



DOCUMENTO DE REQUISITOS

OBES - Sebo Online

Equipe:

509722 – Daniel Santos Fernandes

508159 – Lara Gabrielly Souza Batista Lima

508653 – Victor Anthony Pereira Alves

509278 – Vinícius Lemos Araújo

1. Introdução

Este documento detalha a estrutura organizacional do software, delineando a distribuição de responsabilidades entre diferentes componentes e camadas. A compreensão da arquitetura é essencial para garantir não apenas a funcionalidade adequada, mas também a flexibilidade e a manutenção ao longo do ciclo de vida do projeto. Ao explorar os principais elementos da arquitetura de software, incluindo os componentes do sistema e os padrões de projeto aplicados, buscamos criar uma base para a implementação e evolução contínua do sistema.

2. Componentes do Sistema

2.1. Camada de Visão

- **Activity (Atividade):** Representa uma única tela com uma interface de usuário. É responsável por interagir com o usuário, apresentando uma interface e lidando com eventos.
- **Service (Serviço):** Um componente que executa operações em segundo plano, sem uma interface de usuário visível. Pode ser usado para realizar operações longas, como reprodução de música, download de arquivos, etc.
- **Layouts e Views:** Componentes responsáveis pela aparência da interface do usuário. Layouts definem a estrutura visual, enquanto Views representam elementos visuais individuais.

2.2. Camada de Integração

- **Repository (Repositório):** O componente "Repository" atua como uma camada intermediária entre a camada de dados e a camada de visão (UI).

2.3. Camada de dados

- **Firebase Realtime Database:** Um banco de dados NoSQL em tempo real que permite armazenar e sincronizar dados entre usuários em tempo real.
- **Firebase Authentication:** Oferece serviços de autenticação, permitindo que os usuários se autenticem em seu aplicativo usando e-mail, senha, autenticação social (Google, Facebook, etc.) e outros métodos.

3. Padrões de projeto

- **MVC (Model-View-Controller) :** O padrão MVC promove a separação de preocupações, tornando a aplicação mais modular e mais fácil de manter. Cada

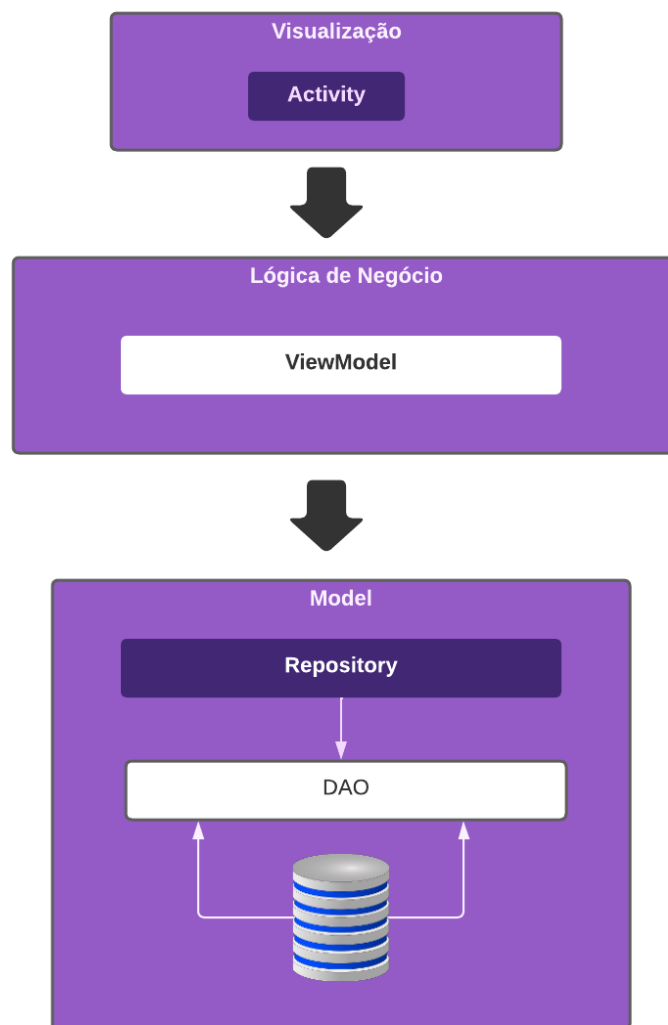
componente tem uma responsabilidade específica e pode ser desenvolvido, testado e mantido independentemente dos outros. No nosso projeto ele será útil para separar as Activities dos Services e dos Repositories.

- **DAO (Data Access Object)** : O padrão DAO é um padrão de projeto que fornece uma interface para alguns tipos de banco de dados ou outro mecanismo de armazenamento de dados. Ele separa a lógica de negócios da lógica de acesso a dados. Ele a princípio nos ajudaria a ter uma base mockada de dados enquanto não realizamos a implementação.

4. Visão Lógica

4.1.1. Arquitetura Inicial

Nesta visão arquitetural estamos usando o padrão de projeto DAO para mockar dados enquanto não implementamos o banco de dados no firebase:



4.2.2. Arquitetura final

Nesta visão arquitetural temos a versão final planejada para o sistema.

