

FUNDAÇÃO ESCOLA DE COMÉRCIO ÁLVARES PENTEADO – FECAP

3º SEMESTRE – CIÊNCIA DA COMPUTAÇÃO

Entrega 3 de Programação Orientada a Objetivos e Estrutura de Dados

Antônio Petri de Moraes Soares de Moura e Oliveira: 24026144

Daniel Kyoshi Moribe: 24026509

Thiago Akira Higa Mitami: 24026254

Vitor Kenzo Kanashiro: 24026113

Estrutura de Dados utilizadas:

O projeto combina estruturas de dados Java (como ArrayList, POJOs e StringBuilder) com bibliotecas poderosas (Retrofit, OkHttp) e componentes do Android SDK para criar uma aplicação eficiente, modular e com boa integração com serviços externos como mapas e APIs REST. As escolhas de implementação refletem boas práticas de desempenho e organização de código.

Coleções: List<Location> e ArrayList<Location>

Usadas para armazenar localizações (waypoints). São utilizadas em várias classes como MainActivity, MapsActivity e ShowSavedLocationsList.

1. **ArrayList** permite adição dinâmica e acesso rápido por índice.
2. Compartilhadas globalmente via **MyApplication**, garantindo persistência entre Activities.
3. Compatíveis com **ArrayAdapter**, facilitando exibição em listas na UI.

Classes de Modelo (POJOs)

Usadas para estruturar os dados trocados com a API.

1. Usuario.java:

-Representa um usuário com ID, nome, email e senha.

-Possui dois construtores e métodos getters.

-Método decryptFromServer() usa a classe Criptografia para descriptação.

2. RespostaServidor.java

-Modelo de resposta simples do servidor (mensagem + ID do usuário).

-Utilizada após operações como cadastro.

3. RespostaLogin.java

-Resposta do servidor após login: sucesso (boolean), mensagem, e o objeto Usuario.

-Evita chamadas adicionais para obter dados do usuário.

Manipulação de Strings: StringBuilder

Usado em Criptografia.java para implementar a cifra de César.

1. Evita uso de String + String em loops, o que geraria múltiplos objetos.
2. StringBuilder permite construção de strings de forma eficiente e com menor uso de memória.

Comunicação com API: Retrofit e OkHttpClient

Utilizados em FormLogin.java e FormCadastro.java.

1. Retrofit

-Facilita chamadas HTTP via interfaces Java.

-Usa GsonConverterFactory para serialização JSON.

-ApiService define endpoints com anotações (@POST, @Field, etc.).

2. Call<T> e Callback<T>

-Call<T> representa uma requisição HTTP.

-enqueue() permite execução assíncrona sem travar a UI.

-Callback<T> trata a resposta (onResponse) ou falhas (onFailure).

Componentes de UI e Adaptadores Android

1. ArrayAdapter

-Usado para exibir List<Location> em ShowSavedLocationsList.java.

-Usa layout padrão e toString() para renderizar os itens.

2. Componentes Visuais

-EditText: entrada de texto em formulários.

-Button: aciona ações (login, cadastro, salvar waypoint).

-Switch: ativa/desativa funcionalidades como rastreamento.

-TextView: exibe informações (localização, mensagens).

-ProgressBar: indica operações em andamento.

3. Classes Android e Google APIs

-Location e Address: dados geográficos.

-Intent: navegação entre Activities.

-Google Location & Maps APIs: mapa interativo, marcadores, rastreamento com FusedLocationProviderClient.