

Desenvolvimento de uma ferramenta para auxiliar alunos no processo de aprendizagem de lógica e algoritmos

Eliel Alves da Silva* Adilso Nunes de Souza†
Anubis Graciela de Moraes Rossetto‡

16 de dezembro de 2022

Resumo

Os elevados índices de evasão e reprovação nos primeiros semestres dos cursos de computação se caracteriza como um problema a nível internacional que atinge as esferas social, acadêmica e econômica. Dentre os diversos motivos, as dificuldades encontradas nas primeiras disciplinas e a falta de motivação são as principais barreiras descritas pelos alunos. Diante disso, desenvolveu-se uma ferramenta multiplataforma que possibilita o estudo de algoritmos e lógica de programação utilizando conceitos inerentes a gamificação. Para validação da aplicação foi realizada uma pesquisa qualitativa utilizando estudantes do curso de Ciência da Computação do IFSul - Campus Passo Fundo. Os resultados da pesquisa demonstram que apesar de cada aluno possuir um histórico de aprendizagem e um contexto totalmente diferente, o aplicativo pode auxiliar alunos ingressantes no processo de ensino e aprendizagem utilizando elementos da estratégia de gamificação que foquem e estimulem a motivação do aluno.

Palavras-chaves: Aplicativo Educacional. Desenvolvimento Multiplataforma. Gamificação.

Introdução

A evasão de acadêmicos nos primeiros anos de estudo em cursos de computação não é um fato recente. Pelo contrário, observa-se em vários casos uma alta taxa de evasão e reprovação nas disciplinas de algoritmos e lógica de programação desde o primeiro semestre nos mais diversos níveis de ensino (DETERS et al., 2008). Dados presentes na Tabela 1 apontam que no ano de 2021 a taxa de evasão dos cursos de Ciência da Computação, Engenharia de Software e Sistemas de Informação juntos correspondem à 57,33%, enquanto a taxa de conclusão do mesmo ciclo corresponde à 9,5%.

* <elielsilva.pf140@academico.ifsul.edu.br>

† <adilosouza@ifsul.edu.br>

‡ <anubisrossetto@ifsul.edu.br>

Tabela 1 – Eficiência Acadêmica

Ano Base	Índice de Eficiência Acadêmica	Conclusão Ciclo	Evasão Ciclo	Retenção Ciclo
2018	17,2%	268 (13,39%)	1.287 (64,32%)	446 (22,29%)
2019	19,9%	328 (14,83%)	1.320 (59,67%)	564 (25,50%)
2020	18,1%	278 (12,41%)	1.255 (56%)	708 (31,59%)
2021	9,50%	234 (9,50%)	1.412 (57,33%)	817 (33,17%)

Fonte: (MEC, 2021)

Filho et al. (2007) afirmam que a evasão é um dos problemas que afligem as instituições de ensino a nível internacional e que, no ensino superior, se trata de desperdícios sociais, acadêmicos e econômicos. No setor público, são recursos públicos investidos sem o devido retorno. Já no setor privado, é uma importante perda de receita. Independente do tipo de setor, a evasão pode ser tomada como fonte de ociosidade de professores, funcionários, equipamentos e até mesmo de espaço físico.

Gomes, Henriques e Mendes (2008) explicam que parte do índice de desistência dos cursos de computação é resultante das dificuldades encontradas pelos alunos em relação a conteúdos e habilidades que são necessárias na disciplina de Algoritmos, uma vez que esta estimula e trabalha com o raciocínio lógico do aluno que, muitas vezes chega ao ensino superior com uma formação básica deficitária no que tange a habilidades matemáticas e de interpretação de textos e enunciados, graças ao ensino recebido nos anos iniciais e a metodologia utilizada pelos professores (CARNEIRO, 2018).

Ao verificar os dados do Câmpus Passo Fundo, é possível observar, através da Tabela 2, que a média de todas as turmas das disciplinas de Algoritmos I (A1) e Algoritmos II (A2), antes do período pandêmico é inferior a seis, com destaque ao semestre 2018/2 que possui média 2,3 em A1. Ademais, a taxa de reprovação na disciplina de A1 é superior à 50% em praticamente todos os anos, com exceção dos semestres 2020/1 e 2021/1.

Tabela 2 – Desempenho acadêmico nas disciplinas de Algoritmos I e Algoritmos II do IFSul Câmpus Passo Fundo

Semestre	Disciplina	Matriculados	Aprovados	Reprovados	Média
2018/1	Algoritmos I	36	12 (33,3%)	24 (66,6%)	4,0
	Algoritmos II	23	14 (60,9%)	9 (39,1%)	4,4
2018/2	Algoritmos I	35	12 (34,3%)	23 (65,7%)	2,3
	Algoritmos II	18	6 (33,3%)	12 (66,7%)	4,0
2019/1	Algoritmos I	46	21 (45,7%)	25 (54,3%)	3,7
	Algoritmos II	23	15 (65,2%)	8 (34,8%)	4,5
2019/2	Algoritmos I	33	14 (42,4%)	19 (57,6%)	3,3
	Algoritmos II	26	18 (69,2%)	8 (30,8%)	5,4
2020/1	Algoritmos I	29	18 (62,1%)	11 (37,9%)	5,3
	Algoritmos II	18	15 (83,3%)	3 (16,7%)	5,2
2020/2	Algoritmos I	28	10 (35,7%)	18 (64,3%)	3,6
	Algoritmos II	19	18 (94,7%)	1 (5,3%)	8,0
2021/1	Algoritmos I	35	22 (62,9%)	13 (37,1%)	6,7
	Algoritmos II	12	11 (91,7%)	1 (8,3%)	7,3
2021/2	Algoritmos I	28	13 (46,4%)	15 (53,6%)	3,3
	Algoritmos II	19	15 (78,9%)	4 (21,1%)	6,4

Fonte: (IFSUL, 2022)

Considerando o cenário apresentado, e a ausência de ferramentas gratuitas e em português com o propósito de auxiliar alunos ingressantes na Computação, foi desenvolvida uma aplicação que busca contribuir com o estudo de lógica e programação dos alunos ingressantes no curso de Ciência da Computação do IFSul Câmpus Passo Fundo. Para validar tal contribuição, foi avaliada a experiência dos estudantes das disciplinas de Algoritmos I e Algoritmos II do Câmpus Passo Fundo durante o uso da aplicação proposta.

O presente trabalho está estruturado da seguinte forma: a seção 1 apresenta o referencial teórico com os conceitos, dados pertinentes e tecnologias utilizadas no desenvolvimento do trabalho, bem como os trabalhos relacionados. Na seção 2 é descrita a metodologia empregada para o trabalho, bem como os passos realizados para a construção do mesmo. Na seção 3 são descritos detalhes do desenvolvimento da aplicação proposta. Na seção 4 são expostos os resultados obtidos através da experiência dos estudantes na utilização do aplicativo. Por fim, são apresentadas as considerações finais.

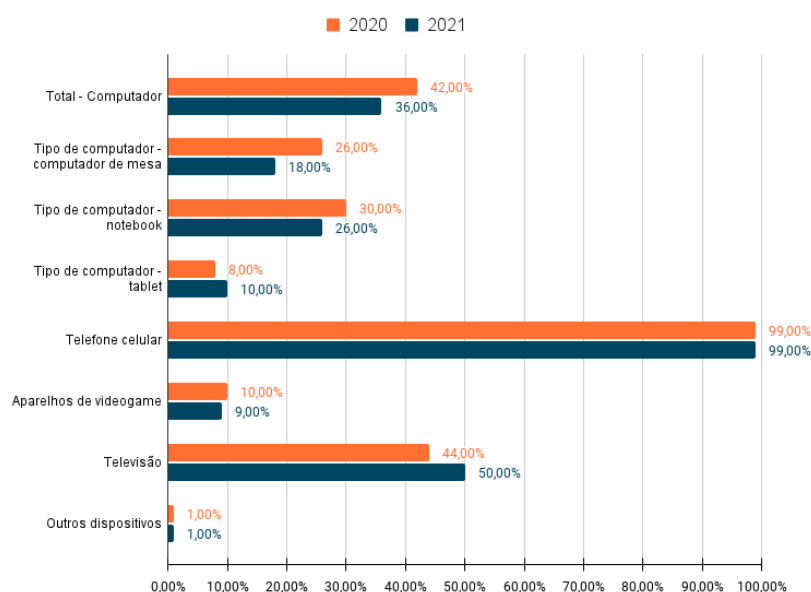
1 Fundamentação Teórica

Esta seção apresenta a fundamentação teórica utilizada para a implementação do presente trabalho, discorrendo conceitos, ferramentas e tecnologias que estão relacionadas com o desenvolvimento do aplicativo.

1.1 Uso de Smartphones

Segundo dados do [Cetic.br \(2021\)](#), no Brasil o celular é o equipamento mais utilizado para acessar a internet. A Figura 1 demonstra que no ano de 2021 a porcentagem das pessoas que acessam à internet por meio de *smartphones* continuou em 99%, enquanto o uso de televisão aumentou em 6%, por outro lado, a porcentagem das pessoas que acessam à internet por meio de microcomputadores ou *tablets* diminuiu ([CETIC.BR, 2021](#)).

Figura 1 – Usuários de internet, por dispositivo utilizado



Fonte: ([CETIC.BR, 2021](#))

Já se tratando do sistema operacional utilizado por esses aparelhos, dentre todos os sistemas operacionais de *smartphones* observa-se dois que possuem destaque no mercado: o Android e iOS, que no ano de 2021 possuem 86,05% e 13,65% do mercado brasileiro respectivamente (STATCOUNTER, 2021). Por outro lado, os demais sistemas operacionais de dispositivos *mobile* contemplam outros 0,3%.

1.2 Tipos de Aplicativos

Os aplicativos multiplataformas são aqueles que são construídos para que funcionem em mais de um sistema operacional ou plataforma, ou seja, são aplicativos que permitem serem utilizados tanto em um celular (Android ou iOS) quanto em um computador. Além do mais, é possível através do mesmo código fonte realizar mudanças em ambos aplicativos (Android, iOS ou *Web*) (FRAY, 2021).

Além dos *apps* multiplataformas, podemos classificar os aplicativos em: a) aplicativos nativos; b) aplicativos híbridos e c) PWAs.

- Aplicativos nativos: segundo MadeinWeb (2020), os aplicativos nativos são aqueles desenvolvidos especificamente para um sistema operacional, um exemplo são aplicativos que funcionam apenas em dispositivos iOS, ou seja, são exclusivos para o uso em iPhones;
- Aplicativos híbridos: diferente dos aplicativos nativos, os aplicativos híbridos são desenvolvidos para mais de um sistema operacional, dessa forma é possível que o mesmo aplicativo seja executado em um dispositivo Android e em um iPhone (MADEINWEB, 2020);
- PWA (Progressive Web Apps): essa categoria de aplicativos é a que mais se difere das demais, pois se trata de uma aplicação que foi construída para ser executada em um navegador e que funciona exatamente como um aplicativo nativo, MadeinWeb (2020) explica que nesse tipo de aplicativo, além do usuário conseguir utilizar a aplicação através de uma *url*, é possível também adicioná-la na tela inicial do *smartphone* como se fosse um aplicativo baixado diretamente da loja.

1.3 Aplicativos Multiplataformas

O desenvolvimento de aplicativos multiplataformas permite a construção de *apps* que funcionarão independente do sistema operacional ou plataforma utilizada, através dessa característica é possível obter um alcance de mercado de 6,64 bilhões de usuários de *smartphones* em todo o mundo (ATHA, 2022).

Além do grande alcance de número de usuários, Maharana (2017) destaca outras vantagens de desenvolver um aplicativo multiplataforma:

- Velocidade de desenvolvimento: Maharana (2017) explica que como o mesmo código pode ser utilizado para mais de uma plataforma, o tempo de desenvolvimento de *apps* multiplataformas é menor;
- Custo menor: como o desenvolvimento desse tipo de aplicativo necessita de menos tempo, o custo é consequentemente reduzido (MAHARANA, 2017);
- Alcance significativo: Maharana (2017) se refere ao fato de aplicativos multiplataforma poder alcançar usuários independente da plataforma ou sistema operacional utilizado, o que aumenta a possibilidade de utilização por usuários de qualquer tipo de dispositivo.

Por outro lado, aplicativos multiplataformas podem apresentar alguns problemas. [Fray \(2021\)](#) explica que eles podem não funcionar tão bem em todas as plataformas e sistemas desejados pois utilizam interfaces próprias, portanto, é possível que a experiência do usuário possa variar de acordo com o tamanho da tela ou da plataforma ou sistema operacional utilizado. Ademais, por não ser um *app* nativo, é de se esperar, por exemplo, que não seja possível utilizar todos os recursos disponíveis de um dispositivo *mobile*.

1.4 Flutter

Graças as vantagens já descritas sobre o desenvolvimento de *apps* multiplataforma, surgem diversos *frameworks* para auxiliar no desenvolvimento desse tipo de aplicação, [Atha \(2022\)](#) destaca o *framework* Flutter.

O Flutter é um SDK desenvolvido e sustentado pelo Google que permite o desenvolvimento de aplicações multiplataformas a partir de uma única base de código. Seu principal objetivo é permitir aos desenvolvedores a criação de *apps* de alta performance com experiência nativa independente da plataforma, dessa forma o usuário se sente exatamente como em um aplicativo nativo ([GOOGLE, 2022a](#)).

[Atha \(2022\)](#) afirma que somente no ano de 2021, mais de 40.000 aplicativos novos foram disponibilizados na Google Play Store se comparado ao ano anterior (2020). Além disso, [Protalinski \(2022\)](#) mostra que 42% dos desenvolvedores usam o Flutter para o desenvolvimento de aplicativos.

Alguns dos recursos que tornam o Flutter "perfeito para o desenvolvimento de aplicativos multiplataforma" ([ATHA, 2022](#)) são:

- Menos código: o Flutter utiliza uma linguagem de programação orientada a objetos que possui o seu estilo de digitação declarativa e reativa, portanto não necessita de uma "ponte JavaScript", ajudando dessa forma os desenvolvedores a escreverem seus códigos em um tempo relativamente menor ([ATHA, 2022](#));
- Alto desempenho: [Atha \(2022\)](#) explica que como a UI (interface do usuário) foi projetada para aquela plataforma, o usuário possui uma experiência totalmente imersiva como se estivesse utilizando um aplicativo nativo;
- Hot Reload: é considerada uma das funcionalidades mais atrativas do *framework*, é um recurso que permite que todas as alterações feitas no código-fonte possam ser visualizadas em tempo real para o desenvolvedor ([ATHA, 2022](#)).

1.5 Gamificação

Dentre os diversos fatores que influenciam no processo de ensino e aprendizagem no atual cenário brasileiro, há um destaque no que se refere a motivação ([FERNANDES, 2020](#)), uma vez que é perceptível que diversos autores, dentre eles [Júnior e Boniati \(2015\)](#), [Zacarias e Mello \(2019\)](#) e [Coelho e Vega \(2018\)](#), possuem preocupação quanto ao fator motivacional durante o processo de ensino e aprendizagem.

Diante disso, [Tolomei \(2017\)](#) propõe o uso de gamificação como estratégia de motivação nos ambientes de aprendizagem. O termo gamificação diz respeito a utilização de conceitos presentes em jogos que possuem um mecanismo de resolução de problemas práticos ou que despertam algum tipo de engajamento a algum público ([VIANNA et al., 2013](#)). Essa metodologia possui alta atratividade, pois ela alia a diversão com desafios que impulsionam a resolução de problemas por parte dos alunos.

É importante destacar que a ideia de utilizar a gamificação é algo relativamente novo, segundo Tolomei (2017) a utilização desses conceitos é uma consequência da proporção da popularidade de jogos, além das possibilidades de resolução de problemas que os *games* trazem para o aprendizado.

Dentre os elementos que devem estar presentes para que essa estratégia seja implementada de forma bem-sucedida, Tolomei (2017) destaca:

- Desafios, conquistas e missões: é considerado por Tolomei (2017) como um dos elementos mais importantes, pois é completando estes desafios e missões que fará com que o jogador se esforce cada vez mais;
- Recompensa: segundo Tolomei (2017), é preciso recompensar o jogador a cada ação positiva, uma das formas de estimular o usuário é dando a ele um *feedback* positivo;
- Progressão: ao contrário da estagnação, a progressão busca evidenciar ao jogador que ele está indo na direção certa e que está adquirindo aprendizado, criando uma lógica de evolução e um sentimento de motivação;
- Feedback instantâneo: Tolomei (2017) explica que a cada fase do jogo, é necessário dar um *feedback*, a fim de que os participantes tenham pleno conhecimento sobre seu rendimento, dessa forma é possível saber onde estão errando e assim identificar os pontos que precisam ser melhorados;
- Competição: segundo Tolomei (2017), a competitividade é inerente à natureza do ser humano, e através de uma disputa saudável em busca da vitória de um desafio, é possível gerar motivação e, conseqüentemente, um melhor desempenho.

Com o objetivo de tornar o aplicativo desenvolvido mais atrativo, optou-se por utilizar das mecânicas dessa metodologia citada acima para ensinar lógica de programação, possibilitando utilizar conceitos de jogos para ensinar alunos através da diversão.

1.6 Armazenamento de Dados

1.6.1 Banco de Dados NoSQL

Foi-se necessário definir como armazenar os dados do aplicativo proposto. Como o *framework* definido para o desenvolvimento foi o Flutter, optou-se por um banco de dados NoSQL pela facilidade de integração com o *framework* selecionado. Segundo Fowler e Sadalage (2012) existem dois principais motivos que levam os desenvolvedores a utilizar um banco de dados NoSQL: a) a produtividade e b) dados em grande escala.

Uma vez que se utiliza muito esforço no gasto de mapeamento dos dados em um banco de dados relacional (CORAZZA, 2018), é recomendado utilizar um banco de dados NoSQL, uma vez que este fornece um modelo de dados que melhor atenda às necessidades do *app* proposto, resultando em menos código para escrever.

Aplicações desenvolvidas para um número incerto de usuários e com informações que não podem ser deletadas com frequência necessitam de um poder maior de captura e processamento de dados. Graças a essas características, é comum que as empresas e organizações optem por utilizar bancos de dados não relacionais, uma vez que estes são projetados para operarem através de *clusters*, o que os torna mais adaptados para estes cenários (CORAZZA, 2018).

Através do cenário da aplicação proposta, definiu-se por utilizar um banco de dados NoSQL para guardar todas as informações da ferramenta.

1.6.2 Google Cloud Firestore

Os bancos de dados do tipo NoSQL possuem quatro classificações principais quanto ao seu modelo de dados: a) chave-valor; b) documentos; c) família de colunas e d) grafo (CORAZZA, 2018). Pretende-se utilizar o Firebase Cloud Firestore como solução para armazenamento em nuvem dos dados da aplicação proposta, que utiliza documentos como modelo de dados.

O Cloud Firestore permite realizar consultas para recuperar documentos individuais ou todos os documentos de uma coleção, além de permitir incluir filtros e combiná-los com classificação (GOOGLE, 2021).

Através de listeners é possível manter dados sincronizados da aplicação em tempo real. Dessa forma, é possível atualizar qualquer dado instantaneamente no aplicativo sem que seja necessário recuperar todo o banco de dados, ou seja, sempre que houver mudanças é possível recuperar somente as alterações (GOOGLE, 2021).

Por fim, o Cloud Firestore ainda armazena os dados em cache, permitindo que o aplicativo leia, escreva e consulte dados mesmo que esteja *offline*. E, assim que o dispositivo conseguir se conectar a internet, irá sincronizar os dados novamente no Cloud Firestore (GOOGLE, 2021).

1.7 Trabalhos Relacionados

Esta seção contempla alguns *apps* relacionados que já existem no mercado e que foram analisados e utilizados como referência para elencar as principais funcionalidades da solução proposta.

O primeiro aplicativo analisado foi o Grasshopper, se trata de um projeto criado por uma equipe de programadores do Google para produtos experimentais, denominado Area 120 (GRASSHOPPER, 2022). É um *app* totalmente gratuito de aprendizado de programação que possui o foco em JavaScript e o seu intuito é ajudar pessoas a entrar no mundo da programação de maneira fácil e divertida (GRASSHOPPER, 2022).

O *app* possui suporte ao idioma Português do Brasil e está disponível para download apenas para dispositivos Android, contando atualmente com mais de 5.000.000 de instalações do aplicativo na Play Store.

Também tem-se o Mimo, que é um *app* multiplataforma (Android e iOS) gratuito desenvolvido pela equipe Mimohello GmbH que oferece diferentes formas de aprendizado e possui uma interface simples e amigável (MIMO, 2022). Este aplicativo possui mais de 13.000 exercícios completos das mais variadas linguagens de programação com código real e feedback imediato, além de possibilitar escrever e executar códigos, salvar projetos e possuir um sistema de gamificação que incentiva o usuário a manter uma sequência para criar o hábito de programar (MIMO, 2022).

Diferente de outras propostas, o Mimo propõe lições mais objetivas e focadas em projetos que o usuário pode desenvolver para serem utilizadas no mundo real (MIMO, 2022). O *app* também oferece uma versão paga podendo variar o preço dependendo das funcionalidades optadas pelo usuário. Atualmente, possui suporte somente para o idioma Inglês e conta com mais de 10.000.000 de instalações somando os números da Play Store e App Store.

Por fim, o Encode é um *app* gratuito desenvolvido pela empresa Upskew Pty. Ltd. que ensina programação com lições divertidas e interativas e desafios práticos (ENCODE, 2021). Além de sua versão oficial lançada inicialmente para dispositivos Android, a empresa disponibilizou no ano de 2022 a sua versão para dispositivos iOS.

Segundo Werneck (2022), o *app* possui base em JavaScript e é muito interessante para iniciantes. Sua interface é fácil de usar e ainda oferece vários tutoriais que são bem explicados

e que possuem passo a passo com observações essenciais para começar, dessa forma se torna simples voltar algumas lições para recordar sempre que o usuário desejar (ENCODE, 2021).

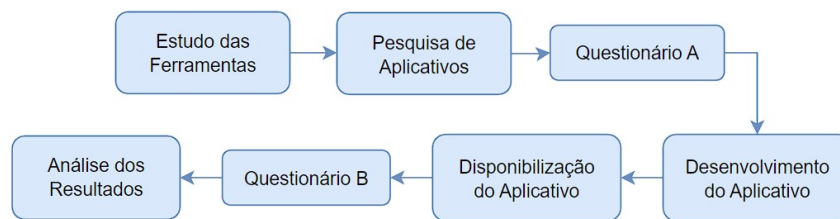
O usuário do aplicativo ainda pode optar por uma versão paga que contém lições e recursos adicionais. Atualmente, possui suporte somente para o idioma Inglês e conta com mais de 500.000 instalações somando os números da Play Store e App Store.

2 Metodologia

Esta seção apresenta a metodologia que foi empregada e suas respectivas etapas utilizadas para o desenvolvimento do presente trabalho.

Seguindo o fluxograma da Figura 2, teve-se como primeiro passo o estudo das ferramentas necessárias para o desenvolvimento da aplicação. Para tal etapa, foi-se necessário o acesso a documentação do Flutter e também do Firebase, o objetivo foi compreender como funciona cada uma das ferramentas, quais as suas funcionalidades e como cada uma delas impactariam ou seriam utilizadas no aplicativo. Além do mais, buscou-se tutoriais e minicursos para que auxiliassem durante o processo de desenvolvimento.

Figura 2 – Fluxograma do desenvolvimento do trabalho



Fonte: Do autor, 2022

Como segundo passo, teve-se a realização de uma pesquisa de aplicativos presentes no mercado com objetivos similares ao da aplicação proposta. Essa pesquisa teve como objetivo identificar quais são os elementos e funcionalidades presentes nos aplicativos que propõem o ensino de algoritmos ou linguagem de programação. Também buscou-se encontrar características ou falhas existentes nos aplicativos encontrados que pudessem ser exploradas na aplicação desenvolvida, buscando desta forma, criar um diferencial das demais aplicações.

Foi-se necessário realizar uma curadoria nos resultados da pesquisa, uma vez que a quantidade de aplicativos encontrados foi elevada para o propósito inicial. O resultado da pesquisa dos aplicativos pode ser encontrada na seção 1.7.

Em seguida, foi criado um questionário utilizando a ferramenta Google Forms (GOOGLE FORMS, 2022). Esse questionário foi disponibilizado durante a primeira semana do mês de setembro de 2022 para os estudantes ingressantes do curso de Ciência da Computação do câmpus Passo Fundo pelos professores das disciplinas de Algoritmos I e II e também através do servidor do curso na plataforma Discord. Os alunos tiveram o período de duas semanas para responder o questionário e seu objetivo foi identificar o cenário e o contexto dos estudantes antes do ingresso ao curso.

Com as informações adquiridas no primeiro questionário foi possível detectar as deficiências dos alunos no que tange ao conhecimento matemático ou raciocínio lógico, portanto, foi possível alimentar o conteúdo do *app* de forma mais assertiva.

O próximo passo consistiu na união das informações obtidas na pesquisa dos aplicativos e no primeiro questionário respondido pelos acadêmicos para desenvolver o aplicativo, os detalhes

do desenvolvimento da aplicação estão descritos na seção 3.2.

Após o desenvolvimento da aplicação, a mesma foi disponibilizada via navegador para ser testada pelos mesmos alunos que responderam o primeiro questionário. O aplicativo foi disponibilizado pelos próprios professores das disciplinas de Algoritmos I e II e também através do servidor do curso na plataforma Discord.

Também foi criado um segundo questionário utilizando a ferramenta Google Forms (GOOGLE FORMS, 2022). O intuito desse questionário foi de descobrir como foi a experiência dos estudantes com o uso do *app*, além de verificar sua contribuição para o aprendizado das necessidades apontadas inicialmente por eles. O questionário foi disponibilizado juntamente com a aplicação na segunda semana do mês de outubro de 2022, os alunos tiveram o período de três semanas para respondê-lo.

Por fim, foram analisadas as respostas dos questionários disponibilizados. Essa etapa teve como objetivo utilizar dos dados obtidos através da pesquisa para compreender o que eles representam, bem como identificar a contribuição do aplicativo desenvolvido no contexto em que foi disponibilizado e visualizar os pontos positivos e negativos do mesmo. Os resultados estão descritos na seção 4.

3 Aplicação Proposta

Esta seção apresenta as tecnologias que foram utilizadas durante o desenvolvimento, a proposta da aplicação e o seu resultado final.

3.1 Tecnologias utilizadas

Para desenvolver o aplicativo foi necessário a utilização de diversas tecnologias que serão descritas a seguir.

3.1.1 IntelliJ IDEA Ultimate

O IntelliJ IDEA é um ambiente de desenvolvimento integrado (IDE) desenvolvido e mantido pela JetBrains (2022). Ele permite desenvolver em linguagens como Java, Kotlin, Groovy e Dart com facilidade.

A IDE possui diversas funcionalidades que auxiliam no desenvolvimento de uma aplicação, dentre elas: a) assistência de codificação: análise, refatoração, depuração e completamento de código; b) compiladores e ferramentas integradas: integração com versionamento de código (Git), bancos de dados, compiladores Gradle e c) ecossistema de *plugins*: suporte a *plugins* que permitem adicionar funcionalidades adicionais ao IDE, como é o caso do desenvolvimento com Dart e Flutter.

3.1.2 Flutter

Como já citado anteriormente neste artigo, optou-se por utilizar o Flutter como tecnologia para desenvolver o aplicativo, dado os conhecimentos prévios por parte do desenvolvedor e também as vantagens e funcionalidades do *framework*.

Por se tratar de uma tecnologia desenvolvida e mantida pelo Google (2022a), o Flutter possui uma documentação oficial vasta e completa contendo exemplos práticos de todos os *widgets* disponíveis para utilização, além de tutoriais em vídeo. Aliado a isso, o Flutter utiliza a linguagem Dart, que por sua vez possui um repositório de pacotes exclusivo para aplicativos desenvolvidos em Flutter e Dart, que também é mantido pelo Google. Nesse repositório é possível

encontrar pacotes e bibliotecas de código tanto dos desenvolvedores oficiais do Flutter quanto da comunidade, que facilitaram o desenvolvimento do aplicativo.

3.1.3 Firebase

O Firebase, é uma plataforma em nuvem que contém um conjunto de serviços que facilitam o desenvolvimento de aplicativos *web* ou móveis de maneira rápida, simples e efetiva (GOOGLE, 2022b). Através dela, foi possível criar toda a infraestrutura do *back-end* necessária de forma totalmente gerenciável.

Dentre os serviços disponibilizados pela plataforma, foram utilizados: a) Cloud Firestore: como visto na seção 1.6.2, o Firestore é um banco de dados NoSQL que permite armazenar, sincronizar e consultar dados facilmente em escala global, permitindo também sincronizar os dados de forma *online* e *offline*; b) Authentication: é uma solução de identidade completa, além de ser seguro, ele possui autenticação compatível com e-mail e senha, telefone, além de vários outros provedores como Google, Facebook, Twitter, Github e outros, também com ele é possível gerenciar os usuários da aplicação de maneira simples e visual e c) Hosting: é um serviço de hospedagem e implantação de sites com segurança, CDN (rede de distribuição de conteúdo), escalabilidade e certificado SSL, foi utilizada na versão *web* do aplicativo.

Assim como o Flutter, o Firebase possui uma documentação completa e detalhada para cada serviço, além do mais, por se tratar de ser uma plataforma amplamente conhecida no mercado, encontra-se facilmente tutoriais e materiais em fóruns e *blogs*.

3.2 O Aplicativo

O *app* desenvolvido foi denominado "Logif", seu principal objetivo é auxiliar no processo de aprendizagem de lógica de programação e conceitos de algoritmos de forma simples, didática e divertida. O aplicativo é organizado em "categorias", que representam os temas e conceitos estudados nas primeiras disciplinas dos cursos que possuem programação como componente curricular. As categorias são trabalhadas por meios de *cards* e *quizzes* da seguinte maneira:

- Card: os *cards* são telas que contêm informações apresentadas de forma sequencial e progressiva, ou seja, cada categoria tem um número variado de *cards*. Cada *card* apresenta um conceito diferente referente a categoria ao qual ele pertence e só é disponibilizado para leitura após a exibição do *card* anterior (com exceção do primeiro);
- Quiz: os *quizzes* são questões que envolvem conceitos e assuntos tratados nos *cards*, contendo apenas uma opção correta. Eles são responsáveis por recompensar o usuário pelo aprendizado, além de criarem um sentimento de desafio e serem o meio pelo qual é verificada a compreensão de uma categoria de estudo. Por este motivo, eles são disponibilizados para serem respondidos apenas ao final da exibição de todos os *cards* de uma categoria de estudo escolhida.

A Tabela 3 apresenta as categorias de estudos que foram disponibilizadas dentro do aplicativo em sua primeira versão.

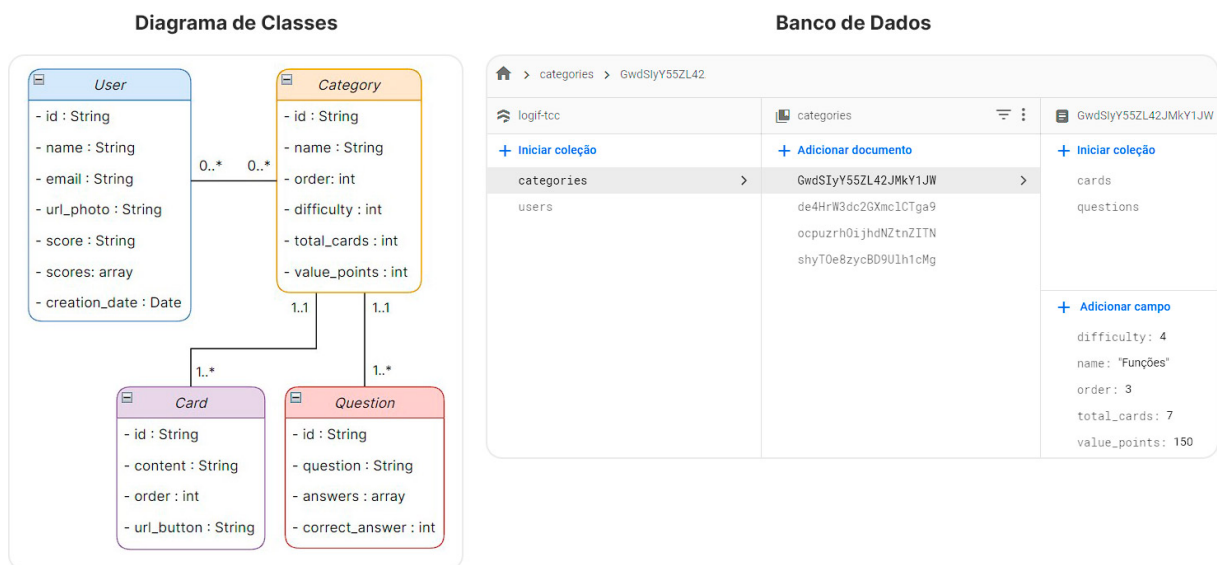
Tabela 3 – Categorias dos conteúdos do aplicativo

Nome	Classificação	Pontos
Proposições Lógicas	Iniciante	50
Operadores Lógicos	Iniciante	50
Condições	Intermediário	100
Funções	Avançado	150

Fonte: Do autor, 2022

Quanto aos dados, a Figura 3 ilustra ao lado esquerdo o diagrama de classes elaborado para representar a sua estruturação dentro do banco de dados, já do lado direito é possível visualizar como os mesmos ficaram estruturados dentro do Cloud Firestore.

Figura 3 – Estrutura do banco de dados do aplicativo

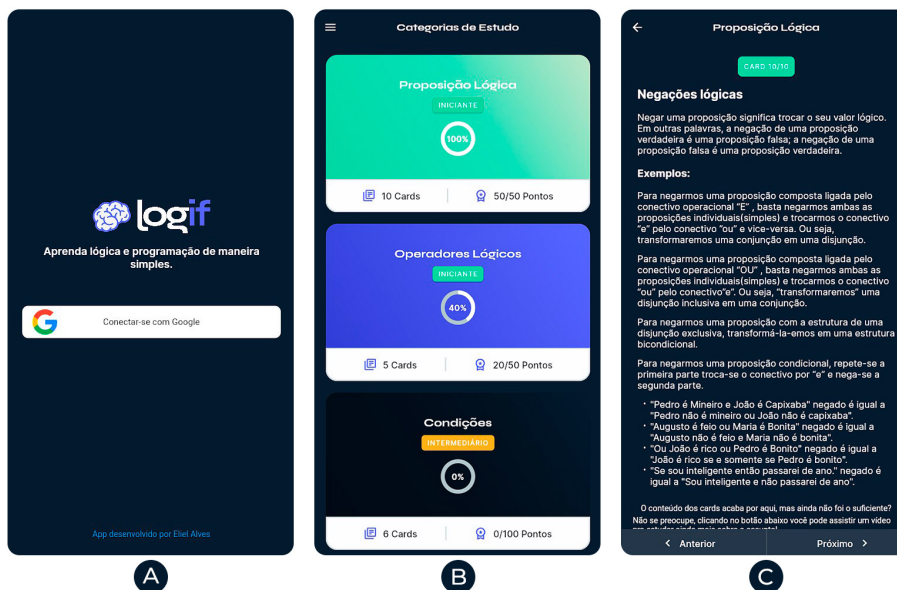


Fonte: Do autor, 2022

A Figura 4 apresenta três telas desenvolvidas, a tela A representa a tela de login e é somente por meio dela que o usuário conseguirá ter acesso as funcionalidades do aplicativo. A tela B representa as categorias descritas anteriormente, para que o usuário estude alguma categoria basta clicar em cima de uma das opções disponibilizadas, também é exibido em cada categoria o seu nível de dificuldade, o progresso do usuário, a quantidade de *cards* disponíveis para estudo e a pontuação máxima alcançável naquela categoria. A tela C representa um dos *cards* da categoria que foi selecionada, é possível observar que nela é exibida a quantidade de *cards* disponíveis naquela categoria, o conteúdo do *card* em si, bem como os botões de avançar e retroceder, permitindo que o usuário navegue entre os *cards* sempre que desejar.

Cada categoria possui um nível de dificuldade que influenciará diretamente na recompensa que será obtida por cada usuário, dependendo de seu desempenho. Essa recompensa será contabilizada e exibida ao usuário apenas ao finalizar um *quiz*. Buscando explorar a competitividade dentro do aplicativo, foi desenvolvido um *ranking* contendo a pontuação de todos os usuários em ordem decrescente, estimulando uma disputa saudável entre os jogadores em busca da vitória.

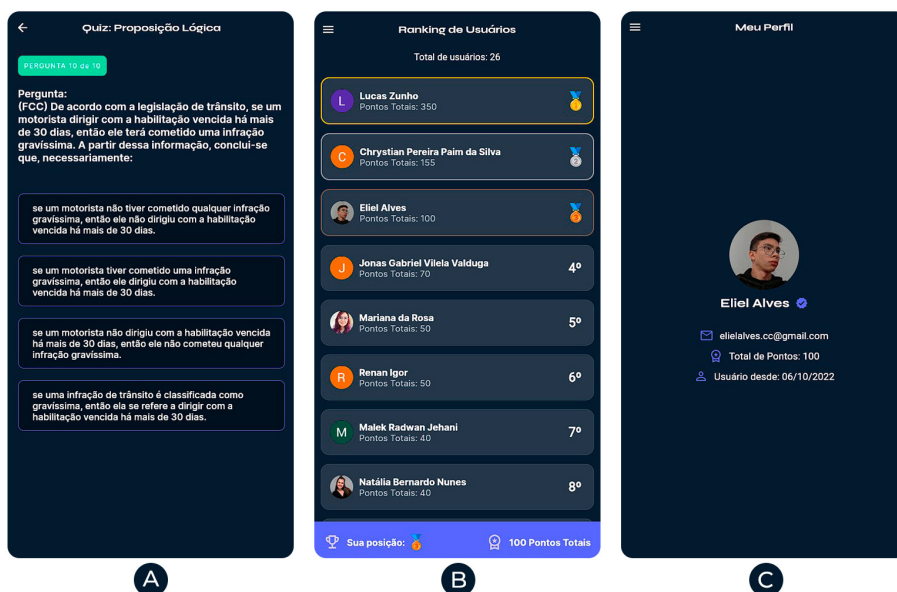
Figura 4 – Telas: login, categorias e cards



Fonte: Do autor, 2022

A Figura 5 apresenta outras três telas desenvolvidas, a tela A representa o *quiz*, nota-se que o usuário poderá visualizar quantas questões já foram respondidas e quantas ainda faltam, entretanto é impossível que o usuário retroceda em questões que já foram respondidas. Já a tela B representa o *ranking* do aplicativo, através dele é possível saber a pontuação e posição atual do usuário, bem como visualizar a pontuação de outros jogadores e suas medalhas. A tela C representa o perfil do usuário, onde são exibidas suas informações. Além destas, a última tela disponível para o usuário é a de informações sobre o aplicativo, nela contêm a descrição da aplicação e alguns *links* úteis para os usuários, como por exemplo *feedback* e reporte de *bugs*.

Figura 5 – Telas: quiz, ranking e perfil



Fonte: Do autor, 2022

4 Análise dos Resultados

Para realizar a análise dos resultados, foram comparados os problemas identificados por parte dos estudantes durante a primeira fase da pesquisa e o *feedback* disponibilizado por eles após a utilização do aplicativo. Buscou identificar os benefícios da utilização do *app*, bem como a contribuição no que tange ao conhecimento dos acadêmicos. Também buscou-se destacar os elementos da estratégia de gamificação que foram utilizados no aplicativo e como estes influenciaram no *feedback* dos alunos. Por fim, a análise da aplicação buscou encontrar diferenciais e pontos fortes da implementação que difere dos trabalhos relacionados descritos na seção 1.7.

O aplicativo foi avaliado através de uma pesquisa qualitativa realizada em duas etapas com os acadêmicos do curso de Ciência da Computação do câmpus Passo Fundo. Na primeira etapa, os alunos ainda não tinham contato com o aplicativo e responderam um questionário contendo perguntas intencionalmente focadas na opinião dos participantes em relação ao curso e disciplinas que envolviam algoritmos.

Nessa etapa da pesquisa, 35 alunos responderam o questionário que foi disponibilizado, observou-se que a maioria dos alunos (74,3%) não possuía nenhum conhecimento prévio em algoritmos antes de ingressar no curso, e que um pouco mais da metade deles (51,4%) avaliam os conhecimentos prévios obtidos no ensino médio como regular (37,1%) ou ruim (14,3%). Esses dados indicam que há fatores que não podem ser controlados por parte da instituição acadêmica, visto que cada aluno possui um histórico diferente e, por esta causa, deve ser levado em conta uma forma de ensino que independa do nível de conhecimento do aluno ao ingressar no curso.

Quanto ao fator tempo, foi possível observar que a maioria dos alunos (68,6%) trabalham, sendo sua maior parte com carga horária superior a vinte e inferior a quarenta horas semanais (40%). As respostas também demonstram que quase metade dos estudantes (45,7%) dedicam apenas de duas a cinco horas semanalmente para os estudos fora do período acadêmico, seguido de 22,9% de estudantes que dedicam menos de duas horas ou nenhuma. Além do mais, nota-se que a maioria dos acadêmicos (65,7%) sente que há falta de tempo para estudar.

Dentre os principais fatores apontados pelos alunos que influenciariam diretamente na evasão do curso, são destacados: a) ausência nas aulas (65,7%); b) falta de tempo para estudar (57,1%); c) pouca compreensão das aulas ou do conteúdo (57,1%); d) dificuldade na compreensão dos enunciados dos exercícios (42,8%) e e) não fazer os exercícios sugeridos pelo professor (40%).

Ao analisar os dados da primeira etapa da pesquisa foi possível observar a dificuldade por parte dos alunos na interpretação dos exercícios que são propostos no ambiente de aula e a preocupação quanto ao tempo disponível para estudo. Esses fatores foram levados em conta no desenvolvimento do aplicativo e, por esta causa, optou-se por utilizar a estratégia de apresentar conteúdos através de *cards* interativos, visando facilitar a compreensão e a progressão em um tempo que é determinado pelo aluno. Além do mais, os mesmos *cards* foram pensados para serem divididos contendo o mínimo conteúdo útil possível, otimizando o tempo do usuário e dando-lhe a sensação de progresso mais rapidamente.

O questionário final da pesquisa foi respondido por 26 alunos, correspondente a 69,4% dos participantes da primeira etapa. Nesta fase da pesquisa, o foco das perguntas foi a experiência dos usuários durante a utilização do aplicativo. As respostas apresentaram que 46,2% dos alunos utilizaram o aplicativo de duas à quatro horas antes de responderem o questionário, seguido de 42,3% que utilizaram aproximadamente uma hora. As respostas das demais perguntas objetivas estão representadas na Tabela 4.

Em comparação ao primeiro questionário, observou-se que a maioria dos alunos utilizam de duas à cinco horas semanais para estudo, que correspondeu ao tempo utilizado no aplicativo pela maioria dos estudantes.

Tabela 4 – Experiência dos usuários utilizando o aplicativo

Afirmações	Respostas
Utilizaria o aplicativo com frequência	Concordo totalmente: 53,8% Concordo: 26,9% Neutro: 19,2%
Achei o aplicativo desnecessariamente complexo	Discordo totalmente: 84,6% Discordo: 15,4%
Achei o aplicativo fácil de usar	Concordo totalmente: 88,5% Concordo: 11,5%
Achei o aplicativo muito inconsistente	Discordo totalmente: 88,5% Discordo: 11,5%
Achei o aplicativo fluído e bem otimizado	Concordo totalmente: 61,5% Concordo: 34,6% Neutro: 3,8%
Imagino que a maioria das pessoas aprenderiam a usar o aplicativo rapidamente	Concordo totalmente: 73,1% Concordo: 26,9%
A interface do sistema é agradável e atrativa	Concordo totalmente: 80,8% Concordo: 19,2%
A organização de informações na tela do sistema é clara	Concordo totalmente: 65,4% Concordo: 34,6%
O aplicativo explica o conteúdo de forma didática e sequencial	Concordo totalmente: 65,4% Concordo: 34,6%
O aplicativo me motivou a estudar mais sobre Algoritmos	Concordo totalmente: 46,2% Concordo: 38,5% Neutro: 15,4%
Consegui compreender melhor certos conteúdos através do aplicativo	Concordo totalmente: 76,9% Concordo: 19,2% Neutro: 3,8%
Creio que meu desempenho em Algoritmos I teria sido melhor se conhecesse o aplicativo antes	Concordo totalmente: 61,5% Concordo: 34,6% Neutro: 3,8%
Recomendaria o aplicativo facilmente para meus colegas	Concordo totalmente: 73,1% Concordo: 23,1% Neutro: 3,8%

Fonte: Do autor, 2022

Ao analisar os dados apresentados na tabela acima, percebe-se que a maioria dos alunos (53,8%) concordaram totalmente com a afirmação de que utilizaria o aplicativo com frequência, enquanto 26,9% apenas concordaram e outros 19,2% ficaram neutros, ou seja, nem concordaram e nem discordaram da afirmação. Já quanto a complexidade do aplicativo, pode-se dizer que todos os usuários concordaram que o aplicativo é fácil de usar.

A ideia de utilizar o *quiz* como forma de validação do estudo foi um dos pontos destacados pelos alunos, pois é um elemento que possui *feedback* instantâneo, além de ser um desafio pro usuário que resulta em uma recompensa, ambas são características da estratégia de gamificação.

Apesar de otimização não ser um dos pontos fortes do Flutter, no que diz respeito ao desempenho, a maioria dos usuários (96,1%) concordaram que o aplicativo é fluído e bem otimizado, outros 3,8% responderam neutro. Também é possível perceber que todos os usuários concordaram que a interface do *app* é agradável e atrativa e que as informações estão expostas de maneira simples e clara. Portanto, observou-se que o Flutter é uma ótima opção para quem deseja desenvolver aplicativos menores e menos complexos, e ao mesmo tempo obter um bom desempenho em aplicativos multiplataformas.

Dentre os principais problemas apontados na primeira etapa da pesquisa, observou-se que um deles é a não compreensão dos conteúdos e dos exercícios disponibilizados pelos professores. Olhando pela perspectiva do ensino e aprendizagem, 100% dos alunos disseram que o aplicativo

explica o conteúdo de forma didática e sequencial. Apesar do aplicativo utilizar de elementos da gamificação e outras estratégias para incentivar o aprendizado, percebe-se que ainda há uma parcela dos alunos (15,4%) que respondeu neutro quanto a afirmação de que o aplicativo o motivou a estudar mais sobre algoritmos, mesmo que 38,5% concordaram e 46,2% concordaram totalmente. Observa-se portanto, que há ainda outros fatores que influenciam na motivação do aluno.

Já sobre o desempenho dos alunos na disciplina de Algoritmos I, a maioria das respostas (96,1%) demonstram que os acadêmicos conseguiram compreender melhor certos conteúdos e conceitos através do aplicativo, além de acreditarem que o desempenho na disciplina de Algoritmos I poderia ter sido melhor caso os mesmos conhecessem o *app* previamente. Com esse dado, é possível afirmar que as estratégias empregadas de utilizar *cards* e *quizzes* são uma ótima opção de ensino gradual e sequencial, o que facilita na compreensão do conteúdo por parte do aluno.

O aplicativo também utilizou da competitividade - que também é um elemento da estratégia de gamificação - para criar uma disputa saudável entre os usuários. Como resultado teve-se o *ranking*, os alunos responderam que gostaram da sensação de competição e das premiações das medalhas de destaque para os primeiros colocados. Por fim, 73,1% dos usuários concordaram totalmente com a afirmação de que recomendaria o aplicativo para outros colegas, enquanto 23,1% apenas concordaram e outros 3,8% responderam neutro, ou seja, nem concordaram e nem discordaram da afirmação.

Com base nos resultados da pesquisa, pode-se confirmar que os alunos possuem interesse na utilização de outras ferramentas que sirvam como auxílio no ensino de lógica de programação e algoritmos. Entretanto, é necessário que esta seja atrativa e que possua mecanismos que motivem a progressão do aluno durante o processo de aprendizagem. Ademais, percebeu-se que a gamificação é uma ótima estratégia e que pode ser utilizada em ambientes acadêmicos, e que através dela é possível apresentar conceitos e conteúdos de forma diferente e atrativa permitindo a evolução dos alunos de forma prática na resolução de problemas.

Com o desenvolvimento do aplicativo utilizando o *framework* Flutter, percebeu-se que é possível desenvolver um *app* que execute em várias plataformas e sistemas operacionais utilizando apenas um código-fonte, facilitando a manutenção da aplicação por parte do desenvolvedor. Por outro lado, foi apontado por um usuário que o *app* não funcionou corretamente no navegador Firefox, ou seja, é possível que a aplicação possua algumas incompatibilidades ou problemas em plataformas específicas. Ainda sim, o desenvolvimento multiplataforma é uma ótima opção para quem quer desenvolver um aplicativo em pouco tempo e busque um maior alcance de usuários.

Graças a documentação fornecida pelos desenvolvedores e aos vídeos tutoriais disponíveis pela internet afora foi possível resolver todos os empecilhos que surgiram durante o desenvolvimento. Apesar de encontrar algumas complicações durante a integração entre as plataformas, não foram encontradas dificuldades maiores durante o estudo do Flutter e do Firebase. Portanto, pode-se dizer que qualquer pessoa conseguirá rapidamente aprender o suficiente sobre o Flutter para construir qualquer aplicação.

Considerações finais

O presente trabalho teve como objetivo o desenvolvimento de um aplicativo multiplataforma utilizando estratégias de gamificação para auxiliar no processo de ensino e aprendizagem de algoritmos e lógica de programação. Através do aplicativo desenvolvido, foi possível apresentar conceitos e definições de temas que são essenciais para o processo de aprendizagem de algoritmos. Para desenvolver o aplicativo, foram pesquisadas outras aplicações com objetivos similares ou parecidos e também ferramentas e tecnologias necessárias.

Apesar de já haver um conhecimento prévio por parte do desenvolvedor em relação as

tecnologias utilizadas, foi necessário estudá-las para desenvolver a aplicação. Mesmo com o estudo da tecnologia, durante o desenvolvimento do *app* foram encontradas dificuldades de integrações entre as plataformas utilizadas, entretanto foi possível saná-las e concluir o desenvolvimento como esperado.

Através dos questionários foi possível compreender o contexto e o perfil dos alunos ingressantes no curso de Ciência da Computação no câmpus Passo Fundo e verificar que o aplicativo proposto contribuiu positivamente no ensino de lógica de programação, além de demonstrar que a gamificação é uma estratégia que permite ensinar de maneira atrativa podendo contribuir e colaborar no processo de ensino destes conteúdos para os acadêmicos. Percebeu-se também que, no que diz respeito ao histórico de aprendizagem dos alunos, há fatores que são exclusivos e inerentes ao contexto que foram ou estão inseridos, e portanto, o aplicativo não pôde contribuir ou auxiliar.

Mesmo tendo poucas categorias de estudo disponibilizadas inicialmente, pode-se afirmar que os objetivos do trabalho foram alcançados, uma vez que a ferramenta desenvolvida tem a capacidade de auxiliar no processo de ensino e aprendizagem de algoritmos e lógica de programação, e que, pode contribuir para a compreensão de conceitos de forma mais atrativa, simples, didática e gradual.

Como trabalhos futuros, pretende-se desenvolver atualizações para o aplicativo levando em conta as sugestões apresentadas pelos usuários e adicionar novas categorias de estudo que contemplem a grade curricular das disciplinas de Algoritmos I e II, como por exemplo ponteiros, *structs*, matrizes e vetores. Dessa forma, será possível tornar o aplicativo uma ferramenta mais flexível e efetiva para ensino.

Pretende-se por fim, adicionar novas funcionalidades, dentre elas destacam-se: a) *feedback* imediato no *quiz* a cada questão, atualmente o aplicativo permite ao usuário visualizar o seu desempenho após responder todas as perguntas, entretanto acredita-se que seja melhor para o usuário saber quais questões ele acertou e errou a cada resposta fornecida; b) medalhas e sistema de conquistas; c) tornar *cards* mais atrativos, destacar trechos de código e deixar a leitura mais dinâmica; d) adicionar animações entre telas; e) sistema de contribuição da comunidade, permitindo que os usuários sugiram novos temas, *cards* e perguntas; f) sistema administrador que permita que um usuário específico consiga adicionar conteúdos dentro do próprio aplicativo.

Development of a tool to assist students in the learning process of logic and algorithms

Eliel Alves da Silva* Adilso Nunes de Souza†
Anubis Graciela de Moraes Rossetto‡

16 de dezembro de 2022

Abstract

The high rate of evasion or failure in the first semesters of computer courses is characterized as an international problem that affects the social, academic and economic spheres. Among the various reasons, the difficulties encountered in the first subjects and the lack of motivation are the main barriers described by the students. Therefore, a multiplatform tool was developed that allows the study of algorithms and programming logic using concepts inherent to gamification. To validate the application, a qualitative research was carried out using students from the Computer Science course at IFSul campus step Fundo. The research results show that although each student has a learning history and a totally different context, the application can help new students in the teaching and learning process using elements of the gamification strategy that focus and stimulate student motivation.

Key-words: Educational Application. Multiplatform Development. Gamification.

Referências

ATHA, H. *Best Cross Platform App Development Frameworks 2022*. 2022. Disponível em: <<https://www.moveoapps.com/blog/best-cross-platform-app-development-frameworks/>>. Acesso em: 28 mai 2022. Citado 2 vezes nas páginas 4 e 5.

CARNEIRO, L. de N. S. *Aprendizagem da matemática: Dificuldades para aprender conteúdos matemáticos por estudantes do ensino médio*. UFPA, 2018. Citado na página 2.

* <elielsilva.pf140@academico.ifsul.edu.br>

† <adilsoSouza@ifsul.edu.br>

‡ <anubisrossetto@ifsul.edu.br>

- CETIC.BR, C. R. d. E. p. o. D. d. S. d. I. *TIC Domicílios 2021: C16 - USUÁRIOS DE INTERNET, POR DISPOSITIVO UTILIZADO*. 2021. Disponível em: <<https://cetic.br/pt/tics/domicilios/2021/individuos/C16/expandido>>. Acesso em: 25 jun 2022. Citado na página 3.
- COELHO, U. M.; VEGA Ítalo S. O uso da técnica de narrativa occ-rdd como apoio para elaboração e ou roteirização de conteúdo das aulas ministradas nas disciplinas dos cursos de computação. *CIET:EnPED*, 2018. ISSN 2316-8722. Disponível em: <<https://cietenped.ufscar.br/submissao/index.php/2018/article/view/812>>. Acesso em: 18 jun 2022. Citado na página 5.
- CORAZZA, P. V. *Um aplicativo multiplataforma desenvolvido com Flutter e NoSQL para o cálculo da probabilidade de apendicite*. [S.l.]: Universidade Federal do Rio Grande do Sul, 2018. Citado 2 vezes nas páginas 6 e 7.
- DETERS, J. I. et al. O desafio de trabalhar com alunos repetentes na disciplina de algoritmos e programação. *Simpósio Brasileiro de Informática na Educação*, 2008. Citado na página 1.
- ENCODE, U. P. L. *Encode for Android - Learn to code with bite-sized programming lessons*. 2021. Disponível em: <<https://upskew.com/encode-android/>>. Acesso em: 20 mai 2022. Citado 2 vezes nas páginas 7 e 8.
- FERNANDES, W. A. *Desenvolvimento de um aplicativo para auxiliar na preparação de alunos de Engenharia de Computação para o ENADE*. 2020. Disponível em: <https://repositorio.pucgoias.edu.br/jspui/bitstream/123456789/1361/1/TCC_II_WARLEY_ALVES_FERNANDES.pdf>. Acesso em: 05 jun 2022. Citado na página 5.
- FILHO, R. L. L. e S. et al. A evasão no ensino superior brasileiro. *Cadernos de Pesquisa*, v. 37, n. 132, p. 641–659, 2007. Citado na página 2.
- FOWLER, M. J.; SADALAGE, P. J. *Nosql Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. [S.l.]: Addison-Wesley Professional, 2012. Citado na página 6.
- FRAY, J. *O que é um aplicativo multiplataforma?* 2021. Disponível em: <<https://www.invoisys.com.br/blog/o-que-e-um-aplicativo-multiplataforma/>>. Acesso em: 20 mai 2022. Citado 2 vezes nas páginas 4 e 5.
- GOMES, A.; HENRIQUES, J.; MENDES, A. J. Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação e Tecnologias*, v. 1, p. 93–103, 2008. Citado na página 2.
- GOOGLE. *Cloud Firestore*. Universidade Federal do Rio Grande do Sul, 2021. Disponível em: <<https://firebase.google.com/docs/firestore/>>. Acesso em: 08 jun 2022. Citado na página 7.
- GOOGLE. *Flutter - Build apps for any screen*. 2022. Disponível em: <<https://flutter.dev/>>. Acesso em: 20 mai 2022. Citado 2 vezes nas páginas 5 e 9.
- GOOGLE, D. *Firebase*. 2022. Disponível em: <<https://firebase.google.com/>>. Acesso em: 16 nov 2022. Citado na página 10.
- GOOGLE FORMS, G. *Formulários Google: Pesquisas e formulários fáceis de criar para todos*. 2022. Disponível em: <<https://workspace.google.com/intl/pt-BR/products/forms>>. Acesso em: 25 jun 2022. Citado 2 vezes nas páginas 8 e 9.
- GRASSHOPPER, G. *Grasshopper - Learn to Code*. 2022. Disponível em: <https://grasshopper.app/pt_br/>. Acesso em: 20 mai 2022. Citado na página 7.

IFSUL, C. de Regulação Acompanhamento e Controle das I. E. *Registros acadêmicos de Algoritmos I e II de 2018 a 2021*. 2022. Citado na página 2.

JETBRAINS. *IntelliJ IDEA: The Capable Ergonomic Java IDE by JetBrains*. 2022. Disponível em: <<https://www.jetbrains.com/idea/>>. Acesso em: 16 nov 2022. Citado na página 9.

JÚNIOR, R. P. M.; BONIATI, B. B. Logicblocks: Uma ferramenta para o ensino de lógica de programação. *Anais do EATI - Encontro Anual de Tecnologia da Informação e Semana Acadêmica de Tecnologia da Informação*, v. 5, n. 1, p. 63–70, 2015. Citado na página 5.

MADEINWEB. *Qual é a diferença entre o aplicativo e o web app?* 2020. Disponível em: <<https://madeinweb.com.br/qual-e-a-diferenca-entre-o-aplicativo-e-o-web-app/>>. Acesso em: 16 nov 2022. Citado na página 4.

MAHARANA, N. *Cross Platform Mobile Apps and Its Pros and Cons*. 2017. Acesso em: 20 mai 2022. Citado na página 4.

MEC. *Plataforma Nilo Peçanha*. 2021. Disponível em: <<http://plataformanilopeçanha.mec.gov.br/>>. Acesso em: 05 mai 2022. Citado na página 2.

MIMO, M. G. *Mimo - Learning to code can be easy and fun*. 2022. Disponível em: <<https://getmimo.com/>>. Acesso em: 20 mai 2022. Citado na página 7.

PROTALINSKI, E. *Google says 500,000 developers use Flutter monthly, outlines release process and versioning changes*. 2022. Disponível em: <<https://venturebeat.com/2020/04/22/google-500000-developers-flutter-release-process-versioning-changes/>>. Acesso em: 28 mai 2022. Citado na página 5.

STATCOUNTER. *Mobile Operating System Market Share Brazil*. 2021. Disponível em: <<https://gs.statcounter.com/os-market-share/mobile/brazil/#monthly-202101-202112-bar>>. Acesso em: 20 mai 2022. Citado na página 4.

TOLOMEI, B. V. *A Gamificação como Estratégia de Engajamento e Motivação na Educação*. 2017. Disponível em: <<https://doi.org/10.18264/eadf.v7i2.440>>. Acesso em: 05 jun 2022. Citado 2 vezes nas páginas 5 e 6.

VIANNA, M. et al. *Gamification, Inc. - Como reinventar empresas a partir de jogos*. [S.l.]: MJV Press, 2013. Citado na página 5.

WERNECK, V. *6 aplicativos para aprender programação*. 2022. Disponível em: <<https://tecnoblog.net/responde/6-aplicativos-para-aprender-programacao/>>. Acesso em: 20 mai 2022. Citado na página 7.

ZACARIAS, R. O.; MELLO, D. R. B. Metodologias de ensino de Lógica de programação e algoritmos em cursos de graduação. *Revista Interdisciplinar do Pensamento Científico*, v. 5, n. 2, p. 29–44, 12 2019. Citado na página 5.