

Digital Thesis Repository

Team Details:

Group 1

1. Sanketh Bolishetti (1002144534)
2. Pradeep Akula (1002140196)
3. Soumya Reddy Bheemreddy (1002193346)
4. Kausheel Bhagavathula (1002118457)
5. Kintur Ketan Shah (1002178072)
6. Madhupriya Bendilli (1002124302)

Login Role Details

We have 3 users for this project:

1. Admin
2. Scholar
3. User

Prerequisites

Before setting up the project, make sure you have the following installed:

Homebrew (for macOS users)
Node.js and npm (can be managed with nvm)
Prisma (for database migrations)

Setup Instructions

Install nvm (Node Version Manager)

Homebrew to install nvm:

brew install nvm

Install Dependencies:

a. Frontend setup:

```
cd frontend  
npm install
```

b. Backend setup:

```
cd backend  
npm install
```

Backend Database setup:

→ .env please edit mysql username and password

→ In Mysql create a schema tms

→ Run Prisma to apply migrations for your database schema:

```
npx prisma migrate dev
```

→ Seed your database with initial data:

```
npm run seed
```

Run local host:

c. Frontend:

```
npm run dev
```

d. Backend setup:

```
npm run dev
```

Setting Up Token in Postman:

To avoid notification errors after logging in, follow these steps:

1. Import the Postman Collection

- Import the ScholarVault.postman_collection.json file into Postman.

2. Create an Environment

- Create a new environment in Postman called tms.
- Set the following variable (adjust the value as needed):
 - BASE_URL: http://localhost:3000

3. Retrieve CSRF Token

- Run the following **GET** request in Postman:

```
GET {{BASE_URL}}/api/auth/csrf
```

- Copy the csrfToken value from the response.

4. Authenticate with Credentials

- Run the following **POST** request in Postman:

POST {{BASE_URL}}/api/auth/callback/credentials

- In the body of the request, include the CSRF token you retrieved earlier. Example:

```
json
{
  "csrfToken": "YOUR_CSRF_TOKEN",
  "email": "admin@scholarvault.com",
  "password": "scholarvault"
}
```

5. Set the Token in Postman Environment

- After a successful login, retrieve the authentication token from the response.
- Add the token to the tms environment as a variable, e.g., csrfToken.

This ensures smooth functionality without triggering login notification errors.

Accounts for Testing

The following accounts are available for testing the platform:

Admin

Email: admin@scholarvault.com

Password: scholarvault

Scholar

Email: scholar@scholarvault.com

Password: scholarvault

User

Email: user@scholarvault.com

Password: scholarvault