

# VRV Security Frontend Assignment Documentation

## Project Overview

The VRV Security Frontend Assignment project is designed to build a Role-Based Access Control (RBAC) interface using React.js. It features user-friendly components for managing users, roles, and permissions.

## Folder and File Structure Overview

### Root Directory

- README.md: Contains the project description, features, and setup instructions.
- frontend: The main folder containing all source code and resources.

### Frontend

1. .gitignore: Specifies intentionally untracked files for Git.
2. package.json: Defines project dependencies, metadata, and scripts.
3. package-lock.json: Locks dependency versions for consistency.
4. public/
  - index.html: The main HTML file for the React app.
  - manifest.json: Configures the web app's manifest for PWA support.
  - robots.txt: Controls search engine crawling.
5. src/
  - Core Files
    - App.js: Entry point for defining the app's main routes and components.
    - index.js: Main entry point to render the app.
    - index.css: Global styles for the app.
  - Components
    - adminpanel/
      - Dashboard.js: The main dashboard for administrators.
    - managepermission/
      - AddPermission.js: Handles adding permissions dynamically.
      - Permission.js: Displays and manages permissions.
    - manageroles/
      - AddRole.js: Adds new roles to the system.
      - ManageRole.js: Allows editing or deleting roles.
    - manageusers/
      - AddUser.js: Adds users to the system.

- EditUser.js: Edits user details.
  - ManageUsers.js: Lists and manages users.
  - authentication/
    - SignIn.js: Handles user sign-in functionality.
    - SignUp.js: Handles user registration.
  - common/
    - Footer.js: Footer of the app.
    - Navbar.js: Navigation bar for the app.
    - Sidebar.js: Sidebar menu for navigation.
    - SmallscreenSidebar.js: Sidebar optimized for smaller screens.
  - context/
    - ThemeContext.js: Provides theme management context for dark/light mode toggling.
  - Other Files
    - reportWebVitals.js: Measures app performance metrics.
    - setupTests.js: Configuration for testing with Jest.
6. media/: Contains images like avatars for user profiles.

## Key Features

1. Role-Based Access Control (RBAC):
  - Add, edit, and delete users, roles, and permissions.
  - Assign roles and permissions dynamically.
2. Responsive UI:
  - Optimized for different screen sizes with mobile-specific components.
3. Theme Management:
  - Integrated light and dark themes using React Context API.

## Technologies Used

- React.js: For building the user interface.
- React Router: For handling navigation.
- Styled-components: For component-level styling.
- Context API: For global state management (e.g., theme toggling).