Introduction: Smart Task Scheduler with Priority Queues

This report documents the complete lifecycle of the project. The subsequent sections will elaborate on the system design, detailing the Task class and the TaskManager logic built around a PriorityQueue. We will describe the implementation of the Graphical User Interface (GUI) using Swing for user interaction, the reminder system using timers for proactive notifications, and the serialization mechanism for data persistence. Finally, the report will discuss the application's functionality, potential future enhancements, and a conclusion on the project's success in achieving its objectives. The primary goal of this project is to create a robust, user-friendly, and intelligent tool that empowers individuals to manage their tasks more effectively.

Abstract: This report details the design, implementation, and functionality of a Smart Task Scheduler, a desktop application developed in Java to enhance personal productivity through intelligent task management. The core of the system is built upon a PriorityQueue data structure, which dynamically sorts tasks based on user-defined priority levels, ensuring that the most urgent items are always presented first.

The application features a graphical user interface (GUI) constructed with Java Swing, providing intuitive controls for users to add, view, and remove tasks. Each task encapsulates critical properties such as a title, a priority integer (where lower numbers signify higher urgency), and a deadline. A key feature of the scheduler is its proactive reminder system, implemented using the java.util.Timer class, which generates alert notifications for tasks with impending deadlines.

For data persistence, the state of the task queue is automatically serialized to a binary file upon exit and deserialized upon startup, allowing users to retain their tasks across sessions. Furthermore, the system provides foundational filtering capabilities to view tasks based on their priority. This project demonstrates a practical application of fundamental data structures, object-oriented design principles, and event-driven programming to create an efficient and user-centric organizational tool.

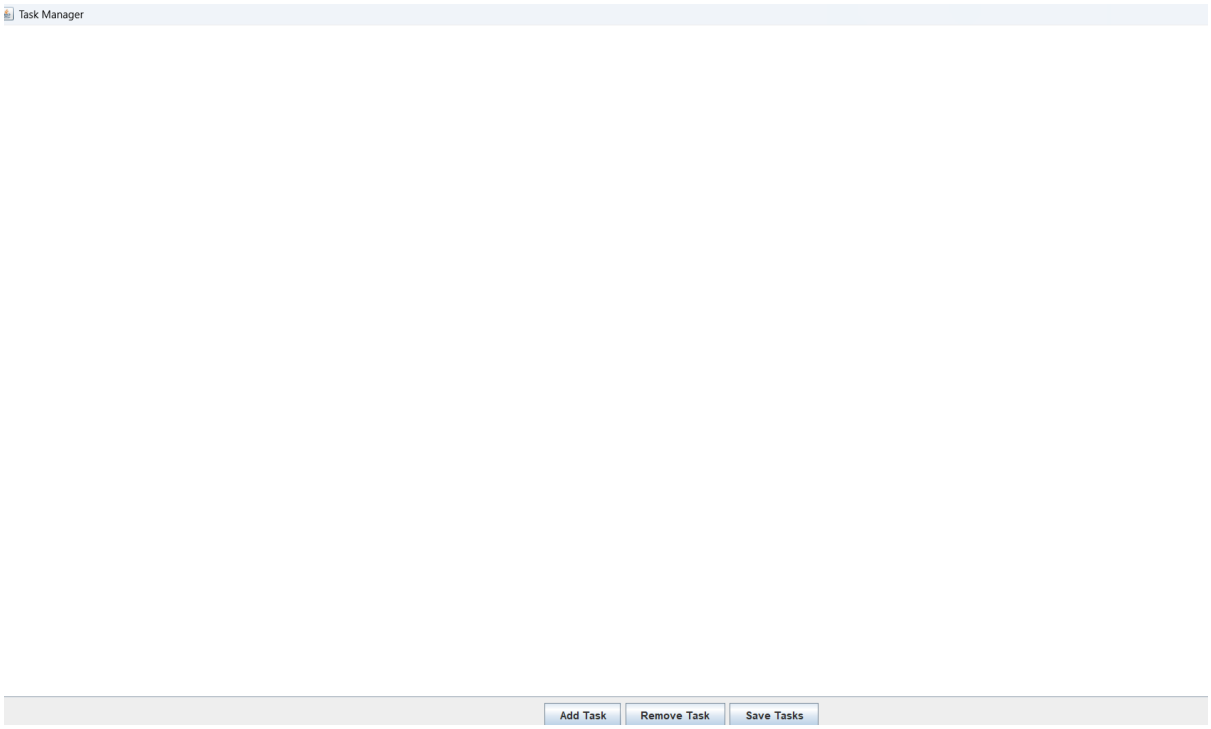
Tools Used: Java, VScode, Git,Maven

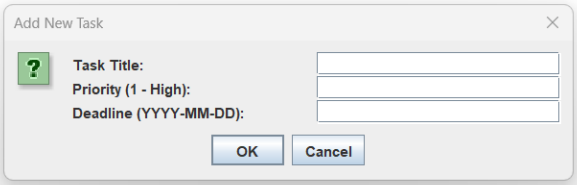
Steps Involved in Building the Project:
-Create project using maven.

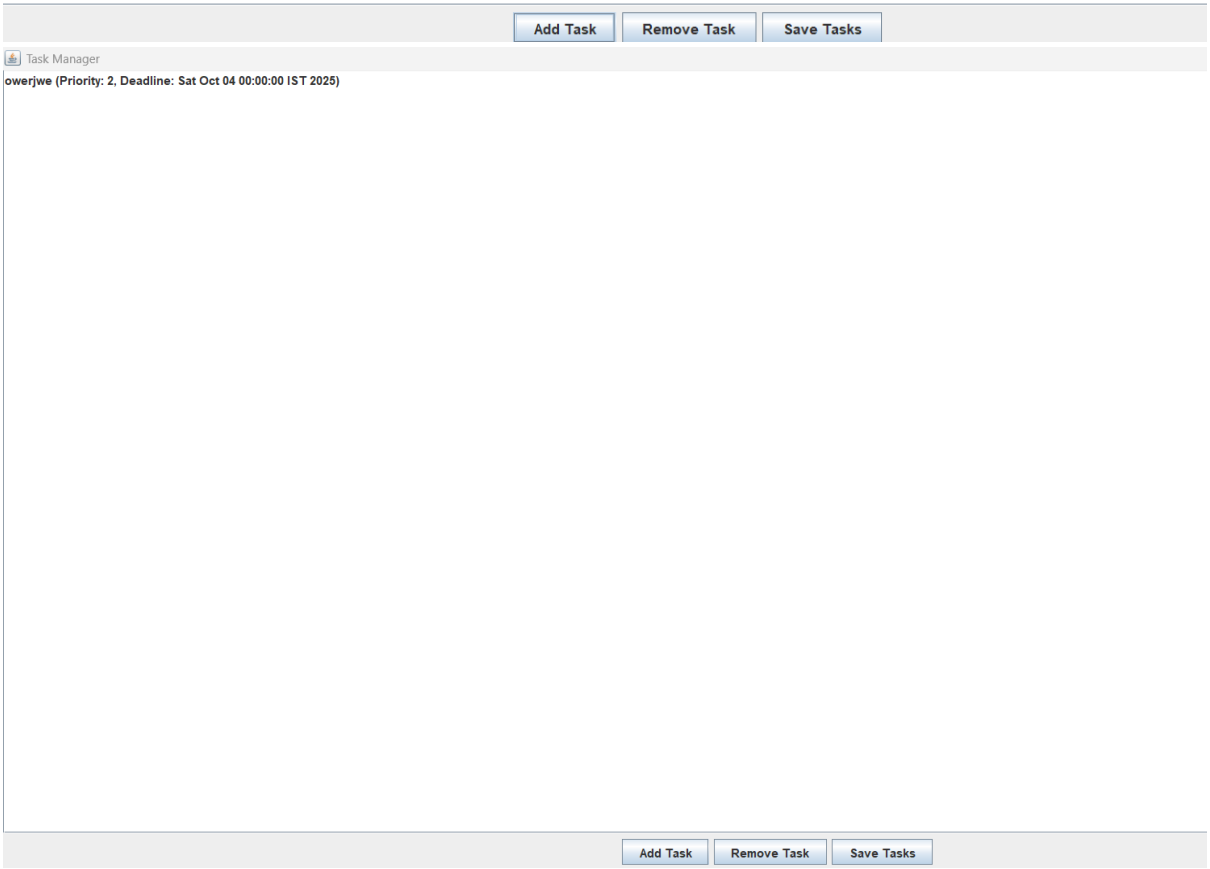
-Import required packages and write code

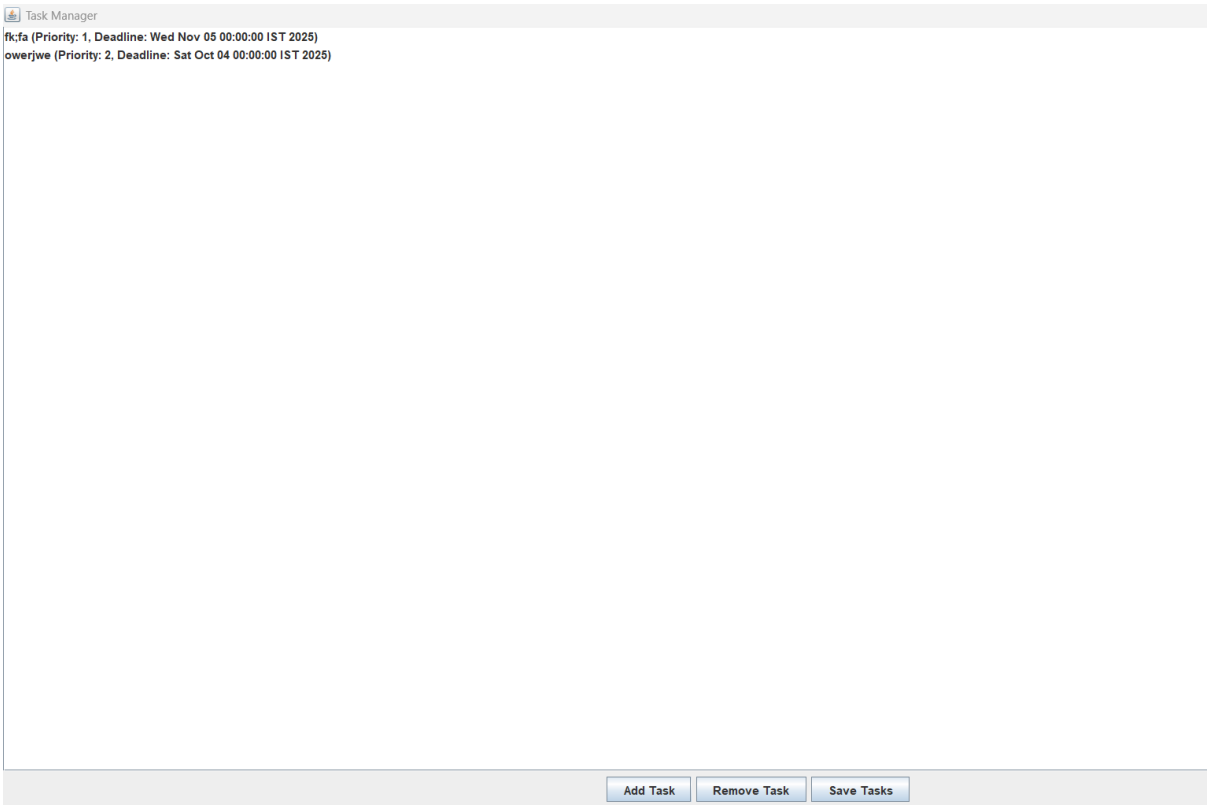
-To run the code: cd src, javac App.java, java App

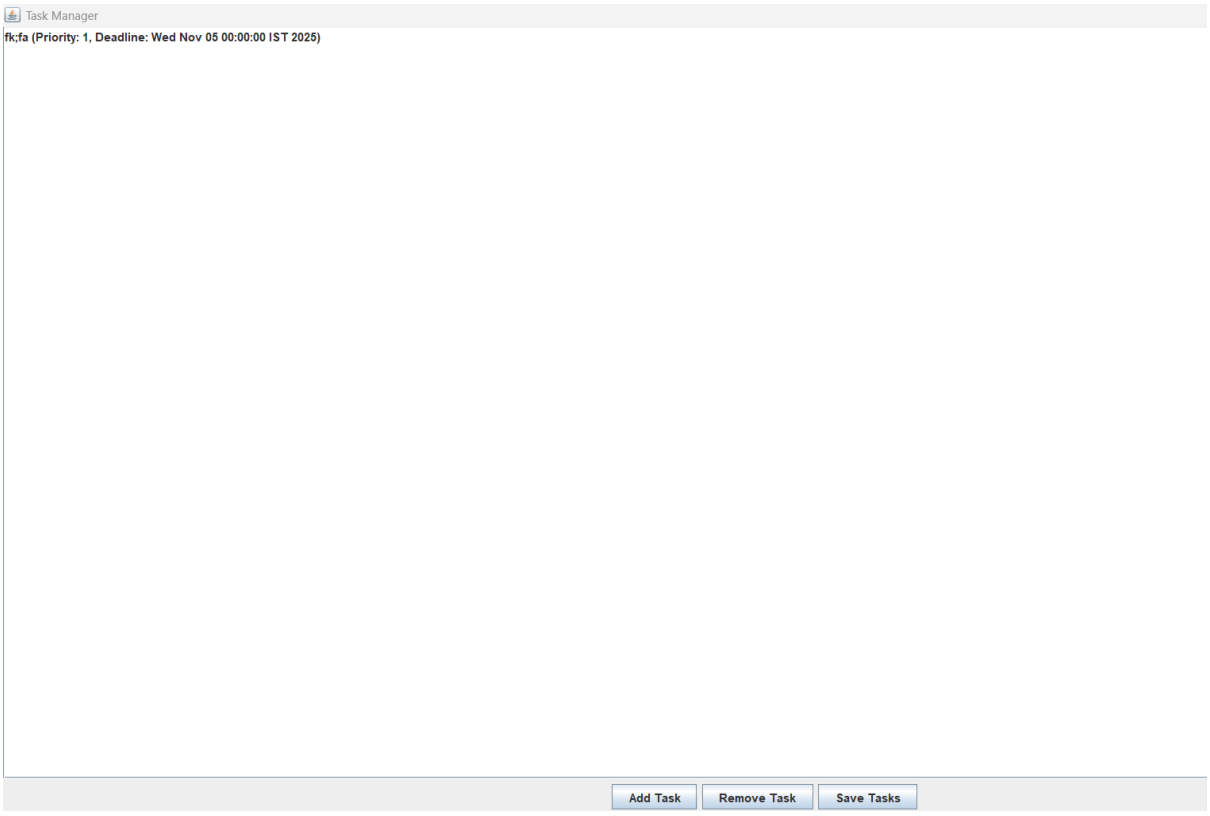
Screenshots:




Add New Task
Task Title:
Priority (1 - High):
Deadline (YYYY-MM-DD):
OK
Cancel


Add Task
Remove Task
Save Tasks
Task Manager
owerjwe (Priority: 2, Deadline: Sat Oct 04 00:00:00 IST 2025)
Add Task
Remove Task
Save Tasks





Conclusion:

The Smart Task Scheduler successfully delivers a handy and straightforward tool for managing daily tasks. It takes the hassle out of prioritizing by automatically sorting tasks by urgency and reminding users of upcoming deadlines. The application is easy to use, with a

clean interface for adding and removing tasks. For the future, the app could be made even more helpful with features like a calendar view or the ability to organize tasks into different projects. Overall, this project created a practical and functional tool that accomplishes its goal of being a simple, smart, and helpful task manager.