

## **/Final project paper – group 13/**

### **a) Problem Definition:**

We have thousands of ideas going through our minds every day, and it's hard for us to pay attention to every detail. We need an assistant to help us mark those important dates; we need a place for us to collect these ideas based on their importance; we need instructions to tell us where to start.

Based on our knowledge, the solution is the TO-DO LIST.

There's a limited amount of human focus. The brain chooses to remember and forget certain things when the work is overloaded. That's why we forgot some important meetings, quizzes, and deadlines.

The to-do list is the place for us to list down all the important events by date. By doing so, we will have a clear picture of what we need to do in the following week/month. If we stick to it, then we won't miss important dates anymore. It assists us to remember and reminds us of its existence.

But what are we supposed to do after listing down the tasks? or where are we supposed to start?

These two situations remind us about the second significance of the to-do list- priority.

One of the important elements in the to-do list is the date. Marking the date tells us how much time we have left for finishing that task, and how many tasks we still have after finishing the current-working one. By getting that information, we can arrange and schedule the tasks' timeline in a more efficient way. The date gives us instructions about where to start the task.

The problem we identified in real life is the need for an efficient and organized way to manage our tasks and deadlines. Often, it can be challenging and overwhelming to keep track of what needs to be done and when especially when the list is already too long that you try to keep track of it with your head but end up missing several things. Our code aims to solve this problem by providing a simple and user-friendly to-do list application so the user may track their task based on the priorities of the task's deadlines.

**b) Idea and Approach: State your ideas and how you intend to solve the problem. Discuss how you applied logical and computational thinking to address the issue.**

We initially assess the resolution options for the predicament at hand, which in our case involves effectively managing our tasks and addressing them based on their priorities, particularly the deadlines. One potential approach is to tackle all the tasks without assigning priority based on the deadlines, instead focusing on their difficulty levels.

Another alternative is to mark the calendar with significant dates and attach post-it notes to our study desk wall, highlighting the respective tasks for each day. We then evaluate the pros and cons of these viable solutions, recognizing that neither is entirely practical. The first option risks missing deadlines despite completing all tasks, while the second option lacks the portability, we desire in a reminder system.

Consequently, we have devised a solution that overcomes the shortcomings of the previous two possibilities by creating a 'to-do list' program that prioritizes tasks based on their deadlines. This program offers a convenient and accessible platform for users to monitor their agendas anytime and anywhere. By incorporating a user-friendly interface and compelling features, our program facilitates enhanced agenda management and fosters increased motivation for completing all tasks."

**c) Code Introduction: Provide an overview of your code, explaining the methodologies you used. Highlight how the code contributes to the solution but remember that the emphasis is on problem-solving rather than complex programming.**

We create a list with a dictionary inside to track the agendas, deadlines, and status. When a user inputs a task, the program will automatically treat the task and sort it based on the deadlines. This will help the user to know which agenda and which deadline will come first, helping them to prioritize the agendas. We also have status features to help them track whether the task is completed or done. The to-do status will assist the user with which tasks to be completed and the 'done' status on the completed tasks will help the user to stay motivated to complete other 'to-do' tasks so the user can change the status to be 'done'. Finally, for the utility, we also use the handling files function in Python. After the user is done editing the list, the

modified list will be stored in a 'txt' file which can also be recalled every time the user runs the code.

**d) Google Colab link: Include the link to your Google Colab  file.**

**Ensure it's well-documented and organized and demonstrates your solution's core functionalities.**

[https://colab.research.google.com/drive/1BzgL2PjLTW1vEfLe8CKOqOnTEpU\\_HW6l?usp=sharing](https://colab.research.google.com/drive/1BzgL2PjLTW1vEfLe8CKOqOnTEpU_HW6l?usp=sharing)

**e) Project Demo Video link:**

<https://www.youtube.com/watch?v=o38fkKjA5Fo>

**f) Group Contribution:**

Name	contribution
109006271 Tuguldur Ts.	Generate the code (40%) + demo video (35%)
111096006 廖盈蓁	Demo video (20%) + video editing + final project paper (50%)
111006237 Shanzelig Putra Wijaya	Generate the code (60%) + demo video (45%) + final project paper (50%)