

# Project Proposal



## Database Management Systems

### Project Name:

Online Task Management System

.....

### Group Members:

Syed Abdul Maarij (21k-4547)

.....

Instructor: Fizza Aqeel.

Lab Instructor: Shaheer Ahmed

# Contents

1. Introduction .....	3
2. Project Description .....	3
2.1. Project Objective .....	3
2.2. Project Boundaries .....	3
2.3. Pre-existing Work .....	4
2.4. Uniqueness Aspect of Our Project .....	4
2.5. Schema .....	4

# 1. Introduction

- **Inspiration:** Develop an effective task organization system to improve efficiency and optimize task management processes.
- **Client:** Educator
- **Project Deliverables:** A fully operational task management platform enriched with supplementary features.
- **Project Duration:** A project timeline of four months.

## 2. Project Description

### 2.1. Project Objective:

- **Functional Objectives**
  1. Task Generation and Removal.
  2. Monitoring Task Status (e.g., 'pending,' 'in-progress,' 'completed').
  3. Managing Task Deadlines.
  4. Implementing Fundamental User Authentication.
- **Technological Objectives**
  5. Frontend Technology: HTML, CSS, JavaScript, Bootstrap and React.
  6. Backend Technology: Node.js
  7. Database: Integrating a Simple SQL Database (e.g., SQLite)
- **Quality Objectives**
  8. Usability: Guaranteeing an Intuitive and User-Friendly Interface
  9. Accessibility: Ensuring Cross-Device and Cross-Browser Compatibility

### 2.2. Project Boundaries:

- Deliver a task management system with the following functionalities:
  - Task creation and removal.
  - Task status tracking.
  - Deadline management.
  - Basic user authentication.
  - Intuitive and user-friendly interface.
  - Exclude advanced features like multi-user collaboration and intricate task assignment.

## 2.3. Pre-existing Work

- Assessment of current task management systems completed.
- Existing systems offer essential task management capabilities.

## 2.4. Uniqueness Aspect of Our Project

- Our project strives to offer a straightforward and user-friendly solution.
- Emphasizing simplicity, usability, and essential task-related functionalities.
- Augmenting existing work while catering to distinct user needs.

## 2.5. Schema

<div>+-----+   User   +-----+   UserID (PK)     Username     Password (hashed)     Email     FirstName     LastName     ProfileImage   +-----+</div>	<div>+-----+   Project   +-----+   ProjectID (PK)     ProjectName     Description     StartDate     EndDate     ManagerID (FK)   +-----+</div>	<div>+-----+   Task   +-----+   TaskID (PK)     TaskName     Description     DueDate     Priority     Status     ProjectID (FK)     AssignedTo (FK)     CreatedBy (FK)   +-----+</div>
<div>+-----+   Comment   +-----+   CommentID (PK)     TaskID (FK)     UserID (FK)     Content     CreatedAt   +-----+</div>	<div>+-----+   Attachment   +-----+   AttachmentID (PK)     TaskID (FK)     UserID (FK)     Filename     ContentType     FileSize   +-----+</div>	<div>+-----+   Notification   +-----+   NotificationID (PK)     TaskID (FK)     UserID (FK)     Content     CreatedAt     Seen     TaskStatus   +-----+</div>
<div>+-----+   Admin   +-----+   AdminID (PK)     UserID (FK)     Role   +-----+</div>		

### **1. User Table:**

- Enhanced with additional fields such as a hashed password and profile image to enrich user profiles.

### **2. Project Table:**

- Maintains its core structure but can incorporate additional project-related details if required.

### **3. Task Table:**

- Expanded with new fields including the task creator (CreatedBy), a priority level, and task status. It establishes foreign keys for project and user Assignment.

### **4. Comment Table:**

- Allows users to post comments on tasks, linking both the task and the user Who made the comment. It also stores comment content and creation timestamp.

### **5. Attachment Table:**

- Stores file attachments related to tasks, with references to the task and user who uploaded the attachment. It records the filename, content type, and file size.

### **6. Notification Table:**

- Keeps a record of notifications for users, referencing the relevant task and user. It includes notification content, creation timestamp, and a flag to indicate if the notification has been viewed.

# Thank You

.....

