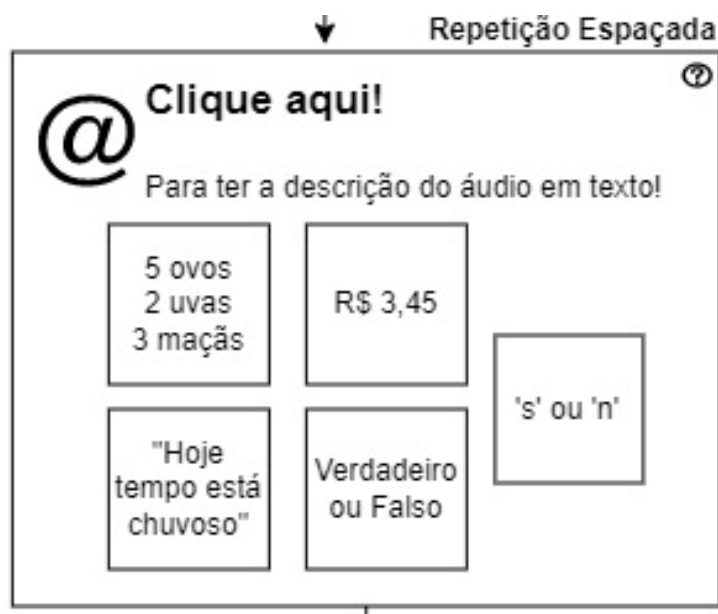


Figura 3. Fluxo de Atividade do Exercício de Assinalar de Algoritmo.

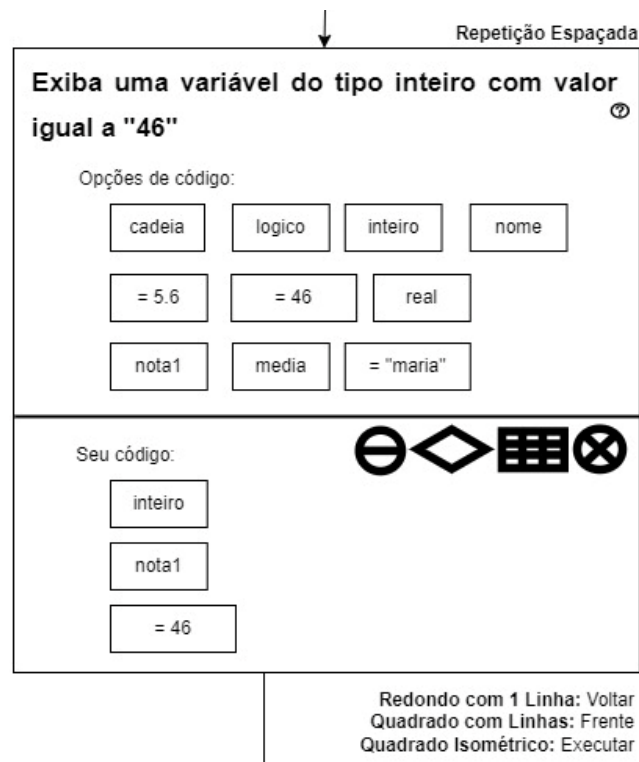


Figura 4. Fluxo de Atividade do Exercício de Codificação de Algoritmo.



Figura 5. Fluxo de Atividade do Exercício de Codificação de Banco de Dados.

Esse fluxo de atividade foi uma idealização de propor exercícios fixos para que possa repetir para quaisquer linguagens, assim podendo adicionar em todos os desafios, tanto que na tela de desafio do docente é possível adicionar desafios com quaisquer um desses exercícios com quaisquer tipos de conteúdo de linguagem.

3.3.6 Considerações da escolha do Software de Prototipagem

O processo de escolha do software para prototipagem foi caracterizado como a parte de investigação de soluções da pesquisa. Partindo desse ponto, a análise do software de prototipagem, foi feito por meio do site Capterra¹ que ajuda encontrar diversos softwares de quaisquer áreas e saber um pouco sobre eles e conseguir compará-las, também é possível por meio desse site ter avaliações e fazer comparações entre os programas para que possa encontrar o software adequado ao seu objetivo final.

Podemos destacar que, não há como afirmar sobre qual a melhor ferramenta para o uso de prototipagem de tela, o que pode ser feito direcionar a escolha a partir das análises de características de cada uma delas e a identificação da que melhor se encaixa com o perfil do seu objetivo do projeto em ação. Para tanto, as perguntas que foram analisadas para escolher o software a ser usado na prototipação das telas e no processo do design do aplicativo educativo *learcod* foram – Tem alguma versão gratuita? Há integração com outras plataformas? Há possibilidade de gerar códigos? É possível compartilhar em tempo real? É simples, rápido e fácil de ser usado? Há interações/animações das passagens de telas? Saber as respostas para buscar igualar com as características é importante a primeiro momento.

Com isso, pudemos analisar que as respostas condiziam com algumas ferramentas, como: JustinMind, Invision, UXPin, WebFlow, FramerJS, pois a maioria dos seus pontos característicos são condizentes com boa parte das respostas de nossas premissas de objetivos do trabalho, porém o WebFlow e FramerJS não foram selecionados, porque não tem opção para desktop, apenas navegador web, o que deixaria mais lento o processo em relação a internet e algumas funcionalidades. Para decidir qual ferramenta a ser utilizada foram realizados testes a partir do uso de cada uma delas para analisar os aspectos de experiência. As ferramentas que ficaram em

¹ Link do Site de Comparações de Software de Prototipagem: <https://www.capterra.com.br/>

análise foram o JustinMind e UXPin. Por fim, o software escolhido para esse processo de prototipagem foi JustinMind, pois além dos dois softwares terem a possibilidade de compartilhar a simulação prévia, o JustinMind é possível por meio dos eventos ter a interação de telas, o que é interessante para esse tópico de prototipagem. Com isso, dá-se seguimento ao desenvolvimento das telas na ferramenta escolhida.

3.3.7 Prototipação

A prototipagem de telas tanto do perfil de usuário discentes como dos docentes tem como parte da ação do trabalho da pesquisa e traz a ideia de mostrar ao usuário como está sendo desenvolvido o aplicativo educativo. Por meio das telas da prototipação é possível fazer interação por uma simulação² para que possa ter mais conhecimento das funções de como vão ocorrer. Ademais, com todo o processo de questionários de usabilidade é analisado e discutido campos que possam ser atualizados e melhorados no desenvolvimento do aplicativo. Adiante foi dividido em duas partes para exemplificar as telas da prototipação, em Prototipação Perfil de Usuário Discente e Docente.

3.3.7.1 Prototipação do Perfil de Usuário Discentes

As telas do perfil de usuário discentes foram concentradas na parte de ensino e aprendizado, pois o objetivo é o aprendizado das linguagens sejam quaisquer uma delas, podendo estar relacionadas com alguma turma de um docente ou não, também é possível fazer desafios extras para poder desenvolver o aprendizado, caso já tenha terminado o fluxo de atividades, é possível também visualizar as estatísticas do usuário ao decorrer do percurso de tarefas que tem feito, bem como outras funções dentro do aplicativo.

² Link da Simulação da Prototipação: <http://ifbaprojetos.com/learcod>

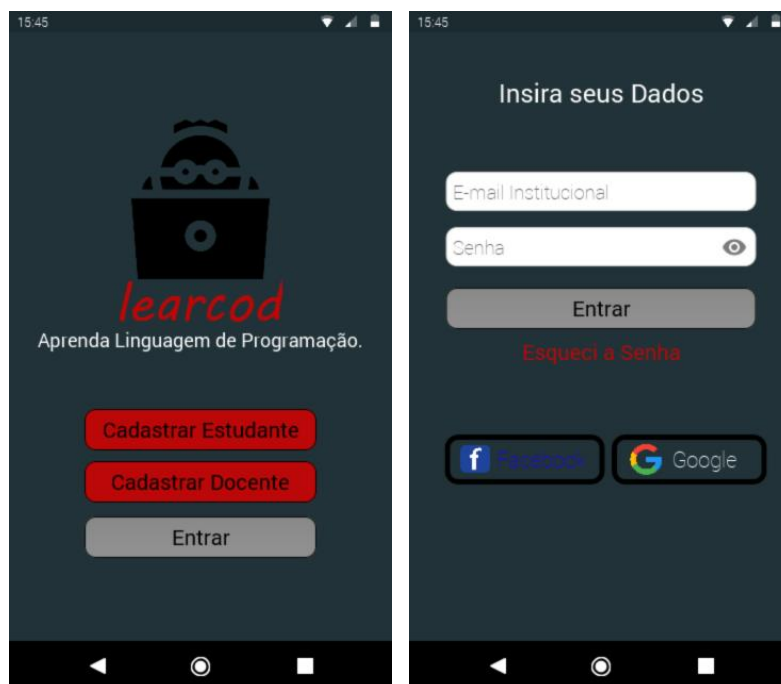


Figura 6. Tela Inicial do Aplicativo *Learcod* para os Usuários.

Na figura 6 apresenta duas telas iniciais da prototipagem do aplicativo educativo *learcod*, a primeira é onde mostra como irá logar no sistema mostrando o logotipo, nome e slogan da marca do app, bem como os botões de cadastro dos usuários de estudantes e docente, o estudante é considerado nesse caso como discente também pela nomenclatura. Após cada usuário realizar seu devido cadastro de acordo com seus requisitos de campos, é possível fazer o login igualmente para ambos pelo e-mail institucional, pois ambos oferecem esse suporte, bem como o acesso a funcionalidade de esqueci a senha, assim podendo ter acesso ao sistema do aplicativo *learcod*, que poderá ver mais telas a seguir.



Figura 5. Tela de Fluxo das Linguagens Disponíveis para o Usuário.

A figura 5 mostra o fluxo de exemplos de algumas linguagens que podem ser inseridas nesse aplicativo como conteúdo para estudo, sendo que nesse fluxo percebe-se que precisa seguir uma ordem para ir para a outra etapa, pois o algoritmo é uma linguagem inicial de ser aprendida e a linguagem POO (Programação Orientada a Objetos) é mais avançada, assim sendo posteriormente estudada, seguindo uma ordem de estudo, bem como o de estrutura de dados também, dentre os outros.

O preenchimento na cor vermelha significa que cada vez que dentro dessa linguagem de programação selecionada você termina alguma tarefa ou desafio irá aumentar seus pontos, assim aumentando o preenchimento de acordo com quanto de exercícios que fez nesse fluxo, caso tenha finalizado todas as atividades de algoritmo por exemplo, o quadrado dele estará todo preenchido em vermelho, caso não tenha feito nenhuma atividade não terá nenhum preenchimento em vermelho. Esse fluxo irá mostrar qual deles irá escolher para qual irá estudar e mostra qual preenchimento está das atividades que já concluiu até o momento.

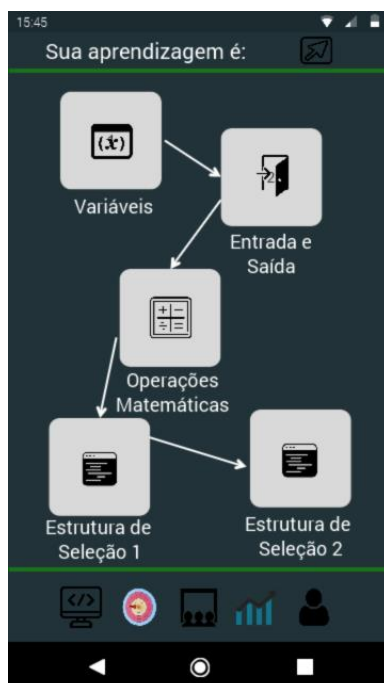


Figura 6. Tela de Fluxo das Subtarefas de uma Linguagem específica para o Usuário.

Após a seleção de uma das linguagens apresentadas na figura 5, é possível entrar nos conteúdos específicos aparecendo o fluxo em ordem, conforme figura 6, nesse caso os conteúdos são acumulativos, ou seja, eles precisam ser feitos de acordo com a ordem, não dá para iniciar a estrutura de seleção 1 sem ter feito a parte de variáveis primeiro, por isso é preciso seguir o fluxo das atividades para terminar todas as etapas.



Figura 7. Tela de Estatística de Dados do Usuário.

Na tela de estatística é possível ver tanto para a parte do discente quanto para a parte do docente. Para o discente, consegue visualizar o quanto teve de pontuação de medalhas em desafios, qual é sua meta diária, qual o nível que está de acordo com a etapa da linguagem que está estudando e fazendo as tarefas, consegue também verificar o nível de experiência (XP) obtido por dia, dentre outros. Em relação ao docente, é possível visualizar as mesmas características do discente, porém dividido por turmas e dentro de cada turma será possível ter os discentes, sendo possível ver o ranking por turma. Contudo, o ranking só é visto pelo docente, podendo visualizar especificamente um estudante e verificar suas estatísticas detalhadas, conseguindo ter todo o progresso do seu estudo de certa forma.

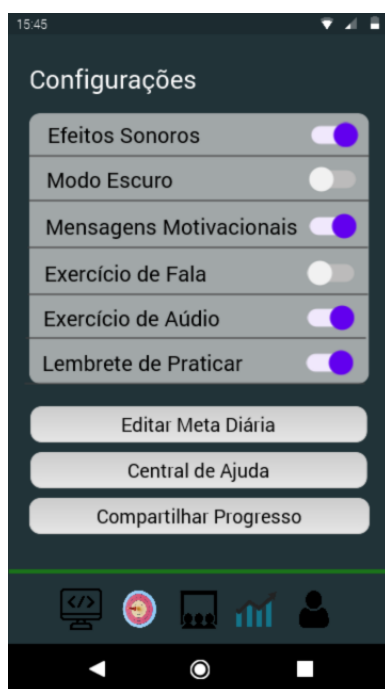


Figura 8. Tela de Configurações de Perfil do Usuário.

No tópico 2.4 desse trabalho foi descrito sobre acessibilidade, nessa tela, figura 8, de configurações do perfil tem algumas partes relacionadas a acessibilidade que podemos distinguir como exercício de fala que pode ser acionada para que possa usar a voz para resolução dos exercícios, não sendo necessário usar as mãos para resolver, digitar ou apenas tocar nos botões. Na parte de exercício de áudio, direcionado para deficientes visuais, o usuário pode utilizar dessa função para ajudá-lo nas atividades por meio de leitor de tela vinculado ao aplicativo.

Dessa forma, algumas percepções foram colocadas, como a acessibilidade para deficientes visuais, em que é fundamental em casos de ter imagens ou botões

sem descrições, assim necessitando de um áudio-descritivo. Portanto, o aplicativo *learcod* tem o intuito de propor a ideia de trazer a acessibilidade, pois textos, imagens, botões normalmente são inacessíveis, fazendo com que tenha uma segunda opção de descrição auditiva irá facilitar esse processo de ensino e aprendizado para esse estudante.

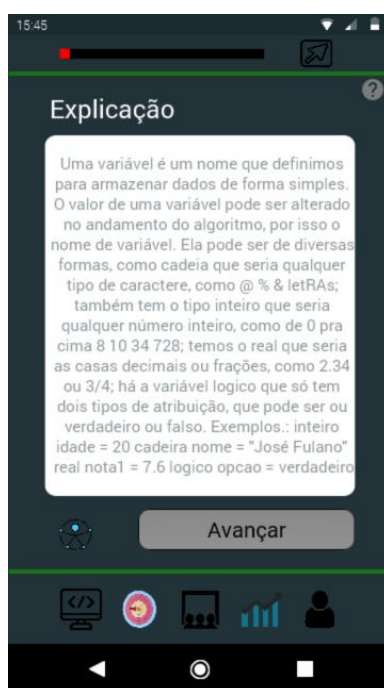


Figura 9. Tela de Explicação Antes do Exercício para o Usuário.

Antes de iniciar qualquer exercício é preciso uma explicação, algum conteúdo teórico que possa esclarecer esse conceito para o discente, assim a tela apresentada na figura 9 mostra o assunto específico daquela atividade que possa gerar compreensão anteriormente ao exercício, e/ou que já tenha estudado em sala de aula com o docente, pois esse é apenas um auxiliar, assim podendo avançar parar a questão e respondê-la, caso esteja correta irá avançar, caso esteja incorreta irá voltar novamente a tela de explicação até que acerte o exercício proposto.



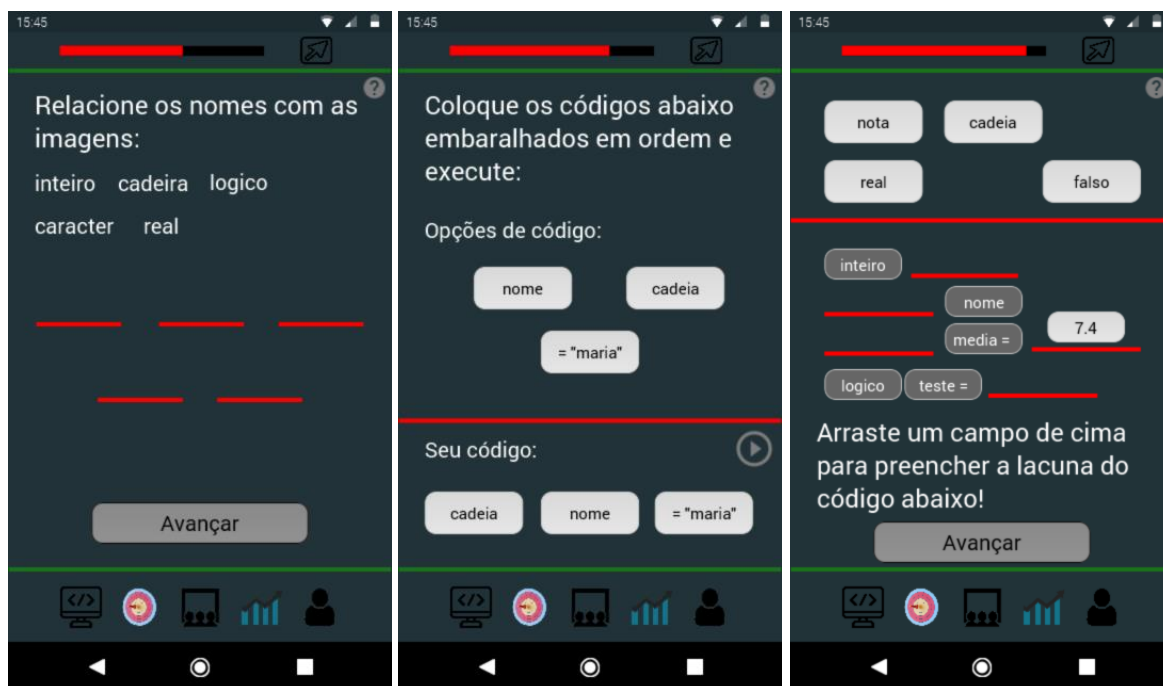


Figura 10. Telas dos Exercícios Existentes de Algoritmo para o Usuário.

Na figura 10 que será descrito em ordem de cima para baixo da direita para esquerda, como a primeira imagem sendo 10 (a), a segunda imagem 10 (b), sucessivamente, 10 (c), 10 (d), 10 (e), 10 (f), 10 (g), 10 (h), 10 (i). Nessas telas são exercícios que foram pré-montados para que pudesse entender um pouco como funcionaria as questões de acordo com os conteúdos. Em cada exercício tem uma barra de progresso em que se avança na medida que vai passando as questões, também tem um ícone de ajuda que é a interrogação, como se fosse um manual para caso esteja tendo alguma dificuldade, também tem na primeira tela o ícone de acessibilidade que pode ser acionado para abrir as configurações devidas e escolhê-las, o botão de clique pode voltar na tela do fluxo de linguagem caso queira sair dos exercícios.

Na tela 10 (a) e 10 (c) tem o exercício que são de assinalar com uma alternativa correta de acordo com a questão, a primeira questão a depender da pergunta pode conter imagens se possível. Na tela 10 (b), no exercício é possível ouvir o áudio e responder escolhendo algumas das alternativas, caso não queira ouvir o áudio é possível acionar a descrição. Na tela 10 (d), tem uma questão que combina os pares que tem correlação e caso tiverem certos irá avançar para a próxima pergunta. Na tela 10 (e), a pergunta é mostrado um código com uma pergunta que terá que ser respondido com alguma das alternativas abaixo. Na tela 10 (f), o exercício tem que

puxar os blocos de cima para baixo da linha vermelha, podendo refazer ou avançar com os botões do lado, após colocar em ordem o código de acordo com a pergunta, pode executar o código clicando no ícone de *play* para verificar se funcionou. Na tela 10 (g), a questão irá relacionar os nomes dos códigos com as imagens abaixo. Na tela 10 (h), o exercício funciona da mesma forma que a questão seis, porém não tem como refazer e avançar, assim dificultando mais, deixando assim para o estudante um desafio maior. Na tela 10 (i), no exercício terá que arrastar os campos de acima para baixo em cima das linhas que acha que fará sentido para que o código funcione com um todo no final.

3.3.7.2 Prototipação do Perfil de Usuário Docentes

Na prototipagem das telas para o docente é pensado mais na parte de avaliação, assim o docente consegue visualizar o processo de aprendizado do discente de forma contínua e processual podendo ser formativa essa avaliação em sala de aula. Assim, pelo aplicativo educativo é possível ver todo o progresso dos estudantes de cada turma que o docente leciona de acordo com a turma que ele irá cadastrar e o estudante cadastrado com a turma também, além disso, é possível cadastrar desafios para as turmas para que os estudantes possam fazer caso já tenham terminado todas as etapas dos conceitos dados da aula, também é importante a aba de estatística onde que é visível os dados de todos os estudantes por turma de estatísticas de pontos, medalhas, desafios conquistados, assim percebendo como foi esse progresso contínuo e processual ao longo dos dias.

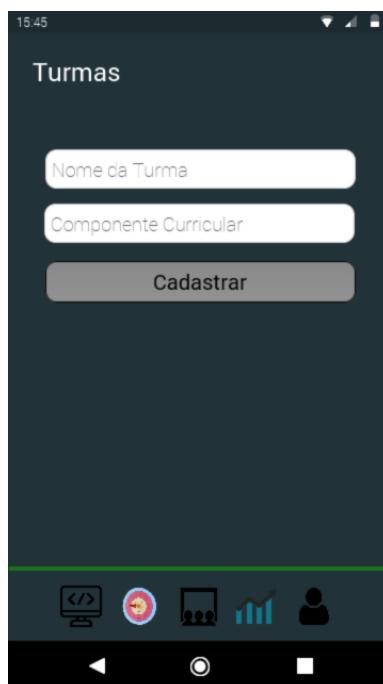


Figura 11. Tela de Criação de Turma pelo Docente.

No processo de cadastro de turma, o docente irá cadastrar o componente curricular, após isso irá receber uma chave para que o estudante possa ingressar nesse componente específico podendo acessar os módulos ou na figura 9 podendo cadastrar diretamente o discente.

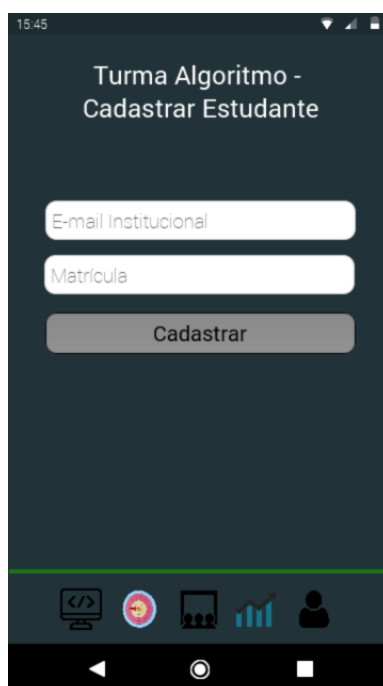


Figura 12. Tela de Cadastro de Estudante por Turma pelo Docente.

Nessa tela o docente pode cadastrar diretamente o estudante por meio da turma que ele deseja inserir, assim não precisando inserir alguma chave, caso tenha algum tipo de problema ou o estudante esteja tendo alguma dificuldade, após isso, o estudante terá acesso a turma do componente matriculado e poderá acessar os módulos com os conteúdos para realizar as atividades no aplicativo.

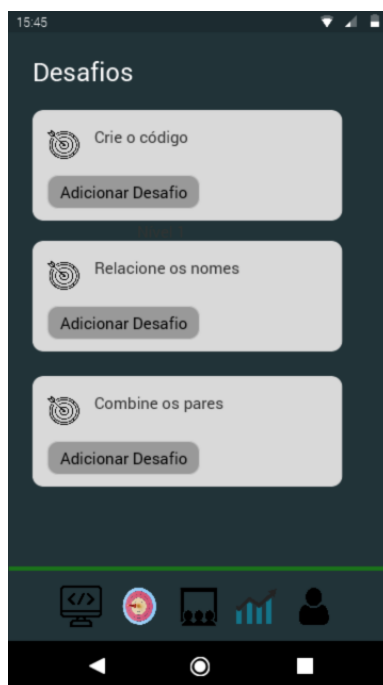


Figura 13. Tela de Adicionar Desafios pelo Docente.

Na figura 13 pode-se visualizar a possibilidade de criar desafios pelo docente para discentes para a turma específica, o intuito é que queira aumentar de pontos na sua estatística caso tenha terminado suas tarefas do módulo principal. Além disso, é visto que, os desafios são propostos de acordo com os mesmos exercícios fixos dos módulos, porém podem ser adaptados com o tipo que queira inserir com quaisquer conteúdos, sendo do tipo de combinação de pares com conteúdo de banco de dados, web, java ou com desafio que relacione os nomes com algum desses conteúdos.

Figura 14. Tela de Descrição dos Campos de Adicionar Desafio pelo Docente.

No momento de adicionar um tipo de desafio será pedido alguns campos, como título, descrição do exercício, pontuação do exercício que será contabilizado na estatística, as opções de entradas separadas por ponto e vírgula e as opções de saída também separadas por ponto e vírgula, todos esses campos ou outros podem estar sujeitos a alterações. Esses campos são enviados para o tipo específico do exercício que é montado e então o desafio é feito para que o estudante o realize.

3.4 Aplicação do Questionário de Usabilidade

Após a produção da prototipagem foi feito o questionário de usabilidade da prototipagem do aplicativo *learcod* que teve como parte observar e refletir sobre os dados, também tem como objetivo de colaborar na construção de forma participante a partir dos dados coletados desse questionário. A análise viabiliza repensar, discutir e reformular todos os passos que possam ser alterados ou inseridos em atualizações da prototipação e implementação do aplicativo. Esse questionário foi dividido tanto para docente como para discente com perguntas diferentes para ambos. Os dois formulários tinham duas partes, uma de usabilidade tecnológica, que significa o processo de prototipação de telas antes da implementação de códigos para os testes de usabilidades. Como também teve uma parte de aspectos pedagógicos, que para além da tecnologia é preciso verificar a relação pedagógica, com isso foi realizada

algumas perguntas para verificar o contexto do ensino e aprendizado do estudante em relação a sua aprendizagem no componente curricular que estiver cursando.

4 Resultados e Discussões

O design do aplicativo educativo *Learcod* desenvolvido neste trabalho, e o seu direcionamento para implementação é pensado para ser utilizado como auxiliar durante aulas e atividades de quaisquer componentes de Linguagem de Programação, tanto no aprendizado para os estudantes como para a avaliação que os docentes terão ao longo do processo do desenvolvimento contínuo dos discentes. Para isso, a obtenção dos dados de coleta para avaliar a usabilidade por meio da experiência do usuário com base no manuseio das telas de prototipagem do *learcod* direcionam as análises e discussões desse trabalho.

4.1 Avaliação da Experiência de Usuário com Docentes

O questionário de usabilidade que tem a característica de observar e refletir sobre os dados da pesquisa, tem também como objetivo avaliar a experiência do usuário com o uso da prototipagem do aplicativo educativo *learcod* como auxiliar em componentes curriculares e no processo de aprendizado e avaliação. Esse formulário foi enviado para professores e estudantes do *campus* Porto Seguro todos da área de Computação, os resultados obtidos no total com discentes e docentes foram 24 (vinte e seis), em que apenas 8 (oito) docentes responderam ao questionário, mesmo com a baixa adesão dos dados coletados, foram analisados e discutidos, mas entende-se que em ações futuras se faz necessário repetir e ampliar as coletas para o perfil de usuário docente.

Para esse formulário foi dividido em duas seções, uma em Usabilidade Tecnológica em que o intuito é que o processo de prototipação de telas antes da implementação de códigos minimize os erros usando os testes de usabilidades, esse questionário tem como objetivo avaliar a experiência do usuário no uso do aplicativo educativo *learcod*, que será repensado e reformulado, discutido para ser analisado e descrito todos os passos que possam ser alterados ou inseridos. Como também tem a seção de aspectos pedagógicos que, irá analisar o contexto do ensino e aprendizado do professor em relação a sua avaliação com os estudantes.

Quadro 3. Análise da Avaliação da Usabilidade dos Docentes.

Categorias	Análises dos Dados
Acessibilidade	Com 2 (duas) docentes responderam que o aplicativo é adaptável para as necessidades de pessoas com deficiência, já 3 (três) integrantes responderam talvez e um disse que não sabe e os outros dois deram opiniões diferentes do padrão existente.
Facilidade de Uso	Com 5 (cinco) docentes afirmando que acharam fácil em usar a prototipação de telas, já um docente disse que foi indiferente e 2 (dois) responderam que discordam parcialmente.
Experiência Prévia	Com 7 (sete) dos docentes discordam que precisa aprender muitas coisas antes de usar o aplicativo para entender a manuseá-lo, já um confirmou que precisa de algum tipo de experiência antes.
Modos Avançados	Apenas 2 (dois) docentes disseram que encontraram meios que pudessem ter modos avançados de aprendizado de acordo com a experiência do usuário (como os desafios), já 3 (três) docentes não sabiam dizer, um respondeu que não tinha, um também informou que talvez e outro docente relatou que não conseguiu identificar a função.
Avaliação de Aprendizagem	Com 5 (cinco) dos docentes responderam que o aplicativo facilita ao processo de avaliação para aprendizagem, já um disse que não sabe e 2 (dois) responderam que talvez.
Satisfação Geral	Cerca de 3 (três) dos docentes se sentiram satisfeitos com o aplicativo <i>learcod</i> , já 4 (quatro) responderam indiferente e um respondeu que não ficou satisfeito.

Na categoria acessibilidade é possível perceber que os dados não foram equivalentes, assim a acessibilidade na prototipagem não foi vista de imediato, tendo que ser revista e analisada, como os botões precisando ser mais evidente e colocados diretamente na primeira tela de forma visível no canto direito superior. Por exemplo, um dos resultados foi que “depende do tipo de deficiência. Para cegos ou portadores de baixa visão, por exemplo, ele não trouxe acréscimos em relação à outras plataformas”, com isso, nas configurações, pode ser especificado os tipos de deficiência para que possa identificar melhor a acessibilidade, pois não estava bem visto isso, além do uso de uma configuração do uso de vídeos para simular interpretação de Libras nos exercícios na parte de linguagem para que o estudante possa compreender, principalmente os termos técnicos.

No quesito de facilidade de uso, teve um resultado positivo da prototipagem, isso indaga que, houve uma boa experiência do usuário com a ferramenta para os docentes que utilizaram, assim não necessitando de tutoriais ou manuais para entender sobre o uso do aplicativo. O fato de serem professores da área de Computação pode indicar uma experiência mais aprofundada em softwares, plataformas educativas, isso também sendo um fator influenciador nessa questão de experiência de usuário para o uso da prototipação.

Na categoria experiência prévia, concordaram que não é necessário aprender tantas coisas antes de usar o aplicativo/prototipagem, isso induz que a prototipagem propôs com o intuito que era de não precisar de fazer algum tipo de tutorial antes para usar o aplicativo e sim mexer por conta própria já seria suficiente para o entendimento e o fluxo de usabilidade, pois a fluidez o tornava independente para essa ação e um usuário explorador da ferramenta.

Com os resultados, percebe-se que não há tanta clareza de como ficará o aplicativo final para ter opiniões mais esclarecedoras, porém é possível analisar que ao longo de todo o percurso e de resultados também satisfatórios, o aplicativo se mostra mais voltado para a gamificação, não como um todo, mas um pouco de seus aspectos de jogos, a idealização de trabalhos futuros do *Coding Dojo* também é citada, tudo isso pode ser analisado como parte da aprendizagem (Debald, 2020). Com isso, os docentes também não conseguiram perceber no aplicativo essa demanda de modos avançados como os desafios que seriam as atividades de desafios para se fazer que estavam expostos nas telas e escrito como na figura 10, assim precise de mais ajustes da prototipação para possíveis esclarecimentos e novas coletas de dados.

No que facilita o processo de avaliação de aprendizagem usando o aplicativo *learcod*, foi analisado os parâmetros que abordam a continuidade de estar sempre acompanhando rotineiramente ou semanalmente as tarefas que possivelmente os estudantes estão fazendo, como também já poder enviar algum tipo de desafio ou avaliação de forma remota, sem necessariamente estar em sala de aula para que isso ocorra, um ponto a ser considerado na parte de avaliação desse aplicativo é que ele poderá ser um meio que irá auxiliar no processo de ensino e aprendizado no componente curricular, mas que não será o fator primordial para a avaliação. Assim,

de acordo com os dados, concordaram que o aplicativo facilita o processo de avaliação do processo de aprendizagem, portanto auxiliando em sala de aula os estudantes, já outra parcela respondeu “não sei”, possa ser que não conseguiram compreender se seria possível essa avaliação, sendo necessário uma prototipação mais completa ou o aplicativo funcionando para poder esclarecer.

No geral, a satisfação em relação aos dados coletados confirmaram que teve uma boa experiência do usuário ao usar a prototipagem fazendo com que não seja tão complexa, a ideia do aplicativo que é instigadora para a aprendizagem de quaisquer linguagens de programação para estudantes, bem como o processo de avaliação que pode ser analisado pelos docentes, trazendo esse interesse do uso para auxiliar em componentes curriculares em sala de aulas motivando os estudantes e auxiliando os professores no desenvolvimento de avaliação. Porém, também tiveram docentes que responderam como indiferentes, assim tendo uma visão que necessita mais de uma prototipagem detalhada e completa ou possivelmente o aplicativo final, para que possa ter opiniões mais esclarecedoras.

Para discutir as outras questões analisadas no questionário, como “Gostaria de usar este aplicativo com frequência.” teve cerca de 5 (cinco) docentes que concordaram com essa sentença, mostrando que esse aplicativo pode auxiliar no processo de ensino e aprendizado dos estudantes em sala de aula e fora, os outros 2 (dois) que marcaram com indiferente é que ainda não tem uma visão mais específica e detalhada de como vai funcionar o aplicativo, necessitando o aplicativo final ou a prototipação completa para opiniões mais certas. Na questão “Achei o aplicativo complexo no geral.” com 7 (sete) dos docentes dizendo que discorda, isso conclui que é simples usar o aplicativo.

Os 3 (três) docentes afirmaram que é possível criar testes, exames ou tarefas usando o aplicativo, podendo perceber o uso de criação de desafios e novas tarefas com o fluxo de atividade com quaisquer linguagens de programação, os outros 3 (três) docentes responderam não sei e 2 (dois) talvez, precisa estar mais claro essa dinâmica na prototipagem por meio das palavras, pois o processo de avaliação ocorre de forma formativa e contínua como menciona na fundamentação teórica 2.2. Com 3 (três) docentes afirmando que o aplicativo complementa outras atividades do componente relacionado a área de programação, e, 4 (quatro) docentes responderam

indiferente e um com resposta que não sabe, assim concluindo que, não deixou explícito o aplicativo a possibilidade de fazer atividades interdisciplinares, assim podendo fazer esses reajustes.

Os 4 (quatro) docentes afirmaram como indiferente na possibilidade de que os usuários possam preparar uma apresentação usando o aplicativo, e, os outros 4 (quatro) docentes ficaram divididos entre sim, não, depende do nível de conhecimento do usuário e caso seja sobre as telas sim, dessa forma, é preciso analisar preciso o quadro de dados para que possa fazer novos testes para ter uma nova base de dados, ou seja, fazer um teste com o próprio aplicativo ou a prototipagem mais completa com os ajustes feitos com os professores e estudantes usando em sala de aula juntos e posteriormente fazer essa nova coleta de dados para esclarecimentos. Com 8 (oito) docentes afirmando que recomendaria o aplicativo para fins de aprendizado e avaliação, e, um dizendo que precisa de ajustes, mas que tem futuro para o seu uso, assim concluindo que é necessário novos ajustes para que possa fazer novas coletas, mas que seu intuito está sendo avaliada de forma satisfatória.

Os 3 (três) docentes no geral, se sentiram satisfeitos com este aplicativo, e, 4 (quatro) docentes acharam indiferente, para essa análise é possível que ainda seja pela possível mudanças na prototipagem que deve ser feita posteriormente, assim sendo feita, essa prototipagem mais ajustada e completa ou até mesmo com o aplicativo final, pode ser levado mais em consideração essa satisfação em não indiferença.

4.2 Avaliação da Experiência de Usuário com Discentes

Nesse formulário de usabilidade que compõe parte da caracterização da pesquisa-ação em observar e refletir sobre os dados, obtiveram resultados no total com discentes e docentes cerca de 24 (vinte e seis) respostas, na contagem de discentes apenas foram 16 (dezesesseis) responderam ao questionário. Para esse questionário foi dividido em duas sessões, uma em usabilidade tecnológica e aspectos pedagógicos, para que possam ser avaliados a experiência do usuário e a aprendizagem e o processo de avaliação do aplicativo educativo *learcod*.

Como já analisado com os docentes, também procede quase o mesmo comportamento, de dados inconsistentes, assim confirmando que são necessários os ajustes na prototipagem em relação a acessibilidade, como os botões serem mais

visíveis em primeira tela, nas configurações ser mais específico os tipos de deficiências para que possa esclarecer as demandas e a possível adaptabilidade de vídeo em exercícios das tarefas da linguagem de estudo do usuário para simular interpretação em Libras.

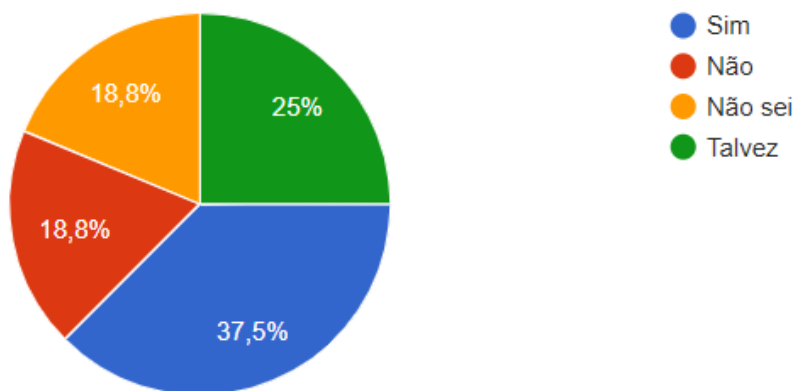


Figura 15. Funcionalidade de Acessibilidade na Prototipação do *Learcod*.

Com os resultados de 75,6% que afirmaram que acham fácil o uso do aplicativo, essa afirmação pode ser analisada pelos discentes serem considerados como os nativos digitais, que são pessoas que nasceram e cresceram familiarizadas com a tecnologia, usando redes sociais, como youtube, facebook, whatsapp, dentre outros. Com isso, a experiência do uso de aplicativos e softwares se torna mais intuitivo e de maneira mais exploratória para os discentes que responderam esse formulário.

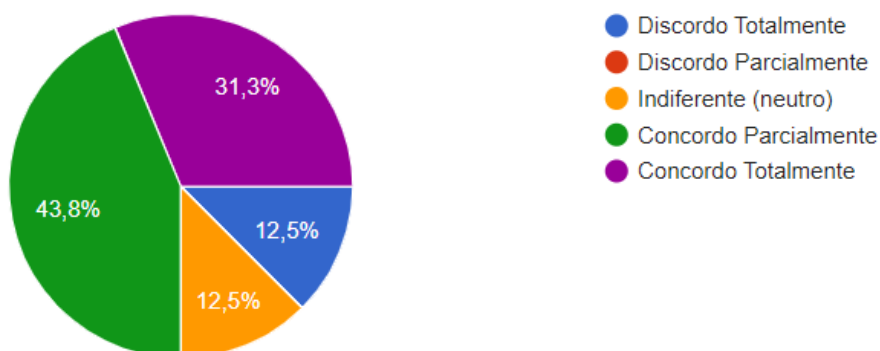


Figura 16. Facilidade de Uso da Prototipação do *Learcod*.

Para os 37,5% que concordaram que precisa aprender monte de coisas antes de usar o aplicativo educativo, essa afirmação pode ser considerada pelo usuário que irá precisar de tutoriais e/ou apoio de professores em sala de aula. Em relação a outra porcentagem dos dados com 37,5% que discorda que precisa aprender muitas coisas antes de usar o aplicativo, se encaixa com o usuário explorador, aquele que é mais intuitivo em suas demandas de usabilidade, não necessariamente precisando de

tutoriais para o uso, com sua própria exploração do aplicativo irá conseguir usar a ferramenta de forma intuitiva.

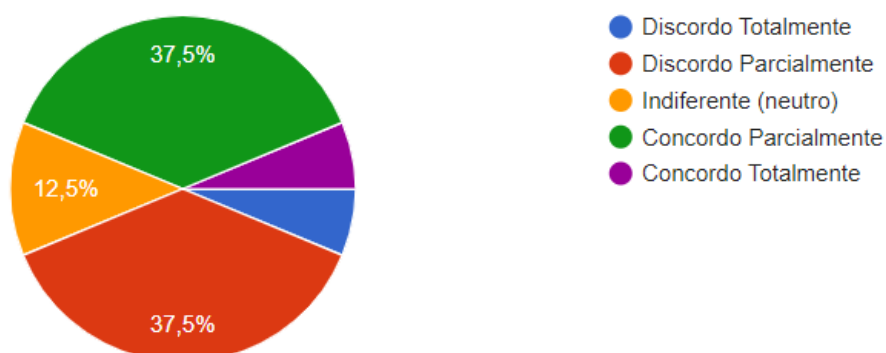


Figura 17. Experiência Prévia para uso da Prototipação do *Learcod*.

Para a compreensão das informações colocadas na prototipagem do aplicativo cerca de 81,3% concordam que foram fáceis de entender, assim percebe-se que, o minimalismo do design está presente no aplicativo, não estando cheio de informações para poluir a tela, apenas as informações básicas que sejam necessárias para o objetivo proposto daquela tela e função específica, pois o minimalismo tem como intuito propor algo adepto daquilo que seja simples e elementar para o usuário, considerando suas funções de acordo que não sobrecarregue muitas informações (Cilumbriello et. al, 2018).

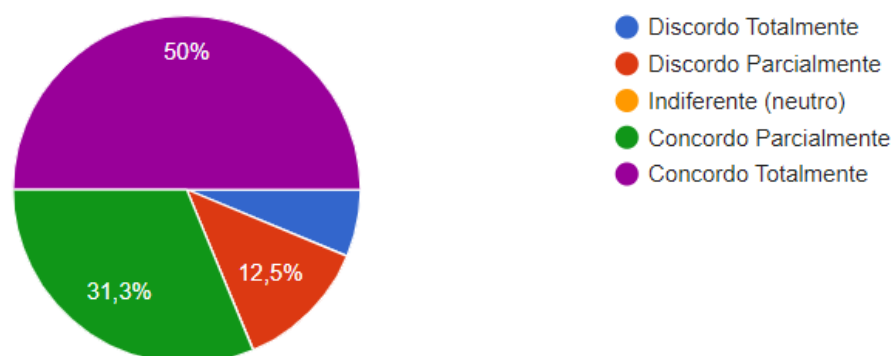


Figura 18. Compreensão da Informação para uso da Prototipação do *Learcod*.

Cerca de 68,8% dos estudantes responderam que concorda que com o aplicativo é possível aprender programação em componentes curriculares, assim pode perceber que o aplicativo pode ser um auxiliar em sala de aula para o ensino e aprendizado de quaisquer linguagens de programação em diversos componentes curriculares dessas áreas, mas que não será o fator e o ponto chave para o aprendizado crucial do componente, ou seja, não seja a primeira ou a única peça para o aprendizado de linguagem de programação no componente curricular, será como

auxiliar nesse caminho facilitando esse processo tanto em sala de aula, como fora para que possa ser feito atividades extraclasse.

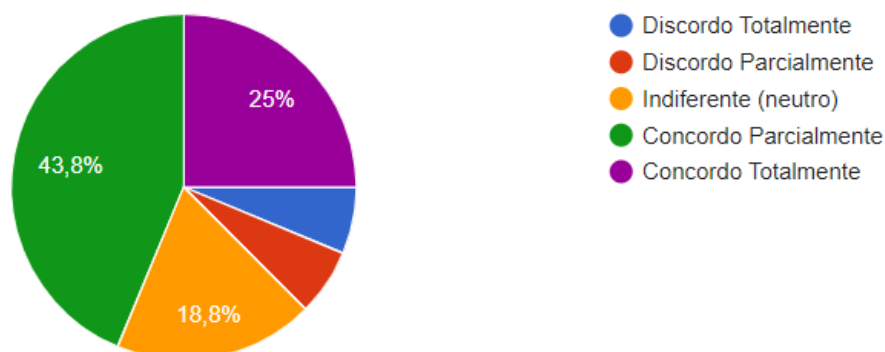


Figura 19. Aprendizado de Linguagens de Programação pelo uso da Prototipação do *Learcod*.

Na discussão dos dados, pode-se analisar outras questões como “Gostaria de usar este aplicativo com frequência.” que 75,1% dos discentes responderam que usariam com frequência, pois acredita-se que, pelos estudantes serem nativos digitais, isso pode facilitar o processo de experiência do usuário com ferramentas de tecnologia e redes sociais.

Na pergunta “Eu gostei de usar a interface do aplicativo, é direto e objetivo em suas funções.” Cerca de 68,8% confirmaram a sentença, assim concluindo que, o minimalismo do aplicativo se sobressai como proposta para que seja simples a abordagem de informações diretas e com funções claras em suas telas.

Na pergunta “Achei o aplicativo complexo no geral.” que contradiz com a pergunta da figura 16, isso pode esclarecer que é preciso mais reajustes na prototipação para que possa ter mais clareza na simulação e posteriormente novas coletas de dados. Na questão “É possível selecionar modos avançados de aprendizado de acordo com a experiência do usuário? (exemplo: os desafios)” com 81,3 % de afirmação pressupõe que conseguiram encontrar as telas de desafios para poder simular essa tarefa e verificar se ela é possível como modo avançado e pelos resultados foi satisfatório. Na questão “O uso do aplicativo se encaixa no ambiente de aprendizado?” com 93,8% de afirmação se obtém conclusão de que é possível se o uso para o aprendizado de forma satisfatória pela coleta de dados e como pela fundamentação teórica que propõe estudos que traz abordagens relevantes para esse parecer. Na questão “Você recomendaria o aplicativo para fins de aprendizado?” 81,3% afirmam que recomendaria para fins de aprendizado, podendo assim entender

que esse aplicativo conseguiu dar interesse e atender demandas de programação e aprendizagem que possam auxiliar no aprendizado e avaliação em sala de aula e fora dela. Na questão “Eu sou capaz de completar ou concluir o meu trabalho, de forma rápida, usando este aplicativo?” Responderam que 75,1% que sim, esse trabalho seria todas as etapas de um processo de aprendizado de uma linguagem, como por exemplo algoritmo, se conseguiu finalizar todas as atividades desse fluxo e não só apenas fez uma e deixou pela metade, assim as atividades pela simulação foram finalizadas, pois caso não concluiu pode também ter sido outros fatores como usabilidade de não ter conseguido mexer na ferramenta ou não ter entendido algo da questão de programação.

Por fim, no geral foi que 81,3% dos estudantes afirmaram que estavam satisfeitos com o aplicativo/prototipação, devida todas as justificativas das questões anteriores, é possível constatar que a fundamentação teórica trouxe abordagens para a estrutura do desenvolvimento da proposta do aplicativo educativo, bem como a abordagem prática da prototipagem, assim com os resultados dos estudantes percebe-se que a usabilidade foi assertiva na prototipagem/aplicativo, tendo em vista ajustes, mas que a ideia e desenvolvimento do mesmo está em conformidade com o perfil de usuário discente.

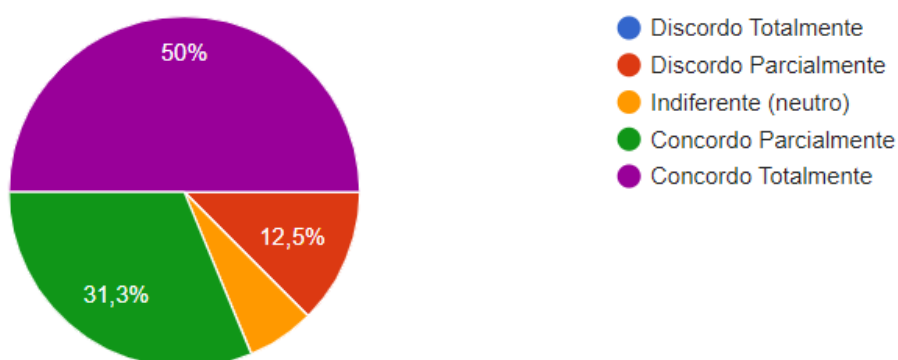


Figura 20. Satisfação Geral no uso da Prototipação do *Learcod*.

Na abordagem do aplicativo *learcod* de etapas de subetapas com minitarefa, a ideia de dividir uma atividade complexa e grande em uma menor, ou seja, uma questão de programação muito grande em miniquestões, minitarefa, assim decompondo elas, para que posteriormente possa compor novamente todo esse processo da questão por completo para que seja compreensível em sala de aula. No aplicativo *learcod* também é perceptível na prototipações e simulação de interação o

que irá acontecer caso a questão esteja correta ou errada, quando a questão estiver correta, irá seguir para a próxima, mas quando errar, irá seguir para a tela de explicação de um conteúdo específico dessa questão e não mostrando qual seria a resposta correta, assim quando rever o assunto, poderá seguir novamente para a questão tendo a oportunidade de refazer, seguindo esse processo com todas as outras questões e tarefas de todo o aplicativo.

Outro contexto que pode abordar é o protagonismo do estudante, em que se torna o atuante principal da sua ação e conhecimentos, no que se refere ao aplicativo é bem visto isso, pois está sempre envolvido com aquilo sendo ele o usuário principal de conhecimentos do aplicativo. Com isso, segundo Dias, Abdalla e Saba (2017), comentam que no âmbito educacional, o protagonismo é mais amplo, pois entende que não se pode trabalhar num campo educacional sem trabalhar em conjunto, assim busca-se por jovens para o seu desenvolvimento social e educacional, em que sua vez irá caracterizar a autonomia no sentido de construir o próprio conhecimento.

5 Considerações Finais

Este trabalho apresenta uma solução de design de aplicativo educativo, denominado *learcod*, a ser utilizada, após a implementação, como ferramenta de auxílio no processo de aprendizado e avaliação em linguagem de programação em componentes curriculares em sala de aula ou fora dela. Foi elaborada uma solução de prototipação de telas do aplicativo *learcod* para o aprendizado e avaliação de programação. Essa solução se mostrou viável por meio do questionário de viabilidade tanto pelos docentes como pelos discentes para obter informações que possam elaborar a arquitetura da solução. As quais demonstraram resultados aceitáveis como em que foi analisado no quadro de levantamento de viabilidade dos dados dos docentes e discentes.

Do ponto de vista da experiência dos usuários discentes e docentes, o aplicativo/prototipagem apresenta uma boa experiência de usabilidade do usuário e requer mais ajustes para esclarecer alguns pontos específicos, como a acessibilidade por exemplo. No entanto, as características do aplicativo trazem grandes interesses para os docentes em seu processo de avaliação e para os discentes no aprendizado, sendo assim possível ser usado com frequência em sala de aula ou fora dela.

Por fim, apesar da solução proposta ser viável, ela está sujeita a novas mudanças para ajustes em suas funções e interfaces para que possa ser mais adaptável ao público utilizador da ferramenta, discente e docente, sendo necessário a adaptação da solução. Além de se avaliar os aspectos de experiência do usuário com discentes e docentes, ampliar um conjunto de usuários com maior diversificação e conhecimento técnico, como analisar os aspectos da acessibilidade visuais por exemplo com público-alvo que seja com pessoas que possuem deficiência visual, como também analisar os dados com uma equipe técnica em design de experiência de usuário e usabilidade.

Partindo da questão de pesquisa, o uso de um *software* educativo contribui no processo de avaliação e aprendizagem processual ativa do discente? Com isso, é possível perceber que com os resultados obtidos a partir da prototipação de telas há uma contribuição para a implementação do aplicativo educativo *learcod* para o ensino

e aprendizado de programação, mesmo que necessite de ajustes para mais esclarecimentos.

5.1 Trabalhos Futuros

Considerando o que foi apresentado são levantados como trabalhos futuros para essa solução os seguintes pontos:

- Acessibilidade: opção de um botão por fala para pessoas com deficiência físico-motora;
- Histórico de todas as ações feita pelo estudante naquele processo de etapa realizado: assim podendo visualizar todos os passos que ele fez e refez;
- Criação de um estilo de aprendizagem: direcionar as atividades mais para o tipo específico de cada estudante (podendo ser utilizado o VARK);
- Utilização de Libras para o melhor entendimento para pessoas surdas: bem como também a tradução para diversos idiomas (principalmente o inglês, como a língua universal);
- Produção de módulos para o aprendizado de Linguagens Web: pois tem o cliente e servidor para codificar;
- Inserir a interdisciplinaridade: colocar módulos que contempla programação com quaisquer conteúdos do dia a dia ou com outros componentes curriculares;
- Uso comum: qualquer pessoa pode usar o aplicativo, não apenas restrito para academia escolar, mas também fora dele;
- Atividade em grupo: usar estratégias diversas, como por exemplo Coding Dojo ou PBL para poder engajar o estudante no processo de criação dos códigos;
- Adaptação da Prototipagem: ajustes na prototipação de acordo com a coleta dos dados obtidos.

Referências Bibliográficas

- ALVES, M. T. G.; FERRÃO, M. E. Uma década da Prova Brasil: evolução do desempenho e da aprovação. **Estudos em Avaliação Educacional**, v. 30, n. 75, p. 688-720, 2019.
- BERGMANN, J.; SAMS, A. Sala de Aula Invertida - Uma Metodologia Ativa de Aprendizagem. 1. ed. Rio de Janeiro: LTC, 2018.
- BORSZCZ, D. C.; BORTOLINI, T.; MARQUES, K. F.; SANTOS, G. N.; RIBEIRO, H. A. LIBRAS PLAY: Um protótipo de aplicativo voltado para a tradução de Libras no ambiente escolar. **Anais da Feira de Ensino, Pesquisa e Extensão do Instituto Federal Catarinense Campus Fraiburgo**, v. 1, n. 1, 2021.
- BRITTO, R.; FILHO, W. G. O., BARROS, C. G., LOPES, E. C. Intelligent tutor system model applied to basic electronics. **12th Iberian Conference on Information Systems and Technologies (CISTI)**, Lisbon, p. 1-5, 2017, DOI: <https://doi.org/10.23919/CISTI.2017.7975832>.
- CARDOSO, I. C. B. O.; VILA, A. J. D. M.; SILVA, J. J. B.; MELO, J. S. Blindaid Market: Proposta De Aplicativo Que Auxilia Nas Compras De Pessoas Cegas. **Deficientes Visuais E Analfabetas**. Sobre Tudo, v. 12, n. 2, p. 299-331, 2021.
- CILUMBRIELLO, N. P. S.; MARTINS, V. F.; ELISEO, M. A.; KAWAMOTO, A. L. S. Avaliação heurística e teste de usabilidade para software de design de interiores. **Revista Ibérica de Sistemas e Tecnologias de Informação**, n. 17, p. 90-101, 2019. Disponível em: <https://search.proquest.com/openview/5720c78f2e17a2732a91b614ef77e567/1?pq-origsite=gscholar&cbl=1006393>.
- DEBALD, B. Metodologias Ativas no Ensino Superior: o protagonismo do aluno. 1. ed. Porto Alegre: Penso, 2020.
- Dias, J.; Abdalla, D.; Saba, H. Clube de Robótica: autonomia e protagonismo juvenil por meio de atividade complementar na escola. In: **Anais do XXIII Workshop de Informática na Escola (WIE)**, 2017, DOI: <http://dx.doi.org/10.5753/cbie.wie.2017.875>.
- FEITOR, C. D. C.; SILVA, R. B. F.; SOUZA, I. I. L. A Pesquisa-ação Como Estratégia Metodológica para o Desenvolvimento da Abordagem Seis Sigma. **Revista Interface-UFRN/CCSA**, v. 10, n. 1, p. 75-102, 2013.
- FERNÁNDEZ, G. E.; CALLEGARI, M. V. Estratégias Motivacionais para Aulas de Língua Estrangeira. 1. ed. São Paulo: Companhia Editora Nacional, 2010.
- FERREIRA, D.H. L.; BRANCHI, A.; SUGAHARA, C. R. Processo de ensino e aprendizagem no contexto das aulas e atividades remotas no Ensino Superior em tempo da pandemia Covid-19. **Revista Práxis**, v. 12, n. 1, p. 19-28, 2020.
- FILARDI, A. L.; TRAINA, A. J. M. Montando questionários para medir a satisfação do usuário: avaliação de Interface de um sistema que utiliza técnicas de recuperação de

imagens por conteúdo. In: **VIII Brazilian Symposium on Human Factors in Computing Systems**. Porto Alegre (RS): IHC'08, p. 176-185, 2008.

FRANZEN, E.; HEMMING, C.; BERCHT, M. Sistema de apoio ao uso da problematização no ensino e aprendizagem de programação. **Revista Destaques Acadêmicos**, v. 10, n. 4, 2018.

GIRAFFA, L.M.; MORA, M.C. Evasão na disciplina de algoritmo e programação: um estudo a partir dos fatores intervenientes na perspectiva do aluno. In: **III Conferencia sobre el Abandono en la Educación Superior. Anais. Cidade do México: Universidad Nacional Autónoma de México**, 2013.

GONSALES, P. Design Thinking e a ritualização de boas práticas educativas. São Paulo: Instituto Educadigital, 2018.

GOMES, M. E. N.; D'EMERY, R. A.; FILHO, G. A. A. C. AAPW: Uma ferramenta para facilitar o aprendizado de programação Web. **Anais do V Congresso sobre Tecnologias na Educação (CSBC)**, p. 1555-1564, 2020.

GOMES, T. C. S.; MELO, J. C. B. App Inventor for Android: Uma Proposta Construcionista para Experiências Significativas de Aprendizagem no Ensino de Programação. **4º Simpósio Hipertexto e Tecnologias na Educação: UFPE**, 2012.

JESUS, G. S.; SANTOS, K.; CONCEIÇÃO, J.; NETO, A. C. Análise dos erros mais comuns de aprendizes de programação que utilizam a linguagem Python. **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)**, v. 29, n. 1, p. 1751, 2018, DOI: <https://doi.org/10.5753/cbie.sbie.2018.1751>.

KAMII, C. A criança e o número: Implicações Educacionais da Teoria de Piaget. 10. ed. Campinas: Papirus, 1989.

KOGURE, S.; OGASAWARA, K.; K.; NOGUCHI, Y.; KONISHI, T.; ITOH, Y. Application of Programming Learning Support System to Object Oriented Language. **Proceedings of the 27th International Conference on Computers in Education. Taiwan: Asia-Pacific Society for Computers in Education**, 2019.

KRZYZANOWSKI, L.; BELETI, C. JR., SANTIAGO, R. JR., TOSTES, R. A. Ensino de programação: um estudo preliminar nos cursos de licenciatura em Computação no Brasil. In **Anais dos Workshops do Congresso Brasileiro de Informática na Educação**, v. 8, n. 1, p. 21, 2019.

LUCENA, S. V.; KEMCZINSKI, A.; GASPARINI, I.; MATOS, V. A.; OGAWA, N. A. Modelagem de requisitos baseada em cenários para o Storyboard da Metodologia para Construção de Objetos de Aprendizagem Interativos. **Nuevas Ideas en Informática Educativa TISE**, 2014.

MENDES, L. O. R.; BUENO, A. J. A.; DESSBESEL, R. S.; SILVA, S. C. R. Gamificação no Processo de Ensino e Aprendizagem de Estudantes Surdos: uma revisão sistemática. **Renote**, Porto Alegre (RS), v. 17, n. 3, p. 142-151, 2019.

- MITTAL, H.; MANDALIKA, S. D. Framework for Evaluation of Programming Language Examinations. **SpringerBriefs in Applied Sciences and Technology**, p. 121–128, 2015, DOI: https://doi.org/10.1007/978-981-287-338-5_10
- MOURA, M. P. C.; NUNES, R. P.; MOCBEL, M. Â. R.; SOUZA, F. F. Protótipo de Aplicativo Educativo para o ensino de POO: Avaliação da Usabilidade e Experiência do Usuário. **Renote**, v. 17, n. 3, p. 305–314, 2019.
- NASCIMENTO, L. F.; CAVALCANTE, M. M. D. Abordagem quantitativa na pesquisa em educação: investigações no cotidiano escolar. **Revista Tempos e Espaços em Educação**, v. 11, n. 25, p. 249–260, 2018. Disponível em: <https://seer.ufs.br/index.php/revtee/article/view/7075>, DOI: <https://doi.org/10.20952/revtee.v11i25.7075>
- OAKLEY, B. Aprendendo a aprender para crianças e adolescentes: como se dar bem na escola. 1. ed. Rio de Janeiro: BestSeller, 2019.
- OLIVEIRA, M.; LEITE, A.; SILVA, M.; BODART, C. M.; NASCIMENTO, G. A História da Condessa Surda de Lovelace: Um Relato de Experiência de Ensino Híbrido e Assistivo de Programação. Anais do XIII Women In Information Technology (WIT), p. 51-60, 2019, DOI: <https://doi.org/10.5753/wit.2019.6712>.
- PERRENOUD, P. Avaliação: Da Excelência à Regulação das Aprendizagens Entre Duas Lógicas. 1. ed. Porto Alegre: Artmed, 1999.
- PINTO, A. V. **O Conceito de Tecnologia**. 2v. ed. Rio de Janeiro: Contraponto, 2005.
- RIBEIRO, L. R. C. Aprendizagem baseada em problemas (PBL): uma experiência no ensino superior. **SciELO – EdUFSCar**, 2008, DOI: <https://doi.org/10.7476/9788576002970>
- ROCHA, R. G. C.; VANDERLEI, I. M.; BION, D.; ARAÚJO, J. C. T. Uma abordagem prática de design thinking no processo de ensino-aprendizagem na engenharia de software. **Renote**, v. 19, n. 2, p. 447-456, 2021.
- RODRIGUES, J. A.; CORDEIRO, N. J. N. O Ensino, em especial o de Matemática, na Austrália. **Anais da V Semana da Matemática da Universidade Estadual Vale do Acaraú, De Matemático a Professor de Matemática: Desafios e Perspectivas**, p. 90, 2013.
- SILVA, W. S.; LIMA, M. S.; RAPOSO, J. C. S., JÚNIOR, L. C. F. S. Levantamento sobre as dificuldades dos discentes nas disciplinas de programação no curso técnico de informática. **Diversitas Journal**, v. 3, n. 3, p. 761–770, 2018.
- TAVARES, J. L.; SILVA, L. T. G. Tipos e Classificações de Softwares Educacionais. **Anais IV Conedu – Congresso Nacional da Educação**, 2017.
- TUMELERO, N. Pesquisa aplicada: material completo, com exemplos e características. Mettzer, 2019. Disponível em: <https://blog.mettzer.com/pesquisa-aplicada/>. Acesso em: 15 de junho de 2022.

VIANA, G.; LOPES, A.; PORTELA, C.; OLIVEIRA, S. Um Survey sobre a Aprendizagem de Programação no Curso de Sistemas de Informação. **Em: Workshop Sobre Educação em Computação (WEI)**, Belém, p. 161-175, 2019.

VICKERY, A. Aprendizagem ativa nos anos iniciais do ensino fundamental. 1. ed. Porto Alegre: Penso, 2016.

VILLA, A. V; DELGADO, R. C. Formación de estudiantes de Medicina como tutores pares en aprendizaje basado en problemas. **Educación Médica Superior**, v. 33, n. 3, 2019.

WING, J. Computational Thinking. **Communications of the ACM**, v. 49, n. 3, p. 33–35, 2006.

Apêndice A – Questionário do Levantamento sobre Viabilidade Tecnológica para os Discentes

Esse levantamento sobre viabilidade tecnológica compõe parte da monografia da estudante Rayana Ribeiro Bonfanti do curso de Licenciatura em Computação do IFBA Campus Porto Seguro. O estudo do trabalho “Aplicação do *Learcod*” tem como objetivo desenvolver um design de um software educativo para o aprendizado e avaliação processual no ensino de linguagem de programação. Esse questionário irá colaborar no planejamento inicial do aplicativo com a coleta de dados para a abertura do projeto.

Dúvidas, entrar em contato com Rayana: E-mail: rayanabonfanti@gmail.com

Aceita participar da coleta de dados dessa pesquisa?*

- ☐ Sim
- ☐ Não
- ☐ Não sei

Explicação do Projeto de Pesquisa da Monografia

Em cursos da área de Computação, seja o Bacharelado ou Licenciatura, há um alto índice de reprovação nos componentes de Programação, podendo ser justificado pela qualidade de ensino precário (Franzen, Hemming e Bercht, 2018). Essa evasão dos estudantes nos componentes de Programação persiste também em outras áreas afins de exatas, como as Engenharias e Matemática. Assim, refletir sobre novas metodologias, formas de ensino e avaliação para Programação é necessário no momento atual. Ademais, o uso das Novas Tecnologias Digitais de Informação e Comunicação (NTDICs) são importantes no ambiente escolar, principalmente nessa situação atual da pandemia, em que o ensino remoto é presente. Com isso, essa monografia tem como objetivo desenvolver um design de um software educativo para o aprendizado e avaliação processual no ensino de linguagem de Programação. Esse software educativo é um aplicativo chamado *Learcod* para o aprendizado e avaliação processual no ensino de linguagem de Programação. Esse aplicativo utiliza de abordagens, como: metodologias ativas, acessibilidade, técnicas de estudos, formas de avaliação, conceitos e/ou exercícios de Programação/Computação, além de englobar assuntos de lógica matemática, raciocínio lógico, interpretação textual, análise semântica, entre outros. Assim, tanto estudante, como professor(a) poderá acessar o aplicativo como forma de aprendizado e avaliação, respectivamente, além de usuários comuns curiosos em aprender Ciência da Computação. Diante disso, esse questionário irá colaborar no desenvolvimento do aplicativo e aplicação de Engenharia de Software para elaboração inicial do projeto, assim construindo de forma participante o projeto com os dados coletados desse questionário, em que se espera uma ação participante dos estudantes e professores nesse processo de desenvolvimento do projeto inicial do aplicativo como forma de auxiliar futuramente ambos em seu aprendizado e avaliação.

Qual seu vínculo com IFBA - Campus Porto Seguro?*

- ☐ Docente [Professor(a)]
- ☐ Discente [Estudante]
- ☐ Não sou do Campus [Sou Docente]
- ☐ Não sou do Campus [Sou Discente]
- ☐ Não sou do Campus [Outro]

Questionário para Estudante

Perguntas restrito para estudantes, ou seja, são questionamentos relacionados a aprendizagem.

Esse software educativo é um aplicativo chamado *Learcod* para o aprendizado e avaliação processual no ensino de linguagem de programação. Esse aplicativo utiliza de abordagens, como: metodologias ativas, acessibilidade, técnicas de estudos, formas de avaliação, conceitos e/ou

exercícios de Programação/Computação, além de englobar assuntos de lógica matemática, raciocínio lógico, interpretação textual e análise semântica, entre outros. Assim, tanto estudante, como professor(a) poderá acessar o aplicativo como forma de aprendizado e avaliação, respectivamente, além de usuários comuns curiosos em aprender Ciência da Computação.

O uso desse aplicativo seria viável no aprendizado em componentes de programação?*

- ☐ Sim
- ☐ Não
- ☐ Não sei

Você tem experiência em usar as Novas Tecnologias Digitais de Informação e Comunicação (NTDICs)?*

Exemplos de NTDIC's: Slides, Vídeo Aulas, Quiz, Wiki, Juízes Online de Programação, Podcasts, Youtube, Netflix, Softwares, Programas, Arduino, Roteador, etc.

- ☐ Sim
- ☐ Não
- ☐ Não sei

Com que frequência você utiliza Plataforma de Ensino?*

Exemplos: Edmodo, Suap, Moodle, Sistema de Matrícula, entre outros.

- ☐ 1 à 2 vezes na semana
- ☐ Até 5 vezes na semana
- ☐ Todos os dias na semana
- ☐ Outro
- ☐ Nenhum

É interessante que nesse aplicativo seja feita a correção da resolução dos exercícios das atividades dadas pelo professor(a)?*

- ☐ Sim
- ☐ Não
- ☐ Não sei

Você já utilizou algum software/aplicativo para aprender programação?*

- ☐ Sim
- ☐ Não
- ☐ Não sei

O quanto acha que conseguiria aprender com esse aplicativo?*

- ☐ Nada
- ☐ Pouco
- ☐ Muito

Prefere fazer os exercícios pelo aplicativo de qual melhor forma?*

- ☐ Individualmente
- ☐ Em grupo
- ☐ Ambos
- ☐ Não sei

Consigo aprender a manusear um novo software/aplicativo.*
(exemplo como esse aplicativo)

- ☐ Discordo totalmente
- ☐ Discordo parcialmente
- ☐ Não concordo, nem discordo
- ☐ Concordo parcialmente
- ☐ Concordo totalmente

O uso de um aplicativo me motivaria a estudar mais programação.*

- ☐ Discordo totalmente
- ☐ Discordo parcialmente
- ☐ Não concordo, nem discordo
- ☐ Concordo parcialmente
- ☐ Concordo totalmente

Acha válido a autoavaliação nesse aplicativo? *

- ☐ Sim
- ☐ Não
- ☐ Não sei

Gostaria que o controle de frequência também fosse aplicado no aplicativo? *

- ☐ Sim
- ☐ Não
- ☐ Não sei

As Novas Tecnologias Digitais de Informação e Comunicação (NTDIC's) são viáveis a ser utilizada para as avaliações. *

Exemplos de NTDIC's: Slides, Vídeo Aulas, Quiz, Wiki, Juízes Online de Programação, Podcasts, Youtube, Netflix, Softwares, Programas, Arduino, Roteador, etc.

- ☐ Discordo totalmente
- ☐ Discordo parcialmente
- ☐ Não concordo, nem discordo
- ☐ Concordo parcialmente
- ☐ Concordo totalmente

Há proveito do uso das Novas Tecnologias Digitais de Informação e Comunicação (NTDIC's) em avaliações. *

Exemplos de NTDIC's: Slides, Vídeo Aulas, Quiz, Wiki, Juízes Online de Programação, Podcasts, Youtube, Netflix, Softwares, Programas, Arduino, Roteador, etc.

- ☐ Discordo totalmente
- ☐ Discordo parcialmente
- ☐ Não concordo, nem discordo
- ☐ Concordo parcialmente
- ☐ Concordo totalmente

Identificação

Nessa seção há dados opcionais e obrigatórios. Com isso, é possível tanto colocar seus dados nos campos opcionais ou não, sendo de qualquer forma mantida em sigilo para a coleta de dados.

Observação: E-mail e/ou Telefone é interessante ser preenchido para possível entrevista posteriormente a esse questionário, para maiores detalhes da pesquisa.

Qual seu curso? *

- ☐ Licenciatura em Computação
- ☐ Técnico em Informática
- ☐ Outro
- ☐ Nenhum

Qual situação se encontra em quaisquer componentes de programação? *

Componentes de Programação podem ser como: Algoritmo, Linguagem de Programação, Programação Orientada a Objetos 1, Programação Orientada a Objetos 2 e Web; Lógica de Programação 1 ou 2.

- ☐ Ainda estou cursando programação
- ☐ Ainda não tive nenhum componente de programação
- ☐ Não estou cursando nenhum componente de programação
- ☐ Nenhuma das Alternativas acima

Qual situação se encontra em quaisquer componentes de programação? *

Componentes de Programação podem ser como: Algoritmo, Linguagem de Programação, Programação Orientada a Objetos 1, Programação Orientada a Objetos 2 e Web; Lógica de Programação 1 ou 2.

- ☐ Reprovi uma vez em programação
- ☐ Reprovi duas vezes em programação
- ☐ Reprovi mais de duas vezes em programação
- ☐ Fui aprovado(a) em programação e nunca reprovi

- ☐ Fui aprovado(a), mas já reprovei uma vez em programação
- ☐ Fui aprovado(a), mas já reprovei duas vezes em programação
- ☐ Fui aprovado(a), mas já reprovei mais de duas vezes em programação
- ☐ Nenhuma das Alternativas acima

Você teve contato/experiência em programação antes de ingressar no curso atual em que está?*

- ☐ Sim
- ☐ Não
- ☐ Não Lembro

Qual sua faixa etária?*

- ☐ Até 18 anos
- ☐ Entre 19 e 21 anos
- ☐ Entre 22 e 25 anos
- ☐ Entre 26 e 30 anos
- ☐ Entre 30 e 40 anos
- ☐ Entre 40 e 60 anos
- ☐ Acima de 60 anos

[Opcional] Digite seu nome

[Opcional] Digite seu e-mail

[Opcional] Digite seu telefone

Whatsapp

[Opcional] Informe seu sexo

ao qual se identifica

- ☐ Feminino
- ☐ Masculino
- ☐ Prefiro não dizer

[Opcional] Qual a quantidade de horas tem disponível por semana para atividades extras?

- ☐ Menos de 1 hora
- ☐ Entre 1 e 2 horas
- ☐ Entre 3 e 4 horas
- ☐ 5 horas
- ☐ Mais que 5 horas

[Opcional] Qual sua atividade profissional?

- ☐ Carteira Assinada (CLT)
- ☐ Empreendedor(a)
- ☐ Concursado(a)
- ☐ Apenas Estudante
- ☐ Outro
- ☐ Nenhum

[Opcional] Qual sua escolaridade?

- ☐ Ensino Fundamental II Incompleto
- ☐ Ensino Fundamental II Completo
- ☐ Ensino Médio Incompleto
- ☐ Ensino Médio Completo
- ☐ Ensino Médio com Curso Técnico Incompleto
- ☐ Ensino Médio com Curso Técnico Completo
- ☐ Ensino Superior Incompleto
- ☐ Ensino Superior Completo
- ☐ Pós-graduação Incompleto
- ☐ Pós-graduação Completo
- ☐ Outro

☐ Nenhum

Habilidades

Questões relacionadas ao estado físico e emocional, pois a depender do seu estado, as respostas podem gerar resultados controversias. Além de identificar algumas habilidades como curiosidade, entre outros, que são importantes para essa pesquisa.

Antes de começar esse questionário estava bem emocionalmente?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

Antes de começar esse questionário estava com alguma dor física?

Por exemplo dor de cabeça, cólica, estomacal, etc.

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

Normalmente planeja e faz suas atividades com antecedência?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

É curioso(a)?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

É criativo(a) ou grande imaginação?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

Há um grau de irritabilidade?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

Apêndice B – Questionário do Levantamento sobre Viabilidade Tecnológica para os Docentes

Esse levantamento sobre viabilidade tecnológica compõe parte da monografia da estudante Rayana Ribeiro Bonfanti do curso de Licenciatura em Computação do IFBA Campus Porto Seguro. O estudo do trabalho “Aplicação do *Learcod*” tem como objetivo desenvolver um design de um software educativo para o aprendizado e avaliação processual no ensino de linguagem de programação. Esse questionário irá colaborar no planejamento inicial do aplicativo com a coleta de dados para a abertura do projeto.

Dúvidas, entrar em contato com Rayana: E-mail: rayanabonfanti@gmail.com

Aceita participar da coleta de dados dessa pesquisa? *

- ☐ Sim
- ☐ Não
- ☐ Não sei

Explicação do Projeto de Pesquisa da Monografia

Em cursos da área de Computação, seja o Bacharelado ou Licenciatura, há um alto índice de reprovação nos componentes de Programação, podendo ser justificado pela qualidade de ensino precário (Franzen, Hemming e Bercht, 2018). Essa evasão dos estudantes nos componentes de Programação persiste também em outras áreas afins de exatas, como as Engenharias e Matemática. Assim, refletir sobre novas metodologias, formas de ensino e avaliação para Programação é necessário no momento atual. Ademais, o uso das Novas Tecnologias Digitais de Informação e Comunicação (NTDICS) são importantes no ambiente escolar, principalmente nessa situação atual da pandemia, em que o ensino remoto é presente. Com isso, essa monografia tem como objetivo desenvolver um design de um software educativo para o aprendizado e avaliação processual no ensino de linguagem de Programação. Esse software educativo é um aplicativo chamado *Learcod* para o aprendizado e avaliação processual no ensino de linguagem de Programação. Esse aplicativo utiliza de abordagens, como: metodologias ativas, acessibilidade, técnicas de estudos, formas de avaliação, conceitos e/ou exercícios de Programação/Computação, além de englobar assuntos de lógica matemática, raciocínio lógico, interpretação textual, análise semântica, entre outros. Assim, tanto estudante, como professor(a) poderá acessar o aplicativo como forma de aprendizado e avaliação, respectivamente, além de usuários comuns curiosos em aprender Ciência da Computação. Diante disso, esse questionário irá colaborar no desenvolvimento do aplicativo e aplicação de Engenharia de Software para elaboração inicial do projeto, assim construindo de forma participante o projeto com os dados coletados desse questionário, em que se espera uma ação participante dos estudantes e professores nesse processo de desenvolvimento do projeto inicial do aplicativo como forma de auxiliar futuramente ambos em seu aprendizado e avaliação.

Qual seu vínculo com IFBA - Campus Porto Seguro? *

- ☐ Docente [Professor(a)]
- ☐ Discente [Estudante]
- ☐ Não sou do Campus [Sou Docente]
- ☐ Não sou do Campus [Sou Discente]
- ☐ Não sou do Campus [Outro]

Questionário para Docente

Perguntas restrito para professores, ou seja, são questionamentos relacionados a avaliação.

Esse software educativo é um aplicativo chamado *Learcod* para o aprendizado e avaliação processual no ensino de linguagem de programação. Esse aplicativo utiliza de abordagens, como:

metodologias ativas, acessibilidade, técnicas de estudos, formas de avaliação, conceitos e/ou exercícios de Programação/Computação, além de englobar assuntos de lógica matemática, raciocínio lógico, interpretação textual e análise semântica, entre outros. Assim, tanto estudante, como professor(a) poderá acessar o aplicativo como forma de aprendizado e avaliação, respectivamente, além de usuários comuns curiosos em aprender Ciência da Computação.

O uso desse aplicativo seria viável para avaliação de componentes de programação?*

- ☐ Sim
- ☐ Não
- ☐ Não sei

É interessante acompanhar o estudante em lista de exercícios de forma individual e/ou mais ágil?*

- ☐ Sim
- ☐ Não
- ☐ Não sei

Você tem experiência em usar as Novas Tecnologias Digitais de Informação e Comunicação

(NTDICs)?*

Exemplos de NTDIC's: Slides, Vídeo Aulas, Quiz, Wiki, Juízes Online de Programação, Podcasts, Youtube, Netflix, Softwares, Programas, Arduino, Roteador, etc.

- ☐ Sim
- ☐ Não
- ☐ Não sei

Com que frequência você utiliza Plataformas de Ensino?*

Exemplo: Edmodo, Suap, Moodle, Sistema de Matrícula, entre outros.

- ☐ 1 à 2 vezes na semana
- ☐ Até 5 vezes na semana
- ☐ Todos os dias da semana
- ☐ Outro
- ☐ Nenhum

Você já instalou ou desinstalou um programa/aplicativo/software em algum computador ou celular?*

- ☐ Sim
- ☐ Não
- ☐ Não sei

Há proveito do uso das Novas Tecnologias Digitais de Informação e Comunicação (NTDIC's) em avaliações.*

Exemplos de NTDIC's: Slides, Vídeo Aulas, Quiz, Wiki, Juízes Online de Programação, Podcasts, Youtube, Netflix, Softwares, Programas, Arduino, Roteador, etc.

- ☐ Discordo totalmente
- ☐ Discordo parcialmente
- ☐ Não concordo, nem discordo
- ☐ Concordo parcialmente
- ☐ Concordo totalmente

Você já utilizou algum software/aplicativo para avaliação de atividades de programação?*

- ☐ Sim
- ☐ Não
- ☐ Não sei

As Novas Tecnologias Digitais de Informação e Comunicação (NTDIC's) são viáveis a ser utilizada para os estudos de programação.*

Exemplos de NTDIC's: Slides, Vídeo Aulas, Quiz, Wiki, Juízes Online de Programação, Podcasts, Youtube, Netflix, Softwares, Programas, Arduino, Roteador, etc.

- ☐ Discordo totalmente
- ☐ Discordo parcialmente
- ☐ Não concordo, nem discordo
- ☐ Concordo parcialmente
- ☐ Concordo totalmente

Se sentiria motivado(a) a usar plataformas voltadas para avaliação de atividades de programação? *

- ☐ Sim
- ☐ Não
- ☐ Não sei

As Novas Tecnologias Digitais de Informação e Comunicação (NTDIC's) são viáveis a ser utilizada para as avaliações.*

Exemplos de NTDIC's: Slides, Vídeo Aulas, Quiz, Wiki, Juízes Online de Programação, Podcasts, Youtube, Netflix, Softwares, Programas, Arduino, Roteador, etc.

- ☐ Discordo totalmente
- ☐ Discordo parcialmente
- ☐ Não concordo, nem discordo
- ☐ Concordo parcialmente
- ☐ Concordo totalmente

Conhece algum outro aplicativo/software com recursos similares desse aplicativo? Comente.

Consigo aprender a manusear um novo software/aplicativo.*

Como esse aplicativo

- ☐ Discordo totalmente
- ☐ Discordo parcialmente
- ☐ Não concordo, nem discordo
- ☐ Concordo parcialmente
- ☐ Concordo totalmente

Identificação Docente

Nessa seção há dados opcionais e obrigatórios. Com isso, é possível tanto colocar seus dados nos campos opcionais ou não, sendo de qualquer forma mantida em sigilo para a coleta de dados.

Observação: E-mail e/ou Telefone é interessante ser preenchido para possível entrevista posteriormente a esse questionário, para maiores detalhes da pesquisa.

Qual sua faixa etária? *

- ☐ Até 18 anos
- ☐ Entre 19 e 21 anos
- ☐ Entre 22 e 25 anos
- ☐ Entre 26 e 30 anos
- ☐ Entre 30 e 40 anos
- ☐ Entre 40 e 60 anos
- ☐ Acima de 60 anos

Qual sua área? *

- ☐ Computação
- ☐ Educação
- ☐ Outro

[Opcional] Digite seu nome

[Opcional] Digite seu e-mail

[Opcional] Digite seu telefone
Whatsapp

[Opcional] Informe seu sexo
ao qual se identifica

- ☐ Feminino

- ☐ Masculino
- ☐ Prefiro não dizer

[Opcional] Qual a quantidade de horas tem disponível por semana para atividades extras?

- ☐ Menos de 1 hora
- ☐ Entre 1 e 2 horas
- ☐ Entre 3 e 4 horas
- ☐ 5 horas
- ☐ Mais que 5 horas

Habilidades

Questões relacionadas ao estado físico e emocional, pois a depender do seu estado, as respostas podem gerar resultados controversas. Além de identificar algumas habilidades como curiosidade, entre outros, que são importantes para essa pesquisa.

Antes de começar esse questionário estava bem emocionalmente?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

Antes de começar esse questionário estava com alguma dor física?

Por exemplo dor de cabeça, cólica, estomacal, etc.

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

Normalmente planeja e faz suas atividades com antecedência?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

É curioso(a)?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

É criativo(a) ou grande imaginação?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

Há um grau de irritabilidade?

- ☐ Sim
- ☐ Não
- ☐ Talvez
- ☐ Não sei

Apêndice C – Questionário de Usabilidade da Prototipagem do Aplicativo *Learcod* para Discente

Esse questionário sobre usabilidade tecnológica compõe parte da monografia da estudante Rayana Ribeiro Bonfanti do curso de Licenciatura em Computação do IFBA Campus Porto Seguro. O estudo do trabalho “Aplicação do *Learcod*” tem como objetivo desenvolver um design de um software educativo para o aprendizado e avaliação processual no ensino de linguagem de programação. Esse questionário irá colaborar com a coleta de dados para o procedimento da escrita final da monografia.

Dúvidas, entrar em contato com Rayana: E-mail: rayanabonfanti@gmail.com

Link de acesso ao vídeo simulador da prototipação: <https://bit.ly/videoapplearcodprofessores>

Link de acesso ao simulador da prototipação: <http://ifbaprojetos.com/learcod>

Termo de Aceitação

Abaixo será aceito a coleta de dados para a monografia para análise, em que todo momento manteremos o sigilo de todo integrante que responderá.

Concorda em participar da pesquisa e usar os dados para análise da monografia? *

- ☐ Sim
- ☐ Não

Usabilidade Tecnológica

Com o intuito de iniciar com o processo de prototipação de telas antes da implementação de códigos para os testes de usabilidades, esse questionário tem como objetivo, colaborar na construção de forma participante no projeto da monografia do aplicativo com os dados coletados desse questionário, que será repensado e reformulado, discutido para ser analisado e descrito todos os passos que possam ser alterados ou inseridos.

O aplicativo é adaptável para as necessidades de pessoas com deficiência? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

Gostaria de usar este aplicativo com frequência. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Achei o aplicativo fácil de usar. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Eu preciso aprender um monte de coisas antes de continuar usando este aplicativo. *

- ☐ Discordo Totalmente

- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Eu gostei de usar a interface do aplicativo, é direto e objetivo em suas funções. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

A organização de informações na tela do aplicativo é clara. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

As informações fornecidas pelo aplicativo são fáceis de compreender. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Achei o aplicativo complexo no geral. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Aspectos Pedagógicos (Usabilidade)

Para além da tecnologia, é preciso verificar a relação pedagógica, com isso, foi preciso algumas perguntas para verificar o contexto do ensino e aprendizado do estudante em relação a sua aprendizagem no componente curricular que estiver cursando. Assim, esse questionário tem como objetivo, colaborar na construção de forma participante no projeto da monografia do aplicativo com os dados coletados desse questionário, que será repensado e reformulado, discutido para ser analisado e descrito todos os passos que possam ser alterados ou inseridos.

É possível selecionar modos avançados de aprendizado de acordo com a experiência do usuário?

(exemplo: os desafios) *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

O uso do aplicativo se encaixa no ambiente de aprendizado? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

O aplicativo complementa outras atividades do componente relacionado a área de programação? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

É possível que os usuários preparem uma apresentação usando esse aplicativo? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

Você atinge seus objetivos de aprendizado com o aplicativo? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

Você recomendaria o aplicativo para fins de aprendizado? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

Posso aprender programação no componente curricular com este aplicativo? *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Eu sou capaz de completar ou concluir o meu trabalho, de forma rápida, usando este aplicativo? *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Este aplicativo tem todas as funções e recursos de ensino e aprendizagem que eu espero que ele tenha. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

No geral, estou satisfeito com este aplicativo. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Apêndice D – Questionário de Usabilidade da Prototipagem do Aplicativo *Learcod* para Docente

Esse questionário sobre usabilidade tecnológica compõe parte da monografia da estudante Rayana Ribeiro Bonfanti do curso de Licenciatura em Computação do IFBA Campus Porto Seguro. O estudo do trabalho “Aplicação do *Learcod*” tem como objetivo desenvolver um design de um software educativo para o aprendizado e avaliação processual no ensino de linguagem de programação. Esse questionário irá colaborar com a coleta de dados para o procedimento da escrita final da monografia.

Dúvidas, entrar em contato com Rayana: E-mail: rayanabonfanti@gmail.com

Link de acesso ao vídeo simulador da prototipação: <https://bit.ly/videoapplearcodestudante>

Link de acesso ao simulador da prototipação: <http://ifbaprojetos.com/learcod>

Termo de Aceitação

Abaixo será aceito a coleta de dados para a monografia para análise, em que todo momento manteremos o sigilo de todo integrante que responderá.

Concorda em participar da pesquisa e usar os dados para análise da monografia? *

☐ Não

☐ Sim

Usabilidade Tecnológica

Com o intuito de iniciar com o processo de prototipação de telas antes da implementação de códigos para os testes de usabilidades, esse questionário tem como objetivo, colaborar na construção de forma participante no projeto da monografia do aplicativo com os dados coletados desse questionário, que será repensado e reformulado, discutido para ser analisado e descrito todos os passos que possam ser alterados ou inseridos.

O aplicativo é adaptável para as necessidades de pessoas com deficiência? *

Sim

Não

Não sei

Talvez

Outro: _____

Gostaria de usar este aplicativo com frequência. *

☐ Discordo Totalmente

☐ Discordo Parcialmente

☐ Indiferente (neutro)

☐ Concordo Parcialmente

☐ Concordo Totalmente

Achei o aplicativo fácil de usar. *

☐ Discordo Totalmente

☐ Discordo Parcialmente

☐ Indiferente (neutro)

☐ Concordo Parcialmente

☐ Concordo Totalmente

Eu preciso aprender um monte de coisas antes de continuar usando este aplicativo. *

☐ Discordo Totalmente

☐ Discordo Parcialmente

☐ Indiferente (neutro)

☐ Concordo Parcialmente

☐ Concordo Totalmente

Eu gostei de usar a interface do aplicativo, é direto e objetivo em suas funções. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

A organização de informações na tela do aplicativo é clara. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

As informações fornecidas pelo aplicativo são fáceis de compreender. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Achei o aplicativo complexo no geral. *

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Aspectos Pedagógicos (Usabilidade)

Para além da tecnologia, é preciso verificar a relação pedagógica, com isso, foi preciso algumas perguntas para verificar o contexto do ensino e aprendizado do professor em relação a sua avaliação com os estudantes. Assim, esse questionário tem como objetivo, colaborar na construção de forma participante no projeto da monografia do aplicativo com os dados coletados desse questionário, que será repensado e reformulado, discutido para ser analisado e descrito todos os passos que possam ser alterados ou inseridos.

É necessário ensinar os estudantes para usar o aplicativo de modo geral? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

O/A docente pode preparar e modificar materiais de aprendizagem usando o aplicativo? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

O aplicativo facilita a avaliação do processo de aprendizagem? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

É possível criar testes, exames ou tarefas usando o aplicativo? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

É possível selecionar modos avançados de aprendizado de acordo com a experiência do usuário?

(exemplo: os desafios)*

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

O uso do aplicativo se encaixa no ambiente de aprendizado? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

O aplicativo complementa outras atividades do componente relacionado a área de programação? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

É possível que os usuários preparem uma apresentação usando esse aplicativo? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

Você recomendaria o aplicativo para fins de aprendizado e avaliação? *

- ☐ Sim
- ☐ Não
- ☐ Não sei
- ☐ Talvez

Outro: _____

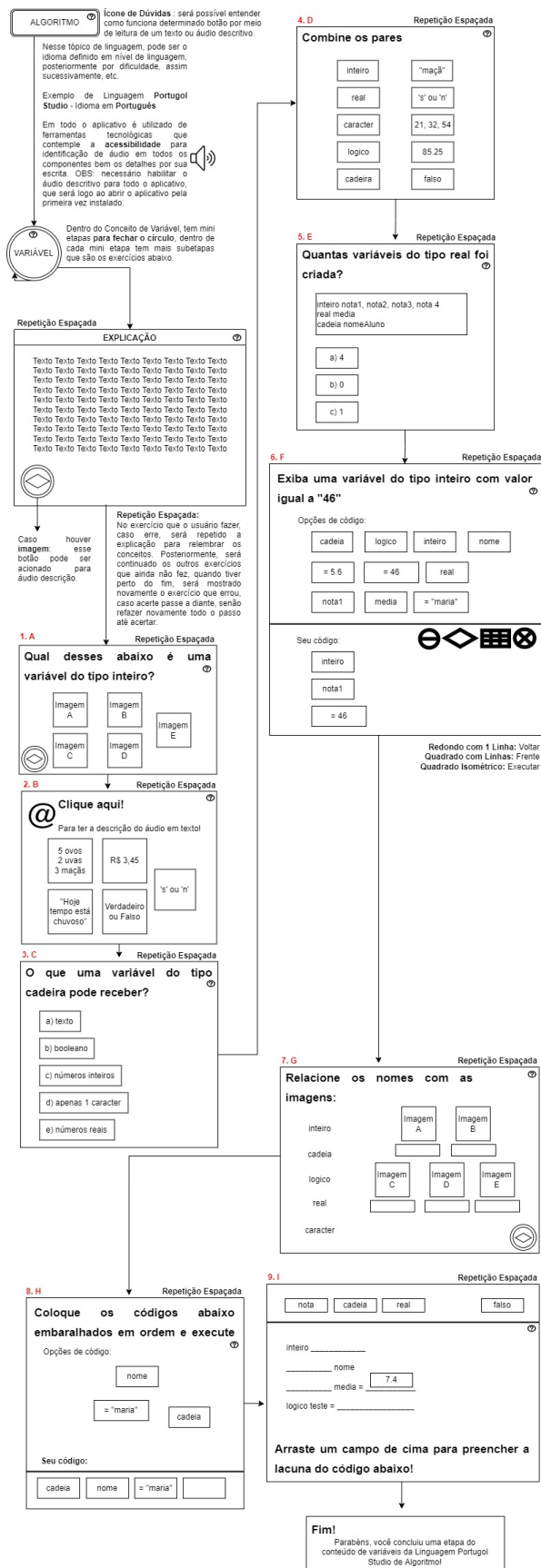
Este aplicativo tem todas as funções e recursos de ensino e aprendizagem que eu espero que ele tenha.*

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

No geral, estou satisfeito com este aplicativo.*

- ☐ Discordo Totalmente
- ☐ Discordo Parcialmente
- ☐ Indiferente (neutro)
- ☐ Concordo Parcialmente
- ☐ Concordo Totalmente

Apêndice E – Protótipo da Sequência Didática de Algoritmo



Link do Apêndice E: <https://bit.ly/fluxobanco>

