**OOP PROJECT REPORT**

**TASK AND PROJECT MANAGEMENT APPLICATION**

**Team Members:**

**TAHA KHAN (22I-2335)**

**AAWAIZ (22I-0845)**

**Introduction (Aim or Motivation):**

The aim of the Task and Project Management Application project is to develop a comprehensive solution that empowers individuals and teams to organize tasks efficiently and enhance collaboration. Motivated by the shortcomings of existing task management tools, the goal is to address challenges related to task organization, team collaboration, and overall project management.

**Background (Research & Project Selection):**

Research into the task and project management domain revealed gaps in existing solutions, prompting the selection of this project. The need for a feature-rich application that facilitates effective task organization and collaboration led to the development of the Task and Project Management Application.

**Project Specification:**

The Task and Project Management Application will encompass the following key features:

1. **User Authentication:**
   * Implemented user registration and login functionality.
2. **Task Creation:**
   * Users can create tasks with details such as task name, description, due date, and team members.
3. **Project Management:**
   * Users can create project to group related tasks together.
4. **Task Team Management:**
   * Task/project owners can add/remove team members.
5. **Task Status Management:**
   * Tasks can be in "to-do," "in-progress," and "completed" states.
6. **Due Date Alerts:**
   * Reminders for upcoming task/project due dates.
7. **Task Notes and Comments:**
   * Users can attach notes to tasks, and there's a comment section for team discussion.
8. **Task Tagging:**
   * Each task can have multiple tags for easy categorization.
9. **Task Searching:**
   * Search tasks by name, tags, due date, or project name.
10. **User Profile:**
    * Users have a profile section displaying projects and tasks with due dates.
11. **Report Generation:**
    * Users can generate easy-to-read reports with tasks, projects, and deadlines.

**Problem Analysis:**

The Task and Project Management Application addresses the following problems:

* **Lack of a comprehensive task and project management tool:**
  + The application provides a holistic solution for managing tasks and projects efficiently.
* **Inefficiencies in task organization and collaboration:**
  + Features such as project management, team collaboration, and task status management address these inefficiencies.
* **Difficulty in tracking project progress:**
  + The application offers tools for monitoring task status, due dates, and project timelines.

**Solution Design (Project Detail, Functionality, and Features):**

Project Detail:

The application is designed with a user-friendly interface to facilitate efficient task and project management. It offers a centralized platform for users to organize tasks and collaborate seamlessly.

Functionality and Features:

* **User Authentication:**
  + Provides secure user registration and login functionality.
* **Task Creation:**
  + Allows users to create detailed tasks, including due dates and team member assignments.
* **Project Management:**
  + Enables users to create projects to group related tasks, enhancing overall organization.
* **Task Team Management:**
  + Allows task/project owners to manage team members efficiently.
* **Task Status Management:**
  + Enables tracking of task progress with "to-do," "in-progress," and "completed" states.
* **Due Date Alerts:**
  + Sends reminders to users for upcoming task and project due dates.
* **Task Notes and Comments:**
  + Facilitates communication through task notes and a comment section for team discussions.
* **Task Tagging:**
  + Allows users to categorize tasks with multiple tags for easy retrieval.
* **Task Searching:**
  + Enables users to search for tasks based on various criteria, including name, tags, due date, or project name.
* **User Profile:**
  + Provides a user profile section displaying all projects and tasks along with their due dates for efficient management.
* **Report Generation:**
  + Empowers users to generate comprehensive reports containing tasks, projects, and deadlines.

**Implementation & Testing**

**Methodologies Employed:**

* + The project is mainly based on the composition and we have used all the concepts of oop.
* **Results:**
  + The application demonstrated robustness and reliability in various testing phases.

**Project Breakdown Structure (Workload Distribution with Timeline)**

The main and global functions and linking the methods from global function were created by Taha. The classes and creating all the functions in the code were done by Aawaiz.

**Conclusion (Summary & Discussion)**

In conclusion, the Task and Project Management Application successfully addresses the identified problems and provides a robust solution for efficient task and project management. Acknowledging the collaborative efforts of team members, the application's success is evident in its user-friendly interface and feature-rich functionality. Despite any limitations, the project lays the foundation for future enhancements and improvements.

**UML DIAGRAM:**

+-------------------------------------+

| Project |

+-------------------------------------+

| - project\_id\_: int |

| - project\_name\_: string |

| - project\_description\_: string |

| - project\_due\_date\_: string |

| - no\_of\_tasks\_: int |

| - project\_users\_: User[MAX\_PROJECT\_USERS] |

| - project\_tasks\_: Task[MAX\_NO\_OF\_TASKS] |

+-------------------------------------+

| + get\_project\_name\_(): string |

| + get\_project\_id\_(): int |

| + get\_due\_date(): string |

| + createUser(): int |

| + authenticateUser(): bool |

| + displayProjectDetails(): void |

| + displayAllProjectUsers(): void |

| + displayUserDetails(id: int): void |

| + generateReport(id: int): void |

| + createTask(name: string, due\_date: string): int |

| + AddTask(user\_id: int, task\_id: int): void |

| + addTaskTags(user\_id: int, task\_id: int, new\_task\_tag: Tags): void |

| + changeTaskStatus(user\_id: int, task\_id: int, task\_status: string): void |

| + displayTasks(): void |

| + displayUserTasks(user\_id: int): void |

| + addTaskNotes(user\_id: int, task\_id: int, add\_note: PublicNotes): void |

| + addTaskNotes(user\_id: int, task\_id: int, add\_note: PrivateNotes): void |

| + displayPublicNotes(task\_id: int): void |

| + displayPrivateNotes(user\_id: int, task\_id: int): void |

| + editTaskDueDate(task\_id: int, new\_due\_date: string): void |

| + sendDueDateAlert(user\_id: int): void |

| + taskExists(task\_id: int): bool |

| + searchUser(id: int): User |

| + searchUser(user\_name: string): User |

+-------------------------------------+

|

|

v

+-------------------------------------+

| User |

+-------------------------------------+

| - user\_id\_: int |

| - user\_name\_: string |

| - user\_email\_: string |

| - user\_password\_: string |

| - user\_role\_: string |

| - vacant\_id\_slots\_: int[10] |

| - user\_count\_: int |

+-------------------------------------+

| + set\_user\_name(name: string): void |

| + set\_user\_email(email: string): void|

| + set\_user\_password(password: string): void |

| + viewProfile(): void |

| + createUser(username: string, email: string, password: string, role: string): int |

| + authenticateUser(email: string, password: string): int |

| + displayUserDetails(): void |

| + generateReport(): void |

| + AddTask(task: Task): void |

| + addTaskTags(task\_id: int, new\_task\_tag: Tags): void |

| + changeTaskStatus(task\_id: int, task\_status: string): void |

| + displayAssignedTasks(): void |

| + addTaskNotes(task\_id: int, add\_note: PrivateNotes): void |

| + addTaskNotes(task\_id: int, add\_note: PublicNotes): void |

| + displayPrivateNotes(task\_id: int): void |

| + sendDueDateAlert(): void |

+-------------------------------------+

|

|

v

+-------------------------------------+

| Task |

+-------------------------------------+

| - task\_id\_: int |

| - task\_name\_: string |

| - due\_date\_: string |

| - task\_count: int |

+-------------------------------------+

| + createTask(name: string, due\_date: string): int |

| + addTaskTags(new\_task\_tag: Tags): void |

| + changeTaskStatus(task\_status: string): void |

| + displayTaskDetails(): void |

| + addTaskNotes(add\_note: PrivateNotes): void |

| + addTaskNotes(add\_note: PublicNotes): void |

| + displayPublicNotes(): void |

| + displayPrivateNotes(): void |

| + editTaskDueDate(new\_due\_date: string): void |

+-------------------------------------+

|

|

v

+-------------------------------------+

| Tags |

+-------------------------------------+

| - tag\_name\_: string |

+-------------------------------------+

| + Tags(name: string): void |

+-------------------------------------+

+-------------------------------------+

| PrivateNotes |

+-------------------------------------+

| - note\_name\_: string |

| - note\_text\_: string |

+-------------------------------------+

| + PrivateNotes(name: string, text: string): void |

| + displayNoteDetails(): void |

+-------------------------------------+

+-------------------------------------+

| PublicNotes |

+-------------------------------------+

| - note\_name\_: string |

| - note\_text\_: string |

| - author\_: string |

+-------------------------------------+

| + PublicNotes(name: string, text: string, author: string): void |

| + displayNoteDetails(): void |

+-------------------------------------+