

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO  
GRANDE DO SUL - CAMPUS CANOAS**

**CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE  
SISTEMAS**

**DOJO DO SPLINTER: SISTEMA HÍBRIDO PARA ACADEMIAS DE ARTES  
MARCIAIS**

**NÍKOLAS TELVI NAHRA DAROIT**

**CANOAS, JUNHO DE 2022**

**NÍKOLAS TELVI NAHRA DAROIT**

**DOJO DO SPLINTER: SISTEMA HÍBRIDO PARA ACADEMIAS DE ARTES  
MARCIAIS**

Dojo do Splinter: sistema híbrido para academias de artes marciais apresentado junto ao Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul, como requisito parcial à obtenção do título Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: prof. Ígor Lorenzato Almeida



**Ministério da Educação**  
**Secretaria de Educação Profissional, Científica e Tecnológica**  
**Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul**  
**Campus Canoas**

**ATA DE DEFESA PÚBLICA DO TRABALHO DE CONCLUSÃO DE CURSO**

Aos 04 dias do mês de agosto de 2022, às 14 horas e 30 minutos, em sessão pública realizada através da plataforma Google Meet, na presença da Banca Examinadora presidida pelo Professor:

Ígor Lorenzato Almeida e

composta pelos examinadores:

1. Profa. Denise Regina Pechmann,

2. Prof. Gustavo Neuberger,

o aluno **Nikolas Telvi Nahra Daroit**

apresentou o Trabalho de Conclusão de Curso intitulado:

***Dojo do Splinter: Aplicativo Móvel para Academias de Luta***

como requisito curricular indispensável para a integralização do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas. Após reunião em sessão reservada, a Banca Examinadora deliberou e decidiu pela **APROVAÇÃO** do referido trabalho, divulgando o resultado formalmente ao aluno e demais presentes e eu, na qualidade de Presidente da Banca, lavrei a presente ata que será assinada por mim, pelos demais examinadores e pelo aluno.

Ígor Lorenzato  
Almeida

Digitally signed by Ígor Lorenzato Almeida  
DN: C=BR, OU=Campus Canoas, O=IFRS, CN=Ígor  
Lorenzato Almeida, E=igor.almeida@canoes.ifrs.edu.br  
Reason: Eu sou o autor deste documento.  
Location: Porto Alegre, RS  
Date: 2022.08.04 15:38:58  
Foxit Reader Version: 9.6.0

Presidente da Banca Examinadora

Denise Regina  
Pechmann

Assinado digitalmente por Denise Regina Pechmann  
DN: C=BR, OU=Campus Canoas, O=IFRS, CN=Denise  
Regina Pechmann, E=denise.pechmann@canoes.ifrs.edu.br  
Reason: Eu sou o autor deste documento.  
Localização: sua localização de assinatura aqui  
Date: 2022.08.12 12:44:45-0300  
Foxit Reader Versão: 10.1.1

Examinador 01

Gustavo  
Neuberger

Assinado digitalmente por Gustavo Neuberger  
DN: C=BR, OU=IFRS Campus Canoas,  
O=Gustavo Neuberger, CN=Gustavo Neuberger,  
E=gustavo.neuberger@canoes.ifrs.edu.br  
Reason: Eu sou o autor deste documento  
Localização: Canoas (RS)  
Date: 2022.08.12 10:54:02  
Foxit Reader Versão: 9.7.2

Examinador 02

Nikolas Telvi Nahra Daroit

DocuSigned by:

40E7E86010D7476...

Aluno

## RESUMO

Este trabalho desenvolve de um sistema web e mobile (Android), onde seja possível realizar o cadastro como aluno, ou instrutor, e ingressar em uma turma de artes marciais. Academias de artes marciais são espaços destinados para a prática de artes marciais, geralmente constituídos de um gestor, um ou mais instrutor, e alunos. Academias de artes marciais de pequeno porte, geralmente utilizam registros físicos e pecam no acompanhamento de tais fichas. Pois, como normalmente os instrutores são os próprios gestores, colegas de faculdade, ou pessoas individuais que decidem abrir estes espaços, não possuem especialização na área de administração, deixando a desejar nesta parte do negócio. Para auxiliar estas organizações, este trabalho desenvolve um sistema, tanto para dispositivos móveis, quanto para web, permitindo o acesso via multiplataforma para que, os instrutores ou gestores responsáveis pela organização, não dependam de fichas cadastrais físicas. Foi utilizado as seguintes tecnologias para o desenvolvimento desta solução: Flutter, um framework para a linguagem Dart que provê as ferramentas necessárias para a criação do sistema Android e Web; Firebase, uma coleção de ferramentas que permitem realizar a hospedagem da aplicação web, bem como dos bancos de dados não relacionais utilizados no projeto.

**Palavras-chave:** Arte marcial; Flutter; Sistema híbrido.

## **ABSTRACT**

This work proposes the development of a web and mobile application (Android), where it is possible to register as a student or instructor and join a martial arts class. Martial arts academies are spaces for the practice of martial arts, usually consisting of a manager, one or more instructors, and students. Smaller martial arts academies generally use physical records, and fail to keep track of such records. Because, as the instructors are usually the managers themselves, college friends, or individuals who decide to open these spaces, they don't have specialization in the administration area, leaving this part of the business unsatisfactory. To help these organizations, this work proposes to develop an application, both for mobile devices and for the web, allowing access via multi-platform so that the instructors or managers responsible for the organization do not depend on physical registration forms. The following technologies were used to develop this solution: Flutter, a framework for the Dart language that provides the necessary tools for creating the Android and Web application; Firebase, a collection of tools that allow hosting the web application, as well as the non-relational databases used in the project.

**Keywords:** Martial art; Flutter; Hybrid system.

## LISTA DE ABREVIATURAS E SIGLAS

API	<i>Applications Programming Interface</i>
APP	Aplicativo
BSD	<i>Berkeley Software Distribution</i>
CI	<i>Continuous Integration</i>
CD	<i>Continuous Delivery</i>
ER	Entidade Relacionamento
GB	<i>Gigabyte</i>
JSON	<i>JavaScript Object Notation</i>
NOSQL	<i>Not Only SQL</i>
SQL	<i>Structured Query Language</i>
UI	<i>User Interface</i>
UID	<i>Unique User Id</i>
UX	<i>User Experience</i>
UML	<i>Unified Modeling Language</i>
VM	<i>Virtual Machine</i>

## LISTA DE TABELAS

Tabela 1: Resultados do estado da arte.....	19
---	----

## LISTA DE FIGURAS E TABELAS

Figura 1: Interface do Aplicativo Idojo Em Versão Web E Mobile .....	16
Figura 2: Painel De Controle Do Dojo Web .....	17
Figura 3: Painel De Controle Do Iblackbelt.....	18
Figura 4: Esquema Sobre Flutter .....	23
Figura 5: Benchmark de linguagem Dart vs Node js.....	25
Figura 6: Exemplo De Widget Em Dart .....	26
Figura 7: Exemplo De Curried Addition Em Dart .....	27
Figura 8: Serviços Em Firebase.....	28
Figura 9: Biblioteca Do Firebase - Pagamento .....	29
Figura 10: Armazenamento De Dados De Pagamento - Firebase .....	30
Figura 11: Tratamento De Erros .....	31
Figura 12: Diagrama De Caso De Uso .....	36
Figura 13: Armazenamento De Token .....	37
Figura 14: Armazenamento De Dados De Alunos .....	38
Figura 15: Armazenamento De Dados De Instrutores .....	39
Figura 16: Diagrama De Classes .....	40
Figura 17: Tela De Firebase Authentication .....	42
Figura 18: Tela De Login De Cadastro .....	43
Figura 19: Tela Principal De Instrutor .....	44
Figura 20: Tela De Listagem E Cadastro De Turmas .....	45
Figura 21: Tela Principal Da Turma.....	46
Figura 22: Telas De Listagem E Cadastro De Turma .....	46
Figura 23: Telas De Calendário De Chamada E Chamada Diária .....	47
Figura 24: Tela De Alunos Cadastrados Ou Alunos Pendentes .....	48
Figura 25: Tela De Consulta De Dados .....	49

Figura 26: Tela De Calendário De Eventos, Listagem E Cadastro De Novos Eventos .....	50
Figura 27: Tela De Consulta De Alunos .....	51
Figura 28: Tela Principal De Alunos .....	52
Figura 29: Tela De Listagem De Turma .....	53
Figura 30: Tela De Turmas Aguardando Mensais .....	54
Figura 31: Diagrama De Sequência .....	55
Figura 32: Lista De Pagamentos Mensais .....	55
Figura 33: Dados De Matrícula .....	56
Figura 34: Calendário e detalhes de eventos.....	57
Figura 35: Arquivo de workflow para o deploy web no Github Actions.....	58
Figura 36: Exemplo de execuções de workflow no Github Actions .....	59
Figura 37: Passo a passo da execução do workflow Deploy Web.....	59
Figura 39: Gráfico gerado pelas respostas das perguntas referente aos objetivos requisitados.....	61
Figura 40: Gráfico gerado pelas respostas da pergunta 1.....	63
Figura 41: Gráfico gerado pelas respostas da pergunta 2.....	64
Figura 42: Gráfico gerado pelas respostas da pergunta 3 .....	65
Figura 43: Gráfico gerado pelas respostas da pergunta 4 .....	66
Figura 44: Gráfico gerado pelas respostas da pergunta 5 .....	67
Figura 45: Gráfico gerado pelas respostas da pergunta 6 .....	68
Figura 46: Gráfico gerado pelas respostas da pergunta 7 .....	69
Figura 47: Gráfico gerado pelas respostas da pergunta 8 .....	70
Figura 48: Gráfico gerado pelas respostas da pergunta 9 .....	71
Figura 49: Gráfico gerado pelas respostas da pergunta 10 .....	72
Figura 50: Gráfico gerado pelas respostas da pergunta 11 .....	73
Figura 51: Detalhes nos cards das turmas matriculadas .....	75
Figura 52: Valor da mensalidade na lista de pagamentos .....	76

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>12</b>
1.1. MOTIVAÇÃO .....	13
1.2.1. Objetivo Geral .....	13
1.2.2. Objetivos Específicos.....	13
<b>2. ESTADO DA ARTE.....</b>	<b>15</b>
2.1. IDOJO .....	15
2.2. DOJOWEB.....	16
2.3. IBLACKBELT .....	17
2.4. COMPARAÇÃO ENTRE OS SISTEMAS .....	18
<b>3. REVISÃO BIBLIOGRÁFICA.....</b>	<b>22</b>
3.1. DISPOSITIVOS MÓVEIS.....	22
3.2. APLICATIVOS MÓVEIS .....	22
3.3. LINGUAGENS DE PROGRAMAÇÃO.....	23
3.3.1. FLUTTER .....	23
3.3.2. 3.2.2DART .....	24
3.3.3. FIREBASE .....	27
<b>4. METODOLOGIA .....</b>	<b>32</b>
<b>5. DESENVOLVIMENTO .....</b>	<b>34</b>
5.1. MODELAGEM.....	34
5.1.1. DOCUMENTO DE VISÃO.....	34

5.1.2.	DIAGRAMA DE CASOS DE USO.....	35
5.2.	O DOJO DO SPLINTER.....	40
5.2.1.	Login e Cadastro.....	41
5.2.2.	Perfil do Instrutor.....	43
5.2.2.1.	Eventos.....	48
5.2.2.2.	Alunos.....	49
5.2.3.	Perfil do Aluno.....	49
5.3.	PIPELINES DE BUILD E DEPLOY.....	55
5.4.	TESTE DE ACEITAÇÃO.....	59
<b>6.</b>	<b>CONSIDERAÇÕES FINAIS.....</b>	<b>76</b>
6.1.	TRABALHOS FUTUROS.....	77
6.1.1.	Verificação e Aprovação do Instrutor cadastrado .....	77
6.1.2.	Tela de relatórios financeiros para o instrutor.....	77
6.1.3.	Alteração da visualização de turmas recomendadas.....	77
6.1.4.	Cadastro de aluno pelo perfil de Instrutor .....	77
<b>7.</b>	<b>REFERÊNCIAS.....</b>	<b>78</b>

## 1. INTRODUÇÃO

A tecnologia da informação vem se tornando cada vez mais uma necessidade no cotidiano das pessoas, através da popularização dos dispositivos móveis, como smartphones e tablets, que vem ocorrendo nos últimos anos. Por consequência, a demanda por aplicativos que possam facilitar as tarefas do dia a dia dos indivíduos aumentou. Este crescimento acarretou no aumento do desenvolvimento de pequenos aplicativos que pudessem oferecer soluções simples para diversas áreas, estas aplicações são denominadas como aplicativos móveis. Em academias de artes marciais existem as seguintes necessidades: Cadastramento das aulas disponibilizadas pela academia; do instrutor; dos alunos; dos pagamentos recebidos pelo disponibilizador de serviços e dos eventos esportivos. No entanto, em pequenas academias este gerenciamento é realizado de forma manual, gerando mão de obra adicional para registros, manutenção e armazenamento destes itens.

Desta forma, o presente trabalho tem como objetivo desenvolver um sistema, composto por um aplicativo móvel e um site responsivo, para auxiliar nas atividades da academia, bem como para os frequentadores do espaço. Tendo em vista que as pessoas têm o hábito de levar seus smartphones para todos os lugares que frequentam e fazem uso de aplicativos para diversos fins, o desenvolvimento de um aplicativo móvel mostra-se prático e essencial para que todos tenham acesso às diversas funcionalidades da academia.

Levando em consideração que as atividades realizadas nas academias são simples, não apresentando lógicas complexas de negócio, a utilização de um aplicativo torna-se uma alternativa viável para proporcionar uma maior praticidade. Dessa forma, servindo de auxílio para realização de atividades administrativas. O foco deste trabalho é propor o desenvolvimento de um aplicativo móvel para academias de artes marciais.

## 1.1. MOTIVAÇÃO

A autonomia e a informalidade no mercado de artes marciais leva à não disponibilização de aplicações que visem este mercado, e, por conta disso, a não utilização de ferramentas que possibilitem o controle financeiro e o gerenciamento de frequência. Além disso, a falta de controle automatizado no âmbito financeiro, além da dificuldade de disponibilizar informações e gerenciar as atividades desenvolvidas ou recursos disponíveis, gerando impacto administrativo na operação. Portanto, os instrutores acabam optando por listas de chamadas e formulários de pagamentos impressos, como pode ser observado no apêndice B, enfrentando dificuldades com a organização de horários e pagamentos, conforme o número de alunos aumenta. O desenvolvimento de um sistema específico para estas academias otimizará o trabalho, centralizando as informações dos alunos, aulas e pagamentos, eliminando os registros em papéis, e mantendo o histórico das atividades realizadas para os instrutores e alunos.

## 1.2. OBJETIVOS

Para o desenvolvimento deste trabalho foram definidos objetivos geral e específicos, que serão apresentados na sequência.

### 1.2.1. Objetivo Geral

Consiste no desenvolvimento de um sistema com interface para dispositivos móveis e navegadores web, direcionado a alunos e instrutores de academias de artes marciais. Ofertando, por conseguinte, um cenário mais prático aos utilizadores do espaço.

### 1.2.2. Objetivos Específicos

Para atingir a construção do sistema foi necessário concluir os seguintes objetivos:

- Realizar análise de requisitos com instrutores de artes marciais;
- Desenvolver uma aplicação híbrida Web/Android usando Flutter;

- Realizar testes de aceitação com alunos e instrutores;
- Disponibilizar a aplicação móvel na loja Play Store;

No presente capítulo, foi realizada apresentação da introdução deste trabalho, bem como a devida motivação para sua execução e os objetivos que o mesmo almejou alcançar durante seu desenvolvimento. Não obstante, detalha-se através de seus tópicos o cenário no qual o projeto encontra seus desafios e razões para desenvolvimento.

## 2. ESTADO DA ARTE

Notoriamente, a tecnologia avança em ritmo acelerado, fomentada por uma indústria voraz de desenvolvimento e aprimoramento de novas ideias. Portanto, a cada dia há um número maior de aplicações surgindo e sendo disponibilizadas no mercado. Não obstante, são inúmeras as ferramentas digitais voltadas para o mercado de saúde física. Afinal, nossa sociedade está cada vez mais preocupada com o bem estar do próprio corpo, seja por vitalidade, qualidade de vida ou estética. Por isso, aplicativos para auxiliar em treinos de musculação, calistênicos, nutrição e artes marciais se popularizaram na internet.

Entre essas ferramentas, algumas populares são iDojo (iDojo, 2022), DojoWeb (DojoWeb , 2022), Tecnofit (Tecnofit , 2022) e iBlackbelt (iDiBlackbelt ojo, 2022) – que se voltam ao auxílio de gestão de academias. Estes sistemas incluem desde tarefas como controle de alunos até outras como geração de relatórios e organização de torneios. Desse modo, essas ferramentas são disponibilizadas das mais variadas formas, por vezes disponíveis apenas como um website, em outros casos como uma plataforma para aplicativos móveis e em outras ocasiões em ambos os formatos.

Para tanto, através da aplicação apresentada neste trabalho, busca-se alcançar soluções que auxiliem na resolução de tarefas que são apresentadas no dia a dia da gestão de uma academia. Como por exemplo:

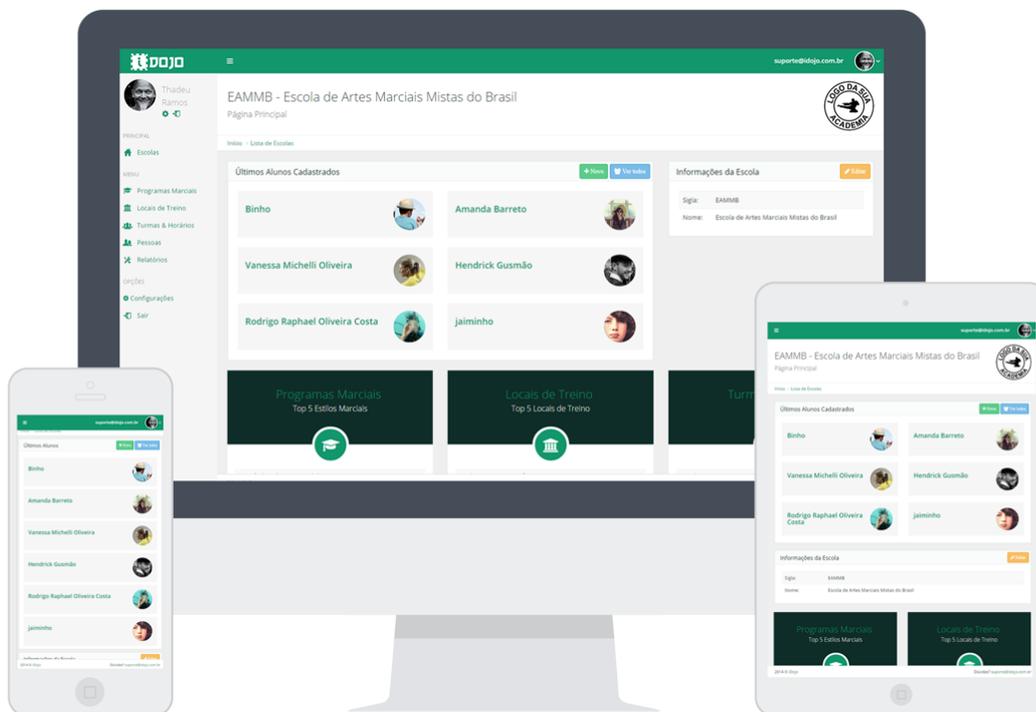
### 2.1. IDOJO

O iDojo é uma plataforma lançada ao final de 2014, que permite, entre suas funcionalidades, o cadastramento de alunos e turmas. Um de seus diferenciais é a personalização do sistema de graduação, no qual é possível registrar os níveis de graduação baseados em cor de faixa para cada arte marcial.

A plataforma caracteriza-se por possuir sistema híbrido, com interface web para acesso de contas destinadas a gestores e instrutores, além de um aplicativo móvel exclusivo para acesso de alunos e pais.

A Figura 1 representa as telas do aplicativo em sua versão *web* para gestores e instrutores. Esta Figura apresenta o menu de gestão de uma escola de artes marciais mistas, com os últimos alunos cadastrados, informações da academia e opções disponíveis aos gestores e instrutores.

Figura 1: Interface do aplicativo iDojo em versão Web e Mobile



Fonte: iDojo

## 2.2. DOJOWEB

O Dojoweb é uma aplicação desenvolvida com intuito de gerir uma academia de artes marciais. A plataforma conta com uma tela inicial onde você pode acompanhar todos os dados mais importantes e a progressão do seu negócio, como mostrado na Figura 2.

Figura 2: Painel de controle do Dojoweb

Fonte: DojoWeb

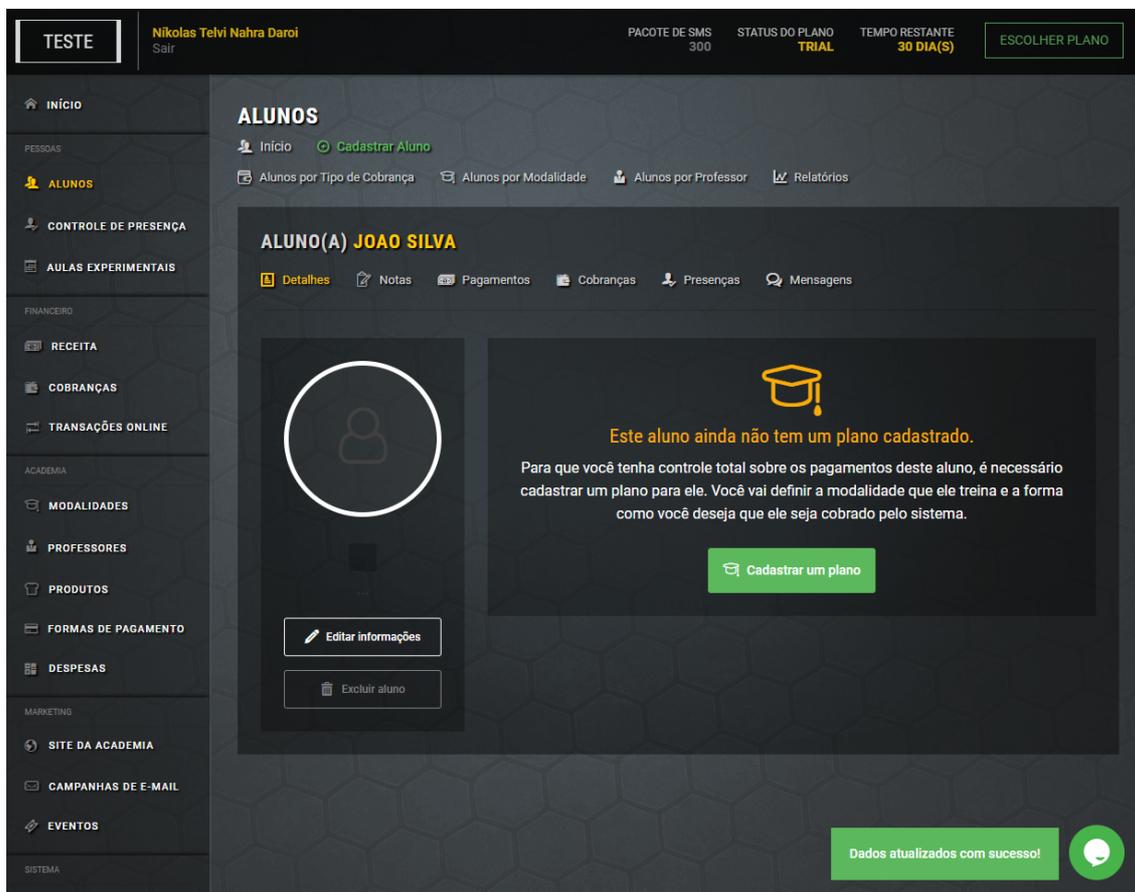
A plataforma demonstra boa organização, sendo estruturada em abas, facilitando desta forma o acesso a menus e ferramentas. Por conta da quantidade de vínculos entre objetos e cadastros, a sua curva de aprendizado pode se tornar um pouco íngreme, tendo em vista a cascada de cadastros necessários para que cada parte do sistema funcione como esperado. Porém, com esta quantidade de camadas, o DojoWeb se mostra uma ferramenta sólida para gerenciar academias quando o número de alunos e turmas aumentam, porém se torna oneroso para academias de pequeno porte.

### 2.3. IBLACKBELT

O iBlackbelt é uma plataforma desenvolvida com o intuito de gerenciar grandes academias, demonstra um design moderno porém simples, com uma interface dividida em menus laterais, e subdividida em categorias, facilitando o acesso às ferramentas. Seu maior diferencial é o sistema de cobrança automático, podendo ser por meio de cadastro de cartão de crédito, boleto ou lembrete de pagamento na academia.

A Figura 3 mostra o aluno cadastrado e as possíveis interações a serem feitas, como o cadastro do plano, edição de informações, ou exibição de detalhes por categoria.

Figura 3: Painel de controle do iBlackBelt



Fonte: Página web do iBlackBelt (<https://iblackbelt.com.br/>)

## 2.4. COMPARAÇÃO ENTRE OS SISTEMAS

A proposta deste trabalho é o desenvolvimento do sistema Dojo do Splinter, que auxilia no gerenciamento de turmas de artes marciais trazendo mais

visibilidade para academias. Foram analisados três sistemas que são utilizados com semelhantes objetivos.

O iDojo é utilizado para fins semelhantes ao que o Dojo do Splinter propõe. Os usuários conseguem cadastrar turmas, e por meio dela gerenciar graduações, horários, além de fornecer a possibilidade de criar um site para a academia e fornecer relatórios com contas a pagar e receber. Disponibiliza aplicativo para os alunos, porém não o disponibiliza para os instrutores. O Dojo do Splinter terá foco no suporte aos dois públicos, alunos e instrutores, disponibilizando versões em diferentes plataformas, *mobile* e *web*, permitindo que o mesmo sistema seja acessado por ambos tipos de usuários.

A segunda plataforma analisada, DojoWeb, é apenas disponibilizada em interface *web*, mas oferecendo design responsivo para ser acessado por dispositivos móveis. Seu foco é na parte financeira da academia, como gestão de pagamentos, controle de mensalidades, indicadores de movimentação e integração para pagamentos, funcionalidade não existente no Dojo do Splinter.

O iBlackbelt, terceira plataforma analisada, possui interface *web*, também com foco na parte financeira, mas seu diferencial é o controle do gerenciamento de aulas experimentais. Seu controle de mensalidade com opção de personalização por aluno, se assemelha muito à proposta pelo Dojo do Splinter, que permite que o valor da mensalidade seja negociado para um aluno específico. Porém, de todos é o que apresenta maior investimento para uso da plataforma.

O diferencial do Dojo do Splinter, diferente dos demais, é a possibilidade do aplicativo ser usado mesmo offline depois do primeiro login, permitindo que seja consultado os dados já vistos, e sendo sincronizado quando o usuário se conectar novamente a internet. Além disso, permite o acesso de instrutores e alunos em suas versões *web* e *mobile*.

A Tabela 1 apresenta o comparativo entre as características das plataformas analisadas e o Dojo do Splinter.

Tabela 1: Resultados do estado da arte

<b>Funcionalidade</b>	<b>iDojo</b>	<b>DojoWeb</b>	<b>iBlackbelt</b>	<b>Dojo Splinter</b>
<b>Controle de Mensalidades</b>	X	X	X	X
<b>Sistemas de Graduação</b>	X	X		
<b>Turmas e Horários</b>	X	X		X
<b>Permissões por Aluno ou Instrutor</b>	X	X	X	X
<b>Locais de Treino</b>	X			
<b>Relatórios</b>	X	X	X	
<b>Cadastro de Dados Pessoais</b>	X		X	X
<b>Configuração de Planos</b>			X	
<b>Registro de Aula e Frequência</b>	X	X	X	X
<b>Interface de Aluno</b>	X			X

<b>Funcionalidade</b>	<b>iDojo</b>	<b>DojoWeb</b>	<b>iBlackbelt</b>	<b>Dojo do Splinter</b>
<b>Contas a Pagar e Receber</b>	x	x	x	
<b>Website da Academia</b>	x		x	
<b>Controle de aula experimental</b>			x	
<b>Aplicativo híbrido</b>	x			x
<b>Sincronização Offline</b>				x
<b>Pagamento Integrado</b>		x	x	
<b>Valor de Mensalidade</b>	grátis (10 alunos) ou a partir de 29,90	entre 29,90 e 109,90	entre 169 e 339,00	<b>grátis</b>

Fonte: O Autor

Nesta sessão, foi analisada e comparada a proposta deste trabalho com plataformas já consolidadas no mercado, mostrando suas características e objetivos. No próximo capítulo, serão apresentadas e explicadas as tecnologias utilizadas para o desenvolvimento do sistema Dojo do Splinter.

### 3. REVISÃO BIBLIOGRÁFICA

O sistema desenvolvido neste projeto tem como objetivo auxiliar no gerenciamento de pequenas academias de artes marciais, através da disponibilização de uma versão *web* e de uma versão para dispositivos móveis, para acesso de alunos e instrutores. Para o desenvolvimento utilizou-se o Flutter, um *Framework Open Source* desenvolvido pelo Google, gerando tanto a aplicação *web* quanto a versão *mobile*, para dispositivos da plataforma Android. O aplicativo possibilita o uso *offline*, e realiza a atualização dos dados uma vez que o usuário volte a ficar *online*. O sistema *web* é hospedado diretamente no Firebase Hosting (FIREBASE, 2022), sem a necessidade de gerenciar uma instância.

#### 3.1. DISPOSITIVOS MÓVEIS

Dispositivos móveis são um tipo de dispositivo computacional que tem como características a portabilidade, a compatibilidade e a fácil utilização, o avanço das tecnologias de processamento, de armazenamento e de visualização proporcionam a criação deste tipo de dispositivo que muito se assemelha aos computadores tradicionais em relação à funcionalidade e ao uso.

O crescimento no uso de dispositivos móveis aumenta também a demanda por soluções de *softwares* que possam operar nesta tecnologia. Os ditos aplicativos móveis, disponíveis especialmente para os dispositivos supracitados.

#### 3.2. APLICATIVOS MÓVEIS

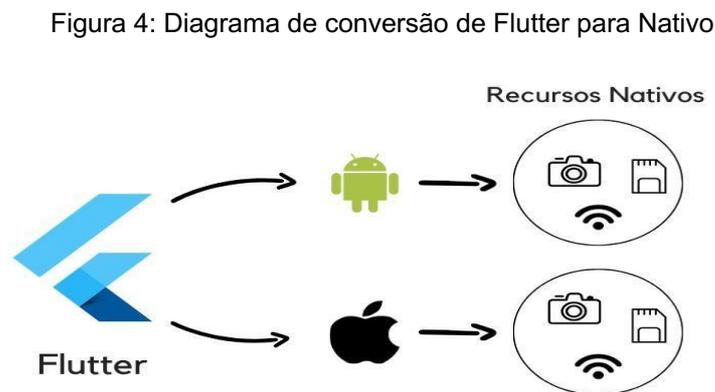
Aplicativos móveis são os softwares que utilizam um dispositivo móvel para realização de determinadas funções para o usuário utilizar o dispositivo, Estes aplicativos são desenvolvidos tendo como objetivo de auxiliar as atividades cotidianas da população alvo, oferecendo serviços simples que podem ser realizados com apenas alguns passos. Existem diversos tipos de aplicativos para gerenciamento de dados, para comunicação, realização de compras, acesso a redes sociais dentre outras atividades. Cada dispositivo possui plataformas diferentes, as mais conhecidas são *Android*, *iOS* e *Windows Phone*.

### 3.3. LINGUAGENS DE PROGRAMAÇÃO

Uma linguagem de programação é um conjunto de comandos, instruções e outras sintaxes usadas para criar um programa. As linguagens utilizadas para escrever código, são chamadas de linguagens de alto nível. Esse código pode então ser compilado para uma linguagem de baixo nível, que é reconhecida diretamente pelo hardware do computador. (CHRISTENSSON, 2021)

#### 3.3.1. FLUTTER

O Flutter é um tipo de *framework*, ou seja, ele é um facilitador no desenvolvimento que possibilita a criação de aplicativos mobile tanto para *Android* como para *iOS* oferecendo várias funcionalidades para os programadores. A Figura 4 representa o funcionamento do Flutter para conversão de Flutter para aplicativo nativo.



Fonte: Página web do Flutter (<https://flutter.dev/>)

### FUNCIONAMENTO DO FLUTTER

O *flutter* possui código aberto sobre a *BSD License* e Multiplataforma, ele possui como linguagem base o *Dart* para criar aplicativos oferecendo facilidade de desenvolvimento. Desta forma ao criar um aplicativo através do *flutter*, seu código é compilado para a linguagem do dispositivo.

Desta forma, as aplicações tornam-se realmente nativas, desta forma o usuário pode acessar os recursos do dispositivo sem a necessidade de auxílio de terceiros, gerando assim um melhor desempenho. As características do Flutter são:

- Multiplataforma: através dele é possível desenvolver aplicações em qualquer sistema operacional;
- Permite a criação de aplicativos nativos, através da utilização de um único código base, onde é possível desenvolver aplicações nativas tanto para Android como para *iOS*;
- Permite o acesso direto aos recursos do sistema, desta forma uma aplicação criada no *Flutter* pode ter acesso nativo aos recursos dos dispositivos como câmeras, *Wi-Fi*, memória, dentre outros; e
- Oferece maior desempenho, assim as aplicações desenvolvidas no Flutter tem maior desempenho quando são comparados ao *React Native*, tendo em vista que todo código-fonte é transformado em código nativo.

Desta forma, o *Flutter* pode ser considerado um *framework* bastante adequado aos requisitos do sistema.

### 3.3.2. 3.2.2DART

Dart é uma linguagem otimizada para o *client*, para aplicativos multi-plataforma, sendo uma linguagem especializada nas necessidades de criação de interface.

A linguagem Dart é considerada uma linguagem de programação em crescimento, sendo utilizada em diversos aplicativos comuns aos usuários de smartphone como: Nubank, Ebay e Google Pay. Se mostrando sólida e versátil para diferentes casos de uso. Sendo assim, podemos constatar que essa linguagem apresenta ferramentas necessárias para o atual desenvolvimento de aplicativos para aparelhos móveis, como o Dojo do Splinter.

A Figura 5 mostra uma comparação de execução de algoritmos entre as linguagens Dart e Node.js, onde Dart se mostra superior em tempo de execução e consumo de memória em cenários como *n-body* e *spectral-norm*, e em

consumo de memória, mas perdendo em tempo de execução nos cenários *k-nucleotide* e *fannkuch-redux*.

Figura 5: Benchmark de linguagem Dart vs Node js

<u>n-body</u>						
source	secs	mem	gz	busy	cpu load	
<u>Dart</u>	<b>7.30</b>	10,352	1311	7.89	100%	0% 4% 4%
<u>Node js</u>	8.60	36,000	1268	8.85	100%	2% 0% 1%

<u>spectral-norm</u>						
source	secs	mem	gz	busy	cpu load	
<u>Dart</u>	<b>1.48</b>	25,380	1196	5.72	97%	97% 97% 95%
<u>Node js</u>	1.64	67,912	999	6.11	92%	93% 96% 92%

<u>k-nucleotide</u>						
source	secs	mem	gz	busy	cpu load	
<u>Dart</u>	16.66	326,548	1520	49.39	77%	71% 73% 75%
<u>Node js</u>	15.82	396,716	1812	44.48	63%	47% 81% 90%

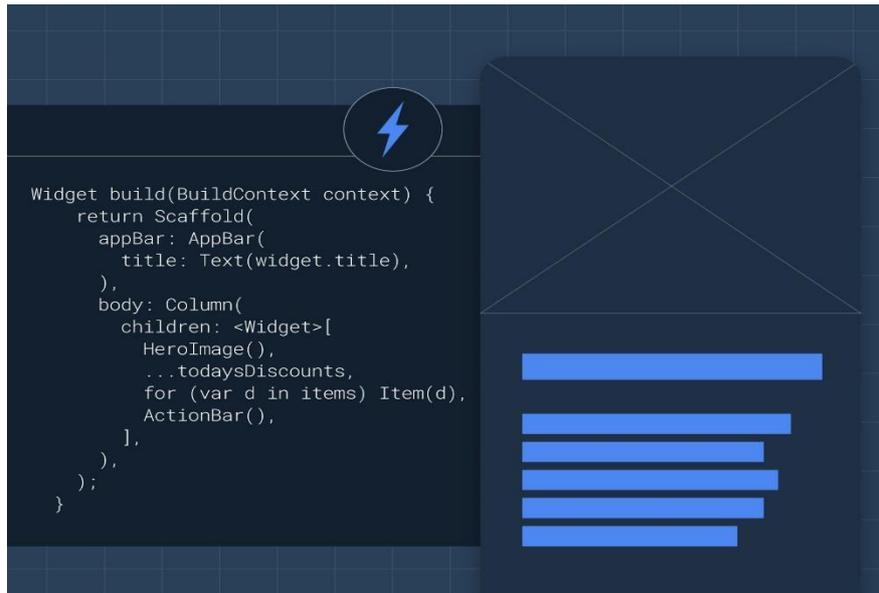
  

<u>fannkuch-redux</u>						
source	secs	mem	gz	busy	cpu load	
<u>Dart</u>	12.70	15,288	1220	50.67	100%	100% 100% 100%
<u>Node js</u>	11.38	65,452	1313	45.00	99%	99% 99% 99%

Fonte: The benchmark game

O *Dart* surgiu com o intuito de substituir o *JavaScript*, como a linguagem principal para os navegadores (BRACHA; BAK. 2011), porém, em 2015, *Dart* passou a ser compilado para *JavaScript* para o uso nos navegadores, pois a ideia de alterar a linguagem principal no *Chrome Virtual Machine (VM)* acabou sendo abandonada. A Figura 6 apresenta um exemplo de *Widget* utilizando *Dart*.

Figura 6: Exemplo de widget em Dart



Fonte: Dart.dev

### A sintaxe do Dart

Dart é um descendente da família de linguagens ALGOL, junto com C, Java, C#, JavaScript, entre outras. No entanto, o Dart procura reduzir os ruídos que são característicos das linguagens que são baseadas em C. Implementando alguns métodos conhecidos de outras linguagens, como Mixins de Ruby e o isolamento de código, utilizado pelos WebWorkers no JavaScript ou nos processos em Erlang.

Sendo assim, é possível afirmar que a linguagem Dart possui similaridades com linguagens já conhecidas dos desenvolvedores, e mostra flexibilidade ao incorporar características de outras linguagens, tornando-se familiar para o desenvolvedor e oferecendo um conjunto de ferramentas completas e robustas.

O Dart suporta características de linguagens funcionais, como armazenamento de funções em variáveis, *currying*, e *lazy parameters*.

Figura 7: Exemplo do uso de curried addition em Dart

```

main() {
  f1(x) => x * 2;           // Define the function f1
  var f2 = f1;            // Assign f1 to the variable f2
  print(f2(7));          // Feel free to call f2 like any other function

  add(a) => (b) => a + b; // Curried addition
  print(add(3)(4));      // Calling curried addition

  var add3 = add(3);     // Combining the
  print(add3(2));       // concepts
}

```

Fonte: StackOverflow (<https://stackoverflow.com/questions/16746807/does-dart-support-functional-programming>)

### 3.3.3. FIREBASE

*Firebase* é uma plataforma para desenvolvimento de aplicativos móveis do Google que auxilia na criação, na melhoria e na expansão de seu aplicativo. Através dele, é possível oferecer experiências de aplicativos mais ricas para os usuários e assim otimizar a performance e a experiência da plataforma. Oferece sincronização de dados JSON em tempo real, colaboração fácil entre vários dispositivos, permite que não seja necessário um servidor para o aplicativo, é otimizado para o uso offline e possui forte segurança baseado no usuário. (GOOGLE, 2021).

O *Firebase* possui um conjunto de ferramentas de desenvolvimento, tais como o *Realtime Database* e o *Cloud Firestore* que podem armazenar os dados estruturados em documentos e sincronizar os aplicativos correspondentes sempre que acontecer a transformação de dados, assim o aplicativo e seu banco de dados ou em um ao outro e proporcionam ao usuário uma experiência integrada no aplicativo. (GOOGLE, 2021).

#### Linguagens suportadas

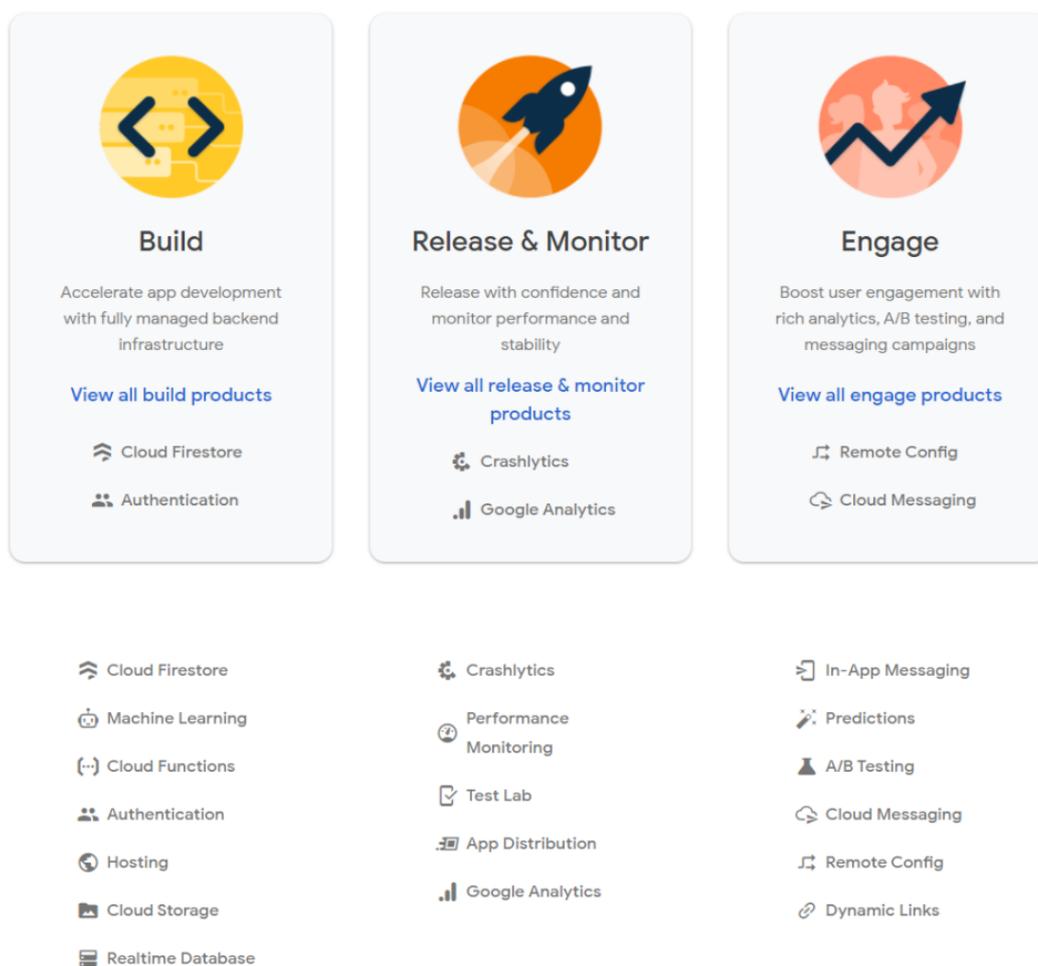
O *Firebase* suporta o desenvolvimento das linguagens: C++, *JavaScript*, *Node.js*, *Objective-C* e *Swift*. Os *frameworks Angular*, *Backbone* e o *React* são suportados pela vinculação de nomes (*bindings*), direto com o banco de dados. Além destas linguagens o google adicionou as bibliotecas: *Firebase UI*, *Geofire*, *Firebase Queue*, *FirebaseJobDispatcher*. (GEEK HUNTER, 2021)

O *Firebase* foi desenvolvido para integração com aplicativos da web e mobile com *SDKs* para uma imensa variedade de linguagens. Ele é uma suíte de aplicativos que são feitos sob medida para qualquer desenvolvedor que necessita de ferramentas para construção de *apps*, mais robustos, com menos erros e mais otimizado para os usuários, ele é escalável e ainda mais lucrativo.

## Serviços

Os principais serviços do *Firebase* estão representados na Figura 8.

Figura 8 - Serviços Firebase



Fonte: Geekhunter

O *firebase* oferece:

- Segurança para comunicação e para transferência de dados;
- Limitações na infraestrutura;
- *Bugs* de atualizações;
- Compatibilidade com diversos tipos de dispositivos diferentes;
- Autenticação;
- Problemas que envolvem UI/UX.

O desenvolvimento utilizando o Firebase juntamente com o Flutter ocorre de maneira bem natural, visto que a biblioteca de suporte ao Flutter oferece todos os recursos necessários para cadastros e consultas.

Na Figura 9 pode ser observado o uso da biblioteca do Firebase para fazer um registro de um pagamento realizado. Tal registro pode ser visto na Figura 10.

Figura 9: Registro de pagamento usando api do Firebase para Flutter

```

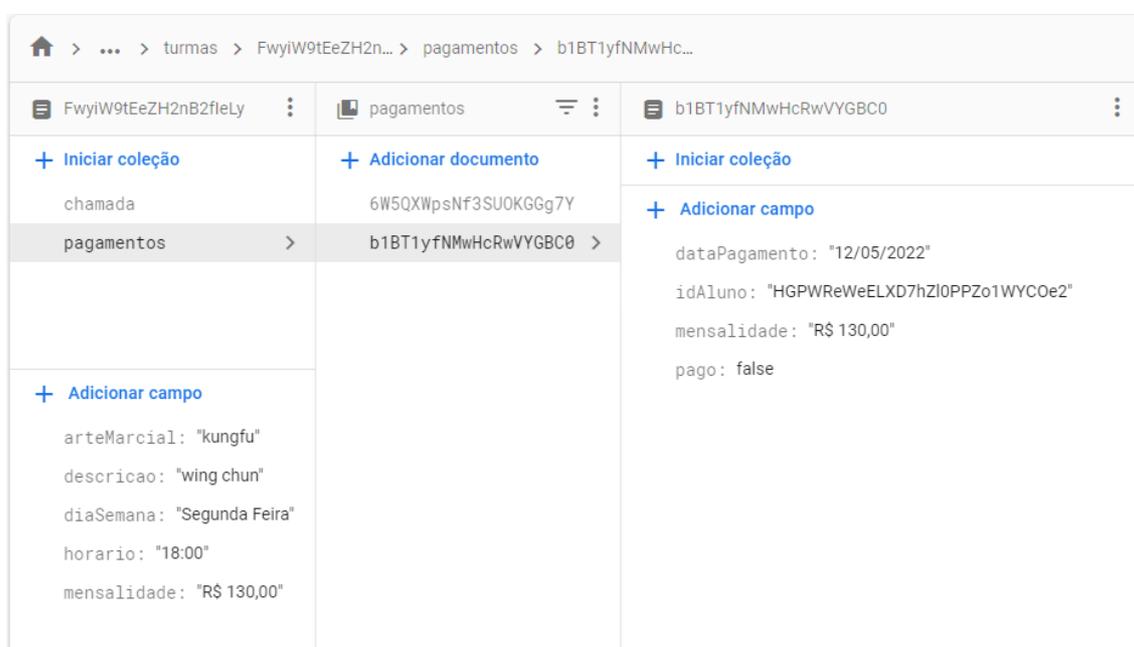
227      await GetIt.I<FirebaseFirestore>()
228          .collection('instrutores')
229            .doc(widget.idInstrutor)
230              .collection('turmas')
231                .doc(widget.idTurma)
232                  .collection('pagamentos')
233                    .doc()
234                      .set({
235                        'dataPagamento': Helper.getData(
236                          _dateSelecionada,
237                        ),
238                        'idAluno': widget.idAluno,
239                        'pago': checkboxValue,
240                        'mensalidade':
241                          mensalidadeController.text
242                      });
243      setState() {
244        fazendoRequisicao = false;
245      });
246      _onSuccess();

```

Fonte: O próprio autor

Neste exemplo, o código está mapeando a coleção de documentos, e fazendo a inserção de um registro em **instrutores > ID> turmas > ID > pagamentos** que pode ser observada na imagem abaixo. Após inserido, o registro de pagamento fica salvo no seguinte endereço: **/instrutores/nZx6nw8dQ6abk1f5bo9ijqpzCZs1/turmas/FwyiW9tEeZH2nB2fleLy/pagamentos/b1BT1yfNMwHcRwVYGBC0**

Figura 10: Armazenamento de dados de pagamento no Firebase



Fonte: O autor

Dentre outras funções que o Firebase oferece, deve-se ressaltar a criação dos usuários, visto que seu sistema de autenticação, abstrai boa parte do que geralmente é necessário ser desenvolvido, como por exemplo a validação por senhas fracas – método que pode ser observado realizando o tratamento deste erro para a interface do usuário na Figura 11.

Figura 11: Tratamento de erros do Firebase para o usuário

```
202 Future<void> _signUp() async {
203   if (_formKey.currentState!.validate()) {
204     try {
205       await GetIt.I<AuthRepository>().createUserWithEmailAndPassword(
206         email: _emailController.text,
207         password: _senhaController.text,
208       );
209     } on FirebaseAuthException catch (e) {
210       if (e.code == 'weak-password') {
211         ScaffoldMessenger.of(context).showSnackBar(
212           const SnackBar(
213             content: Text(
214               'Senha fraca\nPor favor, use uma senha mais forte',
215             ), // Text
216             backgroundColor: Colors.red,
217           ), // SnackBar
218         );
219       } else
220         _onError();
221     } on Exception {
222       ScaffoldMessenger.of(context).showSnackBar(
223         const SnackBar(
224           content: Text('um erro ocorreu'),
225           backgroundColor: Colors.red,
226         ), // SnackBar

```

Fonte: O Autor.

Neste capítulo analisou-se as tecnologias utilizadas para o desenvolvimento deste projeto. O próximo capítulo demonstra o tipo de pesquisa realizada, a metodologia escolhida para o desenvolvimento do sistema, e quais tecnologias e como serão utilizadas.

#### 4. METODOLOGIA

Uma metodologia de desenvolvimento de software é uma maneira de gerenciar o desenvolvimento de um projeto de software. Isto normalmente aborda problemas como selecionar funcionalidades para inclusão na atual versão, quando o software será lançado, quem trabalha em quê e quando o teste é feito (YOUNG, 2012, p.1; tradução nossa).

Para o desenvolvimento deste projeto foi escolhida a metodologia Scrum Solo, uma versão adaptada da metodologia Scrum – que visa organização de desenvolvimento para um time – para projetos contendo apenas um integrante.

O *Scrum* é um *framework* que visa ajudar pessoas, times e organizações a gerar maior valor através de soluções adaptativas para problemas complexos. Normalmente o *Scrum* necessita de um *Scrum Master*, para nutrir o ambiente onde:

- O Dono do Produto solicita trabalho para um problema complexo gerando um *Backlog* de Produto.
- O Time de *Scrum* seleciona uma parte do trabalho para gerar um acréscimo de valor durante uma *Sprint*.
- O Time de *Scrum* e os *stakeholders* inspecionam os resultados e realizam ajustes para a próxima *Sprint*.
- Repete-se o processo. (Pagotto e Golçalves, 2013)

O produto deste projeto será um sistema híbrido, ou seja um aplicativo que é desenvolvido utilizando os recursos web, podendo ser portado como um aplicativo nativo para outras plataformas, como Android e IOS. Possibilitando assim a abrangência de uma quantidade maior de usuários. Segundo Mendes, Garbazza e Terra (2014),

A abordagem híbrida evita que um aplicativo precise ser desenvolvido várias vezes para atingir diversas plataformas, uma vez que utilizando um framework específico, o código desenvolvido com tecnologia web pode ser compilado para diferentes plataformas. (MENDES *et al*, 2014. p.3)

Para organizar o desenvolvimento do trabalho, dividiu-se em cinco etapas, que serão listadas na sequência:

A primeira etapa do trabalho foi uma pesquisa exploratória para criação de referencial teórico, identificação de trabalhos relacionados e melhor definição

de requisitos do sistema para adequação às necessidades comuns das academias de artes marciais.

A segunda etapa foi planejar o sistema utilizando linguagem UML, utilizando de diagrama de classes e de sequência. O sistema e suas funcionalidades serão documentados durante todo o processo de desenvolvimento, iniciado após esta etapa.

A terceira etapa foi a implementação do sistema, dividida entre a configuração do Firebase e o desenvolvimento dos testes unitários e o desenvolvimento em Flutter/Dart.

A quarta etapa foi a publicação do aplicativo na Google Play Store e o teste de aceitação foi feito por meio de pesquisa com alunos, instrutores e não praticantes.

A quinta e última etapa consiste na análise e divulgação dos resultados obtidos ao longo do trabalho.

Neste capítulo foi apresentada a metodologia escolhida para a pesquisa, desenvolvimento e avaliação do projeto Dojo do Splinter. O próximo capítulo descreve a aplicação desenvolvida.

## **5. DESENVOLVIMENTO**

Este capítulo tem como objetivo apresentar as etapas do desenvolvimento do projeto Dojo do Splinter, apresentando a modelagem, a descrição do sistema, abordando as telas e funcionalidades de ambas as aplicações, e o teste de aceitação, juntamente com os resultados obtidos.

### **5.1. MODELAGEM**

Durante a etapa de modelagem e especificação de requisitos, primeiramente foi elaborado o documento de visão para obter um escopo de alto nível do produto. Posteriormente, o diagrama de casos de uso foi desenvolvido para ter uma especificação mais detalhada das funcionalidades a serem implementadas. Por fim, o diagrama de classes para representar as Coleções no banco de dados orientado a documentos.

#### **5.1.1. DOCUMENTO DE VISÃO**

Inicialmente, para o planejamento e estruturação do projeto, foi feito o desenvolvimento do documento de visão com o objetivo de coletar, analisar e definir as principais necessidades levantadas junto a um instrutor de academia. Este documento foca nas necessidades dos alunos e instrutores, que são o público alvo deste trabalho, e como esses requisitos são abordados no sistema Dojo do Splinter. Com isso, foi possível obter e definir os principais pontos a serem desenvolvidos para a aplicação, fornecendo a todos os envolvidos uma descrição compreensível das macro funcionalidades requeridas.

Foi identificado que o Dojo do Splinter, um sistema que proporciona o gerenciamento de academias de artes marciais a alunos e instrutores destas academias, surge como solução para o problema da falta de gerenciamento, fichas de papel esquecidas e falta de engajamento em eventos promovidos. Estes problemas afetam o fluxo de caixa da academia, e o alcance de seus eventos, causando impacto para os instrutores das academias de artes marciais, que enfrentam dificuldade em manter o fluxo de alunos em suas turmas, e um gerenciamento insuficiente. Para os alunos, a dificuldade de encontrar maiores informações sobre as turmas disponíveis na academia e a falta de transparência

quanto a pagamentos em aberto ou eventos agendados. Definiu-se, também, as partes interessadas e suas responsabilidades no sistema, que seriam as seguintes:

- **Instrutores:** Devem utilizar o sistema para criar turmas e eventos e gerenciar os alunos e pagamentos nas respectivas turmas.
- **Alunos:** Devem utilizar o sistema para ver as turmas disponíveis, fazer matrícula, acompanhar eventos e visualizar pagamentos concluídos e em aberto.
- **Administrador:** Deve autorizar o cadastro de Instrutores no sistema.
- **Time de desenvolvimento:** Responsável pelo desenvolvimento do projeto, projetando as funcionalidades de acordo com os requisitos definidos e fazendo alterações necessárias no escopo para melhor atender as necessidades.

Nesta etapa criou-se também a descrição do produto, contendo o que o projeto almeja, o que é e sua comparação com outros produtos já consolidados no mercado. A partir disso, o Dojo do Splinter diferencia-se de outros sites e aplicativos, como o iDojo, DojoWeb, iBlackBelt, pela sua interface simplificada e de fácil uso, sem a necessidade de configurações extensas para o seu primeiro uso, abrangendo os principais requisitos levantados de maneira a manter a praticidade de uso, facilitando a organização de turmas e gerenciamento de alunos e pagamentos. O documento de visão pode ser visualizado em sua completude no Apêndice D deste trabalho.

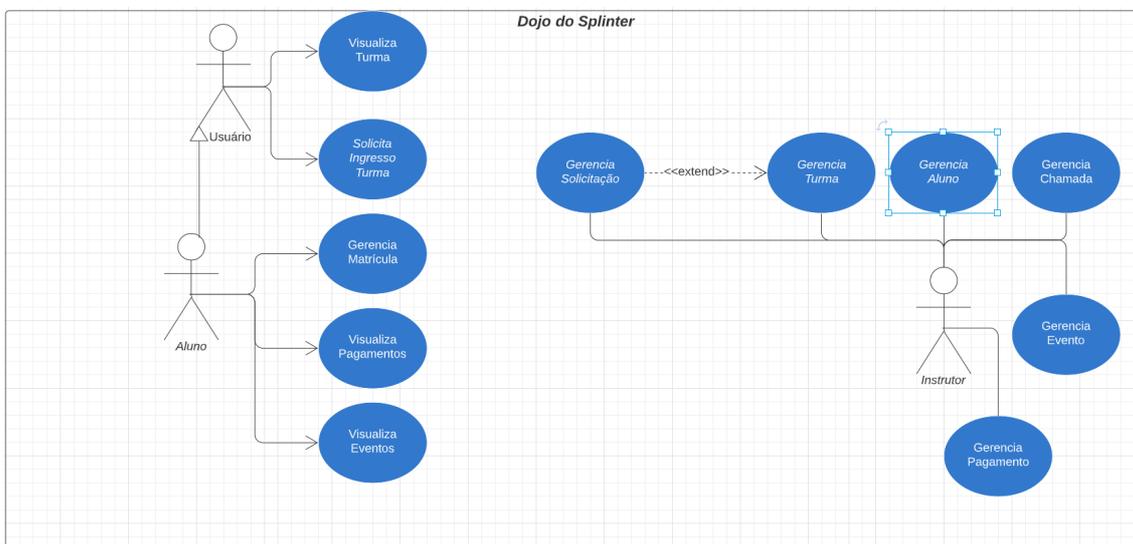
### 5.1.2. DIAGRAMA DE CASOS DE USO

O Diagrama de Casos de Uso foi construído, como auxílio visual, para descrever os atores e as funcionalidades do sistema. Foi utilizado a plataforma LucidChart, “um aplicativo de diagramação inteligente que reúne equipes para ajudá-las a tomar decisões melhores e construir o futuro.” (LucidChart, 2021).

Conforme apresentado no diagrama da Figura 12, o sistema possui três atores, sendo eles: Usuário, Aluno e Instrutor. O Aluno herda as características do Usuário, permitindo-o Visualizar Turmas e Solicitar Ingresso a uma Turma. Quando um Usuário, que Solicita Ingresso a uma Turma, é aprovado pelo

Instrutor, por meio do Gerenciamento de Solicitação, este Usuário então, se torna um Aluno desta Turma. Agora como Aluno, ele pode Gerenciar sua Matrícula nesta turma, Visualizar Pagamentos (pagos e em aberto), e Visualizar Eventos cadastrados para esta Turma. O ator Instrutor por sua vez, pode Gerenciar a Chamada de uma Turma, marcando ou desmarcando presença para os alunos matriculados na Turma em questão, bem como pode também Gerenciar Eventos, que se aplicam a toda academia. O Instrutor pode Gerenciar Pagamentos, que inclui visualizar, editar e criar novas instâncias de Pagamentos para os alunos matriculados na Turma

Figura 12: Diagrama de Caso de Uso do Dojo do Splinter



Fonte: O próprio autor

### 5.1.3 Organização do banco de dados

Foram utilizados dois tipos de bancos de dados, em nuvem, providos pelo Firebase, para o desenvolvimento da aplicação – o Realtime Database, que armazena os dados como JSON e os sincroniza em tempo real para cada cliente conectado (Firebase, 2022), e o Firestore Database, um banco NoSQL, que armazena os dados como documentos, permitindo fazer chamadas mais complexas, além de manter os dados em sincronia em aplicativos cliente usando *listeners* em tempo real.

## Token de Autenticação - Realtime Database

O Firebase Realtime Database foi usado para enviar sinais ao aplicativo para atualizar o token de identidade do usuário, trafegando o mínimo de dados possível, apenas um booleano, serviu como ideal visto o modelo de cobrança do banco em questão, visto que custa 5 USD por GB armazenado e 1 USD por GB baixado; logo ao armazenar e baixar apenas uma flag, os custos se tornam ínfimos.

Figura 13: Armazenamento de token de autenticação de usuário

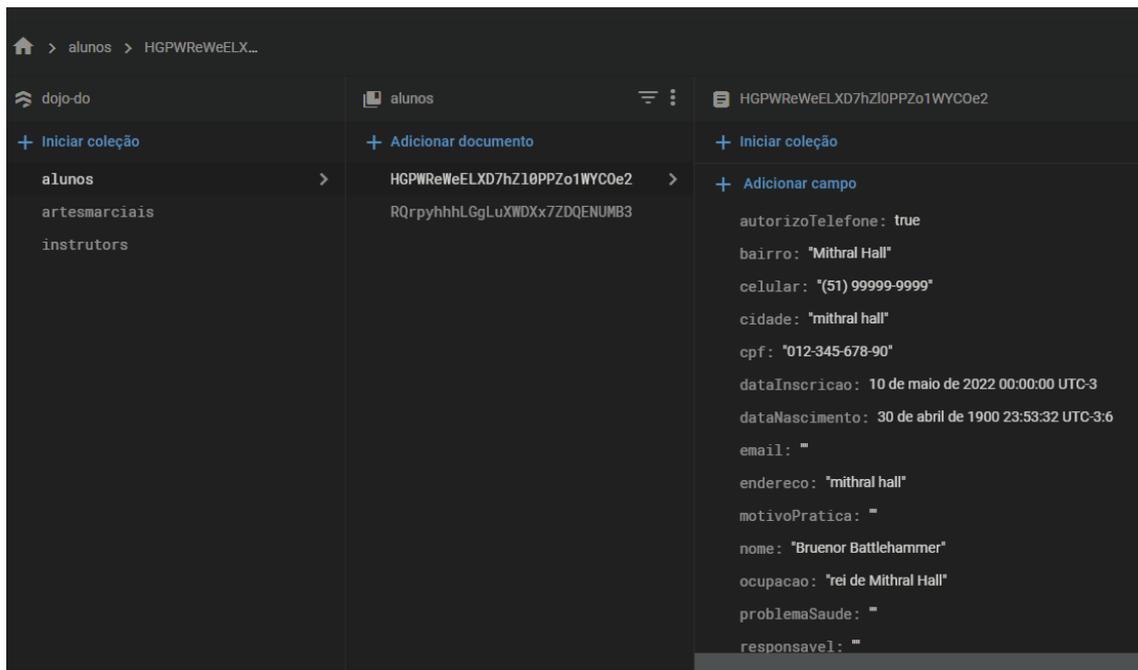


Fonte: O Autor

## Armazenamento - Cloud Firestore - Diagrama ER

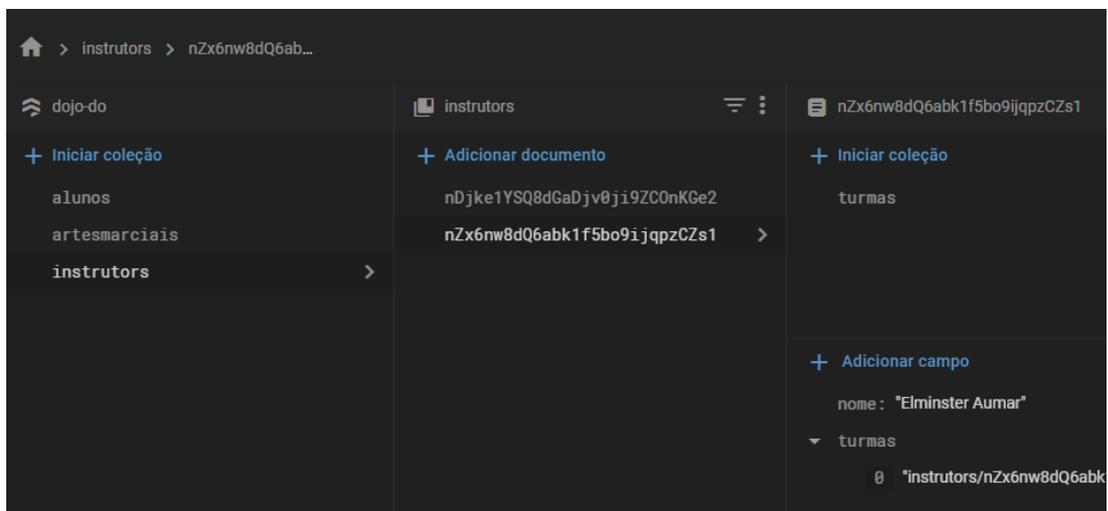
O Firebase Cloud Firestore foi usado para o armazenamento de todos os dados da aplicação, por ser um banco de dados NoSQL, sua estrutura se baseia na criação de documentos aninhados. A figura 14 e 15, mostram a estrutura do cadastro do Aluno e do Instrutor, visualizando-as no painel de controle do Firebase, na figura 16 é apresentado o diagrama ER, tentando trazer uma representação de como se estrutura o banco de dados para armazenamento dos dados do aplicativo.

Figura 14: Armazenamento de dados de Alunos



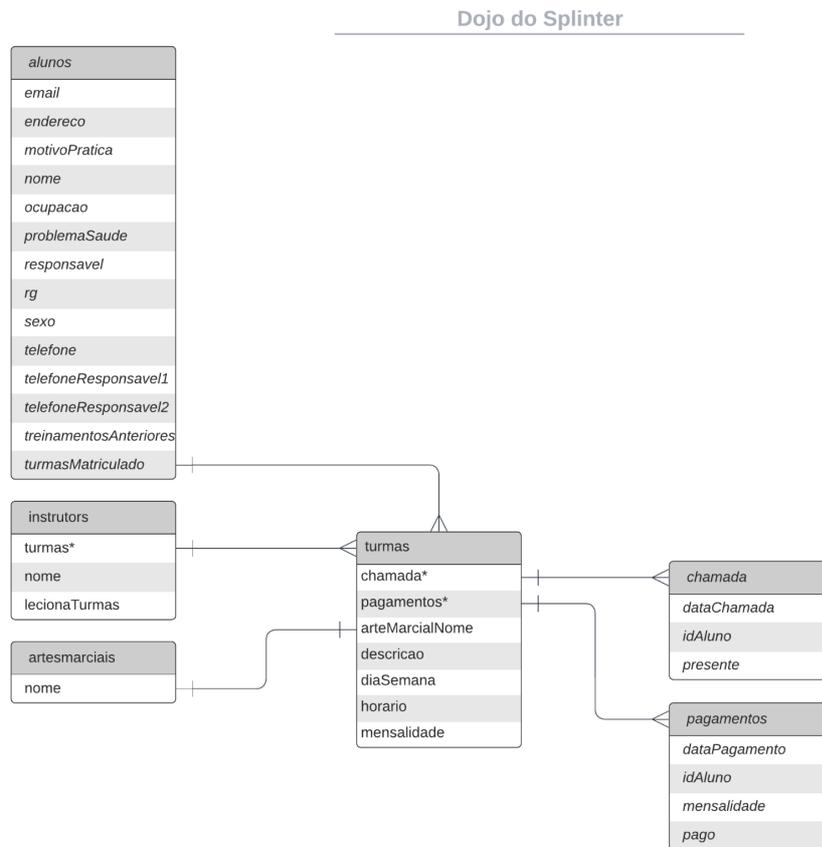
Fonte: O próprio autor

Figura 15: Armazenamento de dados de Instrutores



Fonte: O próprio autor

Figura 16: Diagrama ER



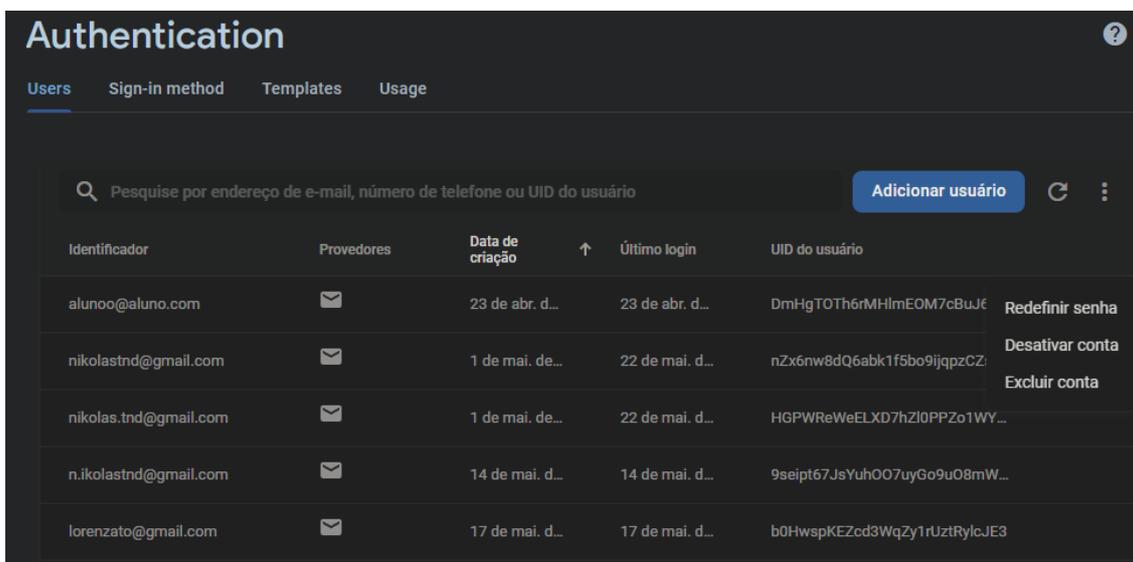
Fonte: O próprio autor

### Autenticação e Validação de email

O Firebase tem como produto o Firebase Authentication, uma plataforma integrada que permite o reconhecimento de atividade de usuário por meio do uso de SDKs e bibliotecas. Oferece suporte a senhas, números de telefone, email e provedores de identidades federadas conhecidas, como Google, Facebook, entre outros.

Na Figura 17 pode ser visto os emails cadastrados, o provedor, data de criação, último login, UID do usuário, bem como funções de gerenciamento disponíveis diretamente no painel de controle.

Figura 17: Tela do Firebase Authentication com usuários cadastrados e operações disponíveis



Fonte: O próprio autor

Optou-se pela autenticação utilizando email e senha, podendo futuramente ser adicionado outros meios de autenticação, que são disponibilizados pelo Firebase Authentication. A plataforma se encarrega do envio de email para confirmação do email cadastrado, permitindo ao administrador customizar o email com quais informações deseja apresentar e seu layout.

## 5.2. O DOJO DO SPLINTER

O Dojo do Splinter foi desenvolvido com o framework Flutter e propõe uma solução para a gestão de academias de arte marcial. A construção do sistema se iniciou com o foco na aplicação móvel, porém, tendo em vista que o Flutter permite que seja feito o build da aplicação para Android, iOS e Web, sendo

necessário apenas fazer ajustes de responsividade, no caso do uso de escalas predefinidas.

A escolha do nome, se deu por um apelido utilizado por um dos mestres da academia, cujo apelido é Mestre Vitor Splinter. Enquanto o termo dojo, significa um lugar onde se treinam artes marciais, que deve ser respeitado como a própria casa dos praticantes. O Dojo do Splinter, busca solucionar problemas em pequenas academias de artes marciais, visando reduzir o número de fichas em folha de papel, e possibilitar que o controle seja feito mesmo em lugares sem acesso a internet.

### **5.2.1. Login e Cadastro**

Na figura 18 é apresentado a primeira tela da aplicação, onde se inicia o fluxo do usuário, realizando-se o login com uma conta existente, ou fazendo a criação de um novo perfil – juntamente com a validação de email.

A criação do perfil do aluno, e o cadastro de seus dados, visam contemplar todos os dados que são utilizados na ficha de papel, que pode ser visualizada no Apêndice A. Os campos de cadastros foram divididos em grupos, sendo eles dados pessoais, Endereço, Contato, e informações Complementares

Figura 18: Telas de Login e Cadastro

The figure displays eight screenshots of a mobile application's user interface, organized in a 2x4 grid. The top row shows the login and registration screens, while the bottom row shows the multi-step registration process for a student.

**Top Row Screenshots:**

- Top Left:** "LOGIN DE USUÁRIO" screen. It features a stylized yellow logo at the top, followed by input fields for "E-mail" and "Senha" (password). A link "Esqueceu a senha?" is below the password field. A yellow "Acessar" button is at the bottom, with a link "Crie uma nova conta" below it.
- Top Middle-Left:** "CADASTRO DE USUÁRIO" screen. It features the same logo, followed by input fields for "E-mail", "Senha", and "Confirme a senha". A yellow "Inscreva-se" button is at the bottom, with a link "Login em uma conta existente" below it.
- Top Middle-Right:** A screen for email verification. It says "Por favor, cheque seu email, e valide-o clicando no link." Below it is a link "Não recebeu o código? Clique para Reenviar" and a yellow "Atualizar" button. A "Sair" link is at the bottom.
- Top Right:** A screen for user selection. It says "Eu quero me inscrever como:" followed by two radio button options: "Instrutor" and "Aluno". A "Sign out" link is at the bottom.

**Bottom Row Screenshots (Student Registration - "ALUNO CADASTRO"):**

- Bottom Left:** "Dados Pessoais" step. It includes input fields for "Data de inscrição", "Nome\*", "Data de nascimento", "Sexo", "Ocupação", "RG", "CPF", and "Responsável". A yellow "CONTINUE" button is at the bottom.
- Bottom Middle-Left:** "Endereço" step. It includes input fields for "Endereço\*", "Bairro", and "Cidade". A yellow "CONTINUE" button is at the bottom.
- Bottom Middle-Right:** "Contato" step. It includes input fields for "Celular\*", "Telefone", "Telefone responsável 1", and "Telefone responsável 2". A yellow "CONTINUE" button is at the bottom.
- Bottom Right:** "Complementares" step. It includes input fields for "Problema de saúde", "Motivo da prática", and "Treinamentos anteriores". A checkbox "Autorizo que meu telefone/e-mail seja compartilhado com os colegas de treino." is present. A yellow "CONCLUIR" button is at the bottom.

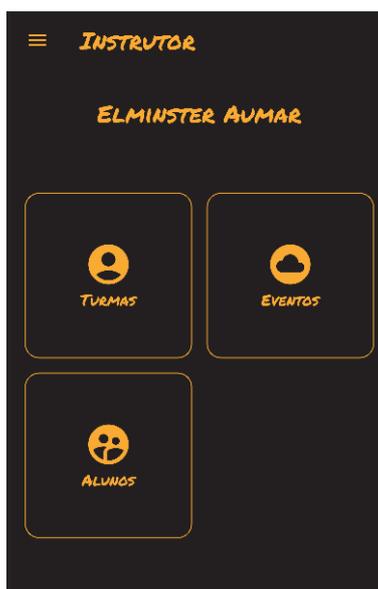
Fonte: O próprio autor

Quando o usuário realiza o cadastro, e valida o e-mail, então lhe é apresentado a tela para que escolha se deseja realizar cadastro como Instrutor ou Aluno. Quando cadastro feito como Instrutor, é exigido apenas o nome para a apresentação, e como aluno, é exigido o cadastro completo.

### 5.2.2. Perfil do Instrutor

Ao realizar acesso como Instrutor, a página inicial do Instrutor é mostrada, onde apresenta os menus disponíveis, como pode ser visto na Figura 19, sendo eles Turmas, Eventos, e Alunos.

Figura 19: Tela Principal do Instrutor

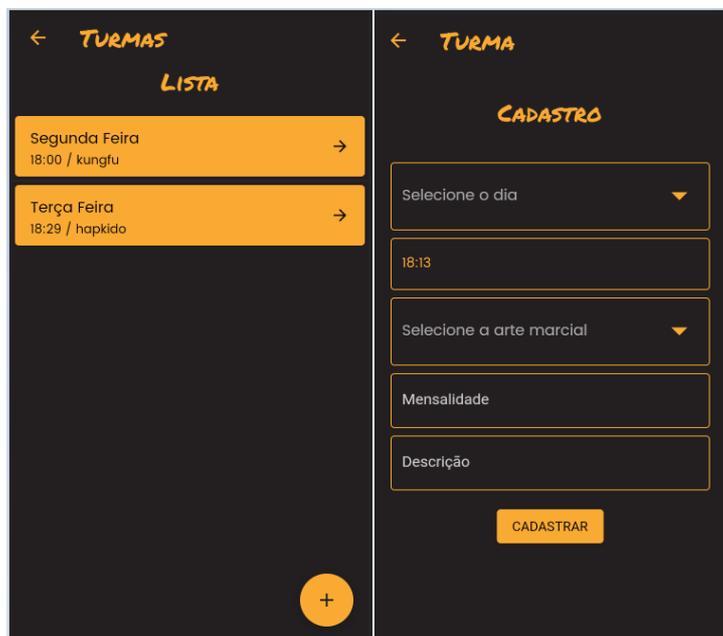


Fonte: O próprio autor

#### Turmas

O primeiro menu apresentado é o menu de Turmas, ao selecioná-lo, uma lista de turmas é apresentada, o botão no canto inferior esquerdo indica a possibilidade da adição de novas turmas pelo instrutor. Ao selecionar uma das turmas listadas, o Instrutor então é redirecionado para a tela inicial daquela turma, onde apresenta os menus específicos, sendo eles: Pagamentos, Chamada e Alunos. Representados pelas Figuras 20 e 21.

Figura 20: Telas de Listagem e Cadastro de Turma



Fonte: O próprio autor

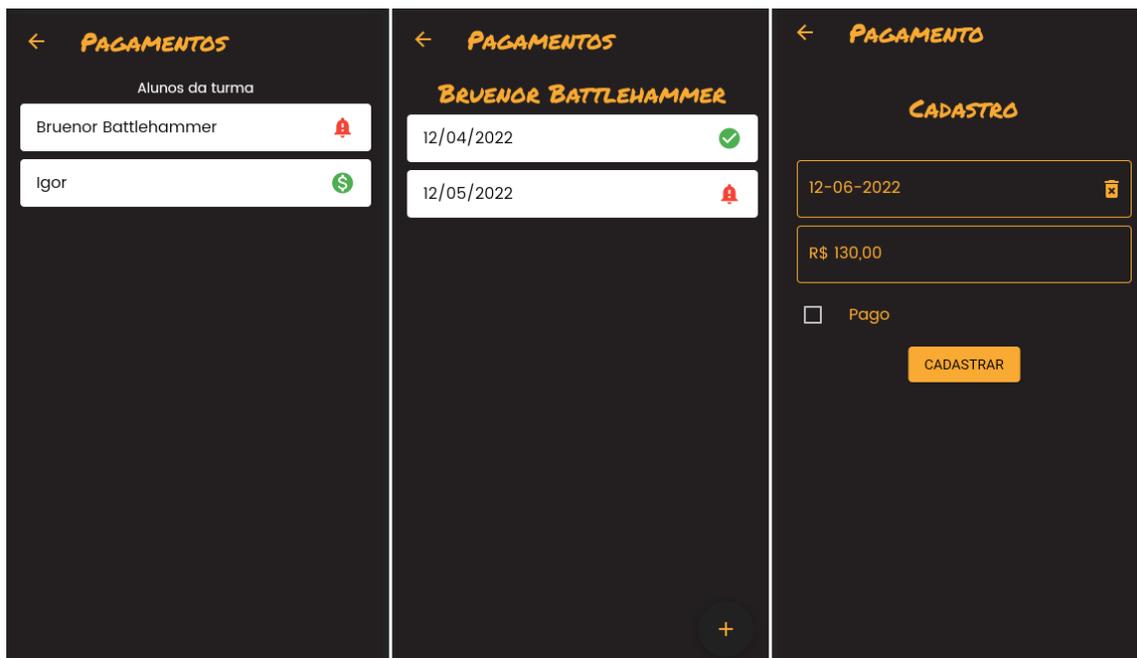
Figura 21: Tela Principal da Turma



Fonte: O próprio autor

Ao abrir o menu de pagamentos, é listado os alunos matriculados, e a situação do pagamento (simbolizado pelo ícone verde ou vermelho). Ao selecionar qual aluno deseja, então é apresentada a lista de pagamentos referente a este aluno. É possível então, selecionando o pagamento referente ao mês desejado, editar os dados do pagamento, como também marcá-lo como pago, como mostra a Figura 22.

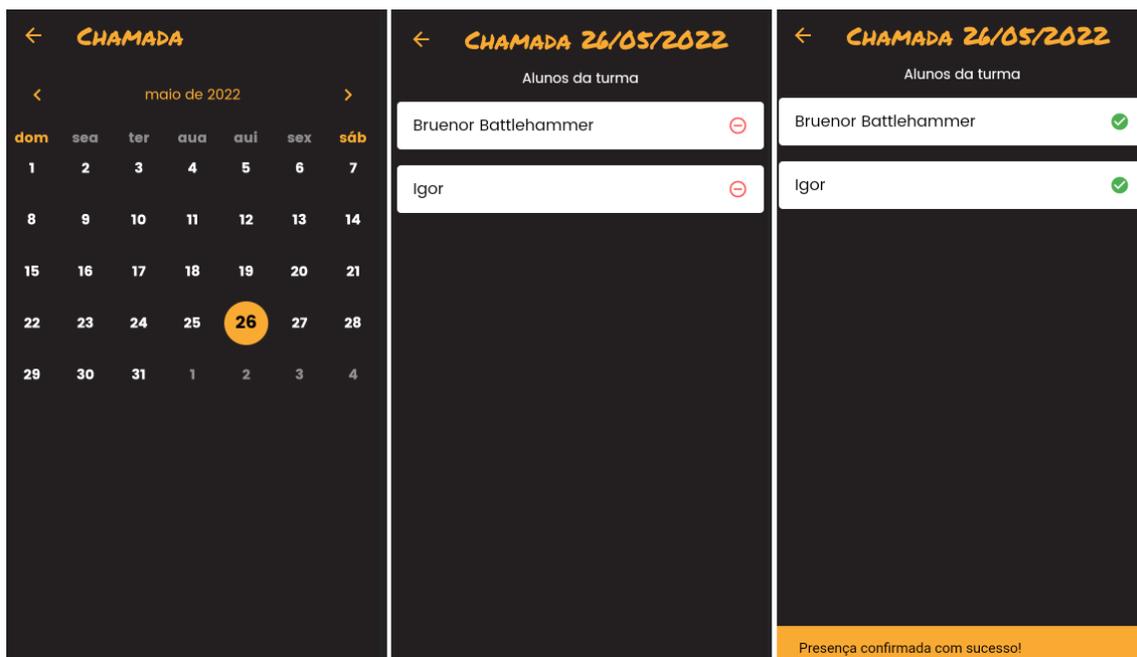
Figura 22: Telas de Listagem e Cadastro de Pagamentos



Fonte: O próprio autor

Ao acessar o menu de Chamada é apresentado um calendário para que o instrutor possa realizar a chamada. Ao selecionar o dia referente a chamada são apresentados os alunos matriculados na turma, e a chamada pode ser feita ao selecionar o nome de cada Aluno, marcando-o como presente ou ausente, como mostra a figura 23.

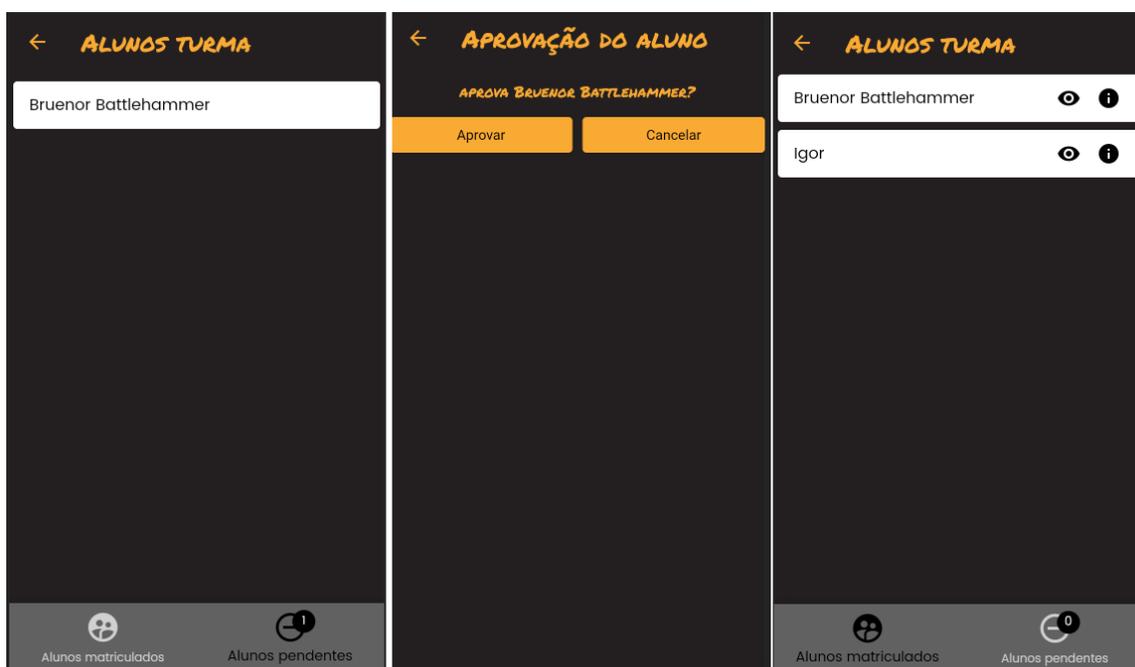
Figura 23: Telas de calendário de chamada e chamada diária



Fonte: O próprio autor

Ao selecionar o menu Alunos, é apresentado dois menus, um de Alunos Matriculados e outro de Alunos Pendentes. Alunos pendentes são aqueles que fizeram seleção de turma, realização do processo de inscrição, mas ainda não tiveram aprovação do instrutor. Ao selecionar um Aluno Pendente, e aprová-lo, ele então é adicionado à lista de Alunos Matriculados, como mostra a Figura 24.

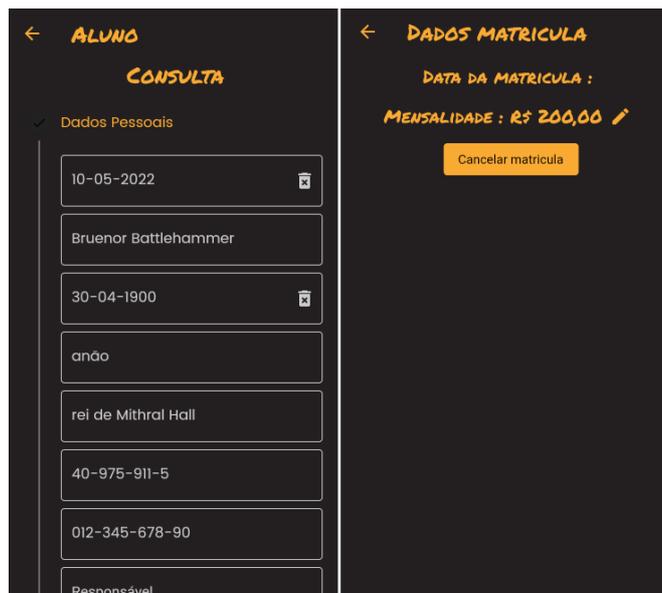
Figura 24: Telas de Alunos cadastrados e Alunos pendentes



Fonte: O próprio autor

Na lista de Alunos Matriculados, é possível ver os detalhes do Aluno e de sua matrícula, nos respectivos ícones ao lado de seus nomes. Os detalhes do aluno e da matrícula são apresentados conforme Figura 25.

Figura 25: Telas de consulta de dados de aluno e dados de matrícula



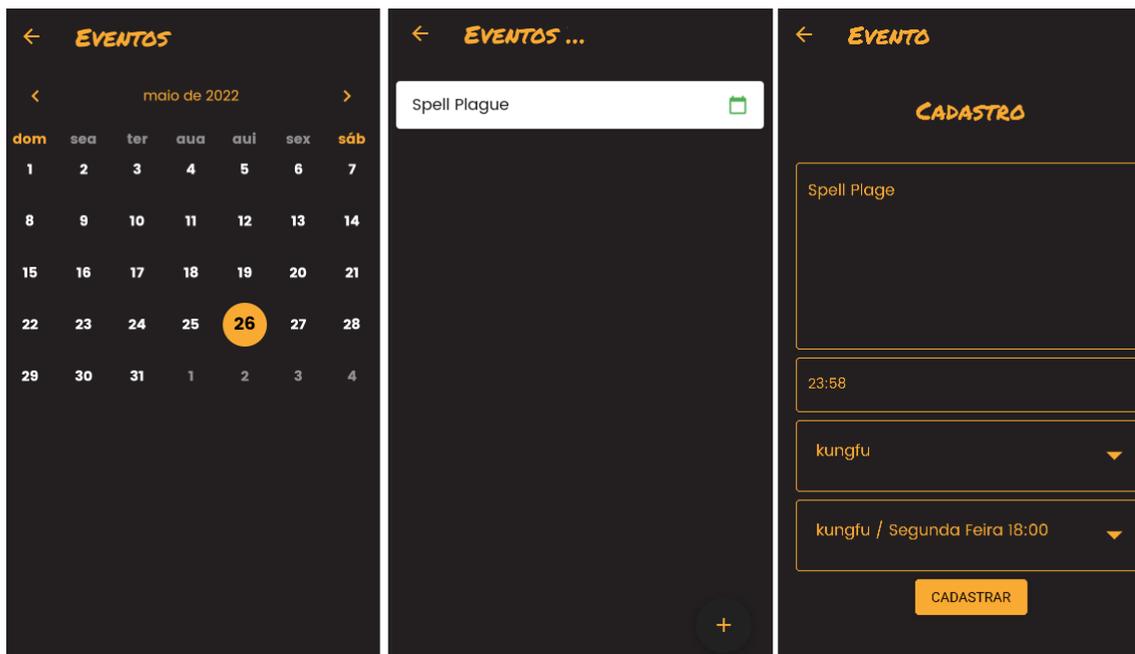
Fonte: O próprio autor

### 5.2.2.1. Eventos

Entre as atividades da academia, além das aulas, também estão os eventos. Entre os eventos que são realizados estão visitas, seminários, aulas abertas etc. Estes eventos também serão gerenciados através do sistema. Ao selecionar o menu “Eventos”.

Ao selecionar o menu de Eventos, um calendário é mostrado, ao selecionar o dia, é possível ver os eventos do dia, ou cadastrar um novo evento, como mostra a Figura 26.

Figura 26: Tela calendário de eventos, listagem e cadastro de novos eventos



Fonte: O Autor.

#### 5.2.2.2. Alunos

Quando o instrutor selecionar o menu de Alunos, é mostrado uma lista com todos os Alunos, sendo possível verificar os dados cadastrados, como mostra a figura 27.

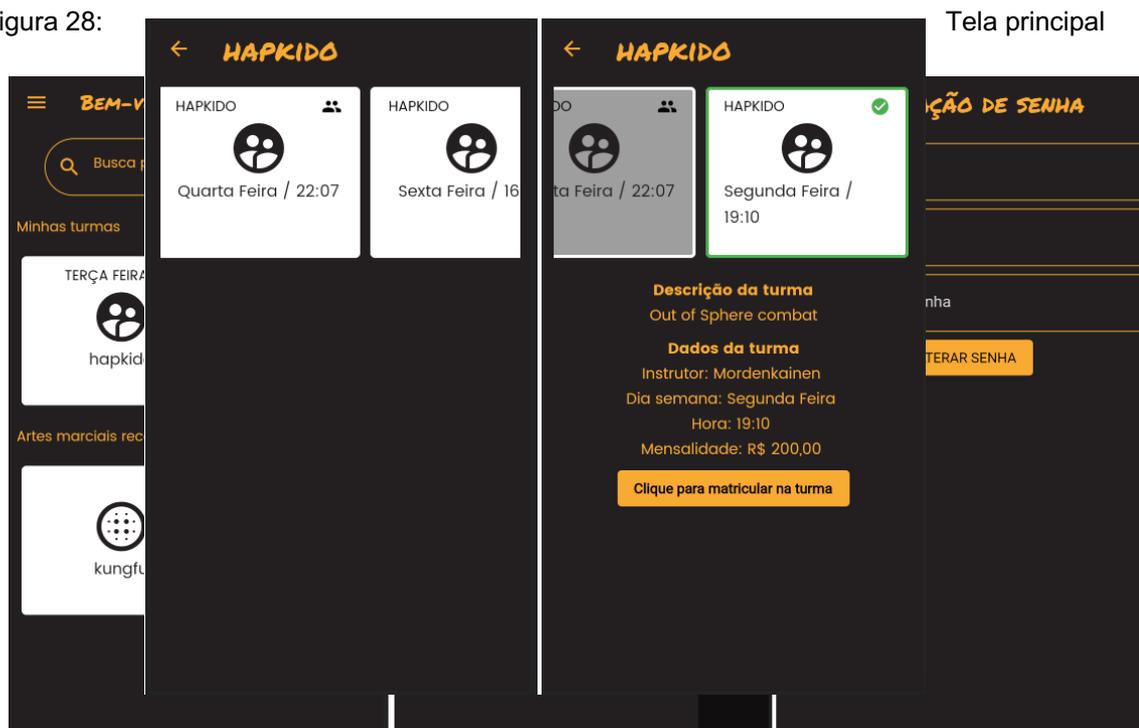
Figura 27: Telas de listagem e consulta de alunos

Fonte: O Autor.

#### 5.2.3. Perfil do Aluno

Ao fazer acesso como Aluno, a primeira página apresenta as Turmas em que o Aluno está matriculado, bem como as Artes Marciais disponíveis. No submenu à esquerda, é possível visualizar os Eventos e Alterar a Senha, além de poder fazer o Logout, conforme pode ser visto na Figura 28.

Figura 28:



de alunos, menu lateral e opção para alteração de senha

Fonte: O próprio autor

## Artes Marciais

Ao selecionar uma das Artes Marciais listadas, são listadas as aulas que são ministradas, e, ao selecionar a aula desejada, é possível matricular-se nesta turma. Tal comportamento pode ser observado na Figura 29.

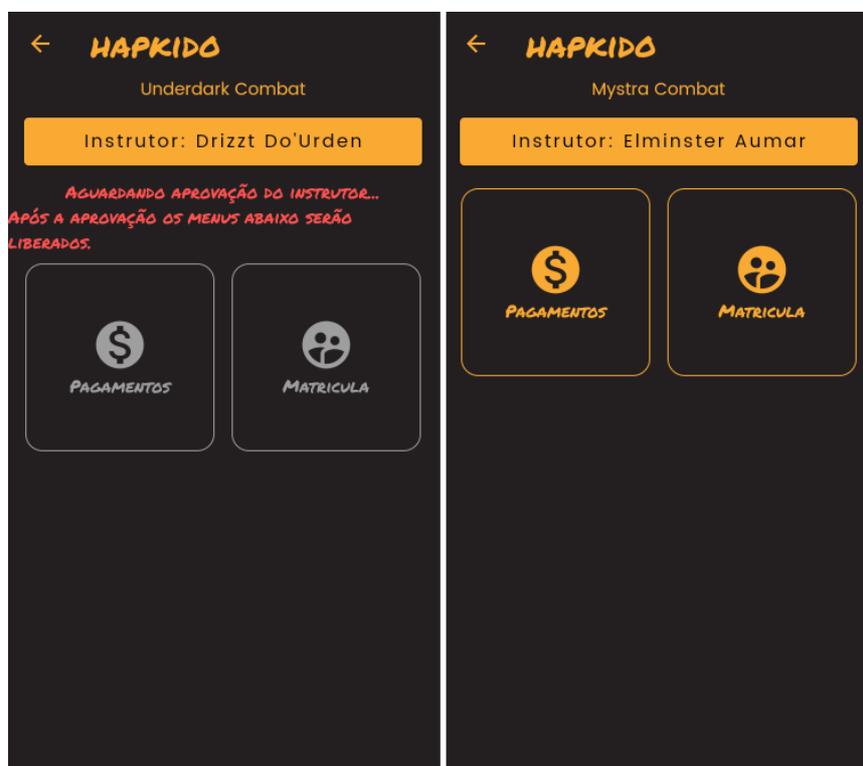
Figura 29: Telas de listagem de turmas disponíveis

Fonte: O próprio autor

## Turma

O Aluno, pode acessar as Turmas que ele está matriculado e, ao acessar uma turma onde o Aluno ainda não foi aprovado, é apresentado a tela da Turma, porém com as opções bloqueadas, e quando aprovado pelo Instrutor, o menu é apresentado, com as opções habilitadas, como pode ser observado na Figura 30.

Figura 30: Telas de turma aguardando aprovação e aprovada a matrícula

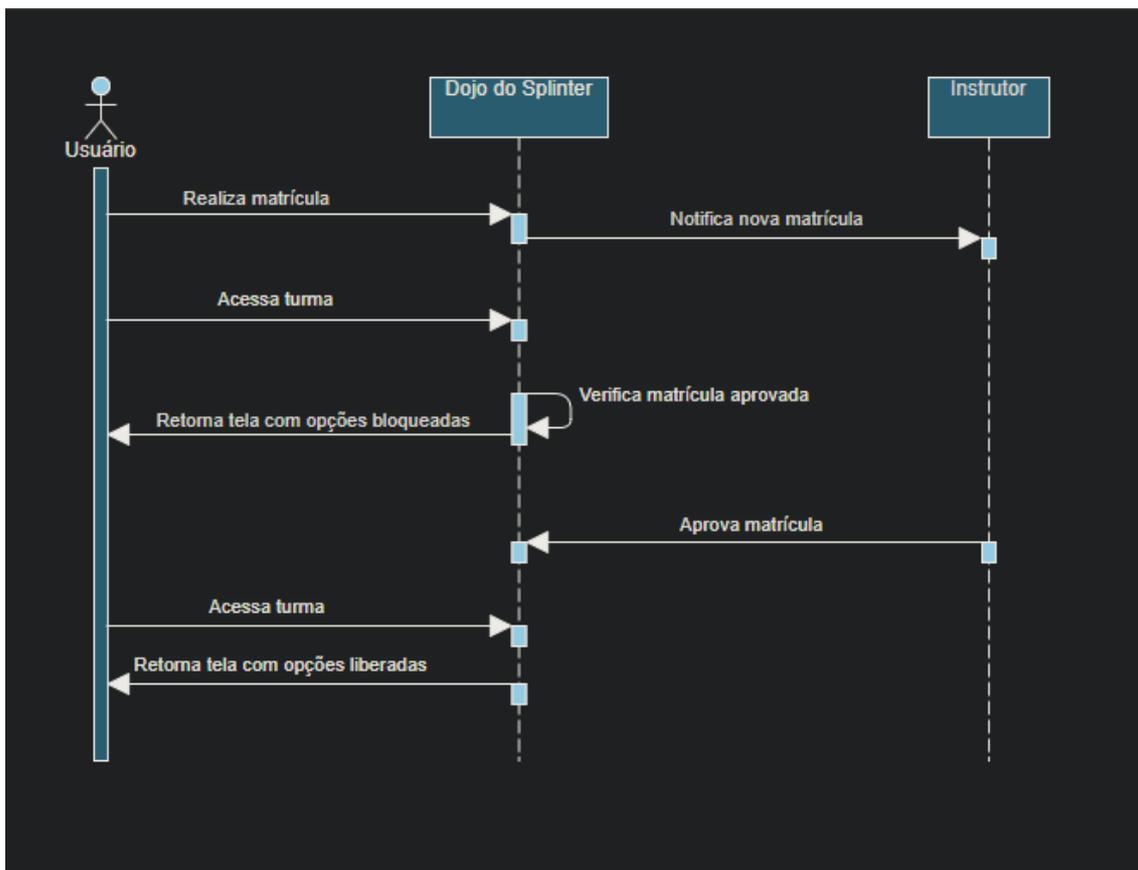


Fonte: O próprio autor.

O diagrama de sequência, representado pela figura 31, apresenta o fluxo de uso para a parte de realização de matrícula em uma turma e aprovação dela por meio do Instrutor responsável. Neste diagrama, o Usuário realiza a sua matrícula pelo sistema, este que então notifica o Instrutor de uma nova matrícula. Ao acessar a turma a qual o Usuário se matriculou, o sistema verifica se a matrícula foi aprovada, e caso não tenha sido, retorna a tela com as opções de

menus bloqueados. Após o instrutor ver a notificação no sistema, e aprovar a matrícula, ao Usuário acessar a turma o sistema retorna a tela com as opções liberadas.

Figura 31: Diagrama de sequência para matrícula em uma turma



Fonte: O próprio autor

## Pagamentos

Ao selecionar o menu de Pagamentos, é mostrado ao aluno a lista de pagamentos registrados, bem como se foi ou não pago, indicado pelos ícones. Sendo o ícone verde, apresentando um *confirmado* em verde, onde indica que o pagamento foi realizado. Caso o pagamento não esteja realizado, o sistema apresenta em tela o símbolo em vermelho de uma sineta. Esse sistema de cores, representado pela figura 32, demonstra como é apresentado para o aluno.

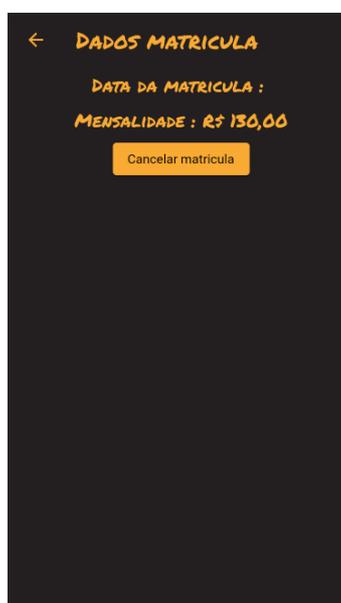
Figura 32: Listagem de pagamentos mensais



Fonte: O próprio autor

Ao selecionar a opção de Matrícula, a tela apresenta a Data de Matrícula, Mensalidade e a opção de Cancelar Matrícula, como pode ser visto na figura 33.

Figura 33: Dados da matrícula



Fonte: O próprio autor

## Eventos

A aba de Eventos, que pode ser acessada pelo menu superior esquerdo, mostra ao Aluno, todos os eventos relacionados às Turmas onde ele se encontra matriculado, e selecionando o dia, são apresentados os eventos da referida data. O instrutor consegue realizar cadastros/edições de eventos para turmas na qual ele tem vínculo. Já o aluno consegue fazer consultas de eventos nas turmas na qual tem matrícula.

Figura 34: Calendário e detalhes de eventos



Fonte: O Autor.

### 5.3. PIPELINES DE BUILD E DEPLOY

Durante o ciclo de desenvolvimento e lançamento é necessário fazer o build da aplicação, ou seja, transformar o código escrito em um pacote que pode ser instalado em um celular Android, ou para arquivos da versão Web. Em seguida ocorre o deploy, ou seja, o lançamento de uma nova versão da aplicação, seja ela disponibilizando uma nova versão na Play Store ou publicando uma nova versão no site. Uma pipeline é a representação dos passos de build e deploy em sequência, para que ao fim, se tenha como resultado uma nova versão da aplicação disponível aos usuários.

Para facilitar o ciclo de desenvolvimento, build e deploy da aplicação, foram desenvolvidos alguns scripts para que estas ações fossem automatizados, permitindo que a atualização da aplicação para uso possa ser feita a partir de qualquer lugar, sem a necessidade de instalação de um ambiente local. Para isto foi utilizado o Github Actions, uma plataforma de CI-CD (continuous integration/continuous delivery) junto com alguns scripts em bash e um makefile para gerenciar estes scripts. A Figura 35 demonstra a pipeline para o deploy da aplicação web. Para o deploy da aplicação na Play Store foi utilizado o Fastlane<sup>1</sup>, uma ferramenta de automatização de releases para aplicativos na Play Store, gerando as capturas de telas necessárias, fazendo as assinaturas de código e o release de uma nova versão da aplicação diretamente para a Play Store, por meio de comandos simples como `bundle exec fastlane android upload` - que realiza o upload da nova versão.

---

<sup>1</sup> <https://docs.fastlane.tools/>

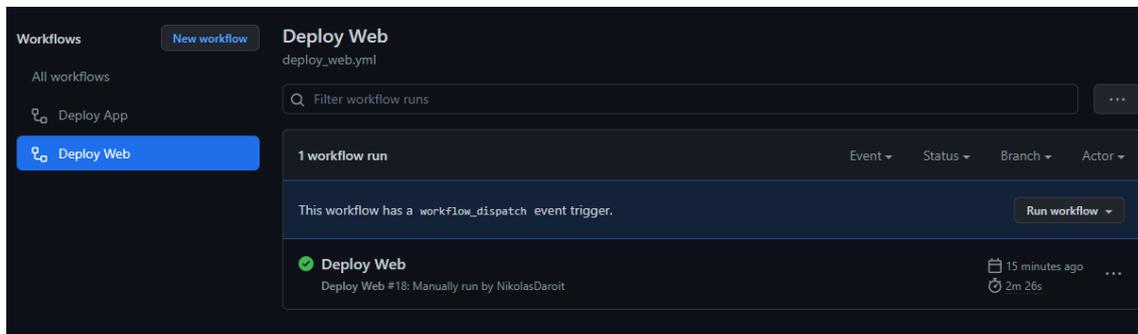
Figura 35: Arquivo de *workflow* para o deploy web no Github Actions

```
24 # Checks-out your repository under $GITHUB_WORKSPACE, so your job can access it
25 - uses: actions/checkout@v3
26
27 - uses: actions/cache@v2
28   with:
29     path: ${ runner.tool_cache }}/flutter
30     key: flutter-3.0.1-stable
31
32 - uses: subosito/flutter-action@v2
33   with:
34     channel: "stable"
35
36 - uses: actions/cache@v2
37   with:
38     path: ${ env.PUB_CACHE }
39     key: flutter-dependencies-${ hashFiles('pubspec.yaml') }
40     restore-keys: |
41       flutter-dependencies-
42
43 - name: Install dependencies
44   run: flutter pub get
45
46 - uses: actions/cache@v2
47   with:
48     path: /usr/local/bin
49     key: firebase-tools-${ hashFiles('/backend/firebase.json') }
50     restore-keys: |
51       firebase-tools-
52
53 - name: install firebase
54   run: |
55     . scripts/install_firebase.sh
56
57 - name: build web
58   run: |
59     flutter build web
60
61 - name: copy web files to public
62   run: |
63     cp -R "${GITHUB_WORKSPACE}/build/web" "${GITHUB_WORKSPACE}/backend/public"
64
65 - name: set FIREBASE_TOKEN from actions secrets
66   run: |
67     echo "FIREBASE_TOKEN=${ secrets.FIREBASE_TOKEN }" >> $GITHUB_ENV
68
69 - name: deploy to firebase
70   run: |
71     cd "${GITHUB_WORKSPACE}/backend"
72     firebase deploy --only hosting
```

Fonte: O próprio autor

Ao executar o *workflow*, os passos são executados proceduralmente, e ao fim, mostra se o todo executou completamente, obtendo sucesso, ou se falhou devido algum problema, como pode ser observado na Figura 36.

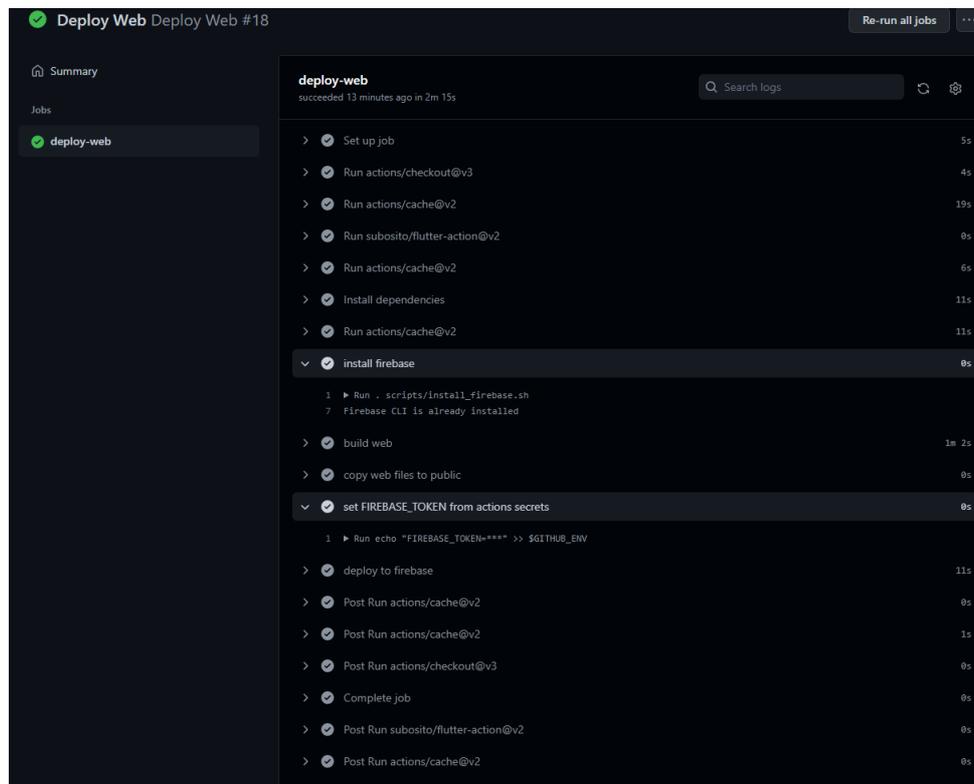
Figura 36: Exemplo de execuções de workflow no Github Actions



Fonte: O próprio autor

A ferramenta permite visualizar os detalhes da execução em cada passo, a Figura 37 mostra os detalhes da execução do *workflow* Deploy Web.

Figura 37: Passo a passo da execução do workflow Deploy Web.



Fonte: O próprio autor

Tais implementações não agregam para o usuário final, mas permitem que durante o desenvolvimento, não seja necessário instalar todas as dependências no ambiente local ou precisar realizar uma sequência de ações manuais para colocar uma nova versão disponível aos usuários, sendo assim, agilizando o processo, e trazendo confiabilidade ao desenvolvedor.

#### 5.4. TESTE DE ACEITAÇÃO

Para os testes de aceitação foi desenvolvido um roteiro de testes, disponível no Apêndice C, que busca instruir o usuário na instalação e teste do aplicativo, seja pela instalação do aplicativo na Play Store<sup>2</sup>, ou acessando a versão Web<sup>3</sup>, bem como um formulário, que visa recolher feedback construtivo e validar a usabilidade do aplicativo. O roteiro de testes foi elaborado de forma que o usuário deveria criar uma conta como aluno, e como instrutor. Como aluno deveria se matricular em uma turma, visualizar informações de sua matrícula, seus pagamentos e os eventos da academia. Como instrutor deveria aprovar a matrícula de um aluno, criar uma turma, criar um pagamento para um aluno e alterar seus dados de cadastro. As perguntas do questionário tinham por objetivo verificar se o sistema estava de acordo com os requisitos. Também foi objetivo do questionário verificar a usabilidade do sistema. Foi disponibilizado campos descritivos, para receber feedback quanto às interações enquanto aluno e instrutor, bem como sugestões de melhorias. O teste foi respondido por alunos, instrutores e administradores de uma academia de artes marciais e também por não praticantes. Os resultados obtidos com as 11 respostas ao questionário estão presentes nos gráficos das figuras.

---

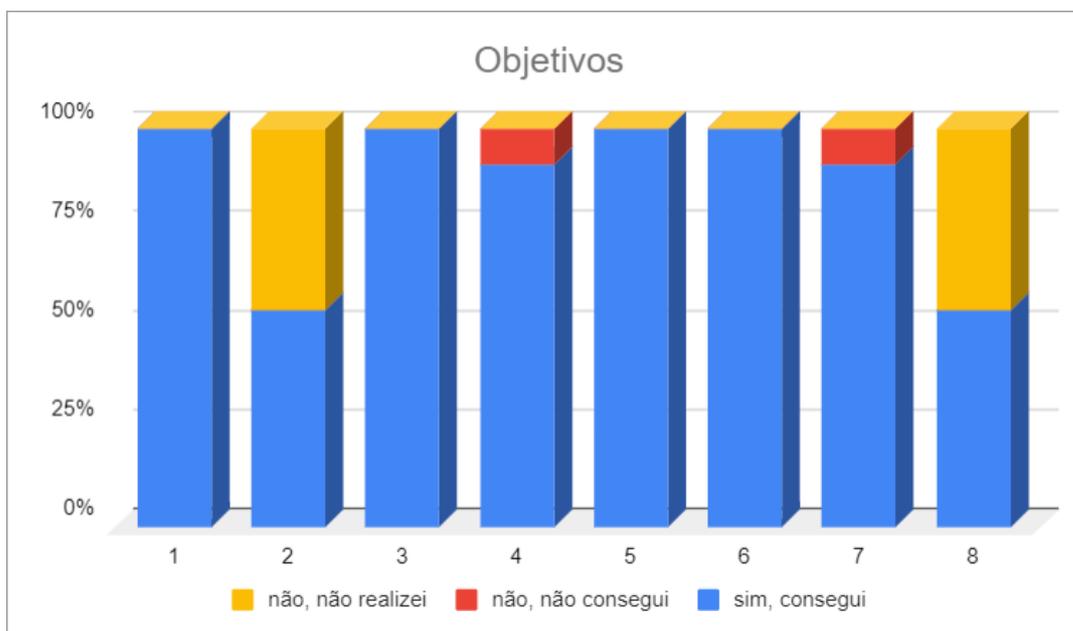
<sup>2</sup><https://play.google.com/store/apps/details?id=nikolasdaroit.com.br.dojosplinter>

<sup>3</sup> <https://dojo-do.web.app/>

A Figura 38 representa o gráfico com as respostas obtidas referentes aos objetivos requisitados aos usuários, as colunas significam respectivamente, “sim, consegui”, “não, não realizei”, “não, não consegui”. As perguntas pertencentes ao gráfico são as seguintes:

1. Você conseguiu criar um perfil de Aluno?
2. Você conseguiu criar um perfil de Instrutor?
3. Você conseguiu visualizar as turmas disponíveis?
4. Você conseguiu acessar os Eventos?
5. Você conseguiu ver os Pagamentos cadastrados para a Turma?
6. Você conseguiu ver os Dados da Matrícula?
7. Na página inicial, ao acessar o menu Turmas, você conseguiu visualizar as turmas disponíveis?
8. Na página inicial, ao acessar o menu Turmas, você conseguiu cadastrar uma Turma?

Figura 39: Gráfico gerado pelas respostas das perguntas referente aos objetivos requisitados



Fonte: O próprio autor

Dos itens que não conseguiram ser acessados e visualizados, nas perguntas 4 e 7, foi investigado a existência de algum bug que impossibilita estes menus de serem acessados, mas não foi encontrado nenhum bug relacionado. Pode ser suposto que o item Eventos não conseguiu ser acessado pois estava em outro menu, mas não foram encontrados motivos que pudessem justificar o porquê o menu das Turmas Recomendadas não conseguir ser acessado.

As figuras a seguir representam os gráficos obtidos pelas respostas de classificação do formulário, onde foi avaliado a interatividade da interface e da usabilidade para realizar a tarefa. As respostas para estas perguntas vão de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo. As perguntas pertencentes aos gráficos são as seguintes:

1. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "eventos"?

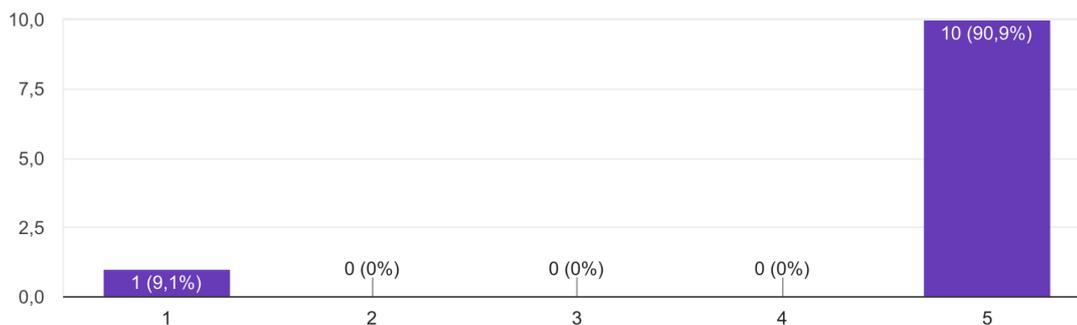
A figura 40 representa o gráfico obtido pelas respostas da primeira pergunta *“Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "eventos"?”*. O resultado obtido foi de que 90,9% dos usuários responderam “5” e 9,1% responderam “1”.

Esta avaliação 1 se destacou frente aos demais que avaliaram com 5. Foi investigado algum possível bug do sistema que pudesse ter gerado alguma dificuldade, mas nada foi encontrado. Desta forma acredita-se que possa ter ocorrido algum problema pontual, ou uma falta de atenção do usuário, visto que no sistema o menu de eventos se encontra na barra lateral .

Figura 40: Gráfico gerado pelas respostas da pergunta 1

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "eventos"?

11 respostas



Fonte: O próprio autor.

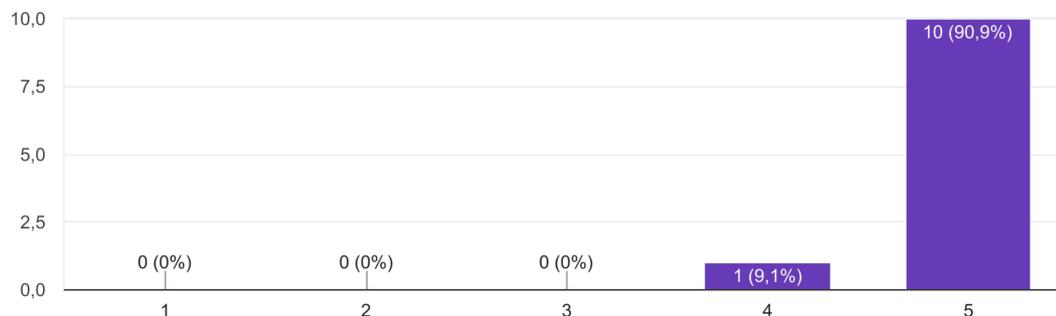
2. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Pago/Não pago"?

A figura 41 representa o gráfico obtido pelas respostas da segunda pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Pago/Não pago"?". O resultado obtido foi de que 90,9% dos usuários responderam "5" e 9,1% responderam "4".

Figura 41: Gráfico gerado pelas respostas da pergunta 2

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Pago/Não pago"?

11 respostas



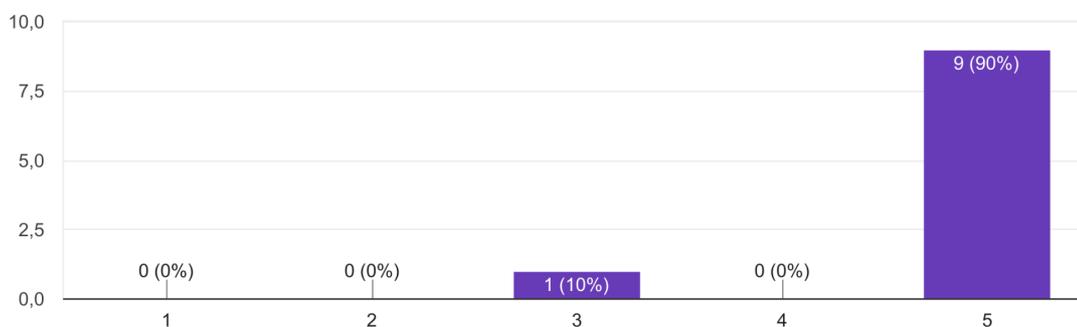
Fonte: O próprio autor

3. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia a usabilidade como aluno?

A figura 42 representa o gráfico obtido pelas respostas da terceira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia a usabilidade como aluno?". O resultado obtido foi de que 90% dos usuários responderam "5" e 10% responderam "3".

Figura 42: Gráfico gerado pelas respostas da pergunta 3

Como você avalia em geral a usabilidade como aluno? Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia a usabilidade como aluno?  
10 respostas



Fonte: O próprio autor

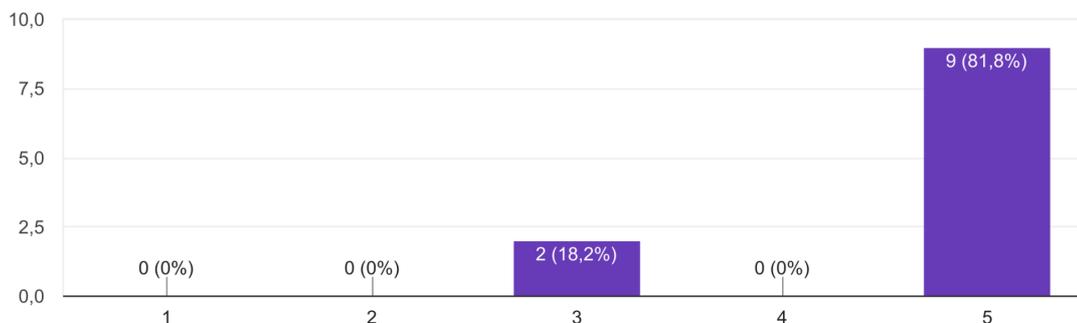
4. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Cadastro de Turma"?

A figura 43 representa o gráfico obtido pelas respostas da primeira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Cadastro de Turma"?". O resultado obtido foi de que 81,8% dos usuários responderam "5" e 18,2 % responderam "4".

Figura 43: Gráfico gerado pelas respostas da pergunta 4

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Cadastro de Turma"?

11 respostas



Fonte: O próprio autor

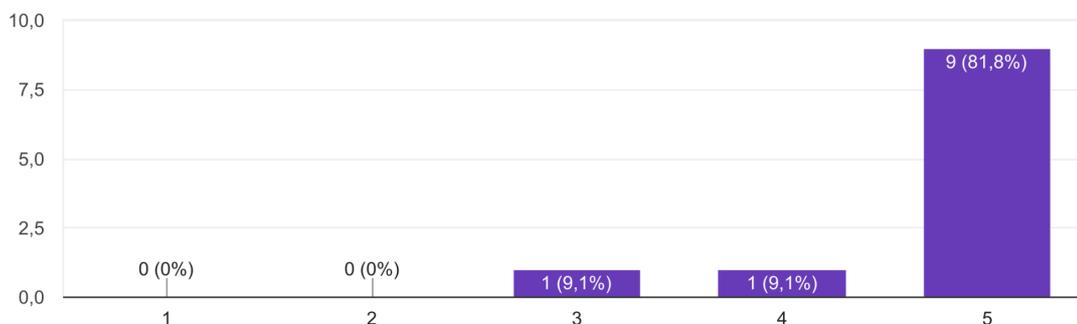
5. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Pagamentos"?

A figura 44 representa o gráfico obtido pelas respostas da primeira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Pagamentos"?". O resultado obtido foi de que 81,8% dos usuários responderam "5", 9,1% responderam "4" e 9,1% responderam "3".

Figura 44: Gráfico gerado pelas respostas da pergunta 5

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Pagamentos"?

11 respostas



Fonte: O próprio autor

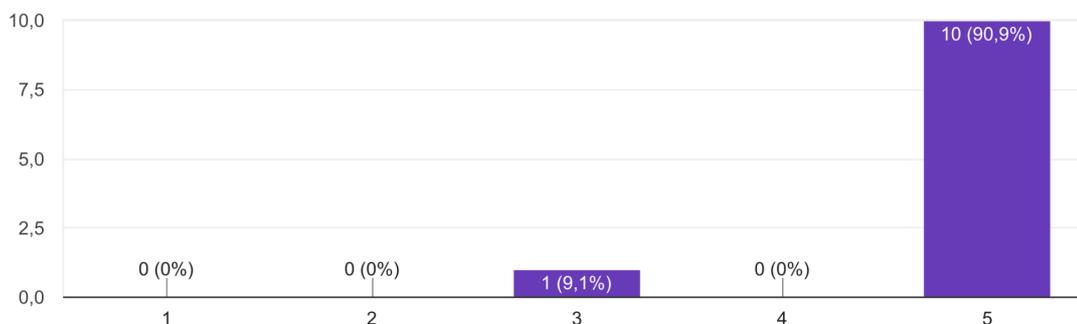
6. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Chamada"?

A figura 45 representa o gráfico obtido pelas respostas da primeira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Chamada"?". O resultado obtido foi de que 90,9% dos usuários responderam "5" e 9,1% responderam "3".

Figura 45: Gráfico gerado pelas respostas da pergunta 6

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Chamada".

11 respostas



Fonte: O próprio autor

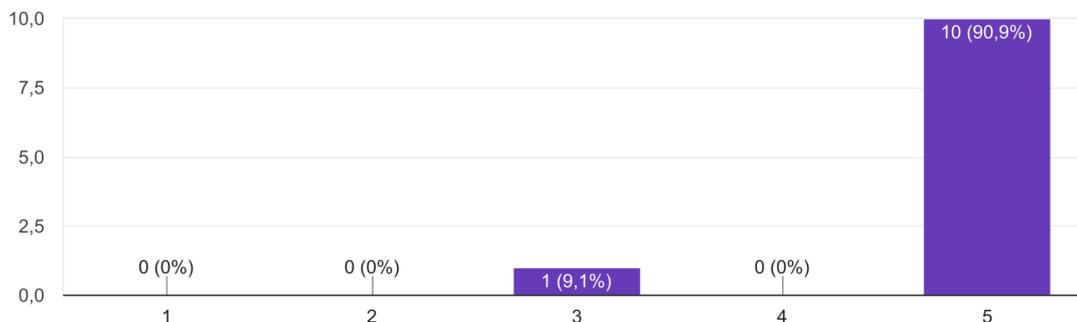
7. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos Matriculados"?

A figura 46 representa o gráfico obtido pelas respostas da primeira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos Matriculados"?". O resultado obtido foi de que 90,9% dos usuários responderam "5" e 9,1% responderam "3".

Figura 46: Gráfico gerado pelas respostas da pergunta 7

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos Matriculados".

11 respostas



Fonte: O próprio autor

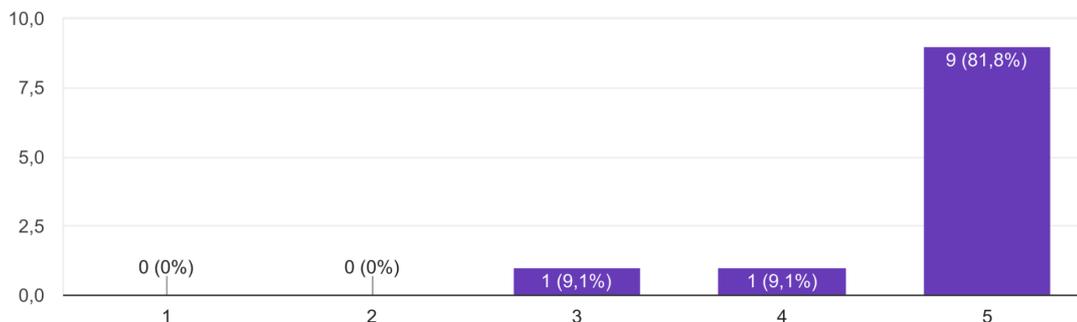
8. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos Pendentes"?

A figura 47 representa o gráfico obtido pelas respostas da primeira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos Pendentes"?". O resultado obtido foi de que 81,8% dos usuários responderam "5", 9,1% responderam "4" e 9,1% responderam "3".

Figura 47: Gráfico gerado pelas respostas da pergunta 8

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos Pendentes"?

11 respostas



Fonte: O próprio autor

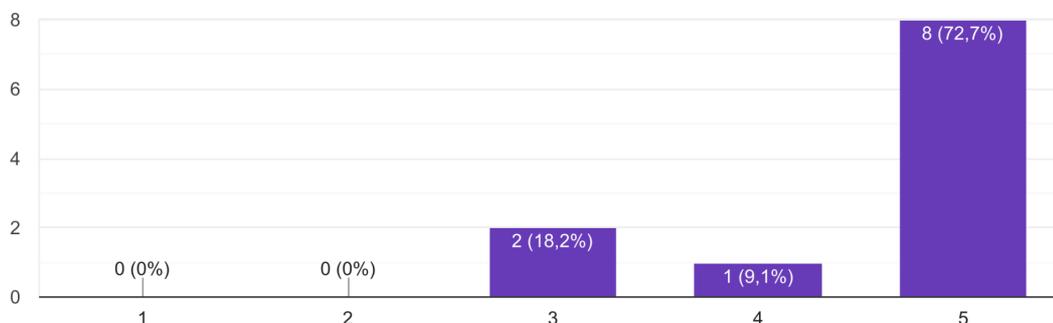
9. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Novo Evento"?

A figura 48 representa o gráfico obtido pelas respostas da primeira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Novo Evento"?". O resultado obtido foi de que 72,7% dos usuários responderam "5", 9,1% responderam "4" e 18,2% responderam "3".

Figura 48: Gráfico gerado pelas respostas da pergunta 9

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Novo Evento"?

11 respostas



Fonte: O próprio autor

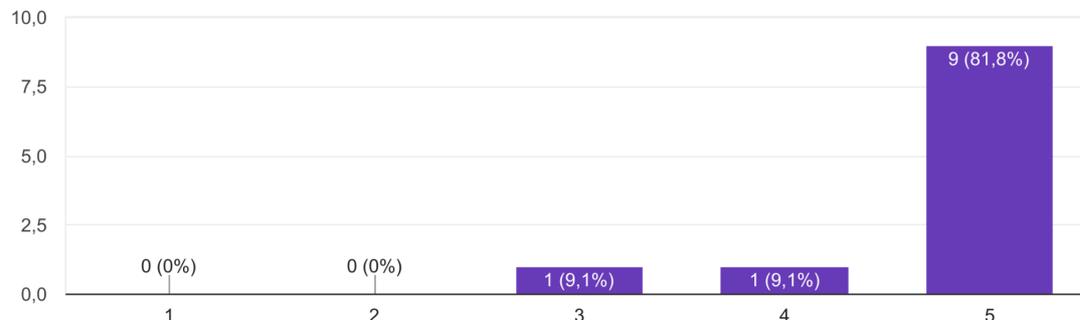
10. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos cadastrados"?

A figura 49 representa o gráfico obtido pelas respostas da primeira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos cadastrados"?". O resultado obtido foi de que 81,8% dos usuários responderam "5", 9,1% responderam "4" e 9,1% responderam "3".

Figura 49: Gráfico gerado pelas respostas da pergunta 10

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia o item "Alunos cadastrados"?

11 respostas



Fonte: O próprio autor

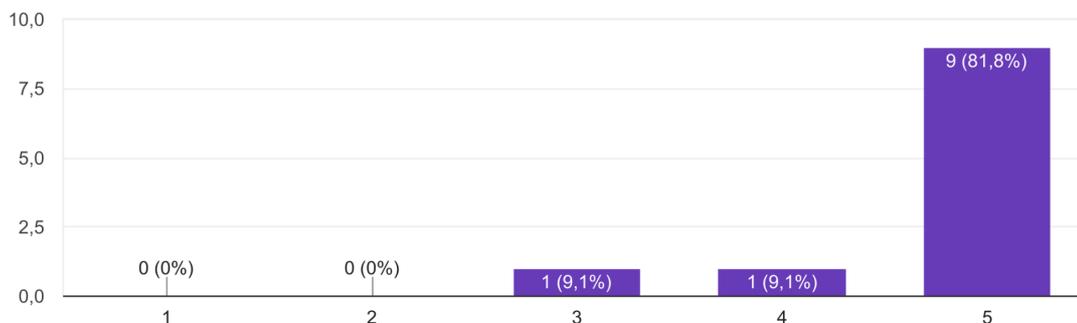
11. Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia a usabilidade como instrutor?

A figura 50 representa o gráfico obtido pelas respostas da primeira pergunta "Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia a usabilidade como instrutor?". O resultado obtido foi de que 81,8% dos usuários responderam "5", 9,1% responderam "4" e 9,1% responderam "3".

Figura 50: Gráfico gerado pelas respostas da pergunta 11

Numa escala de 1 a 5, sendo 1 muito confuso, 2 confuso, 3 bom, 4 intuitivo e 5 muito intuitivo, como você avalia a usabilidade como instrutor?

11 respostas



Fonte: O próprio autor

O teste de aceitação com os usuários demonstrou que o Dojo do Splinter é um sistema recomendável, avaliado com boas notas pela grande maioria dos usuários. Os campos de feedback foram preenchidos por alguns usuários com algumas sugestões de usabilidade ou sobre a visualização de algumas informações ou como *“Muito boa, fácil navegação ao usuário.”*, *“Acredito que o app é extremamente intuitivo e prático.”*, *“O app atende a necessidades do cotidiano de uma forma ampla e eficiente.”*. Informalmente, alguns usuários relataram uma boa experiência com o sistema, tendo tido uma experiência agradável. Dentre os feedbacks recebidos sobre a visualização de algumas informações estão: *“Acho que as turmas deveriam aparecer com o nome posto na descrição, assim facilitaria a escolha mais rápida de qual turma se deseja acessar. Nos dados dos alunos, senti falta de qual informação está sendo visualizada. Muitas são claras por si só. Mas fiquei confuso por não saber a que se referia a primeira data (matrícula) e na parte de problema de saúde. Acharia muito útil se fosse possível cadastrar alunos sem que estes tivessem o APP instalado, ampliando assim para todos os alunos crianças e idosos que muitas vezes não possuem acesso ao aplicativo ou não sabem como utilizá-lo. O layout está sensacional.”*, *“Acredito que poderia melhorar a tela onde mostra as turmas. Ao invés de ter que passar para o lado, que fosse utilizado o espaço ainda sem*

*uso da tela.”, “Eu gostei muito. Minha única sugestão seria, por exemplo, na parte em que mostra os horários disponíveis para aula, ao invés de ser uma lista, ficar em grade.”.*

Acerca do feedback recebido sobre a falta de informações em dadas partes, foi apontado que na visualização da turma, seria melhor conter a descrição dela, pois assim ficaria mais fácil de identificar em casos onde o aluno/instrutor possui mais de uma turma da mesma arte marcial. Além disso, foi apontado sobre a falta de uma legenda para os campos no menu de cadastro de usuário, pois quando a informação está preenchida, ocultando o *placeholder* (o texto de ajuda de um campo), não é possível identificar a que o campo se refere. Também foi apontado que o layout de turmas disponíveis seria melhor recebido caso apresentado em grade ao invés de um *carousel* (como uma sequência de slides que pode ser arrastada).

As seguintes sugestões serão adicionadas como trabalhos futuros:

- Reorganização da lista de turmas recomendadas, utilizando outro modo de apresentação para facilitar a visualização de mais turmas;
- Possibilidade de cadastro de aluno pelo instrutor, permitindo que crianças e idosos que não possuem acesso ao aplicativo ou que não sabem como utilizá-lo, sejam cadastrados;

As seguintes sugestões foram implementadas após os testes:

- Foi adicionada a descrição da turma na listagem da turma, o ajuste está representado na Figura 51;
- Foi adicionado o valor da mensalidade na listagem de pagamentos, o ajuste está representado na Figura 52;

Figura 51: Detalhes nos cards das turmas matriculadas



Fonte: O próprio autor.

Figura 52: Valor da mensalidade na lista de pagamentos



Fonte: O próprio autor

Assim, percebe-se que o teste aplicado demonstrou o adequado funcionamento do sistema, bem como identificou alguns pontos com pontuais melhorias a serem aplicadas. Algumas sendo alteradas durante a interação do desenvolvimento, e outras sendo identificados como possíveis trabalhos futuros. Na sequência serão apresentadas as considerações finais deste trabalho.

## 6. CONSIDERAÇÕES FINAIS

A tecnologia vem avançando a cada dia e isso ocasionou em sua utilização em todos os cenários do cotidiano, o que não foi diferente no que diz respeito a cuidados com o corpo e prática de esportes, como as artes marciais. Principalmente em relação ao dinamismo das academias, onde tentamos facilitar o processo, dando o foco ao instrutor a sua atividade final.

O sistema desenvolvido permite que alunos de artes marciais façam cadastro e escolham turmas para se matricular, podendo visualizar e acompanhar o calendário de eventos da academia de arte marcial, bem como se manter atualizado quanto a seus status de pagamentos em aberto ou concluídos para cada uma de suas turmas. Nesse mesmo sentido, permite que os instrutores de academias consigam aprovar matrículas de alunos, cadastrar e alterar dados de pagamentos e gerenciar eventos da academia.

As escolhas de tecnologias para o desenvolvimento deste projeto visam a integração de funções necessárias para o aplicativo com opções que se assemelhavam mais ao nativo do sistema. Um dos cuidados foi de permitir a sincronização offline de dados, bem como um baixo tráfego de dados, para que seja possível sua utilização em ambientes com velocidade de conexão limitada.

O resultado dos testes realizados se mostraram muito satisfatórios, e bem recebido pelos instrutores da academia, tendo tido uma avaliação de “muito intuitivo” por 85% dos usuários envolvidos no teste.

O desenvolvimento de um aplicativo que facilite a rotina nas academias, torna o atendimento mais rápido e eficaz, tendo em vista que tanto os instrutores quanto os alunos podem realizar consultas através do aplicativo em seus próprios celulares. Com isto as melhorias proporcionadas pelo desenvolvimento deste sistema se mostraram importantes para a academia parceira deste desenvolvimento

## 6.1. TRABALHOS FUTUROS

Nesta sessão, serão levantadas futuras melhorias e soluções que poderão ser implementadas no Dojo do Splinter.

### 6.1.1. Verificação e Aprovação do Instrutor cadastrado

Quando um Instrutor é cadastrado, seu perfil é criado como inativo, e resta ao administrador fazer a aprovação por meio do banco de dados, para liberar o Instrutor. A proposta é construir uma interface administrativa para realizar este tipo de aprovação por um usuário..

### 6.1.2. Tela de relatórios financeiros para o instrutor

Apesar do controle financeiro por usuário e por turma, o aplicativo não oferece uma visualização centralizada do financeiro do instrutor ou da academia. A proposta é construir uma interface, que centralize os dados de cada uma das turmas do instrutor.

### 6.1.3. Alteração da visualização de turmas recomendadas

Com base no feedback coletado, percebeu-se que a apresentação das turmas utilizando um *carousel* não foi tão bem recebida. A proposta é reorganizar a lista de turmas recomendadas, utilizando uma listagem vertical de turmas, disponibilizando uma *modal* (uma janela sobreposta a tela) com as informações adicionais que serão apresentadas ao selecionar a turma.

### 6.1.4. Cadastro de aluno pelo perfil de Instrutor

Para permitir o cadastro de alunos que não tenham acesso ao sistema ou que tenham dificuldade em utilizá-lo, é proposto a criação de um submenu para o Instrutor, que permita o cadastro de um novo usuário.

## 7. REFERÊNCIAS

DIGITAL HOUSE. **O que é Flutter e como funciona.** Disponível em: <<https://www.digitalhouse.com/br/blog/o-que-e-flutter-e-como-funciona>>. Acesso em: 31 ago 2021.

DRAKE, Joshua J. *et al.* **Android Hacker's Handbook.** John Wiley & Sons, v. 3, f. 288, 576 p. 2014.

FIRTMAN, Maximiliano. **Programming the Mobile Web: Reaching Users on iPhone, Android, BlackBerry, Windows Phone, and more.** 2ª. ed. "O'Reilly Media, Inc.", v. 3, f. 387, 774 p. 2013.

GEEK HUNTER. **Firestore o que é e quando usar no desenvolvimento Mobile.** Disponível em: <<https://blog.geekhunter.com.br/firebase-o-que-e-quando-usar-no-desenvolvimento-mobile/>>. Acesso em: 31 ago 2021.

GEEK HUNTER. **Introdução ao Flutter, o framework da google.** Disponível em: <[https://blog.geekhunter.com.br/introducao-ao-flutter-o-framework-do-google/#A\\_linguagem\\_Dart](https://blog.geekhunter.com.br/introducao-ao-flutter-o-framework-do-google/#A_linguagem_Dart)>. Acesso em: 31 ago 2021.

HAGOS, Ted. **Learn Android Studio 3 with Kotlin: Efficient Android App Development.** Apress, v. 3, f. 235, 469 p. 2018. MEDIUM. **Aplicativos Híbridos – Vilões ou mocinhos.** Disponível em: <<https://medium.com/trainingcenter/aplicativos-h%C3%ADbridos-vil%C3%B5es-ou-mocinhos-b22276f523c0>>. Acesso em: 31 ago 2021.

GRIFFITH, Chris. **Mobile App Development with Ionic, Revised Edition: CrossPlatform Apps with Ionic, Angular, and Cordova.** 1ª. ed. "O'Reilly Media, Inc.", v. 3, f. 146, 292 p. 2017.

MENDES, Mariana Ribeiro *et al.* **Desenvolvimento híbrido versus desenvolvimento nativo de aplicativos móveis.** In: VII Semana de Ciência e Tecnologia do IFMG campus Bambuí. 5p. 2014.

NIELSEN, Jakob; BIDU, Amy. **Usabilidade Móvel.** Tradução Sérgio Facchin. 1ª. ed. São Paulo: Elsevier Brasil, v. 2, f. 108, 2014. 215 p. Tradução de: Mobile Usability.

PAGOTTO, Tiago; GONÇALVES, José. **Scrum Solo: Processo de software para desenvolvimento individual.** Laboratório de Inovação – UFTPR. 2014.

SALEH, Hazem *et al.* **Mobile Application Development: JavaScript Frameworks.** Packt Publishing Ltda., v. 3, f. 300, 2017. 599 p

SHOWMETECH. **Evolução do Android na tela Inicial e Navegação.** Disponível em: <<https://www.showmetech.com.br/a-evolucao-do-android-na-tela-inicial-e-navegacao/>>. Acesso em: 31 ago 2021.

SLIDESHARE. **Linguagem Dart.** Disponível em: <<https://pt.slideshare.net/proflborges/linguagem-dart-google>>. Acesso em:

SOFTWEB SOLUTIONS. **All you need to know before hiring a Flutter developer.** Disponível em: <<https://www.softwebsolutions.com/resources/hire-a-flutter-developer.html>>. Acesso em: 31 ago 2021.

SLIDESHARE. **Computadores quânticos e dispositivos portáteis.** Disponível em: <<https://pt.slideshare.net/gficherFDA/computadores-quanticos-e-dispositivos-portteis>>. Acesso em: 31 ago 2021.

SOUGATA, Mukherjea. **Mobile Application Development, Usability, and Security.** IGI Global, f. 160, 2017. 320 p.

TREINA WEB. **O que é Dart?** Disponível em: <<https://www.treinaweb.com.br/blog/o-que-e-dart>>. Acesso em: 31 ago 2021.

WALRATH, Kathy.. **Dart asynchronous programming: Isolates and event loops.** In:MEDIUM

STACK OVERFLOW. **Does Dart support functional programming?** Disponível em: <<https://stackoverflow.com/questions/16746807/does-dart-support-functional-programming/65024423>> Acesso em: 31 ago 2021.

FIREBASE. In; Google Products. Disponível em: <<https://firebase.google.com/products/realtime-database?hl=pt-br>> Acesso em: 31 ago 2021.

REDE GLOBO. **Criadas para a autodefesa, as artes marciais têm origem na pré-história.** Acesso em 28 set 2021. Disponível em <<http://redeglobo.globo.com/globociencia/noticia/2012/09/criadas-para-autodefesa-artes-marciais-tem-origem-na-pre-historia.html>>. 2012.

YOUNG, David. C. **Software Development Methodologies.** In: Alabama Supercomputer Authority. Disponível em: <[asc.edu](http://asc.edu)>. 2013.

S/A. **Dart versus Node js fastest programs vs Dart JIT vs Java vs JavaScript.** In: The Computer Language Benchmarks Game. Acesso em: 28 set 2021. Disponível em: <[benchmarksgame/fastest/dart.html](http://benchmarksgame/fastest/dart.html)>. 2021.

## APÊNDICE A - FICHA DE CADASTRO

Figura 9



# HAPKIDO UYK

Data da Inscrição: \_\_\_\_/\_\_\_\_/\_\_\_\_

Nome: \_\_\_\_\_

Data de Nasc.: \_\_\_\_/\_\_\_\_/\_\_\_\_ Sexo: \_\_\_\_\_

Ocupação: \_\_\_\_\_

RG: \_\_\_\_\_ CPF: \_\_\_\_\_

Fone: \_\_\_\_\_ Celular: \_\_\_\_\_

E-mail: \_\_\_\_\_

End.: \_\_\_\_\_

Bairro: \_\_\_\_\_ Cidade: \_\_\_\_\_

Responsável: \_\_\_\_\_

Fone: (Mãe): \_\_\_\_\_ (Pai): \_\_\_\_\_

Problema de Saúde: \_\_\_\_\_

Motivo da Prática: \_\_\_\_\_

Treinamentos Anteriores: \_\_\_\_\_

Autorizo que meu telefone/e-mail seja compartilhado com os colegas de treino.

Fonte: Vitor

## APÊNDICE B - FICHA DE PAGAMENTO

DATA: ____/____/____	DATA: ____/____/____	DATA: ____/____/____	
VALOR: _____	VALOR: _____	VALOR: _____	
PAG: _____	PAG: _____	PAG: _____	
DATA: ____/____/____	DATA: ____/____/____	DATA: ____/____/____	
VALOR: _____	VALOR: _____	VALOR: _____	
PAG: _____	PAG: _____	PAG: _____	
DATA: ____/____/____	DATA: ____/____/____	DATA: ____/____/____	<p><b>NOME:</b> <i>Nikolas</i></p> <p><b>FONE:</b> <i>Wing Chun</i></p> <p>RUI ANTÔNIO DE OLIVEIRA, 306 COHAB - SAPUCAIA DO SUL</p>
VALOR: _____	VALOR: _____	VALOR: _____	
PAG: _____	PAG: _____	PAG: _____	

## APÊNDICE C - INSTRUÇÕES DE TESTE

### Obrigado por disponibilizar seu tempo para testar o aplicativo Dojo do Splinter!

O Dojo do Splinter é um aplicativo desenvolvido com o intuito de auxiliar no gerenciamento de pequenas academias de artes marciais.

Ao final há um pequeno formulário ( <https://forms.gle/rVvNab8pQCNmowYz7> ) para recolher feedback, é bem importante para meu trabalho que você o preencha.

Siga os passos abaixo para baixar, instalar e utilizar o aplicativo. Caso ainda fique alguma dúvida, por favor, entre em contato! O período de testagem vai até dia 25/06/2022.

1. Baixe o aplicativo na [Play Store](#) ou acesse pelo [versão web](#);
2. Crie uma conta;
  - a. Verifique o email (ele acaba indo para o spam geralmente)
  - b. Crie a conta como Aluno
  - c. Faça login com a conta criada;
  - d. Visualize as artes marciais recomendadas e escolha se matricular em alguma turma do Instrutor Haven;
  - e. Acesse a turma em que se matriculou;
    - i. Será necessário fazer login como o Instrutor Haven para aprovar o acesso a este usuário;
3. Acesse a conta de Instrutor com o usuário e senha abaixo
  - a. Usuário: [instrutordojo@gmail.com](mailto:instrutordojo@gmail.com) Senha: 35791113
  - b. Você verá uma notificação sobre o menu Turmas, selecione-o;
  - c. Selecione a turma das 12:30, deverá haver uma notificação sobre ela também;
  - d. Você verá uma notificação sobre o menu Alunos, selecione-o;
  - e. Selecione a aba Alunos Pendentes, e selecione então seu usuário
  - f. Clique em Aprovar
4. Na aba de alunos matriculados, selecione seu usuário, visualize se os dados estão
 



 corretos clicando no icone \_\_\_\_\_, e altere o valor de sua matricula clicando no
 



 icone \_\_\_\_\_;
  5. Acesse o menu de Pagamentos e crie um novo pagamento para seu usuário;
  6. Acesse o menu de Chamada e marque presença para seu usuário;
  7. Na barra lateral encerre a sessão, e faça login como usuário
  8. Após aprovado o acesso, os campos serão desbloqueados;
    - a. Visualize as opções de Pagamentos e Matrícula;
    - b. Na barra lateral, acesse a aba de Eventos e visualize os eventos cadastrados;
    - c. Na barra lateral, visualise e edite o perfil;

- 
9. Ao final da sua testagem, por gentileza, acessar e responder o formulário através deste [link](#).

Mais uma vez, obrigado por auxiliar em minha conclusão de curso!  
Atenciosamente,  
Nikolas Daroit

## APÊNDICE D - DOCUMENTO DE VISÃO

# Documento de Visão

 Data	 Versão	 Modificação	 Autor
@August 8, 2019	1	<u>Criação do Documento</u>	Nikolas Daroit
@October 10, 2019	1.1	<u>Adição dos requisitos</u>	Nikolas Daroit
@June 26, 2022	1.2	<u>Atualização do Documento</u>	Nikolas Daroit

## 1. Objetivo

O propósito deste documento é coletar, analisar e definir os requisitos de alto nível, e caracterizar o sistema, com o foco sendo nas funcionalidades objetivo dos usuários-alvo. O documento define uma visão do ambiente geral de processos desenvolvidos para o sistema, fornecendo a todos os envolvidos uma descrição compreensível deste e de suas funcionalidades. O Documento de Visão documenta as necessidades e funcionalidades do sistema

## 2. Descrição do Problema

O problema de	Falta de gerenciamento de turmas
Afeta	?
Cujo impacto é	Falta de informação e atraso de pagamentos
Uma boa solução seria	Disponibilização de informações de pagamento e eventos online

## 3. Definição das Partes Interessadas

### Instrutores

Nome	Descrição	Responsabilidades
------	-----------	-------------------

Nome	Descrição	Responsabilidades
Instrutores de academias de artes marciais	São pessoas treinadas em uma arte marcial, e habilitadas a ensinarem	Responsável por utilizar o software desenvolvido para criar turmas, além de gerenciar os alunos e eventos.

### Alunos

Nome	Descrição	Responsabilidades
Alunos de academias de artes marciais	São pessoas que praticam uma arte marcial, e possuem um instrutor para lhes ensinar.	Responsável por utilizar o software desenvolvido para criar visualizar suas turmas, seus pagamentos e visualizar eventos da academia.

### Administrador

Nome	Descrição	Responsabilidades
Administrador do sistema		Responsável por aprovar o cadastro dos instrutores.

### Time de Desenvolvimento

Nome	Descrição	Responsabilidades
Nikolas Telvi Nahra Daroit	Gerente de projeto e desenvolvedor.	Responsável pela elaboração do Documento de visão, Criação dos diagramas de casos de uso e de classe, desenvolver e gerenciar o projeto.

## 4.Descrição do Produto

<b>Para</b>	<b>Instrutores e Alunos de academias de artes marciais</b>
Que	Procuram manter seus pagamentos em dia e estar a par dos eventos da academia
O	"Dojo do Splinter"
É um	software de gerenciamento de turmas de artes marciais
Que	facilita a organização e centraliza informações sobre turmas de artes marciais
Ao contrário	dos exemplos já consolidados na internet, <a href="#">iDojo</a> , <a href="#">DojoWeb</a> , <a href="#">iBlackBelt</a> , que oferecem soluções empresariais para o gerenciamento de academias, mas se tornam complexos demais para pequenas academias por necessitar de muita mão de obra para suas configurações
Nosso produto	Diferencia-se dos outros aplicativos pelo fato de ser único por academia, sendo assim já é pré-configurado, permitindo que os alunos e instrutores tenham mais velocidade para o primeiro acesso e futura utilização.

## 5. Necessidades e Funcionalidades do Produto

1. Gerenciar perfil
  - a. Cadastro de usuário aluno
    - i. Confirmação de email
    - ii. Modificar perfil
    - iii. Alterar e recuperar senha
  - b. Cadastro de usuário instrutor
    - i. Alterar nome de exibição
    - ii. Alterar e recuperar senha

- iii. Aprovar cadastro do instrutor
- 2. Gerenciar Turma
  - a. Gerenciar pagamentos
  - b. Gerenciar chamada
  - c. Visualizar alunos
    - i. Aprovar matrícula de alunos
  - d. Gerenciar matrícula
- 3. Cadastrar eventos

## 6. Proposta Escolhida

O sistema será desenvolvido utilizando o framework Flutter para a linguagem Dart. A interface web e o aplicativo Mobile, serão gerados utilizando o framework Flutter. Os bancos de dados utilizados serão produtos do Firebase, sendo eles: Authentication, Realtime Database e Firestore, sendo os dois últimos bancos de dados não relacionais orientado a documentos.

## 7. Restrições

Funcionais/Negócio:

Instrutor:

- O Instrutor só poderá alterar as turmas que criou.
- O Instrutor só poderá alterar a matrícula de Alunos cadastrados na turma que criou.
- O Instrutor só poderá acessar o sistema depois de seu acesso ser liberado pelo administrador.
- O Instrutor só poderá cadastrar um email que ainda não existe no sistema.

Aluno:

- O Aluno só poderá visualizar detalhes da turma que estiver matriculado se for aprovado pelo Instrutor.
- O Aluno só poderá visualizar eventos das turmas que está matriculado.
- O Aluno só poderá acessar o sistema depois de ter confirmado seu email.
- O Aluno só poderá cadastrar um email que ainda não existe no sistema.