COS 102 – INTRODUCTION TO PROBLEM SOLVING

Project 1 - Practical Examination

Project Title: Smart Task Scheduler with Prioritization

Duration: 1 Week

Pan-Atlantic University has observed that many of its Computer Science students struggle with managing multiple academic and extracurricular responsibilities effectively. From managing assignment deadlines and project milestones to preparing for quizzes and personal commitments, students often find it challenging to prioritize tasks and allocate time efficiently.

To address this issue, the Department of Computer Science proposed the development of a **Smart Task Scheduler** to help students track and manage their daily responsibilities more effectively. Mr. George Uwagbale, the COS 102 course lecturer, suggested that this project would be an excellent opportunity for students to demonstrate their object-oriented programming skills using Python and the Tkinter GUI toolkit.

As a student developer, your task is to design and implement a user-friendly desktop application that enables users to manage tasks with clear visibility into their **priorities**, **deadlines**, and **estimated durations**.

The system must meet the following requirements:

1. Users should be able to add a new task with:
   a. Task title
   b. Optional Description
   c. Priority Level (e.g., 1 = High, 2 = Medium, 3 = Low)
   d. Deadline (Date and Time)
   e. Estimated Duration (in minutes or hours)
2. The system must allow users to edit and delete existing tasks.
3. The system should be able to display a list of all tasks with sorting based on:
   a. Priority level
   b. Urgency (how close the task is to its deadline)
4. The system should display visual cues for overdue and urgent tasks.
5. The system must be easy to use. Ensure all buttons are clearly labeled and user-friendly.
6. The system should validate user input properly.

Project 2 - Practical Examination

Project: Budget Simulator

Duration: 1 Week

Financial literacy is an essential life skill, especially for students learning to manage limited resources such as pocket money, stipends, or part-time earnings. At Pan-Atlantic University, many Computer Science students have expressed challenges in tracking their spending habits and maintaining a consistent budget throughout the semester.

To help address this, the Department of Computer Science has introduced a **Budget Simulator** project aimed at equipping students with both financial awareness and practical programming experience. Mr. George Uwagbale, the COS 102 course lecturer, proposed this project as a way for students to apply object-oriented programming concepts while developing a real-world application using **Python** and **Tkinter**.

You have been tasked with designing and implementing a desktop application that allows users to create and manage a monthly budget, track their income and expenses, and simulate how financial decisions affect their overall spending. This tool should promote responsible financial planning through an interactive and user-friendly interface.

The system must meet the following requirements:

1. Allow users to set a monthly budget (a total amount of money available to spend).
2. Enable users to create **budget categories** (e.g., Food, Rent, Transportation, Savings) and allocate specific amounts to each category from the monthly budget.
3. Provide functionality to **log income and expenses** with:
   - Date
   - Amount
   - Category
   - Description (Optional)
4. Simulate spending over the month by updating:
   - Total spent per category
   - Total remaining budget
   - Unallocated funds (if any)
5. Display **feedback or alerts** in the GUI, such as:
   - Overspending in a category
   - Approaching budget limit
   - Remaining balance after each entry

Project 3 - Practical Examination

Project: Movie Ticket Booking System

Duration: 1 Week

In a bid to modernize its ticketing process and improve customer experience, **Panaview Cinemas**, a mid-sized movie theater located in Lagos, is seeking to replace its outdated manual booking system with a digital solution. The current process—based on physical ticket counters and handwritten logs—often results in long queues, double bookings, and poor seat management.

To solve this problem, the theater's management has approached the Department of Computer Science at Pan-Atlantic University with a request to build a **Movie Ticket Booking System** that can simulate a real cinema environment. The system should allow customers to view available movies, select showtimes, and book seats using a visually interactive interface.

Mr. George Uwagbale, the COS 102 course lecturer, saw this as a valuable learning opportunity for students to apply object-oriented design, GUI development with **Python and Tkinter**, and persistent data storage in a practical context.

You have been selected to lead the design and development of this system. Your goal is to build a desktop application that replicates the experience of booking tickets at a real movie theater—with dynamic seat layouts, clear visual indicators for availability, and persistent records of all bookings.

The system must meet the following requirements:

1. Allow users select a movie from a list of available movies
2. Allow users view all available **time slots** for the selected movie
3. The system should display available and booked seats visually and allow users select available seats.
4. The system should prevent booking of already booked seats
5. Upon seat selection and confirmation, the system should:

   • Persist the booking to a database

   • Reflect the new booking in the seating chart

6. The system should provide visual cues for available, selected, and booked seats.
7. The system must be **easy to use and intuitive**, with clear labeling of movies, time slots, and seat layout.

8. All bookings must be **saved persistently** so that closing and reopening the app retains previous bookings.
9. The interface should update dynamically when:
   - A new movie is selected
   - A new time slot is chosen

Project 4 - Practical Examination

Project: Bug Tracking System (Mini JIRA Clone)

Duration: 1 Week

As part of its ongoing efforts to improve software development practices, **SIBAS**, a startup incubated by Pan-Atlantic University, has identified a growing need for a simple, in-house issue tracking system. The development team frequently encounters bugs and feature requests, but their current method of tracking issues—via spreadsheets and emails—has become inefficient and error-prone.

To solve this, the company has requested a lightweight bug tracking system that mimics the core functionality of popular tools like JIRA but is simple enough to use within small teams. The Computer Science Department sees this as a great opportunity for students to work on a real-world software engineering tool, reinforcing their understanding of object-oriented programming, GUI design, and database integration.

Mr. George Uwagbale, the COS 102 course lecturer, has tasked you with developing this **Mini JIRA Clone** using **Python and Tkinter**. Your mission is to build a desktop application that allows users to create projects, manage tickets, assign responsibilities, and filter issues based on specific criteria.

The system must meet the following requirements:

1. The system should allow users create new projects with a name and description.
2. The system should allow users to **add tickets/bugs** to a project. Each ticket must include:
   - A short **title**
   - **Status** (To Do, In Progress, Done)
   - **Severity level** (e.g., Low, Medium, High)
   - **Assignee** (the person responsible)
3. The system should store all data persistently in a database.
4. Include **search/filter options** to:

   - View tickets by **project**

   - Filter by **status** or **severity**

Project 5 - Practical Examination

Project Title: Quiz Application (MCQ Test System)

Duration: 1 Week

As part of its push to modernize academic evaluation and make self-assessment tools more accessible, the **Department of Computer Science at Pan-Atlantic University** has proposed the development of a lightweight digital quiz platform to aid both lecturers and students in tracking learning progress.

Many lecturers still rely on paper-based or manually graded assessments, which can be time-consuming and error-prone. At the same time, students lack a reliable tool to practice tests in an interactive, feedback-driven environment. To address this gap, Mr. George Uwagbale, the COS 102 course lecturer, has challenged students to build a **Tkinter-based Quiz Application** that simulates a real testing experience using **Multiple Choice Questions (MCQs)**.

You have been selected to build this application, which should allow users to take quizzes, navigate through questions, select answers, and receive real-time scoring and performance feedback.

The system must meet the following requirements:

1. The system must allow users to load quiz questions from a file (JSON or text). Each question must include:
    -   Question text
    -   Four answer options
    -   Correct answer
2. The system should display one question at a time.
3. The system should keep track of the selected option for each question.
4. On submission, the system should display the total correct answers, percentage score, and the total wrong answers.
5. Each quiz should have a timer.