Task Management System

Introduction to Programming

CMPT 120L

Maddie&Gaby

Marist College
School of Computer Science and Mathematics

Submitted To:
Dr. Reza Sadeghi
Spring 2024

Project Progress Report 2 of Task Management System

Team Name
Maddie&Gaby

Team Members
1. Gabriela Ramon gabriela.ramon1@marist.edu (Team Head)
2. Madison Chan madison.chan1@marist.edu (Team Member)

Description of Team Members
1. Gabriela Ramon
   I am a freshman with a major in Applied Mathematics with a concentration in Finance and a minor in Data Science and Analytics. I am taking Intro to Programming this semester to build my computing skills while also fulfilling one of major/minor requirements. Maddie and I chose to work together because we have worked well together on various past assignments in class. I chose to be the team head in order to manage our communication and become responsible for all our submissions.

2. Madison Chan
   I am a sophomore with a major in Business Administration with a concentration in Marketing as well as a double minor in Graphic Design and Data Science & Analytics. I am taking Intro to Programming because it is required for my minor. Gaby and I chose to work together because we worked on previous in-class assignments together. Gaby is the team head because she volunteered to be responsible for all future submissions.

Table of Contents

Table of Figures

Project Objective

A task management system (TMS) displays a calendar for the desired week, month, or year. Also, TMS organizes the personal tasks of different users on a specific day. The users should be able to see their individual calendar data & update them. Your TMS will store the data of different user types in distinct comma-separated value (CSV) files. This system should at least support the following items:

1. The admin user is capable of:
    a. Having an admin user and password for login (a string of at least 8 characters)
    b. Changing the admin user and admin password
    c. Adding a normal user to TMS by creating a new username and password. A normal user is not able to define or remove other users.
    d. Remove users from TMS by removing their username, password, and corresponding recorded data.
2. Each user should be able to:
    a. Add a task to TMS. The task contains: a title, time, duration, and description
    b. Remove a task
    c. Edit a task's details
    d. Search through TMS based on time, title, or duration and list the results on the screen. For instance, it should be able to list all scheduled work for one day
3. TMS should be a user-friendly software, such that:
    a. It shows a welcome page and provides a menu of all functions to the user on all pages
    b. It illustrates the reports in a tabular form. For instance, it displays a well-organized calendar of every month or year.
    c. It shows a warning if the user tries to input contact information with a name that exists in the history
    d. TMS should provide an exit function and thank the user for using this software.
4. Optional: TMS should protect the user information, such that:
    a. TMS passwords and the recorded information should be ciphered. In the simplest case, you can use the Caesar cipher methodology. The easiest way to understand the Caesar cipher is to think of cycling the position of letters. In a Caesar cipher with a shift of 4, A becomes D, B becomes E, C becomes F, etc. When reaching the end of the alphabet it cycles around, so X becomes A, Y becomes B, and Z becomes C.

Private GitHub Repository Address

https://github.com/gabyramon/CMPT-120L-112_Task-Management-System_Maddie-Gaby

Graphical User Experience Design

Login Page:

Input: Username and password in text

Output: Valid username and password enable user to access the main page. Invalid username and password will cause an invalid message box to show.

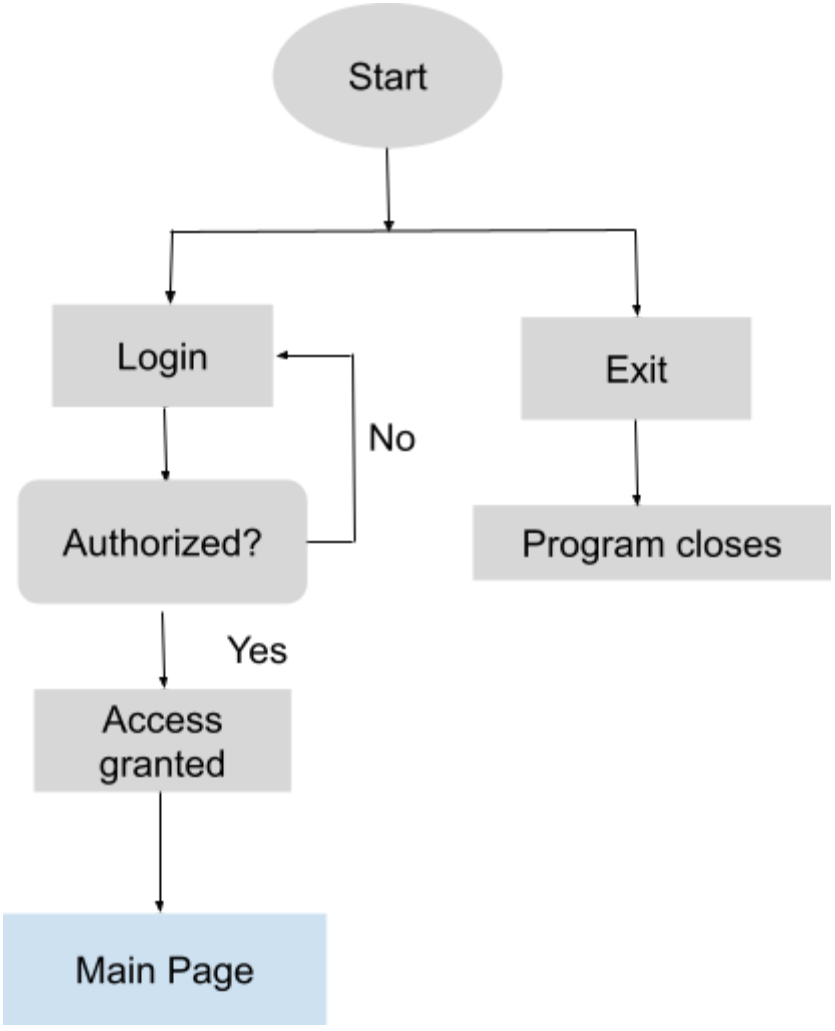

Figure 1: Login Page Flowchart

Main Menu:

Input: User will either select to add/remove/edit/search for a task or select settings or calendar

Output: Their selection will take the user to that page

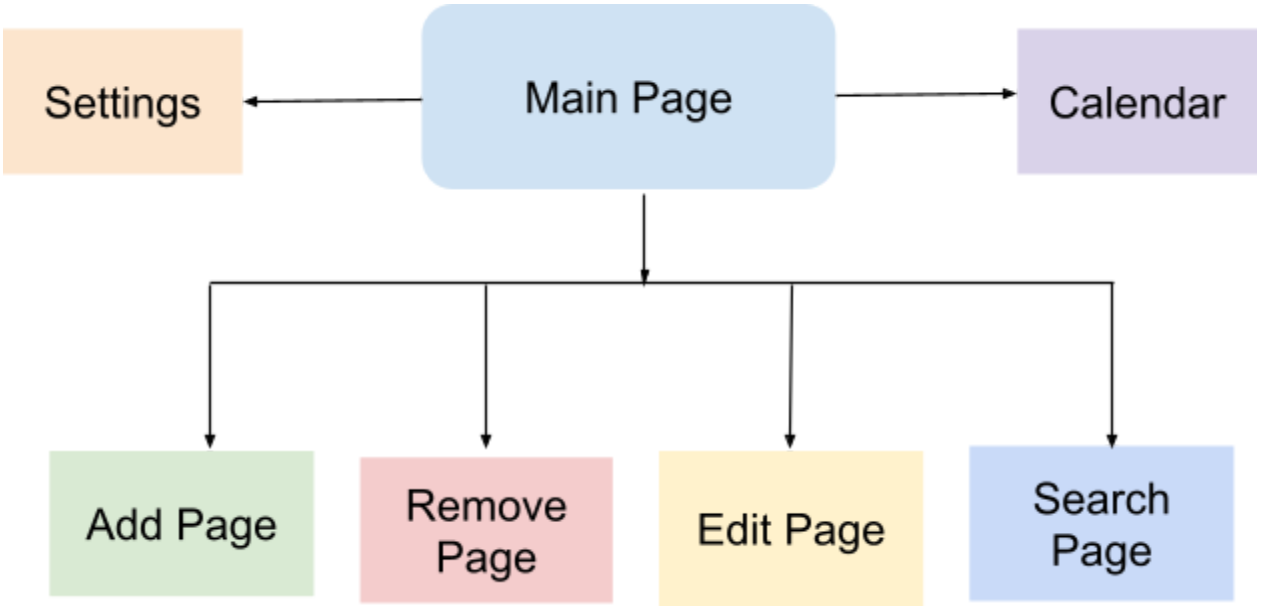

Figure 2: Main Page Flowchart

Add Page:

Input: User will enter a task that contains title, time, duration, and description

Output: Valid input will be saved and a success message will show. Invalid input will prompt the user to reenter the information.

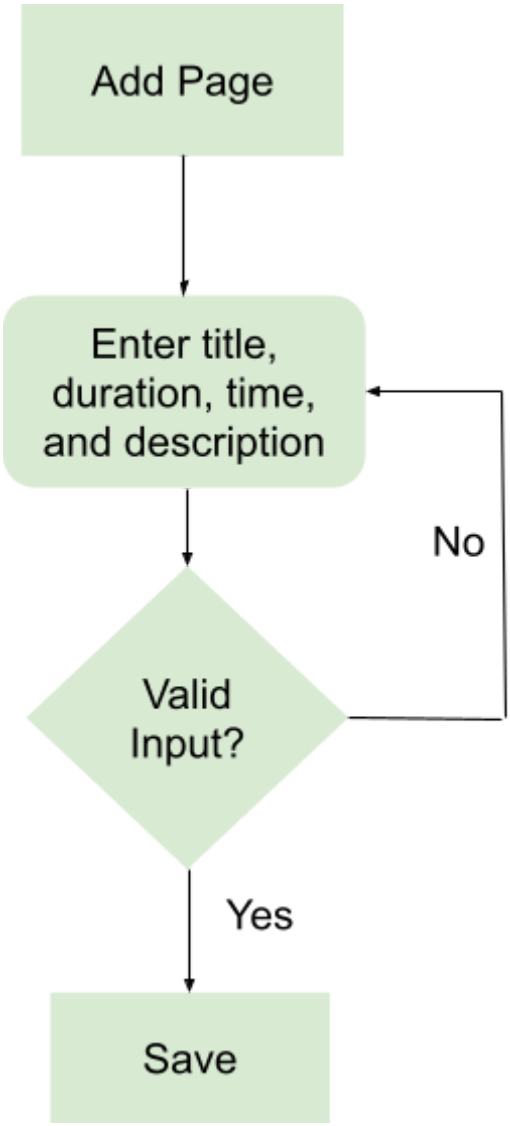

Figure 3: Add Page Flowchart

Remove Page:

Input: User will select the task that they choose to remove

Output: Valid input will be removed from the window' invalid input will show the user an error message and ask the user to choose an existing task.

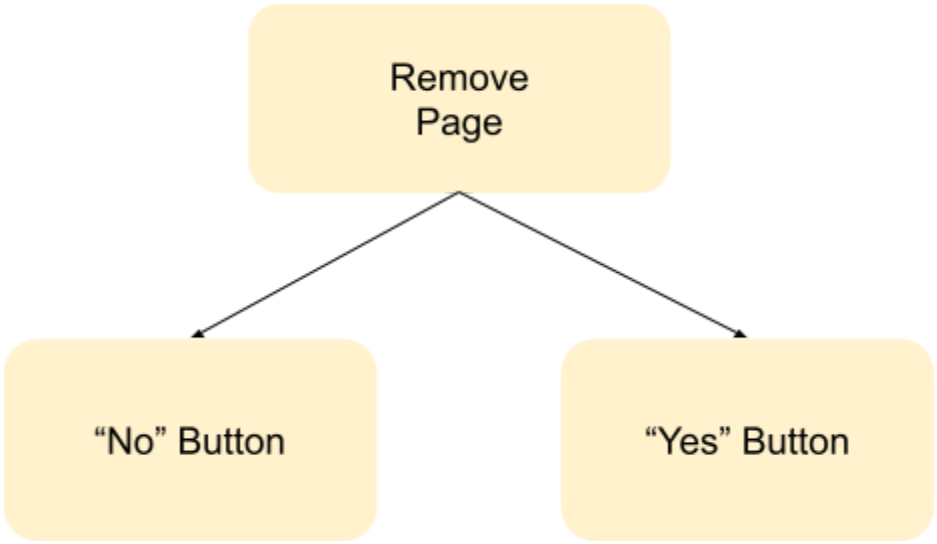

Figure 4: Remove Page Flowchart

Edit Page:

Input: User will select the task that they choose to edit, then select what part of the task they choose to edit (title, time, duration, or description), and then fix the task

Output: Input will be saved and a success message will appear.

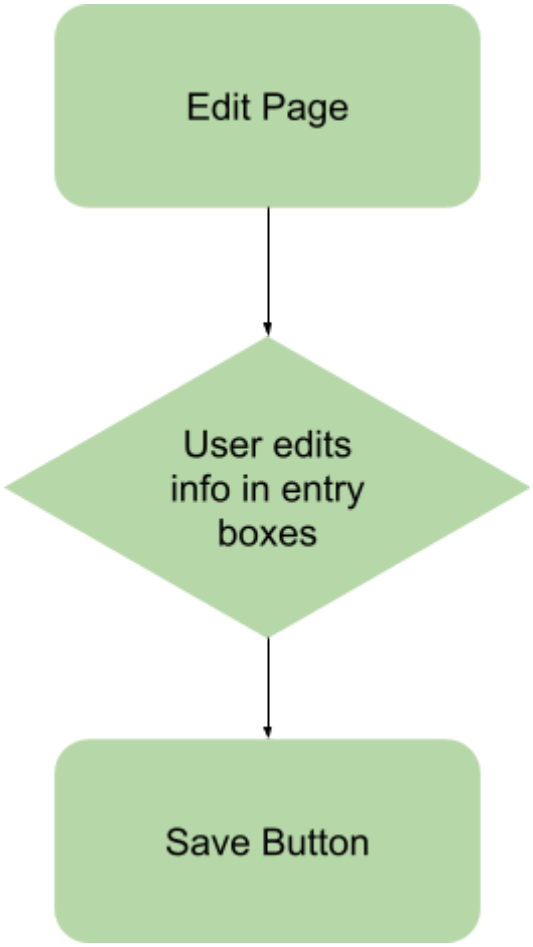

Figure 5: Edit Page Flowchart

Search Page:

Input: User will enter any key words related to the task they are searching for

Output: Program will display all tasks related to the key word(s) that were inputted.

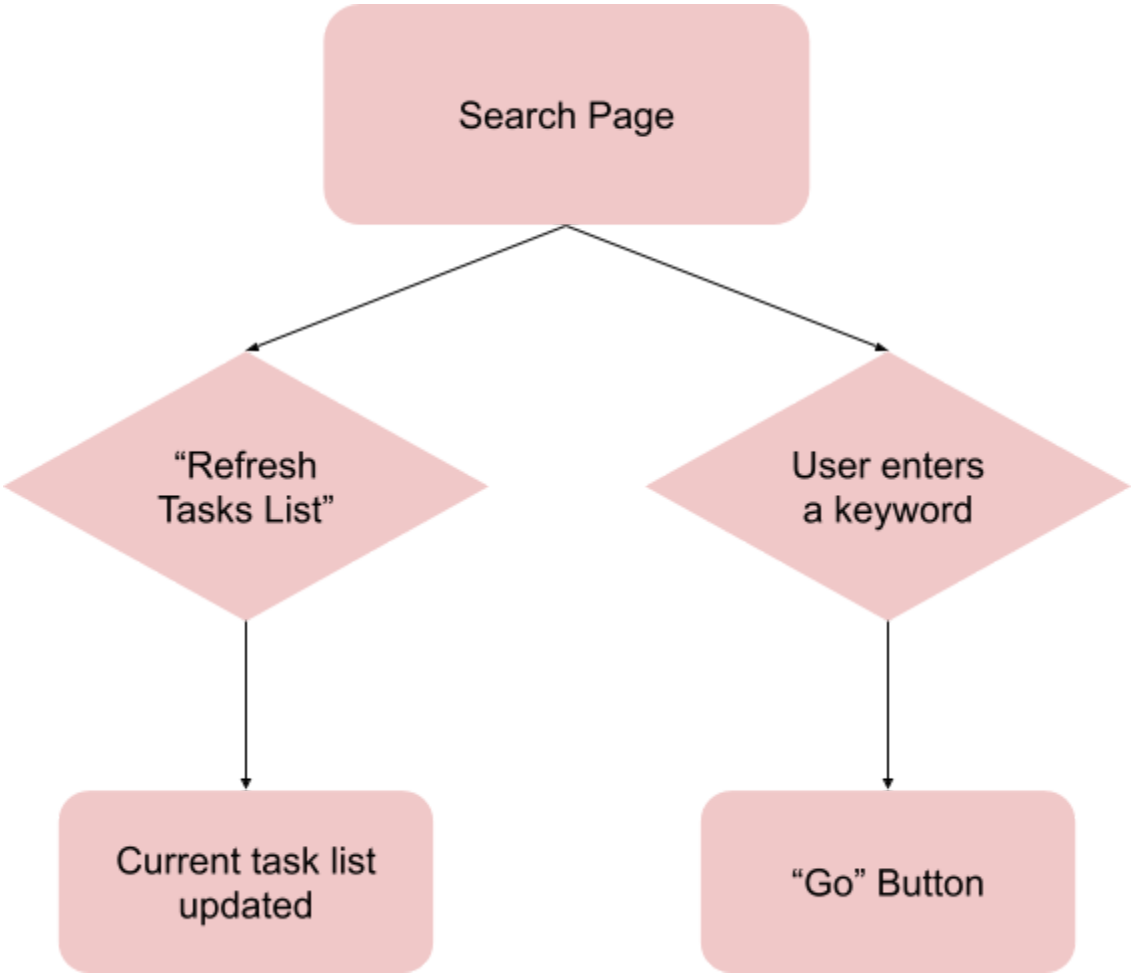

Figure 6: Search Page Flowchart

Settings Page:

Input: User will input the correct username and password. Invalid inputs will display an error message.

Output: Program will display an add, remove, and change button for the user to edit the username and password.

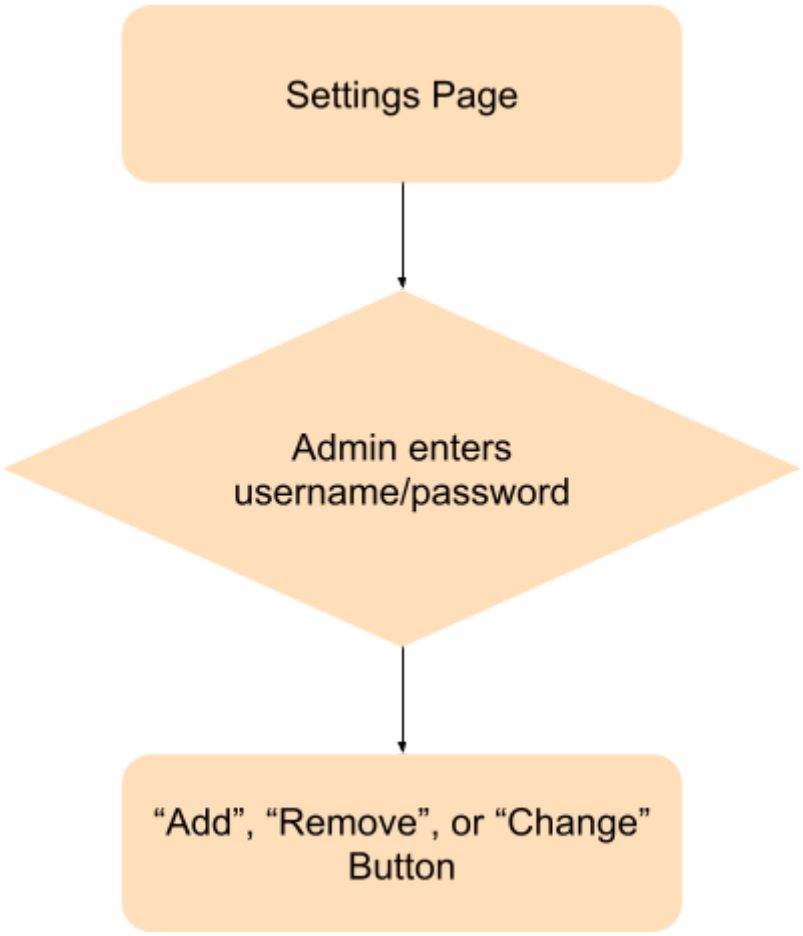

Figure 7: Settings Page Flowchart

Calendar Page:

Input: User will input the month and year that they choose to view.

Output: Main page will be updated with the calendar

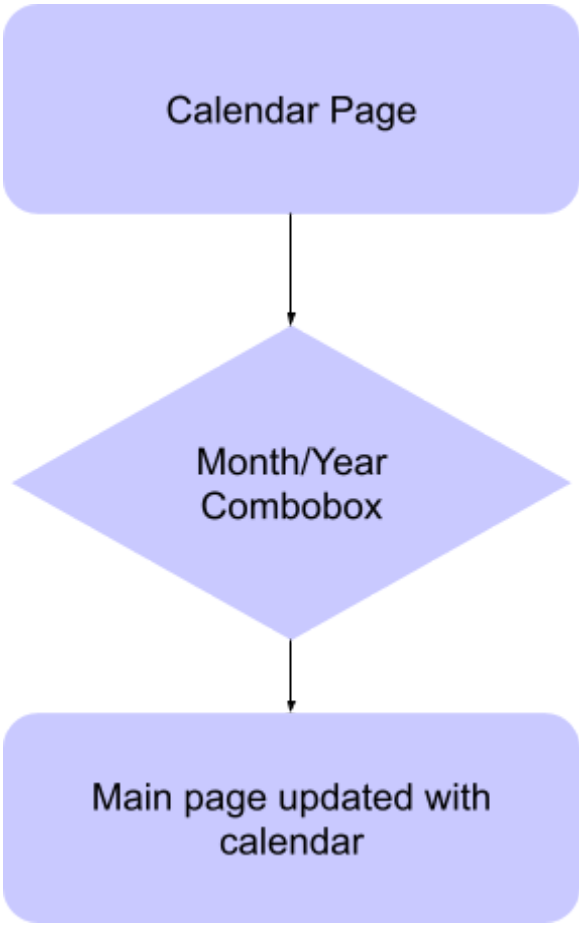

Figure 8: Calendar Page Flowchart