

**Submitted by – Sahil Koundal**

**Student ID – 100941530**

Problem 1

**Automated Study planner**

**Description**

The idea I would really like to work on is an Automated Study Planner. At this stage, Organization and effective time management when you are a student with so many responsibilities and tasks to handle is such a daunting task in most instances. A planner capable of dispensing tasks depending on due dates as well as study time would be helpful to me. It will consume study tasks as well as time estimates and due dates and output a daily study plan so that work is accomplished on schedule. It will also remind me of upcoming due dates and tasks which they may demand more of my attention.

**Uses**

This solution will be immensely useful for students and anyone who needs to give attention and perform multiple tasks at the same time. One of its benefits is that by creating the study schedule according to the priority of the tasks, it will help to reduce procrastination and thus increase timely organizational skills. It will also be useful to manage time shared in study with other activities such as work, family, etc. These set of issues are widespread among students, working individuals or anyone who requires an individualized task organizing application.

**Difficulty**

The challenge of solving this problem comes in how to accept user input plus how to store and retrieve the data of the task, and how to create an effective schedule, one which honors the time horizon, and the time limits due on the tasks involved. To prioritize tasks, I will have to use organizing structures like loops and conditional ones. The effectiveness of the deadlines will be managed through the datetime module in Python and the task storage through csv module.

**Libraries Used:**

* datetime: To keep track of, and systematically sort out, dates and different time horizons.
* csv: To save and load tasks.
* os: To use it as a test file does exist before writing the data into the file.

**Reflection**

While implementing this project, I realized that development of a study planner application corresponds to my working possibilities. It was not very smooth I had to incorporate various operations such as managing tasks, handling files with CSV, and managing dates. Output was the toughest to handle and input validation was the most important of this project. For instance, I had to ensure that when the user entered due date and estimate time required to complete input was correct and to manage invalid input if any. Correct sequencing and proper representation of time estimates was also a concern to ensure that the tasks were presented just as planned. The most useful source I employed to explore the functions of Python and its csv and datetime modules was official Python documentation. This I used in the application to read and write to CSV files, and to manipulate dates, data which is critical when handling task details and due dates. Furthermore, I used online additional tutorial resources like and forums when faced with particular difficulties that required unique coding solutions. These resources were of great help in enabling me accomplish the project to its completion.

**Problem 2:**

**Document and image converter**

**Description**

The process of file conversion has a number of advantages parenting to different users including students, professionals and casual users. This can prove to be a useful utility to make a workflow faster and more convenient by changing files to more convenient formats such as with use of Document Image Converter.

**Uses**

This tool solves the problem of searching for software for file conversions, offering a convenient and free option for all those who work with different types of files. Most of the time it will be used by students writing their assignments, professionals writing reports or even for individual use when cataloging documents.

**Difficulty**

The problem is somewhat complex. Although libraries facilitate the process, dealing with such peculiarities as an unsupported source format, source not found, or insufficient permissions complicates the process. First and foremost, usability is achieved when one is able to present clean and free from errors output.

**libraries used**

* python-docx: For opening and editing word documents for reading and writing .docx files.
* reportlab: For converting the docs or images into PDF format.
* PyPDF2 or pdfplumber: For converting text from PDF to use in formation of .docx.
* Pillow: For converting images to PDF.
* os: For file path management and existence of files.