COMP1002 – Advanced Python
Course Project Proposal

Project Groups:

MÜŞERREF EBRU ERDEN-232010020012

CELAL ÇATAL-232010020019

MELİH KAAN DİREK-232010020015

1. Project Title:

Task Tracking System

---

2. Project Objective:

- The objective of this project is to address the problem of inefficient task management in small teams by developing a task tracking system. The system will allow users to create, assign, and track the progress of tasks in an organized manner.
- This system will fulfill the need for a simple, easy-to-use tool that helps teams keep track of their tasks and deadlines.

---

3. Project Scope:

- Areas Covered by the Project:
    - This project will focus on developing a command-line-based task tracking system where users can create tasks, update them, and track their progress in a streamlined way.
- Technologies and Libraries to be Used:
    - Technologies: Python 3.x
    - Libraries: No external libraries will be used. All operations will be done using Python's built-in libraries:
        - File Handling: Task data will be stored in text files.
        - Data Storage: Instead of a database, tasks will be saved in a JSON file or a text file.
        - Time Management: Python's datetime library will be used to manage task deadlines.
- Types of Users and Expected Interactions:
    - Users will be able to create, update, delete, and assign tasks.

    - The system will not have roles like "Admin" or "User"—all users will have equal access to task management functionalities.

---

4. Methodology:

- Python Libraries and Tools to be Used:
    - File Handling: Task data will be stored and read from text files (e.g., .txt or .json).
    - datetime: To store and manage task deadlines.
    - Command-line Interface: The user will interact with the system via a simple command-line interface.
- Basic Functionality and Data Flow of the Application:
    - Users will be able to add tasks, each containing:
        - Task Title
        - Task Description
        - Assignee
        - Status (Not Started, In Progress, Completed)
        - Due Date
    - Users will be able to update task status and view tasks based on filters like due date, assignee, or status.
    - All task data will be stored in a text or JSON file, and the system will save changes to these files to ensure data persistence.

---

5. Expected Outcomes:

- Final Product or Results:
    - A command-line-based task tracking system with core functionalities such as adding, updating, deleting, and listing tasks.
- User Interface and Reports:
    - The system will be entirely text-based, and users will interact with it through the command line.
    - There will be no graphical reports, but tasks will be displayed as lists and can be filtered based on status, assignee, and deadlines.

---

6. Timeline:

Phase	Description	Duration
Analysis and Research	Requirement analysis and system design	1 week
Design	Designing the system flow, file structure, and data format	1 week
Coding	Developing core functionalities and file handling logic	2 weeks
Testing and Debugging	System testing and bug fixing	1 week

---

7. Resources:

- Data Sources:
  - Sample task data will be manually created for testing purposes.
- Documentation:
  - Python’s official documentation for file handling and the datetime library.
- Books and References:
  - Online tutorials for Python command-line applications and file handling.

---

8. Conclusion:

- This project is important because it provides a simple and efficient way for small teams to manage tasks and track their progress.
- By utilizing only Python's built-in libraries, the system will be lightweight, requiring no additional software or tools.
- The command-line interface will ensure that the system is fast and easy to use, allowing users to stay organized and meet their deadlines effectively.