

## **Project Review**

### **Overview**

Lendsqr is a comprehensive web application built using React, TypeScript, Next.js, and SCSS. The system focuses on managing user data efficiently, providing features such as user authentication, filtering, and various user actions. The project consists of components like Login, User Page, Filter, and Pop-Up. Additionally, the system includes testing using Jest to ensure reliability and JSON Generator to generate a mock User Data. This application is fully responsive across various devices and screen sizes for a seamless user experience

### **Key Features**

#### **Login Component**

The Login component handles user authentication. It includes form fields for email and password, with validation checks. The component integrates with the context API to manage user login status and uses local storage for persistent login information.

#### **UserData Component**

The User Data component displays a paginated list of users where users can filter the displayed data using the filter modal triggered by the "ORGANIZATION" column. The UserSummary component provides a summary or overview section, potentially offering key metrics or insights. A dynamic pop-up reveals additional options when clicking on the ellipsis icon, providing more functionality for each user

#### **User Page Component**

The User Page component displays detailed information about a specific user. It includes sections for personal details, education, socials, and guarantor information. The component also includes options to blacklist or activate users.

#### **Organization Filter Component**

The Organization Filter component identified by the organization header on the userlist table allows users to filter user data based on organization, username, email, date, phone number, and status. It provides a dropdown for organization selection and input fields for other filter criteria.

#### **Pop-Up Component**

The Pop-Up component identified by the ellipsis icon or 3 dots on each user list serves as a menu for actions related to individual users, such as viewing details, blacklisting, and activating users. It enhances the user interface by providing context-specific options.

#### **Pagination Component**

The Pagination component provides a user-friendly interface for navigating through different pages of the large dataset, providing options for moving to the previous and next pages. A dropdown menu enables users to jump directly to a specific page, providing flexibility and efficiency when dealing with a considerable amount of data.

## **Header Component**

The Header component provides essential features for navigation and user interaction like a search functionality and a responsive Menu Toggle allowing users to open or close the navigation sidebar.

## **SideNav Component**

The SideNav component provides a well-organised and visually clear navigation sidebar for navigating through various sections and includes the The log out functionality allowing users log out.

## **Testing with Jest**

The project incorporates Jest for testing, ensuring that the Login component is functioning as expected. The test suite covers scenarios like rendering components, handling form submissions, and ensuring that the login process works correctly.

## **Installation and Usage**

The project includes clear instructions in the README for installation and usage. Users can easily set up the project locally by following the provided steps. Additionally, details about the project's components, their functionalities, and how to run tests are well-documented.