

API Documentation

Overview

This API supports an authentication and task management system divided into two main sections:

1. **Authentication** - For handling user accounts and admin functionality
2. **Task Manager** - For creating, updating, and managing tasks associated with user accounts

Authentication API

Roles:

- **User:** Can create an account, log in, view and manage tasks, and access their profile
- **AdminServer:** Can manage all user accounts and oversee user activity, including task management actions

Endpoints

Authentication Endpoints

1. POST /api/authentication

- **Description:** Creates a new user account and uploads a profile image
- **Role:** User/Adminserver
- **Request:**
 - **Body:**
 - username: String
 - email: String
 - password: String
 - image: File (via Multer for image upload)
- **Response:**
 - 201 Created: Account successfully created
 - 400 Bad Request: Validation error
 - 404 If user or Email exist

2. POST / api/authentication /login

- **Description:** Logs a user in, providing a JWT token for task management
- **Role:** User/Adminserver

- **Request:**
 - **Body:**
 - email: String
 - password: String
- **Response:**
 - 200 OK: Returns token
 - 401 Unauthorized: Invalid Email or Password

3. GET / api/authentication/:userid

- **Description:** Displays the profile of the logged-in user
- **Role:** User/Adminserver
- **Headers:**
 - Authorization: Bearer Token (JWT)
- **Response:**
 - 200 OK: User profile details

4. PUT / api/authentication/:userid

- **Description:** Allows a user to update their account information using their token
- **Role:** User/Adminserver
- **Headers:**
 - Authorization: Bearer Token (JWT)
- **Request:**
 - **Body:**
 - username: Optional, String
 - email: Optional, String
- **Response:**
 - 200 OK: Account updated successfully
 - 400 Bad Request: Validation error

5. DELETE /auth/delete/:userId

- **Description:** Allows AdminServer to delete a user account.
- **Role:** AdminServer

- **Headers:**
 - Authorization: Bearer Token (JWT)
- **Response:**
 - 200 OK: User account deleted
 - 403 Forbidden: User not authorized

6. GET /auth/dashboard

- **Description:** Display all user's task activities for AdminServer, including task creation, updates, deletions, and completions.
 - **Role:** AdminServer
 - **Headers:**
 - Authorization: Bearer Token (JWT)
 - **Response:**
 - 200 OK: Dashboard data of user activities.
-

Task Manager Endpoints

1. POST /api/tasks/:userid

- **Description:** Creates a new task
- **Role:** User
- **Headers:**
 - Authorization: Bearer Token (JWT)
- **Request:**
 - **Body:**
 - title: String
 - description: String
 - dueDate: Date
 - completed: Boolean => (default: false)
- **Response:**
 - 200 Created: Task successfully created
 - 400 Bad Request: Validation error

2. PUT /api/tasks/:userid/:taskId

- **Description:** Updates an existing task by the original creator (title, description, due date, and statuses of completion)
- **Role:** User
- **Headers:**
 - Authorization: Bearer Token (JWT)
- **Request:**
 - **Body:**
 - title: Optional, String
 - description: Optional, String
 - dueDate: Optional, Date
 - completed: Optional, Boolean
- **Response:**
 - 200 OK: Task updated
 - 403 Forbidden: User not authorized to update this task

3. DELETE /api/tasks/:userid/:taskId

- **Description:** Deletes a task created by the user.
- **Role:** User
- **Headers:**
 - Authorization: Bearer Token (JWT)
- **Response:**
 - 200 OK: Task deleted.
 - 403 Forbidden: User not authorized to delete this task.

4. GET /api/tasks/:userid/:taskId

- **Description:** Retrieves tasks based on specific filters
- **Role:** User
- **Headers:**
 - Authorization: Bearer Token (JWT)
- **Request:**

- **Query Parameters:**
 - completed: Optional, Boolean (true for completed tasks, false for incomplete)
 - date: Optional, String (today, prevdays, nextdays)
- **Response:**
 - 200 OK: Filtered task list.