

Political Preparedness App Design Document

1. Overview

- **Name:** Political Preparedness
- **Purpose:** To provide users with information about upcoming elections, candidates, and representatives based on their location.
- **Key Features:**
 - Display upcoming elections.
 - Provide details about candidates and representatives.
 - Allow users to search for representatives based on address.
 - Save favorite elections.
 - Navigation between screens.

2. Architecture

- **Architecture Pattern:** MVVM (Model-View-ViewModel)
- **Components:**
 - **Model:** Represents data and business logic (Repositories, Data Models).
 - **View:** Represents the UI (Activities, Fragments, XML layouts).
 - **ViewModel:** Acts as a bridge between Model and View, holds UI-related data and operations (ViewModels).
 - **Repository:** Handles data operations, provides clean API to ViewModel.
 - **Database:** Room database for local storage.
 - **Network:** Retrofit for making API calls.

3. Key Screens and Navigation

- **Screen 1: Elections List**
 - Display a list of upcoming elections.
 - Clicking on an election navigates to the election details screen.
- **Screen 2: Election Details**
 - Display detailed information about a selected election.
- **Screen 3: Representatives Search**
 - Allow users to search for representatives by address.
- **Screen 4: Representatives List**
 - Display a list of representatives based on the search.

- **Navigation:**
 - Using Navigation Component to handle navigation between fragments.

4. User Interface (UI)

Screen 1: Elections List

- RecyclerView displaying upcoming elections.
- Each item shows the election name, date, and a button to save.
- Clicking on an item navigates to Screen 2.

Screen 2: Election Details

- Display detailed information about the selected election.
- Include election name, date, description, etc.

Screen 3: Representatives Search

- EditText fields for address input (line 1, line 2, city, state, zip).
- "Search" button to initiate the search.

Screen 4: Representatives List

- RecyclerView displaying a list of representatives.
- Each item shows the representative's name, office, and party affiliation.

5. Additional Considerations

- Use ViewModel and LiveData for data handling and UI updates.
- Implement data binding for efficient UI updates.
- Use Retrofit for network requests to retrieve election and representative data.
- Utilize Room for local storage of saved elections.
- Implement SingleLiveEvent for navigation events.

Basic Application UI

Given the provided code snippets and descriptions, I've already outlined the basic UI elements in the XML layout files. These include:

- MainActivity layout with buttons for upcoming elections and representatives.
- Fragment layouts for Elections List, Election Details, Representatives Search, and Representatives List.
- RecyclerViews to display lists of elections and representatives.
- EditText fields for address input.

Remember, this is a simplified representation. In a real-world application, you would further refine and customize the UI based on your design preferences and user experience considerations.

