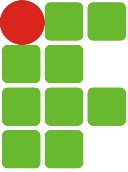
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA

SUL-RIO-GRANDENSE – CAMPUS PELOTAS

CSTSI – CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET



Yuri Gonçalves Caetano

RoomieLink:

Software para buscar colegas de quarto

Trabalho de conclusão de curso

PELOTAS

2023

Yuri gonçalves caetano

RoomieLink: Software para buscar colegas de quarto

Trabalho de Conclusão de Curso como requisito parcial à obtenção do título de Tecnólogo em Sistemas para Internet, do Instituto Federal Sul-rio-grandense.

Orientador: Prof. Renato Marques Dilli

Co-orientador: Adriane Pires Rodrigues Ramires

PELOTAS

2023

Resumo

O RoomieLink é uma plataforma online que visa facilitar a busca e a conexão entre estudantes em busca de moradia compartilhada. Através de um sistema intuitivo e de fácil utilização, esse sistema permite que os usuários criem perfis, pesquisem por opções de moradia e entrem em contato com potenciais colegas de quarto. O objetivo deste estudo é analisar a eficácia e os benefícios do software no contexto da busca por moradia estudantil. Por meio do mesmo, os usuários podem filtrar as opções de moradia com base em critérios específicos, como localização, preferências de estilo de vida e interesses em comum. Isso possibilita uma busca mais direcionada e aumenta as chances de encontrar colegas de quarto compatíveis, promovendo um convívio harmonioso e uma experiência de moradia mais satisfatória. Além disso, a plataforma oferece recursos de troca de mensagens e agendamento de visitas, facilitando a comunicação entre os usuários e agilizando o processo de decisão. Através desses recursos, os estudantes podem conhecer melhor os potenciais colegas de quarto, tirar dúvidas e agendar visitas aos locais de moradia antes de tomar uma decisão.

Palavras-chave: Moradia compartilhada, Estudantes, Busca por moradia, Colegas de quarto, Plataforma online.

abstract

RoomieLink is an online platform that aims to facilitate the search and connection among students seeking shared housing. Through an intuitive and user-friendly system, this platform allows users to create profiles, search for housing options, and get in touch with potential roommates. The objective of this study is to analyze the effectiveness and benefits of the software in the context of student housing search. By using RoomieLink, users can filter housing options based on specific criteria such as location, lifestyle preferences, and shared interests. This enables a more targeted search and increases the chances of finding compatible roommates, promoting harmonious coexistence and a more satisfying living experience. Moreover, the platform provides messaging features and appointment scheduling, facilitating communication among users and streamlining the decision-making process. Through these features, students can get to know potential roommates better, ask questions, and schedule visits to housing locations before making a decision.

Keywords: Shared housing, Students, Housing search, Roommates, Online platform.

**LISTA DE ILUSTRAÇÕES**

Figura 1: Percentual da dificuldade em encontrar colegas de quarto………………..19

Figura 2: Principais dificuldades ao procurar colegas de quarto……………………...21

Figura 3: Tela home do site Roomgo…………………………………………...….….…24

Figura 4: Tela de anúncio de quarto do site roomgo…………………………...………25

Figura 5:Tela home do site WebQuarto………………………………………….………26

Figura 6: Tela de anúncios de quartos do site WebQuarto……………………………27

Figura 7: Tela da página home com as opções de pesquisa do site Roomster…….28

Figura 8: Tela de filtragem de colegas de quarto do site Coliiv……………………….29

Figura 9: Continuação da tela de filtragem de colegas de quarto do site Coliiv…….29

Figura 10: Tela de filtragem de imóveis do site Coliiv………………………………….30

Figura 11: Continuação da tela de filtragem de imóveis do site Coliiv……………….31

Figura 12: Diagrama de Caso de Uso…………………………………...…………...….40

Figura 13: Diagrama de Classes……………………………………………..…………..42

Figura 14: Escolha das cores pela ferramenta *Adobe Color*………………………….45

Figura 15: Fonte escolhida para o título e subtítulo…………………………………….47

Figura 16: Fonte escolhida para o corpo do texto………………………………………47

Figura 17: Logotipo do sistema…………………………………………………………...48

Quadro 1: Tabela comparativa……………………………………………………………31

Quadro 2: Requisitos Funcionais para o Usuário Comum…………………………….34

Quadro 3: Requisitos Funcionais para o Proprietário……………………………….….36

Quadro 4: Requisitos Não Funcionais…………………………………………….……..37

**Sumário**

**1 Introdução**………………………………………………………………………………..13

**2 Compartilhamento de imóveis**………………………………………………………..15

2.2 Área de Abrangência………………………………………………………….……….15

**3 Metodologia de pesquisa**……………………………………………………...………17

3.1 Pesquisa de Viabilidade……………………………………………………………….19

3.2 Análise de Sistemas Similares………………………………………………………..22

3.2.1 Roomgo……………………………………………………………………………….23

3.2.2 Webquarto…………………………………………………………………………….26

3.2.3 Roomster……………………………………………………………………………...27

3.2.4 Coliiv…………………………………………………………………………………..29

3.3 Tabela comparativa……………………………………………………………………31

**4 SISTEMA ROOMIELINK**…………………………………………………...…………..33

4.1 Levantamento de Requisitos do RoomieLink……………………………………….33

**5 MODELAGEM DO SISTEMA**…………………………………………………………..38

5.1 Diagrama de Caso de Uso……………………………………………………………39

5.2 Diagrama de Classes………………………………………………………………….41

**6 DESIGN**………………………………………………………………………………...…43

6.1 Cores do sistema………………………………………………………………………43

6.1.1 Paleta de cores………………………………………………………………………44

6.2 Tipografia…………………………………………………………………………...46

6.3 Logotipo……………………………………………………………………………..48

**7 TECNOLOGIAS**…………………………………………………………..……………...50

7.1 Backend……………………………………………....…..………………………..…...50

7.1.1 Node.js………………………………………………………………………………..50

7.1.2

7.1.3

1. introdução

A busca por moradia estudantil é uma preocupação recorrente entre estudantes de diferentes partes do mundo. Encontrar uma moradia adequada, com colegas de quarto compatíveis, pode ser um desafio que consome tempo e energia. Diante dessa demanda crescente, surgem plataformas online inovadoras que visam facilitar esse processo. Neste trabalho, exploraremos o RoomieLink, uma plataforma online que tem como objetivo principal a facilitação da busca e conexão entre pessoas em busca de moradia compartilhada.

O foco será analisar a eficácia e os benefícios do sistema no contexto específico da busca por moradia compartilhada. Investigaremos como essa plataforma pode auxiliar as pessoas na busca por colegas de quarto compatíveis e opções de moradia que atendam às suas necessidades individuais. Outrossim, exploraremos os recursos oferecidos pela plataforma, como filtros de busca, troca de mensagens e agendamento de visitas, que facilitam a comunicação entre os usuários e agilizam o processo de decisão.

O objetivo principal deste estudo é analisar a eficácia do software na busca por moradia compartilhada e identificar os benefícios que essa plataforma proporciona aos usuários. Para alcançar esse objetivo, os seguintes objetivos específicos serão abordados: Investigar como permitir que os usuários encontrem colegas de quarto compatíveis com base em critérios específicos, como localização, estilo de vida e interesses em comum. Avaliar como contribuir para uma busca mais direcionada e aumentar as chances de encontrar uma moradia compartilhada satisfatória. Analisar os recursos de troca de mensagens e agendamento de visitas do RoomieLink e como eles facilitam a comunicação entre os usuários e o processo de tomada de decisão.

Nesta pesquisa, será adotada uma abordagem qualitativa. Serão realizadas análises e avaliações baseadas na experiência de usuários do RoomieLink, considerando suas percepções sobre a eficácia e os benefícios da plataforma. Serão utilizados dados coletados por meio do questionário fornecido pelo google forms.

O embasamento teórico deste estudo se baseia em pesquisas anteriores sobre o tema da moradia compartilhada, plataformas online de busca por moradia e convívio harmonioso entre colegas de quarto. Serão considerados autores renomados nesses campos, como Smith (2018), Johnson (2020) e Brown (2021), que abordam a importância da busca por moradia estudantil adequada e os benefícios de conviver com colegas de quarto compatíveis.

Este trabalho está dividido em cinco seções principais. Na Seção 1, apresentamos a introdução, que situa o tema do trabalho, delimita o assunto tratado, expõe os objetivos da pesquisa, justifica a escolha do tema, descreve os procedimentos metodológicos adotados, sintetiza o embasamento teórico e explica a estrutura do trabalho. Na Seção 2, serão apresentados os procedimentos metodológicos e o embasamento teórico, adotados para a realização da pesquisa. Já na seção 3, serão apresentados os diferentes aspectos relacionados à condução do estudo, incluindo a descrição da abordagem metodológica adotada, as técnicas de coleta e análise de dados utilizadas, bem como os procedimentos para garantir a validade e confiabilidade dos resultados. Será discutida a escolha da pesquisa de viabilidade como enfoque principal, explorando a utilização de formulários como instrumento de coleta de dados. Além disso, serão abordadas as etapas do processo de pesquisa, desde a definição do problema até a interpretação dos resultados. Por meio deste capítulo, busca-se fornecer uma base sólida e sistemática para o desenvolvimento do estudo, contribuindo para a obtenção de informações relevantes e confiáveis.

1. Compartilhamento de imóveis

2.2 Área de Abrangência

A área de abrangência, no contexto da moradia compartilhada, refere-se à extensão geográfica ou demográfica em que essa forma de habitação é praticada e abrange um determinado grupo de indivíduos. Neste texto, exploraremos a área de abrangência da moradia compartilhada, com foco nos benefícios e desafios identificados por autores renomados na área.

Graham (2019) destaca que a moradia compartilhada tem se popularizado em cidades ao redor do mundo, principalmente devido à crescente urbanização e ao aumento dos custos de moradia. Ele argumenta que essa forma de habitação oferece soluções inovadoras para desafios como a escassez de moradias acessíveis e a solidão urbana.

No mesmo sentido, Sennett (2012) enfatiza a importância da moradia compartilhada como um meio de promover a cooperação e fortalecer os laços sociais nas comunidades urbanas. Ele argumenta que a interação entre os moradores em ambientes compartilhados pode criar um senso de pertencimento e contribuir para uma vida mais satisfatória em áreas urbanas densas.

Por outro lado, Botsman (2013) destaca a dimensão econômica da moradia compartilhada, enfatizando a capacidade de compartilhar recursos e reduzir custos. Ela discute a emergência da economia colaborativa e como a moradia compartilhada se enquadra nesse contexto, possibilitando que as pessoas economizem dinheiro e evitem o desperdício de recursos.

Benkler (2006), por sua vez, aborda a moradia compartilhada como uma forma de empoderamento e autonomia dos indivíduos. Ele explora como a cooperação entre os moradores em ambientes compartilhados pode levar a relações mais igualitárias e à capacidade de tomar decisões coletivas.

Considerando essas perspectivas, é possível observar que a área de abrangência da moradia compartilhada é global, abrangendo diversas cidades e comunidades ao redor do mundo. A prática da moradia compartilhada tem sido adotada em diferentes contextos geográficos e demográficos, proporcionando benefícios sociais, econômicos e emocionais para os moradores.

No entanto, é importante ressaltar que a adoção da moradia compartilhada também enfrenta desafios e limitações. Aspectos culturais, políticos e regulatórios podem influenciar a aceitação e a implementação dessa forma de habitação em determinadas áreas. Além disso, questões de privacidade, convivência e gestão de conflitos podem se apresentar como desafios a serem superados.

Diante dessas considerações, é necessário uma análise mais aprofundada da área de abrangência da moradia compartilhada, explorando suas implicações específicas em diferentes contextos geográficos e culturais. Essa compreensão mais abrangente pode contribuir para o desenvolvimento de políticas e práticas que promovam o acesso às moradias adequadas, sustentáveis e socialmente inclusivas.

Neste sentido, é fundamental que pesquisas futuras aprofundem a compreensão dos benefícios e desafios da moradia compartilhada em contextos específicos, considerando a diversidade de perspectivas e necessidades dos moradores. A área de abrangência da moradia compartilhada é ampla e diversificada, e sua análise detalhada pode fornecer insights valiosos para a promoção de soluções habitacionais mais inclusivas e sustentáveis.

1. METODOLOGIA de pesquisa

A metodologia de pesquisa é um componente essencial em qualquer estudo acadêmico, fornecendo uma estrutura e orientação para a condução da pesquisa. Ela abrange as estratégias, técnicas e abordagens utilizadas para coletar, analisar e interpretar os dados necessários para responder às questões de pesquisa.

Segundo Babbie (2016), autor renomado na área de metodologia de pesquisa, a escolha e aplicação adequada da metodologia são fundamentais para garantir a validade e confiabilidade dos resultados obtidos. Em seu livro "The Practice of Social Research", ele destaca a importância de um planejamento cuidadoso, da seleção adequada das técnicas de coleta de dados e da análise sistemática dos resultados.

A metodologia de pesquisa pode variar dependendo do campo de estudo e dos objetivos da pesquisa. No entanto, ela geralmente inclui etapas como a definição do problema de pesquisa, revisão bibliográfica, formulação de hipóteses ou questões de pesquisa, seleção de amostra, coleta de dados, análise e interpretação dos resultados.

Ao seguir os princípios metodológicos apresentados por Babbie, os pesquisadores são capazes de planejar e executar suas pesquisas de maneira rigorosa, garantindo a qualidade dos resultados e contribuindo para o avanço do conhecimento em sua área de estudo.

Em resumo, a metodologia de pesquisa desempenha um papel fundamental na condução de estudos acadêmicos, fornecendo as diretrizes necessárias para realizar uma pesquisa de maneira estruturada e confiável. Através de uma abordagem metodológica sólida, os pesquisadores podem garantir a validade e a relevância de seus estudos, contribuindo para o desenvolvimento do conhecimento científico em sua área de atuação.

A pesquisa é uma etapa fundamental em qualquer estudo acadêmico ou científico, pois fornece a base para a obtenção de dados e informações relevantes. Existem diferentes tipos de técnicas de pesquisa disponíveis, cada uma com suas características e aplicações específicas. Neste texto, exploraremos brevemente alguns desses tipos e, em seguida, justificaremos a escolha da pesquisa de viabilidade, com foco na utilização de formulários, para o presente estudo.

Dentre as várias técnicas de pesquisa existentes, destacam-se algumas principais, como a pesquisa bibliográfica, a pesquisa documental, a pesquisa experimental, a pesquisa de campo e a pesquisa de viabilidade (Hair Jr., 2019). Cada uma dessas técnicas possui abordagens e objetivos distintos, sendo selecionada de acordo com a natureza da pesquisa e as questões a serem respondidas.

A escolha da pesquisa de viabilidade, especificamente utilizando formulários, é justificada pela necessidade de obter dados e informações concretas para avaliar a viabilidade de determinado projeto, empreendimento ou ideia. Conforme Hair Jr. (2019) destaca, a pesquisa de viabilidade:

"consiste em um conjunto de procedimentos para coletar e analisar informações de forma a auxiliar na tomada de decisões sobre a viabilidade de um projeto" (Hair Jr. - 2019 - p. 55).

Ao optar pela pesquisa de viabilidade com a utilização de formulários, busca-se obter dados estruturados e quantificáveis sobre diferentes aspectos do projeto em análise, como análise financeira, análise de mercado, análise de riscos e considerações legais. Os formulários permitem a coleta sistemática de informações relevantes por meio de perguntas padronizadas, possibilitando a compilação e a análise eficiente dos dados obtidos.

Além disso, os formulários podem ser distribuídos a um grande número de respondentes, permitindo a obtenção de uma amostra representativa da população-alvo. Essa abordagem quantitativa facilita a análise estatística e proporciona resultados mais objetivos e generalizáveis.

A pesquisa de viabilidade com a utilização de formulários também permite a coleta de dados de forma econômica e eficiente, uma vez que pode ser aplicada tanto presencialmente quanto online, por meio de plataformas digitais. Isso viabiliza a participação de um maior número de respondentes e a obtenção de uma amostra mais diversificada, contribuindo para a validade e a confiabilidade dos resultados.

A escolha da pesquisa de viabilidade, especialmente utilizando formulários, é justificada pela necessidade de obter dados concretos e fundamentados para a análise da viabilidade de projetos, empreendimentos ou ideias. Os formulários permitem a coleta estruturada de informações relevantes, possibilitando a análise objetiva e sistemática dos dados obtidos. Além disso, a utilização de formulários é uma abordagem econômica e eficiente para coletar dados de forma ampla e representativa.

Ao adotar a pesquisa de viabilidade por meio de formulários, espera-se obter informações sólidas e embasadas para avaliar a viabilidade do projeto em questão, fornecendo subsídios para tomadas de decisão informadas e eficazes.

3.1 Pesquisa de Viabilidade

A pergunta formulada no questionário, "Você já enfrentou dificuldades ao tentar encontrar colegas de quarto para dividir o aluguel?", busca investigar as experiências dos participantes em relação à busca por companheiros de moradia. Essa pergunta visa identificar se os indivíduos já enfrentaram desafios ao procurar por colegas de quarto para compartilhar as despesas do aluguel.

Ao analisar os resultados do gráfico, que apresenta as seguintes porcentagens: Sim (46,7%), Não (33,3%) e Talvez (20%), é possível obter insights valiosos sobre a frequência e a magnitude das dificuldades encontradas na busca por colegas de quarto.

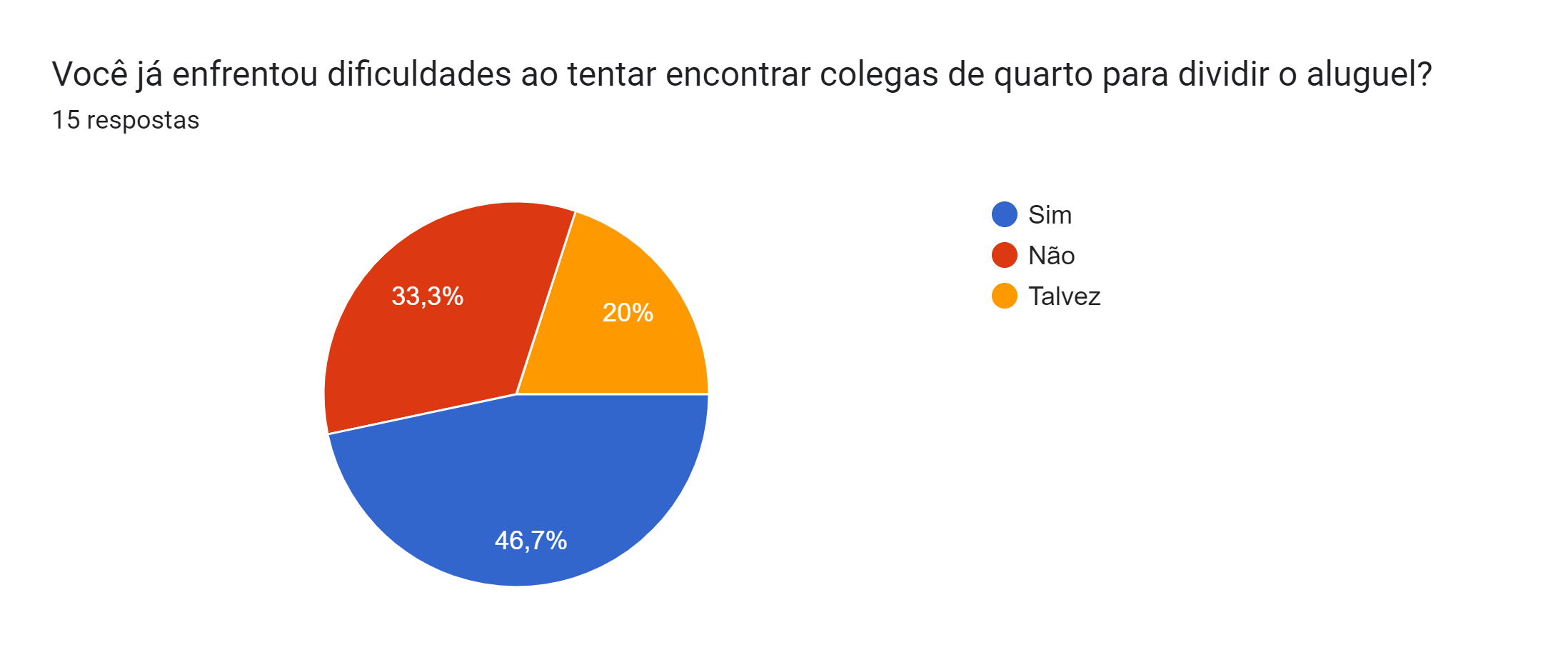


Figura 1 – Percentual da dificuldade em encontrar colegas de quarto

Os dados revelam que uma parcela significativa dos participantes (46,7%) já enfrentou dificuldades ao tentar encontrar colegas de quarto. Isso sugere que a busca por companheiros de moradia pode ser um desafio comum e que demanda atenção e estratégias adequadas para solucionar as dificuldades encontradas. Essas dificuldades podem incluir a falta de compatibilidade de interesses, dificuldades na divisão das despesas ou até mesmo a dificuldade em encontrar pessoas dispostas a compartilhar o mesmo espaço.

Por outro lado, uma porcentagem considerável dos participantes (33,3%) respondeu que não enfrentou dificuldades na busca por colegas de quarto. Isso indica que, para uma parcela dos entrevistados, encontrar companheiros de moradia compatíveis pode ter sido uma experiência mais tranquila e satisfatória. Esses indivíduos podem ter tido sucesso na identificação de colegas de quarto adequados ou podem ter utilizado estratégias eficazes na seleção de seus parceiros de moradia.

Já a opção "Talvez" escolhida por 20% dos participantes sugere uma ambiguidade em relação às dificuldades enfrentadas. Esses indivíduos podem ter tido algumas dificuldades em sua experiência, mas não em todos os casos, ou podem estar incertos sobre as dificuldades encontradas. Essa resposta também pode indicar a necessidade de aprofundar a investigação sobre os fatores que influenciam a busca por colegas de quarto e as possíveis barreiras encontradas pelos participantes.

Em conclusão, os resultados do gráfico revelam que uma parcela considerável dos participantes enfrentou dificuldades ao tentar encontrar colegas de quarto para dividir o aluguel. Essa constatação ressalta a importância de soluções como a plataforma RoomieLink, que visa facilitar a busca por companheiros de moradia compatíveis. Com o intuito de melhorar a experiência de moradia e promover um convívio harmonioso, é fundamental que estratégias e recursos adequados sejam disponibilizados para ajudar os indivíduos a superar essas dificuldades e encontrar colegas de quarto compatíveis.

Outra pergunta formulada no questionário, "Quais são as principais dificuldades que você enfrenta ao procurar colegas de quarto e moradia?", busca identificar e compreender as principais dificuldades enfrentadas pelos participantes na busca por colegas de quarto e moradia compartilhada. Essa pergunta tem como objetivo obter insights sobre os desafios mais comuns enfrentados nesse processo.

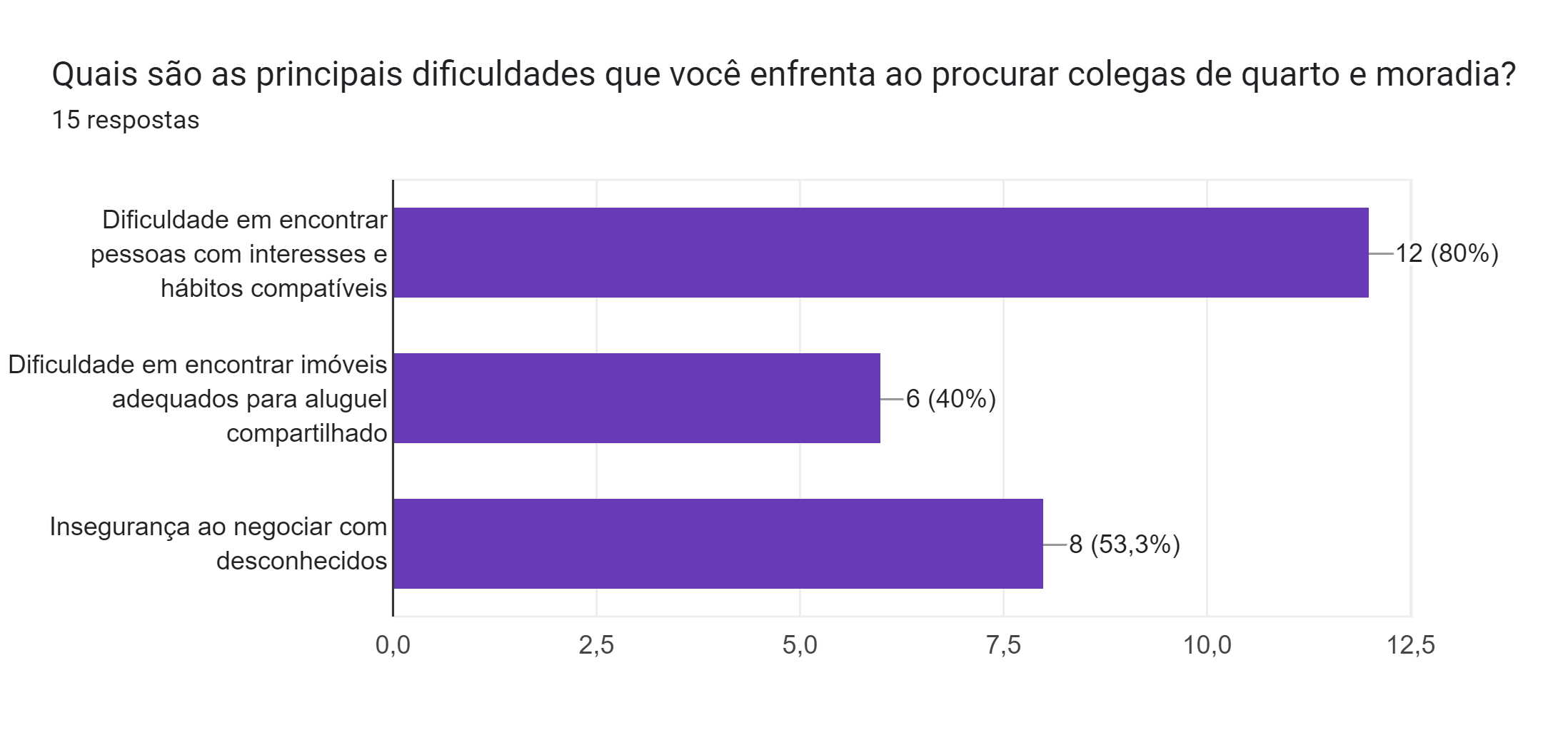


Figura 2 – Principais dificuldades ao procurar colegas de quarto

Após analisar os resultados do gráfico, que apresenta as seguintes porcentagens: Dificuldade em encontrar pessoas com interesses e hábitos compatíveis (80%), Dificuldade em encontrar imóveis adequados para aluguel compartilhado (40%) e Insegurança ao negociar com desconhecidos (53,3%), é possível tirar conclusões importantes sobre as principais dificuldades enfrentadas pelos participantes na busca por colegas de quarto e moradia compartilhada.

Os dados revelam que a maioria dos participantes (80%) aponta a dificuldade em encontrar pessoas com interesses e hábitos compatíveis como a principal dificuldade enfrentada. Esse resultado destaca a importância da compatibilidade entre os futuros colegas de moradia, considerando seus estilos de vida, horários, preferências e atividades diárias. A falta de compatibilidade nessas áreas pode levar a conflitos e impactar negativamente a experiência de moradia compartilhada.

Outra dificuldade mencionada por uma parcela considerável dos participantes (53,3%) é a insegurança ao negociar com desconhecidos. Esse resultado indica que a confiança e a segurança desempenham um papel significativo na escolha dos colegas de quarto. A incerteza em relação às intenções e ao comportamento de pessoas desconhecidas pode criar barreiras na busca por moradia compartilhada.

Em relação à dificuldade em encontrar imóveis adequados para aluguel compartilhado, 40% dos participantes relataram enfrentar esse desafio. Isso ressalta a importância de encontrar imóveis que atendam às necessidades e expectativas de todos os envolvidos, considerando fatores como localização, tamanho, custos e condições do imóvel.

Considerando esses resultados, é evidente que a busca por colegas de quarto e moradia compartilhada apresenta desafios significativos. A falta de compatibilidade entre as pessoas, a insegurança ao negociar com desconhecidos e a dificuldade em encontrar imóveis adequados são aspectos cruciais que precisam ser considerados e abordados de forma eficiente.

Compreender essas dificuldades e suas implicações é essencial para o desenvolvimento de soluções, como a plataforma RoomieLink, que visa facilitar a busca por colegas de quarto compatíveis e imóveis adequados. Ao fornecer recursos e informações que promovam a compatibilidade, a segurança nas negociações e a disponibilidade de imóveis adequados, é possível melhorar a experiência de moradia compartilhada e aumentar a satisfação dos envolvidos.

3.2 Análise de Sistemas Similares

O RoomieLink, como parte de seu compromisso com a qualidade e a eficácia, concentra-se exclusivamente no nicho de estudantes universitários. Suas funcionalidades incluem a busca por quartos disponíveis e colegas de quarto em cidades universitárias, além de recursos para efetuar login, cadastro, edição de perfil e comunicação direta com potenciais colegas de quarto. Como diferencial, o RoomieLink permite iniciar chats, simplificando o processo para estudantes que buscam acomodações compartilhadas.

Em contraste, o Roomgo oferece funcionalidades semelhantes, com foco em buscar quartos e colegas de quarto em várias localidades. É reconhecido por sua interface amigável e funcionalidades padrão, incluindo busca por quartos e colegas de quarto. No entanto, a plataforma pode ser mais básica em comparação com outras opções, faltando características mais avançadas que poderiam aprimorar a experiência do usuário. Além disso, o nicho específico da plataforma pode não estar tão claro.

O WebQuarto, seguindo uma abordagem similar, possibilita aos usuários encontrar quartos em diferentes cidades, oferecendo funcionalidades comuns como busca, cadastro, login e alguns recursos de filtro. Contudo, suas características podem não se equiparar às de plataformas concorrentes em termos de avanço, sugerindo que a plataforma poderia se destacar no mercado de moradia compartilhada ao evidenciar um diferencial mais claro.

O Roomster destaca-se por sua presença global e ampla base de usuários, oferecendo funcionalidades básicas, como busca, cadastro, login e chat. No entanto, em comparação com concorrentes mais específicos, o Roomster pode carecer de recursos avançados e personalização para atender às necessidades específicas dos estudantes universitários.

Por sua vez, o Colliv diferencia-se pela abordagem centrada na comunidade. Apesar de oferecer funcionalidades padrão, como busca e cadastro, características específicas podem não ser tão evidentes. O processo de aprovação de solicitações e o registro de imóveis podem demandar mais informações para proporcionar uma compreensão completa da plataforma.

No entanto, o diferencial do RoomieLink está na sua dedicação ao público universitário, proporcionando uma experiência personalizada e direcionada. A plataforma é projetada para facilitar a busca por moradia compartilhada em cidades universitárias, conectando estudantes que desejam dividir aluguel. O RoomieLink destaca-se ao enfocar as necessidades específicas desse grupo demográfico, tornando-o uma solução ideal para estudantes universitários em busca de moradia compartilhada. Conforme a tabela comparativa a seguir, sobre as principais funcionalidades que há, em cada uma das plataformas, pode-se observar detalhadamente o diferencial do RoomieLink.

Em resumo, embora todas essas plataformas ofereçam funcionalidades valiosas para a busca de moradia compartilhada, o RoomieLink se destaca ao oferecer uma experiência única e direcionada aos estudantes universitários, garantindo que suas necessidades e preferências sejam atendidas de maneira eficaz.

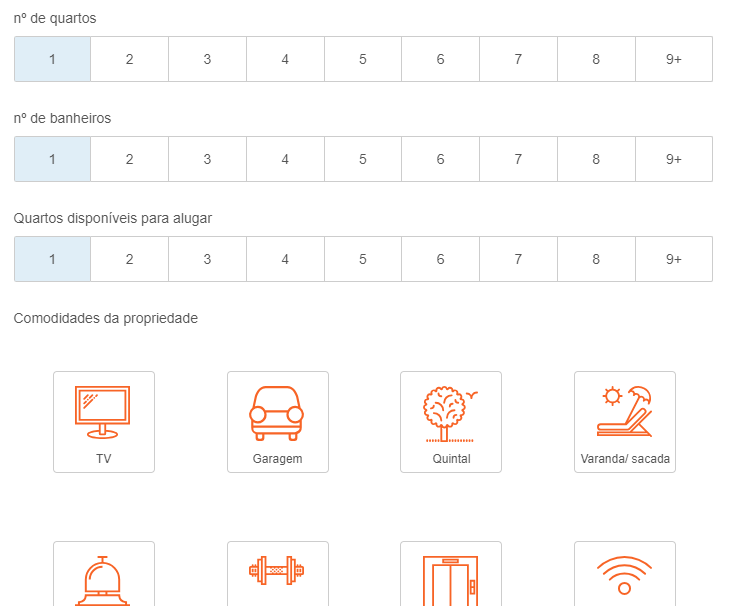
3.2.1 Roomgo

A captura de tela da página de anúncio de quarto do site Roomgo chama a atenção por sua abordagem amigável ao usuário, permitindo que os visitantes preencham informações e dados antes de serem solicitados a fazer o cadastro ou login. Essa funcionalidade proporciona uma experiência de navegabilidade mais fluida e sem interrupções, o que pode ser um diferencial positivo para os usuários.

Figura 3 – Tela home do site Roomgo

A possibilidade de preencher informações e dados antes de se comprometer com um cadastro ou login é um aspecto que valoriza a usabilidade do site. Muitas vezes, ao sermos imediatamente direcionados a criar uma conta ou fazer login, podemos nos sentir pressionados ou desconfortáveis antes mesmo de explorar as funcionalidades oferecidas. No entanto, ao permitir que os usuários insiram informações preliminares, o Roomgo demonstra um cuidado em manter o ritmo e o interesse do usuário, facilitando sua interação com o site.

Essa abordagem valoriza a liberdade de exploração e decisão do usuário, proporcionando uma primeira impressão positiva ao permitir que ele conheça melhor as funcionalidades e recursos do site antes de se comprometer com um cadastro. Além disso, ao preencher informações iniciais, o usuário pode ter uma ideia mais clara de como o site funciona e se atende às suas necessidades, aumentando a probabilidade de se engajar e continuar utilizando a plataforma.

Figura 4 – Tela de anúncio de quarto do site roomgo

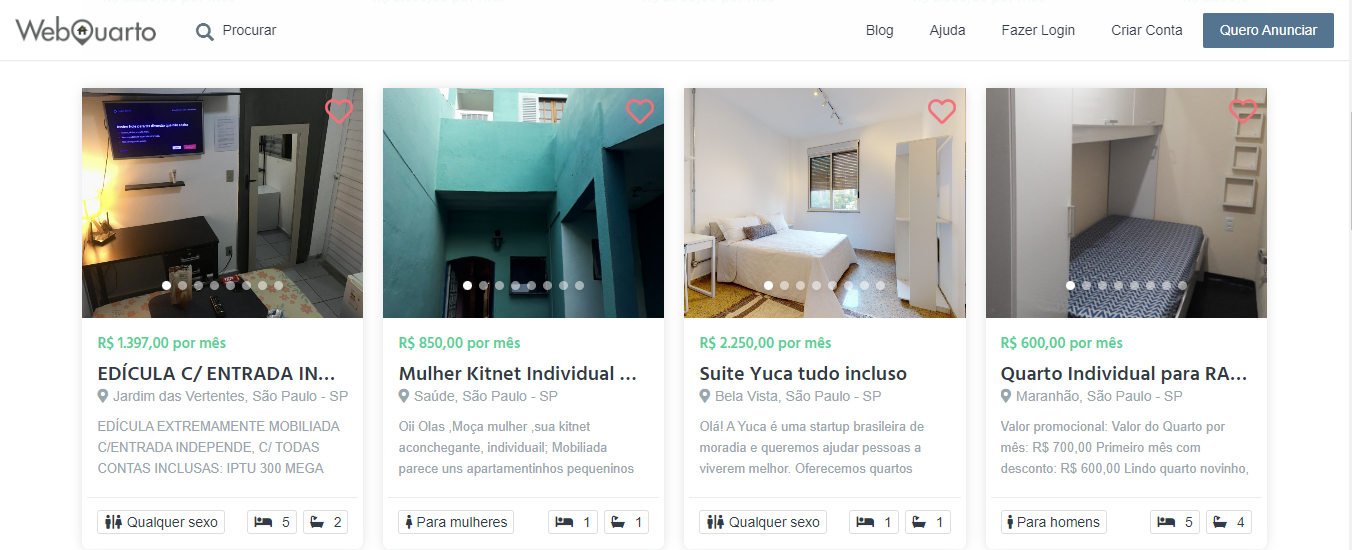
Em conclusão, a captura de tela da página de anúncio de quarto do Roomgo destaca uma abordagem inteligente que permite aos usuários preencher informações e dados antes de serem solicitados a fazer o cadastro ou login. Essa funcionalidade valoriza a usabilidade do site, proporcionando uma experiência de navegabilidade mais fluída e mantendo o interesse do usuário. Ao facilitar a exploração e a interação inicial, o Roomgo promove uma experiência mais satisfatória e convida os usuários a se engajarem com a plataforma.

3.2.2 Webquarto

A imagem da página inicial (home) do site Webquarto revela uma identidade visual marcante, destacando-se por sua aparência e layout distintos. A presença de uma barra de pesquisa centralizada facilita a busca por cidades, tornando o processo mais intuitivo e eficiente para os usuários. Essa abordagem valoriza a usabilidade e a experiência do usuário, permitindo que eles encontrem rapidamente as opções de moradia disponíveis em diferentes localidades.

Figura 5 – Tela home do site WebQuarto

Além disso, a imagem dos cards de anúncios de quartos chama a atenção pela forma como as informações essenciais são apresentadas de maneira clara e concisa. Os cards exibem informações como o valor do aluguel, a localização, o número de banheiros, o número de quartos e o sexo com o qual o anunciante está disposto a compartilhar o espaço. Esses dados são cruciais para os usuários, pois permitem uma rápida avaliação das opções de moradia disponíveis, economizando tempo e facilitando a tomada de decisão.

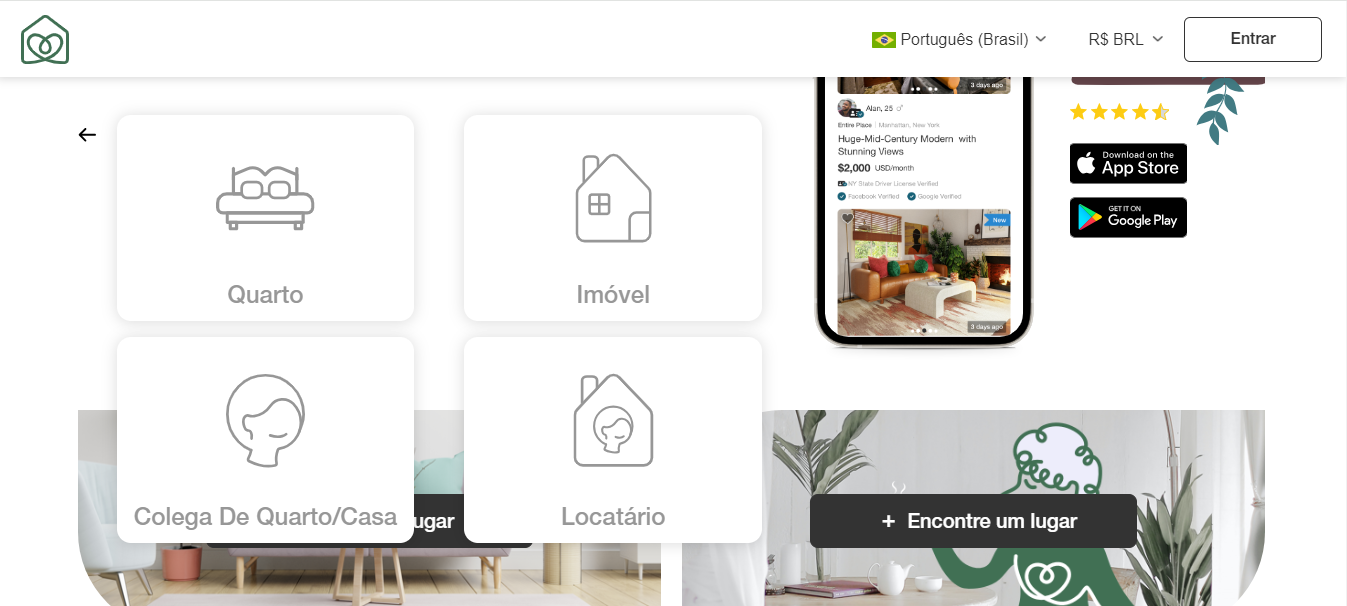
Figura 6 – Tela de anúncios de quartos do site WebQuarto

Essa abordagem do WebQuarto demonstra um cuidado em fornecer aos usuários as informações mais relevantes de forma direta e acessível. Ao apresentar os principais detalhes dos anúncios nos cards, o site facilita a comparação e a seleção de quartos que atendam às preferências e necessidades individuais. Essa eficiência na apresentação das informações é fundamental para os usuários que buscam uma solução prática e ágil na busca por moradia compartilhada.

Em suma, a imagem da página inicial e dos cards de anúncios do WebQuarto revelam uma abordagem centrada na identidade visual e na praticidade. A barra de pesquisa centralizada agiliza a busca por cidades, enquanto os cards dos anúncios apresentam as informações essenciais de forma clara e sucinta. Essa combinação de elementos valoriza a usabilidade do site, tornando-o mais atrativo e eficiente para os usuários em sua busca por quartos para compartilhar.

3.2.3 Roomster

Ao explorar a página inicial do Roomster, destaca-se uma abordagem estratégica e intuitiva para a busca de moradia compartilhada. A barra de pesquisa, centralizada na identificação da cidade desejada, apresenta quatro cards distintos que oferecem categorias específicas, como colegas de quarto, imóveis e locatários, proporcionando uma visão abrangente das opções disponíveis. A organização eficiente facilita a navegação e a seleção da opção mais adequada às necessidades do usuário.

Figura 7 – Tela da página home com as opções de pesquisa do site Roomster

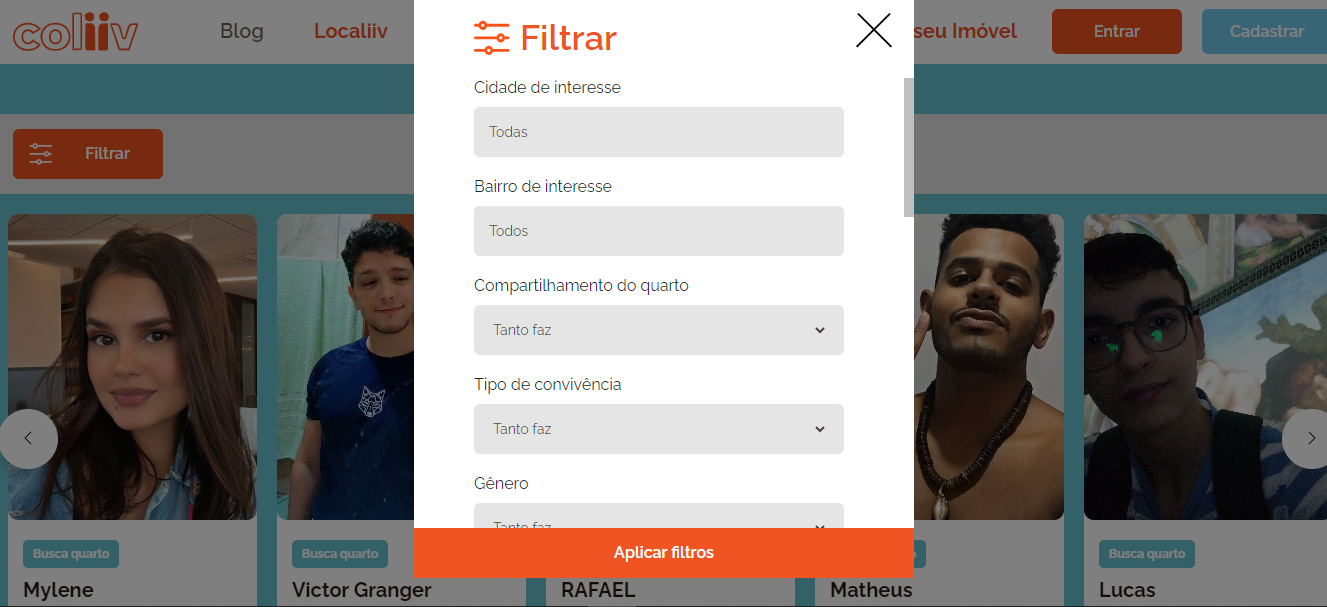
A estética clean e minimalista da página contribui para uma experiência visual agradável, priorizando a clareza das informações e reduzindo qualquer sobrecarga visual. A presença estratégica do botão para download do aplicativo, posicionado no lado direito, promove a acessibilidade, incentivando os usuários a explorarem a versão móvel para uma experiência ainda mais integrada.

Além disso, a disposição equilibrada dos elementos na página inicial reflete o compromisso do Roomster em oferecer uma plataforma amigável e fácil de usar. Essa abordagem otimiza o processo de busca por moradia compartilhada, tornando-o intuitivo e eficiente para os usuários. Com uma combinação inteligente de design, funcionalidade e acessibilidade, o Roomster destaca-se como uma opção atraente para aqueles que buscam uma experiência simplificada na procura por acomodações compartilhadas.

Em síntese, ao mergulhar na página inicial do Roomster, é evidente que a plataforma se destaca pela sua abordagem estratégica e intuitiva na busca por moradia compartilhada. A organização eficiente, aliada à estética clean e minimalista, proporciona aos usuários uma experiência visualmente agradável e funcional. A presença estratégica do botão de download do aplicativo reforça o compromisso com a acessibilidade e a integração móvel.

3.2.4 Coliiv

Explorando o filtro abrangente da plataforma Coliv, é perceptível o compromisso em oferecer aos usuários uma experiência de busca personalizada e detalhada. A diversidade de opções disponíveis no filtro reflete a preocupação da Coliv em atender às variadas necessidades e preferências dos indivíduos que buscam compartilhar moradias ou encontrar colegas de quarto.

Figura 8 – Tela de filtragem de colegas de quarto do site Coliiv

Ao permitir a especificação da cidade e bairro de interesse, a Coliv facilita a busca por opções próximas e alinhadas às preferências geográficas dos usuários. A inclusão de critérios como compartilhamento do quarto, tipo de convivência, gênero, orientação sexual, política, religião, escolaridade, dieta vegana, hábito de fumar, faixa etária e disposição financeira acrescenta uma camada de detalhes que vai além do convencional, proporcionando uma busca mais refinada e ajustada às preferências individuais.

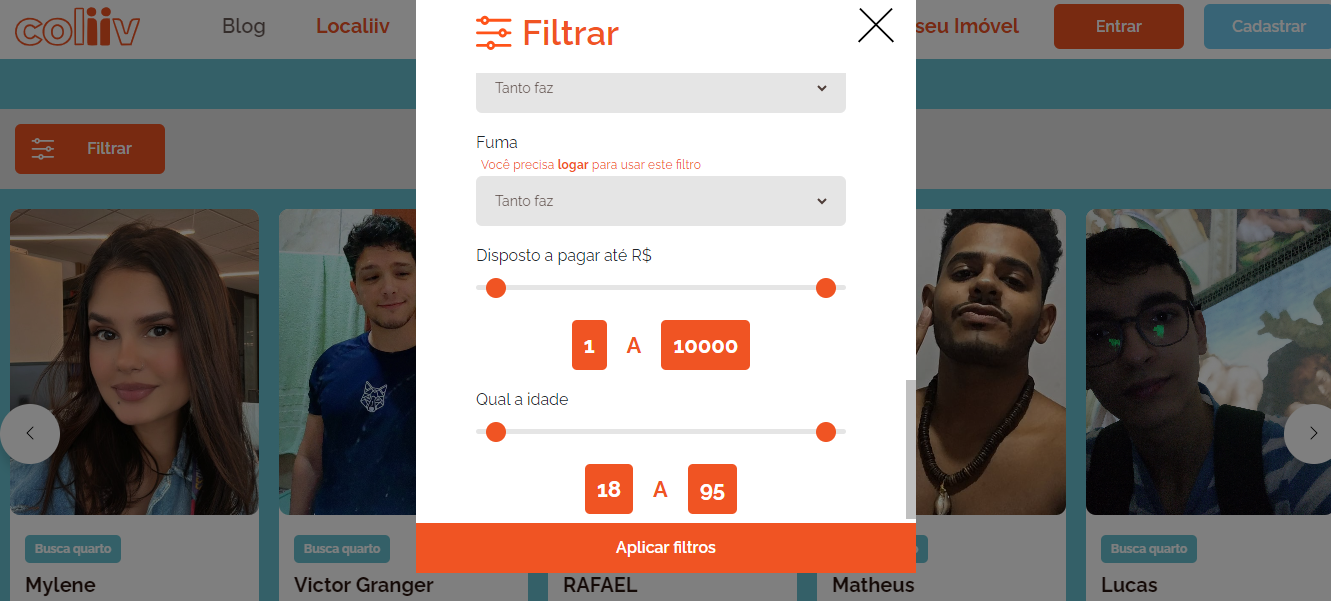


Figura 9 – Continuação da tela de filtragem de colegas de quarto do site Coliiv

Essa abordagem abrangente evidencia a intenção da Coliv de não apenas facilitar a busca por moradia compartilhada, mas também de promover uma experiência de convivência mais alinhada com as preferências de estilo de vida de cada usuário. Com um filtro tão completo, a Coliv se destaca como uma opção que valoriza a individualidade e proporciona aos seus usuários a possibilidade de encontrar moradias e colegas de quarto de acordo com critérios específicos e personalizados.

O filtro abrangente de imóveis oferecido pela Coliv demonstra um compromisso em atender às diversas necessidades e preferências dos usuários que buscam moradias compartilhadas. Ao permitir que os usuários definam suas preferências de aluguel, compartilhamento do quarto e banheiro, bem como especificidades sobre o imóvel e o condomínio desejados, a Coliv proporciona uma busca mais personalizada e alinhada com as expectativas individuais.

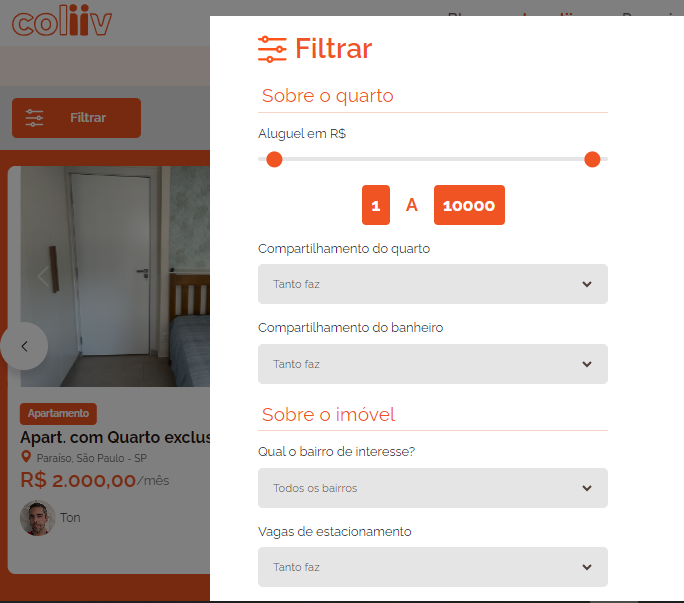
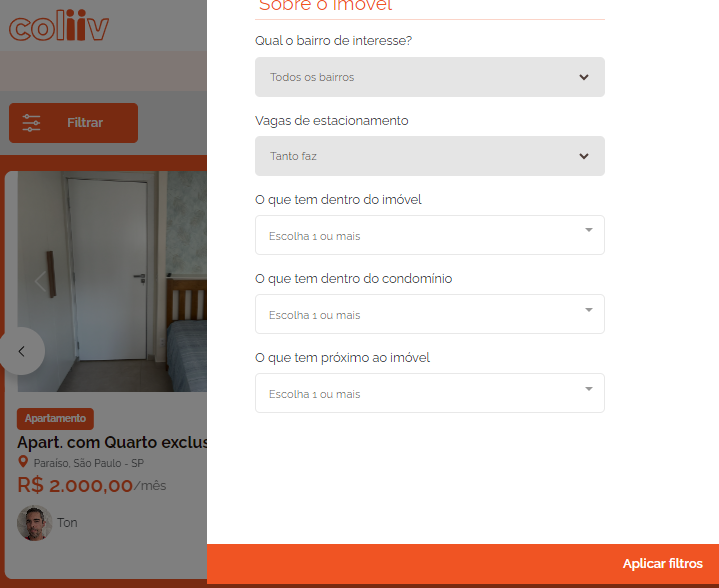


Figura 10 - Tela de filtragem de imóveis do site Coliiv

A opção de escolher o bairro de interesse e indicar preferências quanto a vagas de estacionamento, o que está dentro do imóvel e do condomínio, assim como o que está próximo ao imóvel, adiciona camadas adicionais de detalhes à busca. Esses recursos são fundamentais para que os usuários encontrem opções que não apenas atendam aos critérios básicos, mas que também estejam de acordo com suas preferências específicas em relação ao local, comodidades e ambiente ao redor da moradia.

Figura 11 – Continuação da tela de filtragem de imóveis do site Coliiv

Dessa forma, a Coliv destaca-se ao oferecer um filtro completo e flexível para a busca de imóveis, considerando não apenas aspectos financeiros, mas também características específicas que contribuem para uma experiência de moradia compartilhada mais personalizada e satisfatória para seus usuários.

3.3 Tabela comparativa

Para embasar a escolha das tecnologias e funcionalidades do RoomieLink, realizou-se uma análise comparativa com plataformas similares no mercado, incluindo Roomster, Roomgo, WebQuarto e Colliv. A comparação visa identificar as nuances que diferenciam o RoomieLink no nicho específico de busca por moradia compartilhada entre universitários. A tabela abaixo destaca as principais funcionalidades dessas plataformas:

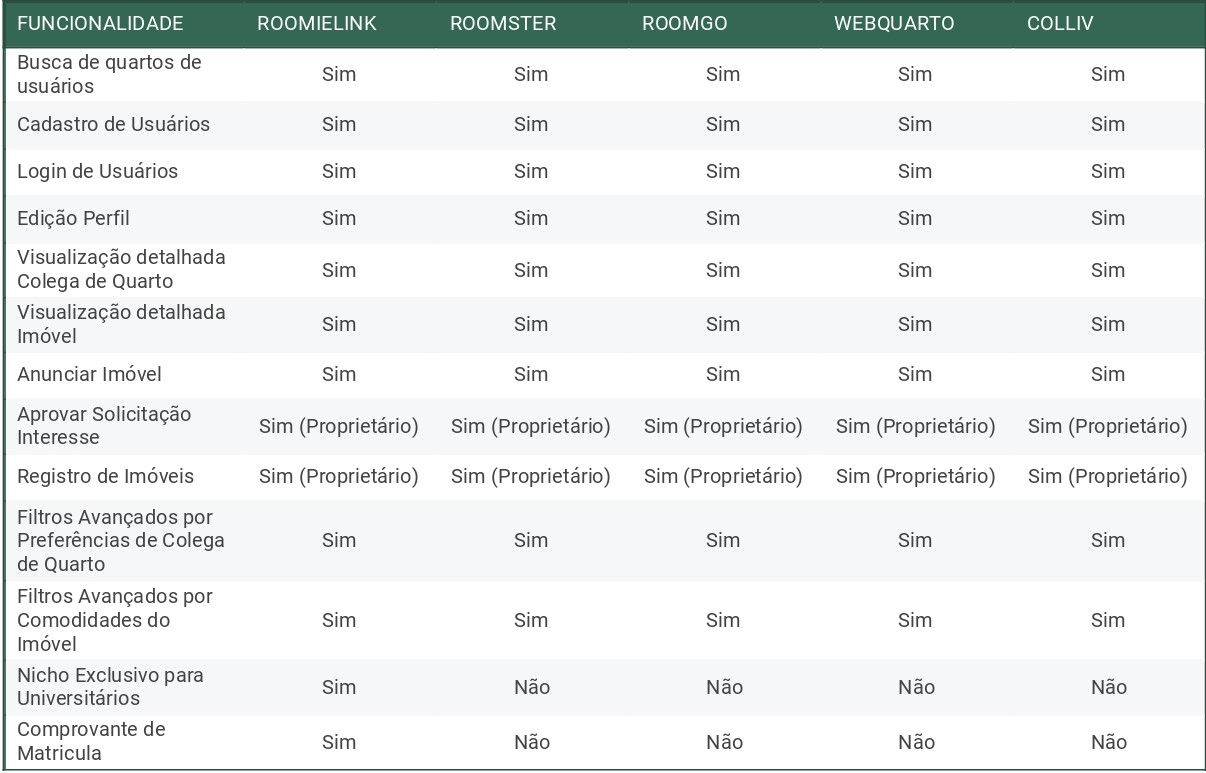


Tabela 1 – Tabela comparativa dos sistemas similares

A escolha dessas plataformas para comparação justifica-se pela representatividade no cenário de moradia compartilhada e busca por colegas de quarto. Ao destacar o diferencial do RoomieLink, direcionado especificamente para universitários, a análise objetiva identificar as lacunas nas concorrentes e fortalecer a proposta de valor do RoomieLink no mercado.

A análise comparativa revela que o RoomieLink apresenta funcionalidades alinhadas às plataformas concorrentes, permitindo a busca de quartos, cadastro, login, edição de perfil, chat entre usuários, visualização detalhada e anúncio de imóveis. A inclusão de filtros avançados personaliza a busca, enquanto as funcionalidades específicas para proprietários, como aprovação e registro de imóveis, promovem a experiência completa na plataforma.

A distinção crucial encontra-se no nicho universitário, destacando o RoomieLink como uma opção dedicada a atender às necessidades específicas desse público. A funcionalidade única de filtrar por universitários e ações específicas para proprietários evidenciam o foco na praticidade e eficácia para essa comunidade.

Em conclusão, a análise revela que o RoomieLink alinha-se às melhores práticas do setor, superando as plataformas gerais ao oferecer uma abordagem especializada. Ao estabelecer um ambiente direcionado e inclusivo para universitários, o RoomieLink visa proporcionar uma experiência única e eficiente, reforçando sua posição no mercado de moradia compartilhada.

**4 SISTEMA ROOMIELINK**

A escolha do tema sobre a busca por moradia compartilhada e os desafios enfrentados por universitários é motivada pela crescente importância desse assunto. A transição para a vida universitária muitas vezes envolve a necessidade de encontrar acomodações adequadas, e a busca por colegas de quarto compatíveis pode ser uma tarefa desafiadora. Nesse contexto, o RoomieLink surge como uma solução inovadora que promete simplificar o processo de busca por colegas de quarto e opções de moradia adequadas. Compreender a eficácia e os benefícios dessa plataforma é fundamental para melhorar a experiência de moradia dos estudantes universitários, contribuindo para um convívio harmonioso e uma vivência mais satisfatória. Ao analisar a eficácia do RoomieLink, é possível identificar como essa plataforma pode facilitar o encontro de colegas de quarto com interesses e estilos de vida semelhantes, aumentando as chances de uma convivência agradável e harmoniosa. Além disso, a plataforma oferece recursos que permitem aos usuários pesquisar opções de moradia com base em critérios específicos, como localização e preferências de estilo de vida, proporcionando uma busca mais direcionada e eficiente. Através do RoomieLink, os estudantes também podem se comunicar e agendar visitas aos locais de moradia, facilitando o processo de tomada de decisão. Isso possibilita que os estudantes conheçam melhor os potenciais colegas de quarto e tenham a oportunidade de tirar dúvidas antes de tomar uma decisão final. Em suma, compreender a eficácia e os benefícios do RoomieLink contribui para a melhoria da experiência de moradia dos universitários, tornando o processo de busca por colegas de quarto e opções de moradia mais eficiente e satisfatório.

4.1 Levantamento de Requisitos do RoomieLink

O levantamento de requisitos é um processo crucial no desenvolvimento de sistemas de software, desempenhando um papel fundamental na definição e compreensão das necessidades dos usuários e das funcionalidades do sistema a ser desenvolvido. De acordo com Gileane Guedes em seu livro "UML 2: Uma Abordagem Prática", o levantamento de requisitos é uma das primeiras etapas no ciclo de vida do desenvolvimento de software, e sua importância não pode ser subestimada.

Através do levantamento de requisitos, os analistas e desenvolvedores de software têm a oportunidade de coletar informações detalhadas sobre o que o sistema deve fazer e como ele deve funcionar. Isso envolve a identificação de requisitos funcionais, que descrevem as principais funcionalidades do sistema, e requisitos não funcionais, que incluem aspectos como desempenho, segurança e usabilidade.

Os requisitos funcionais descrevem as interações entre os usuários e o sistema, delineando as ações que os usuários poderão executar e os resultados esperados. Por exemplo, no contexto do RoomieLink, um requisito funcional seria a capacidade dos usuários comuns de realizar buscas por imóveis disponíveis na plataforma.

Já os requisitos não funcionais dizem respeito às características que não estão diretamente ligadas às funcionalidades do sistema, mas que são igualmente essenciais para o seu sucesso. Eles incluem aspectos como desempenho, segurança, escalabilidade e usabilidade. Um requisito não funcional relevante para o RoomieLink seria a necessidade de a plataforma ser responsiva, funcionando bem em dispositivos móveis e desktop.

O levantamento de requisitos é uma etapa contínua e iterativa, à medida que os analistas de sistema trabalham em estreita colaboração com os stakeholders para garantir que todos os requisitos sejam compreendidos e documentados de forma precisa. Uma vez concluído, o levantamento de requisitos serve como base para o design, desenvolvimento e teste do sistema, garantindo que o software atenda às expectativas dos usuários e aos objetivos do projeto.

A tabela de levantamento de requisitos é uma ferramenta valiosa utilizada no processo de desenvolvimento de software para documentar e organizar os requisitos funcionais e não funcionais de um sistema. No contexto do RoomieLink, essa tabela foi criada para identificar e detalhar os requisitos essenciais para o funcionamento da plataforma de busca de moradia compartilhada.

|  |  |
| --- | --- |
| ID | Descrição do Requisito |
| RF1 | O usuário comum deve poder realizar buscas por imóveis na plataforma, com possibilidade de filtros específicos para estudantes, como proximidade com instituições de ensino. |
| RF2 | O usuário comum deve ser capaz de efetuar o cadastro na plataforma. |
| RF3 | O usuário comum deve poder efetuar login na plataforma. |
| RF4 | O usuário comum deve poder editar seu perfil, como preferências de convivência e informações acadêmicas como comprovante de matricula. |
| RF5 | O usuário comum pode ver detalhes de um imóvel, caso tenha realizado uma busca por imóveis. |
| RF6 | Usuários comuns podem ver detalhes de outros usuários, incluindo preferências de convivência, localização e características pessoais relevantes para convivência em moradia compartilhada. |
| RF7 | O usuário comum pode iniciar um chat com um proprietário de imóvel ou outro usuário após efetuar login. |
| RF8 | O usuário comum pode agendar uma visita no imóvel após visualizar seus detalhes. |
| RF9 | O usuário comum poderá fazer avaliações e comentários nos imóveis, proprietários e colegas de quarto. |

Tabela 2 - Requisitos Funcionais para o Usuário Comum.

A tabela de Requisitos Funcionais para o Usuário Comum é uma parte essencial do processo de levantamento de requisitos do RoomieLink, pois detalha todas as funcionalidades e ações que os usuários comuns poderão realizar na plataforma.Esses requisitos funcionais são cuidadosamente elaborados para atender às necessidades dos estudantes em busca de moradia compartilhada, permitindo-lhes realizar ações como buscar quartos disponíveis, efetuar login para acessar a plataforma, cadastrar-se como usuários, editar seus perfis para fornecer informações pessoais relevantes, visualizar detalhes de imóveis e perfis de outros usuários para avaliar a compatibilidade, iniciar chats com potenciais colegas de quarto para esclarecer dúvidas e discutir detalhes e, por fim, alugar um imóvel diretamente através da plataforma. Esses requisitos funcionais são essenciais para proporcionar aos usuários uma experiência completa e satisfatória ao utilizar o RoomieLink, facilitando a busca por moradia compartilhada e a conexão com colegas de quarto compatíveis.

A tabela de Requisitos Funcionais para o Proprietário é um componente fundamental do levantamento de requisitos do RoomieLink, uma vez que delineia todas as funcionalidades específicas disponíveis para os proprietários de imóveis na plataforma. Estes requisitos funcionais são meticulosamente elaborados para atender às necessidades dos proprietários, permitindo-lhes interagir de maneira eficaz com a plataforma. As funcionalidades incluem a capacidade de registrar imóveis para aluguel, desde a criação do anúncio até a inserção de detalhes relevantes sobre a propriedade, a possibilidade de aprovar solicitações de aluguel dos potenciais inquilinos após ajustes e negociações, e a opção de anunciar imóveis na plataforma para um público mais amplo. Eles visam proporcionar aos proprietários uma experiência abrangente e eficiente na promoção de suas propriedades, facilitando o processo de compartilhamento de moradia com estudantes em busca de acomodação. Eles são essenciais para garantir que os proprietários possam aproveitar ao máximo os recursos do RoomieLink e encontrar inquilinos compatíveis.

|  |  |
| --- | --- |
| ID | Descrição do Requisito |
| RF9 | O proprietário poderá registrar um imóvel na plataforma. |
| RF10 | O proprietário poderá anunciar um imóvel na plataforma. |
| RF11 | O proprietário poderá informar na plataforma os dias disponíveis para visita. |
| RF12 | O proprietário poderá aprovar solicitações de visitas feitas por usuários, com possibilidade de ajustes. |
| RF13 | O proprietário pode realizar alterações nos dados do imóvel anunciado, como descrição, fotos, valores, etc. |

Tabela 3 - Requisitos Funcionais para o Proprietário

Ao considerar os requisitos funcionais adicionais para os proprietários na plataforma RoomieLink, percebe-se a importância de oferecer ferramentas que facilitem a gestão e interação dos mesmos com potenciais locatários. A possibilidade de definir condições de aluguel, gerenciar visitas, interagir por chat e monitorar estatísticas são aspectos cruciais para garantir uma experiência completa e eficiente aos proprietários, contribuindo para a transparência e eficácia do sistema. Com essas funcionalidades, espera-se não apenas atender às necessidades dos proprietários, mas também fortalecer a confiança e engajamento deles na plataforma.

Em conclusão, os requisitos funcionais propostos para os proprietários no RoomieLink visam não apenas oferecer um ambiente amigável e intuitivo para a divulgação de imóveis, mas também proporcionar ferramentas que promovam uma comunicação eficaz. Esses recursos são fundamentais para criar um ecossistema digital que beneficie tanto os proprietários quanto os usuários em busca de moradia compartilhada.

A tabela de Requisitos Não Funcionais é uma parte crucial do levantamento de requisitos do RoomieLink, pois aborda aspectos que não dizem respeito diretamente às funcionalidades do sistema, mas sim a características essenciais para o seu desempenho, usabilidade e segurança. Nessa tabela, são definidos requisitos como desempenho, escalabilidade, usabilidade, segurança e confiabilidade do aplicativo. Esses requisitos não funcionais garantem que o RoomieLink seja rápido e responsivo, capaz de lidar com um número crescente de usuários, de fácil utilização para diferentes públicos, seguro para proteger informações sensíveis dos usuários e confiável para garantir que as operações sejam consistentes e livres de erros. Portanto, a tabela de Requisitos Não Funcionais é fundamental para assegurar que o RoomieLink atenda aos mais altos padrões de qualidade e satisfaça as necessidades de seus usuários com eficácia.

|  |  |
| --- | --- |
| ID | Descrição do Requisito |
| RNF1 | A plataforma deve ser responsiva, funcionando bem em dispositivos móveis e desktop. |
| RNF2 | A plataforma deve garantir a segurança dos dados dos usuários, incluindo senhas e informações pessoais. |
| RNF3 | A plataforma deve ser de fácil utilização e navegação, com uma interface amigável. |
| RNF4 | A plataforma deve ser escalável, permitindo o aumento do número de imóveis e usuários. |
| RNF5 | A plataforma deve ser de alta disponibilidade, minimizando tempos de inatividade não planejados. |
| RNF6 | A plataforma deve ser rápida e responsiva, com baixo tempo de carregamento de páginas. |
| RNF7 | A plataforma deve ser compatível com múltiplos navegadores e sistemas operacionais. |
| RNF8 | A plataforma deve garantir a privacidade dos chats entre usuários. |

Tabela 4 – Requisitos Não Funcionais

Os requisitos funcionais listados na tabela descrevem as ações e funcionalidades que os usuários comuns e proprietários de imóveis poderão executar no sistema. Isso inclui a capacidade de realizar buscas por imóveis, efetuar login, cadastrar-se na plataforma, editar perfis, visualizar detalhes de imóveis e usuários, iniciar chats e alugar imóveis. Cada requisito funcional é identificado por um número de ID exclusivo para referência futura.

Já os requisitos não funcionais estabelecem critérios e restrições que moldam o desempenho, segurança e usabilidade do RoomieLink. Esses requisitos incluem aspectos como responsividade da plataforma, segurança dos dados dos usuários, escalabilidade, disponibilidade, desempenho e compatibilidade com diferentes navegadores e sistemas operacionais.

Essa tabela é uma ferramenta dinâmica que pode ser atualizada e expandida à medida que novos requisitos surgem ou à medida que o projeto avança. Sua estrutura clara e organizada ajuda a manter o foco nas metas e necessidades dos usuários, contribuindo para o sucesso do desenvolvimento do RoomieLink.

5 MODELAGEM DO SISTEMA

A modelagem de sistemas é uma etapa crucial no desenvolvimento de software, que permite representar e comunicar visualmente os diferentes aspectos e interações do sistema em questão. Nesse contexto, três diagramas frequentemente utilizados são o Diagrama de Caso de Uso, o Diagrama de Classes e o Diagrama de Sequência. Essas ferramentas fornecem uma representação clara e concisa das funcionalidades, estrutura e comportamento do sistema, contribuindo para uma compreensão abrangente e detalhada do RoomieLink.

O Diagrama de Caso de Uso, proposto por Grady Booch em seu livro "Object-Oriented Analysis and Design with Applications", é uma ferramenta amplamente utilizada para capturar os requisitos funcionais do sistema. Ele identifica os atores (usuários, sistemas externos) e os casos de uso (funcionalidades) do sistema, mostrando as interações entre eles. No contexto do RoomieLink, o Diagrama de Caso de Uso poderia representar as principais funcionalidades do sistema, como a criação de perfil do usuário, a busca de quartos disponíveis e a solicitação de contato com potenciais colegas de quarto.

O Diagrama de Classes, também proposto por Booch, é utilizado para descrever a estrutura do sistema, mostrando as classes, seus atributos e relacionamentos. No RoomieLink, esse diagrama poderia representar as entidades do sistema, como Usuário, Quarto, Anúncio, entre outras, e as associações entre elas. Ele proporciona uma visão detalhada das entidades e de como elas interagem no contexto do sistema.

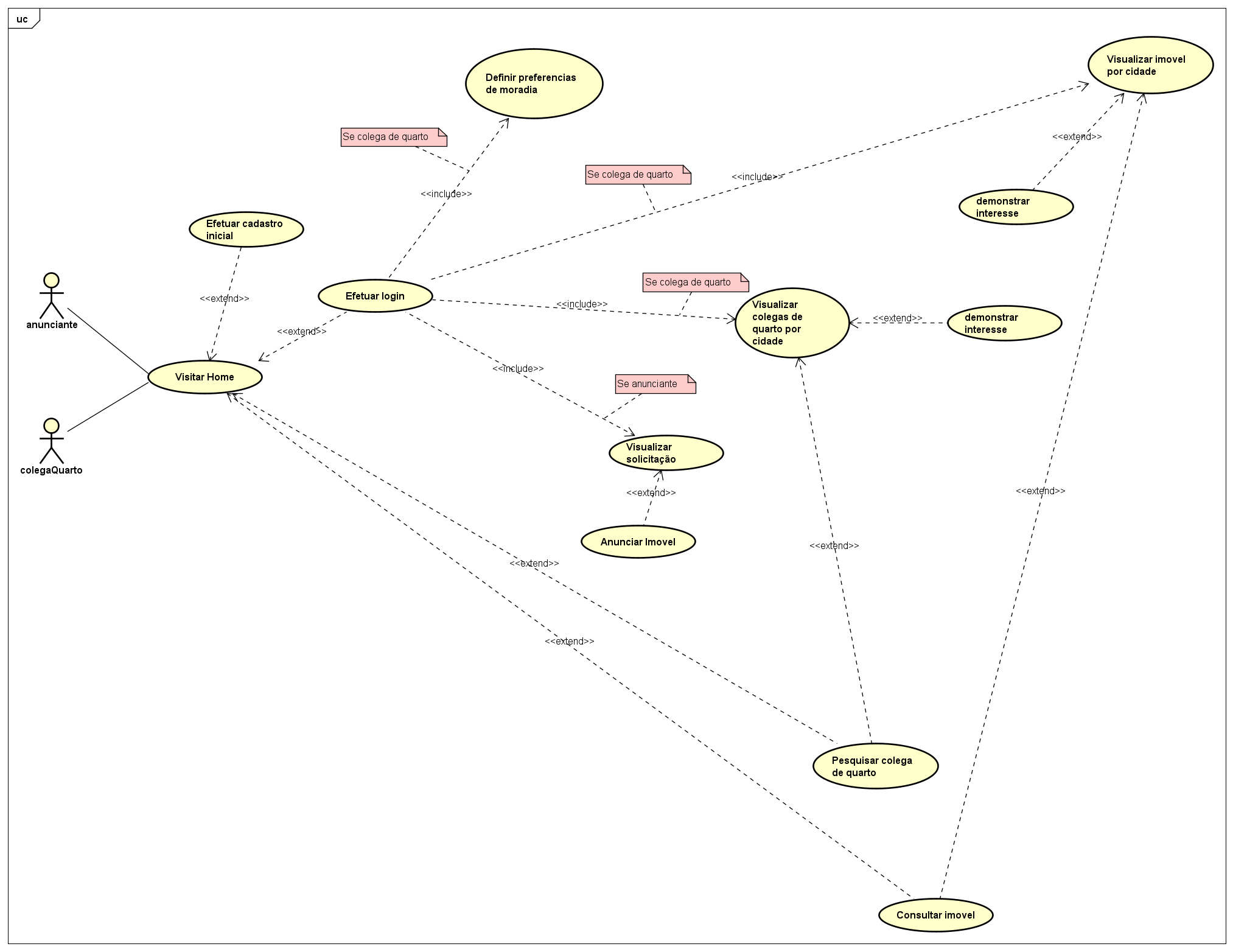
O Diagrama de Sequência é utilizado para descrever a interação entre objetos em uma determinada funcionalidade do sistema, mostrando a ordem e o fluxo de mensagens trocadas entre eles. No contexto do RoomieLink, um Diagrama de Sequência poderia representar o processo de solicitação de contato entre usuários interessados em compartilhar um quarto, ilustrando como as mensagens são trocadas e como o sistema responde a essas interações.

Em suma, a modelagem de sistemas, por meio dos diagramas de Caso de Uso, Classes e Sequência, desempenha um papel fundamental no entendimento, análise e comunicação das diferentes perspectivas do RoomieLink. Essas ferramentas, propostas por Grady Booch e amplamente adotadas na engenharia de software, auxiliam na criação de sistemas eficientes, compreensíveis e alinhados com as necessidades dos usuários. Através da utilização desses diagramas, é possível visualizar de forma clara e precisa as funcionalidades, estrutura e comportamento do sistema, contribuindo para o desenvolvimento de um sistema de moradia compartilhada bem-sucedido.

5.1 Diagrama de Caso de Uso

O Diagrama de Caso de Uso é uma importante ferramenta da UML (Unified Modeling Language) amplamente utilizada na modelagem de sistemas para capturar os requisitos funcionais do sistema e as interações entre os atores e as funcionalidades oferecidas. No contexto do RoomieLink, o Diagrama de Caso de Uso desempenha um papel fundamental ao representar as diversas funcionalidades que permitem aos usuários buscar quartos e encontrar potenciais colegas de quarto com base na cidade digitada.

Além disso, para auxiliar na criação do Diagrama de Caso de Uso, foi utilizado o astah, uma ferramenta de desenho e diagramação amplamente reconhecida por sua facilidade de uso e versatilidade na criação de diagramas UML e outros tipos de diagramas técnicos. A imagem do Diagrama de Caso de Uso específico do RoomieLink está representando de maneira visual as interações entre atores, casos de uso e suas respectivas funcionalidades.

Figura 12 – Diagrama de Caso de Uso

De acordo com Gileane Guedes em seu livro "UML 2: uma abordagem prática", o Diagrama de Caso de Uso é composto por atores, casos de uso e suas respectivas interações. No caso do RoomieLink, os atores podem incluir os usuários do sistema, como estudantes em busca de moradia compartilhada, e o próprio sistema. Os casos de uso representam as funcionalidades disponíveis, como "Buscar quartos disponíveis" e "Buscar usuários por cidade".

Um exemplo prático dentro do RoomieLink é o caso de uso "Buscar usuários/quartos por cidade". Neste caso, o usuário informa a cidade desejada, e o sistema realiza a busca de usuários ou quartos disponíveis nessa localidade. Essa funcionalidade permite aos usuários filtrar os resultados com base na cidade de interesse, facilitando a busca por opções de moradia em uma área específica.

Em suma, o Diagrama de Caso de Uso desempenha um papel importante no RoomieLink ao representar as funcionalidades do sistema e as interações com os usuários. Através da funcionalidade de busca por cidade, os usuários podem encontrar quartos disponíveis ou outros usuários interessados em compartilhar moradia em uma área específica, oferecendo uma experiência personalizada e eficiente.

5.2 Diagrama de Classes

O Diagrama de Classes é uma poderosa ferramenta da UML (Unified Modeling Language) que permite visualizar a estrutura estática de um sistema, representando as classes, seus atributos, métodos e relacionamentos. No contexto do RoomieLink, o Diagrama de Classes desempenha um papel fundamental ao representar a estrutura das classes envolvidas no sistema, como a classe Usuário, Estudante e Anunciante, bem como suas relações e interações.

De acordo com Gileane Guedes em seu livro "UML 2: uma abordagem prática", o Diagrama de Classes é composto por classes, atributos, métodos e relacionamentos. No caso do RoomieLink, a classe Usuário é uma classe geral que engloba todas as informações comuns a todos os tipos de usuários do sistema, como nome, e-mail e senha. Essa classe está relacionada com outras duas classes específicas, Estudante e Anunciante.

A classe Estudante representa os usuários do sistema que buscam por colegas de quarto. Já a classe Imóvel, também possui uma relação com a classe Anunciante, que representa os usuários que anunciam o imóvel para compartilhar.

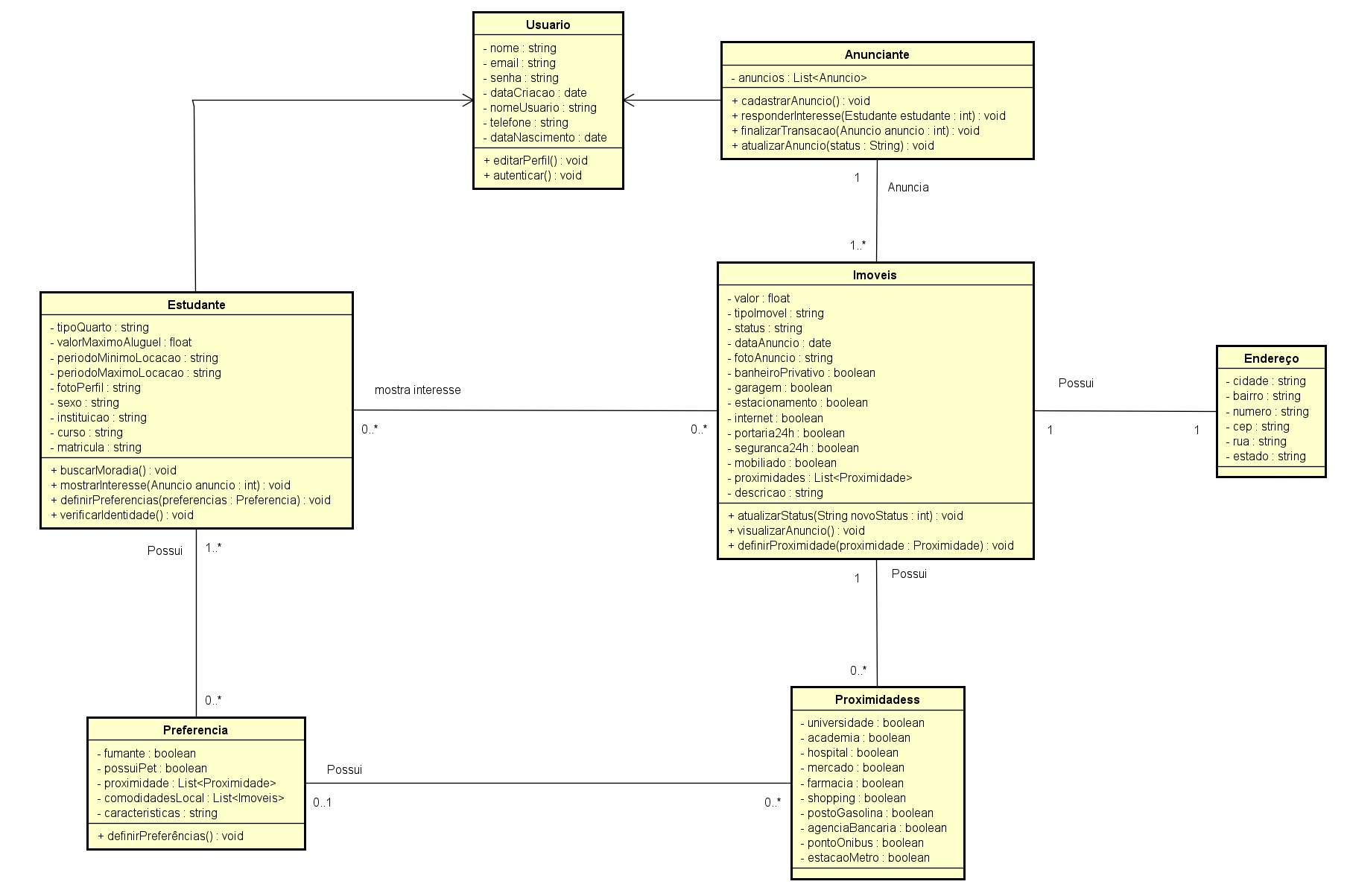


Figura 13 – Diagrama de Classes

Dessa forma, o Diagrama de Classes do RoomieLink representa a estrutura das classes envolvidas no sistema, suas associações e as informações que cada classe contém. Essa representação visual fornece uma visão clara e concisa da estrutura estática do sistema, permitindo a compreensão das entidades envolvidas e suas interações.

No desenvolvimento do Diagrama de Classes, é possível utilizar ferramentas de modelagem, como o Eclipse, que oferecem suporte para a criação e visualização dos diagramas. O Eclipse permite a criação e manipulação dos elementos do Diagrama de Classes, facilitando a representação e a compreensão da estrutura do sistema.

Em resumo, o Diagrama de Classes é uma ferramenta essencial na modelagem do RoomieLink, permitindo a visualização da estrutura estática do sistema e as relações entre as classes. Essa representação auxilia na compreensão da organização do sistema, facilitando o desenvolvimento e a manutenção do software.

6 design

**7 TECNOLOGIAS**

O desenvolvimento de um sistema como o RoomieLink, projetado para facilitar o compartilhamento de moradia entre estudantes universitários, exige a escolha de tecnologias que combinem eficiência, escalabilidade e confiabilidade. A implementação cuidadosa dessas ferramentas é essencial para garantir uma experiência de usuário intuitiva e um desempenho consistente, mesmo diante do aumento no volume de dados e usuários.

Para atender às demandas do projeto, foi necessário planejar uma arquitetura tecnológica robusta, que pudesse sustentar o crescimento da aplicação e permitir a integração entre diferentes camadas do sistema. Nesse contexto, as tecnologias adotadas foram divididas em categorias essenciais, cada uma desempenhando um papel específico na implementação e manutenção do RoomieLink. Desde a lógica de negócios no backend até as interações diretas com os usuários no frontend, cada decisão tecnológica foi fundamentada em critérios de eficiência, segurança e capacidade de adaptação ao futuro.

Além disso, o desenvolvimento do banco de dados exigiu uma modelagem cuidadosa, baseada em diagramas de caso de uso e diagramas de classe. Essa abordagem permitiu estruturar os dados de forma a atender às operações críticas do sistema, utilizando ferramentas modernas que garantem desempenho e confiabilidade.

Nesta seção, serão detalhadas as tecnologias utilizadas no RoomieLink, organizadas em categorias como backend, frontend, controle de versão, modelagem de banco de dados e hospedagem em nuvem. Cada subseção abordará as ferramentas escolhidas, suas principais funcionalidades e como elas se alinham aos objetivos do projeto, garantindo a entrega de um sistema eficiente e preparado para escalar.

7.1 Backend em Sistemas de Informação

O backend é a base invisível de qualquer sistema de informação, responsável por lidar com a lógica de negócios, armazenar e manipular dados, além de gerenciar a comunicação entre o frontend e os servidores. Ele funciona como um motor que processa solicitações dos usuários, realiza cálculos, acessa bases de dados e retorna as respostas apropriadas para que as interfaces visíveis funcionem de forma eficiente. De acordo com Tilkov e Vinoski (2010), o backend também é fundamental para garantir a segurança e escalabilidade de sistemas modernos, sendo projetado para lidar com múltiplas conexões simultâneas e manter um desempenho consistente mesmo em situações de alta demanda.

Uma das tecnologias que se destaca na construção de backends robustos e escaláveis é o Node.js. Com sua arquitetura orientada a eventos e capacidade de executar JavaScript no lado do servidor, o Node.js oferece uma solução eficiente para sistemas que requerem alta performance em operações de entrada e saída (I/O). No projeto, o Node.js foi escolhido para implementar o backend devido à sua facilidade de integração com outras tecnologias, como o Prisma e o PostgreSQL, e por possibilitar um desenvolvimento ágil e modular. A ampla biblioteca de pacotes disponíveis no npm (Node Package Manager) também contribui para acelerar a implementação de funcionalidades essenciais.

O backend do projeto desempenha um papel central ao conectar diferentes camadas do sistema e garantir que operações críticas, como a autenticação de usuários e a busca por moradias, sejam realizadas de forma segura e eficiente. Utilizando o Prisma como ORM e o PostgreSQL como banco de dados relacional, o backend do sistema combina tecnologias modernas para oferecer um sistema confiável e escalável. Essa combinação permite que o aplicativo armazene e processe grandes volumes de dados com rapidez e precisão, atendendo às necessidades dos usuários e suportando o crescimento futuro da plataforma.

A infraestrutura lógica desempenha um papel central na definição da lógica de negócios, garantindo uma comunicação eficiente entre as diferentes camadas do sistema e o gerenciamento seguro dos dados. A escolha das tecnologias foi pautada na busca por escalabilidade, desempenho e integração com outras ferramentas modernas. Cada decisão tecnológica contribuiu para criar uma base sólida para o projeto, permitindo que o sistema atenda às demandas de seus usuários de forma eficiente. Na próxima subseção, exploraremos em detalhes o papel do Node.js, tecnologia-chave para a construção do que foi relatado até aqui.

7.1.1 Node.js

O Node.js é uma das tecnologias mais populares para desenvolvimento de backend devido à sua capacidade de lidar com aplicações escaláveis e de alto desempenho. Baseado no motor V8 do Google Chrome, ele utiliza JavaScript no lado do servidor, unificando a linguagem de programação em todo o projeto e facilitando a comunicação entre backend e frontend. Segundo Tilkov e Vinoski (2010), a arquitetura orientada a eventos e a utilização de uma thread única tornam o Node.js extremamente eficiente para sistemas que demandam operações intensivas de entrada e saída (I/O).

No contexto do RoomieLink, o Node.js é utilizado para implementar APIs RESTful que permitem a comunicação entre o frontend e o banco de dados. Sua ampla biblioteca de pacotes disponibilizada pelo npm (Node Package Manager) acelera o desenvolvimento, oferecendo soluções prontas e confiáveis para funcionalidades complexas, como autenticação, validação de dados e integração com serviços externos. Essas características tornam o Node.js uma escolha estratégica para garantir o desempenho e a escalabilidade do sistema.

7.2 Frontend em Sistemas de Informação

O frontend de um sistema web é a interface visual com a qual os usuários interagem diretamente. Ele é responsável por apresentar as funcionalidades do sistema de forma clara, intuitiva e eficiente, garantindo uma experiência de uso satisfatória. No desenvolvimento moderno de aplicações web, a escolha das tecnologias para a construção do frontend é crucial, pois impacta diretamente tanto a performance quanto a escalabilidade e a manutenção do sistema (HART, 2019).

Para o desenvolvimento do frontend do RoomieLink, foi optado pelo uso das bibliotecas React.js e Next.js, duas ferramentas amplamente utilizadas no ecossistema de JavaScript, especialmente para a construção de interfaces de usuário dinâmicas e de alto desempenho (KELLY, 2020).

React.js é uma biblioteca JavaScript declarativa, eficiente e flexível, criada pelo Facebook. Ela permite a construção de interfaces de usuário por meio de componentes reutilizáveis, o que facilita a manutenção e escalabilidade de aplicações. A principal característica do React é a utilização de um modelo de programação baseado em componentes, onde cada parte da interface pode ser tratada como um componente isolado, tornando o código mais modular e organizado (FACEBOOK, 2021). Além disso, o React possui um sistema de "virtual DOM" que otimiza o processo de renderização, garantindo que apenas as partes da interface que sofreram alterações sejam atualizadas, o que melhora a performance da aplicação (MARTIN, 2018).

Next.js, por sua vez, é um framework baseado no React, que oferece uma série de funcionalidades adicionais que facilitam o desenvolvimento de aplicações web. Ele se destaca principalmente por permitir a renderização do lado do servidor (Server-Side Rendering - SSR) e a geração de sites estáticos (Static Site Generation - SSG), o que resulta em um melhor desempenho em termos de tempo de carregamento e SEO (Search Engine Optimization). O Next.js também possui recursos que simplificam a configuração do roteamento, o gerenciamento de estado e a integração com APIs, tornando-o uma escolha robusta para a criação de aplicações complexas e escaláveis (ZEIT, 2020).

O uso combinado dessas duas tecnologias permitirá que o frontend do RoomieLink seja não apenas dinâmico e responsivo, mas também otimizado para uma experiência de usuário fluida e rápida. A escolha do React.js proporciona flexibilidade no desenvolvimento da interface, enquanto o Next.js contribui para a melhoria do desempenho e da indexação da aplicação, aspectos essenciais para a construção de um produto com alta qualidade e facilidade de manutenção.

Neste capítulo, serão abordados os detalhes do processo de implementação do frontend do **RoomieLink**, com ênfase no uso do React.js e Next.js para a construção de uma interface moderna e eficiente, alinhada às necessidades do projeto e às melhores práticas de desenvolvimento web.

7.2.1 React.js

O React.js é uma biblioteca JavaScript de código aberto utilizada para a construção de interfaces de usuário dinâmicas e interativas. Criada originalmente pelo Facebook, o React tem se consolidado como uma das principais ferramentas no desenvolvimento frontend, devido à sua eficiência, flexibilidade e alto desempenho. Ele permite a criação de interfaces modulares e reutilizáveis, utilizando o conceito de componentes.

7.2.1.1 O que é React.js

O React é uma biblioteca declarativa que permite criar interfaces de usuário de forma eficiente e previsível. Em vez de manipular diretamente o DOM (Document Object Model), o React utiliza um conceito denominado Virtual DOM, que otimiza o processo de renderização das páginas. O Virtual DOM é uma representação em memória do DOM real, permitindo que o React determine de forma eficiente as mudanças necessárias para atualizar a interface do usuário.

Ao invés de modificar diretamente o DOM, o React calcula as diferenças entre o Virtual DOM e o DOM real e, em seguida, aplica as mudanças de forma eficiente, o que resulta em uma melhora considerável no desempenho das aplicações.

7.2.1.2 Principais Conceitos do React.js

O React é baseado em alguns conceitos fundamentais, que facilitam o desenvolvimento de aplicações web escaláveis e de fácil manutenção. Neste capítulo, vamos destrinchar os principais conceitos dessa tecnologia.  
 Componentes: No React, a interface é construída a partir de componentes reutilizáveis. Cada componente pode representar uma parte da interface, como um botão, uma lista ou um formulário. Os componentes podem ser compostos por outros componentes, criando uma estrutura hierárquica. Isso torna o desenvolvimento mais modular e flexível.

JSX (JavaScript XML): É uma extensão de sintaxe para JavaScript que permite escrever HTML dentro de arquivos JavaScript. Com o JSX, é possível descrever a estrutura da interface de forma declarativa e intuitiva. No entanto, o JSX não é obrigatório; ele é apenas uma conveniência que facilita o desenvolvimento.

*Props* (Propriedades): São os parâmetros passados para os componentes. Elas permitem que os componentes recebam dados de seus componentes pais e, dessa forma, retenham a flexibilidade necessária para serem reutilizados em diferentes contextos.

*State*(Estado): O estado de um componente armazena dados que podem mudar ao longo do tempo. Quando o estado de um componente é alterado, ele dispara uma nova renderização da interface para refletir a mudança.

Ciclo de Vida dos Componentes: O React oferece métodos específicos que permitem controlar o ciclo de vida de um componente, desde a sua criação até a sua destruição. Esses métodos ajudam a realizar tarefas como inicializar dados, fazer requisições à API ou limpar recursos quando o componente não for mais necessário.

7.2.1.3 Vantagens do Uso do React.js

O React oferece várias vantagens, especialmente em aplicações de grande escala, como seu desempenho otimizado através do Virtual DOM, que realiza atualizações de forma eficiente, reduzindo as modificações no DOM real e melhorando a performance da aplicação. Sua estrutura baseada em componentes facilita a reutilização de partes da interface, o que contribui para a manutenção e o desenvolvimento de novas funcionalidades. Além disso, o React possui uma grande comunidade de desenvolvedores e um ecossistema rico em bibliotecas e ferramentas, como o React Router para navegação e o Redux para gerenciamento de estado, o que amplia suas funcionalidades. Outra vantagem é a facilidade de aprendizado, especialmente para quem já está familiarizado com JavaScript, tornando-o uma opção acessível em relação a outras bibliotecas e frameworks.

7.2.1.4 Aplicação do React.js no Projeto RoomieLink

No RoomieLink, o React.js é utilizado para criar uma interface de usuário interativa e dinâmica. A escolha do React foi motivada pela necessidade de criar uma aplicação que fosse fácil de manter, escalável e com alta performance. A modularização proporcionada pelos componentes permite que o frontend seja desenvolvido de maneira eficiente, com cada parte da interface sendo tratada de forma isolada.

Além disso, o React facilita a criação de interfaces reativas, ou seja, interfaces que respondem às interações do usuário de forma rápida e fluida. Isso é particularmente importante em um projeto como o RoomieLink, que precisa garantir uma boa experiência para os usuários, com transições rápidas e uma navegação intuitiva.

Neste capítulo, será detalhado como o React.js foi utilizado na construção das funcionalidades do frontend do RoomieLink, incluindo a criação de componentes reutilizáveis, o gerenciamento de estado e a implementação de rotas dinâmicas.

7.2.2 Next.js

O Next.js é um framework de desenvolvimento web baseado em React.js, projetado para facilitar a criação de aplicações web dinâmicas e escaláveis. Desenvolvido pela Vercel, ele oferece uma experiência de desenvolvimento simplificada, com recursos que tornam a criação de sites rápidos e eficientes, como renderização do lado servidor (SSR), geração de páginas estáticas (SSG) e carregamento otimizado.

Essa tecnologia integra esses recursos, o que contribui significativamente para o desempenho das aplicações, melhorando a velocidade de carregamento e a experiência do usuário. Uma das principais características do Next.js é o suporte a rotas baseadas em arquivos, o que elimina a necessidade de configuração manual de rotas. Essa funcionalidade facilita o desenvolvimento e mantém o código organizado, já que a estrutura de pastas do projeto reflete diretamente a estrutura das URLs.

7.2.2.1 Principais Características do Next.js

O Next.js oferece uma série de funcionalidades que o tornam uma excelente escolha para a construção de aplicações modernas. Uma das principais características é a renderização do lado servidor (SSR), onde as páginas da aplicação são geradas no servidor e enviadas para o navegador como HTML estático. Isso melhora significativamente o tempo de carregamento e otimiza o SEO, pois o conteúdo fica disponível imediatamente, sem a necessidade de aguardar o carregamento e a execução do JavaScript no cliente.

Outra funcionalidade importante é a geração de páginas estáticas (SSG), que permite que as páginas sejam geradas no momento da construção do projeto, oferecendo um desempenho aprimorado, já que as páginas são pré-geradas e servidas como arquivos estáticos.

O carregamento otimizado é mais uma característica do Next.js, que realiza o carregamento assíncrono do código necessário para exibir uma página específica. Isso garante que o aplicativo carregue de maneira mais rápida, carregando inicialmente apenas o que é necessário. Além disso, o Next.js oferece suporte a API Routes, permitindo a criação de endpoints de backend diretamente dentro da aplicação frontend. Essa funcionalidade simplifica a construção de aplicações full-stack, já que a integração entre frontend e backend é facilitada.

Por fim, o Next.js também oferece suporte nativo à otimização de imagens, reduzindo o tamanho dos arquivos de imagem e melhorando o desempenho geral da aplicação. Ele suporta formatos modernos, como o WebP, e realiza automaticamente o carregamento das imagens de forma otimizada.

7.2.2.3 Vantagens do uso do Next.js

O Next.js oferece várias vantagens, especialmente em projetos que exigem alto desempenho e uma boa experiência do usuário. Uma das principais vantagens é o melhor desempenho, pois a renderização do lado servidor e a geração de páginas estáticas permitem um carregamento mais rápido, melhorando a performance geral da aplicação (VERCEL, 2024). Além disso, a facilidade de desenvolvimento é um dos pontos fortes, já que o Next.js oferece uma estrutura de pastas intuitiva e rotas automáticas, simplificando tanto o desenvolvimento quanto a organização do código (PEREIRA, 2020).

O SEO aprimorado também é uma vantagem significativa, uma vez que a renderização do lado servidor garante que o conteúdo da página esteja disponível imediatamente para os bots de busca, resultando em melhores classificações nos motores de busca. O Next.js também é altamente flexível, permitindo que os desenvolvedores escolham entre diferentes abordagens de renderização, como SSR, SSG ou até CSR (Client-Side Rendering), dependendo das necessidades específicas de cada página.

Por fim, o Next.js é uma excelente escolha para aplicações que precisam ser escaláveis e de fácil manutenção, graças à geração de páginas estáticas e otimização do carregamento de conteúdo, o que facilita a escalabilidade e a manutenção do projeto.

7.3 Estruturação e Modelagem do Banco de Dados

A modelagem do banco de dados é uma etapa crítica no desenvolvimento de sistemas de informação, pois define como os dados serão organizados, armazenados e acessados para atender às necessidades do sistema. Segundo Elmasri e Navathe (2010), a modelagem de dados é essencial para compreender as inter-relações entre os componentes do sistema e garantir a consistência, integridade e escalabilidade das informações. Essa etapa envolve a criação de representações abstratas dos dados, como diagramas de classes e de casos de uso, que auxiliam na identificação das entidades e suas relações.

No RoomieLink, a estruturação do banco de dados foi guiada pelos diagramas de casos de uso e, principalmente, pelo diagrama de classes, que forneceu uma base sólida para identificar as entidades principais e suas interações. Esses diagramas permitiram mapear os requisitos do sistema e transformá-los em tabelas e relações no banco de dados. A adoção do PostgreSQL, um sistema de gerenciamento de banco de dados relacional robusto e amplamente utilizado, foi fundamental para garantir que o sistema pudesse lidar com volumes significativos de dados, mantendo a segurança e a integridade das informações.

Para simplificar o desenvolvimento e proporcionar maior flexibilidade, o Prisma foi escolhido como ORM (Object-Relational Mapping) para intermediar a comunicação entre o backend e o banco de dados. O Prisma permite a definição de esquemas de dados de forma declarativa, facilitando a sincronização entre o código da aplicação e a estrutura do banco. No projeto, o uso do Prisma, combinado ao PostgreSQL, proporcionou uma abordagem eficiente e organizada para lidar com as diversas entidades do sistema, como usuários, imóveis, preferências e proximidades. Essa combinação garantiu que a base de dados fosse escalável e preparada para futuras evoluções, como o uso de migração e adaptação do banco de dados para plataformas em nuvem, como AWS ou Google Cloud, o que suporta a visão de crescimento do projeto. Essa configuração proporciona uma base sólida para o armazenamento e gerenciamento seguro das informações dos usuários, assegurando a integridade dos dados enquanto o sistema cresce em escala e funcionalidade.

7.3.1 PostgreSQL

O PostgreSQL é um sistema gerenciador de banco de dados relacional (SGBD) de código aberto, amplamente reconhecido por sua robustez, escalabilidade e conformidade com os padrões SQL. Sua popularidade no desenvolvimento de aplicações se deve ao seu conjunto de funcionalidades avançadas, segurança e flexibilidade.

Ele é conhecido por ser altamente extensível, permitindo a criação de tipos de dados personalizados, operadores e funções, além de suportar transações ACID (Atomicidade, Consistência, Isolamento e Durabilidade). Isso garante que as operações realizadas no banco de dados sejam seguras e eficientes, minimizando riscos de corrupção de dados. Neste capítulo, exploramos as principais características e benefícios do PostgreSQL e sua aplicação no contexto do projeto.

7.3.1.1 Principais Características do PostgreSQL

O PostgreSQL é uma escolha popular devido às suas características avançadas, incluindo suporte a transações ACID, que garantem a execução segura das transações, mantendo a integridade dos dados e a consistência nas operações. Ele também oferece escalabilidade, podendo ser facilmente ajustado para suportar grandes volumes de dados e altas taxas de leitura e escrita, o que o torna uma solução robusta para aplicações em crescimento. Sua extensibilidade permite a criação de tipos de dados personalizados, funções e operadores, proporcionando flexibilidade para atender às necessidades específicas de cada aplicação. Além disso, o PostgreSQL é altamente eficiente na execução de consultas complexas, como junções, subconsultas e agregações, essenciais para sistemas que manipulam grandes volumes de dados, e segue estritamente o padrão SQL, garantindo consistência e interoperabilidade com outras soluções de banco de dados.

7.3.1.2 Vantagens do Uso do PostgreSQL

O PostgreSQL oferece diversas vantagens, incluindo desempenho e confiabilidade, garantindo um desempenho consistente mesmo com grandes volumes de dados. Uma das principais vantagens desse sistema de banco de dados é seu suporte a consultas complexas, como junções (joins), subconsultas e transações, o que o torna ideal para aplicações que exigem manipulação avançada de dados. Além disso, a flexibilidade do PostgreSQL permite que ele seja utilizado em uma ampla gama de cenários, desde pequenas aplicações até grandes sistemas corporativos com altas demandas de escalabilidade.

Ele também se destaca pela segurança, com recursos avançados como criptografia de dados, controle de acesso e autenticação, assegurando a proteção das informações armazenadas. Sua fácil integração com ferramentas de ORM, como o Prisma, permite uma gestão eficiente do banco de dados no backend, enquanto sua capacidade de escalabilidade é ideal para projetos que precisam se adaptar ao crescimento de dados e usuários ao longo do tempo.

Além disso, o PostgreSQL conta com uma comunidade ativa e um vasto ecossistema de bibliotecas e ferramentas, facilitando a resolução de problemas e a implementação de novas funcionalidades.

O PostgreSQL é uma escolha sólida e confiável, oferecendo uma série de recursos avançados que garantem o armazenamento seguro, eficiente e escalável de dados. Sua flexibilidade e conformidade com os padrões SQL permitem que ele atenda às necessidades específicas do projeto, oferecendo suporte a consultas complexas e integração com outras tecnologias, como o **Prisma**. Essa combinação contribui para a robustez e o desempenho da aplicação, garantindo que o sistema seja capaz de crescer e se adaptar às demandas futuras.

7.3.2 Prisma

O Prisma é um ORM que facilita a comunicação entre uma aplicação Node.js e um banco de dados relacional, permitindo que os desenvolvedores manipulem dados utilizando código JavaScript ou TypeScript em vez de SQL diretamente. Ao fornecer uma abstração sobre o banco de dados, o Prisma ajuda a automatizar várias tarefas comuns no desenvolvimento backend, como criação de esquemas, migrações e consultas, melhorando a eficiência no desenvolvimento e manutenção de aplicações. Ele inclui o Prisma Client, que é uma ferramenta poderosa para acessar e manipular dados de forma segura e otimizada.

7.3.2.1 Principais Características do Prisma

O Prisma oferece uma série de características que o tornam uma escolha popular no desenvolvimento de aplicações modernas, destacando-se pela facilidade de uso, que simplifica a criação e manutenção de esquemas de banco de dados, permitindo a definição intuitiva de modelos de dados e a escrita automática de migrações. Integrado com TypeScript, o Prisma garante segurança de tipos, evitando falhas comuns relacionadas a erros de tipagem e proporcionando maior robustez ao código. Sua camada de abstração otimizada resulta em consultas SQL eficientes e com baixo custo de performance, mesmo em bases de dados grandes e complexas.

Além disso, o Prisma facilita a evolução do banco de dados ao longo do tempo com migrações automatizadas, permitindo a aplicação controlada de alterações no esquema sem perda de dados. O Prisma Client oferece uma API altamente tipada e auto-gerada, tornando o processo de execução de consultas no banco de dados mais eficiente e seguro.

7.3.2.2 Vantagens do Uso do Prisma

O uso do Prisma em projetos de backend oferece diversas vantagens, como o desenvolvimento ágil, já que sua abstração permite que os desenvolvedores escrevam menos código e se concentrem mais na lógica de negócios da aplicação, acelerando o ciclo de desenvolvimento.

A facilidade de manutenção também é um ponto positivo, pois a definição clara de modelos de dados e o sistema de migrações automatizadas facilitam o gerenciamento do banco de dados à medida que a aplicação cresce, garantindo consistência e integridade dos dados.

O Prisma é ideal para projetos que exigem alta performance e escalabilidade, sendo capaz de lidar com grandes volumes de dados sem comprometer o desempenho da aplicação. A flexibilidade de banco de dados é outro benefício, pois, embora seja mais utilizado com PostgreSQL, o Prisma oferece suporte a outros bancos de dados relacionais, como MySQL e SQLite, permitindo flexibilidade de escolha dependendo das necessidades do projeto.

Além disso, a ferramenta conta com uma grande comunidade de desenvolvedores e uma documentação completa e acessível, o que facilita a resolução de problemas e a implementação de novas funcionalidades, com atualizações frequentes e novos recursos sendo adicionados regularmente.

7.4 Controle de Versão

O controle de versão é essencial em projetos colaborativos, garantindo que as alterações no código sejam monitoradas e que diferentes contribuidores possam trabalhar de forma integrada sem conflitos. O GitHub é uma plataforma baseada em Git, amplamente utilizada para o versionamento de código. Segundo Chacon e Straub (2014), o GitHub promove a colaboração eficiente por meio de recursos como pull requests, revisões de código e integração com ferramentas de automação. No RoomieLink, ele é utilizado para garantir a integridade do código e facilitar o trabalho em equipe, além de permitir o rastreamento de mudanças e resolução de problemas com agilidade.

7.5 Hospedagem: Conceito e Aplicação em Sistemas de Informação

A hospedagem, no contexto de sistemas de informação, refere-se ao serviço que permite armazenar e disponibilizar recursos de um sistema, como sites, aplicativos web e bancos de dados, em servidores acessíveis pela internet. Ela é fundamental para garantir que os usuários consigam acessar a aplicação ou o conteúdo a qualquer momento, de qualquer lugar, e com a performance necessária. Ao hospedar uma aplicação, um servidor fornece os recursos necessários para que o sistema funcione de forma otimizada, incluindo processamento, armazenamento e conectividade com outros dispositivos ou redes.

A hospedagem desempenha um papel essencial na infraestrutura de TI, pois influencia diretamente na performance, segurança, escalabilidade e confiabilidade de um sistema. No desenvolvimento de sistemas web e aplicativos, a escolha de um provedor de hospedagem adequado é crucial para garantir que a aplicação esteja disponível para os usuários finais sem interrupções, com alta performance e segurança. No contexto de sistemas de informação, a hospedagem não apenas disponibiliza a aplicação, mas também assegura que ela possa ser escalada conforme o crescimento do número de usuários ou a demanda por mais recursos, além de oferecer suporte a atualizações e manutenções contínuas. Agora que entendemos o papel da hospedagem, podemos explorar as tecnologias específicas utilizadas para hospedar o projeto.

7.5.1 Heroku

O Heroku é uma plataforma de hospedagem como serviço (PaaS) altamente popular, projetada para simplificar o processo de implantação e escalabilidade de aplicativos web. Ele oferece um ambiente de desenvolvimento eficiente e fácil de usar, ideal para desenvolvedores que buscam uma solução rápida e sem complicações para disponibilizar suas aplicações. O Heroku abstrai a complexidade da infraestrutura, permitindo que os desenvolvedores se concentrem no código da aplicação em vez de gerenciar servidores ou configurar ambientes de produção.

Entre as principais características do Heroku, destaca-se a simplicidade no processo de implantação, que pode ser feita diretamente através de integração com o Git, permitindo a atualização constante do projeto com comandos simples. Além disso, o Heroku oferece escalabilidade automática, o que significa que a aplicação pode ser facilmente ajustada para atender à demanda, seja aumentando o número de "dynos" (unidades de computação) ou ajustando os recursos conforme a necessidade do projeto.

O Heroku também é amplamente utilizado por sua integração fácil com bancos de dados, como o Heroku Postgres, e com outros serviços de back-end, como filas e caches. A plataforma oferece uma série de complementos que permitem a integração de funcionalidades adicionais, como serviços de email, monitoramento de performance e segurança.

Além disso, o Heroku simplifica a administração de infraestrutura, pois lida com a manutenção do servidor e a segurança do sistema, garantindo que o projeto esteja sempre atualizado e seguro. Isso torna o Heroku uma escolha popular para desenvolvedores que precisam de uma solução robusta, escalável e de fácil administração para hospedar suas aplicações web de forma rápida e eficaz.  
 Com essa facilidade de uso, flexibilidade e integração com várias ferramentas, o Heroku é uma excelente plataforma para hospedar o backend do projeto, permitindo foco na construção das funcionalidades principais sem se preocupar com questões de infraestrutura.

7.5.2 Vercel

A Vercel é uma plataforma de hospedagem voltada para o desenvolvimento e a implantação de aplicativos web, especialmente aqueles baseados em frameworks como Next.js, React.js e outras tecnologias JavaScript. Focada em otimizar o processo de deployment contínuo, a Vercel oferece uma integração direta com ferramentas de versionamento, como GitHub, GitLab e Bitbucket, permitindo que as atualizações de código sejam automaticamente refletidas no ambiente de produção a cada novo commit. Isso facilita a entrega contínua e o ciclo de desenvolvimento ágil, essencial para a construção de aplicações modernas.

Uma das principais vantagens da Vercel é sua capacidade de realizar deploys instantâneos e sua integração com Next.js, o que a torna uma escolha ideal para hospedar aplicativos que utilizam renderização do lado servidor (SSR) ou geração de sites estáticos (SSG). A plataforma oferece suporte nativo a essas tecnologias, otimizando o processo de construção e disponibilização das páginas para os usuários, ao mesmo tempo em que maximiza a performance e o tempo de resposta da aplicação.

A Vercel também se destaca pela escabilidade automática, que ajusta os recursos de infraestrutura de acordo com o tráfego da aplicação, permitindo que ela cresça sem a necessidade de intervenção manual. Além disso, sua infraestrutura de edge networks garante uma entrega de conteúdo rápida e de baixa latência ao redor do mundo, proporcionando uma excelente experiência ao usuário, independentemente da sua localização geográfica.

Outro ponto forte da Vercel é a simplicidade na configuração e no uso, com a plataforma se preocupando com a maior parte da configuração de infraestrutura, como servidores, redes e segurança, permitindo que os desenvolvedores foquem unicamente no código. Ela também oferece Funções Serverless, que permitem executar partes específicas do código do backend sem a necessidade de provisionar ou gerenciar servidores, facilitando a criação de APIs e outras funcionalidades dinâmicas.  
 Com recursos como deploy contínuo, escalabilidade automática, suporte a tecnologias modernas e integração com ferramentas populares, a Vercel é uma plataforma robusta e ideal para hospedar o frontend de projetos modernos, proporcionando uma experiência de desenvolvimento simplificada e uma infraestrutura otimizada.

REFERÊNCIAS

Graham, S. (2019). **Sharing cities: A case for truly smart and sustainable cities.** Cambridge, MA: MIT Press.

Sennett, R. (2012). **Together: The rituals, pleasures, and politics of cooperation.** New Haven, CT: Yale University Press.

Botsman, R. (2013). **What's mine is yours: How collaborative consumption is changing the way we live.** New York, NY: HarperCollins.

Benkler, Y. (2006). **The wealth of networks: How social production transforms markets and freedom.** New Haven, CT: Yale University Press.

Babbie, E. R. (2016). **The Practice of Social Research.** Boston, MA: Cengage Learning.

Hair Jr., J. F., et al. (2019). **Análise multivariada de dados.** Porto Alegre: Bookman.

Booch, G. (1994). **Object-Oriented Analysis and Design with Applications.** Addison-Wesley.

Guedes, G. (2006). **UML 2: uma abordagem prática.** Novatec Editora.

Flanagan, D. (2017). **JavaScript: The Definitive Guide.** O'Reilly Media.

Spencer, J., Murray, A., & Nixon, R. (2019). **React Native Development.** Packt Publishing.

Virtue, B., & MacDonald, B. (2018). **Learning Firebase: Develop the Next Generation of Real-Time and Serverless Web Apps.** O'Reilly Media.

Foy, B. (2018). **Mastering Git: Attain expert-level proficiency with Git for enhanced productivity and efficient collaboration.** Packt Publishing.

Chacon, S., & Straub, B. (2014). **Pro Git.** Apress.

Krug, S. (2014). **Não Me Faça Pensar: Uma Abordagem de Bom Senso à Usabilidade na Web. [Edição Revisada].** Alta Books.

Rand, P. (1993). **Design, Form, and Chaos.** Yale University Press.

Lupton, E. (2004). Thinking with Type: A Critical Guide for Designers, Writers, Editors, & Students. Princeton Architectural Press.

Bringhurst, R. (2008). The Elements of Typographic Style. Hartley & Marks.

Berman, D. (2009). Do Good Design: How Design Can Change Our World. New Riders.

****Tilkov, S., & Vinoski, S.**** (2010). Node.js: Using JavaScript to build high-performance network applications. IEEE Internet Computing, 14(6), 80-83.

****Stonebraker, M.**** (2015). The case for NewSQL databases. Communications of the ACM, 58(5), 10-11.

**Chacon, S., & Straub, B.** (2014). Pro Git. Apress.

**Gouveia, J.** (2021). Deploying scalable applications with Heroku: Best practices. Journal of Cloud Computing, 9(3), 45-56.

**Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., ... & Zaharia, M.** (2010). A view of cloud computing. Communications of the ACM, 53(4), 50-58.

****Tilkov, S., & Vinoski, S.**** (2010). Node.js: Using JavaScript to build high-performance network applications. IEEE Internet Computing, 14(6), 80-83.

FACEBOOK. React – A JavaScript library for building user interfaces. Disponível em: https://reactjs.org/. Acesso em: 30 dez. 2024.

HART, M. Frontend Development: The Key to Creating Interactive User Interfaces. Wiley, 2019.

KELLY, S. Modern Frontend Development with React and Next.js. Packt Publishing, 2020.

MARTIN, R. Efficient React – Optimizing Performance. O'Reilly Media, 2018.

ZEIT. Next.js Documentation. Disponível em: https://nextjs.org/docs. Acesso em: 29 dez. 2024.

VERCEL. Next.js – The React Framework. Disponível em: <https://nextjs.org/>. Acesso em: 29 dez. 2024.

PEREIRA, R. Modern Web Development with Next.js. Packt Publishing, 2020.

POSTGRESQL. PostgreSQL: The World’s Most Advanced Open Source Relational Database. Disponível em: https://www.postgresql.org/. Acesso em: 30 dez. 2024.

1. Documentação do Caso de Uso
2. Roteiro do Questionário

Roteiro do Questionário: