Software Requirements and Design Document

For

Group <7>

Version 3.0

Authors:
Alek Coupet
Bilal Taleb
Sayed Haidar Alhussaini

1. Overview (5 points)

We are developing a task management web application using the MERN stack. The application will enable users to manage personal and team tasks, collaborate within organizations, and track tasks via a calendar feature. It will include role-based permissions for administrators to manage users and tasks.

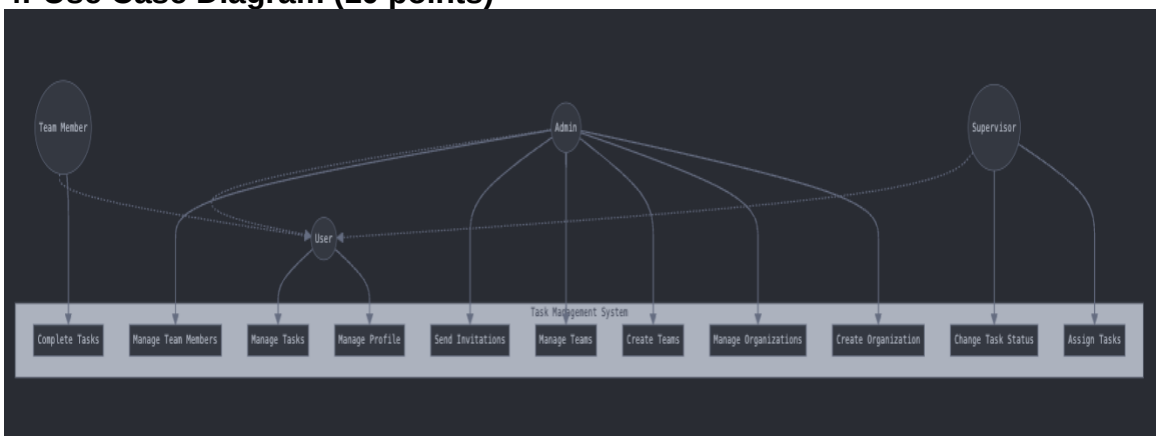
2. Functional Requirements (10 points)

1. User registration and login system – High priority
2. Task creation, editing, and deletion – High priority
3. Role-based access control for administrators and regular users – High priority
4. Calendar integration to track task deadlines – Medium priority
5. Ability for users to mark tasks as complete or incomplete – High priority

3. Non-functional Requirements (10 points)

1. Application must be scalable
2. Application must be responsive and work on multiple devices
3. User data must be secure with encrypted login credentials
4. The User interface must be simple and easy to interact with

4. Use Case Diagram (10 points)



The use case diagram represents the Task Management System's core functionalities and user interactions. A User can sign up, log in, and log out from the web application. They can check the dashboard including task lists along with their profile. Users can manage their profiles by updating personal information and changing passwords. They can also manage basic tasks through the dashboard interface.

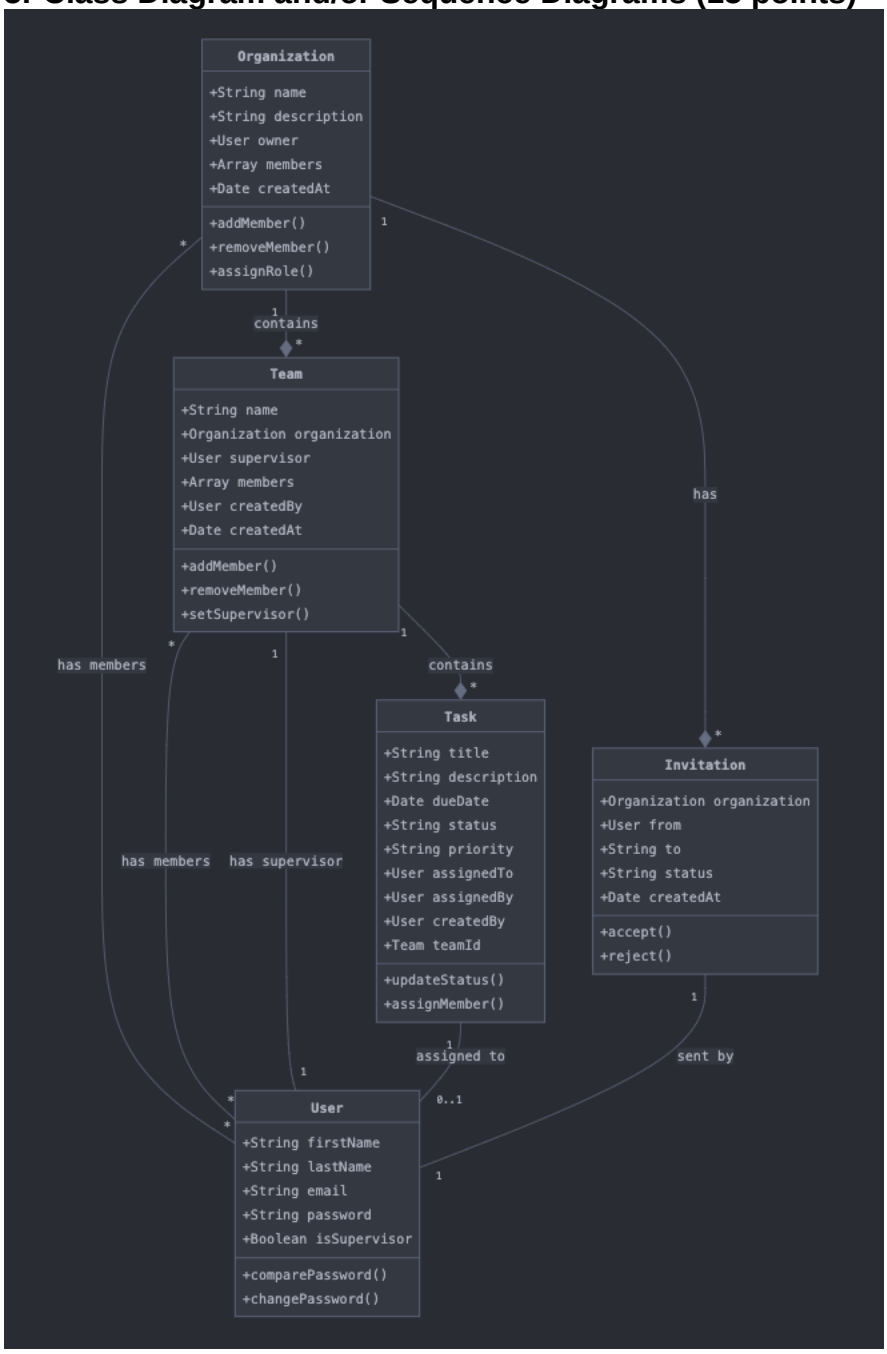
An Admin is a user with extended privileges who can create tasks, manage organizations, create teams, and manage users. They have the ability to create new organizations and oversee their operations. Admins can create teams within organizations and manage team compositions. They can send invitations to potential members and manage organizational roles and permissions.

A Supervisor is a specialized user role that focuses on task management within teams. They can assign tasks to team members and monitor task progress. Supervisors have the authority to change task status and oversee the work of team members under their supervision.

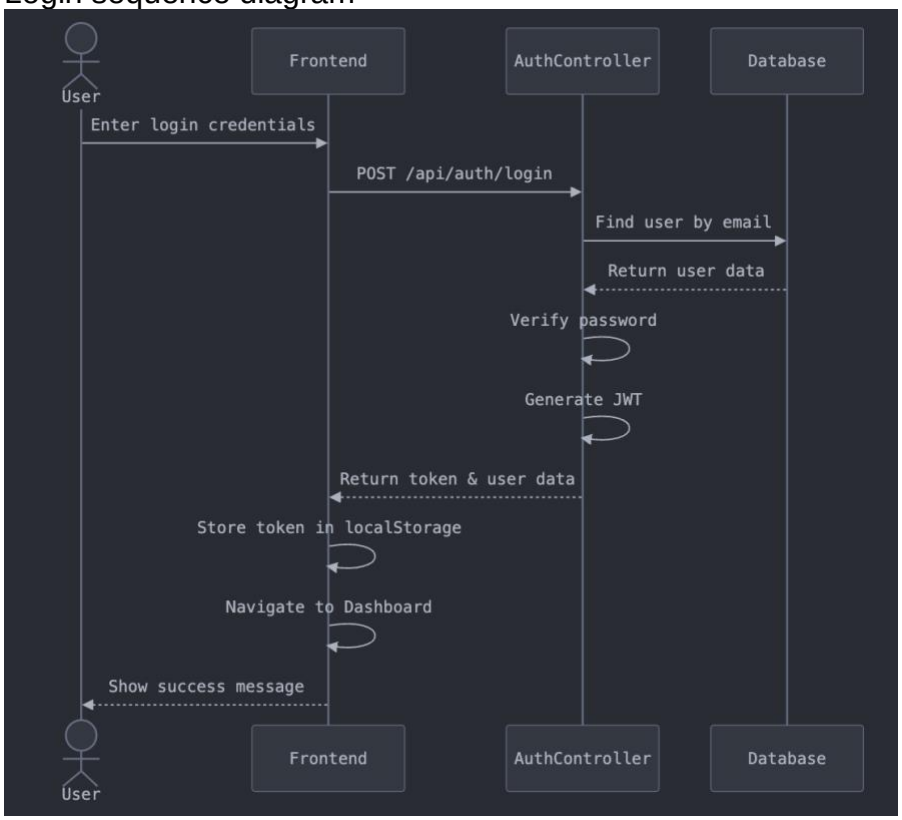
A Team Member is a user who primarily interacts with assigned tasks. They can view tasks assigned to them and complete these tasks as needed. Team members can mark their tasks as complete and view relevant team information. All these roles inherit basic user capabilities while having their specific responsibilities and permissions within the system.

The system is designed to maintain a clear hierarchy and separation of duties, with each role having specific responsibilities that contribute to the overall task management workflow. Each actor interacts with the system through their designated use cases, ensuring proper task management and organizational structure.

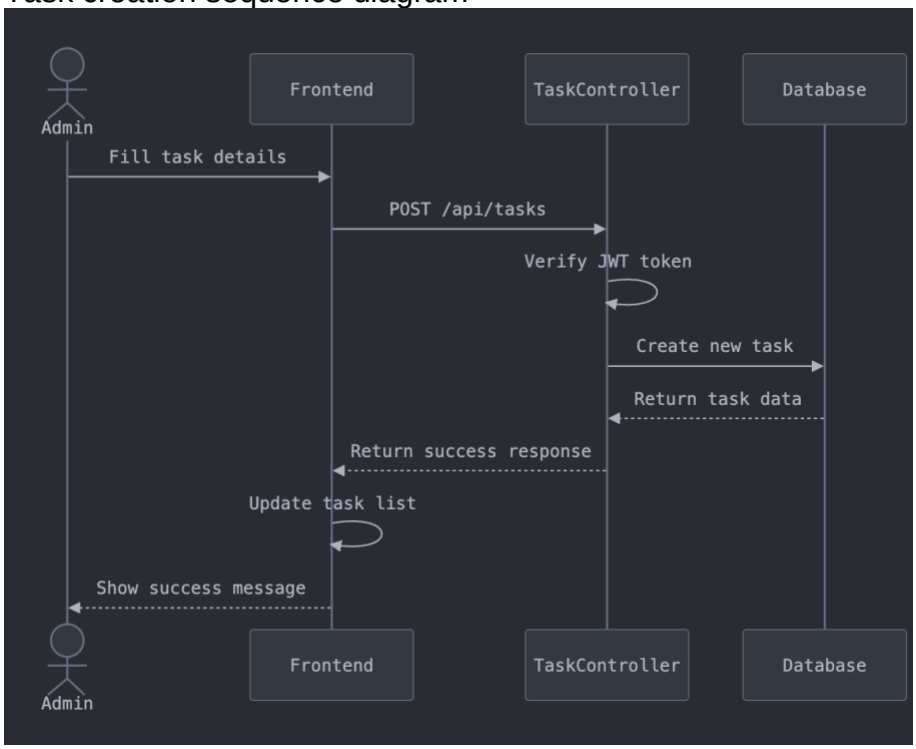
5. Class Diagram and/or Sequence Diagrams (15 points)



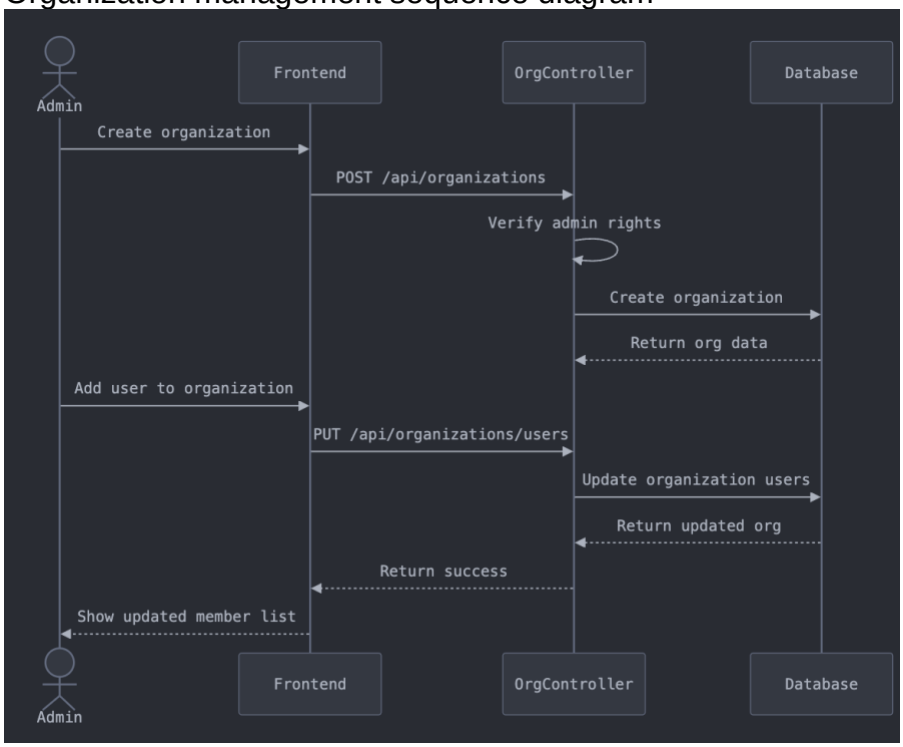
Login sequence diagram



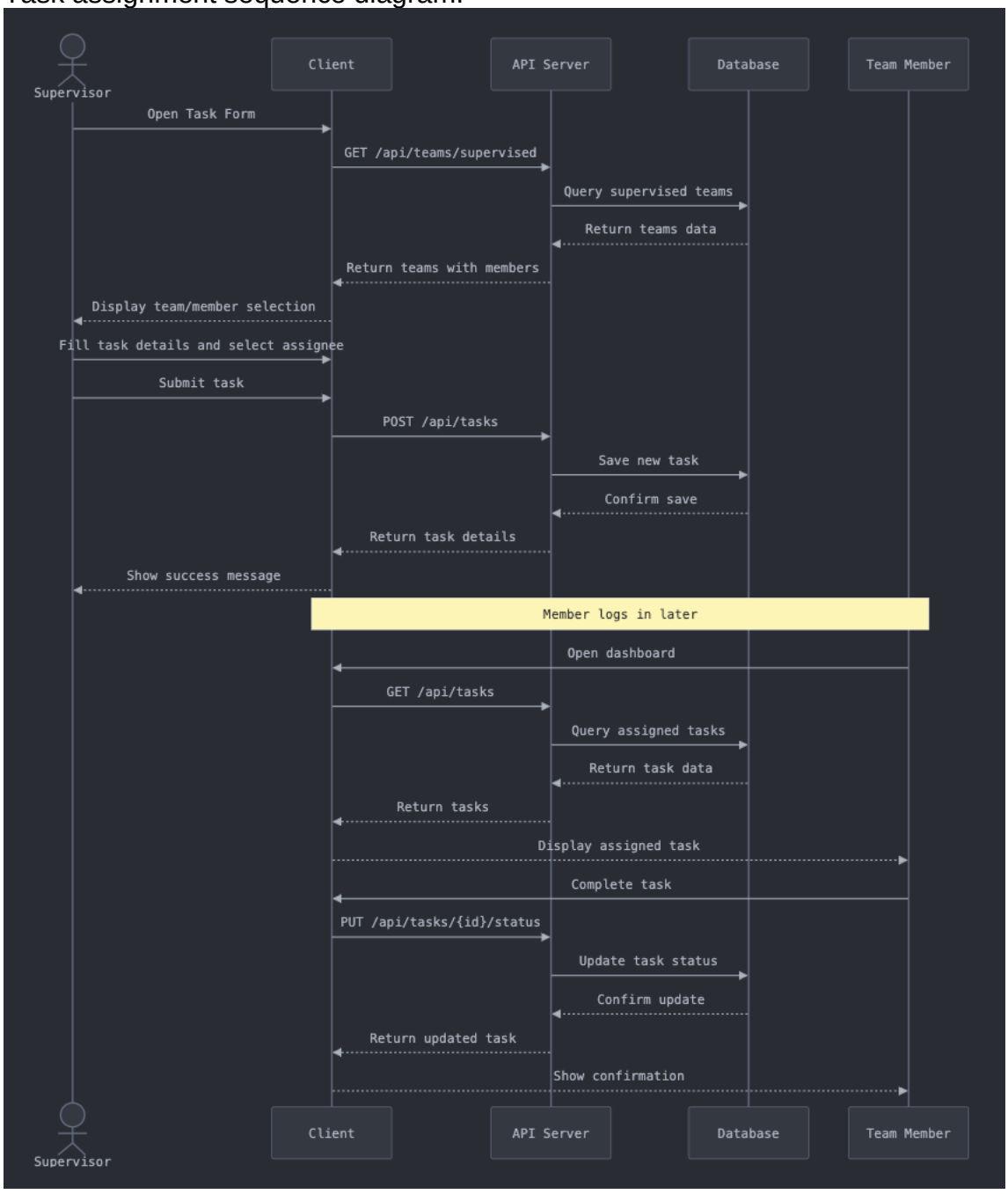
Task creation sequence diagram



Organization management sequence diagram



Task assignment sequence diagram:



6. Operating Environment (5 points)

The software will operate on web browsers and mobile devices. It will be built using React (frontend) and Node.js (backend) on MongoDB.

7. Assumptions and Dependencies (5 points)

● We assume that MongoDB will be reliable and scalable.
● We depend on third-party libraries for features like authentication (JWT) and calendar integration (FullCalendar).
● We depend on third-party libraries for front-end implementations (NPM modules).