

Role-Based Access Control (RBAC) UI - Documentation

Table of Contents

1. Introduction
2. Project Setup
 - Installation
3. Core Functionalities
 - Dashboard Overview
 - User Management
 - Role Management
 - Permission Management
4. Features
5. Technical Details
6. Future Enhancements
7. Usage Instructions
8. API Details
9. Screenshots
10. Conclusion

1. Introduction

The Role-Based Access Control (RBAC) UI project provides a web-based solution for managing users, roles, and permissions in an administrative environment. Administrators can assign roles to users, manage permissions for each role, and maintain control over user access and actions in the system. This project demonstrates secure, scalable, and user-friendly features to support organizations in controlling system access effectively.

2. Project Setup

Setup and Installation Instructions

Prerequisites

1. Node.js: Ensure that Node.js is installed on your system. You can download it from Node.js Official Website.
2. Visual Studio Code (VS Code): Install VS Code for code editing and debugging.
3. Git: Install Git for cloning the repository.

1. Clone the Repository

Open a terminal and run:

```
git clone <repository-link>  
cd <project-folder>
```

2. Install Dependencies

Install the necessary dependencies using npm:

```
npm i
```

3. Build and Run the Project

```
npm run dev
```

3. Modules

Admin Module

Dashboard Overview

The dashboard acts as the central interface for managing the following:

- User List: View all registered users and their roles, statuses, and actions.
- Role List: View all defined roles.
- Role Management: Edit role permissions and delete roles.
- Add Users: Add new users with a role and status.

User Management

- View Users: Displays user details, including name, email, role, and status.
- Actions:
 - Activate/Deactivate Users: Toggle user status between active and inactive.
 - Delete Users: Permanently remove a user from the system.
- Add New Users:
 - Enter user details: name, email, role, and status.
 - Save the information to add the user.

Role Management

- View Roles: Displays a list of all defined roles with options to delete roles.
- Add New Roles: Create new roles by entering a role name.
- Edit Role Permissions:
 - Assign or update permissions (e.g., Create, Read, Update, Delete) for each role.
 - Save the changes to apply updated permissions.

Permission Management

- Existing Roles: View a dropdown of existing roles.
- Edit Permissions: Customize permissions for selected roles:
 - Create
 - Read
 - Update
 - Delete
- Save Permissions: Ensure changes are updated and reflected in the role system.

User Module

The **User** module is designed for end-users with minimal access and functionality, as determined by the admin through role-based permissions.

Functionalities

1. Dashboard Access:

- View a simplified dashboard with relevant metrics or assigned tasks (if any).

2. Profile Management:

- View and edit their own profile details, such as name and email.

3. Access Based on Permissions:

- Permissions (create, read, update, delete) are defined by the admin for the user's assigned role.

- **Example:**

- A read-only user can view details but cannot add, update, or delete anything.
- A user with create permission can add new entries to designated sections.

4. Restricted Actions:

- Cannot access admin functionalities such as adding users, managing roles, or updating permissions.

4. Features

Feature	Description
Dashboard	Centralized interface for managing users and roles.
User CRUD Operations	Add, edit, delete, and toggle user statuses.
Role Management	Create, delete, and update roles.
Permission Management	Edit and assign granular permissions for each role.
Responsive UI	Adapts to various screen sizes and devices.
Security Measures	Input validation, password encryption, and role-based access control.
API Simulation	Mock API calls for user and role management.
Activate/Deactivate Users	Admin can enable or disable user access.

5. Technical Details

- **Frontend:**
 - HTML/CSS, Javascript, React.js, FramerMotion
- **Backend:**
 - Mock API
- **Database:**
 - Schema:
 - Users Table: id, name, email, password, status, role_name, role_id
 - Roles Table: id, role_name, permissions(Array)

6. Future Enhancements

- Add **audit logs** for tracking user and role modifications.
- Implement **two-factor authentication (2FA)** for secure login.
- Introduce **bulk user management** (e.g., import/export users via CSV).
- Enhance UI with **dark mode** and **theme customization**.

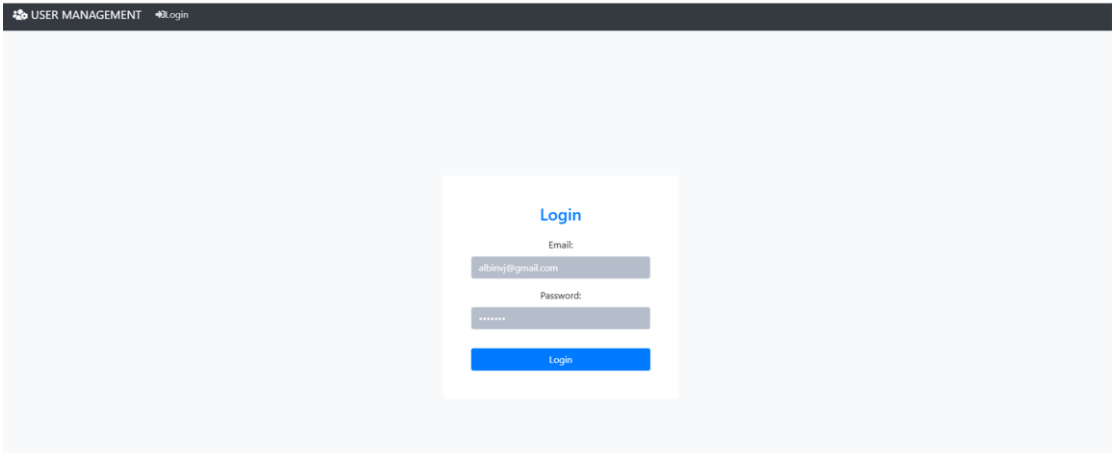
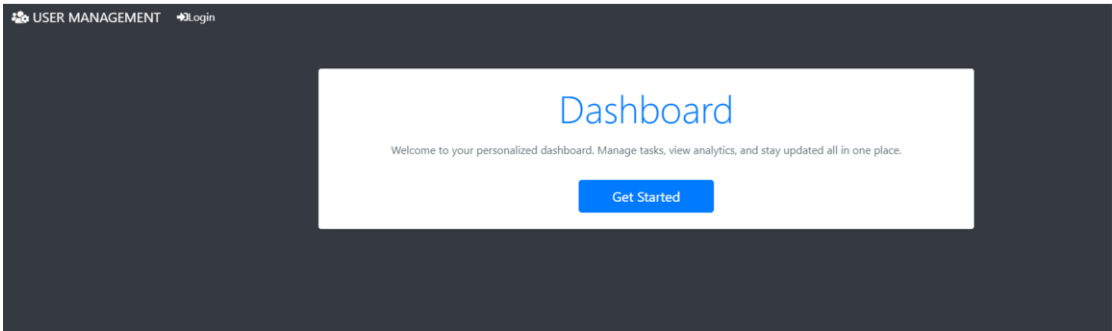
7. Usage Instructions

1. **Login:** Use credentials to access the dashboard.
2. **Manage Users:**
 - Navigate to the "User List" section.
 - Use the "Add User" button to create new users.
 - Edit or delete users using the action buttons.
3. **Manage Roles:**
 - Navigate to the "Role List" section.
 - Add roles using the "Add Role" button.
 - Edit permissions for roles in the "Role Management" section.
4. **Save Changes:** Ensure changes are saved before navigating away.

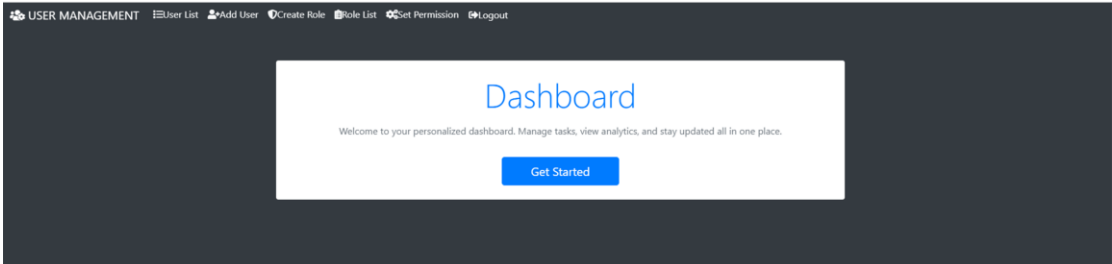
8. API Details

Endpoint	Method	Description
/users	GET	Retrieve the list of users.
/users	POST	Add a new user.
/users/:id	PUT	Update user details.
/users/:id	DELETE	Delete a user.
/roles	GET	Retrieve the list of roles.
/roles	POST	Add a new role.
/roles/:id/permissions	PUT	Update role permissions.

10. Screenshots



Admin



User List

ID	Name	Email	Role	Status	Actions
1	Cenoy P Alias	cenoy@gmail.com	admin	Active	<button>Toggle Status</button> <button>Delete</button>
2	Albin VJ	albin@gmail.com	user	Inactive	<button>Toggle Status</button> <button>Delete</button>
3	Sam	sam@gmail.com	manager	Active	<button>Toggle Status</button> <button>Delete</button>
4	avj	avj@gmail.com	user	Inactive	<button>Toggle Status</button> <button>Delete</button>
5	dave	dev@gmail.com	manager	Active	<button>Toggle Status</button> <button>Delete</button>
6	ALBIN V JOHNSON	albinvj@gmail.com	admin	Active	<button>Toggle Status</button> <button>Delete</button>

Add New User

Name

Enter full name

Role

Select a role

Email

albinvj@gmail.com

Password

.....

Status

Active

Add User

Add New Role

Role Name

Enter role name

Add Role

Role Management

Edit Permissions for Role: admin

Permissions

- ☒ create
- ☒ read
- ☒ update
- ☒ delete

Update Permissions

Cancel

Existing Roles

admin	Edit Permissions	Delete
user	Edit Permissions	Delete
manager	Edit Permissions	Delete
sub admin	Edit Permissions	Delete

Role Management

Edit Permissions for Role: sub admin

Permissions

- ☐ create
- ☒ read
- ☒ update
- ☐ delete

Update Permissions

Cancel

Existing Roles

admin	Edit Permissions	Delete
user	Edit Permissions	Delete
manager	Edit Permissions	Delete
sub admin	Edit Permissions	Delete

Create Role Role List Set Per

localhost:5173 says

Role permissions updated successfully!

OK

Edit Permissions for Role: manager

Permissions

- ☐ create
- ☒ read
- ☐ update
- ☐ delete

Update Permissions

Cancel

Existing Roles

admin	Edit Permissions	Delete
user	Edit Permissions	Delete
manager	Edit Permissions	Delete
sub admin	Edit Permissions	Delete

Role Management

Edit Permissions for Role: user

Permissions

- ☐ create
- ☐ read
- ☐ update
- ☐ delete

Update Permissions

Cancel

Existing Roles

admin	Edit Permissions	Delete
user	Edit Permissions	Delete
manager	Edit Permissions	Delete
sub admin	Edit Permissions	Delete

Role List

ID	Role Name	Actions
1	admin	<button>Delete</button>
2	user	<button>Delete</button>
3	manager	<button>Delete</button>
4	sub admin	<button>Delete</button>

user page

USER MANAGEMENT User List Add User Role List Logout

Access Denied

Login

Your account is inactive. Please contact support.

Email:

avj@gmail.com

Password:

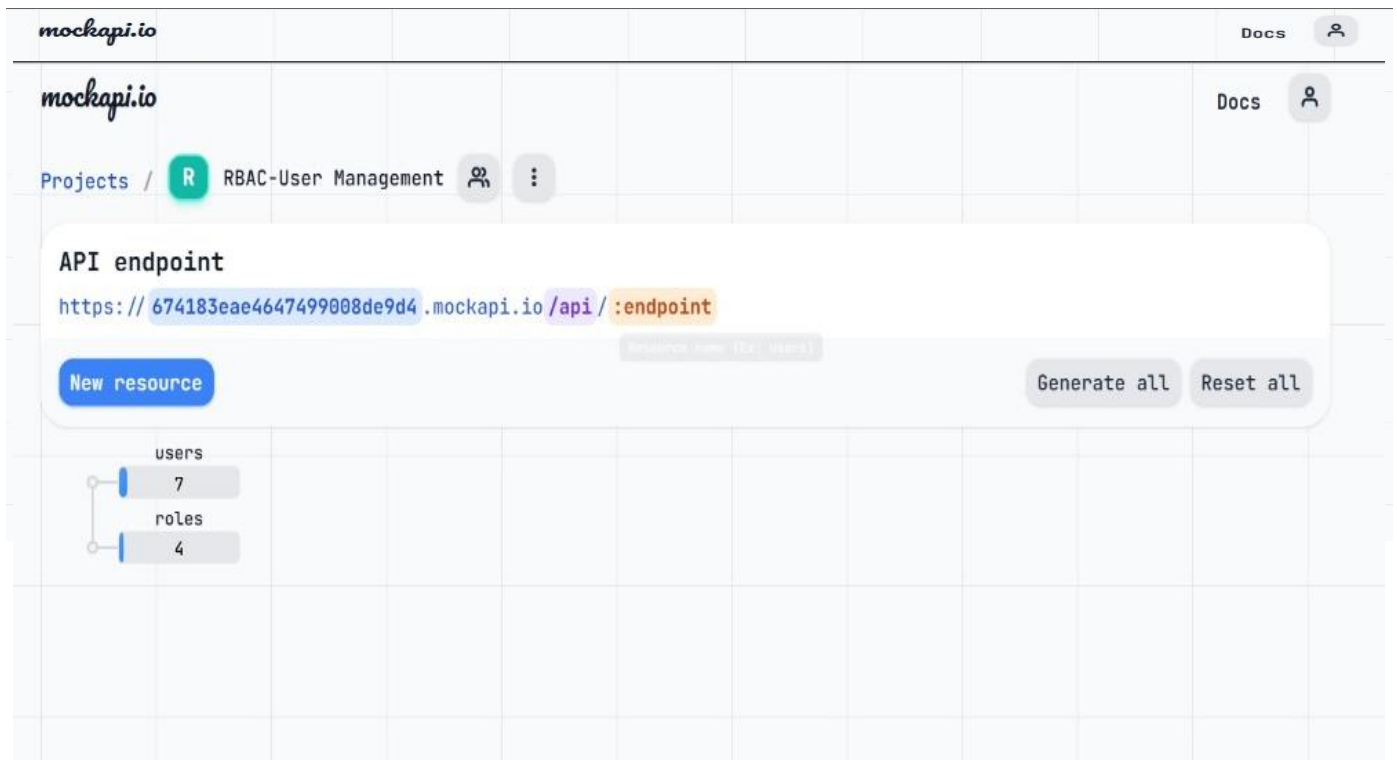
Login

Access denied. You do not have permission to view this page.

User List

ID	Name	Email	Role	Status	Actions
----	------	-------	------	--------	---------

Mock API -Backend



10. Conclusion

The **RBAC UI** is a comprehensive solution for managing users, roles, and permissions efficiently. It offers flexibility, scalability, and security to adapt to any organization's needs. The project is designed with extensibility in mind, allowing future enhancements such as audit logs and bulk operations.