

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA DE CIÊNCIAS EXATAS E DA COMPUTAÇÃO
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO



**DESENVOLVIMENTO DE UM APLICATIVO PARA AUXILIAR NA PREPARAÇÃO
DE ALUNOS DE ENGENHARIA DE COMPUTAÇÃO PARA O ENADE**

WARLEY ALVES FERNANDES

GOIÂNIA
2020

WARLEY ALVES FERNANDES

**DESENVOLVIMENTO DE UM APLICATIVO PARA AUXILIAR NA PREPARAÇÃO
DE ALUNOS DE ENGENHARIA DE COMPUTAÇÃO PARA O ENADE**

Trabalho de Conclusão de Curso apresentado à
Escola de Ciências Exatas e da Computação, da
Pontifícia Universidade Católica de Goiás, como
parte dos requisitos para a obtenção do título de
Bacharel em Engenharia de Computação.

Orientador(a):

Profa. Ma. Ludmilla Reis Pinheiro dos Santos

Banca examinadora:

Prof. Me. Eugênio Júlio Messala Cândido de
Carvalho

Prof. Me. Geraldo Valeriano Ribeiro

GOIÂNIA
2020

WARLEY ALVES FERNANDES

**DESENVOLVIMENTO DE UM APLICATIVO PARA AUXILIAR NA PREPARAÇÃO
DE ALUNOS DE COMPUTAÇÃO PARA O ENADE**

Trabalho de Conclusão de Curso aprovado em sua forma final pela Escola de Ciências Exatas e da Computação, da Pontifícia Universidade Católica de Goiás, para obtenção do título de Bacharel em Engenharia de Computação, em ____/____/____.

Orientador(a): Ma. Ludmilla Reis Pinheiro dos Santos

Profa. Ma. Ludmilla Reis Pinheiro dos Santos
Coordenador(a) de Trabalho de Conclusão de Curso

GOIÂNIA
2020

A Deus pela minha vida e oportunidades.

Aos meus familiares.

À minha mãe Deuseni Pereira Alves, pelo apoio.

AGRADECIMENTOS

À professora Ludmilla Reis Pinheiro dos Santos, orientador acadêmico, pelo apoio e confiança depositada.

À Coordenação da Escola de Ciências Exatas e da Computação, da Pontifícia Universidade Católica de Goiás por ajudar, de forma inestimável, na execução deste projeto.

Aos meus colegas Lucas Rodrigues Rezende e Paulo Henrique Pereira Oliveira pelas discussões técnicas e inestimáveis sugestões.

A todos que direta ou indiretamente colaboraram para materialização deste trabalho.

“Independentemente das circunstâncias, devemos
ser sempre humildes, recatados e despidos de
orgulho.”

Dalai Lama

RESUMO

Este trabalho apresenta o desenvolvimento e a documentação de um aplicativo para auxiliar os estudantes do curso de Engenharia de Computação, participantes do Enade, a se prepararem para a prova. O aplicativo foi desenvolvido no software Android Studio, utilizando o framework Flutter. A linguagem de programação Dart foi utilizada para o desenvolvimento da aplicação, que dispõe de 30 questões para serem respondidas pelos usuários. O aplicativo foi desenvolvido para aplicar os conceitos de gamificação. Os usuários têm acesso ao seu progresso no aplicativo e o sistema gera uma pontuação a cada questionário respondido. A partir da pontuação gerada o ranking é atualizado e disponibilizado para consulta dos usuários. Foi utilizado a ferramenta Firebase para armazenamento de questões, informações das contas dos usuários e dados do ranking de usuários.

Palavras-Chave: Gamificação, Enade, Flutter, Firebase, Dart.

ABSTRACT

This work presents the development and documentation of an application to help students of the Computer Engineering course, participating in Enade, to prepare for the exam. The application was developed in the Android Studio software, using the Flutter framework. The Dart programming language was used to develop the application, which has 30 questions to be answered by users. The application was developed to apply the concepts of gamification. Users have access to their progress in the application and the system generates a score for each questionnaire answered. Based on the score generated, the ranking is updated and made available for users to consult. The Firebase tool was used to store questions, user account information and user ranking data.

Keywords: *Gamification, Enade, Flutter, Firebase, Dart.*

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama de casos de uso do usuário	31
Figura 2 - Diferentes tipos de login do Firebase Authentication	41
Figura 3 - Exemplo de documentos agrupados em uma coleção	42
Figura 4 - Visão geral do Android Studio	43
Figura 5 - Tela inicial do aplicativo	44
Figura 6 - Tela de login	45
Figura 7 - Tela de seleção de conta	45
Figura 8 - Tela principal	46
Figura 9 - Menu de navegação	47
Figura 10 - Tela de inserção de tempo	48
Figura 11 - Tela de exceção de tempo 1	48
Figura 12 - Tela de exceção de tempo 2	49
Figura 13 - Tela de inserção da quantidade de questões	50
Figura 14 - Tela de exceção de quantidade 1	50
Figura 15 - Tela de exceção de quantidade 2	51
Figura 16 - Tela de ranking	52
Figura 17 - Tela de questões	53
Figura 18 - Tela de questão correta	53
Figura 19 - Tela de questão incorreta	54
Figura 20 - Tela de resultado	55
Figura 21 - Tela de relatório	56

LISTA DE QUADROS

Quadro 1 – Requisitos de Usuários	21
Quadro 2 – RF001: Identificar Usuário	22
Quadro 3 – RF002: Armazenar os dados	22
Quadro 4 – RF003: Escolher quantidade de questões	23
Quadro 5 – RF004: Escolher o tempo a ser utilizado	23
Quadro 6 - RF005: Sortear questões a serem respondidas.	24
Quadro 7 – RF006: Gerar Relatório	24
Quadro 8 – RF007: Verificar quantidade de erros.	25
Quadro 9 – RF008: Atribuir pontuação ao questionário	25
Quadro 10 – RF009: Disponibilizar ranking	26
Quadro 11 – RF010: Atualizar ranking	26
Quadro 12 – RF011: Bloquear avanço de questões em branco.	27
Quadro 13 – RQ001: Plataformas	27
Quadro 14 – RQ002: Desenvolvimento	28
Quadro 15 – RQ003: Armazenamento de dados	28
Quadro 16 – RQ004: Disponibilidade	29
Quadro 17 – RQ005: Confiabilidade	29
Quadro 18 – RQ006: Armazenamento	30
Quadro 19 – RQ007: Segurança	30
Quadro 20 – CSU001: Fazer Login	31
Quadro 21 - CSU002: Inserir tempo disponível	32
Quadro 22 - CSU003: Inserir quantidade de questões	33
Quadro 23 - CSU004: Acessar Ranking	34
Quadro 24 - CSU005: Responder Questionário	35
Quadro 25 – CSU006: Atribuir Resultado	36
Quadro 26 – CSU007: Consultar Relatório	36
Quadro 27 - DD1: Dados do usuário	37
Quadro 28 - DD2: Dados das questões	38
Quadro 29 - DD3: Dados do ranking	38

LISTA DE SIGLAS

APP	Aplicativo
ENC	Exame Nacional de Cursos
Enade	Exame Nacional de Desempenho dos Estudantes
GHz	Gigahertz
IDE	Integrated Development Environment ou Ambiente de Desenvolvimento Integrado
INEP	Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira
RAM	<i>Random Access Memory</i> (Memória De Acesso Aleatório)
SDK	Software Development Kit (Kit de Desenvolvimento de Software)
Sinaes	Sistema Nacional de Avaliação da Educação Superior

SUMÁRIO

1. INTRODUÇÃO	14
1.1. Objetivo geral	15
1.2. Objetivos específicos.....	15
1.3. Justificativa	16
1.4. Stakeholders	16
1.5. Visão geral do trabalho.....	17
2. DESCRIÇÃO GERAL DO APLICATIVO.....	18
2.1. Aspecto geral do aplicativo.....	18
2.1.1 Interfaces do aplicativo.....	18
2.1.2. Interfaces do usuário	19
2.1.3. Interfaces de hardware.....	19
2.1.4 Interfaces de software	19
2.2. Características dos usuários	19
2.3. Funcionalidades do aplicativo	20
2.4. Restrições/Limitações	20
3. DOCUMENTAÇÃO DO APLICATIVO	21
3.1. Requisitos de Usuários - Necessidades	21
3.1. Requisitos Funcionais	22
3.2. Requisitos de Qualidade	27
3.3. Diagrama de Casos de Uso	31
3.4. Casos de uso descritivos.....	31
3.5. Dicionário de dados do sistema	37
4. DESENVOLVIMENTO DO APLICATIVO TREINO ENADE	39
4.1. Flutter	39
4.2. Linguagem de programação Dart.....	39
4.3. Firebase	40
4.3.1. Firebase Authentication.....	41
4.3.2. Firebase Cloud Firestore	41
4.4. Android Studio.....	42
5. IMPLEMENTAÇÕES E RESULTADOS	44

5.1. Tela inicial (<i>SplashScreen</i>).....	44
5.2. Tela login.....	44
5.3. Tela principal	46
5.4. Tela de inserção de tempo	47
5.5. Tela de inserção da quantidade de questões.....	49
5.6. Tela de ranking.....	51
5.7. Tela de questões.....	52
5.8. Tela de resultado.....	54
5.9. Tela de relatório.....	55
6. CONSIDERAÇÕES FINAIS	57
6.1. Sugestões para Trabalhos Futuros	57
REFERÊNCIAS.....	58
ANEXO A – TERMO DE AUTORIZAÇÃO DE PUBLICAÇÃO DE PRODUÇÃO ACADÊMICA.....	61

1. INTRODUÇÃO

As gerações atuais, de forma geral, têm dificuldades no que tange à motivação no cenário educacional. Diante deste cenário, um dos desafios da comunidade acadêmica é motivar os alunos a participarem e obterem um bom resultado no Exame Nacional de Desempenho dos Estudantes (Enade), sendo que o conceito de gamificação pode ser uma estratégia motivadora nas escolas e ambientes de aprendizado, pois, de certa forma, os jogos proporcionam a aprendizagem através da diversão (TOLOMEI, 2017).

As instituições superiores do Brasil são submetidas aos exames do Enade de acordo com o calendário trienal. Desde a sua implantação, o Enade é caracterizado como um processo composto de diversos desafios para as Instituições de Ensino Superior, onde destacam-se três deles: (a) se o Projeto Pedagógico do Curso está em consonância com a necessidade do processo de ensino e a prática profissional; (b) sensibilizar os acadêmicos da importância desse momento formativo e avaliativo; e (c) convencer a comunidade acadêmica de que o Enade é um processo de ensino que deve ser refletido e analisado, não apenas um momento específico com data e hora marcada (FIALHO, 2011).

O Enade foi instituído pela Secretaria de Educação Superior através da Lei nº 10.861, de 14 de abril de 2004. O exame substituiu o Exame Nacional de Cursos (ENC), popularmente conhecido como Provão, que vigorou de 1996 a 2003. O Enade integra o Sistema Nacional de Avaliação da Educação Superior (Sinaes), formando o tripé avaliativo composto também pela Avaliação de Cursos de Graduação e pela Avaliação Institucional (INEP, 2019).

O Enade avalia o rendimento dos concluintes dos cursos de graduação em relação aos conteúdos programáticos previstos nas diretrizes curriculares dos cursos, o desenvolvimento de competências e habilidades necessárias ao aprofundamento da formação geral e profissional, e o nível de atualização dos estudantes com relação à realidade brasileira e mundial (INEP, 2019).

O objetivo geral do Enade é avaliar o desempenho dos estudantes em relação aos conteúdos programáticos previstos nas diretrizes curriculares nacionais, as habilidades e competências para atuação profissional, os conhecimentos sobre a realidade brasileira e mundial, bem como sobre outras áreas do conhecimento. A

prova do Enade é formada pelo componente de formação geral, comum a todas as áreas, e pelo componente específico de cada área (INEP, 2019).

Silva, Miranda e Freitas (2017) afirmam que as Instituições de Ensino Superior, visando uma melhor preparação para o Enade, possuem diferentes tipos de ações para engajar os alunos a participarem do exame. Os autores explicam que, enquanto as universidades privadas lançam mão de uma gama de ações, as instituições públicas se limitam a ações de sensibilização, palestras e aulas. Para auxiliar as IES nas ações de preparação para o Enade, este trabalho propõe a criação de um ambiente de estudo que o aluno possa usar fora do ambiente acadêmico, utilizando os conceitos de gamificação para facilitar a dinâmica de aprendizado.

O termo gamificação, do inglês *gamification*, corresponde ao uso de mecanismos de jogos orientados ao objetivo de resolver problemas práticos ou de despertar engajamento entre um público específico (VIANNA, 2013). Os conceitos de gamificação são relativamente novos, derivados da popularidade dos *games* e de todas as possibilidades inerentes de resolver e potencializar aprendizagens em diferentes áreas do conhecimento (TOLOMEI, 2017).

1.1. Objetivo geral

Este trabalho objetiva desenvolver e documentar um aplicativo *mobile* multiplataforma para auxiliar na preparação dos alunos dos cursos de Engenharia de Computação, participantes do Enade, utilizando os conceitos de gamificação.

1.2. Objetivos específicos

- Desenvolver um aplicativo *mobile* multiplataforma utilizando *framework Flutter*;
- Disponibilizar as questões objetivas dos cursos de Engenharia de Computação das edições anteriores do Enade;
- Aplicar os conceitos de gamificação no aplicativo;
- Utilizar um banco de dados online para armazenar os dados do aplicativo;
- Construir uma interface responsiva e amigável que auxilie o usuário na utilização do aplicativo;
- Documentar o desenvolvimento do aplicativo.

1.3. Justificativa

Este trabalho visa a criação de um aplicativo mobile multiplataforma em que os usuários possam testar seus conhecimentos e praticar habilidades específicas relacionadas ao Enade. Para motivar e engajar os usuários na utilização desse, serão utilizados os conceitos de gamificação.

A aplicação dos conceitos de gamificação é o principal diferencial do aplicativo proposto em relação aos aplicativos disponíveis nas lojas virtuais *APP Store* e *Play Store*. A utilização de recompensas e *ranking* de usuários não é uma prática adotada pelos criadores de aplicações de questões do Enade. Em decorrência desse fator, há um baixo interesse dos estudantes na utilização dessas aplicações – o *APP* mais baixado na *Play Store*, disponível para dispositivos Android, possui pouco mais de 5 mil *downloads* (15/11/2020).

De acordo com as avaliações e opiniões dos usuários, os aplicativos disponíveis possuem inconsistências na responsividade de suas telas, funcionando apenas em alguns dispositivos. Além disso, as aplicações disponíveis não são multiplataforma – o aplicativo mais baixado é o APP “Estude para o Enade”, disponível apenas para dispositivos Android.

Este trabalho visa solucionar os problemas de responsividade, funcionando em qualquer dispositivo *mobile* (*smartphones* ou *tablets*), de qualquer tamanho, em qualquer plataforma. Um problema recorrente aos aplicativos disponíveis é a desatualização das questões. Ao final da conclusão deste trabalho, estarão disponíveis questões das provas de 2011, 2014, 2017 e 2019.

1.4. Stakeholders

Segundo Espinha (2020), Stakeholder é todo aquele que tem interesse na gestão de um projeto, como por exemplo: colaboradores, clientes, fornecedores, supervisores, entre outros. Neste trabalho o conjunto de stakeholders é formado por: (a) O autor e desenvolvedor desse trabalho, que tem interesse de entregar um produto de qualidade e que atenda aos requisitos; (b) A orientadora do trabalho, com o interesse de auxiliar no desenvolvimento e garantir a conclusão do mesmo; (c) Os alunos do curso de Engenharia de Computação participantes do Enade, que tem o interesse de se preparar para obter um bom desempenho na prova; e (d) O professor

Geraldo Valeriano Ribeiro, que contribuiu apresentando requisitos para o desenvolvimento do aplicativo.

1.5. Visão geral do trabalho

Este trabalho está dividido em 5 capítulos. O capítulo 1 apresenta a introdução, os objetivos e a justificativa para o desenvolvimento do Treino Enade.

No capítulo 2 são abordadas as características e funcionalidades do sistema, além das características dos usuários. Os itens abordados são de suma importância para a descrição dos requisitos funcionais e não-funcionais do aplicativo.

No capítulo 3 está disponível a documentação utilizada para o desenvolvimento do aplicativo: requisitos de usuários, requisitos funcionais, requisitos de qualidade, diagramas de casos de uso e os casos de uso descritivos.

No capítulo 4 são apresentados os produtos de software e de infraestrutura utilizados para o desenvolvimento do aplicativo: *framework Flutter*, linguagem de programação *Dart*, *Firebase Autentication*, *Cloud Firestore* e Android Studio.

No capítulo 5 estão apresentados os resultados do desenvolvimento do projeto.

No capítulo 6 estão as considerações finais e sugestões para trabalhos futuros.

2. DESCRIÇÃO GERAL DO APLICATIVO

Neste capítulo estão descritas as principais características do aplicativo e dos usuários que irão utilizá-lo.

2.1. Aspecto geral do aplicativo

O aplicativo Treino Enade foi desenvolvido utilizando o *framework* Flutter com o objetivo de ajudar na preparação dos estudantes do curso de Engenharia de Computação para o Enade. O aplicativo permite que os usuários façam o registro no aplicativo com e-mail e senha, para que eles possam ter acesso à conta mesmo com uma eventual troca de dispositivo.

O aplicativo permite que os usuários façam a resolução de questões presentes nas edições anteriores do Enade, calculando a pontuação do usuário ao final de cada questionário.

O aplicativo disponibiliza uma tela de resultados e uma tela de relatório para que o usuário tenha acesso ao seu desempenho de forma detalhada.

2.1.1 Interfaces do aplicativo

O aplicativo se inicia em um *SplashScreen*, tela de abertura, com a logo do aplicativo. A tela de abertura permanece por 5 segundos e a seguir é apresentada a tela de login, onde o usuário pode digitar o e-mail e senha. As demais telas do aplicativo são:

- **Tela de Registro:** O usuário deve acessar essa tela no primeiro acesso ao APP. Na tela de registro o usuário deve digitar o nome, e-mail, senha e a confirmação da senha para realizar o cadastro;
- **Tela principal:** Nesta tela estarão presentes as informações acerca do aplicativo e suas características.
- **Tela de inserção de tempo disponível:** Nesta tela o usuário poderá inserir o tempo disponível para a resolução de questões, com o tempo mínimo de 6 (seis) minutos e máximo de 180 (cento e oitenta) minutos. O aplicativo calcula a quantidade de questões a serem respondidas a partir do tempo disponível inserido pelo usuário. Cada questão tem 6 minutos para ser respondida;

- **Tela de inserção de quantidade de questões:** Nesta tela o usuário poderá inserir a quantidade de questões que deseja responder. A quantidade mínima é de 1 (uma) questão e máxima de 30 (trinta) questões. O aplicativo disponibiliza 6 minutos para a resolução de cada questão;
- **Tela de ranking semanal:** Nesta tela o usuário pode acessar o ranking de usuários, onde estará disponível uma lista com o nome de usuário acompanhado da respectiva pontuação;

2.1.2. Interfaces do usuário

O aplicativo foi desenvolvido para adaptar-se em qualquer tamanho de tela, o que permite o acesso de qualquer dispositivo móvel com acesso à internet.

2.1.3. Interfaces de hardware

Segundo Flutter (2020), os requisitos mínimos de interfaces de hardware para dispositivos Android são:

- Processador de 1.2 GHz;
- Armazenamento interno entre 850 MB e 1.2 GB;
- Mínimo de 512 MB de memória RAM.

Em dispositivos iOS os requisitos mínimos são:

- 512 MB de RAM;
- Processador com 1 GHz.

2.1.4 Interfaces de software

Segundo Flutter (2020), os requisitos de interface de software são:

- Android *Jelly Bean*, v16, 4.1.x ou mais recente;
- iOS 8.0 ou mais recente.

2.2. Características dos usuários

Os usuários que irão utilizar o aplicativo serão preferencialmente alunos do curso de Engenharia de Computação que estiverem cursando 9º ou 10º período. É recomendado que o usuário esteja nos últimos períodos do curso devido à variedade de áreas abrangidas pelas questões. Entretanto, qualquer usuário poderá criar uma conta no Treino Enade.

2.3. Funcionalidades do aplicativo

O aplicativo Treino Enade deverá permitir aos usuários:

- Efetuar login no aplicativo via conta Google
- Responder questões de acordo com o tempo disponível;
- Responder questões de acordo com a quantidade desejada;
- Visualizar ranking;

2.4. Restrições/Limitações

O aplicativo Treino Enade possui as seguintes restrições:

- O aplicativo só pode ser acessado se o dispositivo tiver conexão com a internet, devido ao uso da autenticação e do banco de dados Firebase;
- O aplicativo só pode ser executado no modo retrato do dispositivo móvel.

3. DOCUMENTAÇÃO DO APLICATIVO

Neste capítulo estão dispostas as informações de documentação do aplicativo. Esta documentação inclui: Requisitos de Usuários – Necessidades (RU), Requisitos Funcionais (RF), Requisitos de Qualidade (RQ) e Casos de Uso descritivos (CSU).

3.1. Requisitos de Usuários - Necessidades

Quadro 1 – Requisitos de Usuários

Identificador	Nome	Fonte
RU001	Os usuários deverão fazer o login pela conta do google de modo que todos os dados do usuário no aplicativo estejam relacionados à conta.	Entrevista com o prof. Me. Geraldo Valeriano Ribeiro.
RU002	O sistema deve permitir o usuário escolher a quantidade de questões a serem respondidas.	Entrevista com o prof. Me. Geraldo Valeriano Ribeiro.
RU003	O sistema deve permitir o usuário escolher o tempo disponível para responder as questões.	Entrevista com o prof. Me. Geraldo Valeriano Ribeiro.
RU004	O sistema deve exibir aleatoriamente as questões a serem respondidas.	Entrevista com o prof. Me. Geraldo Valeriano Ribeiro.
RU006	O sistema deve mostrar a quantidade de tempo gasto em cada questão.	Entrevista com o prof. Me. Geraldo Valeriano Ribeiro.
RU007	O sistema deve exibir um relatório com a quantidade de acertos e erros.	Entrevista com o prof. Me. Geraldo Valeriano Ribeiro.
RU008	O sistema deve disponibilizar um ranking de usuários.	Entrevista com o prof. Me. Geraldo Valeriano Ribeiro.

Fonte: Elaborado pelo autor.

3.1. Requisitos Funcionais

Quadro 2 – RF001: Identificar usuário

Identificador RF001	Nome Identificar usuário	
Caso de Uso CSU001.		Autor Warley Alves Fernandes
Descrição O usuário deverá fazer login no aplicativo com a conta do Google.		
Critério de Verificação • Verificar se todos os campos foram preenchidos; • Verificar se as credenciais estão corretas.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 3 – RF002: Armazenar os dados

Identificador RF002	Nome Armazenar os dados	
Caso de Uso CSU002, CSU003.		Autor Warley Alves Fernandes
Descrição Deverão ser armazenados os dados dos usuários, das questões e de ranking.		
Critério de Verificação O usuário deve estar logado.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 4 – RF03: Escolher quantidade de questões.

Identificador RF003	Nome Escolher quantidade de questões.	
Caso de Uso CSU003.		Autor Warley Alves Fernandes
Descrição O usuário pode escolher a quantidade de questões a serem respondidas. O sistema define o tempo necessário a partir da quantidade de questões escolhida.		
Critério de Verificação • Verificar se o campo foi preenchido; • Verificar se é um número válido.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo auto

Quadro 5 – RF004: Escolher o tempo a ser utilizado.

Identificador RF004	Nome Escolher o tempo a ser utilizado.	
Caso de Uso CSU004.		Autor Warley Alves Fernandes
Descrição O usuário pode escolher o tempo disponível para a resolução de questões. O aplicativo calcula a quantidade de questões a partir do tempo escolhido.		
Critério de Verificação • Verificar se o campo foi preenchido; • Verificar se é um número válido.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 6 – RF005: Sortear questões a serem respondidas.

Identificador RF005	Nome Sortear questões a serem respondidas.	
Caso de Uso CSU002; CSU003.		Autor Warley Alves Fernandes
Descrição O aplicativo deve selecionar aleatoriamente as questões a serem respondidas.		
Critério de Verificação • Verificar se a quantidade de questões é válida. • Verificar se o banco de questões possui quantidade de questões maior ou igual à quantidade calculada pelo sistema.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 7 – RF006: Gerar Relatório

Identificador RF006	Nome Gerar Relatório Final	
Caso de Uso CSU002; CSU003.		Autor Warley Alves Fernandes
Descrição O aplicativo deve gerar um relatório ao final de cada sessão de questões respondidas pelo usuário.		
Critério de Verificação • Verificar se todas as questões foram respondidas.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 8 – RF007: Verificar quantidade de erros.

Identificador RF007	Nome Verificar quantidade de erros	
Caso de Uso CSU002; CSU003.		Autor Warley Alves Fernandes
Descrição O aplicativo deve verificar a quantidade de erros ao final de cada questionário respondido pelo usuário.		
Critério de Verificação • Verificar se todas as questões foram respondidas.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 9 – RF008: Atribuir pontuação ao questionário.

Identificador RF008	Nome Atribuir pontuação ao questionário.	
Caso de Uso CSU002; CSU003.		Autor Warley Alves Fernandes
Descrição O aplicativo deve atribuir pontuação ao questionário respondido pelo usuário. Cada questão correta tem o peso de 2 pontos e cada questão errada tem o peso de 0 pontos. Se o usuário não responder uma questão no tempo disponível o sistema atribui a pontuação 0 à questão.		
Critério de Verificação • Verificar se todas as questões foram respondidas.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 10 – RF009: Disponibilizar ranking

Identificador RF009	Nome Disponibilizar ranking	
Caso de Uso CSU004		Autor Warley Alves Fernandes
Descrição O ranking deve ser disponibilizado para consulta dos usuários. O ranking é ordenado pela pontuação acumulada, independente da quantidade de questões respondidas.		
Critério de Verificação • Verificar se há conexão com o banco de dados; • Verificar se há usuários no ranking.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 11 – RF010: Atualizar ranking

Identificador RF010	Nome Atualizar ranking	
Caso de Uso CSU004.		Autor Warley Alves Fernandes
Descrição O ranking deve ser atualizado após o término da resolução de um questionário.		
Critério de Verificação • Verificar se há conexão com o banco de dados.		
Dependência		Prioridade Essencial

Fonte - Elaborado pelo autor.

Quadro 12 – RF011: Bloquear avanço de questões em branco.

Identificador RF011	Nome Bloquear avanço de questões em branco	
Caso de Uso CSU002, CSU003.		Autor Warley Alves Fernandes
Descrição O aplicativo não pode permitir que o usuário troque a questão sem antes tê-la respondido.		
Critério de Verificação • Verificar se a questão não foi respondida.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

3.2. Requisitos de Qualidade

Quadro 13 – RQ001: Plataformas

Identificador RQ001	Nome Plataformas	
Caso de Uso		Autor Warley Alves Fernandes
Descrição O aplicativo deve ser mobile e multiplataforma, podendo ser executado em aparelhos com sistema operacional iOS versão 8.0 e Android versão Jelly Bean, v16, 4.1.x ou mais recente.		
Critério de Verificação • Verificar o comportamento do aplicativo em dispositivos com diferentes tamanhos de tela.		
Dependência		Prioridade

	Essencial
--	-----------

Fonte: Elaborado pelo autor.

Quadro 14 – RQ002: Desenvolvimento

Identificador RQ002	Nome Desenvolvimento	
Caso de Uso		Autor Warley Alves Fernandes
Descrição O sistema deve ser desenvolvido em linguagem Dart com o framework Flutter.		
Critério de Verificação N/A.		
Dependência		Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 15 – RQ003: Armazenamento de dados

Identificador RQ003	Nome Armazenamento de dados	
Caso de Uso		Autor Warley Alves Fernandes
Descrição Os dados do aplicativo devem ser armazenados no banco de dados Firebase, do Google.		
Critério de Verificação N/A.		

Dependência	Prioridade Essencial
--------------------	--------------------------------

Fonte: Elaborado pelo autor.

Quadro 16 – RQ004: Disponibilidade

Identificador RQ004	Nome Disponibilidade	
Caso de Uso		Autor Warley Alves Fernandes
Descrição A base de dados deve estar disponível para que o usuário responda questões sempre que quiser, desde que tenha acesso à internet.		
Critério de Verificação N/A.		
Dependência	Prioridade Essencial	

Fonte: Elaborado pelo autor.

Quadro 17 – RQ005: Confiabilidade

Identificador RQ005	Nome Confiabilidade	
Caso de Uso		Autor Warley Alves Fernandes
Descrição O aplicativo deve informar ao usuário quando ele tentar fazer uma operação não permitida.		
Critério de Verificação		

N/A.	
Dependência	Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 18 – RQ006: Armazenamento

Identificador RQ006	Nome Armazenamento
Caso de Uso	Autor Warley Alves Fernandes
Descrição Os dados devem ser armazenados em nuvem utilizando o banco de dados Firebase, do Google.	
Critério de Verificação N/A.	
Dependência	Prioridade Essencial

Fonte: Elaborado pelo autor.

Quadro 19 – RQ007: Segurança

Identificador RQ007	Nome Segurança
Caso de Uso	Autor Warley Alves Fernandes
Descrição O aplicativo deve ser desenvolvido com técnicas de programação defensiva e com padrões de codificação que reduzem a vulnerabilidade do sistema.	
Critério de Verificação	

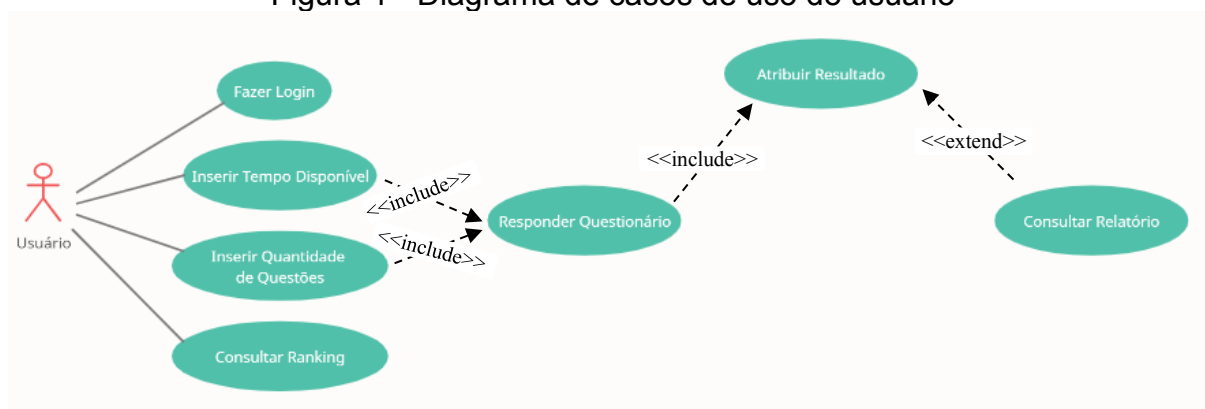
N/A.	
Dependência	Prioridade
	Essencial

Fonte: Elaborado pelo autor.

3.3. Diagrama de Casos de Uso

A Figura 1 apresenta o diagrama de casos de uso do usuário. Nele estão dispostas as ações que usuário pode executar no aplicativo.

Figura 1 - Diagrama de casos de uso do usuário



Fonte: Elaborado pelo autor.

3.4. Casos de uso descritivos

Quadro 20 – CSU001: Fazer Login

Identificador	CSU001
Nome	Fazer Login
Ator(es)	Usuário
Requisitos	RF001
Responsável	Warley Alves Fernandes
Descrição/Resumo	Este caso de uso possibilita ao ator fazer login no aplicativo.
Pré-condições	O ator deve ter usuário e senha registrados no Google.
Pós-condições	

Cenário Principal	1. O ator abre o aplicativo, que apresenta a tela de login; 2. O ator clica na opção 'Entrar com Google'; 3. É exibida uma tela de seleção da conta do Google que o ator deseja autenticar no aplicativo; 4. O ator é autenticado no sistema; 5. O ator é redirecionado para a tela principal do aplicativo; 6. O caso de uso é encerrado.
Cenários Alternativos	2a – Fluxo alternativo – O ator desiste de fazer login. • O ator fecha o aplicativo; • O ator não se autentica no sistema. 3a – Fluxo Alternativo – O usuário já está autenticado no sistema. • O caso de uso avança ao item 5 do cenário principal. • O caso de uso é encerrado.
Cenário de Exceção	2b - O ator digita credenciais erradas • O sistema exibe a mensagem "Falha no Login".
Qualidades	

Fonte: Elaborado pelo autor.

Quadro 21 – CSU002: Inserir tempo disponível

Identificador	CSU002
Nome	Inserir tempo disponível
Ator(es)	Usuário
Requisitos	RF004
00Responsável	Warley Alves Fernandes
Descrição/Resumo	Este caso de uso possibilita ao ator responder questões de acordo com o tempo que ele tem disponível.
Pré-condições	O ator deve estar autenticado no aplicativo.
Pós-condições	
Cenário Principal	1. O ator seleciona a opção "Por Tempo" no menu disponível no canto superior direito; 2. O sistema direciona o ator para a tela de inserção de tempo;

	3. O ator preenche o campo; 4. O ator clica em “Confirmar”; 5. O sistema calcula a quantidade de questões a serem respondidas (cada questão tem 6 minutos para ser respondida). 6. O caso de uso avança ao caso descrito no CSU005.
Cenários Alternativos	3a – Fluxo alternativo - O ator desiste de inserir o tempo. • O ator sai da tela de inserção de tempo; • O sistema exibe a tela principal do aplicativo.
Cenário de Exceção	3a – Fluxo de Exceção - O ator digita um número maior que 180. • O sistema exibe a mensagem “A quantidade máxima é de 180 minutos”; • O ator retorna ao item 3 do cenário principal. 3b – Fluxo de Exceção – O ator digita um número menor que 6. • O sistema exibe a mensagem “A quantidade mínima é de 6 minutos”. • O ator retorna ao item 3 do cenário principal.
Qualidades	

Fonte: Elaborado pelo autor.

Quadro 22 – CSU003: Inserir quantidade de questões

Identificador	CSU003
Nome	Inserir quantidade de questões.
Ator(es)	Usuário
Requisitos	RF003
Responsável	Warley Alves Fernandes
Descrição/Resumo	Este caso de uso possibilita ao ator responder questões de acordo com a quantidade desejada.
Pré-condições	O ator deve estar autenticado no aplicativo.
Pós-condições	

Cenário Principal	1. O ator seleciona a opção “Por Número de Questões” no menu disponível no canto superior direito; 2. O sistema direciona o ator para a tela de inserção da quantidade de questões; 3. O ator preenche o campo; 4. O ator clica em “Confirmar”; 5. O sistema avança ao caso descrito no CSU005.
Cenários Alternativos	3a – Fluxo alternativo - O ator desiste de responder as questões. • O ator sai da tela de inserção do número de questões; • O ator retorna à tela principal do aplicativo.
Cenário de Exceção	3a – Fluxo de Exceção - O ator digita um número maior que 30. • O sistema exibe a mensagem “A quantidade máxima é de 30 questões”; • O ator retorna ao item 5 do cenário principal. 3b – Fluxo de Exceção – O ator digita um número menor que 1. • O sistema exibe a mensagem “A quantidade mínima é de 1 questão”. • O sistema retorna à tela de inserção da quantidade de questões.
Qualidades	

Fonte: Elaborado pelo autor

Quadro 23 – CSU004: Acessar Ranking

Identificador	CSU004
Nome	Acessar ranking
Ator(es)	Usuário
Requisitos	RF009, RF010.
Responsável	Warley Alves Fernandes
Descrição/Resumo	Este caso de uso possibilita ao ator acessar o ranking semanal de usuários.
Pré-condições	O ator deve estar autenticado no aplicativo.

Pós-condições	
Cenário Principal	1. O ator seleciona a opção “Ranking” no menu disponível no canto superior direito; 2. O sistema direciona o ator para a tela de ranking de usuários; 3. O caso de uso se encerra.
Cenários Alternativos	
Cenário de Exceção	
Qualidades	

Fonte: Elaborado pelo autor.

Quadro 24 – CSU005: Responder Questionário

Identificador	CSU005
Nome	Responder Questionário
Ator(es)	Usuário
Requisitos	RF005.
Responsável	Warley Alves Fernandes
Descrição/Resumo	Este caso de uso tem início quando o ator decide acessar as informações do perfil.
Pré-condições	O ator deve executar o fluxo descrito no CSU002 ou o fluxo descrito no CSU003.
Pós-condições	
Cenário Principal	1. O ator responde às questões (cada questão correta tem o peso de 2 pontos). 2. O sistema avança ao caso descrito no CSU006.
Cenários Alternativos	1a – Fluxo alternativo – O ator desiste de responder as questões. • O ator sai da tela de questões; • O sistema exibe a tela principal do aplicativo. 1b – Fluxo alternativo – O ator não responde à questão • O sistema atribui uma resposta incorreta à questão. A resposta está implementada no aplicativo e sempre será diferente da resposta correta.
Cenário de Exceção	

Qualidades	
-------------------	--

Fonte: Elaborado pelo autor.

Quadro 25 – CSU006: Atribuir Resultado

Identificador	CSU006
Nome	Atribuir Resultado
Ator(es)	Usuário
Requisitos	RF007, RF008.
Responsável	Warley Alves Fernandes
Descrição/Resumo	Este caso de uso tem início quando o ator decide acessar as informações do perfil.
Pré-condições	O ator deve executar o fluxo descrito no CSU005.
Pós-condições	
Cenário Principal	1. O sistema calcula o resultado e exibe a tela apresentada na seção 5.8. 2. O caso de uso se encerra.
Cenários Alternativos	1a – Fluxo alternativo – O ator clica na opção ‘Relatório’. • O sistema avança ao caso descrito no CSU007. • O caso de uso é encerrado. 1b – Fluxo alternativo - O ator clica em “Menu Principal”. • O sistema grava os dados do questionário e exibe a tela principal do aplicativo; • O caso de uso é encerrado.
Cenário de Exceção	
Qualidades	

Fonte: Elaborado pelo autor.

Quadro 26 – CSU007: Consultar Relatório

Identificador	CSU 007
Nome	Consultar Relatório
Ator(es)	Usuário

Requisitos	RF006.
Responsável	Warley Alves Fernandes
Descrição/Resumo	Este caso de uso tem início quando o ator decide acessar as informações do perfil.
Pré-condições	O ator deve executar o fluxo descrito no CSU006, fluxo alternativo 1a .
Pós-condições	
Cenário Principal	1. O sistema exibe a tela descrita na seção 5.9. 2. O caso de uso se encerra.
Cenários Alternativos	1a – Fluxo alternativo – O ator clica na opção ‘Voltar’. • O sistema retorna ao caso descrito no CSU006.
Cenário de Exceção	
Qualidades	

Fonte: Elaborado pelo autor.

3.5. Dicionário de dados do sistema

Quadro 27 - DD1: Dados do usuário

Identificador	Descrição			
DD001	Dados do usuário			
Nome	Tamanho	Tipo	Formato	Descrição
Nome	50	String		Nome do usuário fornecido pela autenticação do Google.
E-mail	50	String	usuário@gmail.com	E-mail do usuário fornecido pela autenticação do Google.
Uid	28	String		Uid do usuário fornecido pela autenticação do Google
Pontuacao	8	Int		Pontuação acumulada pelo usuário.

Fonte: Elaborado pelo autor.

Quadro 28 - DD2: Dados das questões

Identificador	Descrição			
DD002	Dados das questões			
Nome	Tamanho	Tipo	Formato	Descrição
Alternativa A	10000	String		Texto da alternativa (a) da questão.
Alternativa B	10000	String		Texto da alternativa (b) da questão.
Alternativa C	10000	String		Texto da alternativa (c) da questão.
Alternativa D	10000	String		Texto da alternativa (d) da questão.
Alternativa E	10000	String		Texto da alternativa (e) da questão.
Texto	10000	String		Texto do enunciado da questão
Alternativa Correta	1	String		Alternativa correta da questão.
Peso	1	Inteiro		Peso atribuído à questão.

Fonte: Elaborado pelo autor.

Quadro 29 - DD3: Dados do ranking

Identificador	Descrição			
DD001	Dados do ranking			
Nome	Tamanho	Tipo	Formato	Descrição
E-mail	50	String	usuário@gmail.com	E-mail do usuário.
Pontos	3	Int		Pontuação acumulada pelo usuário.

Fonte: Elaborado pelo autor.

4. DESENVOLVIMENTO DO APLICATIVO TREINO ENADE

Neste capítulo são apresentadas as tecnologias utilizadas durante o desenvolvimento do aplicativo Treino Enade. A tecnologia empregada para criar as telas foi o framework Flutter, e nas funcionalidades do aplicativo, foi utilizada a linguagem de programação Dart, juntamente com banco de dados Firebase.

4.1. Flutter

O Flutter é um SDK para desenvolvimento de aplicações multiplataforma a partir de um único código com o objetivo de possibilitar a criação de interface nativa e de alta performance (FLUTTER, 2019).

Segundo Abranches (2018), no Flutter o código é compilado em ARM, ou seja, compila o código para cada plataforma agilizando a abertura e o desempenho do aplicativo. Ainda segundo o autor, no Flutter não existe distinção de arquivos para a personalização do layout (JS, HTML, CSS) e tudo é definido em um *Widget* Flutter, facilitando e aumentando a velocidade de desenvolvimento da aplicação.

Os aplicativos desenvolvidos no *framework* Flutter funcionam semelhante a um aplicativo nativo, sendo mais rápidos em comparação com APPs desenvolvidos em outros *frameworks* multiplataforma. Além dos benefícios de performance nativa, o Flutter possui muita flexibilidade para a criação de interfaces personalizadas, animações e facilidades em acessar os recursos do aparelho (PEREIRA, 2019).

É um *framework* de fácil instalação e configuração - ao contrário de seu principal concorrente (*React Native*) - além de produzir menos código, pois o desenvolvimento é feito com um único código na linguagem Dart, poupando tempo e trabalho para os desenvolvedores (CAPPELLI, 2018).

4.2. Linguagem de programação Dart

A linguagem de programação Dart foi apresentada pelo Google em 2011 com objetivo de substituir o JavaScript como principal linguagem embutida nos navegadores (ANDRADE, 2019). Porém, a linguagem evoluiu com o passar dos anos e hoje pode ser considerada uma linguagem multi-paradigma, embora a linguagem apresente fortes estruturas típicas de linguagens orientadas a objeto (GUEDES, 2019).

A linguagem Dart foi desenvolvida para funcionar em todas as plataformas e ter maturidade elevada para desenvolvimento de interfaces de usuário. Suas principais características são: (a) Otimização para IU: visto que ela foi escrita com foco na otimização da interface de usuário; (b) Desenvolvimento Produtivo: *Hot reload* é uma ferramenta muito importante para aumentar a produtividade; e (c) Rápida em todas as plataformas: compila para código de máquina ARM e x64 para celular, *desktop* e *backend*. Além disso compila para JavaScript em sistemas web (PEREIRA, 2019).

4.3. Firebase

O Firebase é um *Backend as a Service (BaaS)* criado em 2004 pelo Google e se tornou uma ferramenta que, em muitos projetos, é a melhor solução. O serviço de armazenamento Firebase é um serviço disponibilizado em que toda a estrutura do *backend* como: configuração de servidor, integração com banco de dados, sistema de *push notification* e outros serviços que fazem parte do *backend* e estão completamente prontos para se integrar com a aplicação (FIREBASE, 2020).

O Firebase conta com uma estrutura pronta de rápida implementação, além de contar com serviço de segurança integrado. O Firebase é recomendado na construção de pequenas aplicações, protótipos ou em produtos simples e baratos (FIREBASE, 2020), que é o caso do aplicativo proposto por este projeto.

Os serviços do Firebase podem ser divididos em três grupos: desenvolvimento, melhoria e crescimento. No grupo de desenvolvimento estão serviços como: (a) *Cloud Firestore*, para armazenamento e sincronização de dados em escala global; (b) *Realtime Database*, para armazenamento e sincronização de dados em tempo real; (c) *Cloud Storage*, para armazenar arquivos; (d) *Authentication*, para autenticação de usuários de modo simples e seguro, entre outros (FIREBASE, 2020)..

No grupo de melhoria estão serviços como: (a) *Crashlytics*, para a correção de erros a partir de relatórios avançados; (b) *Performance Monitoring*, para monitoramento do desempenho do aplicativo; (c) *Test Lab*, para desenvolvimento de aplicativos em dispositivos hospedados pelo Google; e (d) *APP Distribution*, para distribuição de versões de testes dos APPs (FIREBASE, 2020)..

Por fim, o Firebase disponibiliza serviços para expansão dos APPs, como: (a) Google *Analytics* para criar análises de uso do APP; (b) *Cloud Messaging* para enviar mensagens e notificações; e (c) *Remote Config* para modificação do aplicativo sem implantar uma nova versão, entre outros (FIREBASE, 2020)..

4.3.1. Firebase Authentication

O Firebase *Authentication* é um produto do Firebase que fornece serviços de *back-end*, SDKs fáceis de usar e bibliotecas de IU prontas para autenticar usuários no aplicativo. Além disso, o Firebase *Authentication* oferece serviços de autenticação utilizando usuário e senha, números de telefone, conta do Google, Facebook, Twitter, entre outros, conforme ilustrado na Figura 2 (FIREBASE, 2020).

Figura 2 – Diferentes tipos de *login* do Firebase *Authentication*



Fonte: Firebase, 2020.

4.3.2. Firebase Cloud Firestore

O *Cloud Firestore* é um serviço de banco de dados flexível e escalonável para o desenvolvimento de aplicações para dispositivos móveis, Web e servidores a partir do Firebase e do Google *Cloud Platform* (Firebase, 2020). O Firestore é um banco de dados NoSQL armazenado em nuvem que os APPs do iOS, Android, Node.js, Java, Python, Go e da Web podem acessar por meio de diferentes SDKs nativos (FIREBASE, 2020).

Seguindo a sintaxe NoSQL, o Firestore armazena dados na sintaxe JSON. Os dados são divididos em coleções (*collections*) e documentos (*documents*). Cada documento atua como um objeto que possui valores e suas respectivas chaves. Os

valores podem ser do tipo *boolean*, *byte*, *int*, *float*, *String*, *arrays*, *datas*, *null*, entre outros. Uma coleção é o conjunto de documentos, sendo que as coleções não podem ter outras coleções e os documentos não podem ter outros documentos (FERNANDES, 2017). A Figura 3 mostra o exemplo de uma coleção (retângulo amarelo) e seus respectivos documentos (retângulos azuis).

Figura 3 - Exemplo de documentos agrupados em uma coleção.



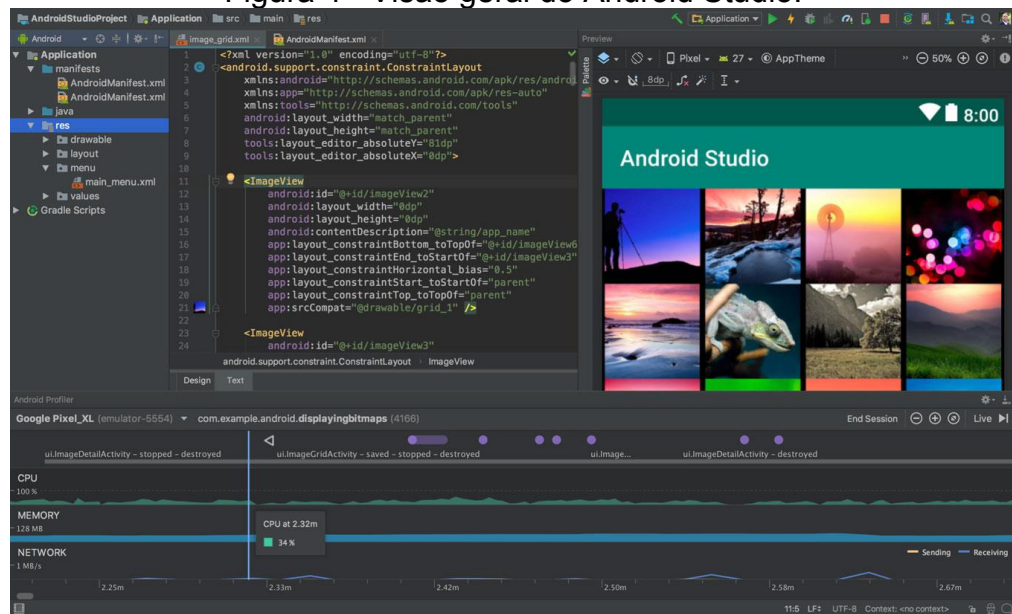
Fonte: Firebase, 2020.

4.4. Android Studio

A IDE Android Studio foi utilizada para desenvolver o aplicativo mobile que pode ser executado nos sistemas operacionais Android e iOS. A Figura 4 apresenta uma visão geral do Android Studio e suas funcionalidades.

A IDE utiliza recursos que facilitam e aumentam a produtividade durante o desenvolvimento dos programas, como: (a) Um sistema de compilação flexível baseado em Gradle; (b) Um emulador rápido com inúmeros recursos; (c) Um ambiente unificado que possibilita o desenvolvimento para todos os dispositivos Android; (d) A aplicação de alterações para enviar alterações de códigos e recursos ao aplicativo em execução sem reiniciar o aplicativo; (e) Modelos de código e integração com GitHub para ajudar a criar recursos comuns de APPs e importar exemplos de código; (f) Ferramentas de lint para detectar problemas de desempenho, usabilidade, compatibilidade com versões, entre outros (ANDROID, 2020).

Figura 4 - Visão geral do Android Studio.



Fonte: Developer Android, 2020.

5. IMPLEMENTAÇÕES E RESULTADOS

Este capítulo apresenta as telas implementadas e os resultados alcançados durante o desenvolvimento do APP.

5.1. Tela inicial (*SplashScreen*)

A Figura 5 apresenta a tela inicial do aplicativo. A SplashScreen é exibida toda vez que o usuário abrir o aplicativo ou efetuar a operação de Logout. A tela permanece visível por 5 segundos e o aplicativo redireciona à tela login descrita na seção 5.2.

Figura 5 - Tela inicial do aplicativo.



Fonte: Elaborado pelo autor.

5.2. Tela login

A Figura 6 apresenta a interface da tela de login do Treino Enade. Na tela de login o usuário deve fazer a autenticação no aplicativo via Google Accounts. Ao clicar no botão 'Entrar com Google' a tela da Figura 7 é exibida e o usuário escolhe a conta que

deseja fazer a autenticação. Após a escolha da conta o usuário é redirecionado para a tela principal descrita no item 5.3

Figura 6 - Tela de login



Fonte: Elaborado pelo autor.

Figura 7 - Tela de seleção de conta

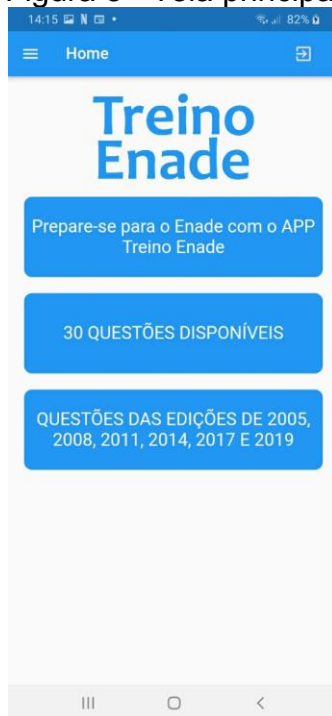


Fonte: Elaborado pelo autor.

5.3. Tela principal

Esta é uma tela de apresentação do aplicativo ao usuário, Figura 8, na qual estão disponíveis a quantidade de questões disponíveis e as edições do Enade que foram retiradas.

Figura 8 - Tela principal

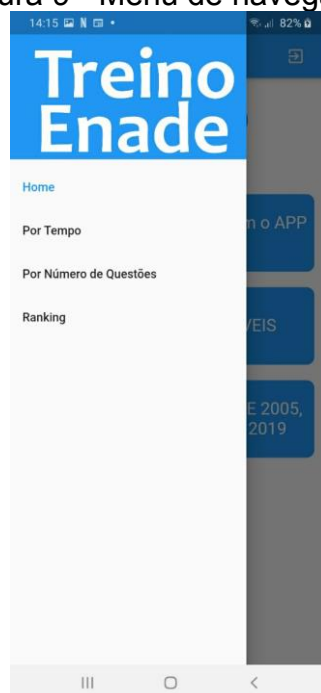


Fonte: Elaborado pelo autor

No canto superior direito encontra-se o ícone de menu, onde é possível navegar entre as funcionalidades do aplicativo, Figura 9. No menu se encontram os seguintes itens:

- Nome do aplicativo na parte superior;
- 'Home' que exibe a tela principal (seção 5.3);
- 'Por Tempo' que exibe a tela de inserção de tempo (seção 5.4);
- 'Por Número de Questões' que exibe a tela de inserção da quantidade de questões (seção 5.5);
- 'Ranking' que exibe o ranking de usuários (seção 5.6);

Figura 9 - Menu de navegação



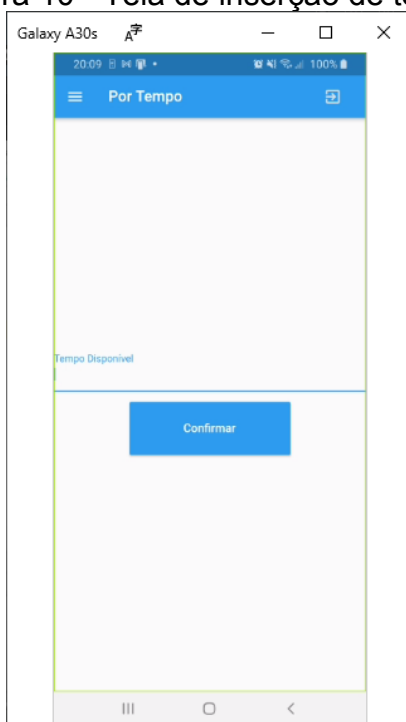
Fonte: Elaborado pelo autor.

5.4. Tela de inserção de tempo

Conforme a Figura 10, nesta tela encontram-se:

- Campo de digitação de tempo disponível, que permite a inserção de 6 a 180 minutos.
- Botão 'Confirmar': quando acionado, calcula a quantidade de questões e inicia o questionário apresentando a tela descrita na seção 5.8.

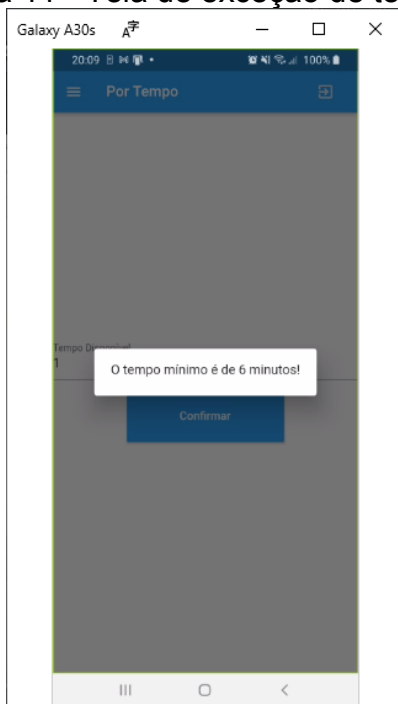
Figura 10 - Tela de inserção de tempo



Fonte: Elaborado pelo autor.

Se o usuário digitar uma quantidade menor que 6 minutos é exibido um alerta conforme a Figura 11.

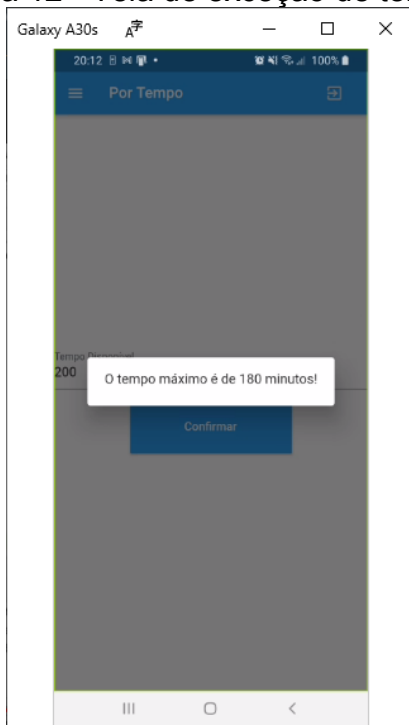
Figura 11 - Tela de exceção de tempo 1



Fonte: Elaborado pelo autor.

Caso o usuário digite uma quantidade maior que 180 o sistema exibe outro alerta, conforme a Figura 12.

Figura 12 - Tela de exceção de tempo 2



Fonte: Elaborado pelo autor.

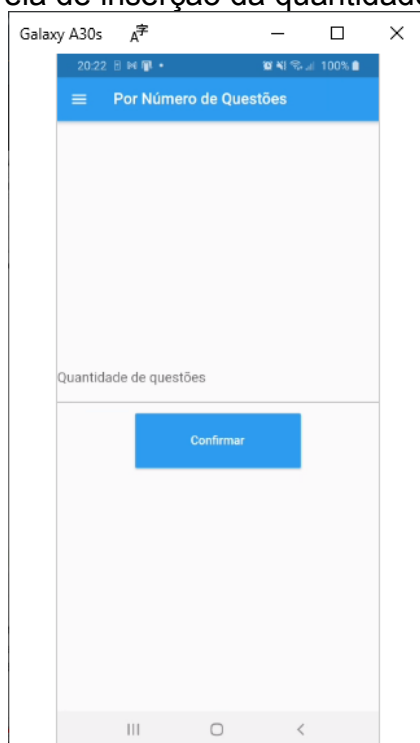
O tempo máximo é 180 minutos devido o sistema considerar que o usuário utiliza em média 6 minutos para a resolução de cada questão.

5.5. Tela de inserção da quantidade de questões

Conforme a Figura 13, nesta tela encontram-se:

- Campo de digitação de tempo disponível, que permite a inserção de 1 a 30 questões;
- Botão 'Confirmar': quando acionado inicia o questionário com a quantidade de questões que foi inserida, apresentando a tela descrita na seção 5.8.

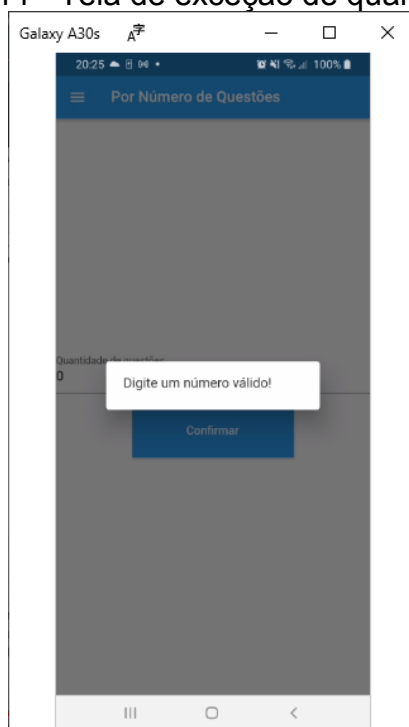
Figura 13 - Tela de inserção da quantidade de questões



Fonte: Elaborado pelo autor.

Se o usuário digitar uma quantidade menor que 1 (uma) questão é exibido um alerta conforme a Figura 14.

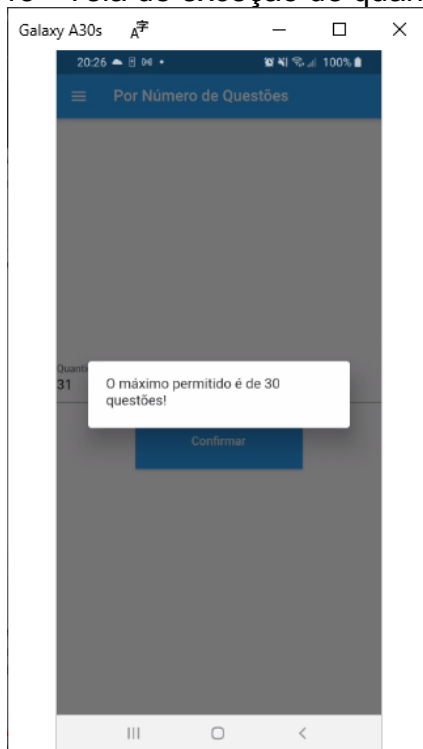
Figura 14 - Tela de exceção de quantidade 1



Fonte: Elaborado pelo autor.

Se o usuário digitar um número maior que 30 (trinta) é exibido um alerta conforme a Figura 15.

Figura 15 - Tela de exceção de quantidade 2



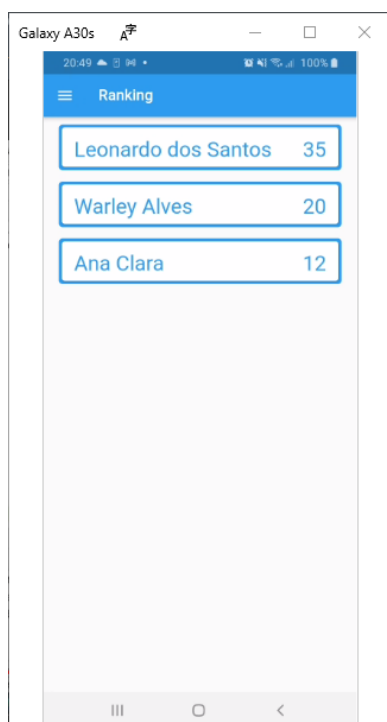
Fonte: Elaborado pelo autor.

A funcionalidade de tempo de questionário não foi implementada.

5.6. Tela de ranking

A tela de ranking apresenta uma lista ordenada dos usuários e suas respectivas pontuações acumuladas, conforme a Figura 16.

Figura 16 - Tela de ranking.



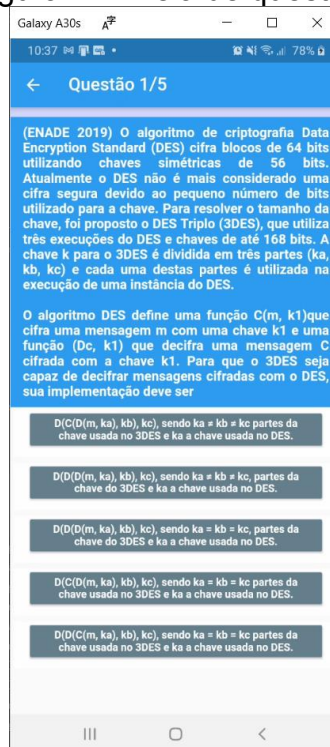
Fonte: Elaborado pelo autor.

5.7. Tela de questões

A tela de questões, Figura 17, possui os seguintes itens:

- Barra com o progresso do questionário, acompanhada de ícone para sair do questionário;
- Enunciado da questão a ser respondida;
- Alternativas da questão.

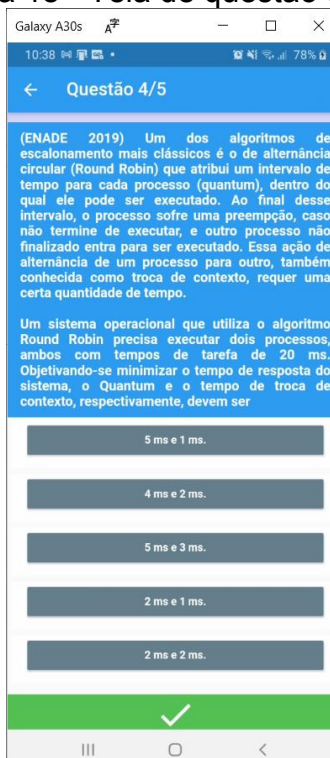
Figura 17 - Tela de questões



Fonte: Elaborado pelo autor.

Quando o usuário selecionar a alternativa correta o sistema exibe uma barra na parte inferior da tela, conforme a Figura 18.

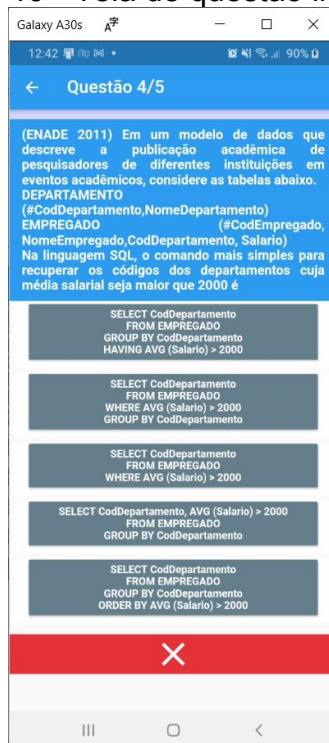
Figura 18 - Tela de questão correta



Fonte: Elaborado pelo autor.

Quando o usuário selecionar uma alternativa incorreta é exibida uma barra na parte inferior da tela, conforme a Figura 19.

Figura 19 - Tela de questão incorreta



Fonte: Elaborado pelo autor.

5.8. Tela de resultado

Na tela de resultado, Figura 20, estão dispostos os seguintes itens:

- Barra de estrela: cada estrela representa 20% da pontuação total. Na Figura 20 foram atribuídas 3 estrelas ao questionário, pois o usuário acertou 60% das questões (6/10);
- Número de questões acertadas acompanhado do número total de questões;
- Pontuação conquistada acompanhada da pontuação máxima do questionário;
- Botão de acesso à tela de relatório: quando o botão é acionado o sistema redireciona o usuário para a tela descrita na seção 5.9;
- Botão de acesso ao menu principal: quando o botão é acionado, o sistema grava os dados do questionário no banco de dados e redireciona o usuário à tela principal descrita na seção 5.3.

Figura 20 - Tela de resultado



Fonte: Elaborado pelo autor.

5.9. Tela de relatório

A tela de relatório, Figura 21, apresenta seguintes informações:

- Informação de êxito na resolução da questão ('Acertou' ou 'Errou!');
- Pontuação conquistada na questão acompanhada do peso considerado da questão;
- Botão 'Voltar' para retornar à tela descrita na seção 5.8.

Figura 21 - Tela de relatório



Fonte: Elaborado pelo autor

6. CONSIDERAÇÕES FINAIS

Este trabalho objetivou desenvolver um aplicativo mobile multiplataforma para auxiliar os alunos do curso de Engenharia de Computação, participantes do Enade, a estudarem para a prova. O Treino Enade utiliza os recursos disponíveis no Android e no Firebase para fazer as operações CRUD e busca de questões. Já a versão iOS do Treino Enade não foi testada devido à falta de um computador com sistema operacional macOS para a execução de um aplicativo iOS.

A IDE utilizada foi o Android Studio, que forneceu todas as bibliotecas usadas no aplicativo. A ferramenta também oferece interface intuitiva que facilita e acelera o processo de desenvolvimento, e integração com outros produtos, como o GitHub e o Firebase.

O Firebase disponibilizou toda a infraestrutura de *backend* para o aplicativo, sendo um serviço de fácil configuração e integração com o aplicativo. Porém, por ser um serviço que objetiva auxiliar no desenvolvimento de protótipos de aplicativos possui muitas limitações que podem comprometer ou limitar o desenvolvimento.

O *framework* Flutter facilita e acelera o desenvolvimento de aplicativos multiplataforma. O Treino Enade foi desenvolvido em um período de 3 meses, incluindo as fases de desenvolvimento e testes.

O trabalho desenvolvido tem enorme potencial para crescimento, pois necessita de novas funcionalidades, porém, este trabalho atingiu o objetivo inicial, que era o desenvolvimento de um aplicativo de questões do Enade.

6.1. Sugestões para Trabalhos Futuros

Como trabalhos futuros sugere-se:

- Aumento do número de questões, incluindo questões de concursos/processos seletivos;
- Filtro de questões por área;
- Compartilhamento do relatório de questões;
- Inclusão de outros cursos;
- Inclusão de questões com imagens no enunciado e nas alternativas;
- Cadastro de dados escolares para filtragem no ranking.

REFERÊNCIAS

ABRANCHES, Junior. **Conhecendo um pouco mais do Flutter**. 2018. Disponível em: <<https://imasters.com.br/framework/conhecendo-um-pouco-mais-flutter>>. Acesso em: 25 maio 2020.

ANDRADE, Kleber de Oliveira. **Introdução a linguagem de programação Dart**. 2019. Disponível em: <<https://medium.com/flutter-comunidade-br/introdução-a-linguagem-de-programação-dart-b098e4e2a41e>>. Acesso em: 25 maio 2020.

ANDROID, Developer. **Conheça o Android Studio**. 2020. Disponível em: <<https://developer.android.com/studio/intro>>. Acesso em: 15 out. 2020.

BRITO, Márcia Regina F. de. **O Sinaes e o Enade: da concepção à implantação**. Avaliação, Campinas, Sorocaba, v. 13, n. 3, p. 841-850, nov. 2008.

CAPPELLI, Elizeu. **Desenvolvimento Híbrido com Flutter: Prós e Contras**. 2018. Disponível em: <<https://medium.com/@devmob/desenvolvimento-híbrido-com-flutter-prós-e-contras-6f3f422c480c>>. Acesso em: 25 maio 2020.

CARVALHO, Mariana. **Banco de Dados**. 2020. Disponível em: <<https://medium.com/sysadminas/banco-de-dados-af36571670ee>>. Acesso em: 15 set. 2020.

COSTA, Luis Antônio. **A "Gamificação" e o poder dos games na transformação da sociedade**. 2017. Disponível em: <http://www.gameblast.com.br/2014/07/a-gamificacao-e-o-poder-dos-games-na.html>

ESPINHA, Roberto Gil. **O que é Stakeholders e qual o papel na Gestão de Projetos**. Disponível em: <<https://artia.com/blog/o-que-e-stakeholders-e-qual-o-papel-nos-projetos/>>. Acesso em: 01 nov. 2020.

FERNANDES, Rosário Pereira. **O que é Cloud Firestore?** 2017. Disponível em: <<https://medium.com/android-dev-moz/firestore-8d225062ffe8>>. Acesso em: 25 out. 2020.

FIALHO, Danielle da Motta Ferreira. **Desafios do Enade para as Instituições de Ensino**. 2011. Disponível em: <<https://administradores.com.br/artigos/desafios-do-enade-para-as-instituicoes-de-ensinohttps://administradores.com.br/artigos/desafios-do-enade-para-as-instituicoes-de-ensino>>. Acesso em: 12 abr. 2020.

FIREBASE, Documentação. **Firestore Authentication**. 2020. Disponível em: <https://firebase.google.com/docs/auth#top_of_page>. Acesso em: 18 out. 2020.

FIREBASE, Documentação. **Cloud Firestore**. 2020. Disponível em: <<https://firebase.google.com/docs/firestore>>. Acesso em: 19 out. 2020.

GRIBOSKI, Claudia Maffini. **O Enade como indutor da qualidade da educação superior**. Est. Aval. Educ., São Paulo, v. 23, n. 53, p. 178-195, set. 2012.

GUEDES, Marylen. **O que é Dart?** 2019. Disponível em: <<https://www.treinaweb.com.br/blog/o-que-e-dart/>>. Acesso em: 20 maio 2020.

GUEDES, Marylen. **SQL vs NoSQL qual usar**. Disponível em: <<https://www.treinaweb.com.br/blog/sql-vs-nosql-qual-usar/>>. Acesso em: 23 ago. 2020.

INEP. **Enade: Histórico**. 2019. Disponível em: <<http://portal.inep.gov.br/web/guest/educacao-superior/enade/historico>>. Acesso em: 17 maio 2020.

INEP. **Exame Nacional de Desempenho dos Estudantes (Enade)**. 2019. Disponível em: <<http://portal.inep.gov.br/enade>>. Acesso em: 30 mar. 2020.

INEP. **Publicadas as diretrizes das provas de formação geral e específicas das áreas avaliadas em 2019**. 2019. Disponível em: <http://portal.inep.gov.br/artigo/-/asset_publisher/B4AQV9zFY7Bv/content/publicadas-as-diretrizes-das-provas-de-formacao-geral-e-especificas-das-areas-avaliadas-em-2019/21206>. Acesso em: 02 abr. 2020.

LEANDRO. **Conheça o iOS 8 para Desenvolvedores**. 2015. Disponível em: <<https://www.devmedia.com.br/conheca-o-ios-8-para-desenvolvedores/32351>>. Acesso em: 30 out. 2020.

LEITE, Alexandre Canosa; REIS, Helena Macedo dos. **COMPARATIVO ENTRE SISTEMAS OPERACIONAIS MÓVEIS – ANDROID X IOS**. In: IV SIMTEC, 4., 2012, Taquaritinga. 2012.

LIMA, Welton Dias. **Android e a influência do Sistema Operacional Linux**. TECNOLOGIAS EM PROJEÇÃO, v. 8, n. 1, p. 100-111, 2017. Disponível em: <<http://revista.faculdadeprojecao.edu.br/index.php/Projecao4/article/view/829>>. Acesso em: 23 ago. 2020.

MICHELS, Tatiana Aparecida; FERREIRA, Aline Maiara Saldanha; PAZ, Daiane de Padula. **A Gamificação como Estratégia no Processo de Ensino**. Revista Mundi Engenharia: Tecnologia e Gestão, Paranaguá, v. 4, n. 1, mar. 2019.

ORACLE. **Banco de Dados: o que é um banco de dados**. 2020. Disponível em: <<https://www.oracle.com/br/database/what-is-database.html>>. Acesso em: 10 out. 2020.

PEREIRA, Manacés. **Flutter: porque você deveria apostar nesta tecnologia**. 2019. Disponível em: <<https://medium.com/tableless/flutter-porque-você-deveria-apostar-nesta-tecnologia-94a510fffd18>>. Acesso em: 25 ago. 2020.

PEREIRA, Manacés. **Iniciando com Flutter**. 2019. Disponível em: <<https://medium.com/manacespereira/iniciando-com-flutter-aba6158635e2>>. Acesso em: 20 set. 2020.

PISSAIA, Luis Felipe; MARTINS, Silvana Neumann; REHFELDT, Márcia Jussara Hepp; COSTA, Arlete Eli Kunz da. **Exame Nacional dos Estudantes - ENADE e o desenvolvimento de competências no ensino superior**. Research, Society and Development, v. 2, n. 7, p. 01-13, jan. 2018.

SILVA, T. D.; MIRANDA, G. J.; FREITAS, S. C. **Ações institucionais preparatórias para o ENADE nos cursos de ciências contábeis**. Revista Universo Contábil, ISSN 1809-3337, FURB, Blumenau, v. 13, n. 1, p. 65-84, jan./mar. 2017.

TOLOMEI, Bianca Vargas. **A Gamificação como Estratégia de Engajamento e Motivação na Educação**. Ead em Foco, Niterói, v. 2, n. 7, p. 145-156, 2017.

VIANNA Ysmar et al. **Gamification: como reinventar empresas a partir de jogos**. 1 Ed. Rio de Janeiro: MJVPress, 2013.

WAINER, Jacques; MELGUIZO, Tatiana. **Políticas de inclusão no ensino superior: avaliação do desempenho dos alunos baseado no Enade de 2012 a 2014**. Educ. Pesqui., São Paulo, v. 44, 12 jan. 2017.

ANEXO A – TERMO DE AUTORIZAÇÃO DE PUBLICAÇÃO DE PRODUÇÃO ACADÊMICA



**PUC
GOIÁS**

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
GABINETE DO REITOR

Av. Universitária, 1069 • Setor Universitário
Caixa Postal 86 • CEP 74605-010
Goiânia • Goiás • Brasil
Fone: (62) 3946.1000
www.pucgoias.edu.br • reitoria@pucgoias.edu.br

RESOLUÇÃO n° 038/2020 – CEPE

ANEXO I

APÊNDICE ao TCC

Termo de autorização de publicação de produção acadêmica

O(A) estudante Wendry Alves Fernandes
do Curso de Engenharia de Computação matrícula 20152003300488,
telefone: 62996318565 e-mail emp.walry@gmail.com na qualidade de titular dos
direitos autorais, em consonância com a Lei n° 9.610/98 (Lei dos Direitos do autor),
autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás) a disponibilizar o
Trabalho de Conclusão de Curso intitulado
Desenvolvimento de um aplicativo para auxiliar na preparação dos estudantes
de Eng. de Computação gratuitamente, sem ressarcimento dos direitos autorais, por 5
(cinco) anos, conforme permissões do documento, em meio eletrônico, na rede mundial
de computadores, no formato especificado (Texto (PDF); Imagem (GIF ou JPEG); Som
(WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da
área; para fins de leitura e/ou impressão pela internet, a título de divulgação da
produção científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 18 de Fevereiro de 2021.

Assinatura do(s) autor(es): Wendry Alves Fernandes

Nome completo do autor: Wendry Alves Fernandes

Assinatura do professor-orientador: Ludmilla R. P. dos Santos

Nome completo do professor-orientador: Ludmilla Reis Pinheiro dos Santos