**Project Objectives:**

Develop a user-friendly Online Task Management System that allows individual users to manage their tasks efficiently.

**Functional Requirements:**

1. User Registration and Authentication:
2. Users should be able to register for an account with a unique username and password.
3. Users should be able to log in securely to access their task lists.
4. Task Management:
5. Users can create, edit, and delete tasks.
6. Tasks can have a title, description, due date, and priority.
7. Tasks can be organized into multiple task lists (e.g., personal, work).
8. Users can mark tasks as complete.
9. Task Lists:
10. Users can create, edit, and delete task lists.
11. Each task list can have a unique name.
12. Users can switch between different task lists.
13. User Profile:
14. Users can view and update their profile information.
15. Profile information may include name, email, and profile picture.
16. Search and Filter:
17. Users can search for tasks and task lists.
18. Users can filter tasks based on priority, due date, or completion status.

**Non-Functional Requirements:**

1. User Interface:
2. The user interface should be intuitive and responsive, accessible on various devices.
3. Modern design principles should be followed for a visually appealing user experience.
4. Security:
5. User data should be stored securely, including hashed passwords.
6. Data transmission should be encrypted.
7. Protection against common web vulnerabilities (e.g., XSS, CSRF) should be implemented.
8. Performance:
   1. The system should provide fast response times for user interactions.
   2. Efficient database queries should be used to retrieve data.
9. Scalability:
10. The system should be designed to handle potential growth in the number of users and tasks.
11. Documentation:
12. Comprehensive user documentation should be provided.
13. Code documentation and comments for developers should be maintained.
14. Testing:
15. Comprehensive testing should be performed, including unit testing and integration testing.
16. User acceptance testing should be conducted to ensure usability.

**Constraints and Assumptions:**

1. The system will be developed using Angular (TypeScript) for the front end and ASP.NET Core for the back end.
2. The project will have a defined budget and timeline.
3. The system will be hosted on a cloud platform (e.g., Azure).
4. Users are assumed to have basic internet connectivity and web browser access.