

ScanPoint

Guilherme Amorim, 107162

Grupo 10

Computação Móvel

Mestrado em Engenharia Informática



Repositório [GitHub](#)



Motivação

O projeto ScanPoint foi desenvolvido no âmbito da cadeira Computação Móvel, como parte do módulo de Flutter. O objetivo principal é que o utilizador possa visitar diversos pontos de interesse espalhados por todo o mundo e fazer scan dos mesmos, podendo guardar um histórico de locais visitados e ver os sítios visitados pelos amigos

Solução

A solução apresentada consiste em:

- Ecrã de Login
 - É o primeiro ecrã a ser mostrado ao utilizador
 - Pode fazer login, ou criar conta
 - Após o login ser efetuado, são apresentadas 3 tabs horizontais: Ecrã Principal, Ecrã de Pesquisa, Ecrã de Amigos
- Ecrã Principal
 - Apresenta um mapa, com vários locais marcados
 - É mostrada a localização atual do utilizador
 - Também é possível ver os 10 locais mais próximos
 - Ao clicar num local, o utilizador é redirecionado para um ecrã com mais detalhes sobre o mesmo
 - Há uma funcionalidade extra neste ecrã: ao abanar o telemóvel, caso haja um local relativamente perto, é sinalizado e o utilizador pode ir para o ecrã detalhado
- Ecrã de Detalhes sobre o Local
 - É neste ecrã que o utilizador vê detalhes sobre o scan efetuado ao ponto de interesse
 - No caso de o utilizador ainda não ter visitado o local, pode abrir a câmara e tirar uma fotografia no momento
- Ecrã de Pesquisa
 - Todos os locais disponíveis são apresentados em várias listas agrupadas pelo país dos mesmos
 - O utilizador também pode pesquisar por locais específicos, utilizando uma *search bar*
- Ecrã de Amigos
 - São listados os amigos do utilizador, assim como também é possível ver os locais visitados por cada um deles
 - É também neste ecrã que dá para adicionar novos amigos, quer escrevendo diretamente o *username* como fazendo scan de um *QR code*
 - Por essa razão, é necessário que haja um botão que permita a um utilizador gerar um *QR code* com o seu *username*
 - O utilizador pode fazer logout

Arquitetura e Opções Técnicas

A arquitetura é composta pela API, base de dados e aplicação Flutter.

A base de dados foi desenvolvida em SQLite e contém informação sobre utilizadores (credenciais), locais (localização) e a relação entre os 2.

A API foi construída usando *FastAPI* e expõe serviços RESTful que permitem a comunicação entre a aplicação Flutter e a base de dados. Garante a autenticação dos utilizadores e permite-os enviar e receber dados, incluindo o envio de imagens. Essas são armazenadas diretamente num diretório específico do servidor da API e o URL correspondente é guardado na base de dados.

Em relação à aplicação, esta oferece as seguintes funcionalidades:

- **Mapas e Localização:** com os pacotes *flutter_map* e *geolocator*, é possível exibir o mapa com a posição em tempo real do utilizador, além de pontos de interesse próximos e a distância para o mesmo.
- **Sensores:** a aplicação deteta um movimento de *shake* para verificar a proximidade do utilizador com o ponto mais próximo.
- **QR Code:** Para gerar os qr code, é usado o pacote *qr_flutter*. Já para fazer o scan do mesmo, é possível com o *mobile_scanner*.

Para uma maior organização, a lógica de UI e backend foi dividida – métodos como *LocalService*, *Friends Service*, etc, são responsáveis por comunicar diretamente com o backend, enquanto o código focado na interface é desenvolvido separadamente dos anteriores, fazendo a conexão com os mesmos apenas quando necessário.

Objetivos alcançados e problemas encontrados

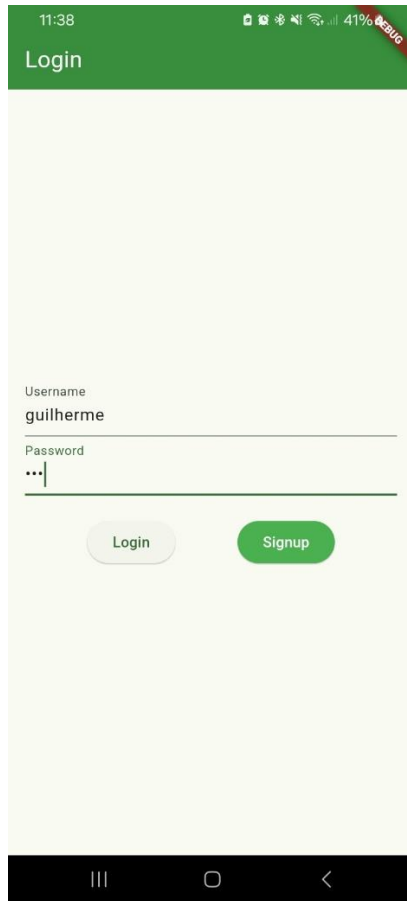
A aplicação cumpriu os principais objetivos propostos, incluindo as funcionalidades de localização, mapa, conexão social, e gestão de visitas a pontos de interesse.

Durante o desenvolvimento, um problema menor ficou por resolver: ao fazer scan do QR Code de um amigo, a aplicação, ocasionalmente, exibe um ecrã preto e fecha inesperadamente. No entanto, após reiniciar a app, verifica-se que o amigo foi adicionado com sucesso, o que indica que o processo de adição é concluído corretamente apesar do erro na interface. Dado que o problema não impede a funcionalidade de adicionar amigos, não o considerei grave.

Tutorial

Fazer Login

O utilizador faz login com a sua conta, ou no caso de não ter, cria uma com as credenciais inseridas



A screenshot of a mobile application's login screen. The screen has a green header bar with the word "Login" in white. Below the header, the background is a light yellow. There are two input fields: "Username" with the text "guilherme" and "Password" with masked characters "..." and a cursor. Below the fields are two buttons: a light green "Login" button and a green "Signup" button. The top status bar shows the time "11:38", signal strength, Wi-Fi, and battery level "41%". A red "debug" banner is in the top right corner. The bottom navigation bar is black with three icons: a hamburger menu, a home icon, and a back arrow.

11:38 41% debug

Login

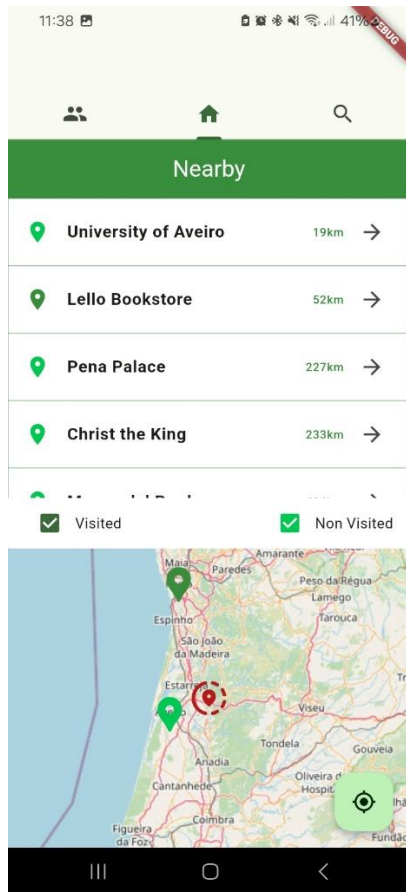
Username
guilherme

Password
...|

Login Signup

Ecrã Principal

No ecrã principal, é mostrado o mapa com todos os locais disponíveis. Estão disponíveis filtros para mostrar os locais já visitados pelo utilizador ou os que ainda estão por descobrir. Além disso, é mostrada a lista de 10 pontos mais próximos: ao clicar, num deles o mapa é centrado na posição do mesmo. Ao clicar na seta, ou no marcador respetivo a um local, o utilizador é levado para uma página de detalhes

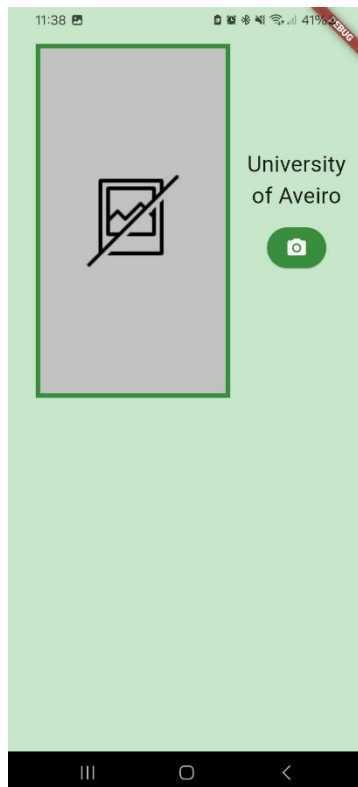


Ecrã Detalhado do Local

Neste ecrã é possível ver detalhes do scan feito ao ponto de interesse, como a fotografia e a data do mesmo

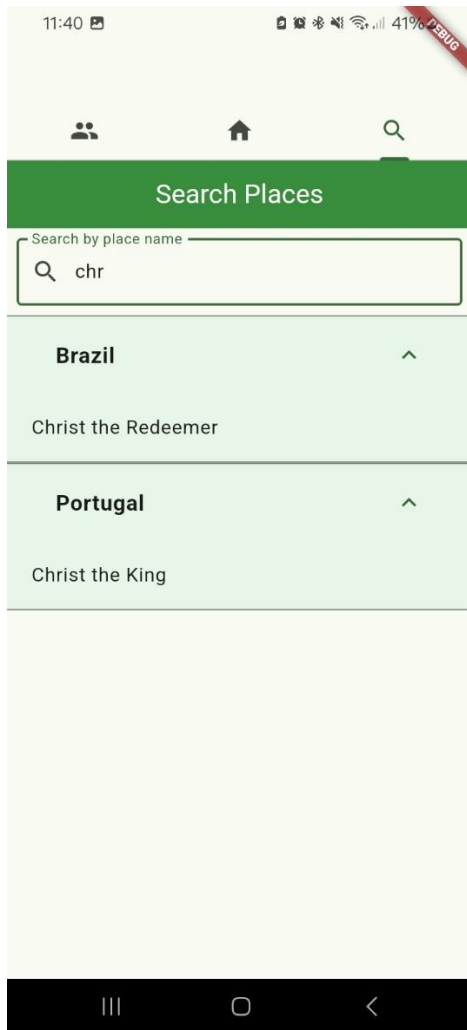


No caso de o ponto de interesse ainda não ter sido visitado, o utilizador pode usar a camara para fotografar o local e o scan fica guardado



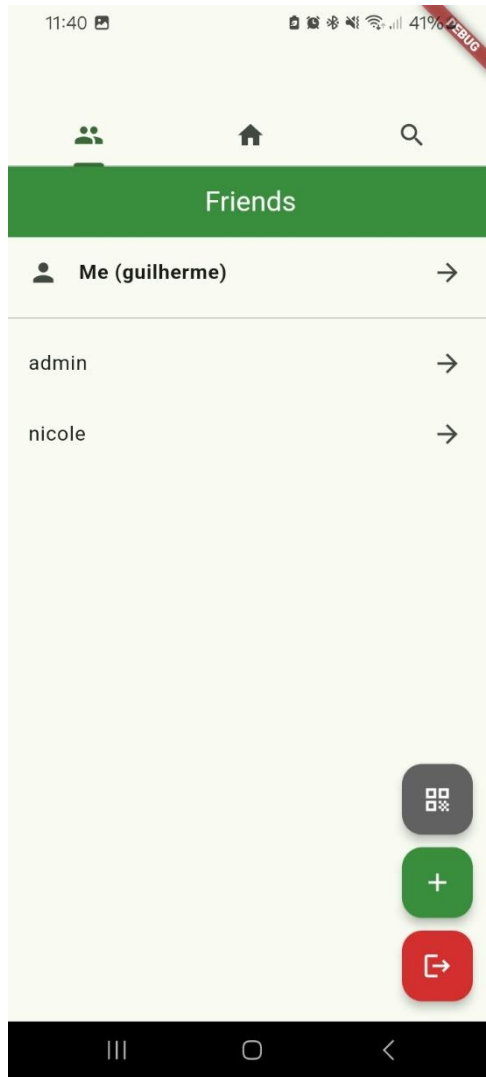
Ecrã de Pesquisa

Numa das tabs, é possível ver todos os locais disponíveis, agrupados pelo país, assim como procurar por um nome específico



Ecrã de Amigos

Neste ecrã, encontra-se um painel com a lista dos amigos do utilizador. Ao clicar num deles, é possível ver todos os scans feitos, num slide



Ainda no ecrã dos amigos, é possível gerar o qr code, adicionar um amigo por username ou fazendo scan do qr code e fazer logout

