

Desenvolvimento de um aplicativo de reservas de imóveis utilizando Flutter

Jade Mendes e Silva de Almeida¹, Matheus Nogueira Farina¹, Thales Mendonça Moreira Pinto¹

¹Instituto de Computação
Universidade Federal Fluminense (UFF) – Niterói, RJ – Brasil
{jademendes, nogueiramatheus, thalesmendonca}@id.uff.br

Resumo. *Este trabalho apresenta o desenvolvimento de um aplicativo para reservas de imóveis utilizando Flutter e SQLite. O sistema permite que usuários cadastrem e reservem propriedades, gerenciem seus imóveis cadastrados e avaliem outras propriedades. A experiência de desenvolvimento proporciona aos alunos aplicação prática de conceitos de banco de dados, desenvolvimento mobile e integração com APIs externas, preparando-os para desafios reais do mercado de trabalho.*

1. Introdução

A crescente digitalização de serviços no setor imobiliário tem impulsionado o desenvolvimento de soluções tecnológicas que facilitam a comunicação entre proprietários de imóveis e locatários. Nesse contexto, este projeto propõe o desenvolvimento de um aplicativo para reservas de imóveis, permitindo que usuários anunciem seu próprios imóveis para locação ou busquem acomodações de maneira eficiente.

O sistema é composto por duas partes: cadastro de propriedades e reserva de acomodações. Os proprietários podem cadastrar seus imóveis, incluindo detalhes como localização (obtida automaticamente através da API ViaCEP), imagens e valores de diária. Os usuários interessados em realizar reservas, por sua vez, podem filtrar propriedades com base em diversos critérios, verificar detalhes e finalizar a reserva, armazenando os dados no banco SQLite. O sistema também implementa funcionalidades como gerenciamento de reservas e avaliações de propriedades.

Este documento descreve as etapas de desenvolvimento do aplicativo, desde a concepção da ideia até a implementação, abordando os requisitos do projeto, a fundamentação teórica, a metodologia aplicada e os resultados obtidos. O desenvolvimento foi realizado utilizando o framework Flutter, garantindo compatibilidade com dispositivos Android e iOS, e a biblioteca SQLite para armazenamento local dos dados, garantindo persistência e eficiência na manipulação das informações do sistema.

2. Fundamentação Teórica

Este projeto baseia-se nos conceitos de desenvolvimento para dispositivos móveis, bancos de dados locais e integração com APIs.

Quanto ao desenvolvimento para dispositivos móveis, o grupo optou pelo framework Flutter. Além de ter sido a ferramenta utilizada em sala de aula no decorrer do semestre, Flutter é um framework de desenvolvimento de aplicativos multiplataforma criado pelo Google, baseado na linguagem Dart. Ele permite a construção de interfaces responsivas e de alto desempenho para Android e iOS a partir de um único código-fonte. Seu uso no projeto garante maior eficiência no desenvolvimento, manutenção e expansão futura do sistema.

Para garantir a persistência dos dados no aplicativo, foi utilizado o banco de dados SQLite, acessado por meio do plugin sqflite. SQLite é um banco de dados relacional leve e eficiente, amplamente utilizado em aplicações móveis devido ao seu baixo consumo de recursos e integração simplificada com sistemas baseados em SQL. A estrutura das tabelas foi definida para armazenar informações sobre usuários, propriedades, endereços, reservas e imagens.

Para facilitar o preenchimento dos endereços no cadastro de propriedades, o sistema utiliza a API ViaCEP, que permite buscar informações detalhadas de um CEP fornecido. Esse recurso melhora a usabilidade do sistema e reduz a probabilidade de erros no cadastro dos endereços.

A funcionalidade de reservas segue um modelo de relacionamento entre usuários e propriedades, garantindo que imóveis já reservados em determinado período não sejam listados como disponíveis. Além disso, as avaliações dos hóspedes são armazenadas e influenciam na classificação das propriedades exibidas na busca.

Com essa base teórica, o projeto busca oferecer uma solução competente para a intermediação entre anfitriões e hóspedes, garantindo uma experiência intuitiva e funcional para os usuários do aplicativo.

3. Metodologia

O desenvolvimento do aplicativo foi conduzido em etapas bem definidas para garantir organização e funcionalidade. Os alunos utilizaram Git e Github para controle de versões do código e adotaram o modelo Git Flow para organizar o fluxo de trabalho. Inicialmente, os requisitos do programa foram analisados e estruturados, estabelecendo as funcionalidades principais, como cadastro de usuários, registro de propriedades, gerenciamento de acomodações e realização de reservas.

A implementação foi realizada, como já mencionado, utilizando Flutter para a interface e SQLite para o armazenamento local dos dados. A modelagem do banco seguiu os princípios de normalização, resultando na criação de tabelas inter-relacionadas para usuários, endereços, registros de hospedagem e imagens. Além disso, foi integrada a API ViaCEP para preenchimento automático dos dados de localização a partir do CEP informado pelo usuário.

Para a lógica de reservas, foi adotado um mecanismo que evita conflitos de datas, permitindo apenas agendamentos válidos. A interface foi projetada visando acessibilidade e facilidade de uso, proporcionando uma experiência intuitiva.

Os testes foram realizados em diferentes dispositivos e cenários, garantindo a confiabilidade das funcionalidades. Durante essa fase, foram aplicadas correções e refinamentos para aprimorar a usabilidade e o desempenho do sistema.

4. Desenvolvimento do Aplicativo

Ao iniciar o aplicativo, a tela de login permite que os usuários autenticem suas credenciais por meio de um diálogo interativo. Caso o usuário não esteja cadastrado, há a opção de acessar a tela de cadastro, onde é possível inserir nome, e-mail e senha. O gerenciamento desses dados é feito por meio do UserService, que interage com o banco de dados SQLite para armazenar e validar as informações.

A tela principal do aplicativo, definida no index.dart, organiza a estrutura de navegação, garantindo que o usuário tenha acesso às funcionalidades adequadas após a autenticação. A comunicação com a API ViaCEP também está integrada ao sistema, permitindo que endereços sejam preenchidos automaticamente a partir do CEP inserido.

O desenvolvimento seguiu boas práticas de separação de responsabilidades, utilizando controladores para gerenciar as interações entre a interface e a camada de dados. Foram realizados testes para validar a experiência do usuário e corrigir possíveis inconsistências antes da conclusão do projeto.

5. Resultados e conclusão

O desenvolvimento do aplicativo permitiu a implementação de um sistema funcional para cadastro, listagem e reserva de imóveis, atendendo aos requisitos propostos. A utilização do Flutter proporcionou uma interface intuitiva e responsiva, enquanto o SQLite garantiu a persistência dos dados dos usuários, propriedades e reservas.

Durante o processo, desafios como a integração com a API ViaCEP, a gestão eficiente do estado e a estruturação do banco de dados foram superados. A experiência contribuiu para o aprimoramento das habilidades técnicas e organizacionais dos envolvidos, preparando-os melhor para desafios reais no desenvolvimento de software.

6. Referências

<https://docs.flutter.dev/>

Acesso em Fevereiro, 2025.

<https://www.sqlite.org/docs.html>

Acesso em Fevereiro, 2025.

<https://viacep.com.br/>

Acesso em Fevereiro, 2025.