**Authentication API Documentation**

**Base URL:**

http://localhost:5000/api/auth

This API provides routes for user authentication, including registration, login, and accessing user profile information.

**1. Register a New User**

**Endpoint**

* **POST** /register

**Description**

* Registers a new user in the system. The user needs to provide their name, email, password, and role.

**Request Body**

* **Content-Type:** application/json
* **Body Parameters:**
  + name (string, required): Full name of the user.
  + email (string, required): Unique email address for the user.
  + password (string, required): Password for the account.
  + role (string, required): User role, which can be one of the following:
    - startup
    - clerk
    - authority
    - stakeholder

**Response**

* **Success (200 OK):**
  + **Body:**

json

{

"token": "<JWT Token>"

}

* + **Description:** A JWT token is returned, which should be used for subsequent authenticated requests.
* **Error (400 Bad Request):**
  + **Body:**

json

{

"msg": "User already exists"

}

* + **Description:** This error occurs if a user with the provided email already exists.
* **Error (500 Internal Server Error):**
  + **Body:**

json

{

"msg": "Server error"

}

* + **Description:** General server error during registration.

**2. Login a User**

**Endpoint**

* **POST** /login

**Description**

* Authenticates a user and returns a JWT token if the credentials are valid.

**Request Body**

* **Content-Type:** application/json
* **Body Parameters:**
  + email (string, required): Registered email address of the user.
  + password (string, required): Password associated with the account.

**Response**

* **Success (200 OK):**
  + **Body:**

json

{

"token": "<JWT Token>"

}

* + **Description:** A JWT token is returned, which should be used for subsequent authenticated requests.
* **Error (400 Bad Request):**
  + **Body:**

json

{

"msg": "Invalid email credentials"

}

* + **Description:** This error occurs if the email does not exist.
  + **Body:**

json

{

"msg": "Invalid password credentials"

}

* + **Description:** This error occurs if the password does not match the one stored in the system.
* **Error (500 Internal Server Error):**
  + **Body:**

json

{

"msg": "Server error"

}

* + **Description:** General server error during login.

**3. Get Authenticated User Details**

**Endpoint**

* **GET** /me

**Description**

* Retrieves the details of the authenticated user.

**Headers**

* **Authorization:** Bearer <JWT Token>
  + The JWT token received from the /login or /register route should be passed in the Authorization header.

**Response**

* **Success (200 OK):**
  + **Body:**

json

{

"\_id": "<user\_id>",

"name": "<user\_name>",

"email": "<user\_email>",

"role": "<user\_role>",

"date": "<registration\_date>"

}

* + **Description:** The details of the authenticated user are returned, excluding the password.
* **Error (401 Unauthorized):**
  + **Body:**

json

{

"msg": "Token is not valid"

}

* + **Description:** This error occurs if the JWT token is missing or invalid.
* **Error (500 Internal Server Error):**
  + **Body:**

json

{

"msg": "Server error"

}

* + **Description:** General server error while retrieving user details.

**Security Considerations**

* All routes that require authentication should include a JWT token in the Authorization header using the Bearer scheme.
* The JWT token is used to securely identify the user without the need to store session data on the server.

**HTTP Status Codes**

* **200 OK**: The request was successful.
* **400 Bad Request**: The request could not be understood or was missing required parameters.
* **401 Unauthorized**: Authentication failed or the user does not have permissions for the desired action.
* **500 Internal Server Error**: An error occurred on the server.

This documentation should be used as a reference when implementing or interacting with the authentication API endpoints. Make sure to handle responses and errors properly in your frontend to ensure a smooth user experience.