

# Project Documentation

## Application Overview

This application is a full-stack asset finance management platform with the following features:

- **User Registration and Authentication:** Secure registration, login, and JWT-based authentication.
- **Profile Management:** View and update user profile information.
- **Finance Application Management:** CRUD operations for finance applications.

The application is built with a React frontend and a Node.js/Express backend and utilizes MongoDB for data storage.

## 1. Prerequisites

Ensure you have the following installed on your system:

- **Node.js** (version 16 or higher)
- **npm** (Node Package Manager)
- **MongoDB** (for local setup, or MongoDB Atlas for a cloud database)
- **Git** (for version control)

## 2. Project Setup

### Step 1: Clone the Repository

**Repository Link:** <https://github.com/RakenShahi/asset-finance-management>

Clone the project repository and navigate to the project folder

➔ `git clone <repository-url> cd <project-folder>`

### Step 2: Set Up the Backend

#### 1. Navigate to the Backend Directory from the Terminal:

➔ `cd backend`

#### 2. Install Dependencies:

Install the required Node.js packages:

➔ `npm install`

### 3. Create a .env File:

In the backend directory, create a .env file for environment variables:

➔ MONGO\_URI=<Your MongoDB connection string>

#### Working URI

MONGO\_URI=mongodb+srv://raken:raken@afmt.fkw5h.mongodb.net/assetfinanancemanagement?retryWrites=true&w=majority&appName=AFMT

If this URI is used, the default user is mentioned below in User section.

### 4. Start the Backend Server:

To start the server, run:

➔ npm start

- The server should run on <http://localhost:5001> unless otherwise specified.
- Check that MongoDB is connected, and the server is listening on the specified port.

### Step 3: Set Up the Frontend

#### 1. Navigate to the Frontend Directory:

➔ cd ../frontend

#### 2. Install Dependencies:

Install the required Node.js packages for the frontend:

➔ npm install

#### 3. Configure API Endpoints (If Necessary):

Update the API URLs in your frontend code if needed to match your backend deployment URL. For local testing, ensure API calls in axios use <http://localhost:5001>.

#### 4. Start the Frontend Application:

➔ npm start

This should start the frontend on <http://localhost:3000>.

## Usage Instructions

### 1. Register a New User:

- Go to the registration page and enter your details. This will create a new account in the MongoDB database.
- If the URI given above is used in the .env file:  
Default user: [user1@gmail.com](mailto:user1@gmail.com)  
Password: user1

### 2. Login:

- After registration, log in to access the main application. Upon successful login, a JWT token is stored in localStorage for session management.

### 3. Profile Management:

- Navigate to the Profile page to view and update profile information. You can update your name, email, and password.

### 4. Finance Application Management:

- Navigate to the Application List page to view all finance applications.
- Use the Add/Edit Application page to create new applications or edit existing ones.
- Use the Delete button in the Application List to remove applications.

## Additional Information

- **Logout:** The JWT token is removed from localStorage upon logout to secure the session.
- **Error Handling:** The application has basic error handling, which can be expanded as needed.

This concludes the documentation for setting up and running the asset finance management platform. Please feel free to reach out if additional clarification is needed.