

Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Engenharia de Software

**Orienta UnB: Proposta de Sistema de
Recomendação *mobile* para Promover o
Processo de Matrículas em Trabalho de
Conclusão de Curso**

Autor: Denys Rógeres Leles dos Santos
Leonardo dos Santos Silva Barreiros

Orientador: Prof. Dr. John Lenon Cardoso Gardenghi

Brasília, DF

2023



Denys Rógeres Leles dos Santos
Leonardo dos Santos Silva Barreiros

**Orienta UnB: Proposta de Sistema de Recomendação
mobile para Promover o Processo de Matrículas em
Trabalho de Conclusão de Curso**

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Dr. John Lenon Cardoso Gardenghi

Brasília, DF

2023

Denys Rógeres Leles dos Santos

Leonardo dos Santos Silva Barreiros

Orienta UnB: Proposta de Sistema de Recomendação *mobile* para Promover o Processo de Matrículas em Trabalho de Conclusão de Curso/ Denys Rógeres Leles dos Santos

Leonardo dos Santos Silva Barreiros. – Brasília, DF, 2023-

115 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. John Lenon Cardoso Gardenghi

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB

Faculdade UnB Gama - FGA , 2023.

1. Palavra-chave01. 2. Palavra-chave02. I. Prof. Dr. John Lenon Cardoso Gardenghi. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Orienta UnB: Proposta de Sistema de Recomendação *mobile* para Promover o Processo de Matrículas em Trabalho de Conclusão de Curso

CDU 02:141:005.6

Denys Rógeres Leles dos Santos
Leonardo dos Santos Silva Barreiros

**Orienta UnB: Proposta de Sistema de Recomendação
mobile para Promover o Processo de Matrículas em
Trabalho de Conclusão de Curso**

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Trabalho aprovado. Brasília, DF, 20 de Dezembro de 2023:

**Prof. Dr. John Lenon Cardoso
Gardenghi**
Orientador

Prof. Dra. Milene Serrano
Convidado 1

Prof. Dr. Renato Coral Sampaio
Convidado 2

Brasília, DF
2023

Agradecimentos

Gostaríamos de expressar nossos agradecimentos iniciais a Deus, pois é Ele quem nos concede a força e a coragem para enfrentar cada desafio. Através da fé em Deus, superamos obstáculos e momentos difíceis.

Agradecemos às nossas famílias por todo o apoio, a motivação e o incentivo que nos foram dedicados ao longo desta caminhada. Acreditamos que a família tenha sido um pilar fundamental nessa jornada, e somos gratos pelo seu apoio constante, que nos fortaleceu para alcançar nossos sonhos.

Aos amigos, este proporcionaram a alegria nos momentos necessários, principalmente pelo apoio em decorrência da jornada até o momento.

Por fim, agradecemos aos docentes da FGA pelo ensinamento, pelo conhecimento transmitido, que foram fundamentais em todo o processo de amadurecimento para a construção de nossas carreiras. Em especial, agradecemos ao nosso orientador, Prof. Dr. John Lenon Cardoso Gardenghi, pelas horas dedicadas e pela oportunidade de realizarmos este trabalho.

*"Ainda que eu ande pelo vale da sombra da morte,
não temerei mal algum,
pois tu estás comigo."
(Bíblia Sagrada, Salmos 23, 4)*

Resumo

Este trabalho é uma solução que funciona como uma alternativa para mitigar as principais causas dos déficits de orientação nos Trabalhos de Conclusão de Curso (TCC) no curso de Engenharia de *Software* da Universidade de Brasília, especificamente na Faculdade do Gama. Com o intuito de aprimorar a gestão, integração e engajamento entre docentes e discentes, foi concebida a proposta da criação de um aplicativo para *smartphones*, denominado Orienta UnB. Este aplicativo, orientado pelos princípios de *Smart Campus*, é uma iniciativa para superar as adversidades relacionadas as grandes quantidades de alunos por docente. Dessa forma, o aplicativo busca distribuir as elevadas demandas do cenário atual, simplificando a colaboração em projetos acadêmicos, otimizando a quantidade de orientações, ao unir alunos e professores com os mesmos interesses. Além disso, fornece uma experiência personalizada por meio de um sistema de recomendação com filtragem baseada em conteúdo, que está associado às áreas de interesse dos alunos e professores. Por fim, esta é uma contribuição para a Universidade de Brasília, visto que os benefícios estão direcionados à comunidade acadêmica e podem auxiliar durante os períodos semestrais, além de fornecer conhecimento para contribuições no desenvolvimento de *softwares* por terceiros.

Palavras-chaves: Engenharia de *Software*. Campus Inteligente. Sistema de Recomendação. Filragem Baseada em Conteúdo. Desenvolvimento de *Software*. Aplicativo *Mobile*. Desenvolvimento *Mobile*.

Abstract

This work aims to propose a solution that serves as an alternative to mitigate the main causes of guidance deficits in the Final Course Projects (FCP) in the Software Engineering course at the UnB, specifically at the FGA. With the intention of enhancing management, integration, and engagement between professors and students, the proposal for the creation of a smartphone application, named Orienta UnB, has been conceived. This application, guided by the principles of Smart Campus, is an initiative to overcome adversities related to the large numbers of students per professors member. Thus, the application seeks to distribute the high demands of the current scenario, simplifying collaboration in academic projects and optimizing the number of guidance sessions by bringing together students and professors with similar interests. In addition, it provides a personalized experience, incorporating a recommendation system with content-based filtering associated with the areas of interest of both students and professors. By the conclusion of this work, the aim is to contribute to the University of Brasília, benefiting the academic community with a tool that can assist during semester periods, as well as provide insights for contributions in software development by third parties.

Key-words: Software Engineering. Smart Campus. Recommendation System. Content-Based Filtering. Software Development. Mobile Application. Mobile Development.

Lista de ilustrações

Figura 1 – <i>Conceptual framework of smart campus</i>	29
Figura 2 – Impacto versus Complexidade	30
Figura 3 – Fluxo de atividades TCC 1	47
Figura 4 – Fluxo de atividades TCC 2	48
Figura 5 – Fluxo de desenvolvimento	50
Figura 6 – Cronograma TCC 1	50
Figura 7 – Cronograma TCC 2	51
Figura 8 – Diagrama de Caso de Uso	58
Figura 9 – Arquitetura geral Orienta UnB	59
Figura 10 – Arquitetura interna Orienta UnB	60
Figura 11 – Modelo esquemático do banco de dados	61
Figura 12 – <i>Match</i> entre participantes	63
Figura 13 – Lista de notificações	64
Figura 14 – Solicitar orientação	65
Figura 15 – Confirmar orientação	66
Figura 16 – Tela de recomendação	67
Figura 17 – Exemplo de usuário recomendado	68
Figura 18 – Exemplo de usuário pouco recomendado	68
Figura 19 – JWT gerado pelo <i>Login</i>	69
Figura 20 – Estrutura de história de usuário	71
Figura 21 – Gráfico ilustrativo do resultado de respostas da 1 ^o seção.	76
Figura 22 – Gráfico ilustrativo do resultado da pergunta 1 da 2 ^o seção.	77
Figura 23 – Gráfico ilustrativo do resultado da pergunta 2 da 2 ^o seção.	77
Figura 24 – Comentários sobre a pergunta 3 da 2 ^o seção.	77
Figura 25 – Gráfico ilustrativo do resultado da pergunta 4 da 2 ^o seção.	78
Figura 26 – Comentários sobre a pergunta 5 da 2 ^o seção.	78
Figura 27 – Gráfico ilustrativo do resultado da pergunta 6 da 2 ^o seção.	78
Figura 28 – Gráfico ilustrativo do resultado da pergunta 1 da 3 ^o seção.	79
Figura 29 – Gráfico ilustrativo do resultado da pergunta 1 da 3 ^o seção.	79
Figura 30 – Comentários sobre a pergunta 3 da 3 ^o seção.	80
Figura 31 – Gráfico ilustrativo do resultado da pergunta 4 da 3 ^o seção.	80
Figura 32 – Gráfico ilustrativo do resultado da pergunta 5 da 3 ^o seção.	81
Figura 33 – Gráfico ilustrativo do resultado da pergunta 6 da 3 ^o seção.	81
Figura 34 – Comentários sobre a pergunta 7 da 3 ^o seção.	82
Figura 35 – Gráfico ilustrativo do resultado da pergunta 8 da 3 ^o seção.	82
Figura 36 – Gráfico ilustrativo do resultado da pergunta 8 da 3 ^o seção.	83

Figura 37 – Gráfico ilustrativo do resultado da pergunta 1.	83
Figura 38 – Gráfico ilustrativo do resultado da pergunta 2.	84
Figura 39 – Gráfico ilustrativo do resultado da pergunta 3.	84
Figura 40 – Gráfico ilustrativo do resultado da pergunta 4.	85
Figura 41 – Gráfico ilustrativo do resultado da pergunta 5.	85
Figura 42 – Gráfico ilustrativo do resultado da pergunta 6.	85
Figura 43 – Gráfico ilustrativo do resultado da pergunta 7.	86
Figura 44 – Imagens das telas principais da aplicação.	87
Figura 45 – Formulário para aluno - 1º seção	97
Figura 46 – Formulário para aluno - 2º seção - Parte 1: Aluno com orientador . . .	98
Figura 47 – Formulário para aluno - 2º seção - Parte 2: Aluno com orientador . . .	99
Figura 48 – Formulário para aluno - 2º seção - Parte 1: Aluno sem orientador . . .	100
Figura 49 – Formulário para aluno - 2º seção - Parte 2: Aluno sem orientador . . .	101
Figura 50 – Formulário para professor - Parte 1	102
Figura 51 – Formulário para professor - Parte 2	103

Lista de quadros

Quadro 1 – Resumo das Tecnologias	43
Quadro 2 – Fase I: Planejamento preliminar	54
Quadro 3 – Fase II: Identificação das áreas	54
Quadro 4 – Fase III: Aquisição de dados	55
Quadro 5 – Fase IV: Análise dos dados	55
Quadro 6 – Fase V: Categorização dos problemas	55
Quadro 7 – Fase VI: Definição das estratégias	55
Quadro 8 – <i>Backlog do Produto: parte 1</i>	72
Quadro 9 – <i>Backlog do Produto: parte 2</i>	73
Quadro 10 – Atividades da primeira etapa do TCC	90
Quadro 11 – Atividades da segunda etapa do TCC	90

Lista de abreviaturas e siglas

API	<i>Application Programming Interface</i>
DER	Diagrama Entidade Relacionamento
FGA	Faculdade do Gama
HTTP	<i>Hypertext Transfer Protocol</i>
IDF	<i>Inverse Document Frequency</i>
IoT	<i>Internet of Things</i>
MER	Modelo Entidade Relacionamento
MVP	<i>Minimum Viable Product</i>
PIBIC	Programa Institucional de Bolsas de Iniciação Científica
PIBID	Programa Institucional de Bolsas de Iniciação à Docência
PIBITI	Programa Institucional de Bolsas de Iniciação em Desenvolvimento Tecnológico e Inovação
TCC	Trabalho de Conclusão de Curso
TF	<i>Term Frequency</i>
UnB	Universidade de Brasília
XP	<i>Extreme Programming</i>

Sumário

1	INTRODUÇÃO	23
1.1	Contexto	23
1.2	Justificativa	24
1.3	Questão de Pesquisa	25
1.4	Objetivos	25
1.4.1	Objetivo Geral	25
1.4.2	Objetivo Específico	25
1.5	Organização da Monografia	26
2	REFERENCIAL TEÓRICO	27
2.1	<i>Smart Campus</i>	27
2.1.1	Concepção de um <i>framework</i> de <i>smart campus</i>	28
2.1.2	Aplicabilidade viabilizando o presente trabalho	31
2.2	Sistema de Recomendação	31
2.2.1	Definição	31
2.2.2	Técnicas de Recomendação	32
2.2.3	Filtragem Baseada em Conteúdo	32
2.2.4	Similaridade	33
2.2.5	Vantagens e Limitações	34
2.2.6	Filtragem Colaborativa	35
2.2.7	Algoritmos Baseados em Memória	35
2.2.7.1	Baseado em Usuário	35
2.2.7.2	Baseado em Item	35
2.2.8	Algoritmos Baseados em Modelo	36
2.2.9	Vantagens e Limitações	36
2.2.10	Filtragem Híbrida	37
2.3	Resumo do Capítulo	38
3	SUORTE TECNOLÓGICO	39
3.1	React Native	39
3.2	Nest.js	39
3.3	MongoDB	40
3.4	Git e Github	40
3.5	Mural	41
3.6	Figma	41
3.7	Google Forms	42

3.8	Expo	42
3.9	Expo Go	42
3.10	Amazon EC2	42
3.11	Resumo do Capítulo	42
4	METODOLOGIA	45
4.1	Classificação da Pesquisa	45
4.1.1	Abordagem	45
4.1.2	Natureza	45
4.1.3	Objetivos	45
4.1.4	Procedimentos	46
4.2	Fluxo de Atividades	46
4.3	Metodologia de Desenvolvimento	48
4.4	Metodologia de Análise de Resultados	49
4.5	Cronograma	50
4.6	Resumo do Capítulo	51
5	ESTUDO ORIENTADO AO DESENVOLVIMENTO DO APLICATIVO ORIENTA UNB	53
5.1	Contexto	53
5.2	Visão Geral do aplicativo Orienta UnB	54
5.2.1	Objetivo de aplicação	55
5.3	Prova de Conceito Preliminar	55
5.3.1	Aplicativo Orienta UnB	56
5.3.2	Desenvolvimento de Software	56
5.3.2.1	<i>Lean Inception</i>	56
5.3.2.2	Regras de negócio	57
5.3.2.3	Diagrama de caso de uso	57
5.3.3	Arquitetura de Software	58
5.3.4	Modelo de dados	60
5.3.5	Protótipo	61
5.3.5.1	Cadastro de usuário	62
5.3.5.2	Demonstração de interesse mútuo	62
5.3.5.3	Lista de notificações	64
5.3.5.4	Solicitar orientação	65
5.3.5.5	Confirmar solicitação	66
5.3.5.6	Sistema de recomendação	67
5.3.5.7	Algoritmo de comparação	69
5.4	Detalhamentos dos requisitos	71
5.5	Resumo do Capítulo	73

6	ANALISE DE RESULTADOS	75
6.1	Pesquisa-Ação	75
6.2	Etapa Exploratória	75
6.2.1	Questionário Aluno	76
6.2.1.1	Estado Atual de Orientação	76
6.2.1.2	Procurando Orientação	76
6.2.1.3	Orientação Definida	79
6.2.2	Questionário Professor	83
6.3	Etapa Planejamento	86
6.4	Etapa Ação	87
6.5	Etapa Avaliação	88
6.6	Resumo do Capítulo	88
7	CONCLUSAO	89
7.1	Contexto Geral	89
7.2	Status Geral do Trabalho	89
7.3	Objetivos e Questões de Pesquisa	91
7.3.1	Objetivos	91
7.3.2	Questões de Pesquisa	91
7.4	Trabalhos Futuros	92
	REFERÊNCIAS	93
	APÊNDICES	95
	APÊNDICE A – PESQUISA DE CAMPO	97
	APÊNDICE B – PROTÓTIPO DE ALTA FIDELIDADE	105
	APÊNDICE C – POLÍTICA PRIVACIDADE	107
	ANEXOS	109
	ANEXO A – CANVAS MVP	111
	ANEXO B – SEQUENCIADOR	112
	ANEXO C – REGRAS	113
	ANEXO D – CADASTRO DO PERFIL PROFESSOR: PARTE 1	114

ANEXO E – CADASTRO DO PERFIL PROFESSOR: PARTE 2 . . 115

1 Introdução

Neste capítulo são apresentados conceitos introdutórios para o embasamento deste trabalho, realizando a contextualização das abordagens desta monografia. Buscando evidenciar e ressaltar a importância da aplicação de métodos modernos para o desenvolvimento e solução para a problemática em estudo. Em seguida, é apresentada a justificativa para este trabalho, seguido pela questão de pesquisa que norteia a motivação para realização desta pesquisa, além de pontuar os objetivos a serem alcançados. Por fim, uma breve descrição da organização do presente trabalho.

1.1 Contexto

Os avanços tecnológicos proporcionaram à humanidade grandes benefícios para o desenvolvimento humano, bem como exigências destinadas às perspectivas sobre a educação. Ao longo do tempo, novas tecnologias vêm revolucionando os processos educacionais, ao produzir novos métodos para promover aspectos que impulsionem os processos de ensino e aprendizagem (SCHENATZ; CUNHA; KUGLER, 2019).

A apresentação de novos métodos educacionais são direcionados pelos conceitos de *Smart Campus*. Trata-se de uma evolução da indústria da educação baseada na internet das coisas e nos conceitos de *Smart City*. Portanto, Schenatz, Cunha e Kugler (2019) afirmam que, por meio das atividades desenvolvidas, pesquisas e inovações tecnológicas são possíveis de gerar potencial para criar mudanças nos aspectos relacionados à vida acadêmica, de modo a promover o desenvolvimento do processo de ensino-aprendizagem e também a qualidade dos serviços oferecidos pelas instituições, tanto para os funcionários quanto para os discentes, propiciando experiências mais integradoras nos câmpus.

Tendo em vista a qualidade de experiência nos câmpus, Schenatz, Cunha e Kugler (2019) afirmam que um dos fatores impactantes é a evasão por parte dos estudantes, que pode ser considerada um dos maiores problemas enfrentados, e que implica o desperdício dos recursos aplicados, que é contra os interesses das instituições de ensino, seja pública ou privada, e também reduzindo as oportunidades para os alunos ingressantes. Portanto, investir em novas fontes, sejam organizacionais, para o gerenciamento e a administração do câmpus, sejam tecnológicas, para disponibilizar recursos que auxiliem no meio acadêmico, é uma maneira de combater esse problema.

Uma das etapas que contempla todo o processo de aprendizagem do estudante durante o curso é a elaboração do Trabalho de Conclusão de Curso (TCC). Segundo Moraes *et al.* (2021), a capacitação para realizar atividades da área referentes à graduação é

garantida ao profissional por meio da conclusão do TCC. No decurso, o estudante desenvolve habilidades de pesquisa, análise crítica e escrita acadêmica, a fim de aprofundar-se na área de interesse. Nessa etapa, o papel do orientador é de fundamental importância para guiar o estudante ao longo do processo de elaboração.

De acordo com [Guimarães e Sobrinho \(2020\)](#), existem aspectos a serem enfrentados durante a realização do TCC. Em 2004, foi realizado um estudo em uma Universidade privada que aponta fatores que facilitam e dificultam o processo de elaboração do TCC. [Guimarães e Sobrinho \(2020\)](#) evidenciam que, independentemente do curso de graduação, há um rol de fatores – psicológicos, materiais, sociológicos, cognitivos, relacionais – que apresentam um panorama, seja facilitador ou limitador, que produz apreensões, expectativas, hesitações, perspectivas e sentimentos em decorrência do desenvolvimento do trabalho. O estudo ressalta que, dentre as problemáticas enfrentadas pelos estudantes, destacam-se o tempo, o custo e a procura por um orientador.

Tendo em vista esse contexto, e enfatizando o último aspecto, os estudantes de Engenharia de Software da Universidade de Brasília, Campus Gama, têm percebido a mesma realidade ao buscar por orientadores para a elaboração de seus Trabalhos de Conclusão de Curso. Portanto, para o presente trabalho faz-se necessário evidenciar dois aspectos: (i) a vulnerabilidade dos discentes da Universidade de Brasília (UnB) do campus da Faculdade do Gama (FGA), para o curso de Engenharia de Software, uma vez que há o risco de descontinuidade no processo institucional e perda de produtividade e efetividade devido à alta demanda de alunos que desejam prestar a disciplina de Trabalho de Conclusão de Curso, e (ii) está relacionado ao primeiro aspecto, trata-se da má distribuição de alunos por professor para cursar esta disciplina, no curso de Engenharia de Software, devido a fatores que estão relacionados à falta de comunicação devida, alinhamento de interesses e visibilidade de disponibilidade dos docentes.

1.2 Justificativa

De forma geral, a fundamentação para o presente trabalho baseia-se em dois fatores principais: o significativo aumento de estudantes no curso de Engenharia de Software na FGA e a ausência de um corpo docente capaz de atender à grande demanda desses alunos. Desse modo, essas problemáticas evidenciam a necessidade de uma aplicação que auxilie os estudantes na busca de um orientador para o TCC, com o intuito de reduzir o número de alunos que atrasam a conclusão do curso por falta de orientador.

Nesse contexto, observa-se um expressivo aumento do número de estudantes matriculados no curso de Engenharia de Software da FGA, motivados pela crescente demanda do mercado de trabalho por profissionais qualificados na área da tecnologia. Esse crescimento significativo gera uma demanda crescente por orientação acadêmica, que não tem

sido adequadamente suprida devido à insuficiência de docentes disponíveis para orientação aos estudantes que atendem aos requisitos necessários para a realização do TCC.

Considerando o exposto, é incontestável a urgente demanda por uma intervenção que aborde estratégias para a alocação de orientadores e orientandos no contexto da superlotação de estudantes nos cursos de Engenharia de Software da FGA. Uma abordagem pertinente seria a criação de uma aplicação que facilite a identificação e junção de alunos e professores com base em seus interesses de pesquisa, para possibilitar a formação de duplas de estudantes com interesses temáticos afins. Essa abordagem colaborativa poderia contribuir para a otimização do processo de TCC, promovendo uma distribuição mais eficiente de orientadores, e incentivando a cooperação e sinergia entre estudantes, professores e temas de pesquisa para o TCC.

Em resumo, a justificativa para a criação de uma aplicação que auxilie estudantes de Engenharia de Software a encontrarem orientadores é respaldada pelo crescimento do número de alunos no curso, pela escassez de professores disponíveis para orientação e pelas consequências negativas que esses fatores podem ter na formação dos estudantes.

1.3 Questão de Pesquisa

Toda tomada de decisão para o desenvolvimento deste trabalho será norteadada pelo seguinte questionamento: *É possível melhorar o alinhamento de expectativas e interesses, proporcionando visibilidade e disponibilidade entre professor-orientador e concludente?*

1.4 Objetivos

Tendo em vista a questão de pesquisa na seção 1.3, foram estabelecidos objetivos caracterizados em termos gerais e específicos, visando responder ao questionamento.

1.4.1 Objetivo Geral

O objetivo geral do presente trabalho foi desenvolver um aplicativo móvel que utilize sistema de recomendação para estabelecer conexões entre alunos e professores que compartilham interesses comuns, atendendo à necessidade dos alunos de Engenharia de Software da Universidade de Brasília de encontrar um professor orientador para cursar a disciplina de Trabalho de Conclusão de Curso.

1.4.2 Objetivo Específico

Para que fosse possível atingir o objetivo geral, foi importante cumprir com os objetivos específicos a seguir:

- Elaboração de um referencial teórico que apresente conceitos sobre a utilização de aplicativos móveis como ferramenta para auxílio universitário e sistemas de recomendação;
- Documentação das dificuldades apresentadas pelos alunos na busca por orientadores para seus Trabalhos de Conclusão de Curso (TCC);
- Desenvolvimento de um *backlog* do produto que contemple as possíveis funcionalidades;
- Prototipação do aplicativo móvel;
- Implementação do aplicativo com base no contexto, e
- Análise e documentação do processo de pesquisa.

1.5 Organização da Monografia

Esta monografia segue a seguinte organização:

- Capítulo 2 - Referencial Teórico: O objetivo deste capítulo é fornecer os principais conceitos que compõem a base de fundamentos do presente trabalho, em especial os conceitos relacionados ao Sistema de Recomendação e ao *Smart Campus*;
- Capítulo 3 - Suporte Tecnológico: Apresentar o contexto dos recursos utilizados, os quais possibilitam o desenvolvimento do projeto;
- Capítulo 4 - Metodologia: Definição da metodologia adotada como norteadora do presente trabalho, ao expor os detalhes que definem a classificação da pesquisa e os moldes que caracterizam as metodologias de desenvolvimento e análise de resultados;
- Capítulo 5 - Estudo Orientado ao Desenvolvimento do Aplicativo Orienta UnB: Apresentação detalhada do estudo deste trabalho, englobando contexto, desenvolvimento e visão do aplicativo móvel para promover o processo de matrículas em Trabalho de Conclusão de Curso, bem como as provas de conceito e as funcionalidades implementadas;
- Capítulo 6 - Análise de Resultados: Disposição e análise dos principais resultados obtidos até o momento, e
- Capítulo 7 - Conclusão: Finalização do trabalho, retomando o contexto, principais resultados obtidos, questões de pesquisa e objetos, além de sugestões para pesquisas futuras.

2 Referencial Teórico

Neste capítulo, serão apresentados os principais conceitos e referências para o embasamento deste trabalho, tendo em vista que o foco para este projeto é realizar contribuições na área de desenvolvimento de aplicativos móveis para a Universidade de Brasília.

Inicialmente, discorre-se acerca dos conceitos de *Smart Campus* e os aspectos relacionados aos benefícios de sua utilização. Além disso, considerando-se que o trabalho é um projeto de um aplicativo destinado a professores e alunos que possuem interesse em comum, faz-se necessário abordar conceitos que envolvam recomendação e similaridade. Por fim, um breve resumo dos principais aspectos deste capítulo.

2.1 *Smart Campus*

Smart Campus é um conceito que se refere a um campus universitário que utiliza tecnologias avançadas para melhorar a eficiência, a segurança e a experiência dos estudantes, professores e funcionários. O objetivo principal é criar um ambiente de aprendizagem inovador que integre tecnologia, ensino e pesquisa para atender às necessidades dos alunos e da comunidade acadêmica em geral (SCHENATZ; CUNHA; KUGLER, 2019).

Segundo BANDEIRA e NETO (2022), gestores, professores e alunos estão enfrentando um momento de transição e de desafios tecnológicos. Esta afirmação dá-se em virtude de estudos que têm mostrado que o uso de *smartphones* tem aumentado nos câmpus universitários. Desta forma, é factível o investimento em plataformas e meios dos quais podem impulsionar a qualidade de ensino. Entretanto, cria-se a necessidade de adaptação. Nesse contexto, o conceito de *smart campus* surge a partir deste princípio, de integrar as necessidades de uma organização em plataformas que possam atendê-las.

O conceito de *smart campus* é baseado no de *smart city*, sendo definido como um ambiente que se assemelha a uma cidade que compartilha problemas, desafios e oportunidades de uma cidade convencional em sua grande maioria. Esse conceito que surgiu dentro da *Internet of Things* (IoT), sendo uma entidade que utiliza tecnologia e infraestrutura para incrementar em seus processos, seja educacional, seja para habitação do campus. Portanto, é notável que as áreas como administração e ciências da informação podem contribuir bastante, com uma abordagem gerencial e estratégica para o benefício de um campus, além de evidenciar que as abordagens de um campus inteligente é mais eficiente e superior que as abordagens tradicionais (BANDEIRA; NETO, 2022).

De acordo com Neves *et al.* (2017), são classificadas em três segmentos as iniciativas para campus inteligente: (i) metodologias e *frameworks*; (ii) aplicações desenvolvidas em

uma área definida; e (iii) pesquisas (*surveys*) de tecnologias específicas.

Em 2014, a Faculdade de Engenharia de Sorocaba (FACENS) desenvolveu o programa denominado *Smart Campus* FACENS para criar uma cidade inteligente dentro do campus universitário. Embora seja de cunho educacional, a proposta idealizada do projeto visa à unificação das diversas atividades desenvolvidas pelo campus da faculdade para atender às necessidades e proporcionar aos estudantes uma visão holística do mundo contemporâneo (ROMANO; PINTO; PACHECO, 2016).

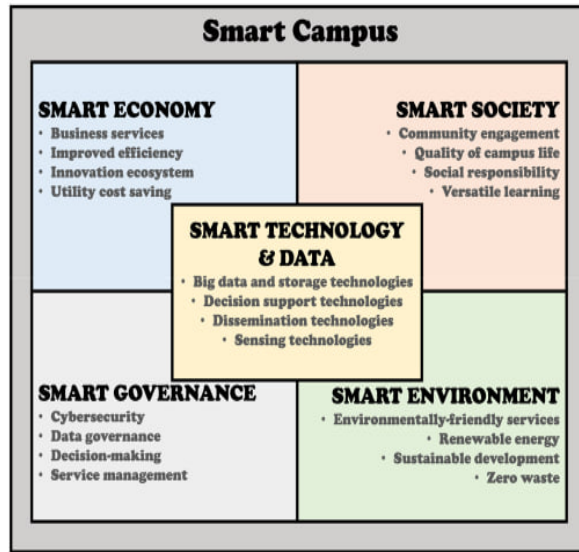
Segundo Neves *et al.* (2017), é importante a circulação da informação entre os projetos e as comunidades, ressaltando a necessidade do envolvimento e da participação dos usuários, bem como o ambiente de experimentação para o desenvolvimento de testes, para realizar adaptações quando necessárias. Portanto, os projetos de *smart campus* devem permitir o acesso aos dados coletados e criar novas experiências, visando às necessidades dos envolvidos no ambiente universitário, além de apresentar um modelo multidisciplinar com estratégias integradas e estruturadas que possam abranger diversos setores e atores.

2.1.1 Concepção de um *framework* de *smart campus*

Neves *et al.* (2017) afirmam que para auxiliar na construção da visão de um projeto *smart campus* é preciso especificar uma metodologia. Portanto, a iniciativa pode ser baseada na utilização de um *framework* conceitual, o qual pode ser adaptado e escalado para os diferentes tipos de contextos.

Na concepção abordada no estudo por Polin *et al.* (2023), evidencia-se a criação de um *framework* categorizado em quatro domínios: a economia, a sociedade, o ambiente, a governança e, por fim, e mais abrangente, a perspectiva de dados e tecnologia para o desenvolvimento de um campus inteligente conforme a Figura 1.

Para cada domínio, Polin *et al.* (2023) evidenciam tópicos de orientação para embasamento de pesquisa, logo tem como base publicações de artigos realizados na área de *smart campus*. Desta forma, pode-se deduzir que estes são os aspectos comuns enfrentados e estudados.

Figura 1 – *Conceptual framework of smart campus*

Fonte: Polin *et al.* (2023)

A definição para economia (*Smart Economy*) é dada como investimentos ativos e negócios, responsáveis por provisionar, eficientemente, os serviços para operação e organização da sociedade. Ademais, para os centros universitários, promove o empreendimento para gerar capital de investimentos com o intuito de financiar pesquisas científicas.

No domínio social (*Smart Society*), são observados os problemas enfrentados pelos usuários do campus, no que diz respeito aos aspectos de segurança, saúde e qualidade da vida acadêmica, levando em consideração a observância do uso dos recursos e serviços, promovendo assim, o campus e seus integrantes para atender aos aspectos citados.

Para o domínio ambiental (*Smart Environment*), são observados os aspectos de infraestrutura, construções e elementos da natureza, em que Polin *et al.* (2023) afirmam que deve haver um consenso para que se mantenha um ambiente sustentável e saudável.

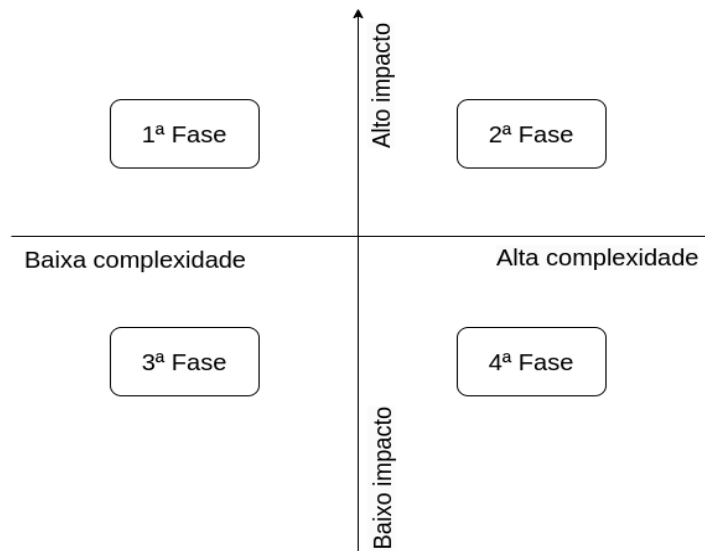
A governança (*Smart Governance*) é responsável por lidar com as tomadas de decisões, de modo que sejam criadas estratégias que melhorem o engajamento dos usuários. Além disso, é responsável pelo gerenciamento e organização, por meio de políticas públicas e administrativas.

O quinto domínio, perspectiva de dados e tecnologia (*Smart Technology & Data*), abrange todos os anteriores. Polin *et al.* (2023) apresentam esta quinta categoria como uma interação dos quatro domínios por meio da tecnologia. Portanto, as informações que são possíveis de serem coletadas, se tornam dados, que são armazenados e ordenados pela parte operacional. A finalidade deste domínio é ser um recurso que possa melhorar e otimizar as funções do campus, por meio de estudo e análise desses dados.

Tendo em vista as orientações de domínios, é importante fazer o estudo de caso

para que a proposta seja atendida, adequando-se às necessidades da instituição. Portanto, Neto e Bandeira (2019) apresentam o gráfico ilustrado na Figura 2, como uma referência para ter uma base de análise.

Figura 2 – Impacto versus Complexidade



Fonte: Neto e Bandeira (2019)

Cada uma das fases é subjetiva ao levantamento das ações propostas para o projeto do *smart campus*. Estas ações partem das necessidades dos usuários do campus. Portanto, justifica-se a subjetividade em que cada fase pode ser classificada quanto à sua complexidade de execução e impacto na universidade. Desse modo, esta análise complementa a criação de um *framework*, uma vez que é possível discutir os aspectos que devem ser levados em consideração para cada fase.

Neves *et al.* (2017) apresentam um *framework* conceitual, em que especificam etapas para poder guiar todo o processo de tomada de decisão. São definidas seis fases, sendo elas:

- Fase I - Planejamento preliminar: é um levantamento prévio, cujo objetivo é enumerar os desafios, apresentar os reais potenciais, as áreas a serem investigadas, bem como os usuários beneficiados. Define também os dados que serão coletados, e como serão feita a coleta e a viabilidade do projeto;
- Fase II - Identificação das áreas: identificação das áreas que serão afetadas;
- Fase III - Aquisição de dados: construção de uma base de dados com base na área selecionada;
- Fase IV - Análise dos dados: tratamento dos dados e extração de conhecimento baseado no objetivo final;

- Fase V - Categorização dos problemas: avaliação dos dados para identificar possíveis deficiências, e
- Fase VI - Definição das estratégias: definição das melhores estratégias a serem desenvolvidas para o problema a ser resolvido.

2.1.2 Aplicabilidade viabilizando o presente trabalho

Para o presente trabalho, o foco é no domínio de sociedade. Cujo a solução é um sistema de recomendação que possui um ferramental que possibilite ampliar o engajamento entre alunos e professores. Tendo isso em vista, foi observada a criticidade de atendimento das demandas de TCC no campus FGA para o curso de Engenharia de Software.

Segundo Neto e Bandeira (2019), o *smart campus* é um modelo de proposta que pode ser utilizado para reduzir impactos de crise orçamentária nas universidades, além de poder apoiar outras necessidades legais como ações de sustentabilidade, transparência e governança.

Neto e Bandeira (2019) ressaltam que é necessário um estudo longitudinal para avaliar os impactos da criação de um *smart campus*. Por meio desse estudo, é possível observar que este modelo pode ser adaptado às necessidades de cada instituição, na problemática em questão. Portanto, uma vez que a universidade apresenta problemas que sejam pertinentes, a criação de um estudo direcionado e a injeção do mesmo pode ser eficiente.

2.2 Sistema de Recomendação

2.2.1 Definição

Pode-se definir sistema de recomendação como um mecanismo que realiza filtragem de informações com o intuito de fornecer sugestões personalizadas ao usuário com base em seus interesses pessoais (SILVA, 2014). Para isso, são utilizadas diversas técnicas do princípio da personalização da recuperação da informação, que depende da identificação e da modelagem das preferências do usuário (SILVA, 2014).

Dessa forma, os sistemas de recomendação são soluções que surgiram visando lidar com a sobrecarga de informações, oferecendo aos usuários sugestões de uma ampla variedade de itens, tais como livros, séries, filmes e músicas, por exemplo (BARBOSA, 2014). No contexto atual, os sistemas de recomendação são amplamente utilizados em lojas *online*, redes sociais, serviços de *streaming* de música, plataformas de *streaming* de vídeo e até mesmo em aplicativos de namoro. Eles desempenham um papel crucial na

experiência do usuário, permitindo novas descobertas com base nos interesses individuais do usuário.

2.2.2 Técnicas de Recomendação

No âmbito dos sistemas web, há uma variedade de técnicas de recomendação amplamente empregadas, cada uma com suas próprias vantagens e limitações. Entre as principais, estão a recomendação baseada em conteúdo, a filtragem colaborativa e a filtragem híbrida. Cada uma dessas abordagens desempenha um papel fundamental na personalização das recomendações, considerando diferentes aspectos, como características dos itens, comportamentos e preferências dos usuários, para fornecer sugestões relevantes e úteis.

2.2.3 Filtragem Baseada em Conteúdo

A filtragem baseada em conteúdo utiliza as características e os atributos dos itens com o intuito de fornecer sugestões personalizadas. Para tanto, é necessário considerar o cálculo de similaridade entre os atributos, realizando comparações dos itens para identificar padrões, e realizar recomendações com base na similaridade encontrada (SILVA, 2014).

Para implementar um sistema de recomendação baseado em conteúdo, é necessário, primeiramente, realizar a representação dos dados textuais. Esses dados podem ser estruturados ou semiestruturados, sendo que a abordagem mais comum para esta representação é a *Bag-of-words* que tem como intuito a representação de documentos textuais em vetores de palavras. Em seguida, verifica-se a frequência das palavras em cada documento (SILVA, 2014).

Dessa forma, segundo Silva (2014), o processo de construção da representação pode ser dividido em três etapas principais.

A primeira etapa é a extração. Nela, o intuito é separar o texto em partes, que estão atreladas à representação escolhida. Durante essa etapa, é importante aplicar filtros para remover palavras sem significados específicos e reduzir palavras à sua raiz (SILVA, 2014).

Na segunda etapa, que é a seleção, são utilizadas as palavras-chave extraídas da etapa anterior, e definida a importância de cada palavra dentro do documento por meio da aplicação de pesos. Segundo Silva (2014), a forma mais comum de atribuir pesos para verificar a importância de termos em um documento é utilizando a combinação de *Term Frequency* (TF) e *Inverse Document Frequency* (IDF), também conhecido como TF-IDF.

O algoritmo *Term Frequency-Inverse Document Frequency* (TF-IDF) utiliza a combinação do TF, que mede a frequência de um termo em um documento específico, e do IDF que mede a relevância de um termo em relação a um conjunto de documentos. Com

isso, é definido o valor do TF-IDF, que representa a importância de um termo para um documento em relação aos outros termos.

A frequência de um termo (TF) é determinada dividindo-se o número de ocorrências de um determinado termo em um documento, sobre o número total de ocorrências dos outros termos neste mesmo documento. Abaixo, é representada a equação matemática:

$$\text{TF}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \quad (2.1)$$

$\text{TF}_{i,j}$ representa a frequência do termo i no documento j , $n_{i,j}$ é o número de ocorrências do termo i no documento j , e $\sum_k n_{k,j}$ é a soma de todas as ocorrências de todos os termos no documento j .

A frequência inversa (IDF) é determinada pelo logaritmo da divisão da quantidade total de documentos no conjunto, pela quantidade total dos documentos em que o termo em questão está presente. A seguir, está a representação matemática:

$$\text{IDF}_j = \log \left(\frac{n}{df_j} \right) \quad (2.2)$$

IDF_j representa a relevância do termo j , n é o número total de documentos no conjunto, e df_j é o número de documentos que contêm o termo específico considerado.

O valor final do TF-IDF resulta da multiplicação destas duas frequências. Dessa forma, o TF-IDF considera tanto a frequência do termo dentro do documento (TF), quanto a sua relevância em toda a coleção de documentos. Além disso, quanto maior o valor do TF-IDF, mais importante será o termo em um documento específico em relação à coleção de documentos.

A última etapa de construção de representação de documentos é a redução de dimensionalidade. Segundo [Silva \(2014\)](#), realizar a redução de dimensionalidade, e manter a representatividade dos documentos são um dos desafios para manter escalabilidade na recomendação baseada em conteúdo.

Após finalizar a construção da representação de conteúdo, é necessário realizar a construção do vetor de características baseado nos valores do TF-IDF. Nesse processo, cada componente do vetor é responsável por representar a importância do termo correspondente.

2.2.4 Similaridade

Finalizada a etapa de construção da representação de conteúdo, e vetor de características baseado no TF-IDF, é iniciada a etapa do cálculo da similaridade entre os itens. Para este fim, será utilizada a similaridade por cosseno, utilizando o ângulo entre os vetores TF-IDF de pesos dos termos relevantes, dada pela equação abaixo:

$$\text{Similaridade} = \cos(\theta) = \frac{A \cdot B}{\|A\| \cdot \|B\|} = \frac{\sum_{i=1}^n A_i \cdot B_i}{\sqrt{\sum_{i=1}^n A_i^2} \cdot \sqrt{\sum_{i=1}^n B_i^2}} \quad (2.3)$$

Realizado o cálculo de similaridade, é possível gerar um *ranking* indicando a ordem dos itens mais similares à preferência do usuário. Essa informação é utilizada para realizar as recomendações personalizadas (SILVA, 2014).

Para o sistema de recomendação são necessários parâmetros que possam ser utilizados como base de cálculo da equação, sendo valores significativos para atribuir a conjuntos que possam ser comparados, sendo, por exemplo, valores característicos que possam ser comparados entre os domínios, tal qual possibilite operações de união e interseção.

2.2.5 Vantagens e Limitações

A filtragem baseada em conteúdo textual é uma abordagem de recomendação utilizada em diversos sistemas. Essa abordagem possui vantagens e desvantagens, que podem variar conforme o contexto e com a aplicação. Segundo Silva (2014), as principais vantagens da filtragem baseada em conteúdo são:

- Para realizar uma recomendação baseada em conteúdo, basta ter conhecimento dos interesses individuais do usuário, não sendo necessário conhecer usuários com perfis semelhantes, e apropriar-se dessa semelhança para realizar uma recomendação.
- Transparência, uma vez que na recomendação não são utilizados dados de outros usuários, mas apenas interesses do próprio usuário.
- Diferente de outras abordagens de recomendação, a filtragem baseada em conteúdo não necessita de avaliações de usuários para realizar uma recomendação. Como neste caso são utilizadas as características dos itens, é possível recomendar um item que tenha sido adicionado recentemente na base de dados.

No entanto, é importante destacar algumas limitações que devem ser consideradas. Dentre elas estão:

- Como a filtragem baseada em conteúdo visa recomendar itens similares com base nos interesses individuais do usuário, essa abordagem, conseqüentemente, não possui variedade na recomendação; isto é, a recomendação fica restrita aos interesses específicos apresentados pelo usuário, sem explorar opções mais diversas.
- Desafio na qualidade da recomendação, uma vez que o conteúdo utilizado é apenas o textual. Além disso, alguns aspectos mais sutis que dependem do contexto, e vão além da similaridade textual, podem ser desconsiderados.

2.2.6 Filtragem Colaborativa

Segundo [Silva \(2014\)](#), diferente da filtragem por conteúdo que utiliza similaridade entre os itens para realizar uma recomendação, a filtragem colaborativa utiliza as preferências do usuário em questão e também as preferências de usuários semelhantes a ele. Isso só é possível, pois há uma tendência de que usuários com interesses semelhantes tenham preferências similares ([BARBOSA, 2014](#)). Dessa forma, a filtragem colaborativa fornece recomendações com uma variedade maior e, por conta disso, a filtragem colaborativa é extremamente popular em sistemas de recomendação. Segundo o estudo feito por [Silva \(2014\)](#), os algoritmos de filtragem colaborativa são divididos e baseados em duas categorias: memória e modelo.

2.2.7 Algoritmos Baseados em Memória

Os algoritmos de filtragem colaborativa baseado em memória possuem a capacidade de guardar informações relacionadas aos interesses do usuário em memória, para que em seguida seja feito o cálculo da predição da avaliação do usuário ([SILVA, 2014](#)). Nesse caso, a predição de uma avaliação pode ocorrer em duas abordagens, baseadas em usuário ou item.

2.2.7.1 Baseado em Usuário

O algoritmo baseado em usuário é uma técnica cujo intuito é fazer recomendações baseadas nas avaliações do usuário em questão, em consonância com as preferências de outros usuários semelhantes.

Dessa forma, primeiramente é necessário coletar avaliações do usuário e, com base nessas avaliações, são identificados os "vizinhos" deste usuário, ou seja, outros usuários com preferências similares. Uma vez identificados os vizinhos, são utilizadas as avaliações deles para realizar uma recomendação. Ou seja, se um vizinho avaliou positivamente um item que o usuário em questão ainda não tenha visto, o sistema pode sugerir este item para o usuário, uma vez que ambos possuem preferências em comum.

Essa abordagem faz uso de dados coletados dos usuários semelhantes para fornecer recomendações personalizadas. Este tipo de recomendação é comumente utilizada em sites de *streaming*.

2.2.7.2 Baseado em Item

Segundo [Silva \(2014\)](#), esse algoritmo ganhou popularidade, principalmente, em lojas virtuais, após ser adotado pela Amazon no ano de 2003. Sua abordagem consiste em selecionar um item e, a partir dessa seleção, busca-se identificar os itens semelhantes, também conhecidos como vizinhos, que compartilham características e avaliações similares.

Uma vez concluída a etapa de identificação dos vizinhos, é possível oferecer recomendações personalizadas ao usuário.

Esse algoritmo é amplamente utilizado em plataformas de comércio eletrônico, especialmente na apresentação de itens aos usuários. Quando um usuário acessa um item específico, é comum que o sistema automaticamente exiba os itens vizinhos, acompanhados de frases como "quem se interessou por este item também se interessou por estes outros itens".

2.2.8 Algoritmos Baseados em Modelo

Ao contrário dos métodos baseados em memória, nos quais as recomendações são feitas com base na similaridade entre usuários ou itens, os algoritmos baseados em modelo buscam criar um modelo que capture as características dos usuários e itens para fazer previsões.

Dessa forma, são criados modelos preditivos a partir de um conjunto de dados que contém informações sobre as preferências e avaliações dos usuários com o intuito de realizar recomendações. No entanto, a utilização de algoritmos de aprendizado de máquina na construção desses modelos torna a etapa de treinamento mais custosa (SILVA, 2014).

Uma vez construído o modelo, ele pode ser usado para realizar recomendação para itens não avaliados pelos usuários. Essas recomendações são baseadas nas informações contidas no modelo, que captura os padrões e relações entre os usuários e itens.

2.2.9 Vantagens e Limitações

A filtragem colaborativa fundamenta-se nas relações de preferências entre os usuários, utilizando suas avaliações para proporcionar recomendações personalizadas. Assim como a filtragem baseada em conteúdo, a filtragem colaborativa possui algumas vantagens e limitações que merecem destaque. Alguns aspectos a serem considerados:

- Ao utilizar os interesses e avaliações de um grupo de usuários semelhantes, a filtragem colaborativa é capaz de oferecer recomendações abrangentes e inesperadas, tornando a experiência de recomendação mais enriquecedora. Desse modo, é possível que sejam feitas recomendações que levem o usuário a realizar descobertas de interesses.
- A filtragem colaborativa tem a capacidade de abranger uma ampla variedade de conteúdos, como livros, notícias, produtos, músicas ou filmes, e não há restrições relacionadas a idiomas ou ambiguidade nas palavras (SILVA, 2014). Além disso, a filtragem colaborativa permite incorporar conceitos que vão além do contexto textual, como a noção de qualidade.

Conforme o estudo realizado por [Silva \(2014\)](#), existem algumas limitações que devem ser consideradas:

- Dificuldade em lidar com esparsidade, uma vez que a presença de uma grande quantidade de valores ausentes ou avaliações escassas podem dificultar a identificação de vizinhos adequados e afetar a qualidade das recomendações.
- Em consequência da ausência de avaliações, surge o problema da partida a frio (*cold-start problem*), dividido em duas categorias. A primeira é o problema do novo usuário (*new user problem*), em que é desafiador recomendar itens para usuários que possuem poucas ou nenhuma avaliação. A falta de informações sobre suas preferências dificulta a identificação de usuários semelhantes e, conseqüentemente, afeta o processo de recomendação. A segunda categoria é o problema do novo item (*new item problem*), que ocorre quando um novo item é adicionado ao sistema e não possui avaliações dos usuários. Nesses casos, a falta de dados dificulta a incorporação do item no processo de filtragem colaborativa.
- A filtragem colaborativa baseada em modelo requer um processo de treinamento e construção do modelo, o que pode demandar maiores recursos computacionais e maior tempo, que em comparação com os métodos baseados em memória e filtragem baseada em conteúdo.

2.2.10 Filtragem Híbrida

As abordagens de filtragem apresentadas até o momento possuem tanto vantagens quanto desvantagens, conforme mencionado anteriormente. A filtragem híbrida surge com o intuito de unir duas ou mais técnicas de recomendação, com o propósito de combinar as vantagens e diminuir as desvantagens das abordagens, visando alcançar resultados mais eficientes.

Tem-se como exemplo de filtragem híbrida a combinação de sistemas de recomendação baseada em conteúdo e em filtragem colaborativa, concentrando-se nos itens recomendados por ambas as abordagens. Outra estratégia utilizada é iniciar as recomendações com base na filtragem colaborativa e, posteriormente, aprimorar a lista de itens por meio da filtragem baseada em conteúdo ([COUTINHO, 2022](#)).

Segundo [Coutinho \(2022\)](#), um dos desafios dos métodos híbridos reside no fato de como combinar diferentes técnicas, alcançando uma combinação eficiente. Para isso, destaca-se a utilização de métodos baseados em redes neurais pois possibilitam a aprendizagem de representações intermediárias considerando diversas informações ([COUTINHO, 2022](#)).

2.3 Resumo do Capítulo

Este capítulo apresenta os principais conceitos e referências que darão suporte para o presente trabalho.

Inicialmente, introduziu-se o conceito de *smart campus*, concepção de um *framework* de *smart campus* e viabilidade da aplicabilidade para a UnB. Posteriormente, na parte de sistema de recomendação, foram exploradas técnicas de recomendação existentes, ao detalhar a concepção e a utilização de cada método, de modo a apresentar as vantagens e limitações de cada abordagem. Por fim, foi apresentado a Filtagem Híbrida.

3 Suporte Tecnológico

Este capítulo tem como objetivo apresentar os principais recursos tecnológicos que foram utilizados no presente trabalho.

Dado que o propósito do trabalho é desenvolver um aplicativo, foram abordadas ferramentas que deram suporte ao seu desenvolvimento, versionamento e implantação, tais como: React Native, Nest.js, MongoDB, Git, Github, Expo, Expo Go e Amazon EC2. Além disso, foram apresentadas ferramentas de fluxograma e prototipação, como Mural e Figma. Para a coleta de dados por meio de formulários, foi utilizado o Google *Forms*. Por fim, tem-se o resumo do capítulo.

3.1 React Native

React Native é um *framework open-source* de desenvolvimento de aplicativos móveis, que utiliza a linguagem de programação Javascript. O React Native foi criado pelo Facebook, e permite o desenvolvimento de aplicativos para os sistemas operacionais Android e iOS, com a utilização de um único código-fonte, que economiza tempo e esforço no desenvolvimento de aplicativos [MARCAC \(2021\)](#).

Além disso, com o desenvolvimento de aplicativos utilizando React Native é possível utilizar recursos nativos do próprio dispositivo, tais como: câmera, geolocalização e notificação, permitindo desenvolvedores criarem aplicativos robustos sem a necessidade de conhecimentos específicos de programação nativa para cada plataforma.

Tudo isso torna o React Native uma poderosa ferramenta para o desenvolvimento de aplicativos móveis multiplataforma, que combina a produtividade e a facilidade de uso do JavaScript.

3.2 Nest.js

Segundo a documentação ([NEST.JS, 2023](#)), é um *Framework* de desenvolvimento de aplicações Node.js eficientes e escaláveis do lado do servidor. Ele utiliza a linguagem Javascript e oferece suporte para Typescript. Além disso, combina elementos de programação orientada a objetos, programação funcional e programação reativa funcional.

O Nest.js tem uma forte inspiração na arquitetura do *Framework front-end* Angular e, por conta disso, consegue fornecer uma estrutura modular e extensível. Dessa forma, o Nest.js utiliza conceitos de módulos, controladores e provedores para organizar e separar as partes do aplicativo em componentes reutilizáveis e independentes.

Além disso, o Nest.js, além de oferecer uma experiência de desenvolvimento ágil, também oferece suporte para uma ampla variedade de recursos e integrações. Por conta deste e outros motivos, o Nest.js é uma ótima opção para criar aplicações altamente testáveis, escaláveis, fracamente acoplados e de fácil manutenção.

3.3 MongoDB

Segundo a documentação oficial ([MongoDB, 2023](#)), o MongoDB é um banco de dados não relacional orientado a documentos, e possui alto desempenho por conseguir gerenciar grandes volumes de dados de forma eficiente e escalável. Dentre as diversas características que torna o MongoDB uma ótima opção de banco de dados para desenvolvimento, estão:

- Escalabilidade horizontal: O MongoDB é altamente escalável e, por ser um banco de dados não relacional, possui suporte à escalabilidade horizontal, visto que permite distribuir dados em vários servidores e *clusters* para melhorar o desempenho e a capacidade de resposta;
- Flexibilidade do esquema: O MongoDB permite que você adicione, modifique ou remova campos dos documentos sem afetar a estrutura dos outros documentos. Essa flexibilidade é extremamente útil durante as fases iniciais do desenvolvimento, quando os requisitos estão sujeitos a mudanças frequentes, e
- Integração com linguagens diversas linguagens de programação: O MongoDB é compatível com várias linguagens de programação, incluindo JavaScript, Python, Java, C, Ruby e muitas outras. Ele fornece suporte e bibliotecas oficiais para facilitar a integração do MongoDB com diferentes ecossistemas de desenvolvimento.

Estas são as principais características que o tornam uma excelente opção para o desenvolvimento de aplicativos que exigem gerenciamento eficiente de grandes volumes de dados e adaptação às mudanças nos requisitos do projeto.

3.4 Git e Github

Pode-se dizer que o Git é um sistema de controle de versão distribuído, com o intuito de facilitar o rastreamento e controle de mudanças em projetos de desenvolvimento de software ([GIT, 2023](#)). Dessa forma, o Git permite que diversos desenvolvedores trabalhem em único projeto de forma colaborativa e o Git faz o registro completo de todas as alterações feitas por cada desenvolvedor.

Segundo a documentação ([GITHUB, 2023](#)), o Github é uma plataforma de hospedagem de código para controle de versão que utiliza o Git e permite que pessoas trabalhem em conjunto em projetos de qualquer lugar.

No GitHub, os projetos são organizados em repositórios, que são locais onde o código-fonte, documentação e recursos relacionados são armazenados. Os desenvolvedores podem enviar alterações nos arquivos do repositório e manter um histórico completo de todas as alterações realizadas.

Além disso, O GitHub oferece uma variedade de recursos poderosos que facilitam o desenvolvimento de software e a colaboração entre desenvolvedores. Por meio dos recursos do *Git*, é possível criar versões separadas do código para realizar tarefas sem afetar a versão principal. Por meio dos recursos do Github, é possível realizar revisões e discussões antes de mesclar alterações ao projeto e criação de *issues*, que permite a documentação de melhorias e o que está sendo feito. Por conta disso, o GitHub é uma ótima opção para o gerenciamento de código.

3.5 Mural

Mural é uma ferramenta *online* para criação de fluxogramas em geral, podendo utilizar modelos pré-definidos, ou de própria autoria. Essa ferramenta permite a colaboração síncrona entre os usuários. Além do ambiente com alta capacidade de personalização, possui um conjunto de funcionalidades que podem contribuir para a elaboração do projeto e também para o engajamento dos contribuintes.

3.6 Figma

Figma é uma ferramenta de *design* de interface que permite que usuários criem, colaborem e compartilhem interfaces de maneira eficiente ([FIGMA, 2023](#)). Dentre suas principais características, estão:

- Colaboração em tempo real: permite que vários usuários trabalhem simultaneamente em uma mesma interface, tornando o trabalho em equipe mais eficaz;
- Prototipagem interativa: permite criar fluxos de interação entre as interfaces criadas, sendo possível realizar testes de usabilidade com a ferramenta, e
- Inspeção de informações: permite que os desenvolvedores verifiquem informações como dimensão, cor, estilo e outras propriedades das interfaces criada para serem utilizadas no desenvolvimento.

Por conta dessas características e por ser uma ferramenta amplamente adotada no mercado por equipes de *design*, o Figma foi escolhido como ferramenta para realizar prototipações.

3.7 Google Forms

O Google *Forms* é uma ferramenta *online* criada pelo Google, que permite criar questionários e formulários de forma gratuita. Além disso, a ferramenta também possui análise de respostas em tempo real, com a possibilidade de exportar os dados em uma planilha para facilitar a visualização dos dados coletados.

3.8 Expo

Expo é um *framework* de código aberto para desenvolver aplicativos com React Native. Sua principal vantagem é a possibilidade de executar aplicativos React Native sem a necessidade de utilizar ferramentas de compilação de código nativo. O Expo simplifica e acelera o processo de desenvolvimento, sem a necessidade de lidar com muitos detalhes de configuração (EXPO, 2023b).

3.9 Expo Go

Expo Go é um aplicativo que possibilita a execução de aplicativos desenvolvidos por meio do Expo diretamente em dispositivos iOS e Android. Ele permite visualizar e testar aplicações em desenvolvimento em tempo real, que proporciona praticidade no desenvolvimento de aplicativos móveis (EXPO, 2023a).

3.10 Amazon EC2

Amazon *Elastic Compute Cloud*, ou simplesmente Amazon EC2, é um serviço de computação em nuvem provida pela Amazon *Web Services*. Este serviço permite a implantação de aplicações de forma escalável através de máquinas virtuais (também chamadas de instâncias). O Amazon EC2 permite que os usuários tenham controle da instância, visto que pode ser configurado e personalizado conforme a necessidade do usuário (SERVICES, 2023).

3.11 Resumo do Capítulo

No capítulo, foram introduzidas as principais ferramentas e tecnologias utilizadas para auxiliar no desenvolvimento deste trabalho. Essas ferramentas abrangem áreas como

prototipação, desenvolvimento de software, diagramação e entre outras.

As principais ferramentas e tecnologias abordadas neste capítulo incluem React Native, Nest.js, MongoDB, Git, GitHub, Mural, Figma, Expo, Expo Go e Amazon EC2. Além disso, serão utilizadas ferramentas para edição de texto (i.e Overleaf); comunicação entre os membros (i.e Telegram e Discord) e elaboração de diagramas (i.e Draw.io). Tais ferramentas, assim como as descritas anteriormente, serão de extrema importância para responder às questões de pesquisa. O Quadro 1 apresenta um resumo das ferramentas utilizadas.

Quadro 1 – Resumo das Tecnologias

Nome	Descrição	Link
React Native	<i>Framework</i> para desenvolvimento de aplicativos móveis multiplataforma.	< https://reactnative.dev >
Nest.js	<i>Framework</i> JavaScript para construção de aplicativos back-end escaláveis e eficientes.	< https://nestjs.com >
Mongodb	Banco de dados não relacional orientado a documentos.	< https://mongodb.com >
Git	Sistema de controle de versão distribuído.	< https://git-scm.com >
Github	Plataforma para armazenamento e colaboração de código-fonte.	< https://github.com >
Draw.io	Ferramenta <i>online</i> gratuita para criação de diagramas e fluxogramas.	< https://draw.io >
Figma	Aplicação web para desenho e criação de interfaces.	< https://figma.com >
Telegram	Aplicativo de mensagens instantâneas com foco em segurança e privacidade 3	< https://web.telegram.org >
Discord	Plataforma de comunicação por voz, vídeo e texto	< https://discord.com >
Mural	Ferramenta <i>online</i> colaborativa de fluxograma.	< https://app.mural.co >
Google Forms	Ferramenta <i>online</i> para criação de formulários.	< https://docs.google.com/forms >
Expo	<i>Framework</i> para aplicações React Native.	< https://docs.expo.dev >
Amazon EC2	Serviço de computação em nuvem escalável.	< https://aws.amazon.com/pt/ec2/ >
Expo Go	Aplicativo para visualizar e testar projetos Expo.	< https://expo.dev/client >
Overleaf	Editor <i>online</i> de documentos LaTeX.	< https://overleaf.com >

Fonte: Autor

4 Metodologia

O intuito deste capítulo é detalhar os aspectos metodológicos que conduziram este trabalho. Dessa forma, é apresentada a classificação de pesquisa para posicionar a pesquisa em relação à abordagem, natureza, objetivo e procedimentos. Em seguida tem-se o fluxo de atividades para a primeira e segunda etapa do TCC. Depois, tem-se o detalhamento da metodologia de desenvolvimento e metodologia de análise de resultados. Os cronogramas são apresentados para fornecer detalhamento temporal. Por fim, tem-se o resumo do capítulo.

4.1 Classificação da Pesquisa

De acordo com [Gerhardt e Silveira \(2009\)](#), uma pesquisa tem como intuito encontrar respostas para um problema em específico. Além disso, uma pesquisa pode ser definida considerando diferentes aspectos, como a abordagem, a natureza, os objetivos e os procedimentos. Com base nessa perspectiva, os tópicos seguintes buscam detalhar cada um destes aspectos.

4.1.1 Abordagem

Para este trabalho, foi utilizada uma abordagem mista, combinando tanto aspectos qualitativos quanto quantitativos. Neste trabalho, foram estabelecidas métricas que possuem um caráter quantitativo, como a média de orientações conduzidas por professores por semestre. Além disso, também foram utilizados aspectos qualitativos, como as dificuldades enfrentadas pelos alunos no processo de busca por um orientador.

4.1.2 Natureza

O objetivo geral deste trabalho foi desenvolver um aplicativo móvel que possa facilitar o processo de busca por professor orientador baseado em interesses mútuos. Dessa forma, a natureza desta pesquisa pode ser definida como aplicada.

4.1.3 Objetivos

Em relação à classificação dos objetivos, a pesquisa pode ser definida como exploratória. Nesse sentido, a pesquisa visa proporcionar uma maior familiaridade com o tema ao buscar uma compreensão mais aprofundada para formular hipóteses e torná-lo mais explícito ([GERHARDT; SILVEIRA, 2009](#)).

4.1.4 Procedimentos

Em relação aos procedimentos, esta pesquisa pode-se definir como: pesquisa bibliográfica, visto que a realização de levantamento bibliográfico envolvendo algoritmos de filtragem e *smart campus* deram suporte para realização deste trabalho (GERHARDT; SILVEIRA, 2009); pesquisa-ação, pois foram utilizados métodos científicos em consonância com as ações práticas a fim de solucionar um problema coletivo (GERHARDT; SILVEIRA, 2009).

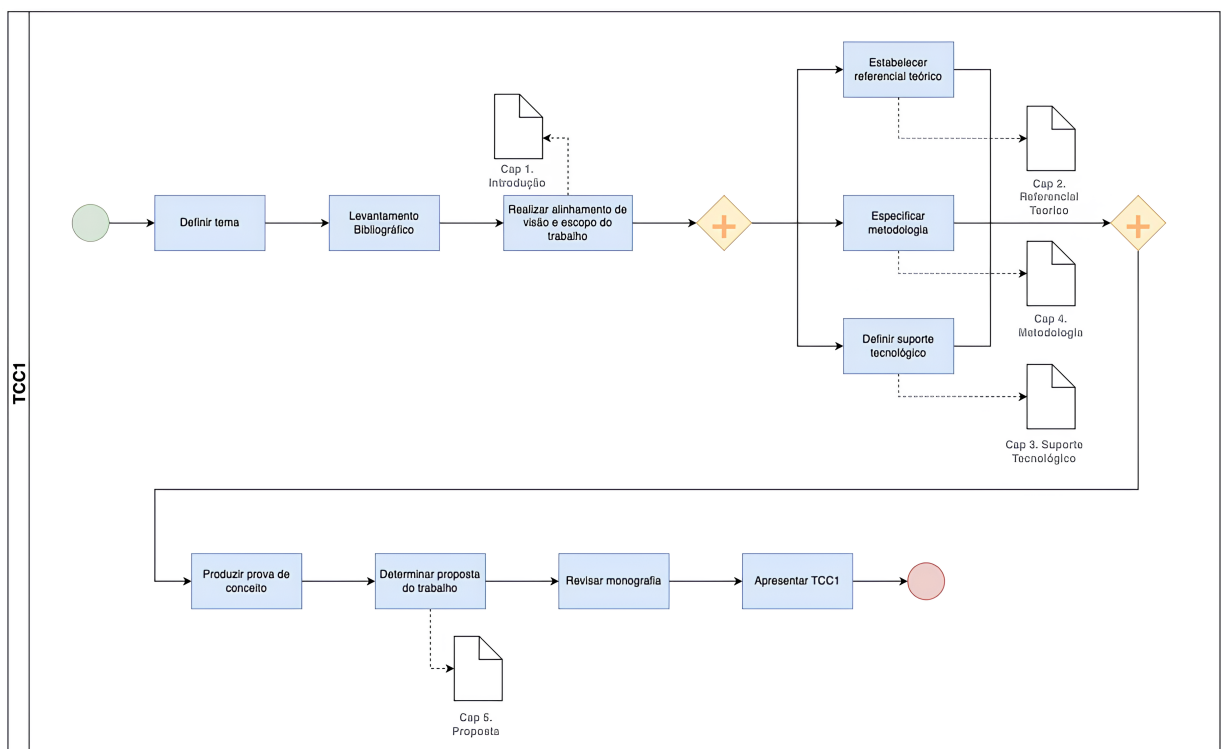
4.2 Fluxo de Atividades

Conforme pode ser visto na Figura 3, as atividades e subprocessos relacionados à primeira etapa do TCC foram:

- Definir tema: definir um tema que tenha fundamentação na área de Engenharia de *Software* para a execução do trabalho. Desse modo, observada a dificuldade enfrentada pelos alunos da FGA em encontrar um orientador, foi definido o tema "Orientação UnB: Proposta de sistema de recomendação móvel para promover o processo de matrículas em Trabalho de Conclusão de Curso". *Status*: Realizado;
- Levantamento bibliográfico: realizar pesquisa bibliográfica envolvendo o tema definido e aspectos que viabilizam a execução do trabalho. *Status*: Realizado;
- Realizar alinhamento de visão e escopo do trabalho: utilizando processo de *Lean Inception*, definir visão e escopo do trabalho. Dessa forma, foram definidas as principais *features* que o aplicativo deve contemplar. A *Lean Inception* é descrita na seção 4.3 e foi fundamental para estabelecer uma visão do trabalho a ser feito, que é descrita no Capítulo 1. O resultado do processo é detalhado no Capítulo 5. *Status*: Realizado;
- Estabelecer referencial teórico: definir principais conceitos que dão suporte para o trabalho. *Status*: Realizado - Capítulo 2 (Referencial Teórico);
- Definir suporte tecnológico: definir quais serão as ferramentas e tecnologias que darão auxílio para o desenvolvimento do trabalho. *Status*: Realizado - Capítulo 3 (Suporte Tecnológico);
- Especificar metodologia: expor detalhamento metodológico da pesquisa, conforme mostrado no presente capítulo. *Status*: Realizado;
- Produzir prova de conceito: realizar idealização da aplicação utilizando o protótipo de alta fidelidade que será desenvolvido. *Status*: Realizado - Capítulo 5;

- Determinar proposta do trabalho: detalhar sobre a proposta, abordando aspectos técnicos, mostrando a prova de conceito e outros detalhes, os quais estão presentes em Proposta 5. *Status*: Realizado;
- Revisar monografia: ajustes finais da escrita e do conteúdo entregue na primeira etapa do trabalho. *Status*: Realizado, e
- Apresentar TCC1: entregar e apresentar a monografia para membros da banca avaliadora. *Status*: Realizado.

Figura 3 – Fluxo de atividades TCC 1



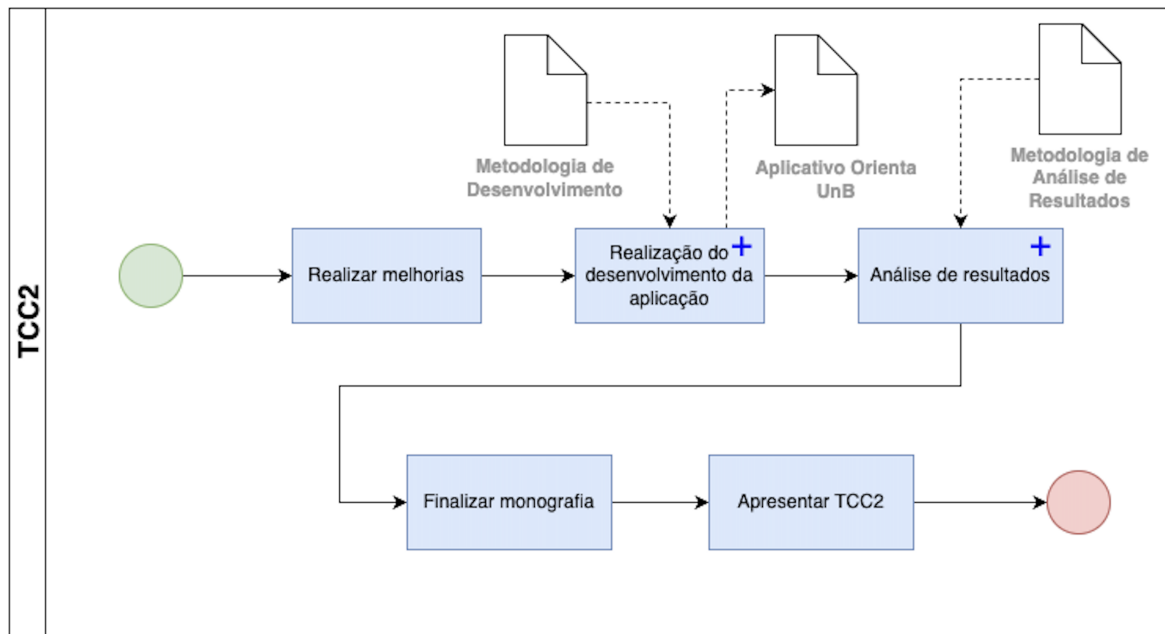
Fonte: Autor

Conforme pode ser visto na Figura 4, as atividades e subprocessos relacionados à segunda etapa do TCC foram:

- Realizar melhorias: realizar melhorias na monografia tendo como base os *feedbacks* fornecidos pela banca após a apresentação. *Status*: Realizado;
- Realização do desenvolvimento da aplicação: este subprocesso foi guiado com base na Metodologia de Desenvolvimento, descrito na seção 4.3. *Status*: Realizado;
- Análise de resultados: coleta de métricas a fim de avaliar o trabalho proposto. Essa etapa foi orientada à Metodologia de Análise de Resultados, descrita na seção 4.4. *Status*: Realizado;

- Finalizar monografia: realizar melhorias na escrita e no conteúdo a ser entregue na segunda etapa do trabalho *Status*: Realizado, e
- Apresentar TCC2: entregar e apresentar monografia para membros da banca avaliadora. *Status*: Realizado.

Figura 4 – Fluxo de atividades TCC 2



Fonte: Autor

4.3 Metodologia de Desenvolvimento

Segundo [SCHWABER e SUTHERLAND \(2020\)](#), o *Scrum* é um *framework* utilizado para o gerenciamento e desenvolvimento de produtos complexos, no qual o trabalho é segmentado em pequenas iterações conhecidas como *sprints*. As *sprints* têm uma duração fixa que comumente varia entre duas a quatro semanas e, ao final de cada *sprint*, uma parcela do produto é entregue.

O *Kanban* é um método de gestão de mudanças que permite a visualização do fluxo de trabalho utilizando um quadro que mostra as tarefas planejadas, em desenvolvimento, e realizadas ([BOEG, 2010](#)).

O *eXtreme Programming* (XP) pode ser definido como uma metodologia ágil de desenvolvimento de *software*, com três características marcantes, sendo elas: *feedback* constante, abordagem incremental e o encorajamento da comunicação entre pessoas ([SOUZA; OLIVEIRA, 2021](#)). Uma das práticas do XP que vale dar destaque é a programação

pareada, na qual duas pessoas trabalham juntas: uma assume o papel de programador, enquanto a outra acompanha e oferece dicas ou sugestões, visando melhorar a qualidade do código.

Segundo (CAROLI, 2018), a *Lean Inception* tem como principal objetivo alinhar a visão de produto, definir um escopo mínimo viável, e criar plano de ação para o desenvolvimento a partir de um processo colaborativo e iterativo.

Diante do exposto, para este trabalho, foi utilizada uma combinação entre as metodologias ágeis levantadas. O desenvolvimento foi segmentado por *sprints* e guiado pelas rotinas do *Scrum*. Com intuito de melhorar a qualidade, o uso da programação pareada foi aplicado. Além disso, foi utilizado um quadro Kanban para visualizar o progresso do trabalho. Por fim, a técnica *Lean Inception* foi utilizada para permitir um alinhamento sobre a visão do produto e definição de um plano de ação.

Dessa forma, a Metodologia de Desenvolvimento do projeto foi guiado pelas seguintes etapas:

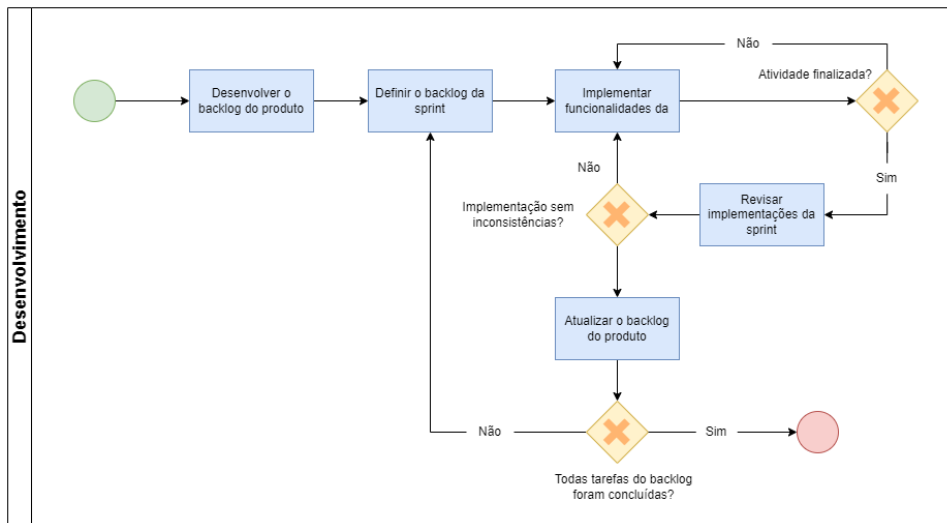
- Desenvolver o *backlog* do produto: deverá ser criado e priorizado uma lista de histórias de usuários que contemple os requisitos que o sistema deve atender;
- Definir *backlog* da *sprint*: escolha das histórias de usuários que deverão ser desenvolvidas na *sprint*;
- Implementar funcionalidades da *sprint*: desenvolver as funcionalidades das histórias de usuário da *sprint*;
- Revisar implementações da *sprint*: realizar revisão das implementações realizadas na *sprint* e garantir que não há inconsistências, e
- Atualizar o *backlog* do produto: realizar a atualização do *backlog* do produto fechando as tarefas já finalizadas ou, quando necessário, criando novas tarefas.

4.4 Metodologia de Análise de Resultados

Para que sejam analisados os resultados obtidos ao longo deste trabalho, foi utilizada a metodologia pesquisa-ação. Suas etapas são segmentadas em:

- Etapa Exploratória: etapa inicial em que são coletados dados e informações para poder ser compreendido o problema. Para isto, foram utilizados formulários com o intuito de fazer um diagnóstico da situação.
- Etapa Planejamento: o intuito dessa etapa é dar continuidade à etapa exploratória e definir um plano detalhado para guiar o processo da pesquisa ação;

Figura 5 – Fluxo de desenvolvimento



Fonte: Autor

- Etapa Ação: momento em que é realizada a aplicação prática baseada nas etapas anteriores. Nessa fase, são avaliados os dados coletados e são propostas ações de pesquisa, e
- Etapa Avaliação: etapa final do processo de pesquisa-ação cujo intuito é verificar os resultados das ações e suas consequências.

4.5 Cronograma

As Figuras 6 e 7 representam, respectivamente, os cronogramas para primeira e segunda etapa do TCC.

Figura 6 – Cronograma TCC 1

	2023			
	Abril	Maio	Junho	Julho
Definir tema	■			
Levantamento Bibliográfico	■			
Realizar alinhamento de visão e escopo do trabalho		■		
Estabelecer referencial teórico		■		
Definir suporte tecnológico		■		
Especificar metodologia			■	
Produzir prova de conceito			■	
Determinar proposta do trabalho				■
Revisar monografia				■
Apresentar TCC 1				■

Fonte: Autor

Figura 7 – Cronograma TCC 2

	2023				
	Agosto	Setembro	Outubro	Novembro	Dezembro
Realizar melhorias	■				
Realizar desenvolvimento da aplicação	■	■	■		
Análise de resultados				■	
Finalizar monografia					■
Apresentar TCC 2					■

Fonte: Autor

4.6 Resumo do Capítulo

O presente capítulo teve como intuito apresentar, detalhadamente, os aspectos envolvendo a metodologia que foi utilizada no trabalho.

Em relação à classificação da pesquisa, foi utilizada uma abordagem híbrida (quantitativa e qualitativa); sua natureza é aplicada; seus objetivos são definidos como exploratório; e seus procedimentos são de pesquisa bibliográfica e pesquisa-ação.

Sobre a metodologia de desenvolvimento, foi utilizada uma combinação entre o *Scrum*, *Kanban*, *XP* e *Lean Inception*. Para metodologia de análise de resultados, foram apresentadas as etapas da pesquisa-ação.

Além disso, foi apresentado o fluxo de atividades, mostrando as atividades e subprocessos que foram utilizados na primeira e segunda etapa do TCC, e, ao final, foi apresentado um cronograma considerando as etapas do trabalho.

5 Estudo Orientado ao Desenvolvimento do Aplicativo Orienta UnB

Este capítulo tem o propósito de apresentar a construção do presente trabalho, levando em consideração todo o contexto deliberado até o momento, visto que a finalidade é apresentar uma solução que possa atender a realidade do campus FGA.

Para o cumprimento do trabalho, são apresentados, na sequência, uma visão geral do aplicativo Orienta UnB, a prova de conceito preliminar, e os detalhamentos dos requisitos com base na *Lean Inception*. Por fim, apresenta-se o resumo deste capítulo.

5.1 Contexto

Em decorrência do aumento do uso de *smartphones* nos câmpus universitários, afirmado por [BANDEIRA e NETO \(2022\)](#), a elaboração de aplicativos móveis possibilita a melhora das capacidades tecnológicas dentro de universidades, gerando e promovendo aspectos para o impulsionamento do processo de ensino e aprendizagem.

Na observação feita por [Schenatz, Cunha e Kugler \(2019\)](#) acerca de inovações tecnológicas, o presente trabalho pretende apresentar uma solução que possa atender aos desejos mútuos entre professores e alunos da UnB FGA, em decorrência do déficit de uma plataforma que possibilita tratar questões relacionadas ao meio de comunicação.

Visto que esta é uma observação apresentada pelos discentes que desejam realizar trabalhos das modalidades TCC, PIBIC, PIBID e PIBITI, o objeto de estudo para esta pesquisa são as demandas de preenchimento de vagas entre alunos e professores durante os períodos semestrais, já que as dificuldades são apresentadas por ambos.

Nesse contexto, o desafio principal desta pesquisa consiste em lidar com as distribuições de quantidade de orientações, de maneira que possa trazer visibilidade dessas quantidades e que, por meio de uma plataforma interativa, apresente sinergias entre alunos e professores, alinhando interesses e capacidade de atendimento para que, conseqüentemente, melhore o processo de inclusão e engajamento entre ambos.

Para o desenvolvimento do aplicativo, foram observados os conceitos de Filtragem Baseada em Conteúdo e *smart campus*, presentes no Capítulo 2 deste trabalho. Além disso, o embasamento para esta proposta dar-se-á pelos norteamentos determinados pelo *framework* definido por [Neves et al. \(2017\)](#), com foco no domínio social, apresentados pela Figura 1 de [Polin et al. \(2023\)](#).

Portanto, o objetivo final do presente trabalho é promover uma aplicação para dispositivos móveis, que empregue sistema de recomendação para facilitar a conexão entre alunos e professores que possuem interesses similares, a fim de atender à demanda dos estudantes de Engenharia de Software da Universidade de Brasília que buscam um orientador para a disciplina de Trabalho de Conclusão de Curso.

5.2 Visão Geral do aplicativo Orienta UnB

O desenvolvimento do aplicativo Orienta UnB foi norteado pelo *framework* conceitual de *smart campus* por [Neves et al. \(2017\)](#) e, portanto, foram levantados os seguintes aspectos para cada uma das fases, conforme indicado nos Quadros 2 a 7:

Quadro 2 – Fase I: Planejamento preliminar

Desafios	Potencial local	Áreas investigadas	Usuários afetados
Demanda de vagas	FGA	Comunicação	Professores e estudantes
Gestão de orientações	FGA	Gestão	Professores
Visibilidade de disponibilidade	FGA	Comunicação	Professores e estudantes
Gestão de processos	FGA	Gestão	Coordenação
Procura por oferta	FGA	Comunicação	Estudantes

Fonte: Autor

Quadro 3 – Fase II: Identificação das áreas

Áreas	Descrição
Gestão	Visa apontar um modelo de gestão que condicionará a utilização da plataforma no meio acadêmico.
Comunicação	Trata-se dos aspectos impactados pelo atual modelo existente para comunicação.

Fonte: Autor

Quadro 4 – Fase III: Aquisição de dados

Coletor	Descrição
Questionário	Por meio de um questionário especificado para o apontamento das dificuldades enfrentadas por alunos e professores no atual modelo da universidade.

Fonte: Autor

Quadro 5 – Fase IV: Análise dos dados

Análise	Descrição
Métricas	Através dos resultados coletados por meio do questionário aplicado no campus.

Fonte: Autor

Quadro 6 – Fase V: Categorização dos problemas

Análise	Descrição
Problemática	Estudo dos resultados para realizar a revisão do levantamento preliminar.

Fonte: Autor

Quadro 7 – Fase VI: Definição das estratégias

-	Descrição
Aplicativo <i>Mobile</i>	Criação de um aplicativo móvel que possa ser utilizado por alunos e professores como ferramenta de auxílio.

Fonte: Autor

5.2.1 Objetivo de aplicação

O objetivo da utilização do *framework* consiste em ser uma forma consolidada de poder alcançar os objetivos determinados na seção 1.4. Portanto, com base nos conceitos de *smart campus*, deseja-se melhorar a qualidade de comunicação por meio de aplicativo, a fim de proporcionar uma experiência personalizada, além de contribuir para a progressão e execução dos processos de matrículas.

5.3 Prova de Conceito Preliminar

A fim de validar a proposta do presente trabalho, foi realizada a criação do aplicativo Orienta UnB, utilizando os suportes tecnológicos acordados no Capítulo 3 - Suporte

Tecnológico. A conclusão do projeto dá-se em três etapas: (i) Coleta de dados referentes ao levantamento preliminar dos desafios apontados no Quadro [??tab:levantamento_preliminar](#), *por meio da util...*

5.3.1 Aplicativo Orienta UnB

Orienta UnB é um aplicativo que utiliza sistema de recomendação, especificamente Filtragem Baseada em Conteúdo, para estabelecer conexões entre alunos e professores que compartilham interesses comuns. Sua proposta é simplificar a colaboração em projetos acadêmicos, otimizando o número de orientações ao unir alunos com mesmo interesse a professores que compartilham dessa área de interesse.

Para utilizar o aplicativo, é necessário que o usuário tenha um email institucional da Universidade de Brasília, e esteja conforme o termo de política de uso e privacidade, que pode ser consultado no Apêndice — C.

O resultado do desenvolvimento do aplicativo pode ser conferido no repositório, acessível na plataforma GitHub pelo seguinte *link*: <https://github.com/Orienta-Universidade-de-Brasilia/TCC-POC>.

5.3.2 Desenvolvimento de Software

Para o desenvolvimento do aplicativo, foram utilizadas técnicas cujas referência encontram-se na seção 4.3. Trata-se da utilização da *Lean Inception* para estabelecer a visão do produto; o escopo de funcionalidades, e quais são as personas envolvidas (*stakeholders*) com seus devidos tipos de perfis. Trata-se ainda das funcionalidades planejadas para uma primeira versão do aplicativo (*Minimum Viable Product - MVP*), e futuras funcionalidades que podem agregar valor para o crescimento deste projeto.

5.3.2.1 *Lean Inception*

Caroli (2017) define etapas para a utilização da *Lean Inception*, estas construídas ao longo do tempo conforme a cronologia padronizada pelo autor. Nesse contexto, vale ressaltar algumas etapas essenciais para a descrição desta subseção. (i) O Canvas MVP é o documento de representatividade do projeto, que descreve o aplicativo de maneira que possa ser apresentado e entendido por uma pessoa leiga, descrito pelo anexo A; (ii) Sequenciador, que define as funcionalidades que serão desenvolvidas, conforme o anexo B.

No Canvas MVP é possível observar os principais aspectos para a descrição do produto. Além disso, é notória a definição dos tipos de perfis que irão utilizar o aplicativo, além de pontuar as respectivas jornadas. Os perfis consistem na visão do professor, de um aluno com alto valor de compatibilidade, e outro aluno com baixa compatibilidade com o perfil do professor. Dessa forma, foi possível traçar o escopo do aplicativo, de maneira que possa cumprir com os objetivos estabelecidos na seção 1.4 deste trabalho.

Uma vez que a visão do produto está definida, e as regras de negócio estabelecidas, foi criado o Sequenciador, que representa as funcionalidades determinadas para o aplicativo Orienta UnB. As funcionalidades serão detalhadas na seção 5.4.

5.3.2.2 Regras de negócio

Durante a determinação da visão do produto, foram definidas regras para estabelecer o escopo para o aplicativo, estas que estão conforme o Anexo — C. Segue uma descrição detalhada destes aspectos:

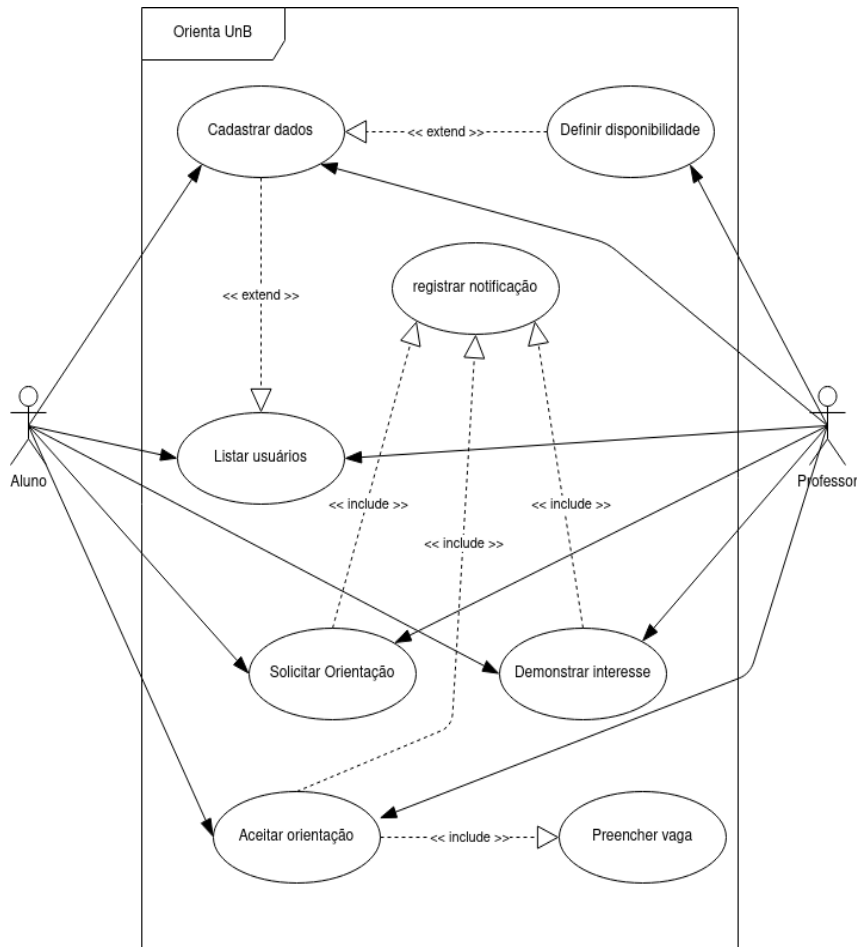
- É um aplicativo de combinação de interesses, que realiza cálculos para estabelecer o nível de compatibilidade entre os usuários;
- É um aplicativo exclusivamente *mobile*, disponível para as plataformas Android e iOS;
- O aplicativo não é uma rede social, entretanto possui comportamentos semelhantes;
- O aplicativo não possui um *chat* para os usuários;
- O aplicativo não realiza a exportação de dados dos participantes para nenhum meio externo;
- O aplicativo não realiza o preenchimento de formulário para preenchimento no SI-GAA ou Matrícula Web;
- O aplicativo não realiza interações de cadastro das modalidades TCC, PIBIC, PIBID ou PIBITI.

Todas as tarefas da *Lean Inception* possuem um valor significativo para poder avançar em cada etapa. Foi seguido conforme o planejado, a fim de poder alcançar um MVP com um grau de refinamento suficiente para alcançar os objetivos. Além disso, segundo Caroli (2017), é possível recomeçar, pois o MVP é incremental.

5.3.2.3 Diagrama de caso de uso

Em decorrência dos tipos de perfis, existem características do sistema exclusivas para cada tipo. Pode ser descrita pelo diagrama de caso de uso a seguir:

Figura 8 – Diagrama de Caso de Uso



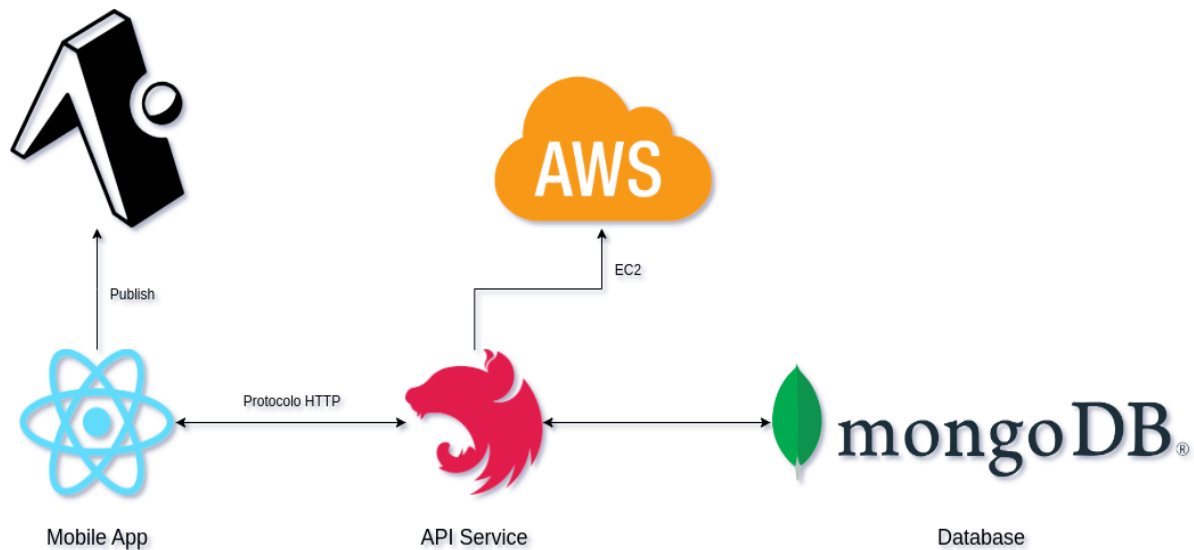
Fonte: Autor

Para o aplicativo Orienta UnB, foram planejados dois tipos de usuário, professor e aluno, em que algumas das funcionalidades e disponibilidades são citadas a seguir. (i) Para o usuário professor, todas as funcionalidades estão habilitadas e é de sua exclusividade o controle da quantidade de vagas disponíveis para orientação. Além disso, possui informações extras no formulário de dados, para serem visualizadas ao ocorrer um *match*; (ii) Para o usuário aluno, haverá a disponibilidade de todas as funcionalidades do sistema, exceto o controle da quantidade de vagas. Na seção 5.4, os detalhes de cada funcionalidade são apresentados, além de corresponder com o protótipo de alta fidelidade na subseção 5.3.5.

5.3.3 Arquitetura de Software

A Figura 9 descreve o comportamento da arquitetura do aplicativo Orienta UnB:

Figura 9 – Arquitetura geral Orienta UnB



Fonte: Autor

A arquitetura do aplicativo Orienta UnB é subdividida em duas frentes: (i) *Client side*, definido pelo *front-end*, que se trata da interface de visualização e realiza requisições para a API (*Application Programming Interface*), ao utilizar o protocolo HTTP (*Hyper-text Transfer Protocol*); (ii) *Server side*, que se refere ao *back-end* e que detém de uma arquitetura monolítica, *API service*.

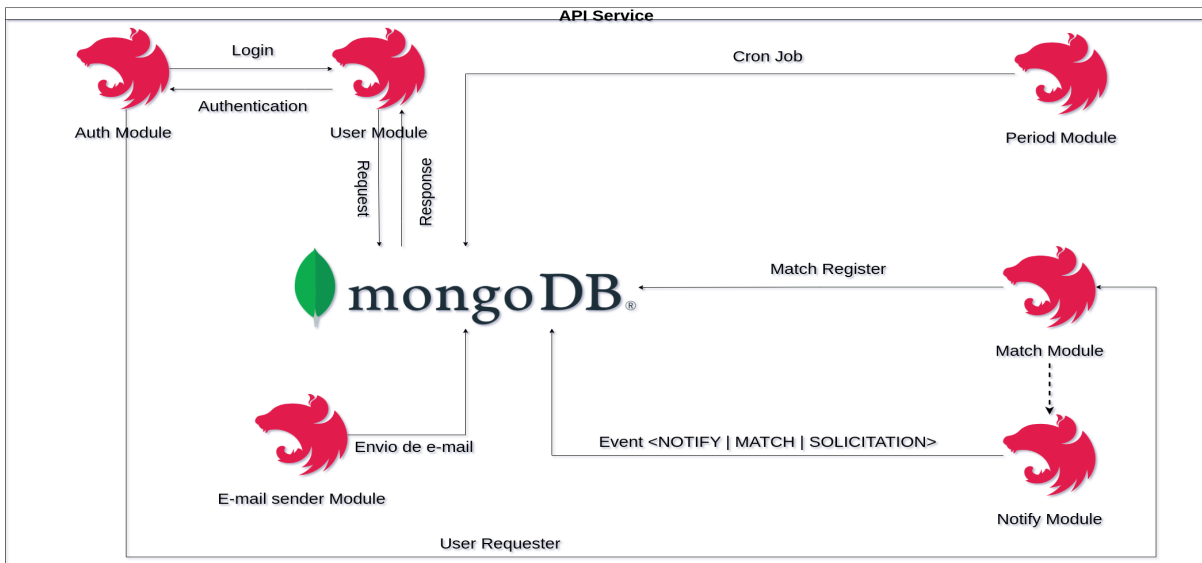
A *API Service* é o conjunto de funcionalidades essenciais para o funcionamento do sistema. Portanto, todos os comportamentos visuais que são demonstrados pela interface estão vinculados às regras de negócio que foram implementadas para esta solução, valendo ressaltar a principal ação, que é a de recomendações baseadas em conteúdo 2.2 durante o acesso da listagem de perfis.

Majoritariamente a efetivação das tarefas internas da solução é feitas partindo de rotas (*endpoints*) da API. Entretanto, existem funcionalidades assíncronas que ocorrem para a criação de notificações designadas aos usuários, que são realizadas por meio de eventos. Estas notificações são disparadas exclusivamente para tratar os aspectos de ações da *feature* de *Match*.

O modelo adotado para a arquitetura atende aos requisitos para o desenvolvimento do sistema, produzindo resultados satisfatórios, que por sua vez eliminam complexidades relacionadas à integração e ao uso de serviço de terceiros, podendo concluir que as tomadas de decisões, incluindo padrões de projeto e arquitetura, contribuiram para a manutenibilidade e escalabilidade do serviço.

A Figura 10 descreve o comportamento da arquitetura interna da *API Service* do aplicativo Orienta UnB:

Figura 10 – Arquitetura interna Orienta UnB



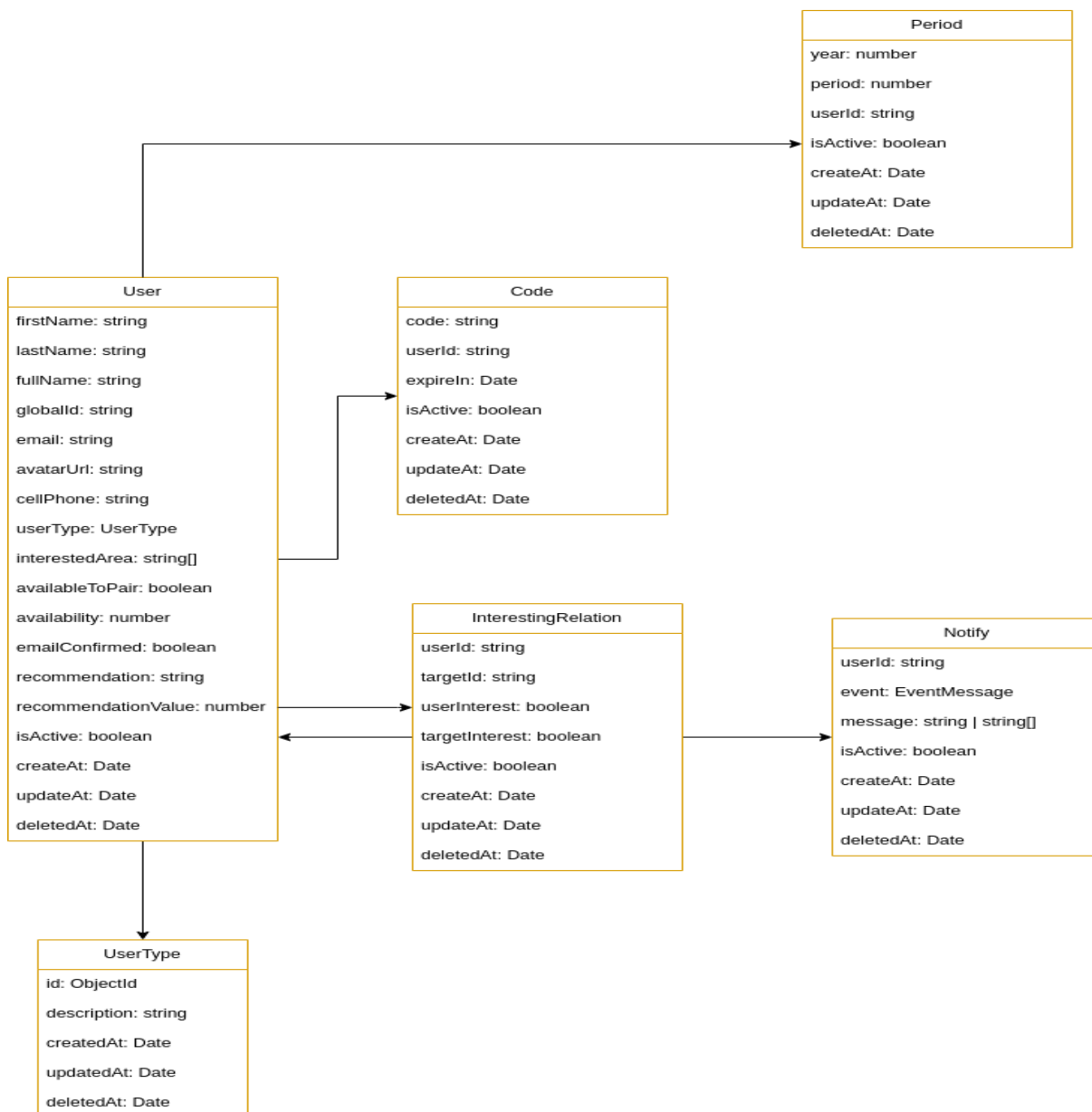
Fonte: Autor

Foram desenvolvidos seis módulos efetivos para este serviço. Cada um possui responsabilidades únicas, tendo em vista que todas as ações geradas são registradas no banco de dados do serviço. Dessa forma, permite o controle efetivo e autêntico dos resultados das tarefas, além de possuir segurança significativa, provida por meio da criptografia dos dados.

5.3.4 Modelo de dados

O aplicativo Orienta UnB possui um banco de dados não-relacional, logo, não faz uso de MER (Modelo Entidade Relacionamento) ou DER (Diagrama Entidade Relacionamento). Todavia, a representação esquemática do banco de dados pode ser descrita na Figura 11.

Figura 11 – Modelo esquemático do banco de dados



Fonte: Autor

O modelo de dados representa a abstração realizada para as principais funcionalidades do sistema, que são: cadastro do usuário conforme o perfil, professor ou aluno, além da criação da ação de demonstrar interesse, e quando há interesses mútuos (*Match*). Por meio desse modelo esquemático, também é possível realizar os cálculos para realizar a análise de compatibilidade, conforme a seção 2.2.

5.3.5 Protótipo

A fim de validar a ideia, foi desenvolvido um protótipo de alta fidelidade para o presente trabalho, em que o objetivo é demonstrar as interações e os recursos do aplicativo,

sendo uma simulação das expectativas desejadas.

Os principais aspectos serão apontados por esta Seção, todavia, uma visão detalhada do protótipo encontra-se no Apêndice — B.

Para o primeiro acesso ao aplicativo, após realizar o cadastro, é apresentado o termo de política e privacidade referente à utilização, que pode ser consultado no Apêndice — C. Após sua aceitação, o fluxo inicial é iniciado. Uma vez já realizado o cadastro, o termo não irá ser apresentado novamente.

5.3.5.1 Cadastro de usuário

A tela de cadastro é o ponto onde o perfil do usuário é definido. A diferenciação ocorre durante a validação do cadastro, utilizando o e-mail institucional e verificando o domínio específico. Dado que professores e alunos possuem domínios distintos, essa característica possibilita a distinção entre os tipos de perfis.

Dando seguimento ao cadastro, o usuário deve selecionar suas áreas de interesse, as quais serão utilizadas como base para realizar as recomendações, utilizando Filtragem Baseada em Conteúdo. Ao final do cadastro, os usuários são redirecionados para a listagem de pessoas, onde poderão utilizar os recursos do aplicativo. Vale ressaltar que a visão do professor e do alunos é distinta. O primeiro consegue visualizar os alunos que possuem interesse em realizar o TCC sozinho ou em dupla. O segundo visualiza os professores e os alunos que desejam realizar trabalhos em dupla. Ademais, para ambos, na tela de listagem de perfis, é possível visualizar a compatibilidade por meio dos cálculos do algoritmo de recomendação.

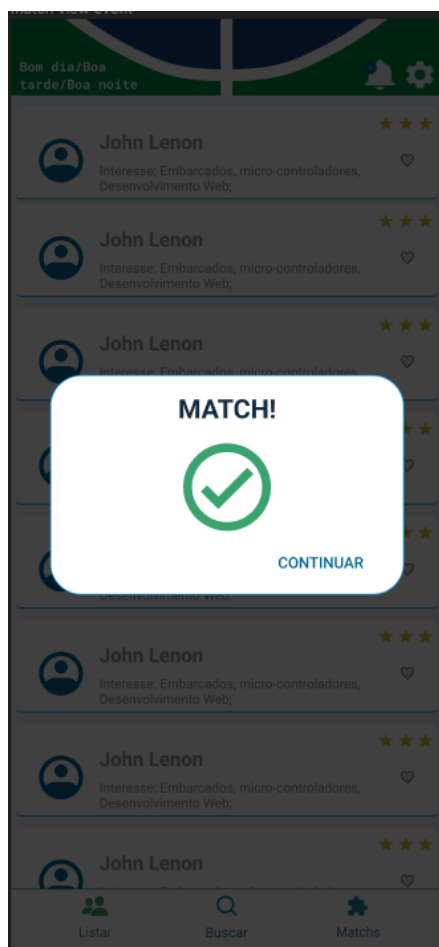
5.3.5.2 Demonstração de interesse mútuo

Conforme ocorrem as interações dos usuários durante o uso do aplicativo, quando há uma demonstração de interesse mútuo, isto é, interesse entre professor e aluno, é apresentada uma aba (*dialog*) informando o evento de correspondência entre os participantes.

Para realizar a demonstração de interesse, basta clicar no ícone de coração na listagem de pessoas e, assim, o sistema irá notificar o usuário correspondente à ação. Da mesma forma, para o participante que recebeu a demonstração de interesse, é possível executar a mesma ação para quem se interessou.

A Figura 12 apresenta a visão do aluno quando ocorre uma ação de interesse mútuo:

Figura 12 – Match entre participantes



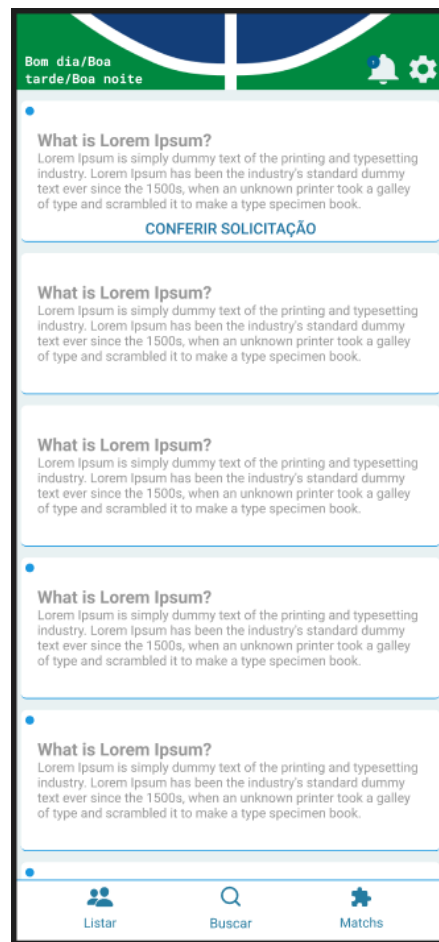
Fonte: Autor

5.3.5.3 Lista de notificações

Os eventos de notificações correspondem às mensagens informativas para a comunicação do sistema com o usuário. Existem notificações que possuem características de ações, ou seja, o usuário pode executar interações por meio delas, conforme a Figura 13.

As ações encontram-se na aba "Matches" e a confirmação de uma orientação só é possível mediante uma solicitação de orientação. Para acontecer uma solicitação, os usuários devem realizar a ação de "demonstrar interesse". A seguir, uma sequência descritiva das ações: solicitação de orientação e confirmação de orientação, com as Figuras 14 e 15, respectivamente.

Figura 13 – Lista de notificações

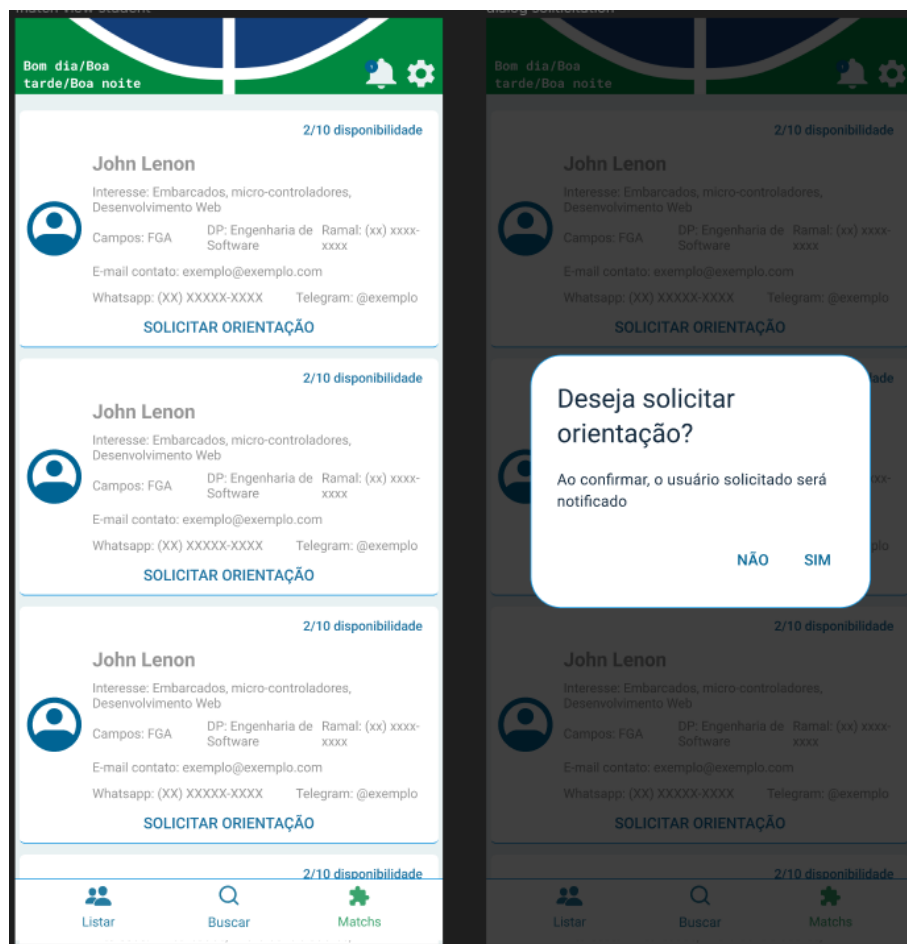


Fonte: Autor

5.3.5.4 Solicitar orientação

Uma solicitação de orientação ocorre apenas com a existência de uma correspondência entre os usuários.

Figura 14 – Solicitar orientação

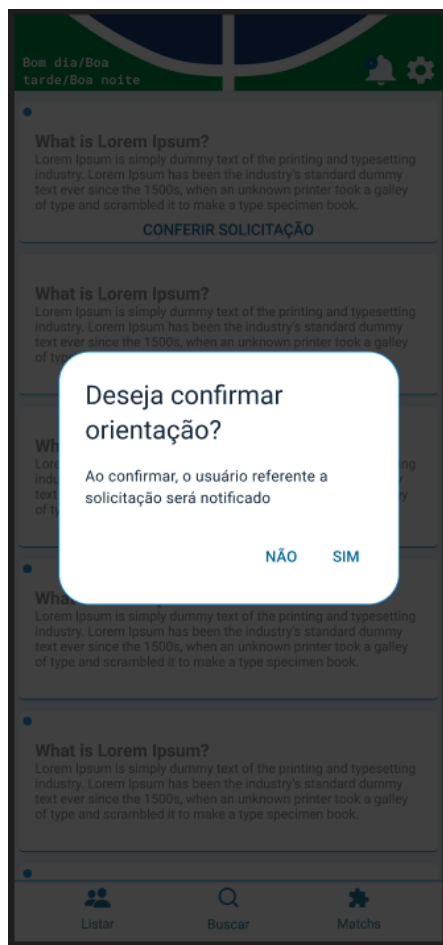


Fonte: Autor

5.3.5.5 Confirmar solicitação

Após realizar a solicitação, o usuário solicitante depende da resposta da ação do outro que foi solicitado, portanto, esta é uma ação para o usuário solicitado. Uma vez que o solicitado aceita a orientação, este valor será contabilizado e incrementado à disponibilidade do professor, atualizando o valor demonstrado para as vagas.

Figura 15 – Confirmar orientação



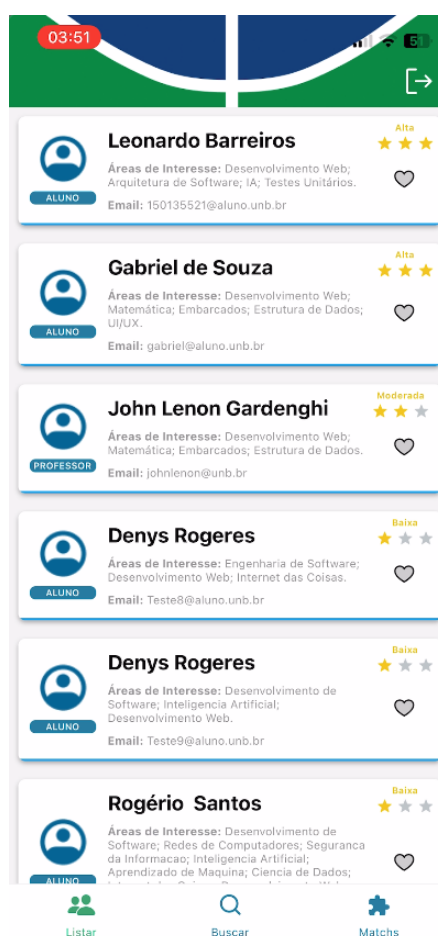
Fonte: Autor

5.3.5.6 Sistema de recomendação

Inicialmente, para realizar esta etapa, foi necessário implementar funcionalidades que capturassem o contexto do usuário da requisição. Quando se faz o *login*, cria-se um *token JWT (JSON Web Tokens)*, que por sua vez, possui informações criptografadas do usuário. Essencialmente, esse *token* possui as áreas de interesse do usuário após efetuar o *login*.

No momento em que a requisição de listagem dos usuários no *app* é realizada, é levado em consideração o critério do tipo de perfil do acessante para realizar a filtragem dos dados. Após essa configuração, é feito, para cada um dos registros, o cálculo de recomendação. Este cálculo consiste em comparar as áreas de interesses do acessante com as áreas de interesses de cada um dos usuários listados, criando-se um valor probabilístico de 0 (zero) até 1 (um). Ao final, o valor é traduzido para um formato com um teor de compreensão amigável para o usuário acessante. Dessa forma, o valor representado por estrelas, reflete o grau de indicação da recomendação ou compatibilidade entre os usuários, variando de 1 (baixo), 2 (moderado) a 3 (alto), conforme demonstrado na Figura 16.

Figura 16 – Tela de recomendação



Fonte: Autor

Figura 17 – Exemplo de usuário recomendado

```
{
  "id": "656bed2dc5eeba12bf402631",
  "firstName": "Denys",
  "lastName": "Rogeres",
  "fullName": "Denys Rogeres",
  "email": "denysrogeres@aluno.unb.br",
  "cellPhone": "61996541556",
  "userType": {
    "isActive": true,
    "createdAt": "2023-10-18T22:51:19.178Z",
    "updatedAt": "2023-10-18T22:51:19.178Z",
    "deletedAt": null,
    "description": "STUDENT",
    "_id": "65306167e849418abf270cd0"
  },
  "interestedArea": [
    "Desenvolvimento Web",
    "Arquitetura de Software",
    "IA",
    "Testes Unitários",
    "UI/UX",
    "Embarcados",
    "Matemática"
  ],
  "availableToPair": true,
  "emailConfirmed": false,
  "isActive": true,
  "recommendation": "Recommended",
  "recommendationValue": 1
},
```

Fonte: Autor

Figura 18 – Exemplo de usuário pouco recomendado

```
{
  "id": "656c9d18e64c62c24e1f9765",
  "firstName": "Gabriel",
  "lastName": "de Souza",
  "fullName": "Gabriel de Souza",
  "email": "gabriel@aluno.unb.br",
  "cellPhone": "61996541556",
  "userType": {
    "isActive": true,
    "createdAt": "2023-10-18T22:51:19.178Z",
    "updatedAt": "2023-10-18T22:51:19.178Z",
    "deletedAt": null,
    "description": "STUDENT",
    "_id": "65306167e849418abf270cd0"
  },
  "interestedArea": [
    "Desenvolvimento Web",
    "Matemática",
    "Embarcados",
    "Estrutura de Dados",
    "UI/UX"
  ],
  "availableToPair": true,
  "emailConfirmed": false,
  "isActive": true,
  "recommendation": "Not Recommended",
  "recommendationValue": 0.25
},
```

Fonte: Autor


```

calculateSimilarity(
  currentUser: UserModelView,
  user: GetUserModelView,
): number {
  const intersection = currentUser.interestedArea.filter((value) =>
    user.interestedArea.includes(value),
  );

  const similarity =
    intersection.length / currentUser.interestedArea.length;

  return similarity;
}
}

```

O algoritmo para calcular a similaridade foi definido como uma classe que executa uma ação, conforme o padrão de arquitetura definido, *Layer Architecture*. Portanto, o método *execute* recebe dois parâmetros: *'currentUser'*, do tipo *'UserModelView'*, e *users*, uma lista de objetos do tipo *'GetUserModelView'*. Ao final retornar a lista de elementos que representam os usuários.

A função itera sobre a lista de usuários fornecida, usando o método *'forEach'*. Para cada usuário, calcula um valor de similaridade, por meio da função *'calculateSimilarity'*, que é uma medida da similaridade entre o usuário atual (*currentUser*) e o usuário na iteração atual.

Com base no valor de similaridade calculado, o código define o nível de recomendação (*recommendation*) para esse usuário. Se o índice de similaridade estiver dentro da faixa de 0 a 0.3, o nível de recomendação é considerado baixo (uma estrela); se for superior a 0.3 e inferior a 0.5, é classificado como moderado (duas estrelas); e se ultrapassar 0.5, é considerado alto (três estrelas). Por fim, a lista de usuários é classificada com base em seus valores de recomendação, do maior para o menor, conforme ilustrado na Figura 16.

A função *'calculateSimilarity'* recebe dois parâmetros: o *'currentUser'*, do tipo *'UserModelView'*, e *'user'*, do tipo *'GetUserModelView'*. Essa função calcula similaridade entre os usuários da iteração com base em suas áreas de interesse.

Primeiro, a função identifica a interseção das áreas de interesse entre o usuário atual e o usuário fornecido. Isso é feito filtrando os elementos da lista de áreas de interesse do usuário atual e verificando se eles também estão presentes na lista de áreas de interesse do usuário fornecido. A interseção resultante contém os elementos que estão presentes em ambas as listas de áreas de interesse.

Em seguida, a função calcula a similaridade dividindo o número de elementos na interseção pelo número total de elementos na lista de áreas de interesse do usuário atual. Isso fornece uma medida da proporção de áreas de interesse compartilhadas entre os dois usuários em relação ao total de áreas de interesse do usuário atual.

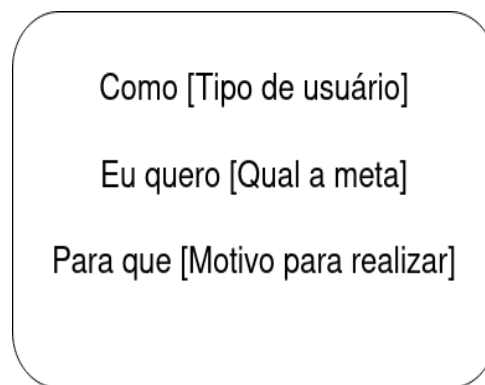
Em resumo, essa função compara as áreas de interesse do usuário atual com as áreas de interesse de outro usuário e retorna um valor que representa a similaridade entre essas áreas de interesse. Quanto maior o valor retornado, maior é a similaridade entre os interesses dos dois usuários.

5.4 Detalhamentos dos requisitos

Visto na seção 4.3, foi adotado o modelo de metodologia de desenvolvimento *Scrum* associado ao *Kanban* para gerenciar e organizar o projeto. A partir da *Lean Inception*, o *Product Backlog* foi elaborado seguindo a priorização estabelecida pelas ondas do sequenciador (CAROLI, 2017).

Segundo Alsaadi e Saeedi (2022), uma história de usuário é definida conforme ilustrado na Figura 20:

Figura 20 – Estrutura de história de usuário



Fonte: Adaptado de Alsaadi e Saeedi (2022).

Antes de detalhar as *features*, o processo de elicitación de requisitos *Lean Inception* é adaptado para a metodologia de desenvolvimento 4.3 *Scrum*. Os requisitos são reformulados em forma de história de usuário (*User Stories*), definida por meio do sequenciador construído para criar o MVP. Essa reescrita permite a compreensão dos requisitos, ao prover, por meio de padrões de escrita, um entendimento objetivo e simples das metas a serem alcançadas e expõe os principais autores (*stakeholders*) impactados pela *feature*, além de realizar o encapsulamento por meio de sentenças (Para quem? Para quê? Por quê?) (ALSAADI; SAEEDI, 2022).

No modelo *Scrum*, são planejados ciclos de desenvolvimento, estes nomeados como *sprint*. Uma *sprint* dura em média duas semanas, padrão atribuído e idealizado para entregas contínuas (ALSAADI; SAEEDI, 2022).

Conforme o modelo de Caroli (2017), cada *sprint* é representada por cada onda do sequenciador gerado para a definição do MVP. Dessa forma, realizando-se as devidas adaptações, as histórias de usuário para o aplicativo Orienta UnB são representadas pelos Quadros 8 e 9

Quadro 8 – Backlog do Produto: parte 1

ID	História de usuário	feature	Sprint
US1	Como usuário, eu quero realizar <i>login/logout</i> , para que possa acessar o aplicativo.	<i>Login</i>	<i>Sprint 0</i>
US2	Como sistema, eu quero definir os tipos de perfis, para que possa separar a visão entre aluno e professor.	<i>Login</i>	<i>Sprint 0</i>
US3	Como usuário, eu quero cadastrar os dados por meio de um formulário, para que possa registrar informações necessárias para o sistema.	Perfil	<i>Sprint 0</i>
US4	Como usuário, desejo editar informações sobre o perfil, para que possa atualizar as informações.	Perfil	<i>Sprint 1</i>
US5	Como usuário, desejo inserir múltiplas áreas de interesse, para que possa informar o sistema.	Perfil	<i>Sprint 1</i>
US6	Como aluno, desejo visualizar a listagem de professores, para que possa interagir com os participantes.	Análise de dados	<i>Sprint 1</i>
US7	Como professor, desejo visualizar a listagem de alunos, para que possa interagir com os participantes.	Análise de dados	<i>Sprint 2</i>
US8	Como usuário, desejo realizar a ação de interesse, para que possa criar conexão com um participante.	Ações	<i>Sprint 2</i>
US9	Como professor, desejo definir quantidade de disponibilidade, para que possa ser preenchida pelos alunos.	Sistema	<i>Sprint 2</i>
US10	Como sistema, desejo realizar <i>match</i> entre os candidatos, para que possa vincular os participantes.	Ações	<i>Sprint 3</i>

Fonte: Autor

Quadro 9 – Backlog do Produto: parte 2

ID	História de usuário	feature	Sprint
US11	Como professor, desejo abrir vagas para trabalhos em dupla, para que possa permitir mais de um participante no mesmo projeto.	Ações	<i>Sprint 3</i>
US12	Como sistema, desejo possuir uma aba para os <i>matches</i> , para que possa informar aos usuários as interações realizadas.	Sistema	<i>Sprint 3</i>
US13	Como professor/aluno, desejo visualizar candidatos mais compatíveis primeiro, para que eu possa saber os usuários mais compatíveis.	Análise de dados	<i>Sprint 4</i>
US14	Como aluno, desejo poder localizar potenciais parceiros para o Trabalho de Conclusão de Curso, para que possa realizar projeto de maneira colaborativa.	Ação	<i>Sprint 4</i>
US15	Como sistema, desejo fazer envio de notificações, para que possa informar o usuário de qualquer evento importante	Sistema.	<i>Sprint 4</i>

Fonte: Autor

5.5 Resumo do Capítulo

Este capítulo tem o intuito de detalhar a proposta do projeto Orienta UnB em relação a cada uma das etapas para a construção do aplicativo.

Acerca das etapas para o desenvolvimento, foram detalhados os principais aspectos e recursos, com base nos conceitos da literatura, seguido da demonstração do modelo de comportamento de uso, modelo arquitetural, modelo esquemático do banco, modelo de comportamento de funcionalidades e, por último, o protótipo de alta fidelidade, seguido pelo detalhamento dos requisitos e priorização dos ciclos de desenvolvimento do projeto.

6 Análise de Resultados

O presente capítulo tem como propósito abordar a análise dos resultados obtidos no contexto do desenvolvimento do aplicativo Orienta UnB, em que os detalhes foram expostos no Capítulo 5. A análise de resultados foi conduzida seguindo o protocolo de Pesquisa-Ação apresentado no Capítulo 4. Este protocolo é composto pelas seguintes fases: Exploratória, Planejamento, Ação e Avaliação. Ao final, é apresentado um resumo do capítulo.

6.1 Pesquisa-Ação

Conforme descrito na Metodologia de Análise de Resultados (seção 4.4), este trabalho segue as fases do protocolo de Pesquisa-Ação. Inicialmente, ocorreu a Etapa Exploratória, na qual foram utilizados formulários para alunos e professores, a fim de coletar informações acerca do contexto. Logo em sequência, dando continuidade a etapa anterior, ocorreu a Etapa de Planejamento, na qual foi detalhado o planejamento para realizar o desenvolvimento da aplicação. Seguido pela Etapa de Ação, que consistiu no desenvolvimento da aplicação, tendo como base as etapas anteriores. Por fim, há as considerações feitas na Etapa de Avaliação.

6.2 Etapa Exploratória

Essa etapa marca o início do processo de pesquisa, em que foram coletados dados e informações essenciais para a compreensão do problema, e a realização de um diagnóstico da situação. A coleta de dados foi realizada por meio da aplicação de formulários.

Assim, o objetivo da coleta de dados via formulário foi reunir informações relacionadas ao processo de Trabalho de Conclusão de Curso, abordando o tempo de espera dos alunos para orientação, a modalidade de TCC escolhida e as dificuldades encontradas na busca por um orientador.

Para alcançar esse propósito, foram disponibilizados dois formulários desenvolvidos na plataforma Google *Forms*, um destinado aos professores e outro aos alunos. Os formulários permaneceram abertos por duas semanas, resultando em 31 respostas para o formulário de alunos e 11 respostas para o formulário de professores.

6.2.1 Questionário Aluno

Para entender a situação que os alunos da FGA se encontram, foi criado um questionário com algumas perguntas, obrigatórias ou não. Vale ressaltar que o formulário foi respondido por alunos do curso de Engenharia de Software da UnB.

O formulário foi dividido em três seções, sendo elas:

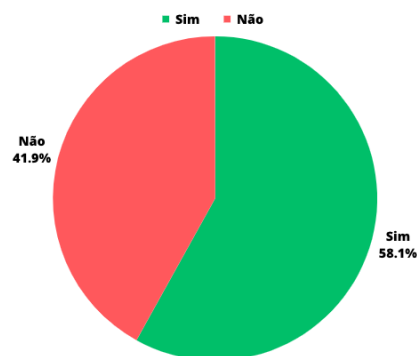
1. Estado Atual de Orientação;
2. Procurando Orientação;
3. Orientação Definida.

6.2.1.1 Estado Atual de Orientação

Essa seção serviu para identificar quantos alunos já estão com o orientador para o TCC definido. Neste caso, foi feita a seguinte pergunta:

- Você já definiu seu orientador para o TCC?

Figura 21 – Gráfico ilustrativo do resultado de respostas da 1ª seção.



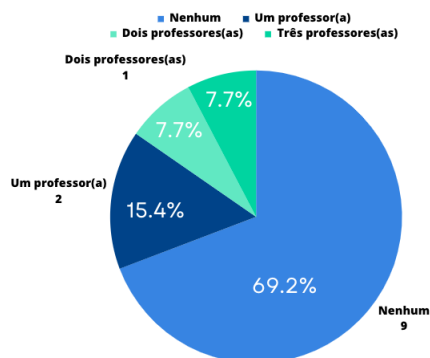
Fonte: Autor

6.2.1.2 Procurando Orientação

Esta seção tem como objetivo a coleta de dados referentes aos alunos que ainda não conseguiram encontrar um orientador para o TCC. Foram abordadas algumas perguntas com o intuito de compreender melhor a situação dos alunos. Abaixo, estão as questões formuladas, seguidas dos resultados representados visualmente:

1. Você já começou a procurar um professor para orientação do seu TCC? Se sim, quantos professores você procurou até o momento?

Figura 22 – Gráfico ilustrativo do resultado da pergunta 1 da 2ª seção.



Fonte: Autor

2. Caso já tenha entrado em contato com algum professor buscando orientação, qual meio de comunicação você utilizou?

Figura 23 – Gráfico ilustrativo do resultado da pergunta 2 da 2ª seção.



Fonte: Autor

3. Caso já esteja na procura por orientador, qual está sendo sua maior dificuldade?

Figura 24 – Comentários sobre a pergunta 3 da 2ª seção.

Escolha do tema

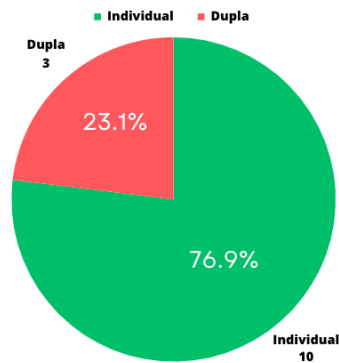
Achar as áreas e disponibilidades dos professores

Disponibilidade do orientador para o semestre pretendido

Fonte: Autor

4. Em qual formato você deseja fazer seu TCC?

Figura 25 – Gráfico ilustrativo do resultado da pergunta 4 da 2ª seção.



Fonte: Autor

5. Se você pretende realizar o TCC em dupla, você já escolheu o seu parceiro(a)? Em caso afirmativo, como vocês se conheceram?

Figura 26 – Comentários sobre a pergunta 5 da 2ª seção.

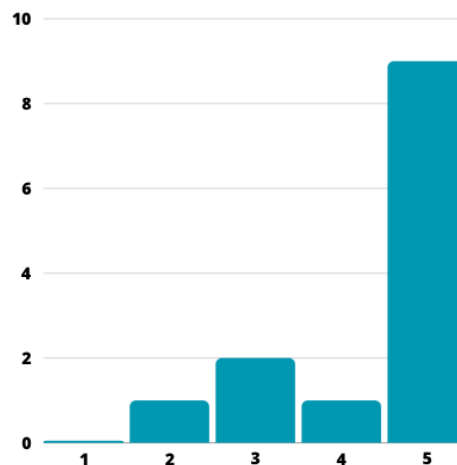
Sim. A gente se conheceu cursando disciplinas juntos.

Sim, disciplina de EPS

Fonte: Autor

6. Em uma escala de 1 a 5, qual a chance de você utilizar uma plataforma baseada em sistema de recomendação para conectar alunos e professores com interesses em comum, visando facilitar projetos acadêmicos, como o TCC?

Figura 27 – Gráfico ilustrativo do resultado da pergunta 6 da 2ª seção.



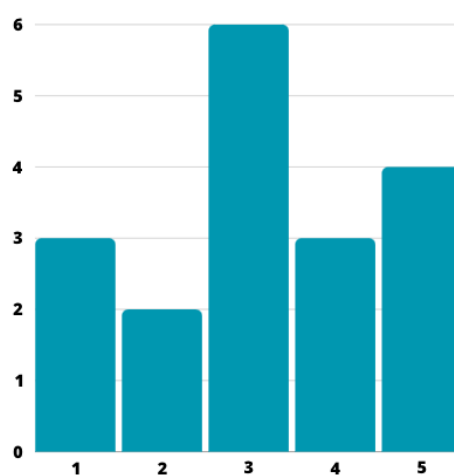
Fonte: Autor

6.2.1.3 Orientação Definida

Esta seção, foi direcionada para expor a coleta de dados que relaciona aos alunos que já encontraram um orientador para o Trabalho de Conclusão de Curso. Nela buscou-se mapear o tempo de espera dos alunos para orientação, a modalidade de TCC escolhida e as dificuldades encontradas na busca por um orientador. A seguir, são apresentadas as perguntas indagadas, acompanhadas da representação visual dos resultados obtidos.

1. Em uma escala de 1 a 5, qual nível de dificuldade você teve para encontrar um orientador?

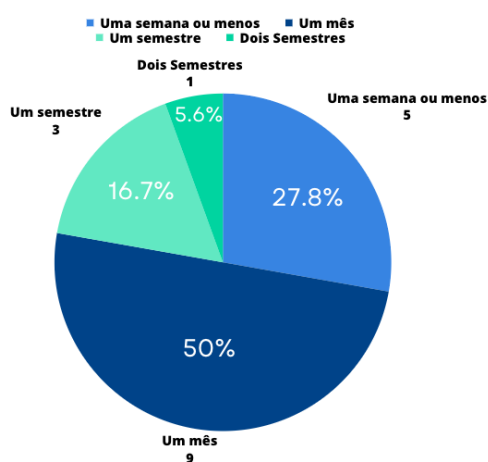
Figura 28 – Gráfico ilustrativo do resultado da pergunta 1 da 3ª seção.



Fonte: Autor

2. Quanto tempo levou para encontrar um orientador?

Figura 29 – Gráfico ilustrativo do resultado da pergunta 1 da 3ª seção.



Fonte: Autor

3. Qual a maior dificuldade que você sentiu na procura por um orientador?

Figura 30 – Comentários sobre a pergunta 3 da 3ª seção.

Encontrar um orientador disponível
Algum disponível
Encontrar alguém disponível
Achar um professor disponível e um tema interessante
falta de contato com os professores do curso e saber como pedir para alguém me orientar de forma muito informal foram minhas principais dificuldades
Professor especialista no tema.
Entrar em contato para mostrar minha proposta inicial
Disponibilidade dos professores e escopo dos projetos e suas áreas.
Encontrar um orientador para o meu tema proposta
Não tive dificuldade. Aproveitei que eu já estava em um projeto de pesquisa com ele
Quase nenhum professor tinha vagas disponíveis
Professor com área de estudo em comum
A maior dificuldade é saber se eles estão disponíveis e quais temas eles possuem ou aceitam para orientação
No meu caso foi relativamente simples, mas isso porque eu estava disposto a esperar por mais um semestre. Se não fosse por isso, seria bem mais complicado.
como contactar um professor

Fonte: Autor

4. Quantos professores você procurou para solicitar orientação?

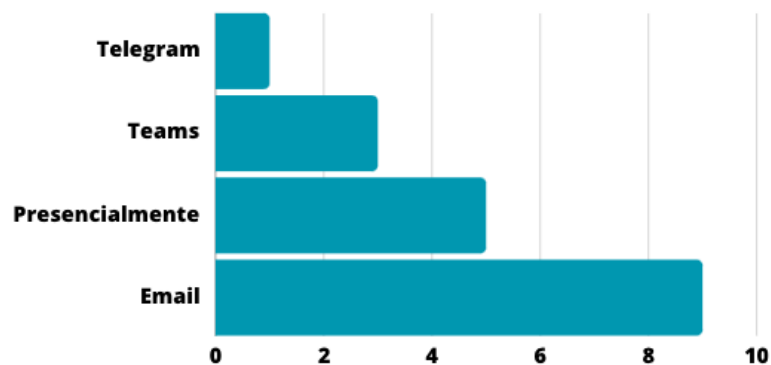
Figura 31 – Gráfico ilustrativo do resultado da pergunta 4 da 3ª seção.



Fonte: Autor

5. Qual meio de comunicação você utilizou para encontrar seu orientador?

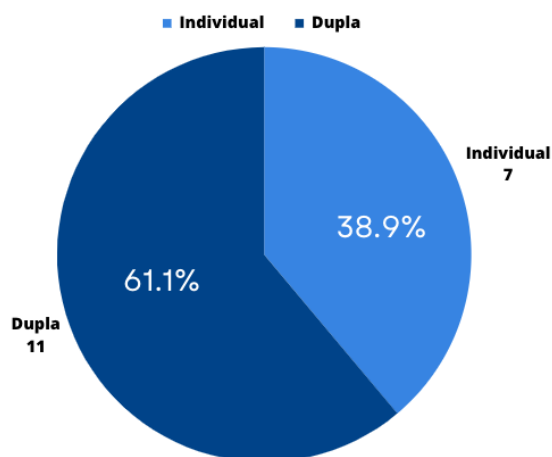
Figura 32 – Gráfico ilustrativo do resultado da pergunta 5 da 3ª seção.



Fonte: Autor

6. De qual forma esta fazendo seu TCC?

Figura 33 – Gráfico ilustrativo do resultado da pergunta 6 da 3ª seção.



Fonte: Autor

7. Caso esteja fazendo ou tenha feito o TCC em dupla, como você conheceu seu parceiro(a) para o TCC?

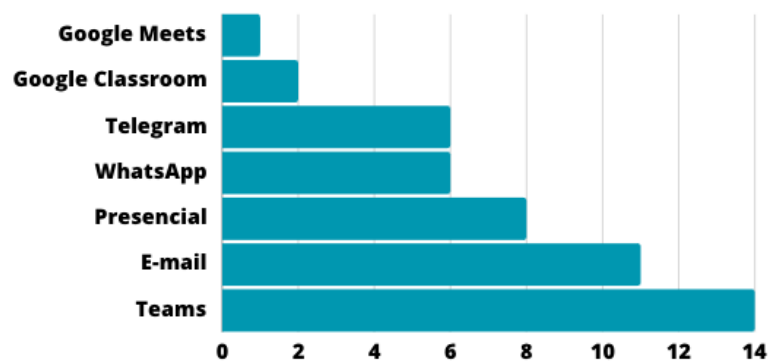
Figura 34 – Comentários sobre a pergunta 7 da 3ª seção.

Des do primeiro semestre
No ensino médio
Entrou na faculdade no mesmo semestre que eu.
É minha colega desde o início do curso
Em aula
Indicação de amigos
Meu amigo da faculdade
Amigo e Monitor
Conheci no segundo semestre da faculdade através de amigos em comum
com o orientador

Fonte: Autor

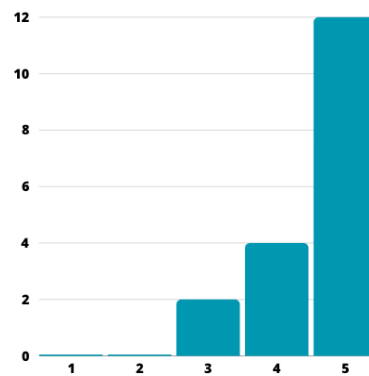
8. Quais os meios de comunicação utilizado durante a orientação?

Figura 35 – Gráfico ilustrativo do resultado da pergunta 8 da 3ª seção.



Fonte: Autor

9. Se durante o processo de escolha de um orientador, você tivesse a oportunidade de utilizar uma plataforma baseada em sistema de recomendação para conectar alunos e professores com interesses em comum, facilitando projetos acadêmicos, como o TCC, qual seria a probabilidade de você usar essa plataforma (Em uma escala de 1 a 5)?

Figura 36 – Gráfico ilustrativo do resultado da pergunta 8 da 3^o seção.

Fonte: Autor

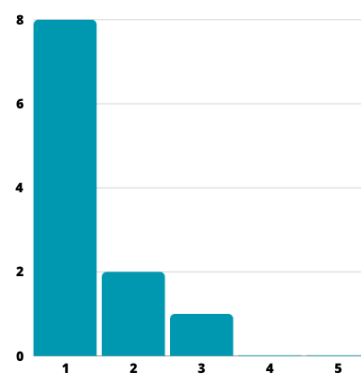
6.2.2 Questionário Professor

Com o objetivo de avaliar o contexto enfrentado pelos professores da FGA, foi aplicado um questionário abrangendo questões tanto obrigatórias quanto opcionais relacionadas ao processo de orientação. Vale destacar que participaram da pesquisa 10 professores da Engenharia de Software e um professor da Engenharia Eletrônica.

Abaixo, são apresentadas as perguntas formuladas, acompanhadas da representação visual dos resultados obtidos.

1. Em uma escala de 1 a 5, qual a dificuldade para encontrar alunos para orientar?

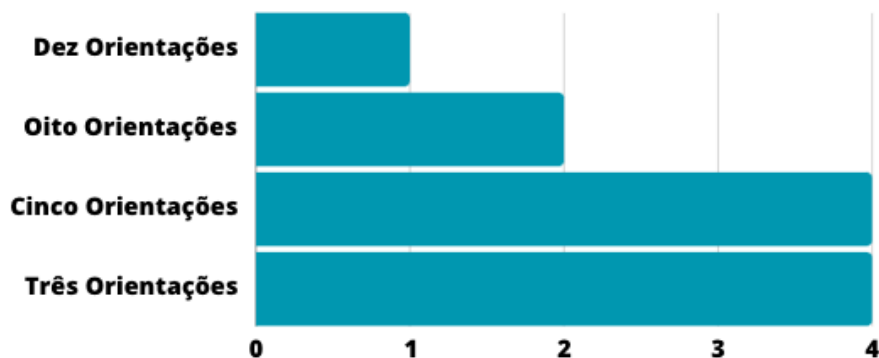
Figura 37 – Gráfico ilustrativo do resultado da pergunta 1.



Fonte: Autor

2. Em média, quantas orientações você costuma realizar por semestre, levando em conta que as duplas são consideradas como uma única orientação?

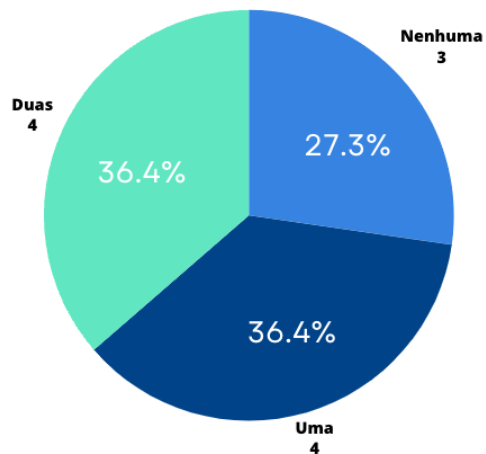
Figura 38 – Gráfico ilustrativo do resultado da pergunta 2.



Fonte: Autor

3. Dentro dessa média de orientações por semestre, quantas são em dupla?

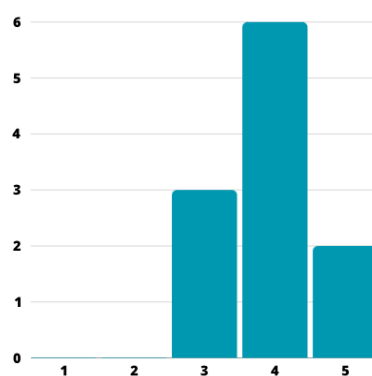
Figura 39 – Gráfico ilustrativo do resultado da pergunta 3.



Fonte: Autor

4. Em uma escala de 1 a 5, qual o grau de dificuldade para conduzir todas essas orientações durante o semestre?

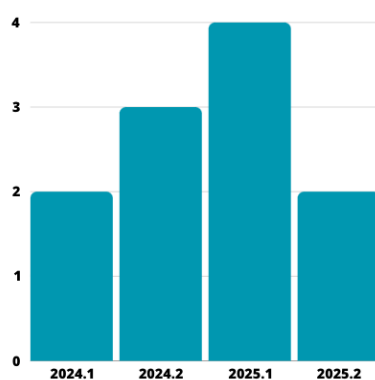
Figura 40 – Gráfico ilustrativo do resultado da pergunta 4.



Fonte: Autor

5. Considerando a ordem de espera na fila de orientação, a partir de qual semestre você tem disponibilidade para aceitar novos orientandos?

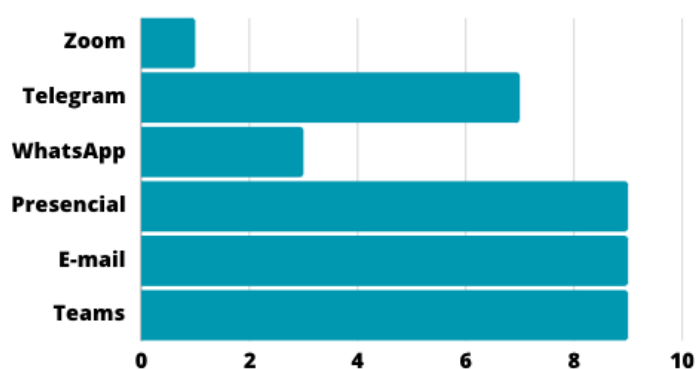
Figura 41 – Gráfico ilustrativo do resultado da pergunta 5.



Fonte: Autor

6. Quais os meios de comunicação utilizado durante a orientação?

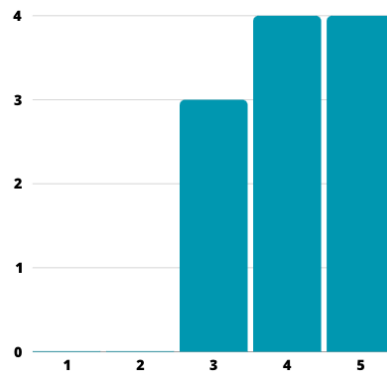
Figura 42 – Gráfico ilustrativo do resultado da pergunta 6.



Fonte: Autor

7. Se você tivesse a oportunidade de utilizar uma plataforma baseada em sistema de recomendação para conectar alunos com interesses na sua área de pesquisa/interesse, facilitando projetos acadêmicos, como o TCC, qual seria a probabilidade de você usar essa plataforma (Em uma escala de 1 a 5)?

Figura 43 – Gráfico ilustrativo do resultado da pergunta 7.



Fonte: Autor

6.3 Etapa Planejamento

Como resultado da fase anterior, identificou-se que os alunos enfrentam desafios significativos, destacando-se a dificuldade em encontrar um orientador disponível, localizar um professor com a área de interesse desejada e estabelecer contato efetivo com os docentes. Nesse contexto, observou-se que, entre os alunos que já definiram seus orientadores, 66,7% precisaram contatar pelo menos três professores antes de encontrar a orientação desejada. Notavelmente, a comunicação predominante foi por e-mail, com uma média de um mês ou mais para concluir o processo de busca por orientação. Além disso, constatou-se que uma parcela significativa dos alunos que ainda não iniciaram o TCC planeja realizá-lo de forma individual. No entanto, ao analisar os estudantes já engajados no processo de elaboração do TCC, mais da metade optou pelo modelo de dupla.

Em relação aos professores, constatou-se que todos eles relataram ter pouca ou nenhuma dificuldade em encontrar orientandos. No entanto, identificou-se um certo grau de desafio em gerenciar todas as orientações ao longo do semestre. Esta complexidade pode ser atribuída ao elevado número de orientações, uma vez que mais da metade dos professores que participaram da pesquisa conduzem cinco ou mais orientações. É relevante destacar que uma parte significativa dos orientadores opta por realizar orientações em dupla, e a grande maioria deles só disponibiliza vagas para orientação a partir do semestre 2024.2.

Em relação à pergunta sobre a possibilidade de utilizar uma plataforma baseada em sistema de recomendação para conectar alunos e professores, observou-se consenso entre

os respondentes, indicando que a maioria dos usuários demonstrou uma alta probabilidade de utilizar tal plataforma.

A partir dessa coleta e análise de dados, realizou-se à priorização e ajuste dos requisitos que foram elicitados na etapa de *Lean Inception*, os quais foram transpostos para um backlog. Além disso, foram executadas a modelagem dos dados, o detalhamento da arquitetura e o refinamento dos protótipos. Todas essas etapas encontram-se devidamente documentadas e acordadas no Capítulo 5 do presente trabalho.

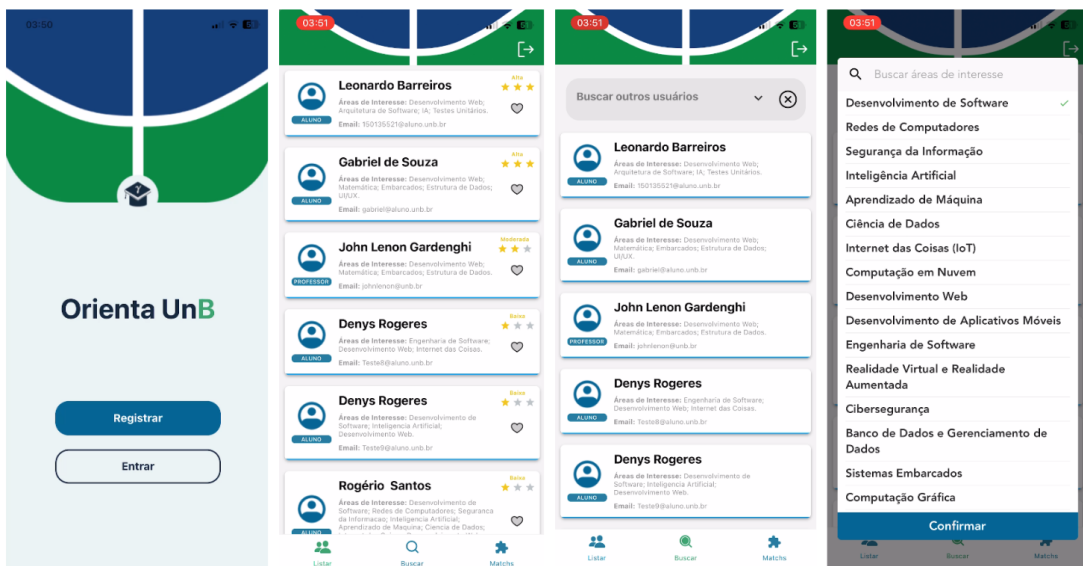
6.4 Etapa Ação

Com base no planejamento realizado na etapa anterior, esta fase teve como objetivo realizar o desenvolvimento do Aplicativo Orienta UnB. Para isso, utilizou-se o React Native para o desenvolvimento mobile, em conjunto com a Expo. O *backend* foi desenvolvido utilizando o *framework* Nest.js, enquanto o MongoDB foi adotado como banco de dados. O detalhamento dessas tecnologias encontra-se no Capítulo 3.

Além disso, o processo de desenvolvimento foi conduzido pela Metodologia estabelecida no Capítulo 4, orientando-se por práticas ágeis. Dessa forma, foi adotada uma combinação entre *Scrum*, *Kanban*, *XP* e *Lean Inception*.

A implementação real do aplicativo é demonstrada na Figura 44 e a aplicação completa está disponível para visualização e utilização no repositório do GitHub através do seguinte *link*: <https://github.com/Orienta-Universidade-de-Brasilia/TCC-POC>.

Figura 44 – Imagens das telas principais da aplicação.



Fonte: Autor

6.5 Etapa Avaliação

A fase de avaliação em uma pesquisa-ação desempenha um papel fundamental na reflexão e na validação dos resultados obtidos, proporcionando informações para melhorias e aprendizados. No entanto, dada a natureza participativa e dinâmica da Pesquisa-Ação, o tempo disponível para a condução do presente trabalho acabou sendo um fator limitante para realizar esta etapa, impossibilitando sua execução durante o período.

É importante destacar que, embora a fase de avaliação não tenha sido executada, as fases anteriores da pesquisa-ação foram conduzidas com rigor, buscando a identificação de problemas, a coleta de dados, o planejamento estratégico e o desenvolvimento da solução. A integridade e a qualidade dessas etapas iniciais oferecem uma base sólida para a compreensão do contexto estudado. Acredita-se que as informações e percepções colhidas até o momento contribuem para o entendimento do problema em questão, oferecendo uma base sólida para discussão e conclusões.

É recomendável que pesquisas futuras possam explorar a fase de avaliação, aprimorando a intervenção proposta e ampliando o entendimento do contexto envolvido. Esta recomendação não se restringe à conclusão de uma etapa, mas visa explorar oportunidades de melhoria nas práticas sugeridas, aumentando o impacto da pesquisa e enriquecendo as contribuições para o campo em questão.

6.6 Resumo do Capítulo

Neste capítulo, foram apresentados os resultados obtidos ao longo das fases de Pesquisa-Ação. Na Etapa Exploratória, coletou-se dados e informações para compreender o problema e realizar o diagnóstico da situação. Em sequência, a Etapa de Planejamento, é dada continuidade à Etapa Exploratória, definindo um plano detalhado para realizar o desenvolvimento da aplicação. Em seguida, na Etapa de Ação, compreendeu-se o desenvolvimento da aplicação Orienta UnB, fundamentando-se nas etapas anteriores. Por fim, a Etapa de Avaliação, foram realizadas considerações e recomendações acerca desta etapa.

7 Conclusão

Este capítulo tem como intuito apresentar as considerações finais em relação ao desenvolvimento deste trabalho. Primeiramente, será repassado o contexto geral que justifica a realização deste trabalho. Na sequência, será exposto o *status* geral do trabalho, abrangendo a primeira e segunda etapa, além de trazer os *status* da implementação. Em seguida, avalia-se se os objetivos específicos foram alcançados e respondendo à pergunta de pesquisa deste trabalho. Por fim, são propostas possíveis direções para trabalhos futuros.

7.1 Contexto Geral

É notório que, durante a execução do Trabalho de Conclusão de Curso, há desafios a serem enfrentados, independentemente da área de graduação. As principais problemáticas identificadas incluem questões de custo e a busca por um orientador, conforme destacado por [Guimarães e Sobrinho \(2020\)](#).

Adicionalmente, é observado que há uma certa dificuldade por parte dos estudantes de Engenharia de Software da Universidade de Brasília em encontrar um orientador para a elaboração do Trabalho de Conclusão de Curso.

Diante do exposto, surge a necessidade de uma intervenção para este impasse. Este estudo apresenta uma forma de reduzir este problema a partir do desenvolvimento de um aplicativo baseado em sistema de recomendação para conectar alunos e professores com interesses em comum, facilitando projetos acadêmicos, como o TCC. Para isso, foi realizada uma pesquisa exploratória de caráter aplicado que utilizou como Metodologia de Análise de Resultados, a Pesquisa-Ação e, como Metodologia de Desenvolvimento, uma mescla entre *Scrum*, *Kanban*, *XP* e *Lean Inception*.

7.2 Status Geral do Trabalho

Para a primeira etapa deste trabalho, foi realizada a documentação do embasamento teórico para sustentar os princípios desta pesquisa. Visando apresentar uma solução para um problema real dentro da UnB. Nesse sentido, o Quadro 10 demonstra a situação das atividades realizadas na fase inicial do trabalho.

Quadro 10 – Atividades da primeira etapa do TCC

Atividades	Status
Definir tema	Concluído
Levantamento Bibliográfico	Concluído
Realizar alinhamento de visão e escopo do trabalho	Concluído
Estabelecer referencial teórico	Concluído
Definir suporte tecnológico	Concluído
Especificar metodologia	Concluído
Produzir prova de conceito	Concluído
Determinar proposta do trabalho	Concluído
Revisar monografia	Concluído
Apresentar TCC 1	Concluído

Fonte: Autor

Para a segunda etapa do TCC, as atividades planejadas consistem em realizar ajustes na escrita da monografia, mas principalmente em desenvolver o aplicativo Orienta UnB. No início dessa etapa, foi divulgado o formulário para o corpo estudantil e professores para o levantamento de métricas. O Quadro 11 apresenta a situação das atividades da segunda etapa para o TCC.

Quadro 11 – Atividades da segunda etapa do TCC

Atividades	Status
Realizar pesquisa com alunos e professores	Concluído
Realizar melhorias	Concluído
Realizar desenvolvimento da aplicação	Concluído
Análise de resultados	Concluído
Finalizar monografia	Concluído
Apresentar TCC 2	Concluído

Fonte: Autor

Em relação às Histórias de Usuário, todas foram concluídas com êxito, com exceção da História de Usuário 9 (US9), História de Usuário 10 (US10) e História de Usuário 11 (US11), que estão dispostas nos Quadros 8 e 9. Isso se deve às prioridades definidas pelos autores e ao tempo disponível para o desenvolvimento.

7.3 Objetivos e Questões de Pesquisa

7.3.1 Objetivos

Foram definidos como objetivos do trabalho:

- Elaborar um referencial teórico que apresente conceitos sobre a utilização de aplicativos móveis como ferramenta para auxílio universitário, e sobre sistemas de recomendação. *Status*: Cumprido, apresentado no Capítulo 2;
- Documentar as dificuldades apresentadas pelos alunos na busca por orientadores para seus Trabalhos de Conclusão de Curso (TCC). *Status*: Cumprido, apresentado no Capítulo 6;
- Desenvolver um *backlog* do produto que contemple as possíveis funcionalidades. *Status*: Cumprido, apresentado no Capítulo 5;
- Prototipação do aplicativo móvel. *Status*: Cumprido, apresentado no Capítulo 5;
- Implementar o aplicativo com base no contexto. *Status*: Cumprido, apresentado no Capítulo 6;
- Análise e documentação do processo de pesquisa. *Status*: Cumprido, apresentado no Capítulo 6.

Ao atingir os objetivos específicos, o objetivo geral deste trabalho, que consistia em "Desenvolver um aplicativo móvel que utilize sistema de recomendação para estabelecer conexões entre alunos e professores que compartilham interesses comuns, atendendo à necessidade dos alunos de Engenharia de Software da Universidade de Brasília de encontrar um professor orientador para cursar a disciplina de Trabalho de Conclusão de Curso.", também foi atingido.

7.3.2 Questões de Pesquisa

A questão de pesquisa do presente trabalho foi "É possível melhorar o alinhamento de expectativas e interesses, proporcionando visibilidade e disponibilidade entre professor orientador e concludente, por meio de um aplicativo?".

A partir da análise detalhada no Capítulo 6 foi visto que, entre as dificuldades encontradas pelos alunos de Engenharia de Software da UnB ao buscar por orientação, destacam-se: encontrar um orientador disponível; encontrar professor com área de estudo pretendida pelo aluno e saber as áreas de interesse dos professor. No entanto, como a Etapa Avaliação da Pesquisa-Ação não pode ser feita, conforme descrito na 6.5, não é

possível afirmar que um aplicativo poderia ser eficaz para alinhar interesses, proporcionar visibilidade sobre as áreas de interesse e disponibilidade dos professores orientadores.

A implementação dessa última etapa permitiria verificar tal fato, além de explorar a possibilidade de otimizar o processo de orientação por meio da formação de duplas com interesses semelhantes, que seria respaldada pelo fato de que mais da metade dos professores realizam, em média, pelo menos uma orientação em dupla por semestre, conforme evidenciado no Capítulo 6.

7.4 Trabalhos Futuros

Este estudo proporciona oportunidades de aprimoramento que podem ser abordado em trabalhos futuros relacionados. Dentre eles, podem ser mencionados:

- Mesclar estratégias de filtragem colaborativa e filtragem baseada em conteúdo com o objetivo de aprimorar a eficiência do sistema de recomendação;
- Finalizar as Histórias de Usuário 9, 10 e 11, trazendo informações relativas à disponibilidade do professor orientador e realização de *Matches*;
- Implantar o aplicativo no ambiente universitário, realizando testes piloto com uma amostragem representativa de alunos e professores a fim de avaliar a eficácia do sistema, e
- Estender a abordagem para outros projetos acadêmicos, como projetos de Iniciação Científica.

Referências

- ALSAADI, B.; SAEEDI, K. Data-driven effort estimation techniques of agile user stories: a systematic literature review. *Artificial Intelligence Review*, Springer, p. 1–32, 2022. Citado 2 vezes nas páginas 71 e 72.
- BANDEIRA, L. K. R.; NETO, M. d. S. A. O que é um smart campus? *Perspectivas em Gestão & Conhecimento*; v. 12 n. 1 (2022); 175-188, v. 24, n. 2, p. 188–175, 2022. Citado 2 vezes nas páginas 27 e 53.
- BARBOSA, C. E. M. Estudo de técnicas de filtragem híbrida em sistemas de recomendação de produtos. *Monografia. Centro de Informática, Ciência da Computação, UFPE*, 2014. Citado 2 vezes nas páginas 31 e 35.
- BOEG, J. Kanban em 10 passos. *Tradução de Leonardo Campos, Marcelo Costa, Lúcio Camilo, Rafael Buzon, Paulo Rebelo, Eric Fer, Ivo La Puma, Leonardo Galvão, Thiago Vespa, Manoel Pimentel e Daniel Wildt. C4Media*, p. 27, 2010. Citado na página 48.
- CAROLI, P. Lean inception. *MartinFowler. com*, 2017. Citado 4 vezes nas páginas 56, 57, 71 e 72.
- CAROLI, P. *Lean Inception: Como Alinhar Pessoas e Construir o Produto Certo*. [S.l.: s.n.], 2018. Citado na página 49.
- COUTINHO, J. G. Um estudo sobre sistemas de recomendação baseados em transformers. 2022. Citado na página 37.
- EXPO. *Expo Go - Expo Documentation*. 2023. Disponível em: <<https://docs.expo.dev/get-started/expo-go/>>. Citado na página 42.
- EXPO. *FAQ - Expo Documentation*. 2023. Disponível em: <<https://docs.expo.dev/faq/>>. Citado na página 42.
- FIGMA. *Guide to Developer Handoff: Components, Styles, and Documentation*. 2023. Disponível em: <<https://www.figma.com/best-practices/guide-to-developer-handoff/components-styles-and-documentation/>>. Citado na página 41.
- GERHARDT, T. E.; SILVEIRA, D. T. *Métodos de pesquisa*. [S.l.]: Plageder, 2009. Citado 2 vezes nas páginas 45 e 46.
- GIT. *Git - Documentation*. 2023. Disponível em: <<https://git-scm.com/docs/git>>. Citado na página 40.
- GITHUB. *GitHub Documentation*. 2023. Disponível em: <<https://docs.github.com/pt/get-started/quickstart/hello-world>>. Citado na página 41.
- GUIMARÃES, J. de C.; SOBRINHO, F. D. da S. Fatores facilitadores e dificultadores à construção do tcc. *Revista Brasileira de Administração Científica*, v. 11, n. 3, p. 82–99, 2020. Citado 2 vezes nas páginas 24 e 89.

- MARCAC, M. *React Native Pros and Cons*. 2021. Disponível em: <<https://pagepro.co/blog/react-native-pros-and-cons/>>. Citado na página 39.
- MongoDB. *MongoDB Documentation*. 2023. Disponível em: <<https://www.mongodb.com/docs/atlas/>>. Citado na página 40.
- MORAES, E. C. T. de *et al.* Dificuldades encontradas por alunos de graduação durante a elaboração de trabalhos de conclusão de curso-tcc difficulties encountered by graduate students during the preparation of works for completing the course-tcc. 2021. Citado na página 23.
- NEST.JS. *Nest.js Documentation*. 2023. Disponível em: <<https://docs.nestjs.com/>>. Citado na página 39.
- NETO, M. d. S. A.; BANDEIRA, L. K. R. Smart campus: uma estratégia de gestão para a crise orçamentária das instituições federais de ensino superior? 2019. Citado 2 vezes nas páginas 30 e 31.
- NEVES, A. R. d. M. *et al.* Iniciativa smart campus: um estudo de caso em progresso na universidade federal do pará. In: SBC. *Anais do I Workshop de Computação Urbana*. [S.l.], 2017. Citado 5 vezes nas páginas 27, 28, 30, 53 e 54.
- POLIN, K. *et al.* The making of smart campus: A review and conceptual framework. *Buildings*, MDPI, v. 13, n. 4, p. 891, 2023. Citado 3 vezes nas páginas 28, 29 e 53.
- ROMANO, R. R.; PINTO, L. G. P.; PACHECO, S. C. P. Smart campus facens–construindo uma cidade inteligente em um campus universitário utilizando-se do fablab. Disponível http://fablearn.org/wp-content/uploads/2016/09/FLBrazil_2016_paper_150.pdf, 2016. Citado na página 28.
- SCHENATZ, B. N.; CUNHA, M. A. V. C. d.; KUGLER, J. L. C. Smart campus e analytics na gestão de instituições de ensino superior para redução da evasão e promoção da permanência. 2019. Citado 3 vezes nas páginas 23, 27 e 53.
- SCHWABER, K.; SUTHERLAND, J. *Scrum Guide*. 2020. <<https://scrumguides.org/>>. Acessado em 12 de julho de 2023. Citado na página 48.
- SERVICES, A. W. *Elastic Compute Cloud*. 2023. Disponível em: <<https://aws.amazon.com/pt/ec2/>>. Citado na página 42.
- SILVA, R. G. N. e. *Sistema de Recomendação baseado EM Conteúdo Textual: Avaliação E Comparação*. Instituto de Matemática. Departamento de ciência da Computação, 2014. Disponível em: <<https://repositorio.ufba.br/handle/ri/19281>>. Citado 7 vezes nas páginas 31, 32, 33, 34, 35, 36 e 37.
- SOUZA, J. C. A.; OLIVEIRA, M. R. Metodologias ágeis: um comparativo entre extreme programming (xp) e scrum. *Ciência & Tecnologia*, v. 13, n. 1, p. 133–141, 2021. Citado na página 48.

Apêndices

APÊNDICE A – Pesquisa de campo

As Figuras 45 a 51 apresentam os formulários utilizados para a coleta de métricas e para o levantamento dos desafios enfrentados tanto por alunos quanto por professores no contexto do Trabalho de Conclusão de Curso no campus FGA.

Figura 45 – Formulário para aluno - 1º seção

Coleta de dados - TCC | Perfil Aluno

Este formulário tem o propósito de coletar informações relacionadas ao processo de Trabalho de Conclusão de Curso (TCC), abrangendo o tempo de espera dos alunos para orientação, a modalidade de TCC escolhida e as dificuldades encontradas na busca por um orientador.

O preenchimento deste formulário leva aproximadamente 1 minuto e meio e as informações pessoais dos participantes não serão expostas publicamente.

[Faça login no Google](#) para salvar o que você já preencheu. [Saiba mais](#)

*** Indica uma pergunta obrigatória**

Você já definiu seu orientador para o TCC ou já concluiu seu TCC???? *

Sim

Não

Fonte: Autor

Figura 46 – Formulário para aluno - 2º seção - Parte 1: Aluno com orientador

Qual nível de dificuldade você teve para encontrar um orientador? *

1 2 3 4 5

Baixa dificuldade Alta dificuldade

Quanto tempo levou para encontrar um orientador? *

Cerca de uma semana ou menos

Cerca de um mês

Cerca de um semestre

Aproximadamente dois semestres

Mais do que dois semestres

Qual a maior dificuldade que você sentiu na procura por um orientador?

Sua resposta _____

Quantos professores você procurou para solicitar orientação? *

1

2

3

4

5

6

7

8

9

10 ou +

Fonte: Autor

Figura 47 – Formulário para aluno - 2º seção - Parte 2: Aluno com orientador

Qual meio de comunicação você utilizou para encontrar seu orientador?

Sua resposta _____

Como está fazendo ou fez o TCC? *

Individual

Dupla

Caso esteja fazendo ou tenha feito o TCC em dupla, como você conheceu seu parceiro(a) para o TCC?

Sua resposta _____

Quais os meios de comunicação utilizado durante a orientação? *

E-mail

Telegram

Whatsapp

Teams

Discord

Presencial

Google Meets

Google Classroom

Zoom

Outro: _____

Se durante o processo de escolha de um orientador, você tivesse a oportunidade * de utilizar uma plataforma baseada em sistema de recomendação para conectar alunos e professores com interesses em comum, facilitando projetos acadêmicos, como o TCC, qual seria a probabilidade de você usar essa plataforma?

1 2 3 4 5

Pouca possibilidade Alta possibilidade

Fonte: Autor

Figura 48 – Formulário para aluno - 2º seção - Parte 1: Aluno sem orientador

Você já começou a procurar um professor para orientação do seu TCC? Se sim, *
quantos professores você procurou até o momento?

Nenhum (ainda não iniciou a busca por orientador)

1

2

3

4

5

6

7

8

9

10 ou +

Caso já tenha entrado em contato com algum professor buscando orientação, qual
meio de comunicação você utilizou?

Sua resposta _____

Caso já esteja na procura por orientador, qual está sendo sua maior dificuldade?

Sua resposta _____

Fonte: Autor

Figura 49 – Formulário para aluno - 2º seção - Parte 2: Aluno sem orientador

Em qual formato você deseja fazer seu TCC? *

Individual

Dupla

Se você pretende realizar o TCC em dupla, você já escolheu o seu parceiro(a)? Em caso afirmativo, como vocês se conheceram?

Sua resposta

Qual a chance de você utilizar uma plataforma baseada em sistema de recomendação para conectar alunos e professores com interesses em comum, visando facilitar projetos acadêmicos, como o TCC? *

1 2 3 4 5

Pouca possibilidade Alta possibilidade

Fonte: Autor

Figura 50 – Formulário para professor - Parte 1

Coleta de dados - TCC | Perfil Professor

Este formulário tem o propósito de coletar informações relacionadas ao processo de Trabalho de Conclusão de Curso (TCC), abrangendo o tempo de espera dos alunos para orientação, a modalidade de TCC escolhida e as dificuldades encontradas na busca por um orientador.

O preenchimento deste formulário leva aproximadamente 1 minuto e as informações pessoais dos participantes não serão expostas publicamente.

[Faça login no Google](#) para salvar o que você já preencheu. [Saiba mais](#)

*** Indica uma pergunta obrigatória**

Em qual curso você ministra aulas? Ex.: Engenharia de Software *

Sua resposta _____

Qual a dificuldade para encontrar alunos para orientar? *

1 2 3 4 5

Baixa dificuldade Alta dificuldade

Em média, quantas orientações você costuma realizar por semestre, levando em conta que as duplas são consideradas como uma única orientação? *

Sua resposta _____

Dentro dessa média de orientações por semestre, quantas são em dupla? *

Sua resposta _____

Qual o grau de dificuldade para conduzir todas essas orientações durante o semestre?

1 2 3 4 5

Baixa dificuldade Alta dificuldade

Fonte: Autor

Figura 51 – Formulário para professor - Parte 2

Considerando a ordem de espera na fila de orientação, a partir de qual semestre *
você tem disponibilidade para aceitar novos orientandos?

2024.1

2024.2

2025.1

2025.2

2026.1

2026.2

2027.1

2027.2

Quais os meios de comunicação utilizado durante a orientação? *

E-mail

Telegram

Whatsapp

Teams

Discord

Presencial

Google meets

Google Classroom

Zoom

Outro: _____

Se você tivesse a oportunidade de utilizar uma plataforma baseada em sistema *
de recomendação para conectar alunos com interesses na sua área de
pesquisa/interesse, facilitando projetos acadêmicos, como o TCC, qual seria a
probabilidade de você usar essa plataforma?

1 2 3 4 5

Pouca possibilidade Alta possibilidade

Fonte: Autor

APÊNDICE B – Protótipo de alta fidelidade

O projeto de alta fidelidade está disponível na plataforma Figma, que tem o objetivo de editar gráfico de vetores e prototipagem. O *link* a seguir redireciona para acessar a página do protótipo, a fim de obter uma visão detalhada do projeto de alta fidelidade:

<<https://www.figma.com/file/dP68EOceyJdJShtxis8X1S/Design-system?type=design&node-id=0%3A1&mode=design&t=OGKAKL4jDmu5O2Uv-1>>

APÊNDICE C – Política Privacidade

Abaixo encontra-se o termo de política de privacidade, em que expõe a intenções na utilização dos dados solicitados durante o uso do aplicativo:

Política Privacidade: A sua privacidade é importante para nós. É política do Orienta UnB respeitar a sua privacidade em relação a qualquer informação sua que possamos coletar no aplicativo Orienta UnB.

Solicitamos informações pessoais apenas quando realmente precisamos delas para lhe fornecer um serviço. Fazemo-lo por meios justos e legais, com o seu conhecimento e consentimento. Também informamos por que estamos coletando e como será usado.

Apenas retemos as informações coletadas pelo tempo necessário para fornecer o serviço solicitado. Quando armazenamos dados, protegemos dentro de meios comercialmente aceitáveis para evitar perdas e roubos, bem como acesso, divulgação, cópia, uso ou modificação não autorizados. Não compartilhamos informações de identificação pessoal publicamente ou com terceiros, exceto quando exigido por lei.

O nosso aplicativo pode ter *links* para sites externos que não são operados por nós. Esteja ciente de que não temos controle sobre o conteúdo e práticas desses sites e não podemos aceitar responsabilidade por suas respectivas políticas de privacidade. Você é livre para recusar a nossa solicitação de informações pessoais, entendendo que talvez não possamos fornecer alguns dos serviços desejados.

O uso continuado de nosso aplicativo será considerado como aceitação de nossas práticas em torno de privacidade e informações pessoais. Se você tiver alguma dúvida sobre como lidamos com dados do usuário e informações pessoais, entre em contato conosco.

Compromisso do Usuário: O usuário se compromete a fazer o uso adequado dos conteúdos e da informação que o Orienta UnB oferece no site com caráter enunciativo, mas não limitativo:

A) Não se envolver em atividades que sejam ilegais ou contrárias à boa fé e à ordem pública;

B) Não difundir propaganda ou conteúdo de natureza racista, xenofóbica, jogos de sorte ou azar, qualquer tipo de pornografia ilegal, de apologia ao terrorismo ou contra os direitos humanos;

C) Não causar danos aos sistemas físicos (*hardwares*) e lógicos (*softwares*) do Orienta UnB, de seus fornecedores ou terceiros, para introduzir ou disseminar vírus informáticos ou quaisquer outros sistemas de *hardware* ou *software* que sejam capazes de

causar os danos anteriormente mencionados.

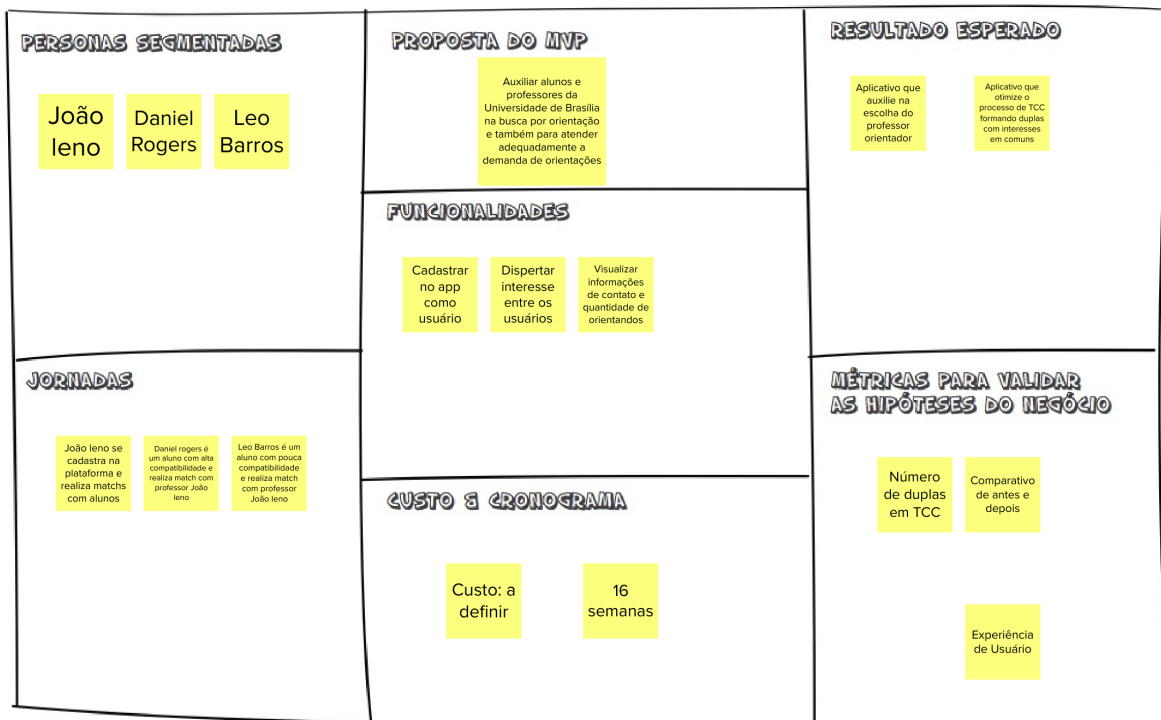
Mais informações: Esperemos que esteja esclarecido e, como mencionado anteriormente, se houver algo que você não tem certeza se precisa ou não, geralmente é mais seguro deixar os *cookies* ativados, caso interaja com um dos recursos que você usa em nosso site.

Anexos

ANEXO A – Canvas MVP

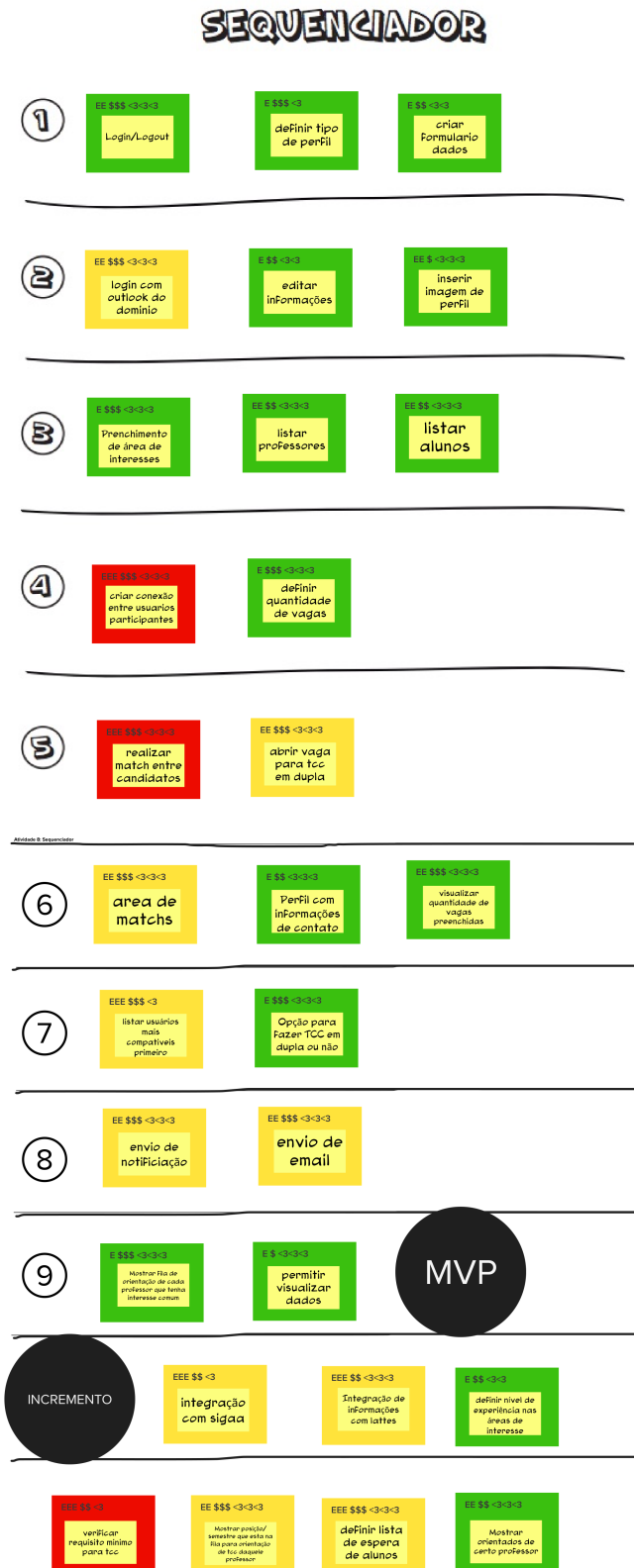
Canvas MVP 1

CANVAS MVP



Fonte: Autor

ANEXO B – Sequenciador



ANEXO C – Regras

É-Nãoé-Faz-Nãofaz 1

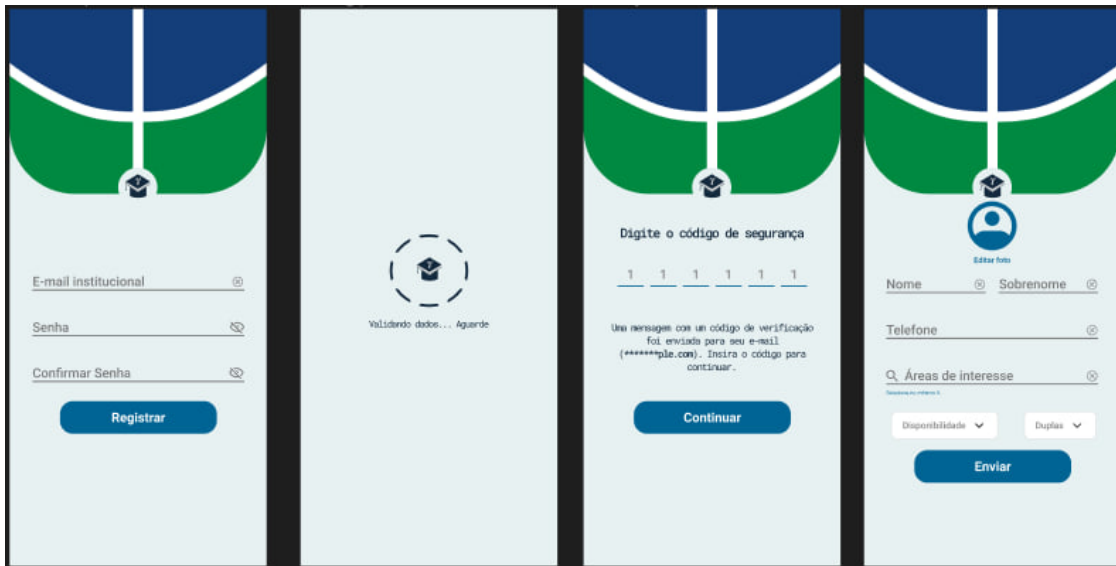
<p>É</p> <p>Aplicativo mobile</p> <p>App de combinação de interesses</p> <p>App de análise de compatibilidade de interesses</p>	
<p>FAZ</p> <p>algoritmo de análise combinatória (Cruzamento de dados)</p> <p>disponibiliza informações para os usuários</p> <p>realiza a função de 'match'</p>	

É-Nãoé-Faz-Nãofaz 2

	<p>NÃO É</p> <p>Aplicativo de relacionamento</p> <p>Rede social</p> <p>Novo linkedin/ lattes</p> <p>Sigaa</p> <p>Moodle</p>
	<p>NÃO FAZ</p> <p>criação de chats por meio do app</p> <p>exportação de dados</p> <p>preenchimento formulário para tcc</p> <p>cadastro de tcc</p> <p>armazenamento de dados sensíveis dos usuários</p>

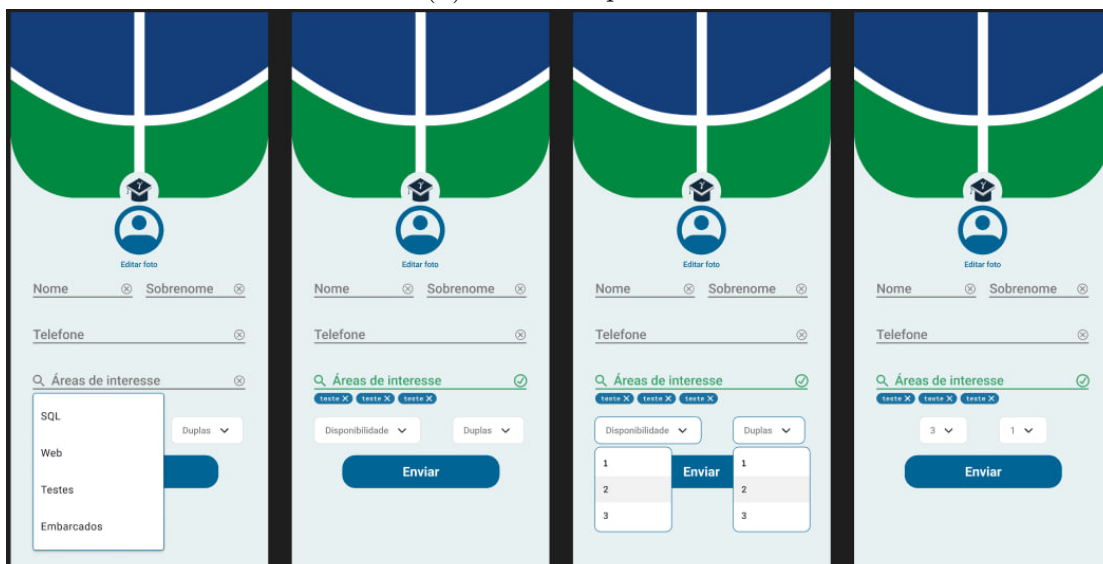
ANEXO D – Cadastro do perfil professor: parte 1

(a) Cadastro: parte 1



Fonte: Autor

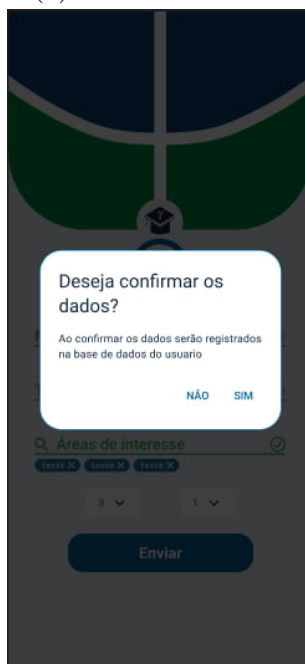
(b) Cadastro: parte 2



Fonte: Autor

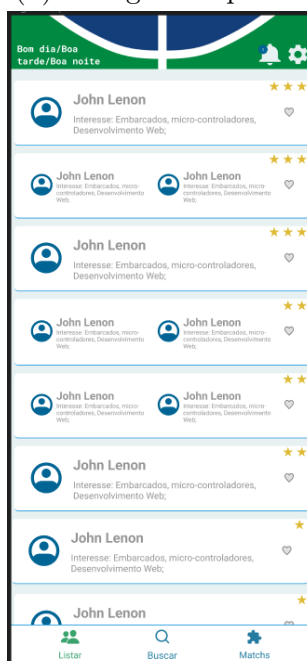
ANEXO E – Cadastro do perfil professor: parte 2

(a) Confirmar Dados



Fonte: Autor

(b) Listagem de pessoas



Fonte: Autor