



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE AUTOMAÇÃO E SISTEMAS
CURSO DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Fernando Leite de Sousa

Criação de aplicativo para uma comunidade digital de futebol: Engenharia aplicada às novas formas de consumir o esporte mais popular do Brasil

Florianópolis
2023

Fernando Leite de Sousa

Criação de aplicativo para uma comunidade digital de futebol: Engenharia aplicada às novas formas de consumir o esporte mais popular do Brasil

Relatório final da disciplina DAS5511 (Projeto de Fim de Curso) como Trabalho de Conclusão do Curso de Graduação em Engenharia de Controle e Automação da Universidade Federal de Santa Catarina em Florianópolis.

Orientador: Prof. Ronaldo dos Santos Mello, Ph.D.

Florianópolis
2023

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

de Sousa, Fernando Leite de
Criação de aplicativo para uma comunidade digital de
futebol: Engenharia aplicada às novas formas de consumir o
esporte mais popular do Brasil / Fernando Leite de de
Sousa ; orientador, Ronaldo dos Santos Mello, 2023.
77 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro Tecnológico,
Graduação em Engenharia de Controle e Automação,
Florianópolis, 2023.

Inclui referências.

1. Engenharia de Controle e Automação. 2. Scrum. 3.
Futebol. 4. Design Thinking. 5. Flutter. I. Mello, Ronaldo
dos Santos. II. Universidade Federal de Santa Catarina.
Graduação em Engenharia de Controle e Automação. III. Título.

Fernando Leite de Sousa

Criação de aplicativo para uma comunidade digital de futebol: Engenharia aplicada às novas formas de consumir o esporte mais popular do Brasil

Esta monografia foi julgada no contexto da disciplina DAS5511 (Projeto de Fim de Curso) e aprovada em sua forma final pelo Curso de Graduação em Engenharia de Controle e Automação

Florianópolis, 14 de dezembro de 2023.

Prof. Marcelo De Lellis Costa de Oliveira, Dr.
Coordenador do Curso

Banca Examinadora:

Prof. Ronaldo dos Santos Mello, Ph.D.
Orientador
UFSC/CTC/INE

Prof. Rodrigo Castelan Carlson, Dr.
Avaliador
Universidade Federal de Santa Catarina

Prof. Eduardo Camponogara, Ph.D.
Presidente da Banca
UFSC/CTC/DAS

Este trabalho é dedicado aos meus queridos pais e
amigos.

AGRADECIMENTOS

Gostaria de expressar minha sincera gratidão a todos que contribuíram para realização deste trabalho e a todos que fizeram parte da minha caminhada na Universidade Federal de Santa Catarina. Agradeço aos meus pais, por sempre me incentivarem a buscar o conhecimento e me apoiarem durante este trabalho, bem como durante o curso de graduação. Ao meu irmão, Felipe, que esteve comigo e também foi apoiador deste projeto. À Mari, pelo seu incansável apoio durante este projeto, e também pela inspiração para concluir este trabalho. E aos colegas e amigos que fiz no curso na Universidade Federal de Santa Catarina, sem vocês nada disso seria possível.


*"Não há um tipo criativo. Como se a criatividade fosse um verbo, um verbo que consome muito tempo. Trata-se de pegar numa ideia que temos na cabeça e transformá-la em algo real. E isso vai ser sempre um processo longo e difícil. Se o estivermos a fazer bem, vai parecer trabalho."
(Glaser)*

DECLARAÇÃO DE PUBLICIDADE

Florianópolis, 01 de dezembro de 2023.

Na condição de representante do Isto é Avaí na qual o presente trabalho foi realizado, declaro não haver ressalvas quanto ao aspecto de sigilo ou propriedade intelectual sobre as informações contidas neste documento, que impeçam a sua publicação por parte da Universidade Federal de Santa Catarina (UFSC) para acesso pelo público em geral, incluindo a sua disponibilização *online* no Repositório Institucional da Biblioteca Universitária da UFSC. Além disso, declaro ciência de que o autor, na condição de estudante da UFSC, é obrigado a depositar este documento, por se tratar de um Trabalho de Conclusão de Curso, no referido Repositório Institucional, em atendimento à Resolução Normativa n° 126/2019/CUn.

Por estar de acordo com esses termos, subscrevo-me abaixo.

 Documento assinado digitalmente
FERNANDO LEITE DE SOUSA
Data: 01/03/2024 15:05:03-0300
Verifique em <https://validar.iti.gov.br>

Fernando Leite de Sousa
Isto é Avaí

RESUMO

Este trabalho descreve o desenvolvimento prático de um projeto de software para o Canal Isto é Avaí (empresa-caso). Trata-se da aplicação de engenharia de automação na prestação de serviços do ramo de cultura e entretenimento, a fim de facilitar a relação entre a criação de conteúdo da empresa em questão e seu público. O objetivo deste trabalho é demonstrar o potencial da automação não apenas no campo industrial, mas também em outros setores da economia: nesse caso, atuando na criação de um aplicativo para uma comunidade digital de futebol: uma das novas formas de consumir o esporte mais popular do Brasil. Ao adotar uma abordagem centrada no ser humano, através do Design Thinking, o projeto buscou compreender as necessidades e experiências dos usuários, resultando em soluções mais alinhadas com suas demandas reais e também com as demandas contemporâneas de resolução de problemas. A metodologia Scrum, utilizada em conjunto, viabilizou entregas ágeis e de valor para o projeto. A aplicação Progressive Web App (PWA) desenvolvida neste trabalho utilizou o framework Flutter, que permite criar aplicativos nativos para iPhone Operating System (iOS) e Android a partir de um único código-fonte. O Flutter utiliza a linguagem de programação Dart, que deriva do Javascript (esta que é uma linguagem de programação orientada a objetos).

Palavras-chave: Automação. Scrum. Design Thinking. Flutter. Javascript. Canal Isto é Avaí. Futebol.

ABSTRACT

This work describes the practical development of a software project for Canal Isto é Avaí (a case company). It involves the application of automation engineering in the provision of services in the culture and entertainment industry, aiming to facilitate the relationship between the content creation of the company in question and its audience. The objective of this work is to demonstrate the potential of automation not only in the industrial field but also in other sectors of the economy: in this case, by creating an application for a digital football community, one of the new ways of consuming Brazil's most popular sport. By adopting a human-centered approach through Design Thinking, the project sought to understand the needs and experiences of users, resulting in solutions more aligned with their real demands and also with contemporary problem-solving demands. The Scrum methodology, used in conjunction, enabled agile and value-driven deliveries for the project. The PWA application developed in this work used the Flutter framework, allowing the creation of native iOS and Android applications from a single source code. Flutter uses the Dart programming language, derived from Javascript, which is an object-oriented programming language.

Keywords: Automation. Scrum. Design Thinking. Flutter. Javascript. Canal Isto é Avaí. Football.

LISTA DE FIGURAS

Figura 1 – Header do canal no Youtube.	21
Figura 2 – Diagrama do Model View Controller.	26
Figura 3 – Os seis princípios do Scrum.	32
Figura 4 – Scrum x O Modelo Tradicional Cascata (Waterfall).	35
Figura 5 – Fluxo de um projeto Scrum.	38
Figura 6 – Como funciona a Sprint.	41
Figura 7 – Design Thinking.	43
Figura 8 – Levantamento na Desk Research.	48
Figura 9 – Mapa de empatia.	50
Figura 10 – Quadro de dores e necessidades.	51
Figura 11 – Brainstorming.	52
Figura 12 – Categorização.	52
Figura 13 – Matriz de Priorização.	53
Figura 14 – Crazy 8's.	54
Figura 15 – User Flow.	55
Figura 16 – Design das telas no Figma.	56
Figura 17 – Comando flutter doctor.	56
Figura 18 – Diagrama da arquitetura da aplicação.	57
Figura 19 – Features.	58
Figura 20 – Diagrama de casos de uso.	59
Figura 21 – Formulário de Feedback.	69
Figura 22 – Telas de início do aplicativo.	70
Figura 23 – Telas de Home, Lista de vídeos e Player de vídeos.	71
Figura 24 – Telas de Notícias e Parcerias.	72

LISTA DE QUADROS

Quadro 1 – Doze princípios de um processo ágil.	30
Quadro 2 – Casos de teste.	63

LISTA DE TABELAS

Tabela 1 – Caso de Teste 1.	63
Tabela 2 – Caso de Teste 2.	64
Tabela 3 – Caso de Teste 3.	64
Tabela 4 – Caso de Teste 4.	65
Tabela 5 – Caso de Teste 5.	65
Tabela 6 – Caso de Teste 6.	66
Tabela 7 – Caso de Teste 7.	66
Tabela 8 – Caso de Teste 8.	67
Tabela 9 – Caso de Teste 9.	68

LISTA DE ABREVIATURAS E SIGLAS

API	Application Programming Interface
CBF	Confederação Brasileira de Futebol
CSS	Cascading Style Sheets
DSDM	Dynamic Systems Development Method
ESPN	Entertainment and Sports Programming Network
EUA	Estados Unidos da América
EY	Ernst & Young
FDD	Feature Driven Development
FIFA	Fédération Internationale de Football Association
GE	General Eletric
HTML	HyperText Markup Language
HTTPS	Hypertext Transfer Protocol Secure
IEEE	Institute of Electrical and Electronics Engineers
iOS	iPhone Operating System
Lean	Lean Manufacturing
MIT	Massachusetts Institute of Technology
MVC	Model-View-Controller
NBA	National Basketball Association
OODA	Observar, Orientar, Decidir e Agir
PFC	Projeto de Fim de Curso
PIB	Produto Interno Bruto
PWA	Progressive Web App
SBOK	Scrum Body of Knowledge
SCRUM	Sprint, Cycle, Review, Update e Meeting
SDK	Software Development Kit
TV	televisão
VSCoDe	Visual Studio Code
WSL	World Surf League
XP	Extreme Programming

SUMÁRIO

1	INTRODUÇÃO	17
1.1	APRESENTAÇÃO DO TEMA	17
1.2	OBJETIVOS	17
1.2.1	Objetivo Geral	17
1.2.2	Objetivo Específico	18
1.3	JUSTIFICATIVA	18
1.4	ESTRUTURA DO DOCUMENTO	20
2	OS PROCESSOS DA EMPRESA	21
2.1	ISTO É AVAÍ	21
2.2	PRODUTOS, PROCESSOS E PROBLEMAS DA EMPRESA	22
3	REFERENCIAL TEÓRICO	23
3.1	APLICATIVOS MÓVEIS	23
3.1.1	Aplicações Nativas	23
3.1.2	Aplicações Híbridas	24
3.1.2.1	JavaScript	24
3.1.2.1.1	<i>Dart</i>	25
3.1.2.1.2	<i>Flutter</i>	25
3.2	ARQUITETURA DE SOFTWARE	25
3.2.1	Arquitetura MVC	26
3.3	FIREBASE	27
3.3.1	Autenticação de Usuário	27
4	METODOLOGIA	28
4.1	METODOLOGIA ÁGIL	28
4.1.1	Metodologia Scrum	31
4.1.1.1	Princípios do Scrum	32
4.1.1.2	Papéis do Scrum	36
4.1.1.3	Processos do Scrum	37
4.1.1.4	Eventos do Scrum	38
4.2	DESIGN THINKING	40
4.2.1	Etapas do Design Thinking	42
4.3	METODOLOGIA DO PROJETO	44
5	O PROJETO	47
5.1	ETAPA 1: PLANEJAMENTO	47
5.1.1	Imersão	47
5.1.1.1	Desk Research	47
5.1.2	Definição	49
5.2	ETAPA 2: DESIGN	51

5.2.1	Ideação	51
5.2.2	Prototipação	51
5.2.3	Teste (do protótipo, com usuários)	54
5.3	ETAPA 3: DESENVOLVIMENTO	54
5.3.1	Desenvolvimento da solução	55
5.3.1.1	Preparação do ambiente	55
5.3.1.2	Arquitetura da aplicação	57
5.3.1.3	Casos de Uso	57
5.3.1.4	Funcionalidades	57
5.3.1.5	Integrações	62
5.3.1.5.1	<i>Firestore Authentication</i>	62
5.3.1.5.2	<i>Application Programming Interface (API)</i>	62
5.4	ETAPA 4: TESTES	62
5.4.1	Caso de Teste 1: Usuário faz cadastro no aplicativo.	62
5.4.1.1	Pré-requisito	62
5.4.1.2	Descrição	63
5.4.1.3	Resultado	63
5.4.2	Caso de Teste 2: Usuário cadastra senha inválida.	63
5.4.2.1	Pré-requisito	63
5.4.2.2	Descrição	63
5.4.2.3	Resultado	63
5.4.3	Caso de Teste 3: Usuário faz login.	64
5.4.3.1	Pré-requisito	64
5.4.3.2	Descrição	64
5.4.3.3	Resultado	64
5.4.4	Caso de Teste 4: Usuário tenta fazer login com email nao cadastrado.	64
5.4.4.1	Pré-requisito	64
5.4.4.2	Descrição	64
5.4.4.3	Resultado	65
5.4.5	Caso de Teste 5: Usuário tenta fazer login com senha incorreta.	65
5.4.5.1	Pré-requisito	65
5.4.5.2	Descrição	65
5.4.5.3	Resultado	65
5.4.6	Caso de Teste 6: Usuário vira membro do canal.	65
5.4.6.1	Pré-requisito	65
5.4.6.2	Descrição	66
5.4.6.3	Resultado	66

5.4.7	Caso de Teste 7: Usuário assiste vídeo no youtube através do aplicativo.	66
5.4.7.1	Pré-requisito	66
5.4.7.2	Descrição	66
5.4.7.3	Resultado	66
5.4.8	Caso de Teste 8: Usuário acessa notícias através do aplicativo.	67
5.4.8.1	Pré-requisito	67
5.4.8.2	Descrição	67
5.4.8.3	Resultado	67
5.4.9	Caso de Teste 9: Usuário acessa a página de um patrocinador através do aplicativo.	67
5.4.9.1	Pré-requisito	67
5.4.9.2	Descrição	67
5.4.9.3	Resultado	67
5.5	ETAPA 5: LANÇAMENTO	68
5.5.1	Lançamento beta (demo)	68
5.5.2	Feedback dos usuários	68
5.6	ETAPA 6: REVISÃO	68
6	ANÁLISE DE RESULTADOS	70
6.1	TELAS DA APLICAÇÃO	70
6.2	AVALIAÇÃO FUNCIONAL	71
6.3	ANÁLISE DE RESULTADOS	72
7	CONCLUSÃO	74
	REFERÊNCIAS	75

1 INTRODUÇÃO

Este capítulo contém uma breve contextualização do projeto desenvolvido, apresentando o tema a ser trabalhado, os objetivos gerais e específicos do projeto, além da motivação e justificativa para se resolver o problema. Ao final, apresenta-se a organização do documento.

1.1 APRESENTAÇÃO DO TEMA

A engenharia de automação faz uso de sistemas de controle e de tecnologia da informação, com o objetivo de diminuir a dependência do trabalho humano na fabricação de equipamentos e na prestação de serviços. Seu propósito é reduzir a necessidade de envolvimento constante dos sentidos humanos na inspeção, e também a capacidade mental humana para controle.

A substituição do trabalho manual por tecnologia, muitas vezes, passa pelo desenvolvimento de um software. E para isso é importante identificar os objetivos, pontos de dor, e gerar soluções que abordam os desafios envolvidos em cada proposta.

Hoje, com a transformação digital, é cada vez mais comum perceber o uso desses softwares no dia a dia da população. E eles aparecem em todos os tipos de cenários e indústrias. Os avanços tecnológicos na área de dispositivos móveis têm transformado significativamente a maneira como as pessoas interagem com a tecnologia. Acompanhando esse movimento, popularizou-se também a figura de criadores de conteúdo nos meios digitais, e comunidades virtuais sobre diferentes assuntos.

Apresenta-se ao longo deste trabalho o desenvolvimento prático de um projeto de software para o Canal Isto é Avaí (empresa-caso). Trata-se da aplicação de engenharia de automação na prestação de serviços do ramo de cultura e entretenimento, a fim de facilitar a relação entre o criador de conteúdo e seu público. A expectativa é demonstrar o potencial da automação não apenas no campo industrial, mas também em outros setores da economia: nesse caso, criando um aplicativo para uma comunidade digital de futebol: uma das novas formas de consumir o esporte mais popular do Brasil.

1.2 OBJETIVOS

Nas seções abaixo estão descritos o objetivo geral e os objetivos específicos deste Projeto de Fim de Curso.

1.2.1 Objetivo Geral

O objetivo deste projeto de fim de curso é a criação de um aplicativo PWA, para facilitar a relação entre o criador de conteúdo e seu público, aplicado no contexto do Canal Isto é Avaí.

1.2.2 Objetivo Específico

A partir do objetivo geral, se desdobram os seguintes objetivos específicos deste Projeto de Fim de Curso:

- Levantar metodologias para utilizar no projeto;
- Identificar as necessidades da empresa-caso e seu público;
- Idear soluções que resolvam as necessidades levantadas;
- Elaborar um plano de desenvolvimento para a aplicação PWA proposta;
- Desenvolver a aplicação PWA para o Canal Isto é Avaí;
- Testar a aplicação PWA.

1.3 JUSTIFICATIVA

Segundo estudo realizado pela Confederação Brasileira de Futebol (CBF) em parceria com a consultoria Ernst & Young (EY), o futebol movimentou um total de R\$ 52,9 bilhões na economia do País em 2018, o que representou 0,72% do total do Produto Interno Bruto (PIB) brasileiro na época. Este impacto na economia acontece não apenas pelas contribuições diretas da CBF, Federações Estaduais e clubes, mas também indiretamente a partir da relação entre estes e a mídia, patrocinadores, torcedores, indústrias logísticas e governos. 37,8 bilhões de reais da contribuição da indústria do futebol no PIB brasileiro em 2018 foi através desses efeitos indiretos, mostrando sua importância para a economia brasileira.

Intitulado "O Impacto do Futebol Brasileiro"(CBF, 2019a), a pesquisa mostra a relevância do futebol como negócio e não só como lazer. Com o pensamento de relacionar o futebol com a economia, não se levou em consideração o esporte propriamente dito, mas sim sua execução como uma indústria e, conseqüentemente, identificar onde esse setor se encontra no cenário econômico. "É um setor produtivo que afeta diretamente nossa economia"(FELDMAN, Walter). Segundo o ex-assessor especial do Ministério da Economia, Caio Megale: "A economia do esporte é muito importante, o futebol, principalmente, por ser o mais popular do país".

A evolução do segmento também é expressiva. Em 10 anos, a receita dos 27 principais clubes brasileiros aumentou em 96%, já descontada a inflação. Esse crescimento foi maior do que qualquer outro segmento do mercado (dados do estudo (CBF, 2019b)). E, segundo o mesmo estudo, ainda existe um grande potencial de crescimento a ser explorado.

Adicionalmente, a tecnologia também funciona como um catalisador para o crescimento da indústria do futebol no Brasil. O hábito de assistir os jogos pela televisão

está mudando - impulsionado pelas novas plataformas de transmissão e investimentos recorrentes em banda larga. Novos atores estão concorrendo pelos direitos de transmissão expandindo as possibilidades para os Clubes. Novas plataformas permitem uma maior interação com o fã, gerando inúmeras possibilidades de monetização. (CBF, 2019b).

Somado a essa antecipação que o estudo trouxe, o mundo todo vivenciou a pandemia da Covid-19 e a o crescimento exponencial da digitalização que se potencializou em decorrência dela. Todos os setores sentiram seus efeitos, e a indústria do futebol não ficou de fora. A pandemia e a tecnologia aceleraram as mudanças na forma de consumir futebol.

Exemplo dessas mudanças é a recente escolha de um clube de futebol brasileiro e até mesmo da Fédération Internationale de Football Association (FIFA) para a transmissão de seus jogos. Entre tantas televisões relevantes, como Globo, Entertainment and Sports Programming Network (ESPN) ou SporTV, a FIFA decidiu inovar e escolheu a CazéTV para transmitir os jogos da Copa do mundo Feminina 2023. O Club Athletico Paranaense, desde 2021, também deixou os direitos de transmissão de seus jogos mandantes exclusivamente pelo canal. A CazéTV é um canal no YouTube, e retransmitido na *Twitch*, especializado na transmissão de eventos esportivos. Seus 9 milhões de inscritos no Youtube mostram a projeção alcançada pelas suas transmissões descontraídas, leves, de forma simples e autêntica - que também garantiu a ele a prioridade da FIFA para transmissão do campeonato e experimentação de novos recursos digitais.

Mas não apenas de grandes canais no YouTube vive esse novo mercado alternativo de transmissão e produção de conteúdo sobre futebol. São inúmeras as comunidades digitais no futebol, e elas tem se mostrado uma maneira poderosa de conectar fãs de todo o mundo, proporcionando um espaço para compartilhar informações, expressar opiniões, celebrar vitórias e superar derrotas. Essas comunidades fortalecem o sentimento de pertencimento e a paixão pelo esporte, tornando o futebol uma experiência social e globalmente conectada. Hoje no Brasil, a mídia identificada de clubes vem crescendo cada vez mais e tomando espaço da mídia tradicional, principalmente dentro de plataformas sociais.

O presente trabalho busca explorar uma oportunidade no universo das comunidades digitais de futebol, e solucionar - através da Engenharia de Controle e Automação - uma dor dos criadores de conteúdo nessa área. Desenvolvendo um projeto com base no Isto é Avaí, um canal de comunicação sobre um time de futebol de Florianópolis, propõe-se a criação de um aplicativo para ser o ponto de encontro dessa comunidade, promovendo mais valor para torcedores e patrocinadores.

1.4 ESTRUTURA DO DOCUMENTO

O capítulo 1 deste trabalho introduziu o tema a ser trabalhado, objetivos, motivação e justificativa do projeto.

O capítulo 2 abordará a descrição do Canal Isto é Avaí, empresa de realização deste Projeto de Fim de Curso (PFC), assim como seus processos.

O capítulo 3 será realizada uma análise da base teórica, tratando conceitos fundamentais e tecnologias relacionadas ao desenvolvimento de aplicações PWA.

O capítulo 4 são apresentados os conceitos de metodologia que embasaram este trabalho, assim como a metodologia mista entre Design Thinking e Sprint, Cycle, Review, Update e Meeting (SCRUM) que guiou o projeto.

O capítulo 5 é apresentado o desenvolvimento do aplicativo, através das tecnologias detalhadas e apresentadas no Capítulo 3, e da metodologia explicada no Capítulo 4. Neste capítulo também é descrito como cada funcionalidade opera no dispositivo, assim como resultados visuais da aplicação em operação.

O capítulo 6 apresentará a análise funcional do aplicativo e também dos resultados alcançados.

Por fim, o capítulo 7 apresenta as conclusões obtidas a partir do desenvolvimento da proposta, bem como sugestões de trabalhos futuros para uma melhor utilização da aplicação desenvolvida.

2 OS PROCESSOS DA EMPRESA

Este capítulo apresenta a empresa-caso deste PFC, seus produtos, processos e problemas, afim de justificar a importância de desenvolvimento do presente projeto para a mesma.

2.1 ISTO É AVAÍ

O Isto é Avaí, objeto de estudo para desenvolvimento da aplicação PWA deste trabalho, é um canal no Youtube para a torcida Avaiana, mas também uma comunidade para a torcida do time da capital catarinense. Hoje contam com três principais canais de interação: YouTube, WhatsApp e Instagram.

Em cerca de 2 anos a comunidade do Isto é Avaí, assim como o negócio, vem crescendo rapidamente. O programa começou pequeno, entre amigos que se identificavam com o time centenário. Mas encontraram oportunidade neste nicho e hoje, próximos de completar sua 200ª transmissão ao vivo, o canal mostra grande relevância no universo do futebol da capital.

O canal no Youtube (figura 1) tem hoje mais de 5 mil inscritos e 540 mil visualizações nos vídeos postados na plataforma, além de 150 assinantes do canal¹. Nos grupos do WhatsApp, contam com a participação ativa de mais de 800 fãs do clube, distribuídos entre os três grupos do canal no aplicativo, além de um grupo extra exclusivo para os membros pagantes. No Instagram, o perfil tem mais de 4.700 seguidores engajados, absorvendo os conteúdos postados diariamente².

Figura 1 – Header do canal no Youtube.



Fonte: Do autor.

¹ <http://youtube.com/IstoeAvai>

² <http://instagram.com/istoeavaipodcast>

Ao longo dessa trajetória curta, mas intensa, o canal já teve episódios com participação de grandes celebridades da história do clube. Desde de Marquinhos Santos, o ídolo avaiano; Zenon, ex-jogador da década de 70; Paulo Silas, ex-jogador, ex técnico e hoje comentarista esportivo na ESPN Brasil; Jean Cleber, Raniele, Bruno Silva, Arthur Chaves entre outros jogadores do elenco. O canal também foi palco de entrevistas no período eleitoral do clube, em 2021, recebendo as chapas de Carlos Bonatelli e Gilson Kremer, Julio Heerdt e Bruno Comicholli, Francisco Battistotti e Amaro Lúcio. Posteriormente visitaram novamente o programa o atual presidente Julio Heerdt, seu ex-vice presidente Bruno Comicholli, além do coordenador de comunicação Rafael Xavier, o atual diretor de futebol Eduardo Freeland e também os ex-diretores de futebol Jorge Macedo e André Martins.

Além dos membros e espectadores, a história vem sendo construída ao lado de apoiadores e patrocinadores, e contou também com a participação do histórico e lendário comentarista esportivo Miguel Livramento, que fez parte do canal durante os últimos 6 meses de sua vida.

2.2 PRODUTOS, PROCESSOS E PROBLEMAS DA EMPRESA

O principal produto oferecido pelo Canal Isto é Avaí é a sua produção de conteúdo. Desde os episódios semanais ao vivo no Youtube, transmissões de jogos fora da casa, vídeos com a torcida, entrevistas de figuras importantes do clube e demais conteúdos informativos.

Hoje em dia, para um canal ser relevante ao seu público, é necessário estar presente em várias redes sociais. Cada uma com seu algoritmo, que muda a todo momento. Isso exige uma constante adaptação nos processos da empresa, revisão dos conteúdos a serem criados e até mesmo mudanças no formato dos conteúdos (vídeo, imagem estática, pessoas aparecendo, duração, hashtags, etc). Nas redes sociais, é preciso estar em alta com o algoritmo para que mesmo seus membros e pessoas que já te seguem, continuem recebendo seu conteúdo.

A monetização do canal, que o viabiliza como negócio e permite que continue crescendo, também passa pelo dilema das redes sociais. Além de toda a disputa com o extenso catálogo de opções, as taxas são altas. Hoje, 30% do valor pago mensalmente pelos assinantes do canal, fica com a plataforma do Youtube.

Todos os processos de criação de conteúdo do canal podem acabar rendendo pouco resultado, pois dependem do algoritmo das redes sociais e suas retenções financeiras. Dessa forma, o presente projeto visa desenvolver uma aplicação PWA que permita ao canal ter mais autonomia no contato e interação com seus membros e fãs.

3 REFERENCIAL TEÓRICO

Nesta fase, é realizada uma análise da base teórica, procurando por informações em fontes confiáveis que abordem os temas e as tecnologias relacionadas ao assunto abordado neste trabalho.

O objetivo aqui é apresentar conceitos relacionados às tecnologias no desenvolvimento de aplicações para dispositivos móveis, explorando as principais tecnologias usadas em aplicações nativas e híbridas, entre outros elementos-chave no desenvolvimento desses aplicativos. Afinal, com o rápido crescimento do uso de smartphones e tablets, tornou-se imperativo desenvolver aplicativos móveis que atendam eficientemente e de maneira intuitiva às necessidades dos usuários.

3.1 APLICATIVOS MÓVEIS

Também chamados popularmente de apps, os aplicativos móveis tornaram-se uma parte integral da vida cotidiana com o avanço da tecnologia de dispositivos móveis. Eles são uma resposta direta à demanda por experiências de usuário mais personalizadas e convenientes. E, de acordo com Kubben, são também uma fonte importante de dados.

As aplicações móveis são programas desenvolvidos para serem executados em dispositivos móveis, como smartphones e tablets. Elas têm como objetivo oferecer uma experiência interativa e otimizada para o usuário, distinguindo-se das aplicações web pela sua capacidade de serem baixadas e instaladas diretamente no dispositivo.

3.1.1 Aplicações Nativas

Dois grandes sistemas de operação são importantes para as aplicações móveis: iOS (Apple) e Android (Google), com uma quota de mercado total de 99% (iOS 54% e Android 45% em maio de 2018, medidos nos Estados Unidos da América (EUA)). Estes dois sistemas de operação não são compatíveis entre si, o que significa que a programação para ambos requer uma abordagem diferente (DEKKER; DUMONTIER; KUBBEN, 2020).

As aplicações nativas são projetadas e desenvolvidas para uma plataforma específica, como iOS ou Android, utilizando as linguagens de programação e ambientes de desenvolvimento recomendados por cada uma das plataformas.

Kubben menciona que essas aplicações possuem acesso completo aos recursos do dispositivo, incluindo câmera, GPS e sensores. Esse acesso proporciona um desempenho otimizado, e uma experiência de usuário mais fluida com a criação de experiências personalizadas e ricas ao aproveitar totalmente os recursos específicos de cada dispositivo. Apesar da série de benefícios apresentados, criar aplicações nati-

vas demanda conhecimentos específicos para cada plataforma, o que resulta em um aumento no tempo e custo de desenvolvimento dos aplicativos.

3.1.2 Aplicações Híbridas

As aplicações híbridas buscam superar as limitações das aplicações nativas, permitindo o desenvolvimento de uma única base de código que pode ser utilizada em múltiplas plataformas. Isso porque elas combinam tecnologias web com um invólucro nativo, proporcionando eficiência no desenvolvimento.

A possibilidade de redução no seu tempo de desenvolvimento e custo é bastante atrativa para empresas e profissionais desenvolvedores. No entanto, as aplicações híbridas podem apresentar desempenho ligeiramente inferior às aplicações nativas e ter acesso limitado aos recursos do dispositivo, dependendo de implementações ou recursos fornecidos por empresas terceirizadas ou pela comunidade envolvida na tecnologia escolhida.

Diversos *frameworks* e ferramentas estão disponíveis para o desenvolvimento de aplicações móveis. O desenvolvimento nativo muitas vezes utiliza Swift, Objective-C, Java ou Kotlin, enquanto o desenvolvimento híbrido é facilitado por frameworks como React Native, Flutter e Xamarin.

3.1.2.1 JavaScript

JavaScript é uma linguagem de programação amplamente utilizada no desenvolvimento web. Originalmente desenvolvida para criar páginas web interativas, foi adotada por grandes navegadores do mercado, como o Google Chrome, Firefox, Safari e Opera (CROCKFORD, 2008). JavaScript evoluiu ao longo do tempo e agora é uma linguagem de programação versátil e poderosa que pode ser executada em diversos ambientes, não se limitando apenas a navegadores da web.

A linguagem é interpretada e orientada a objetos, o que significa que as ações são realizadas por meio de objetos e funções, tornando-a flexível e dinâmica. O JavaScript é conhecido por seu suporte à programação assíncrona, o que permite que operações sejam executadas sem interromper o fluxo principal do programa.

Uma das características mais marcantes do JavaScript é sua integração direta com HyperText Markup Language (HTML) e Cascading Style Sheets (CSS), permitindo a manipulação dinâmica e interativa do conteúdo das páginas. Além disso, com o advento de ambientes de execução do lado do servidor, como o Node.js, o JavaScript passou a também ser usado no desenvolvimento de aplicativos fora do contexto do navegador, expandindo ainda mais suas capacidades.

A popularidade do JavaScript cresceu significativamente nos últimos anos, impulsionada por frameworks como Flutter, React, Angular e Vue.js, que simplificam o desenvolvimento de interfaces de usuário modernas. Dessa forma, o JavaScript de-

semprenha um papel crucial na criação de experiências interativas e responsivas na web e em outras aplicações.

3.1.2.1.1 Dart

Dart é uma linguagem de programação relativamente recente e que já se desenvolveu muito desde seu primeiro lançamento. Desenvolvida pela Google e lançada em 2011, foi projetada para criar aplicações web, aplicações móveis e de desktop. Seu objetivo é fornecer uma alternativa eficiente para o JavaScript, especialmente para o desenvolvimento de aplicações complexas e de grande escala.

A linguagem Dart permite que o código criado rode tanto no lado do cliente quanto no servidor. Em seu site oficial, a linguagem é definida como otimizada para clientes, desenvolvida para criação de aplicações rápidas em qualquer plataforma.

Dart é uma linguagem orientada a objetos, o que significa que ela organiza o código em classes e objetos, facilitando a modelagem de estruturas complexas. É uma escolha poderosa para desenvolvedores que buscam uma linguagem moderna e eficiente para construir aplicações web e móveis, especialmente quando combinada com o ecossistema do Flutter. Parafraseando (LELER, 2017), a linguagem Dart é um grande motivo pela qual os desenvolvedores adoram Flutter.

3.1.2.1.2 Flutter

O Flutter é um popular framework open-source, também criado pela Google, de desenvolvimento de aplicativos móveis, que usa a linguagem Dart para criar interfaces de usuário bonitas e fluidas de maneira eficiente e rápida para dispositivos iOS, Android, Windows, Mac, Linux, Fuchsia e WEB (FLUTTER, 2023).

Uma das características mais destacadas do Flutter é o "Hot Reload". Ele permite que os desenvolvedores vejam imediatamente as alterações feitas no código refletidas no aplicativo em execução, acelerando significativamente o processo de desenvolvimento.

Com o Flutter, é possível criar aplicativos para iOS, Android e web a partir de uma única base de código (uma aplicação híbrida). Como levantado anteriormente, isso reduz a necessidade de manter códigos separados para plataformas diferentes.

O Flutter ganhou popularidade devido à sua abordagem inovadora e eficiente para o desenvolvimento de aplicativos móveis, tornando-se uma escolha atraente para desenvolvedores e empresas.

3.2 ARQUITETURA DE SOFTWARE

Segundo a definição dada pela Institute of Electrical and Electronics Engineers (IEEE), pode-se dizer que: "Arquitetura de software pode ser definida como a orga-

nização fundamental de um sistema, seus componentes, as relações entre eles e o ambiente que guia seu design e evolução.”.(IEEE, s.d.)

A arquitetura de software refere-se à estrutura fundamental do sistema, incluindo componentes, conexões, padrões e diretrizes que guiam o desenvolvimento e a organização do código. É uma visão de alto nível que define como os diferentes elementos de um software interagem e se comunicam.

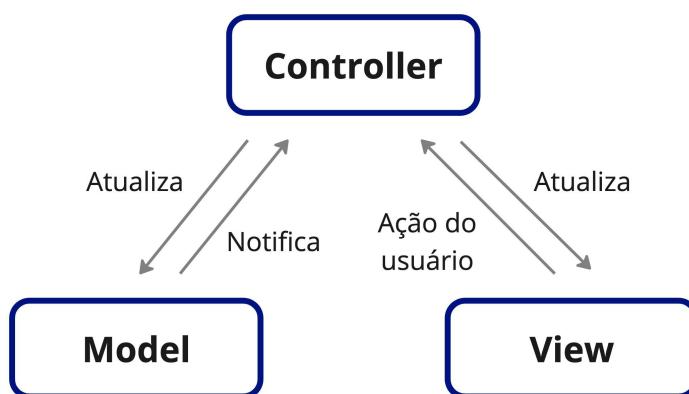
Ela compreende componentes como módulos, camadas, serviços, bancos de dados, entre outros, e as relações entre eles. São utilizados padrões arquiteturais, como Model-View-Controller (MVC), microservices, monolítico, entre outros, para orientar a concepção e o design.

3.2.1 Arquitetura MVC

A arquitetura de código MVC é um padrão de arquitetura muito utilizado na engenharia de software, por oferecer uma estrutura organizacional que separa as responsabilidades entre os componentes, facilitando a manutenção e a evolução do código.

O MVC baseia-se na divisão da aplicação em três camadas, sendo uma camada responsável pela apresentação dos dados (onde o usuário interage com o sistema), uma para modelagem dos dados e outra para a manipulação desses dados (IVANOVICH *et al.*, 2019). As três camadas estão listadas e ilustradas (figura 2) a seguir:

Figura 2 – Diagrama do Model View Controller.



miro

Fonte: Adaptado de (KEARNEY, 2019)

- **Model:** Representa a camada de dados e lógica de negócios da aplicação. Ele gerencia o acesso aos dados, processa a lógica do negócio e notifica a View sobre as alterações nos dados.

- **View:** É responsável pela apresentação dos dados ao usuário. Ela exibe as informações e recebe as interações do usuário, repassando essas interações para o Controller.
- **Controller:** Age como intermediário entre o Model e a View. Ele recebe as entradas do usuário da View, processa essas entradas, atualiza o Model conforme necessário e atualiza a View para refletir as mudanças nos dados.

O uso da arquitetura MVC contribui para o desenvolvimento de aplicações mais organizadas, escaláveis e de fácil manutenção, sendo aplicável em diversos tipos de projetos, desde aplicações web até sistemas desktop.

3.3 FIREBASE

O Firebase é uma plataforma de desenvolvimento de aplicativos móveis e web, popular e gratuita oferecida pelo Google Cloud. Ele fornece uma variedade de serviços e ferramentas prontas para uso, facilitando o desenvolvimento, o crescimento e a manutenção dos aplicativos nos quais é integrado, simplificando várias tarefas complexas e fornecendo uma estrutura completa de desenvolvimento de aplicativos baseada em nuvem. O Firebase fornece uma documentação detalhada e SDKs multiplataforma de desenvolvimento de apps que ajudam a criar e entregar apps para iOS, Android, Web, Flutter, Unity e C++ (FIREBASE, 2023).

O Firebase é amplamente utilizado por desenvolvedores devido à sua facilidade de uso, integração eficiente com outras ferramentas do Google Cloud e à gama abrangente de serviços oferecidos. Ele é especialmente valioso para startups, pequenas e médias empresas que desejam desenvolver e escalar aplicativos de forma rápida e eficaz.

3.3.1 Autenticação de Usuário

O serviço de autenticação de usuário do Firebase, o Firebase Auth, é uma parte integrante do conjunto de serviços oferecidos pela plataforma. Ele fornece uma solução abrangente para gerenciar a autenticação e a identidade dos usuários em aplicativos móveis e da web.

De acordo com (FORONES, 2021), a implementação de uma boa autenticação previne certos tipos de ataques à aplicação. O Firebase Auth implementa práticas de segurança robustas, garantindo a proteção das credenciais dos usuários e a transmissão segura de dados usando Hypertext Transfer Protocol Secure (HTTPS).

4 METODOLOGIA

A adoção de uma metodologia de desenvolvimento de sistema orienta os integrantes dos projetos no planejamento das tarefas, na execução e nas ações necessárias para desenvolver uma solução de qualidade (MALL, 2014). De acordo com (DENNIS; WIXOM; ROTH, 2014), uma metodologia de desenvolvimento tem a função de formalizar a ordem de desenvolver um software, organizar a fase de seu ciclo de vida e ter o equilíbrio entre os processos e os dados. Atualmente há vários tipos de metodologia de desenvolvimento e cada empresa pode construir a sua conforme suas necessidades. Porém, todas essas propostas de metodologias têm o mesmo objetivo, formalizar a fase de desenvolvimento e organizar os dados (informações).

O problema é que as práticas profissionais que foram desenvolvidas e aperfeiçoadas ao longo de muitos anos e que, em conjunto, representam o nosso conhecimento e experiência compartilhados na indústria, estão demasiadas vezes aprisionadas em cadeias de métodos proprietários. A única opção que as organizações e equipes de desenvolvimento vêem como tendo é adotar este ou aquele método por atacado e rejeitar todos os outros - enquanto que, na realidade, o que é necessário é que as organizações e equipes sejam livres de selecionar as práticas profissionais de que necessitam, onde quer que estas possam ser definidas, e utilizá-las em quaisquer permutações e combinações que sejam apropriadas para satisfazer o conjunto exato de circunstâncias e desafios que enfrentam (JACOBSON; STIMSON, 2018).

Para o desenvolvimento deste projeto, foi utilizada uma metodologia híbrida. A proposta tem como base elementos extraídos do *framework Scrum* em conjunto com etapas da metodologia de Design Thinking.

4.1 METODOLOGIA ÁGIL

As metodologias tradicionais fizeram e fazem parte do cotidiano de muitos profissionais envolvidos na construção de projetos de software. Tais modelos apresentam características com foco em processos rígidos, orientadas aos documentos e com planos bastante extensos. Conforme (SINGHTO; DENWATTANA, 2016) as metodologias tradicionais seguem um modelo sequencial de etapas a serem vencidas e são recomendadas para serem aplicadas em projetos que demandam maior controle de qualidade. O modelo tradicional mais conhecido é o Waterfall ou cascata, o qual ainda é utilizado em muitos projetos.

Os métodos ágeis representam uma abordagem oposta ao modelo Waterfall, que requer uma extensa fase de planificação para basear a execução do projeto. Isso

dificulta a implantação de mudanças nas fases posteriores ao planejamento. A publicação do Manifesto Ágil em 2001 marcou o início da transformação não apenas nos métodos de trabalho em cascata entre desenvolvedores, mas também em diversos setores de uma organização.

Este modelo de metodologia foi proposto para superar as deficiências apresentadas pelos modelos tradicionais, como por exemplo, inverter o foco em agregar valor para o cliente com as entregas mais rápidas, diminuindo processos e planos (MALL, 2014). Serve para adaptar o modo de trabalho às condições de um projeto, garantindo maior flexibilidade e rapidez à sua execução. Eles são fruto da Cultura Ágil, conjunto de ideias e práticas que buscam aprimorar a agilidade, flexibilidade e eficiência nos processos de trabalho, tornando-os mais adaptáveis às mudanças do mercado e atendendo assim às demandas do cliente e da empresa.

As metodologias ágeis chegaram como uma resposta à volatilidade do cenário digital e à tendência crescente da inovação. As metodologias mais tradicionais, que prezavam por documentação e pré-projetos bem estabelecidos, não davam conta de lidar com esse cenário.

Os primeiros métodos ágeis começaram a ser utilizados na década de 1990, como alternativa à abordagem de desenvolvimento de sistemas dirigido a planos. Eles tomaram força a partir de 2001, com a criação do Manifesto Ágil, quando 17 especialistas em métodos considerados como leves se reuniram no estado norte-americano de Utah. Apesar de utilizarem abordagens e métodos diferentes, eles constataram que suas ideologias eram bastante semelhantes e decidiram unificá-las originando, assim, o Manifesto Ágil (SUTHERLAND, Jeff; SUTHERLAND, J.J, 2019).

Esses tais de Métodos Leves eram os que hoje conhecemos como SCRUM, Dynamic Systems Development Method (DSDM), Adaptive Software Development, Crystal, Feature-Driven Development, Pragmatic Programming entre outros.

(DINGSØYR *et al.*, 2018) afirmaram que, após o “Manifesto Ágil”, as metodologias ágeis revolucionaram a prática de desenvolvimento de software com foco na tolerância à mudança, à entrega evolutiva e ao envolvimento ativo do usuário final. Essa afirmação representa uma real transformação na maneira de trabalhar de muitas fábricas de software, utilizando prática ágeis em seus modelos.

Como o Manifesto Ágil apresenta grande relevância para as mudanças na forma de trabalhar com projetos de software, sempre é importante apresentar seus valores e princípios, pois eles sedimentam abordagens de qualquer pesquisa.

A importância do Manifesto Ágil nas transformações da abordagem em projetos de software é evidente. Por isso, é sempre relevante apresentar seus valores e princípios, pois estes constituem a base para qualquer pesquisa relacionada a esse contexto, proporcionando uma sólida fundação para as abordagens adotadas. Os conceitos-chave do Manifesto Ágil são (BECK *et al.*, 2001):

- (a) indivíduos e interações mais que processos e ferramentas;
- (b) software em funcionamento mais que documentação abrangente;
- (c) colaboração com o cliente mais que negociação de contratos;
- (d) responder às mudanças mais que seguir um plano.

Além de seus valores-chave, o manifesto também enuncia os doze princípios de um processo ágil, conforme apresentado no Quadro 1 (BECK *et al.*, 2001).

Quadro 1 – Doze princípios de um processo ágil.

ID	Princípio.
P1	A prioridade é satisfazer ao cliente por meio de entregas contínuas e frequentes.
P2	Receber bem as mudanças de requisitos, mesmo em uma fase avançada do projeto.
P3	Entregas com frequência, sempre na menor escala de tempo.
P4	As equipes de negócio e de desenvolvimento devem trabalhar juntas diariamente.
P5	Manter uma equipe motivada fornecendo ambiente, apoio e confiança necessários.
P6	A maneira mais eficiente da informação circular por meio de uma conversa face a face.
P7	Ter o sistema funcionando é a melhor medida de progresso.
P8	Processos ágeis promovem o desenvolvimento sustentável.
P9	Atenção contínua à excelência técnica e a um bom projeto aumentam a agilidade.
P10	Simplicidade é essencial.
P11	As melhores arquiteturas, requisitos e projetos provêm de equipes organizadas.
P12	Em intervalos regulares, a equipe deve refletir sobre como se tornar mais eficaz.

Em resumo, as principais características dos métodos ágeis são:

- O foco está na entrega de um produto funcional, de qualidade e que realmente atenda os desejos e necessidades do cliente;
- O cliente é convidado a participar com frequência, porque tem um papel decisivo na definição de novos requisitos;
- Como o foco está na entrega, a execução do projeto é prioridade. Por isso, a maior parte da documentação é gerada a partir das próprias ferramentas de produção;

- As pessoas são fundamentais no desenvolvimento de todo projeto, por isso é de suma importância haver uma boa comunicação entre as partes e equipes do projeto;

Dentre as metodologias ágeis mais utilizadas, destacam-se: Extreme Programming (XP), SCRUM, Crystal, Feature Driven Development (FDD), DSDM, Open-Source Software Development (OSSD). Todos esses métodos usam princípio semelhantes, mas são diferentes em suas práticas e formas de condução do processo de desenvolvimento. Dentre eles, ressaltam-se o XP e o SCRUM (PAULA FILHO, 2009).

Este projeto utilizou do método Scrum, tornando imperativo fornecer uma descrição mais detalhada de suas características. Portanto, na seção seguinte, será realizada uma apresentação mais aprofundada dessa metodologia.

4.1.1 Metodologia Scrum

Dentre as diferentes metodologias ágeis, destaca-se o Scrum: é a mais utilizada nos projetos de software, de acordo com a pesquisa realizada por (FUIOR, 2019). A metodologia é a de maior destaque entre os modelos ágeis, o que evidencia uma grande comunidade de usuários (HRON; OBWEGESER, 2022).

O nome Scrum deriva de uma jogada de rugby, com o objetivo de reposição da bola no jogo. É o momento onde os jogadores dos dois times se juntam com a cabeça abaixada e se empurram com o objetivo de ganhar a posse de bola. Trata-se de um momento em que a equipe está toda unida com um único propósito, e em uma formação específica, na qual a participação de todos é essencial. A falta de comprometimento de um membro pode fazer a formação cair, então a união e o foco no objetivo (mover a bola em direção ao gol) é primordial (CRUZ, 2015). Takeuchi e Nonaka propõem que o desenvolvimento do produto não deve ser como uma sequência de corrida de revezamento, mas sim semelhante ao jogo de rugby em que o time trabalha em conjunto, passando a bola para frente e para trás movendo-se através do campo como uma unidade (SBOK™, 2015).

A metodologia Scrum teve início em meados de 1990, idealizada por Ken Schwaber e Jeff Sutherland, com base nos métodos Lean Manufacturing (Lean) (ciclo do processo de produção da Toyota) e Observar, Orientar, Decidir e Agir (OODA) (ciclo da aviação de combate dos USA). Para seus idealizadores, o Scrum é um método leve, simples de entender, porém, de difícil domínio (SUTHERLAND, Jeff; SUTHERLAND, J.J, 2019).

Sua definição trata de um *framework* para desenvolver, entregar e manter produtos complexos. Foi criado inicialmente para gerenciar projetos, com ênfase no desenvolvimento de softwares, mas por ser uma ferramenta extremamente adaptativa, hoje também é utilizado em várias outras áreas das organizações, como vendas, tecnologia, marketing, etc.

Trata-se de uma ferramenta onde as pessoas podem abordar problemas adaptativos complexos, enquanto entregam de forma produtiva e criativa produtos do mais alto valor possível. O atrativo do Scrum reside na capacidade de viabilizar projetos cujas necessidades são difíceis de quantificar desde o início.

Metodologias ágeis, como o Scrum, priorizam muito mais a entrega de valor. Isso dá certo na prática porque elas são pensadas para lidar com imprevistos e mudanças no planejamento ao longo do percurso.

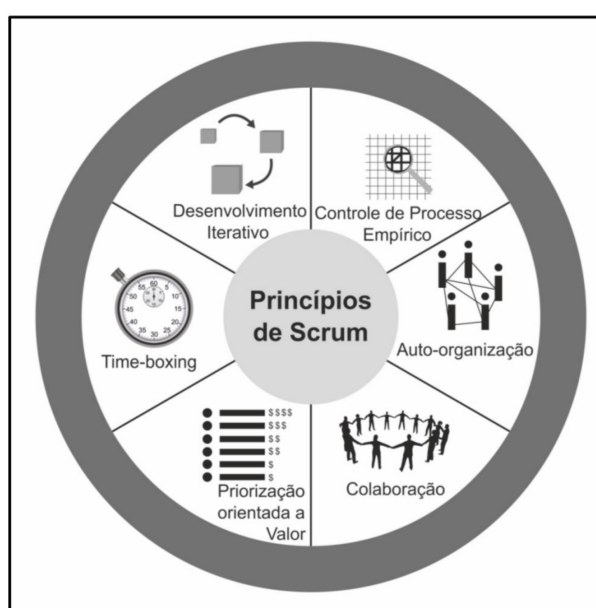
Essa abordagem visa alcançar resultados mais eficazes em prazos reduzidos. Mas é primordial esclarecer que o Scrum, assim como as demais metodologias ágeis, não necessariamente prezam por entregas relâmpago ou estritamente dentro de prazos. Nesse contexto, ser ágil significa ter a capacidade de lidar com mudanças e adaptar o projeto a elas, de forma organizada e segura para todos os envolvidos.

Para entregar esse resultado, o Scrum funciona através de um conjunto de cerimônias, processos e responsabilidades padrão da ferramenta. A seguir, seguem algumas dessas definições e características:

4.1.1.1 Princípios do Scrum

De acordo com o Guia Scrum Body of Knowledge (SBOK)TM, a metodologia Scrum tem seis princípios (figura 3), que são as diretrizes fundamentais para a aplicação do *framework* e devem ser usados em todos os projetos que utilizem da metodologia. São eles:

Figura 3 – Os seis princípios do Scrum.



Fonte: (SBOKTM, 2015).

1. Controle de Processos Empíricos

Segundo (SCHWABER K.; SUTHERLAND, 2017), fundadores da metodologia, “o Scrum é fundamentado nas teorias empíricas de controle de processo, ou empirismo”. Ou seja, todas as decisões de um projeto Scrum devem ser tomadas com base em observação e em experimentos, ao invés no planejamento inicial detalhado. O Controle de Processos Empíricos se baseia em três idéias principais:

- (a) **Transparência:** Permite que todos os ângulos, de qualquer processo do projeto com Scrum, seja observado por qualquer pessoa. Dessa forma o fluxo de informações se torna fácil e transparente, criando uma cultura de trabalho aberta.
- (b) software em funcionamento mais que documentação abrangente;
- (c) **Inspeção:** As pessoas envolvidas no projeto devem inspecionar com frequência o progresso e as entregas produzidas para detectar possíveis variações. É representada pelo uso de um Scrumboard para acompanhar o processo de todo o time. Além disso, também conta com a coleta constante de feedbacks dos clientes e dos stakeholders e com a inspeção e avaliação das entregas.
- (d) **Adaptação:** A adaptação acontece quando o Time Central do Scrum e os Stakeholders aprendem através da transparência e da inspeção e, em seguida, adaptam o processo ao fazerem melhorias no trabalho que está sendo realizado.

2. Auto-organização

No Scrum, os membros da equipe são incentivados a demonstrar proatividade e assumir responsabilidades crescentes, destacando a auto-organização como sua característica principal. Essa abordagem é favorecida quando a distinção entre líder e liderados é minimizada, e todos colaboram de forma equitativa em prol do projeto.

Além disso, a auto-organização cria um ambiente propício à inovação e à criatividade, transformando a responsabilidade em um esforço coletivo e elevando a satisfação da equipe.

3. Colaboração

Nenhum projeto pode sair do papel sem colaboração. Por essa razão, a equipe Scrum deve colaborar de forma estreita com os stakeholders para criar e validar as entregas do projeto. Essa abordagem assegura que todos os envolvidos e

interessados no progresso do projeto estejam alinhados, garantindo a realização da visão final.

As dimensões principais do trabalho colaborativo são:

- **Consciência:** Os colaboradores precisam estar alinhados e cada membro da equipe deve saber o que os outros membros estão fazendo.
- **Articulação:** A atribuição de tarefas deve ser organizada entre os membros da equipe. Dessa forma, o produto final é dividido em unidades, designado a diferentes membros e, quando concluído, é reintegrado de maneira coordenada.
- **Apropriação:** Refere-se à adaptação da tecnologia às necessidades diárias. Ter suporte tecnológico para auxiliar nas operações de trabalho é essencial, especialmente quando se trata de colaboração.

4. **Priorização Baseada em Valor**

O objetivo central do Scrum é proporcionar o máximo valor de negócio em um período mínimo de tempo. Para atingir esse propósito, a priorização baseada em valor é um recurso essencial. Itens com potencial para gerar maior valor recebem prioridade mais elevada na fila de atividades. Além do valor, é necessário considerar os riscos e as dependências entre as atividades ao realizar essa ordenação.

5. **Time-boxing**

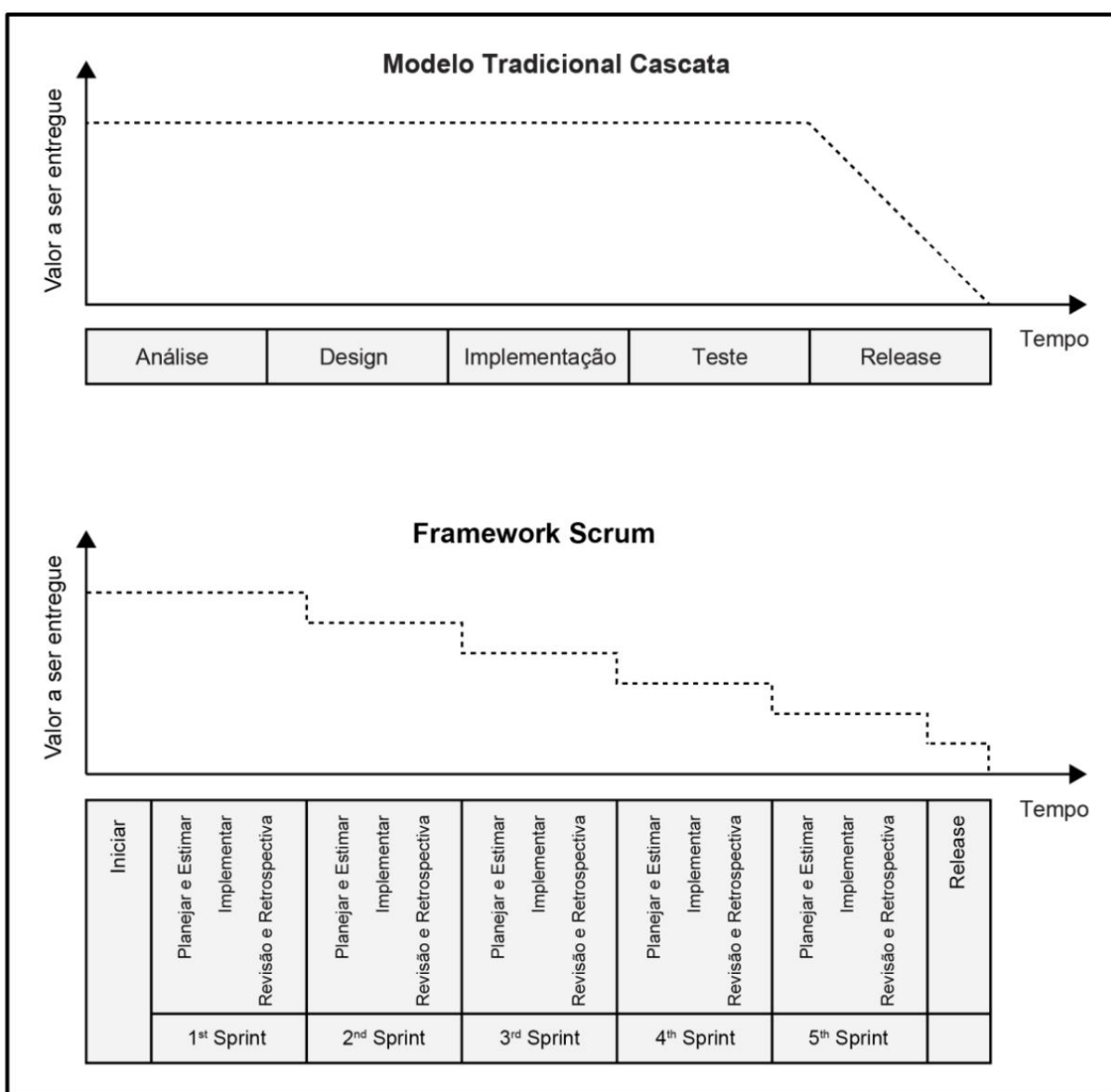
A limitação de tempo de cada evento é outro ponto muito característico deste *framework*. Essas restrições são estabelecidas antecipadamente para garantir que a equipe cumpra os prazos e as entregas não sejam atrasadas. Isso busca garantir que os membros do Scrum não usem muito tempo (ou pouco) em um trabalho específico, e que não gastem o seu tempo e energia em um trabalho no qual eles tenham pouco conhecimento. O Time-boxing pode ser aplicado em muitos processos Scrum, como na condução da Reunião Diária (que não deve ultrapassar 15 minutos) e na definição do prazo da "Sprint"(que deve durar no máximo 6 semanas). As vantagens dessa abordagem incluem a economia nas despesas gerais, o aumento da velocidade de trabalho e o aumento da produtividade. No entanto, é essencial ter cuidado ao definir os limites de tempo (time-boxing) para evitar efeitos contraproducentes. A imposição demasiadamente rígida pode gerar apreensão e desmotivação em alguns membros da equipe.

6. **Desenvolvimento Iterativo**

Esse princípio fundamenta-se na realização repetida de sprints ao longo do projeto, visando a geração contínua de valor para o produto.

Na maioria dos projetos complexos, é comum que o cliente não consiga definir completamente os requisitos ou ainda não tenha clareza sobre como deve ser o produto final. O modelo iterativo é mais flexível para garantir que qualquer alteração solicitada pelo cliente possa ser incorporada ao projeto. Conforme representado na figura 4, os projetos Scrum são concluídos de maneira iterativa, entregando valor ao longo de seu ciclo de vida.

Figura 4 – Scrum x O Modelo Tradicional Cascata (Waterfall).



Fonte: (SBOK™, 2015).

Enquanto a metodologia em cascata compreende cinco etapas e começa a fornecer valor apenas na última fase do projeto, o Scrum opera em ciclos repetidos a cada sprint. Isso viabiliza a entrega contínua de valor. No entanto, essa não é

a única vantagem do desenvolvimento iterativo.

Outra vantagem significativa é a capacidade de realizar correções ao longo do projeto, uma vez que todos os envolvidos têm uma compreensão clara do que precisa ser entregue. Isso resulta na redução do tempo e do esforço necessários para atingir o resultado final, permitindo que a equipe se auto-organize e produza resultados aprimorados.

4.1.1.2 Papéis do Scrum

- **Product Owner**

O primeiro papel é o Product Owner, ou dono do produto. Essa pessoa é encarregada por articular as tarefas a serem realizadas pela equipe e por garantir a qualidade do produto ao longo de sua execução e entrega. Sua responsabilidade é maximizar o valor do produto e do trabalho do time de Desenvolvimento. Também representa as demandas dos stakeholders frente ao Scrum Master e ao time de Desenvolvimento (SCHWABER K.; SUTHERLAND, 2017).

As principais responsabilidades do Product Owner incluem:

- Gerenciar o Product Backlog
- Manter o Product Backlog atualizado
- Manter a equipe informada sobre o Product Backlog
- Planejar a Sprint
- Definir os critérios de aprovação para as tarefas
- Aceitar/reprovar incrementos durante a Sprint

- **Scrum Master**

O termo "Scrum Master", traduzido do inglês, significa "mestre do Scrum". Sua responsabilidade principal é orientar a equipe, oferecendo suporte tanto ao Product Owner quanto ao time de desenvolvimento.

O Scrum Master procura assegurar que a equipe adote e siga os valores e práticas do Scrum. Atua como moderador nas reuniões e é um papel tipicamente exercido por um gerente de projeto ou um líder técnico, mas pode ser qualquer pessoa da equipe (SCHWABER K.; SUTHERLAND, 2017). É de sua responsabilidade nortear a equipe, oferecendo suporte tanto ao Product Owner quanto ao time de desenvolvimento, mas ao invés de cobrar resultados e performance, esse tipo de líder deve ajudar a alcançá-los.

As responsabilidades do Scrum Master incluem:

- Facilitar as Sprints;

- Dar suporte ao Product Owner;
- Remover obstáculos que possam prejudicar a equipe e o progresso do projeto;
- Servir como ponte entre diferentes áreas e partes envolvidas.

- **Time de Desenvolvimento**

O Time de Desenvolvimento constitui-se dos profissionais encarregados de fornecer os incrementos do produto ao final de cada sprint. A equipe se compromete a executar uma coleção de atividades da sprint e o Product Owner se compromete a não apresentar novos requisitos para a equipe durante a sprint. Mudanças serão permitidas apenas em outra sprint (SCHWABER K.; SUTHERLAND, 2017).

É fundamental evitar a segregação dos membros por funções, garantindo assim um senso de unidade na equipe. Dessa forma, as atividades são realizadas como um esforço coletivo, não como responsabilidades individuais.

As atribuições da equipe de desenvolvimento são:

- Executar a Sprint
- Inspeccionar e adaptar o próprio desempenho e auto-organização
- Refinar o Product Backlog
- Auxiliar no planejamento da Sprint

4.1.1.3 Processos do Scrum

Com o objetivo de proporcionar um processo adequado para o projeto, os itens priorizados do product backlog são desenvolvidos em ciclos chamados de Sprints, os quais resultam em entregas ao final de cada ciclo.

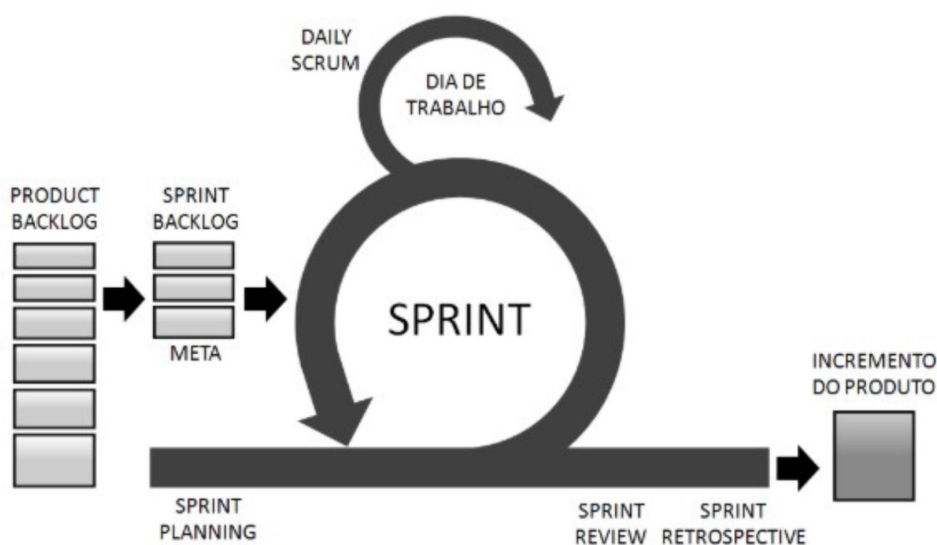
A figura 5 fornece uma visão geral do fluxo de um projeto Scrum.

O Scrum enfatiza a utilização de um conjunto de "padrões de processo de software" que se mostraram eficazes para projetos com prazos ajustados, requisitos mutáveis e alta criticidade de negócio. Cada padrão define um conjunto de atividades de desenvolvimento.

- **Product Backlog**

O Product Backlog consiste em uma lista ordenada de todos os elementos necessários para compor o software, e descreve o trabalho previsto de maneira organizada e flexível, servindo como consulta para todos os integrantes do time. Suas entregas são priorizadas e sequenciadas conforme o nível de valor que podem gerar para o cliente. Quanto mais importante for a entrega, maior a sua urgência em começar a desenvolvê-la.

Figura 5 – Fluxo de um projeto Scrum.



Fonte: Sabbagh (2013).

No início de um projeto Scrum, é comum que a lista de tarefas não esteja totalmente completa no Product Backlog. Especialmente no desenvolvimento de software, às necessidades e satisfação do cliente são constantemente exploradas para criar novas tarefas e funcionalidades, refinando progressivamente o escopo do projeto ao longo de sua execução.

- **Sprint Backlog**

A cada sprint, durante a reunião de planejamento, são escolhidos os itens que farão parte do Backlog da Sprint. Em outras palavras, esse artefato é utilizado para orientar o desenvolvimento dos entregáveis que o compõem.

- **Incremento do produto**

No Scrum, o foco principal é a entrega de produtos que satisfaçam as necessidades dos clientes, em pequenos incrementos iterativos que sejam utilizáveis. Dessa forma, o incremento representa a soma dos itens do Product Backlog que devem estar prontos ao final da sprint e caracteriza o trabalho concluído.

4.1.1.4 Eventos do Scrum

- **Sprint**

Dentro da metodologia Scrum, uma Sprint é o período de algumas semanas no qual uma versão do produto é desenvolvida. A essência da Sprint é cum-

prir uma meta específica dentro de um prazo determinado. Para (SCHWABER K.; SUTHERLAND, 2017), a sprint é o coração do Scrum.

É um timeboxed de um mês ou menos, durante o qual um incremento de produto potencialmente liberável é criado. Uma nova sprint inicia logo assim que a anterior termina.

As sprints são definidas por meio da reunião de planejamento e seu acompanhamento é feito durante as reuniões diárias. As entregas ocorrem nas reuniões de revisão, onde também pode ser conduzida uma retrospectiva da sprint.

A realização de uma Sprint envolve a ocorrência de cinco eventos, são eles:

1. **Reunião de planejamento da Sprint:**

Durante essa reunião, a equipe define o que será entregue ao final da Sprint (objetivo) e discute como realizará o trabalho ao longo do período da Sprint. Todos os membros da equipe participam dessa reunião, e é nesse momento que as tarefas são distribuídas. Trata-se de uma reunião colaborativa, onde o Product Owner define o objetivo da Sprint, enquanto o Scrum Master, em conjunto com o Time de Desenvolvimento, trabalha na produção dos componentes de software a serem incrementados. O planejamento é baseado no Product Backlog, que serve como fonte de informações. Tendo definido o objetivo da sprint e selecionado os itens de Backlog do produto da sprint, o time de Desenvolvimento decide como irá construir essas funcionalidades durante a sprint e transformá-las em um incremento de produto “Pronto” (SCHWABER K.; SUTHERLAND, 2017).

2. **Execução:**

Após a conclusão da Reunião de Planejamento, o time inicia o trabalho de acordo com os planos estabelecidos e as tarefas atribuídas. Durante essa fase, a equipe se dedica à execução das atividades, mantendo sempre o foco nos requisitos do produto e no prazo final da Sprint.

3. **Reunião Diária (Daily):** A Daily é, como o próprio nome sugere, um evento diário durante a execução da Sprint. Tem como objetivo permitir que a equipe avalie o progresso das tarefas e compartilhe atualizações de forma rápida e objetiva. Geralmente, essas reuniões são breves, durando no máximo 15 minutos, e cada membro da equipe responde a três perguntas fundamentais:

- * O que eu fiz ontem?
- * O que eu farei hoje?
- * Estou enfrentando algum obstáculo?

Essas respostas proporcionam uma visão clara do andamento do trabalho, facilitam a identificação de possíveis impedimentos e promovem a colaboração e a resolução de problemas dentro da equipe.

4. **Revisão da Sprint:**

Esta reunião ocorre no final de cada Sprint e tem como objetivo inspecionar o incremento e adaptar o Product Backlog, se necessário. A participação de todos os envolvidos, incluindo o Product Owner, Scrum Master, o Time de Desenvolvimento e as partes interessadas-chave (Stakeholders), é crucial. Os principais pontos de discussão envolvem o esclarecimento dos itens que ficaram e não ficaram “prontos”, a reflexão sobre o que foi bem e os problemas encontrados ao longo do período. Tudo isso pode gerar sugestões para a nova versão do Backlog. A Revisão da Sprint fornece informações importantes para o planejamento da próxima Sprint, com base nas lições aprendidas e no feedback obtido durante a reunião. Essa abordagem colaborativa e de inspeção contínua é fundamental para garantir a adaptação e melhoria contínua do processo.

5. **Retrospectiva da Sprint:**

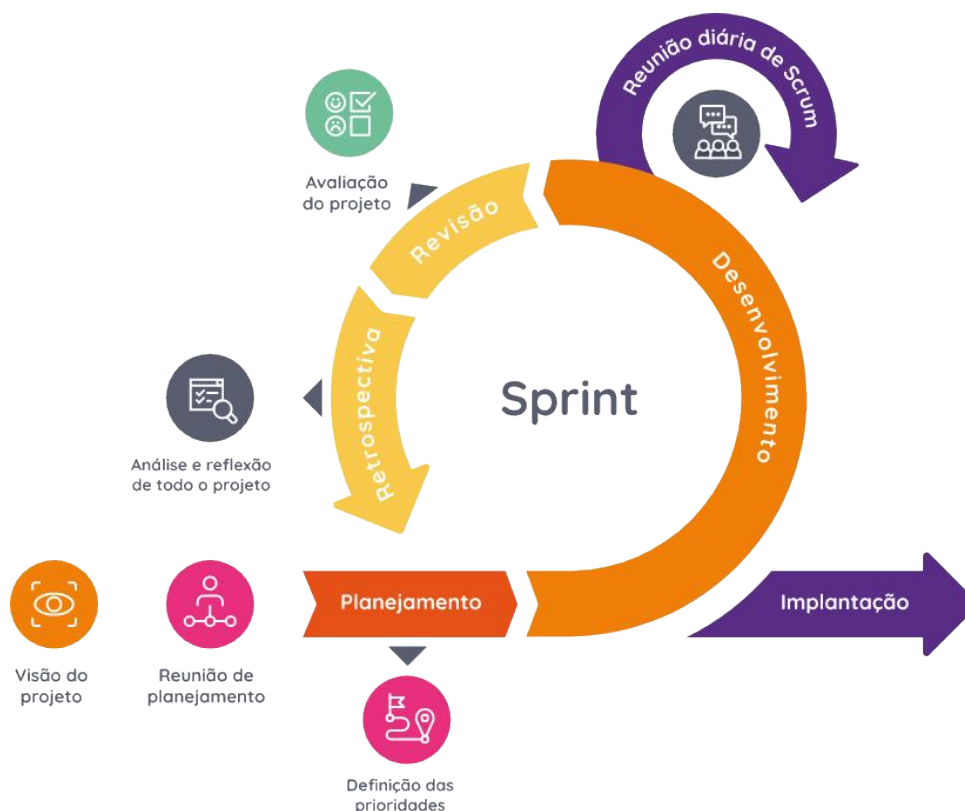
Enquanto a Revisão da Sprint tem como objetivo avaliar o produto e o desempenho da equipe, a retrospectiva oferece à equipe a oportunidade de se avaliar internamente. É o momento em que os membros podem refletir sobre suas práticas e identificar oportunidades de melhoria. O encerramento desta reunião oficializa o término da Sprint. A figura 6 a seguir mostra de maneira resumida como todo esse processo se relaciona.

4.2 DESIGN THINKING

Vencedor de um prêmio Nobel, o economista Herbert Simon, da Universidade de Stanford, propôs e desenvolveu a ideia de que projetos de engenharia deveriam ser centrados no ser humano. O Design Thinking é um método de inovação que parte com foco na solução de problemas, que teve sua origem na interseção de diversas disciplinas, como engenharia, psicologia, antropologia e artes. Trata-se de uma abordagem, uma forma de pensar e encarar problemas, focada na empatia, colaboração e experimentação (SIMON, 1969).

De acordo com (BROWN, 2010), o design thinking é uma abstração do modelo mental que designers utilizam há muitos anos para dar vida a ideias. Seus conceitos são acessíveis e aplicáveis a qualquer pessoa, em diversos contextos de negócios ou sociais. Não é um conceito nem uma prática nova, pois já existe no âmbito do design,

Figura 6 – Como funciona a Sprint.



Fonte: PAIXÃO, Marina(2021).

seja de forma consciente ou inconsciente.

O termo "Design Thinking" foi oficialmente reconhecido a partir de 2003, quando o consultor de design da IDEO, David Kelley, começou a popularizar o conceito.

Segundo (BONINI; SBRAGIA, 2011), o Design Thinking transcende o mero conceito de design, tornando-se uma ferramenta estratégica para transformação organizacional. Essa abordagem não apenas influencia processos criativos, mas também se integra aos aspectos estratégicos e operacionais das empresas. O Design Thinking, segundo os autores, atua como um catalisador de inovação, estimulando a criatividade e proporcionando soluções centradas no usuário.

Atualmente, o Design Thinking é empregado em diversas áreas como uma ferramenta de resolução de problemas complexos, inspiração para a criatividade e propulsor de inovações. A metodologia se destaca por sua ênfase na empatia, colaboração e iteração contínua. Ao adotar uma abordagem centrada no ser humano, o Design Thinking busca compreender profundamente as necessidades e experiências dos usuários, resultando em soluções mais alinhadas com as demandas reais.

O diferencial do Design Thinking reside na capacidade dos processos de trabalho dos designers em auxiliar pessoas de outras áreas - como a engenharia no caso deste projeto - a extrair de maneira sistemática, ensinar, aprender e aplicar técnicas

centradas no ser humano para abordar problemas de forma criativa e inovadora.

Grandes marcas globais, como Apple, Google, Samsung e General Eletric (GE), incorporaram rapidamente a abordagem do Design Thinking. Além disso, o Design Thinking está sendo incorporado nos currículos das principais instituições de ensino superior do mundo, incluindo a d.school em Stanford, Harvard e Massachusetts Institute of Technology (MIT).

A forma de pensar do Design Thinking pode trazer inúmeros benefícios para projetos onde é aplicada. Essa metodologia oferece uma estrutura flexível e orientada para o usuário, permitindo uma análise aprofundada do problema em questão e estimulando a criatividade na busca por soluções inovadoras. Além disso, a ênfase na colaboração e iteração contínua pode enriquecer significativamente o processo de pesquisa e desenvolvimento do projeto.

Em suma, o Design Thinking transcende as fronteiras do design tradicional, emergindo como uma abordagem transformadora capaz de impulsionar a inovação em diversos contextos. Sua aplicação em projetos de engenharia não apenas fortalece a qualidade das soluções propostas, mas também proporciona uma experiência enriquecedora e alinhada com as demandas contemporâneas de resolução de problemas.

4.2.1 Etapas do Design Thinking

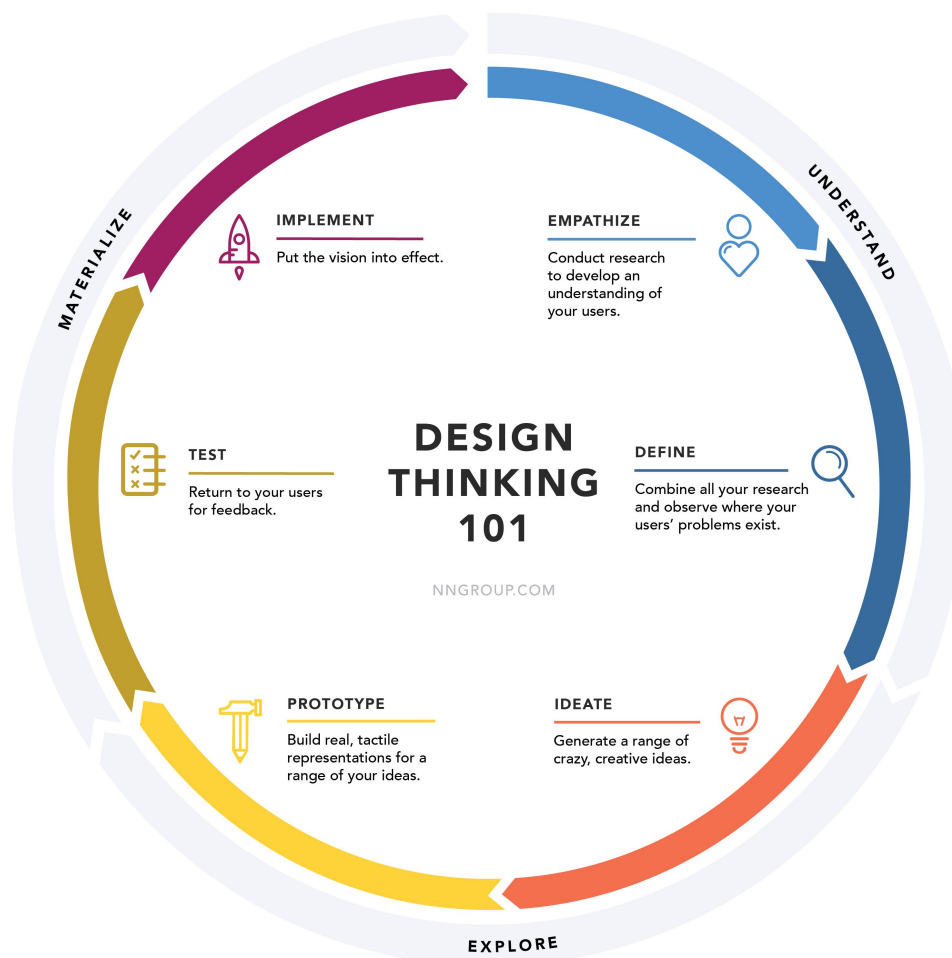
O Design Thinking é, essencialmente, um processo exploratório para impulsionar a inovação contínua. Não há uma maneira correta ou precisa de seguir esse processo. Embora forneça pontos de partida e referências úteis durante a jornada, o continuum da inovação pode ser visto mais como um sistema de espaços que sobrepõe do que um sistema de passos ordenados (BROWN, 2010).

De acordo com o (GROUP, 2016), a estrutura do Design Thinking segue um fluxo geral de 3 fases: compreender, explorar e materializar. Mas dentro destas etapas principais, encontram-se na verdade 6 fases: empatia, definição, ideação, protótipo, teste e implementação. No entanto, (BONINI; SBRAGIA, 2011) enfatizam que essas fases não seguem uma ordem fixa e não ocorrem necessariamente apenas uma vez. O ciclo pode e deve ser repetido conforme necessário para identificar a melhor solução para o problema em questão. A figura 7 mostra o processo, etapas e sub-etapas de Design Thinking proposto pelo Nielsen Norman Group.

1. Empatia:

Esta primeira etapa é voltada para a realização de pesquisas, a fim de compreender as necessidades das pessoas envolvidas no problema do projeto, que podem ser consumidores, colaboradores, público, usuários... O objetivo é realmente coletar o máximo de observações sobre o contexto do projeto para exercer jus-

Figura 7 – Design Thinking.



Fonte: (GROUP, 2016).

tamente o que o nome da etapa sugere: empatia. Busca-se entender o que as pessoas envolvidas precisam, gostam e desejam.

2. Definir:

Neste momento é feita a consolidação de todas as informações obtidas na etapa anterior, e a consequente análise dos dados para identificar as áreas onde os usuários têm problemas. Ao identificar as necessidades dos usuários, é importante destacar as oportunidades de inovação e insights, pois elas orientarão o desenvolvimento das soluções do projeto.

3. Ideação:

Nesta etapa devem ser realizadas sessões de criação para gerar diversas ideias que atendam às necessidades dos usuários que tiverem sido identificadas na fase de definição. O intuito é gerar uma série de ideias extravagantes e criativas, com

foco em quantidade e não na qualidade. Para isso podem ser usadas diferentes técnicas criativas, sendo o mais comum o brainstorming (tempestade de ideias, em tradução livre). Para todas, é importante ter um ambiente com total liberdade, livre de julgamentos, para que toda a equipe possa colaborar.

4. **Protótipo:**

O propósito desta fase é entender quais elementos das ideias geradas anteriormente são eficazes e quais não são. Para isso, devem ser criadas representações tangíveis e concretas de ideias selecionadas, tornando-as palpáveis. Durante esta etapa, é importante considerar a relação entre o impacto e a viabilidade das suas ideias, obtendo feedback por meio dos protótipos gerados.

5. **Teste:**

É o momento de colocar os protótipos para teste com clientes reais, e verificar se as soluções projetadas atendem os objetivos planejados, e se melhorou a forma como os usuários se sentem, pensam ou realizam suas tarefas.

6. **Implementação:**

Esta etapa é a mais crucial do design thinking, mas muitas vezes é negligenciada. Conforme defendido por Don Norman (GROUP, 2016), "precisamos de mais design *doing*". Por mais impactante que o design thinking possa ser para uma organização, ele só conduz à verdadeira inovação se a visão for efetivamente implementada. O sucesso do design thinking está intrinsecamente ligado à sua capacidade de transformar um aspecto da vida do usuário final. Este sexto passo - a implementação - é crucial para alcançar esse sucesso.

4.3 METODOLOGIA DO PROJETO

Conforme citado anteriormente, este projeto une as práticas de Design Thinking ao desenvolvimento ágil (com a metodologia Scrum).

(STICKDORN; SCHNEIDER, 2014) defendem que o Design Thinking tem como finalidade a delimitação de um problema, para que a partir disso seja possível pensá-lo de modo criativo e gerar novas soluções. O presente projeto buscou se beneficiar dessa abordagem para gerar uma solução de engenharia de automação que realmente solucione as necessidades de seus usuários, no caso, com foco no Canal Isto é Avaí.

De forma complementar, um dos principais motivos que impulsionou a popularidade da metodologia Scrum, e a sua escolha para ser usada neste projeto, é a sua aplicação em situações onde uma definição abrangente do escopo não é inicialmente possível, como no presente projeto.

De acordo com o Guia SBOK™, os princípios do Scrum não são negociáveis, mas os aspectos e processos do método, no entanto, podem ser modificados para

atender aos requisitos do projeto ou da organização. Embora este seja um trabalho individual, e o Scrum esteja focado na condução de trabalhos em equipe, os princípios e as lições dessa estrutura podem ser aplicados como forma de guiar a organização e gestão do projeto (como a definição de backlog, assim como o trabalho planejado e desenvolvido em sprints).

As atividades gerais desenvolvidas em um projeto de software são requisitos, análise, projeto, implementação e testes. Essas atividades estão presentes em todas as metodologias, assim como neste projeto, e diferem-se apenas na forma de serem trabalhadas. Temos então uma combinação de metodologias que visa à criatividade, inovação e entregas de valor. Partindo desde a definição do problema, até a implementação da aplicação PWA proposta.

O presente projeto será então desenvolvido de acordo com as seguintes etapas, que estarão divididas em sprints:

1. Planejamento

- Imersão
- Definição

2. Design

- Ideação
- Prototipação
- Teste (do protótipo, com usuários)

3. Desenvolvimento

- Desenvolver funcionalidade 1 + Integrar e testar
- Desenvolver funcionalidade 2 + Integrar e testar
- Desenvolver funcionalidade 3 + Integrar e testar
- Desenvolver funcionalidade 4 + Integrar e testar

4. Teste

5. Lançamento

- Lançamento beta (demo)
- Feedback dos Usuários

6. Revisão

Na etapa de Planejamento será feito o levantamento de dados sobre o produto em si, seus usuários e stakeholders. Essa imersão inicial culminará na definição de quais são os problemas a serem solucionados pelo projeto em questão. A partir daí, na etapa de Design, será realizada a ideação dessas soluções, desenvolvidos os protótipos e os testes com usuários antes de dar início à materialização de fato.

A etapa de desenvolvimento é dividida por cada funcionalidade, e além dos testes de cada uma delas, também será feita uma etapa de teste final, antes de um lançamento demo para teste com número de usuários controlado, e então feedback e ajustes.

5 O PROJETO

Este capítulo apresenta o desenvolvimento de cada etapa da metodologia proposta, que resultou na aplicação PWA para o Canal Isto é Avaí, objetivo deste projeto.

5.1 ETAPA 1: PLANEJAMENTO

Nesta etapa é realizado o levantamento de dados acerca de todo o contexto do projeto, a instituição no qual se aplica, seus usuários e stakeholders. Através desta imersão e definição iniciais, tem-se maior clareza dos objetivos do projeto e é dado início às etapas de ideação e desenvolvimento das soluções.

5.1.1 Imersão

A imersão é a primeira fase em um processo de inovação guiado pelo Design Thinking. Trata-se de entender o problema antes de pensar na solução. É nesse momento que a equipe (no caso, o autor) se aproxima do contexto do projeto e inicia o processo de identificação e classificação dos problemas a serem resolvidos.

Muito importante para dar o input no pensamento e nas ações do projeto, essa etapa consiste em levantar informações sobre os problemas reais do negócio, usuários finais, stakeholders e do próprio mercado - a fim de entender a jornada e propor soluções tangíveis para o projeto.

Existem diferentes ferramentas que podem ser utilizadas para as pesquisas de contextualização do problema, coleta de dados e formulação de hipóteses. Para este projeto foi realizada a Pesquisa Desk (Desk Research): uma forma útil, rápida e financeiramente viável para as empresas realizarem boas pesquisas.

5.1.1.1 Desk Research

Este formato de pesquisa consiste na busca de informações sobre o tema do projeto em fontes diversas, baseada a partir de informações, documentos e fontes já existentes (websites, livros, revistas, blogs, artigos, entre outros).

É uma pesquisa secundária, com objetivo de concretizar o entendimento das mais variadas situações e contextos para a tomada de decisões. Projetos, protótipos e testes de produtos (como o presente trabalho) costumam aplicar essa técnica para coletar uma grande quantidade de dados e informações, de diferentes fontes, para seus objetivos. Assim, garantem um maior índice de compreensão acerca dos seus assuntos de projeto.

Para desenvolver a solução para um problema, é importante conhecer as raízes do desafio que se quer superar. Os primeiros passos para isso são compreender o que está acontecendo, classificar o que deve ser resolvido e identificar as oportunidades.

Isso tudo pode ser feito através de pesquisas de contextualização do problema, coleta de dados e formulação de hipóteses.

Na Desk Research desenvolvida, foram levantados dados de diferentes frentes para melhor compreensão do contexto.

Canal Isto é Avaí:

A empresa para a qual o projeto está sendo desenvolvido. Trata-se de um canal e comunidade para a torcida do Avaí, time da capital catarinense. Hoje a comunidade conta com três principais pilares: YouTube, WhatsApp e Instagram. Em cerca de 2 anos a comunidade, e o negócio, vem crescendo rapidamente.

No YouTube são mais de 5 mil inscritos e 540.000 visualizações até hoje, além de 150 assinantes do canal. Nos grupos do Whatsapp há mais de 800 participantes muito ativos dentro dos 3 grupos no aplicativo, além do grupo especial de membros pagantes. No Instagram são mais de 4.700 seguidores engajados que consomem o conteúdo postado diariamente.

Além dos membros e espectadores, a história vem sendo construída ao lado de apoiadores e patrocinadores, e contou também com a participação do histórico e lendário comentarista esportivo Miguel Livramento, que fez parte do canal durante os últimos 6 meses de sua vida.

A figura 8 ilustra o levantamento feito durante a Desk Research.

Figura 8 – Levantamento na Desk Research.



Fonte: do Autor.

Indústria do futebol:

O futebol é reconhecido como o esporte mais popular do mundo, e tornou-se também o mais rentável deles dentro da crescente economia do entretenimento. Ele vai muito além de ser apenas uma atividade de lazer. Hoje os clubes têm variadas fontes de receitas e contam com toda uma estrutura mercadológica dentro da indústria futebolística. Essas receitas bilionárias ajudam a impulsionar a economia local, nacional e mundial, gerando empregos diretos e indiretos em setores relacionados.

Segundo pesquisa realizada pela EY, o futebol movimenta um total de R\$ 52,9

bilhões na economia do País, o que representa 0,72% do total do PIB brasileiro, mostrando que R\$ 37,8 milhões são de efeitos indiretos.

No Brasil, em média 25,7 milhões de pessoas acompanham as partidas pela televisão (TV). Locais de encontro, como bares, recebem uma quantidade próxima a 15 milhões de pessoas. O esporte é cultural, e está atrelado a paixão e entretenimento de um povo ou local e por isso move toda uma indústria que trabalha para que possa acontecer.

A paixão do torcedor agrega valores bilionários aos seus clubes de coração e gera um poder de influência sem precedentes dos seus clubes e jogadores preferidos sobre uma comunidade, cidade e até um país. Hoje no Brasil, a mídia identificada de clubes vem crescendo cada vez mais e tomando espaço da mídia tradicional, principalmente dentro de plataformas sociais

Comunidades virtuais:

Talvez a comunidade virtual mais conhecida do mundo, a Wikipédia, que não tem objetivo de vender nenhum produto, seja uma prova da força que uma comunidade virtual pode ter.

Dentro desse contexto, surgem comunidades virtuais focadas nos esportes: National Basketball Association (NBA) Fantasy, World Surf League (WSL) Fantasy e Cartola, que alinham interação e engajamento. Aumentando o foco para o futebol, também temos uma infinidade de comunidades criadas por torcedores de clubes. Essas comunidades virtuais têm se mostrado uma maneira poderosa de conectar fãs de todo o mundo, proporcionando um espaço para compartilhar informações, expressar opiniões, celebrar vitórias e superar derrotas. Essas comunidades fortalecem o sentimento de pertencimento e a paixão pelo esporte, tornando o futebol uma experiência social e globalmente conectada. Hoje no Brasil, a mídia identificada de clubes vem crescendo cada vez mais e tomando espaço da mídia tradicional, principalmente dentro de plataformas sociais.

5.1.2 Definição

A segunda etapa dentro da fase de Planejamento, é a etapa de Definição. Este é o momento de sintetizar as informações coletadas na fase anterior para a iniciar a geração de insights. O objetivo desta etapa é entender com clareza em qual cenário os stakeholders do projeto estão envolvidos, e o que será fundamental para a próxima fase.

Durante esta fase foi utilizada uma ferramenta para mapear os principais pontos de dor e necessidade das pessoas envolvidas no projeto, de forma que as funcionalidades desenvolvidas venham para solucionar problemas relevantes para os mesmos.

Em decorrência do curto prazo para a realização da etapa, foi necessário adaptar a ferramenta, que deriva do Mapa de Empatia, já que não seria viável inserir neste

momento as entrevistas com usuários para a identificação de padrões de comportamento, pois seriam muitas personas diferentes (a audiência do Canal Isto é Avaí inclui: torcida organizada, pré-adolescentes, idosos, torcedores aficionados pelo time, torcedores de outros times, entre outros). Dessa forma, foi aplicado um quadro de dores e necessidades, levando em consideração a visão da equipe do canal e também a audiência do Isto é Avaí (de forma geral).

A figura 9 retrata um Mapa de Empatia, que serviu de base para a adaptação da ferramenta.

Figura 9 – Mapa de empatia.



Fonte: CanvasBrasil (2012).

Foram consideradas as personas do projeto:

Equipe do Canal

É composta por um grupo de pessoas envolvidas na criação, produção e gerenciamento do conteúdo publicado nas redes sociais.

Audiência

O público-alvo do canal, composto por indivíduos que acompanham regularmente todo o conteúdo publicado, aqueles que o assistem ocasionalmente e assinantes. Os assinantes do canal efetuam contribuições financeiras com o intuito de promover o desenvolvimento da plataforma. Essa modalidade representa uma das estratégias de monetização adotadas pelo YouTube, na qual o inscrito realiza a assinatura do canal e desfruta de determinados privilégios, cuja natureza varia conforme a proposta de cada canal específico.

A figura 10 ilustra o quadro de dores e necessidades elaborado para o projeto.

Figura 10 – Quadro de dores e necessidades.

	<p style="text-align: center;">EQUIPE</p>	<p style="text-align: center;">AUDIÊNCIA</p>
DORES	<p>acompanhar as regras dos algoritmos das redes sociais</p> <p>produzir conteúdos para as redes sociais</p> <p>manter o relacionamento para os patrocinadores</p>	<p>quarta saber quais patrocinadores tem parceria com o podcast</p>
NECESSIDADES	<p>aumentar a fidelização da audiência</p> <p>ter mais argumentos de vendas para os patrocinadores</p> <p>aumento de receita</p> <p>viabilidade (cônego)</p>	<p>acompanhar conteúdos de tempo de conexão por fora das mídias tradicionais</p> <p>espaço para discutir sobre o Avaf</p> <p>acompanhar um programa sobre o Avaf para "likes", torcedores também</p>

Fonte: do Autor.

Com os usuários identificados, bem como os desafios, e oportunidades, passa-se para a próxima fase.

5.2 ETAPA 2: DESIGN

Esta etapa é uma das fases centrais e envolve a criação de soluções criativas e inovadoras para os problemas ou desafios identificados anteriormente. Essa etapa é fundamental para desenvolver soluções que atendam às necessidades dos usuários de forma eficaz. Os passos dessa etapa são: Ideação, Prototipação e Teste.

5.2.1 Ideação

Esta fase é o momento de ter ideias e criatividade, para apresentar os insights e visão do projeto. Em síntese: inovar. A partir da identificação de dores e necessidades dos usuários (equipe e audiência), foi feito um brainstorming (figura 11) para gerar ideias de como solucionar esses pontos levantados.

As ideias geradas foram categorizadas (figura 12) para a melhor compreensão do contexto e em seguida aplicadas em uma Matriz de Priorização (figura 13) para auxiliar na tomada de decisão sobre quais funcionalidades deveriam seguir para desenvolvimento no projeto.

5.2.2 Prototipação

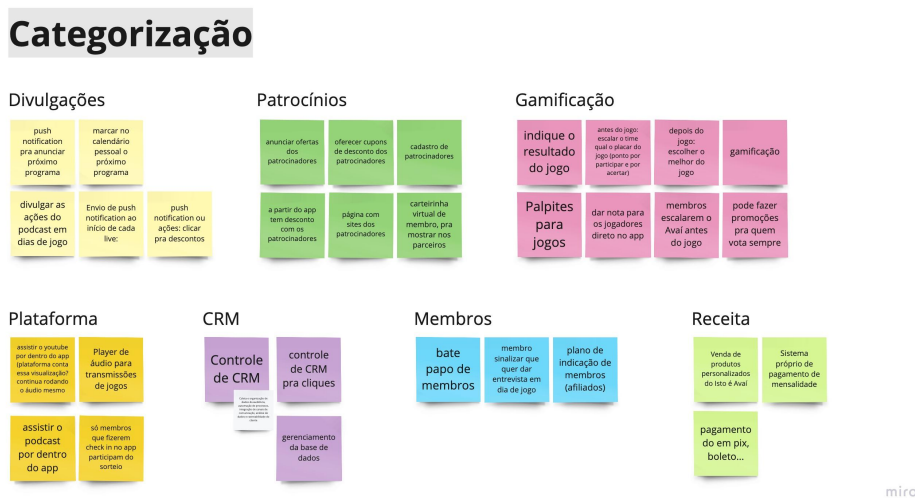
Na etapa de prototipação é importante tangibilizar as ideias geradas anteriormente, para então testar e avaliar sua viabilidade e impacto para com os usuários e

Figura 11 – Brainstorming.



Fonte: do Autor.

Figura 12 – Categorização.



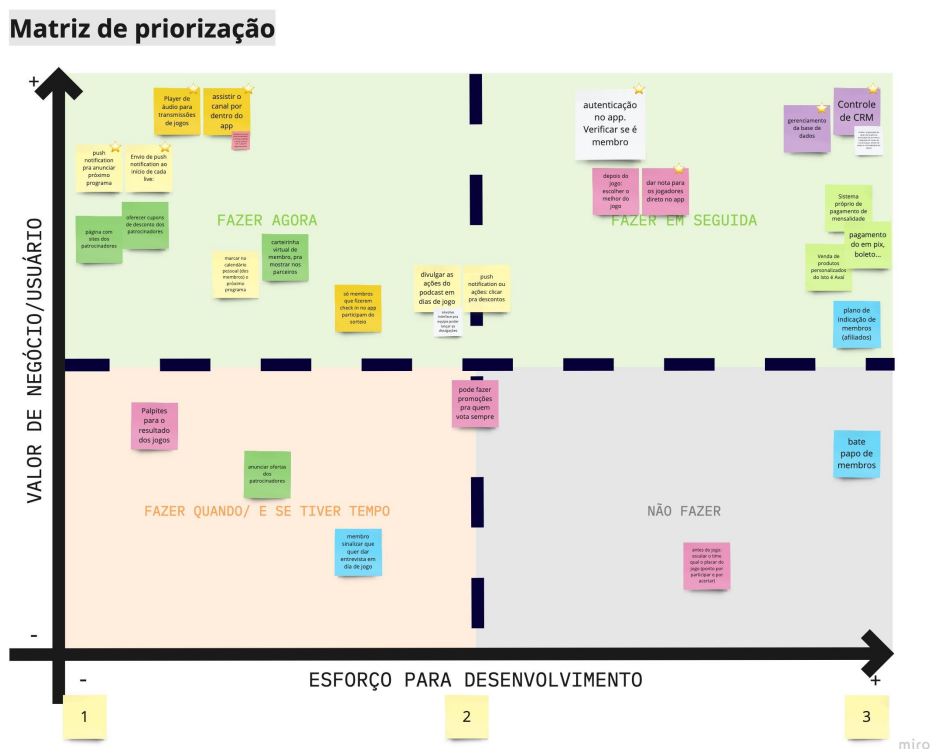
Fonte: do Autor.

stakeholders do projeto.

Para começar a tornar palpáveis as soluções propostas, foi feita mais uma sessão de ideação, mas dessa vez com foco apenas nas ideias priorizadas na etapa anterior.

Esta nova dinâmica de ideação foi facilitada pela designer do Canal Isto é Avai,

Figura 13 – Matriz de Priorização.



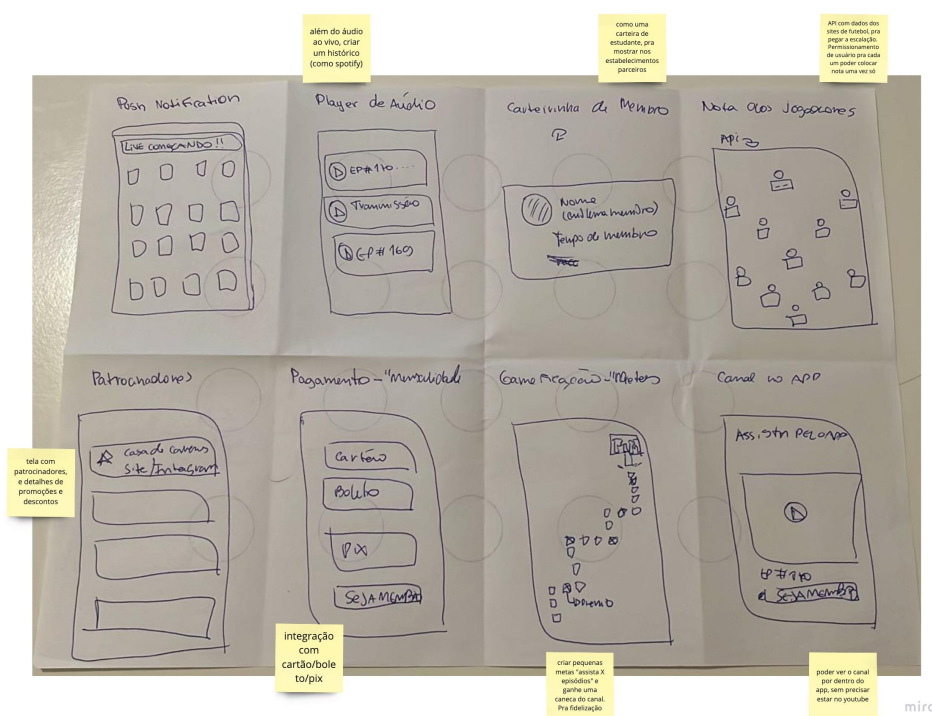
Fonte: do Autor.

e utilizou a técnica do Crazy 8's (figura 14) para explorar ideias e criar soluções. Este recurso foi desenvolvido pelo designer Jake Knaap, e é um exercício rápido que desafia as pessoas a esboçar oito ideias distintas em oito minutos. Pode-se descrever esse processo como uma prática de esboço rápido, que se concentra em evitar ideias clichês e óbvias (aquelas que surgem inicialmente) e, em vez disso, apresentar soluções mais inovadoras para o problema em questão. É bastante útil para projetos de interface, pois une a exploração de ideias com uma visualização rápida desse processo.

Antes de seguir para o desenho das telas, o autor desenvolveu um breve *User Flow* do aplicativo, ou seja, o fluxo do usuário - similar aos Casos de Uso, já conhecidos do desenvolvimento de software. O ideal é que esses diagramas sejam feitos antes de desenhar protótipos como *wireframes* ou *mockups*, pois ajudam a desenhar o caminho que seus usuários terão que percorrer para chegar no objetivo do seu negócio e obter o valor que procuram.

Assim que foi finalizado o fluxo de usuário, o autor deu início a estudos mais aprofundados de viabilidade técnica das funcionalidades priorizadas, enquanto a designer do Canal Isto é Avai desenvolveria os protótipos para teste com usuários. No entanto, algumas das funcionalidades escolhidas se mostraram mais complexas do que o estimado e o teste acabou por ser despriorizado, tendo em vista que seria inves-

Figura 14 – Crazy 8's.



Fonte: do Autor.

tido um montante de tempo e as funcionalidades a serem testadas não eram mais as mesmas que seriam desenvolvidas.

Ainda assim, a designer desenhou algumas telas (figura 16) que seriam implementadas mais tarde pelo autor. Essas telas foram desenvolvidas por ela no Figma, uma aplicação Web colaborativa para a concepção de interfaces, que facilita levar ideias do design à produção com eficiência.

5.2.3 Teste (do protótipo, com usuários)

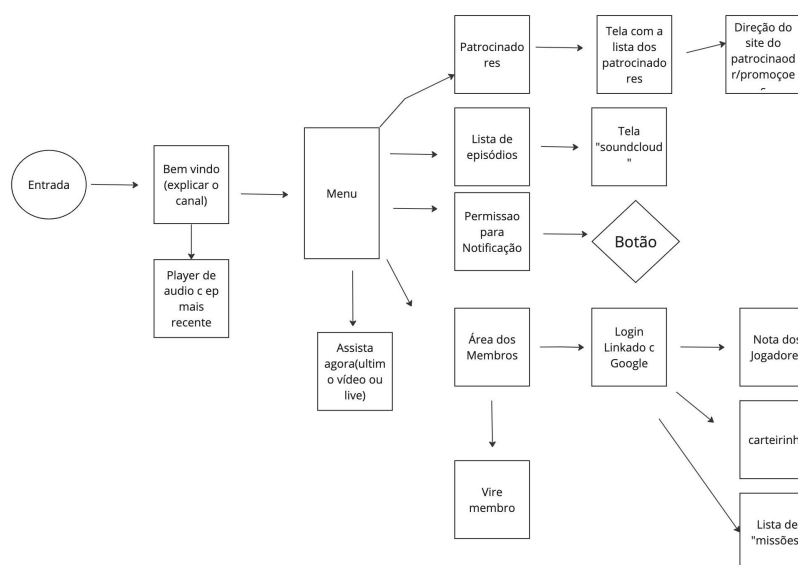
Conforme explicado no final da etapa anterior, esta fase de testes não aconteceu em decorrência de uma repriorização das funcionalidades e etapas de desenvolvimento do projeto.

Nesse momento, fica evidente a importância do uso de metodologias não lineares, que permitem mudanças de rota no decorrer do projeto, como o Design Thinking e o Scrum.

5.3 ETAPA 3: DESENVOLVIMENTO

Nesta etapa, conforme o Capítulo 4 de Metodologia, acontece o desenvolvimento, integração e testes das funcionalidades propostas. No entanto, antes de

Figura 15 – User Flow.



Fonte: do Autor.

desenvolvê-las é importante ter claras algumas informações do projeto que serão discutidas a seguir.

5.3.1 Desenvolvimento da solução

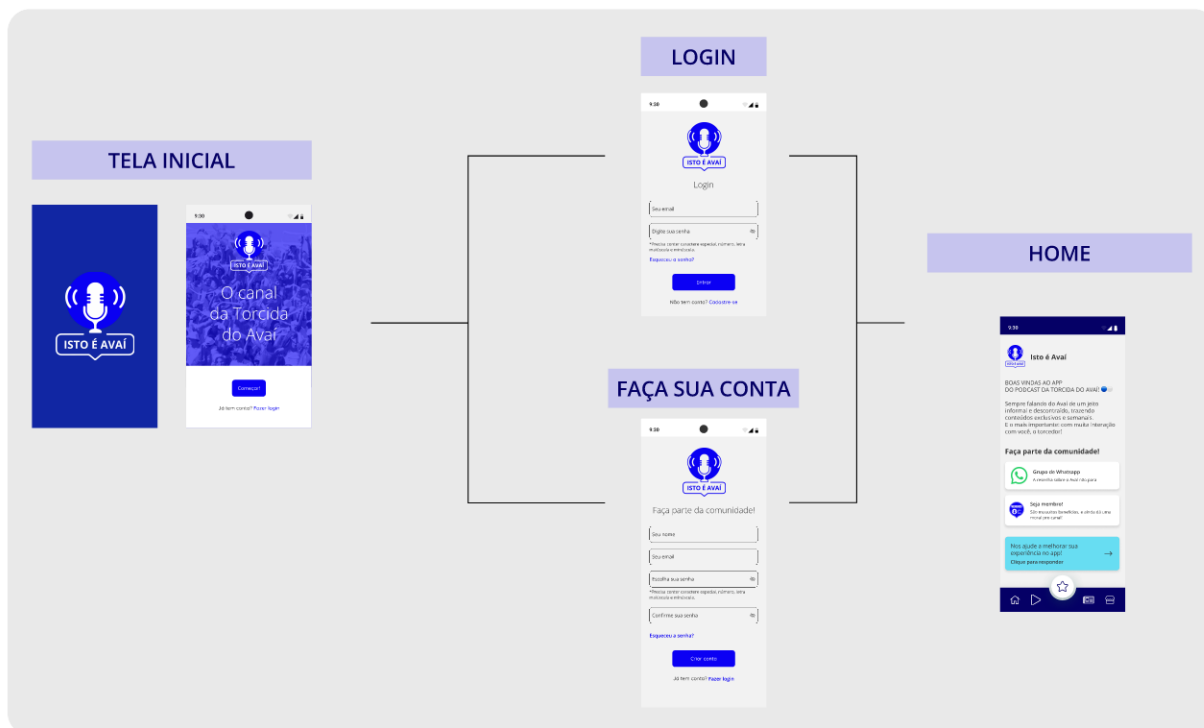
5.3.1.1 Preparação do ambiente

Para iniciar um projeto em Flutter, é necessário que sua Software Development Kit (SDK) seja instalada, bem como um conjunto de ferramentas para Android. É importante verificar se a instalação de todos os pré-requisitos foi cumprida, e para isso foi utilizado o comando *flutter doctor* (figura 17).

Por este projeto utilizar o Flutter, o comando utilizado para a criação do projeto é o *flutter create nomedoprojeto*. Após criar o projeto, ele deve ser aberto no ambiente de desenvolvimento utilizado, neste caso o Visual Studio Code (VSCode). Também foi necessário instalar a extensão do Dart e do Flutter. Após criar o novo projeto, tem-se acesso a uma estrutura de arquivos e pastas necessários para a execução do projeto em Flutter.

- android: arquivos de compilação para Android
- ios: arquivos de compilação para iOS
- lib: arquivos principais do nosso projeto
- linux: arquivos de compilação para Linux

Figura 16 – Design das telas no Figma.



Fonte: do Autor.

Figura 17 – Comando flutter doctor.

```
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 3.13.9, on Microsoft Windows [versão 10.0.19045.3570], locale pt-BR)
[✓] Windows Version (Installed version of Windows is version 10 or higher)
[✓] Android toolchain - develop for Android devices (Android SDK version 33.0.2)
[✓] Chrome - develop for the web
[✓] Visual Studio - develop Windows apps (Visual Studio Community 2022 17.7.4)
[✓] Android Studio (version 2022.2)
[✓] VS Code (version 1.84.0)
[✓] Connected device (3 available)
[✓] Network resources

• No issues found!
```

Fonte: do Autor.

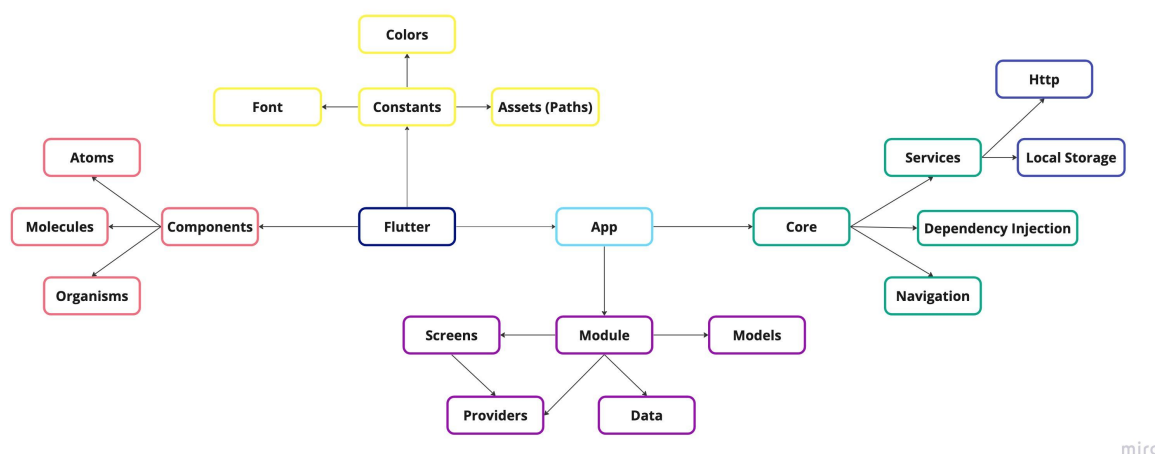
- test: arquivos de testes
- macos: arquivos de compilação para macOS
- web: arquivos de compilação para web
- windows: arquivos de compilação para Windows
- .gitignore: arquivos e pastas ignorados pelo git

- pubspec.lock: arquivo gerado a partir do pubspec.yaml
- pubspec.yaml: arquivo de informações e dependências do projeto

5.3.1.2 Arquitetura da aplicação

A partir do estudo dos diagramas anteriores e também com auxílio dos conhecimentos adquiridos nas etapas anteriores, foi elaborado o diagrama da arquitetura da aplicação (figura 18). Conforme explicado no Referencial Teórico, a arquitetura foi baseada no modelo MVC, separando as preocupações da aplicação.

Figura 18 – Diagrama da arquitetura da aplicação.



Fonte: do Autor.

Utilizando o modelo MVC, dentro da pasta *lib*, as *features* foram divididas desta maneira (figura 19).

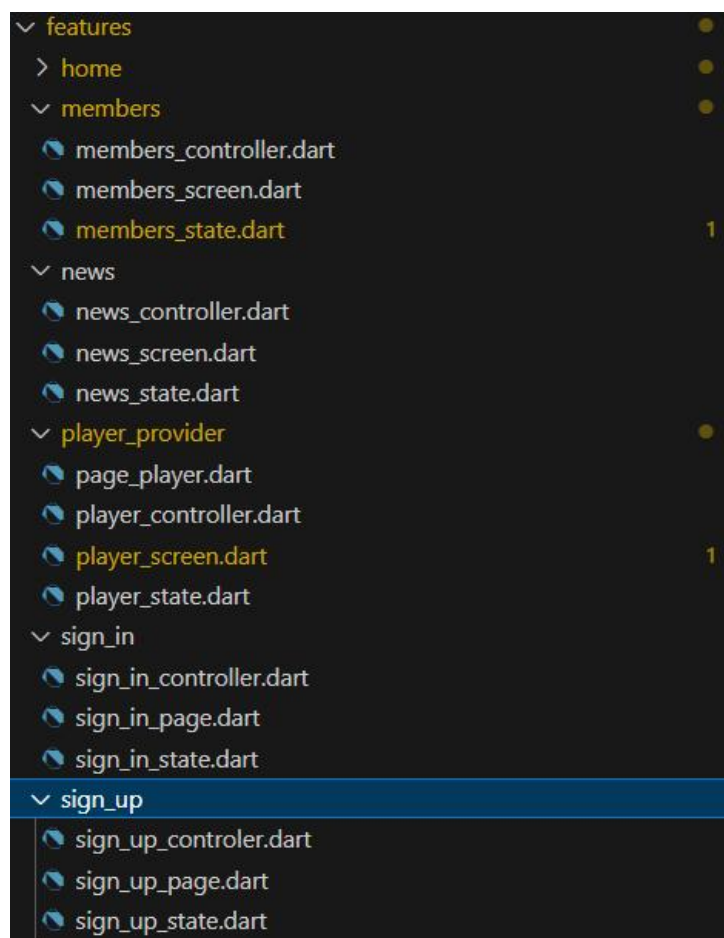
5.3.1.3 Casos de Uso

Para uma análise mais evidente das necessidades e requisitos da aplicação, foi desenvolvido um diagrama de caso de uso para uma compreensão mais clara. Através desse diagrama, foram identificados quatro atores externos, sendo o usuário do aplicativo o principal deles. Na figura 20 é apresentado o diagrama de caso de uso que representa o ator principal da aplicação e outros três atores externos, sendo o ator principal (usuários do aplicativo) responsável pela maior parte dos casos de uso da aplicação.

5.3.1.4 Funcionalidades

- Cadastro

Figura 19 – Features.



Fonte: do Autor.

Para que o usuário tenha acesso ao aplicativo, é necessário que ele realize um cadastro utilizando seu email e uma senha com padrões pré definidos. Este cadastro ficará registrado no Firebase e servirá de consulta para o login no aplicativo.

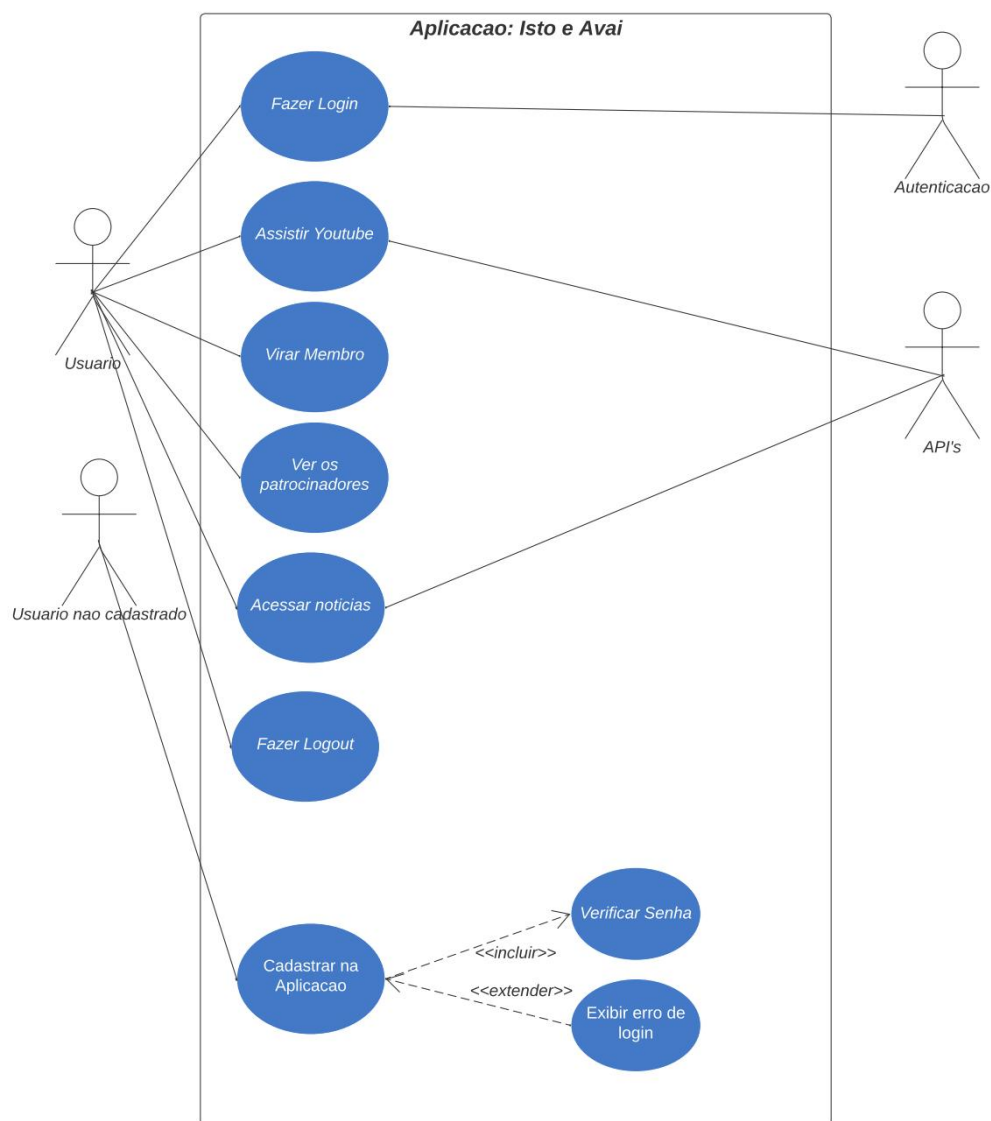
- Login

O usuário necessita fazer login no aplicativo para poder acessar aos conteúdos. Desta maneira o Firebase Authentication, por meio do serviço de autenticação, verifica se o usuário está utilizando um login já existente - caso não esteja, uma mensagem de erro aparecerá na tela.

- Youtube

Visando desempenhar um papel fundamental, proporcionando comodidade e um incentivo a consumir os conteúdos do canal, foi utilizada a API do Youtube para

Figura 20 – Diagrama de casos de uso.



Fonte: do Autor.

criar a funcionalidade de trazer os vídeos do canal, juntamente com o player dos vídeos. Essa abordagem visou principalmente:

- Acesso rápido e direto

Ao exibir uma lista de vídeos diretamente no aplicativo, os usuários têm acesso rápido aos últimos conteúdos publicados no canal, sem a necessidade de abrir o aplicativo do YouTube ou buscar manualmente.

- Comodidade e retenção

A conveniência de visualizar os vídeos no próprio aplicativo aumenta a pro-

habilidade de retenção do usuário, uma vez que eles podem consumir o conteúdo sem interrupções externas.

- Branding e identidade

Ao manter os usuários dentro do aplicativo para assistir aos vídeos, há o fortalecimento da presença de marca e da identidade visual do canal, contribuindo para o reconhecimento e a fidelização.

- Controle de conteúdo

Garantir que os usuários assistam aos vídeos diretamente no aplicativo proporciona maior controle sobre a experiência do usuário, evitando que se percam em outros conteúdos no YouTube.

- Engajamento

Ao oferecer uma lista organizada de vídeos, é possível incentivar o engajamento contínuo, sugerindo conteúdos relacionados, playlists ou mesmo chamadas para ação específicas.

- Experiência do Usuário Unificada

Oferecer uma experiência de usuário contínua, integrando a visualização de vídeos no aplicativo, cria uma experiência mais unificada, contribuindo para uma navegação coesa e intuitiva.

- Notícias

Com o objetivo de trazer informação e praticidade, a página de notícias foi criada utilizando a NEWS API. Esta funcionalidade traz o resultado para as buscas mais recente para a palavra-chave "Avaí". Além disso, esta funcionalidade prezou também por:

- Relevância e Interesse

Ao fornecer notícias relacionadas ao Avaí, o aplicativo atende diretamente aos interesses dos usuários que buscam informações específicas sobre o clube. Isso aumenta a relevância do aplicativo para os torcedores e entusiastas do Avaí.

- Retenção de Usuários

A página de notícias cria um motivo adicional para os usuários permanecerem dentro do aplicativo, uma vez que podem acessar informações atualizadas sobre o Avaí sem precisar recorrer a outras fontes externas.

- Atualizações em Tempo Real

Manter uma seção de notícias proporciona aos usuários acesso a atualizações em tempo real sobre eventos, jogos, transferências de jogadores e outras notícias relevantes ao Avaí, incentivando a interação contínua.

- Futuro

Ao acostumar o usuário com notícias, o canal pode também se tornar um portal de notícias e opiniões sobre o clube.

- Patrocinadores

A ideia da tela é oferecer uma visibilidade exclusiva, promovendo as marcas nela citadas. Também é um protótipo para a criação de um clube de benefícios, onde as marcas poderiam proporcionar descontos a todos que viessem por meio do aplicativo. Neste viés, são alguns pontos que destacam a importância desta tela:

- Visibilidade para os Patrocinadores

Uma tela dedicada aos patrocinadores oferece a eles visibilidade exclusiva, permitindo que sua marca seja destacada em um ambiente controlado e voltado para os usuários do aplicativo.

- Fortalecimento de Parcerias

A criação de um espaço específico para os patrocinadores demonstra o compromisso do aplicativo em promover e fortalecer as parcerias comerciais. Isso pode contribuir para relações comerciais mais sólidas e duradouras.

- Exibição de Conteúdo Personalizado

A tela de patrocinadores possibilita a exibição de conteúdo personalizado, como logotipos, informações sobre os produtos ou serviços oferecidos pelos patrocinadores, promoções especiais, entre outros.

- Geração de Receita

A tela de patrocinadores pode ser uma fonte adicional de receita para o aplicativo, permitindo a inclusão de espaços publicitários ou funcionalidades especiais que promovam as marcas dos patrocinadores.

- Valor Adicional para os Usuários e Promoções

Ao apresentar informações relevantes sobre os patrocinadores, como ofertas exclusivas, descontos ou eventos especiais, a tela agrega valor à experiência do usuário, proporcionando benefícios exclusivos.

Os patrocinadores podem aproveitar a tela exclusiva para apresentar promoções especiais aos usuários do aplicativo, incentivando a interação e o engajamento.

- Membros

A tela dedicada aos membros assinantes, que são pessoas que ajudam a financiar o canal, tem como objetivo trazer clareza sobre este assunto, para que

todos entendam como podem contribuir para o desenvolvimento do canal e também futuramente criar funcionalidades exclusivas para os membros dentro do aplicativo.

5.3.1.5 Integrações

5.3.1.5.1 *Firebase Authentication*

O serviço do Firebase Authentication foi escolhido para o projeto por oferecer uma solução completa para a autenticação de usuários em aplicativos. Ele fornece funcionalidades para o gerenciamento de usuários e também adota práticas robustas de segurança, além de ser uma escolha popular e gratuita.

5.3.1.5.2 *API*

- Youtube

O serviço de API do Youtube é gratuito, e foi utilizado para trazer as informações do canal para dentro do aplicativo. Esta API conta com uma documentação completa feita no Console de Desenvolvedores do Google. Por meio desta API foi possível trazer diversas informações do canal para o aplicativo. As escolhidas foram: nome, ícone, vídeos, número de inscritos e também a reprodução dos vídeos.

- News

O serviço da News API é gratuito e foi utilizado por oferecer uma grande variedade de fonte de notícias. Por meio dela foi possível oferecer uma maneira eficiente de acessar as notícias em tempo real.

5.4 ETAPA 4: TESTES

Para realização dos testes do aplicativo, foi utilizado como base o sistema Android. A proposta para a etapa é estressar ao máximo o aplicativo para entender se existem falhas ou pontos de melhoria. Os casos de testes escolhidos visaram as funcionalidades do aplicativo em seu estado de uso. Os testes resultaram no Quadro 2.

5.4.1 **Caso de Teste 1: Usuário faz cadastro no aplicativo.**

5.4.1.1 Pré-requisito

- Possuir o aplicativo instalado

Quadro 2 – Casos de teste.

ID	Título.
T1	Usuário faz cadastro no aplicativo.
T2	Usuário cadastra senha inválida.
T3	Usuário faz login.
T4	Usuário tenta fazer login com email nao cadastrado.
T5	Usuário tenta fazer login com senha incorreta.
T6	Usuário vira membro do canal.
T7	Usuário assiste vídeo no youtube através do aplicativo.
T8	Usuário acessar notícias através do aplicativo.
T9	Usuário acessa a página de um patrocinador através do aplicativo.

5.4.1.2 Descrição

Este teste planeja verificar se o usuário consegue fazer cadastro no aplicativo com sucesso.

Tabela 1 – Caso de Teste 1.

Passo	Ação	Resultado Esperado
1	Abrir o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no botão "Seja bem-vindo".	É exibida a tela de cadastro.
3	Usuário faz cadastro correto e clica em criar conta.	A tela home é exibida.

Fonte: do Autor.

5.4.1.3 Resultado

Foi possível realizar o cadastro ao executar todos os passos elencados por este caso de teste.

5.4.2 Caso de Teste 2: Usuário cadastra senha inválida.

5.4.2.1 Pré-requisito

- Possuir o aplicativo instalado

5.4.2.2 Descrição

Este teste planeja verificar se o usuário consegue fazer cadastro com senha inválida no aplicativo com sucesso.

5.4.2.3 Resultado

Foi possível realizar teste para o cadastro com a senha inválida ao executar todos os passos elencados por este caso de teste.

Tabela 2 – Caso de Teste 2.

Passo	Ação	Resultado Esperado
1	Abrir o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no botão "Seja bem-vindo".	É exibida a tela de cadastro.
3	Usuário faz cadastro incorreto e clica em criar conta.	O aviso de senha inválida aparece.

Fonte: do Autor.

5.4.3 Caso de Teste 3: Usuário faz login.

5.4.3.1 Pré-requisito

- Possuir o aplicativo instalado
- Possuir cadastro no aplicativo

5.4.3.2 Descrição

Este teste planeja verificar se o usuário consegue fazer login com email e senha válidas no aplicativo com sucesso.

Tabela 3 – Caso de Teste 3.

Passo	Ação	Resultado Esperado
1	Abrir o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no "Fazer Login".	É exibida a tela de login.
3	Usuário preenche corretamente os campos.	É exibida a tela de home

Fonte: do Autor.

5.4.3.3 Resultado

Foi possível realizar teste de login ao executar todos os passos elencados por este caso de teste.

5.4.4 Caso de Teste 4: Usuário tenta fazer login com email não cadastrado.

5.4.4.1 Pré-requisito

- Possuir o aplicativo instalado

5.4.4.2 Descrição

Este teste planeja verificar se o usuário consegue fazer login com email inválido no aplicativo.

Tabela 4 – Caso de Teste 4.

Passo	Ação	Resultado Esperado
1	Abrir o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no "Fazer Login".	É exibida a tela de login.
3	Usuário preenche incorretamente os campos.	O aviso de login incorreto aparece

Fonte: do Autor.

5.4.4.3 Resultado

Foi possível realizar teste de login ao executar todos os passos elencados por este caso de teste.

5.4.5 Caso de Teste 5: Usuário tenta fazer login com senha incorreta.

5.4.5.1 Pré-requisito

- Possuir o aplicativo instalado
- Possuir cadastro no aplicativo

5.4.5.2 Descrição

Este teste planeja verificar se o usuário consegue fazer login com senha inválida no aplicativo.

Tabela 5 – Caso de Teste 5.

Passo	Ação	Resultado Esperado
1	Abrir o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no "Fazer Login".	É exibida a tela de login.
3	Usuário preenche incorretamente os campos.	O aviso de senha incorreta aparece

Fonte: do Autor.

5.4.5.3 Resultado

Foi possível realizar teste de login ao executar todos os passos elencados por este caso de teste.

5.4.6 Caso de Teste 6: Usuário vira membro do canal.

5.4.6.1 Pré-requisito

- Possuir o aplicativo instalado
- Estar logado no aplicativo

5.4.6.2 Descrição

Este teste planeja verificar se o usuário consegue virar membro do canal por meio do aplicativo.

Tabela 6 – Caso de Teste 6.

Passo	Ação	Resultado Esperado
1	Abrir o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no "Fazer Login".	É exibida a tela de login.
3	Usuário preenche corretamente os campos.	É exibida a tela de home
4	Usuário clica no botão "Seja Membro"(Estrela)	O aplicativo redireciona ao Youtube.

Fonte: do Autor.

5.4.6.3 Resultado

Foi possível realizar teste de virar membro ao executar todos os passos elencados por este caso de teste.

5.4.7 Caso de Teste 7: Usuário assiste vídeo no youtube através do aplicativo.

5.4.7.1 Pré-requisito

- Possuir o aplicativo instalado
- Estar logado no aplicativo

5.4.7.2 Descrição

Este teste planeja verificar se o usuário consegue assistir o canal no aplicativo.

Tabela 7 – Caso de Teste 7.

Passo	Ação	Resultado Esperado
1	Abrir o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no "Fazer Login".	É exibida a tela de login.
3	Usuário preenche corretamente os campos.	É exibida a tela de home
4	Usuário clica no botão "Youtube"(Player)	É exibida a tela com os vídeos do canal.
5	Usuário clica em algum vídeo	É exibido o vídeo.

Fonte: do Autor.

5.4.7.3 Resultado

Foi possível realizar teste de assistir um vídeo ao executar todos os passos elencados por este caso de teste.

5.4.8 Caso de Teste 8: Usuário acessa notícias através do aplicativo.

5.4.8.1 Pré-requisito

- Possuir o aplicativo instalado
- Estar logado no aplicativo

5.4.8.2 Descrição

Este teste planeja verificar se o usuário consegue acessar as notícias no aplicativo.

Tabela 8 – Caso de Teste 8.

Passo	Ação	Resultado Esperado
1	Abriu o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no "Fazer Login".	É exibida a tela de login.
3	Usuário preenche corretamente os campos.	É exibida a tela de home
4	Usuário clica no botão "Notícias"(Jornal)	É exibida a tela com as notícias.
5	Usuário clica em alguma notícia	É exibido a notícia.

Fonte: do Autor.

5.4.8.3 Resultado

Foi possível realizar teste de acessar as notícias ao executar todos os passos elencados por este caso de teste.

5.4.9 Caso de Teste 9: Usuário acessa a página de um patrocinador através do aplicativo.

5.4.9.1 Pré-requisito

- Possuir o aplicativo instalado
- Estar logado no aplicativo

5.4.9.2 Descrição

Este teste planeja verificar se o usuário consegue acessar a página dos patrocinadores por meio do aplicativo.

5.4.9.3 Resultado

Foi possível realizar teste acessar a página dos patrocinadores ao executar todos os passos elencados por este caso de teste.

Tabela 9 – Caso de Teste 9.

Passo	Ação	Resultado Esperado
1	Abrir o aplicativo.	É exibida a tela de boas-vindas.
2	Clica no "Fazer Login".	É exibida a tela de login.
3	Usuário preenche corretamente os campos.	É exibida a tela de home
4	Usuário clica no botão "Patrocinadores"(Mercado)	É direcionado a página dos patrocinadores.

Fonte: do Autor.

5.5 ETAPA 5: LANÇAMENTO

O aplicativo tem previsão de lançamento para a primeira metade de Dezembro, quando o canal irá comemorar no seu episódio de número 200.

O lançamento para usuários Android não apresenta empecilhos. Entretanto a versão para iOS depende de testes e aprovação da Apple Store.

5.5.1 Lançamento beta (demo)

O lançamento do aplicativo, como comemoração do 200º episódio do Canal Isto é Avaí, será feito em formato beta (demo), apenas para os membros assinantes do canal.

Essa estratégia permite um ambiente mais controlado para os testes de lançamento, com menos usuários, e também valoriza a assinatura do canal, trazendo um super benefício exclusivo para aqueles que pagam mensalmente uma contribuição para o canal.

5.5.2 Feedback dos usuários

Para recolher feedback dos usuários na etapa de lançamento, além do espaço no grupo exclusivo de membros no Whatsapp também será disponibilizado o link para um formulário de feedback do aplicativo (figura 21).

5.6 ETAPA 6: REVISÃO

Um aplicativo bem-sucedido não se limita apenas a uma boa fase de desenvolvimento, mas também demanda suporte contínuo, manutenção e atualizações regulares para garantir a permanente interação e envolvimento dos usuários.

Dessa forma, após o lançamento e posteriormente de forma regular, será feito o acompanhamento de bugs e problemas técnicos, para que possa ser feita a correção dos mesmos.

Além disso, está planejada a continuação do desenvolvimento de novas funcionalidades para o aplicativo, a fim de incrementar valor ao produto e à empresa.

Figura 21 – Formulário de Feedback.

The figure displays three sequential screenshots of a Google Forms interface for a feedback survey. The survey title is "O Podcast da torcida do Avaí" and the main question is "Conta pra gente o que achou do novo aplicativo do Isto é Avaí?".

- First Screenshot:** Shows the survey title, a "Not shared" status, and a 5-point rating scale question: "Que nota você daria para a experiência que teve no aplicativo até agora?". The scale options are 1, 2, 3, 4, and 5, with 5 selected. Below the scale are two text input fields: "Quer explicar porque deu essa nota?" and "Quer deixar alguma sugestão?". At the bottom, there is a "Next" button and a "Clear form" link.
- Second Screenshot:** Shows the "Obrigado por isso!" (Thank you) section. It includes a "Qual seu nome?" field with a placeholder "<nome>" and a "Qual seu whatsapp?" field with a placeholder "48999999999". At the bottom, there are "Back", "Submit", and "Clear form" buttons.
- Third Screenshot:** Shows the "Obrigado por deixar sua opinião!" (Thank you for your opinion) confirmation screen. It includes a "Submit another response" link and a footer with "Google Forms" branding and a disclaimer: "This content is neither created nor endorsed by Google".

Fonte: do Autor.

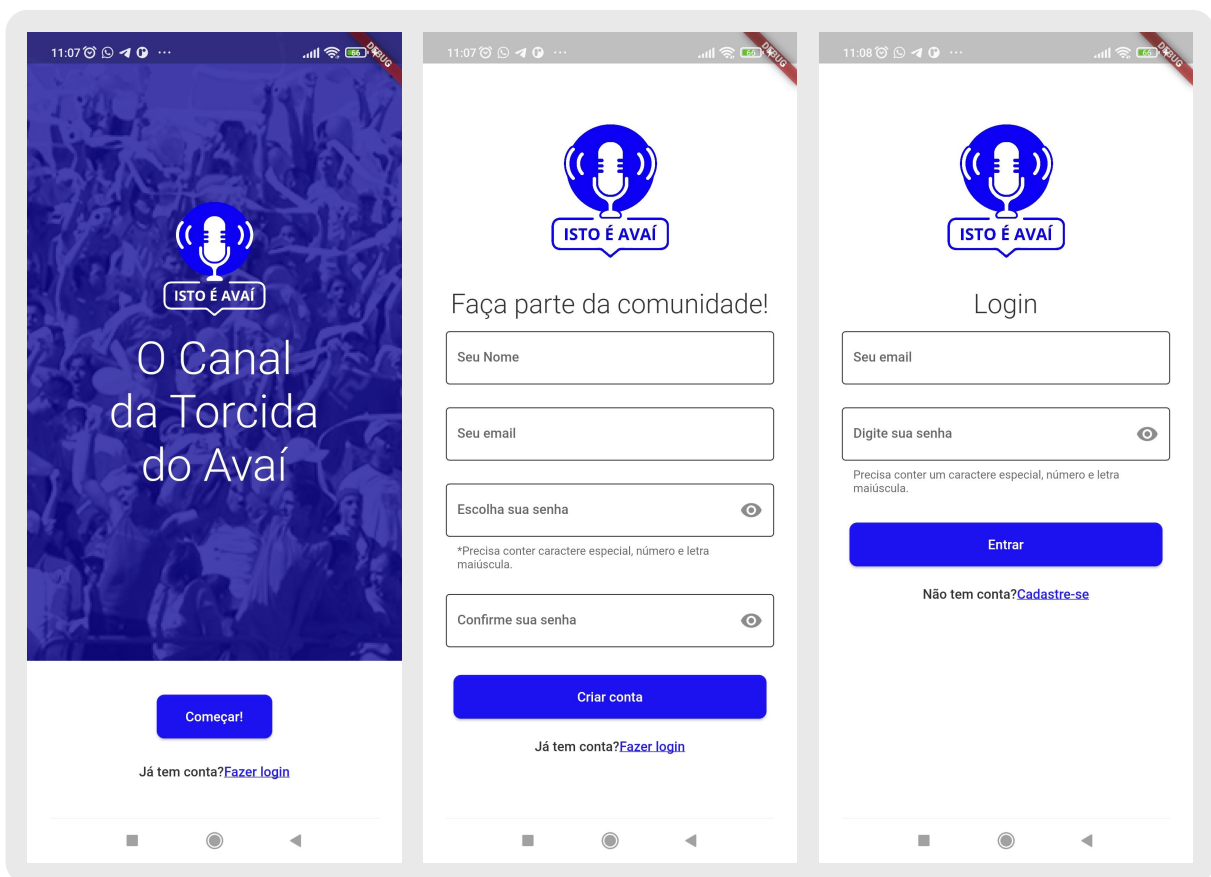
6 ANÁLISE DE RESULTADOS

Este capítulo apresenta os resultados de implementação do projeto. Na seção 6.1, são exibidas as telas elaboradas para o aplicativo. A seção 6.2 abrange os testes das funcionalidades conduzidos para verificar a eficácia do sistema. A seção 6.3 oferece uma análise dos resultados obtidos, destacando os objetivos do projeto e sua efetividade para empresa.

6.1 TELAS DA APLICAÇÃO

Nesta seção são apresentadas as telas da aplicação: resultado final do projeto. A figura 22 mostra as telas iniciais do aplicativo, como a tela de abertura, tela de criação de conta e tela de login.

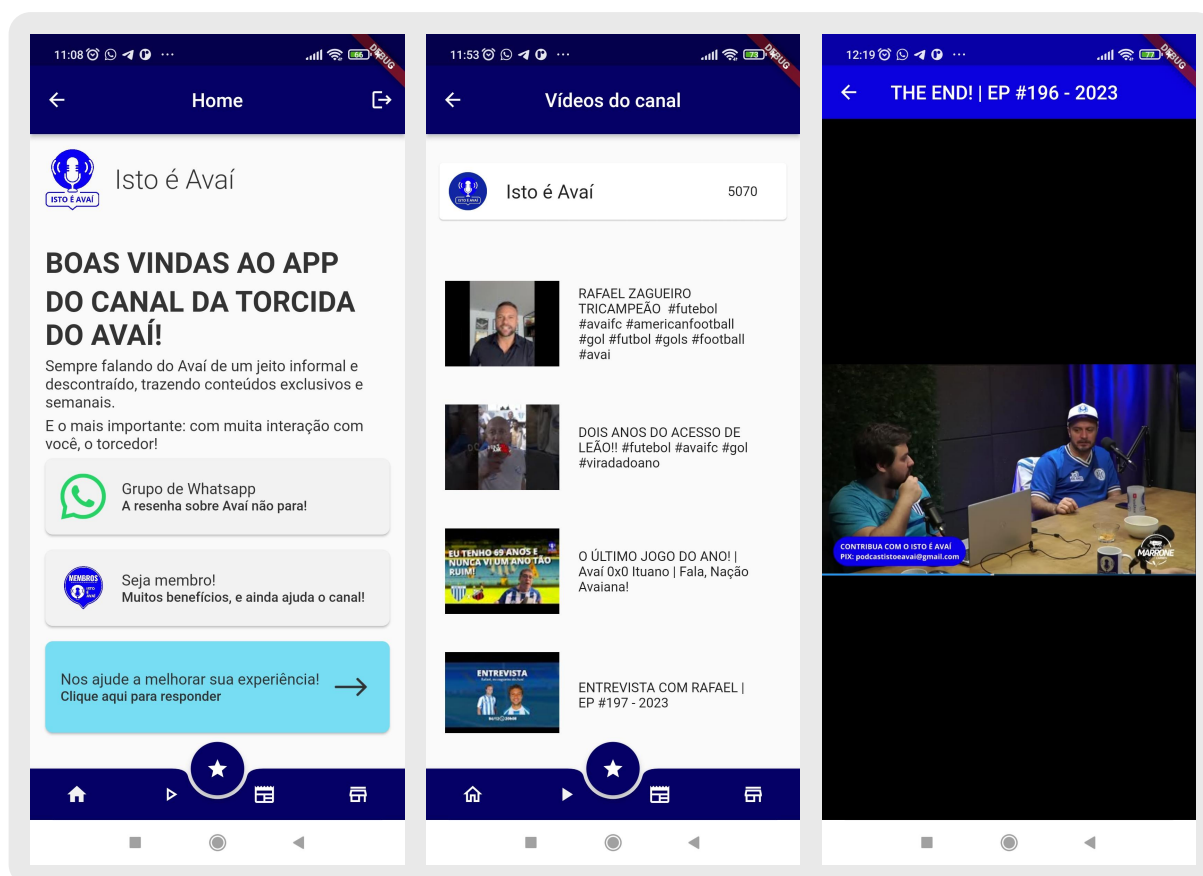
Figura 22 – Telas de início do aplicativo.



Fonte: Do autor.

A figura 23 mostra a tela da Home, a tela com todos os vídeos do canal e também a tela do player de vídeo. Já a figura 24 mostra as telas de Notícias e de Parcerias.

Figura 23 – Telas de Home, Lista de vídeos e Player de vídeos.



Fonte: Do autor.

6.2 AVALIAÇÃO FUNCIONAL

Com base no resultado dos testes (5.4), é possível concluir que o aplicativo apresenta uma interface de usuário funcional, pois conseguiu alcançar os objetivos traçados nos testes.

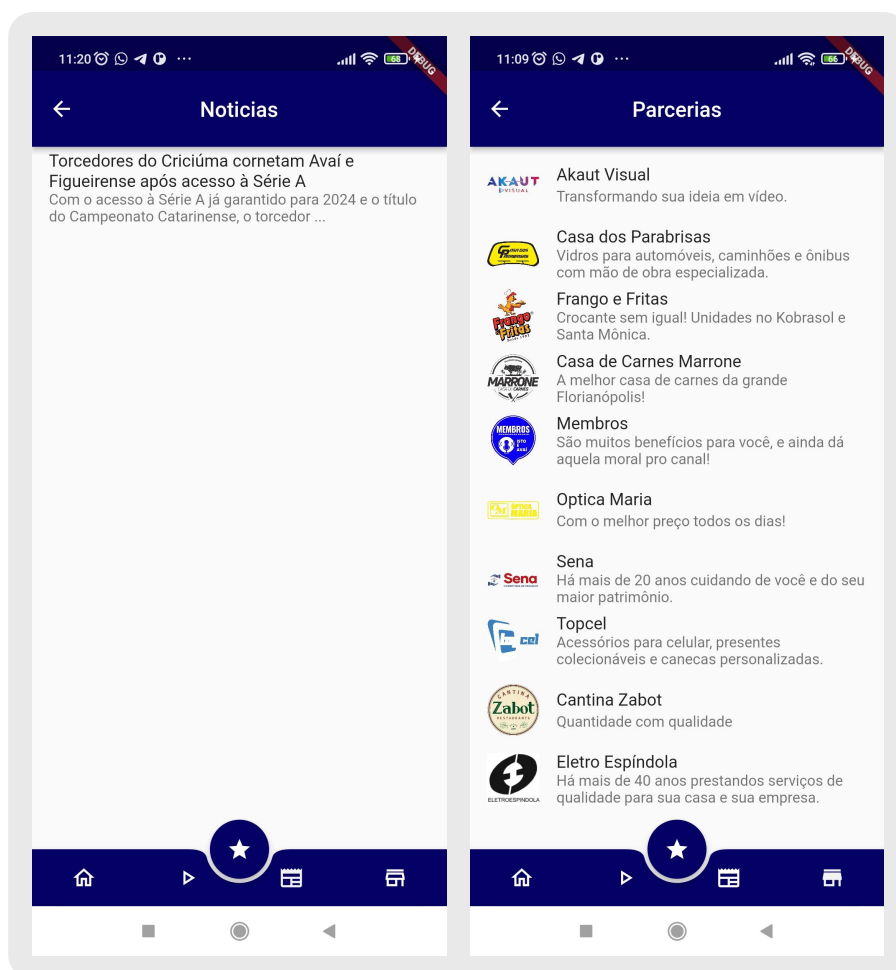
Em relação às funcionalidades que dependem de integrações, estas mostraram-se bastante fluidas e confiáveis. Sobre as integrações:

- *Firestore*: Como levantado na Fundamentação Teórica e aplicado durante o Desenvolvimento, mostrou-se completo para o que se propõe: a autenticação dos usuários. Durante os testes funcionou como o esperado, sem travamentos e bugs.
- *APIs*:

Youtube

Como esperado durante a etapa de Desenvolvimento, mostrou-se um serviço robusto e gratuito, podendo inclusive agregar mais funcionalidade ao aplicativo

Figura 24 – Telas de Notícias e Parcerias.



Fonte: Do autor.

no futuro. Durante os testes funcionou normalmente, sem travamentos ou bugs.

News

Apresentou-se uma opção qualificada e gratuita, com uma vasta base de consulta. Não apresentou travamento nem bugs durante a etapa de testes.

6.3 ANÁLISE DE RESULTADOS

O objetivo deste projeto, conforme descrito no Capítulo 1.2, nas seções 1.2.1 e 1.2.2, estava na criação de um aplicativo PWA para facilitar a relação entre o criador de conteúdo (Canal Isto é Avaí) e seu público. Verifica-se que o objetivo foi cumprido, desenvolvendo um ambiente dedicado a esta comunidade digital, e que permite ao canal ter mais autonomia no contato e interação com seus membros e fãs, sem depender exclusivamente das redes sociais e seus algoritmos em constante mudança.

A expectativa de resultado do projeto para a empresa é enorme, visto que coloca nas mãos da audiência um produto com acesso rápido e direto aos seus conteúdos, que propicia comodidade e retenção, além de fortalecer o branding e identidade do canal.

Os benefícios para os membros também aumentam com o lançamento do produto, pois além do aplicativo em si, o espaço que proporciona e suas funcionalidades, também incentiva os patrocinadores do canal a oferecerem mais benefícios para a audiência, como cupons de desconto. O espaço dedicado à divulgação dos seus patrocinadores também demonstra o compromisso do Canal Isto é Avaí em promover e fortalecer as parcerias comerciais, o que pode contribuir para relações comerciais mais sólidas e duradouras.

Seu desenvolvimento e lançamento inicial se preocupam em testar a aceitação do produto pelo público do canal, mas abre o caminho para a inserção de novas funcionalidades: como uma loja própria e pagamento das mensalidades diretamente pelo aplicativo (sem depender das taxas do Youtube).

7 CONCLUSÃO

O objetivo central deste trabalho foi entender as necessidades, idear soluções, planejar, desenvolver e testar um aplicativo PWA que facilitasse a relação entre a criação de conteúdo do Canal Isto é Avaí e seu público, proporcionando à comunidade do canal um ambiente próprio, livre das regras e taxas das redes sociais.

O desenvolvimento do aplicativo foi realizado utilizando o *framework Flutter*, que permitiu a entrega de uma aplicação para mais de uma plataforma sem a necessidade de desenvolvimentos diferentes para cada uma, por ser uma aplicação híbrida.

A metodologia utilizada (uma união do Design Thinking com Scrum) se mostrou de extrema valia para o projeto. Primeiro pela abordagem experimental e centrada no ser humano, mas também por não se tratar de um método linear, abrindo espaço para alterações durante o projeto. O segundo fator também está presente na metodologia Scrum, e foi de enorme importância durante o projeto, visto que a priorização das funcionalidades precisou ser alterada no decorrer do desenvolvimento.

As funcionalidades implementadas no aplicativo proporcionam uma solução compatível com o escopo do projeto e de grande relevância para a empresa em questão e sua audiência. Com o lançamento do projeto, será possível mensurar o engajamento da audiência com o aplicativo e a partir dessa interação criar, priorizar e desenvolver novas funcionalidades que agreguem ainda mais para o produto. Durante o projeto já foram levantadas ideias como a criação de uma carteira de identificação dos membros assinantes do canal, para ganhar descontos especiais com os parceiros; criação de uma loja própria para venda dos produtos físicos da empresa; um sistema de cobrança de mensalidade direto na plataforma; dar notas para os jogadores após cada partida; push notifications com promoções e lembretes de eventos do canal, entre outras.

A arquitetura utilizada e o código bem documentado, permitem que alterações e incrementos sejam realizados com facilidade no aplicativo.

Em geral, avalia-se que o desenvolvimento deste projeto representa um enorme incremento para a comunidade do Canal Isto é Avaí, sendo mais um diferencial e consequentemente gerando mais confiança e relevância para a empresa perante a audiência, seus patrocinadores e demais mídias do esporte catarinense.

REFERÊNCIAS

BECK, Kent *et al.* Manifesto for Agile Software Development, 2001.

BONINI, Luiz Alberto; SBRAGIA, Roberto. O MODELO DE DESIGN THINKING COMO INDUTOR DA INOVAÇÃO NAS EMPRESAS: UM ESTUDO EMPÍRICO. Revista de Gestão e Projetos-GeP, 2011.

BROWN, Tim. **Design thinking: uma metodologia poderosa para decretar o fim das velhas ideias.** [S.l.]: Elsevier, 2010.

CBF. **Impacto do futebol brasileiro.** [S.l.: s.n.], 2019a. Disponível em: https://conteudo.cbf.com.br/cdn/201912/20191213172843_346.pdf.

CBF. **Impacto do futebol brasileiro.** [S.l.], 2019b. Disponível em: https://conteudo.cbf.com.br/cdn/201912/20191213172843_346.pdf.

CROCKFORD, Douglas. **JavaScript: The Good Parts.** [S.l.]: O'Reilly Media, 2008.

CRUZ, Fabio. **Scrum e Agile em Projetos - Guia Completo.** [S.l.]: BRASPORT, 2015.

DEKKER, Andre; DUMONTIER, Michel; KUBBEN, Pieter. **Fundamentals of clinical data science.** [S.l.]: Saint Philip Street Press, out. 2020.

DENNIS, Allan; WIXOM, Barbara Haley; ROTH, Roberta M. Roberta M. **Análise e Projeto de Sistemas.** [S.l.]: LTC, 2014. v. 5.

DINGSØYR, Torgeir; DYBÅ, Tore; GJERTSEN, Mette; JACOBSEN, Anette Odgaard; MATHISEN, Tor-Erik; NORDFJORD, Jan Ole; RØE, Kjetil; STRAND, Kjetil. Key lessons from tailoring agile methods for large-scale software development. arXiv, 2018.

FIREBASE. [S.l.: s.n.], 2023. Disponível em: <https://firebase.google.com/>.

FLUTTER. **Documentação Flutter.** [S.l.: s.n.], 2023. Disponível em: <https://flutter.dev/>.

FORONES, Jeffrey Marvin. **Authentication and Authorization**. [S.l.: s.n.], 2021.

Disponível em: <https://medium.com/codex/native-vs-hybrid-mobile-appdevelopment-cost-factors-a-breakdown-a97c196287c4>.

FUIOR, Flaviu. Key elements for the success of the most popular Agile methods. **Rev. română inform. și autom.**, ICI Bucharest, v. 29, n. 4, p. 7–16, 2019.

GROUP, Nilsen Norman. **Design Thinking 101**. [S.l.: s.n.], 2016. Disponível em: <https://www.nngroup.com/articles/design-thinking/>.

HRON, Michal; OBWEGESER, Nikolaus. Why and how is Scrum being adapted in practice: A systematic review. **J. Syst. Softw.**, 2022.

IEEE. [S.l.: s.n.]. Disponível em: https://en.wikipedia.org/wiki/IEEE_1471.

IVANOVICH, Bondarenko Vitaliy; VLADIMIROVICH, Bilousov Vyacheslav; VICTOROVICH, Nedopekin Fedor; VIKTOROVICH, Bodriaha Viktor; VITALIEVNA, Antropova Larisa. Using MVC pattern in the software development to simulate production of high cylindrical steel ingots. **Journal of Crystal Growth**, v. 526, 2019. Disponível em:

<https://www.sciencedirect.com/science/article/pii/S0022024819304555>.

JACOBSON, Ivar; STIMSON, Roly. Tear Down the Method Prisons! Set Free the Practices! **Association for Computing Machinery**, 2018.

KEARNEY, Adam. **The MVC Schema**. [S.l.: s.n.], 2019. Disponível em:

<https://medium.com/@adamkearney124/model-view-controller-f2bdbf1ee999>.

LELER, Wm. **Why Flutter Uses Dart**. [S.l.: s.n.], 2017. Disponível em:

<https://hackernoon.com/why-flutter-uses-dart-dd635a054ebf..>

MALL, Rajib. **Fundamentals of Software Engineering**. [S.l.]: PHI Learning Privated Limited, 2014. v. 4.

PAULA FILHO, Wilson de Padua. **Engenharia de Software Fundamentos Métodos Padrões**. [S.l.]: LTC, 2009. v. 2.

SBOK™, Guia. **A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™GUIDE)**. [S.l.]: SCRUMstudy™, 2015. v. 3.

SCHWABER K.; SUTHERLAND, J. **Um guia definitivo para o Scrum: As regras do jogo**. [S.l.: s.n.], 2017. Disponível em:

<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwiNm9XXsuqCAxWBqJUCHdldD4YQFnoECAkQAQ&url=https%3A%2F%2Fscrumguides.org%2Fdocs%2Fscrumguide%2Fv2020%2F2020-Scrum-Guide-PortugueseBR-3.0.pdf&usq=A0vVaw3u5Ha9tgHMPTjyKaZLbUzo&opi=89978449>.

SIMON, Hebert. **The Sciences of the Artificial**. [S.l.: s.n.], 1969. Disponível em:

http://pt.wikipedia.org/wiki/Design_thinking#cite_ref2simon_1969_020.

SINGHTO, Wantana; DENWATTANA, Nuansri. An experience in blending the traditional and Agile methodologies to assist in a small software development project. **IEEE**, 2016.

STICKDORN, Marc; SCHNEIDER, Jakob. **Isto é design thinking de serviços: Fundamentos, ferramentas, casos**. [S.l.]: Bookman, 2014.

SUTHERLAND, Jeff; SUTHERLAND, J.J. **Scrum: A arte de fazer o dobro do trabalho na metade do tempo**. [S.l.]: Sextante, 2019.