

Task Management System

Introduction to Programming
CMPT 120L



Marist College
School of Computer Science and Mathematics

Submitted To:
Dr. Reza Sadeghi
reza.sadeghi@marist.edu

Fall 2023

Final Project Progress Report Phase #3

Team Name

MC Squared

Team Members

- | | |
|-------------------------|------------------------------------------------|
| 1. Andrew Lombardo | andrew.lombardo1@marist.edu (Team Head) |
| 2. Benjamin Brandt | benjamin.brandt1@marist.edu (Team Member) |
| 3. Nicholas Muratore | nicholas.muratore1@marist.edu (Team Member) |
| 4. Christopher Castillo | christopher.castillo1@marist.edu (Team Member) |

Description of Team Members

1. Andrew Lombardo

My name is Andrew Lombardo, and I am a freshman here at Marist. I have been programming for around nine years, mainly using Java. My current team is the same team I've been with for the rest of the year, so I am comfortable and confident in their abilities. The reason I am a team leader is that I took the initiative first by creating the documentation, GitHub repository, and got the team organized in our objectives.

2. Benjamin Brandt

I'm a freshman at Marist studying Computer Science with a concentration in Software Development. This is the group I've worked with for the last few InClass projects, and I thought we worked well together. We all have attributes that complement each other, and I think we produced good work. I thought it made sense to continue working with the same group for this project since we all have experience working with each other.

3. Nicholas Muratore

I'm a senior at Marist studying Cybersecurity with minors in CS, IS, and IT. I have broad exposure to Python using it for Quantum Computing and Algorithms, BLE device identification using Pyshark, Administration, Web Development, Automation, and now GUI Programming. This is the same group I've been with all year and I'm excited to work with them further on this project.

4. Christopher Castillo

I am a freshman at Marist studying Computer Science with a concentration in Game Design and Programming. My reason for being part of this group is because I have been working with the members for all group projects and I do not see any reason to be in another group. For this project, we selected Andrew since he took the initiative and created the documentation and the GitHub repository while also organizing the team to figure out all our objectives.

Table of Contents

| | |
|------------------------------------------|-------|
| 1. GitHub Repository..... | 6 |
| 2. Project Objective..... | 7 |
| 3. Graphical User Experience Design..... | 9-15 |
| 4. Graphical User Interface Design..... | 16-48 |
| 5. Virtual Envirotment..... | 49 |
| 6. Libraries..... | 50 |
| 7. References..... | 51 |

Table of Figures

| | |
|----------------------------------------------------------------------|---------|
| 1. Figure 1: Login Page Flowchart..... | 8 |
| 2. Figure 2: Main Page Flowchart..... | 9 |
| 3. Figure 3: Add Task Flowchart..... | 10 |
| 4. Figure 4: Edit Task Flowchart..... | 11 |
| 5. Figure 5: Remove Task Flowchart..... | 12 |
| 6. Figure 6: Search Task Flowchart..... | 13 |
| 7. Figure 7: Calendar Page Flowchart..... | 14 |
| 8. Figure 8: Navigation Bar Flowchart..... | 15 |
| 9. Figure 9: Login Layout Drawing..... | 16 |
| 10. Figure 10: Login Layout | 16 |
| 11. Figure 11: User Layout Drawing..... | 18 & 24 |
| 12. Figure 12: User Page Layout..... | 19 |
| 13. Figure 13: Admin User Navigation Bar Layout Drawing..... | 22 |
| 14. Figure 14: Admin User Navigation Bar Layout..... | 22 |
| 15. Figure 15: Admin User's Task Management Page Layout..... | 25 |
| 16. Figure 16: Admin User's User Management Page Layout Drawing..... | 28 |

| | |
|-----------------------------------------------------------------------|----|
| 17. Figure 17: Admin User's User Management Page Layout..... | 28 |
| 18. Figure 18: Admin User's Admin Management Page Layout Drawing..... | 30 |
| 19. Figure 19: Admin User's Admin Management Page Layout..... | 30 |
| 20. Figure 20: Calendar Page Layout Drawing..... | 32 |
| 21. Figure 21: Calendar Page Layout..... | 33 |
| 22. Figure 22: Management System Layout..... | 37 |
| 23. Figure 23: Virtual Environment Video..... | 50 |

GitHub Repository Address [1]

https://github.com/Levalvus/Final_Project--Task_Manager--MCSquared/tree/main

Project Objective [2]

Project Title: Task Management System

Summary: 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 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 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 a 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 the letters. In a Caesar cipher with a shift of 3, 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.

Graphical User Experience Design [3]

Login Page

Abstract: The program is run and the login page pops up. On this page, the user or admin will log on to our Task Management System. When credentials are entered, the program will check to see if the credentials are a match. If not, then the user will have to re-enter credentials. Otherwise, they will be granted access. If admin credentials are entered it will bring them to the admin page. If user credentials are entered it will bring them to the normal page. Lastly, the exit option will close the program.

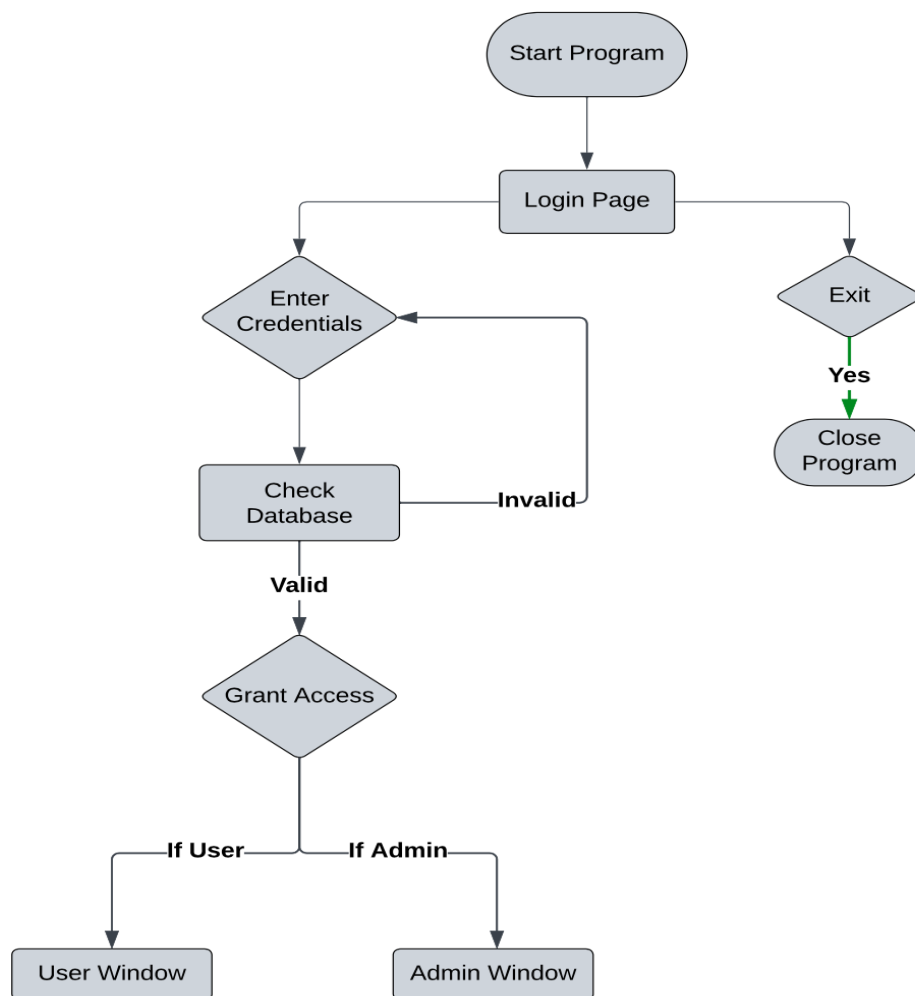


Figure 1: Login Page Flowchart

Graphical User Experience Design

Main User Page

Abstract: After logging in, the main window pops up. In this window, there is an add task which the user can enter attributes for the task and save it to our system. An edit task, where the user edits an existing task and saves to our system. A remove task, where the user can select a task created and remove it. A search task, where the user can search a task by name. Then, our admin page has the option to go to the task page, user management, or the admin management. The privileges from that consist of adding a user and by creating its credentials, editing a user allowing the admin to edit the credentials, removing the user by username and deleting its data. Lastly, the logout button to go back to the login page and the exit option which will close the program.

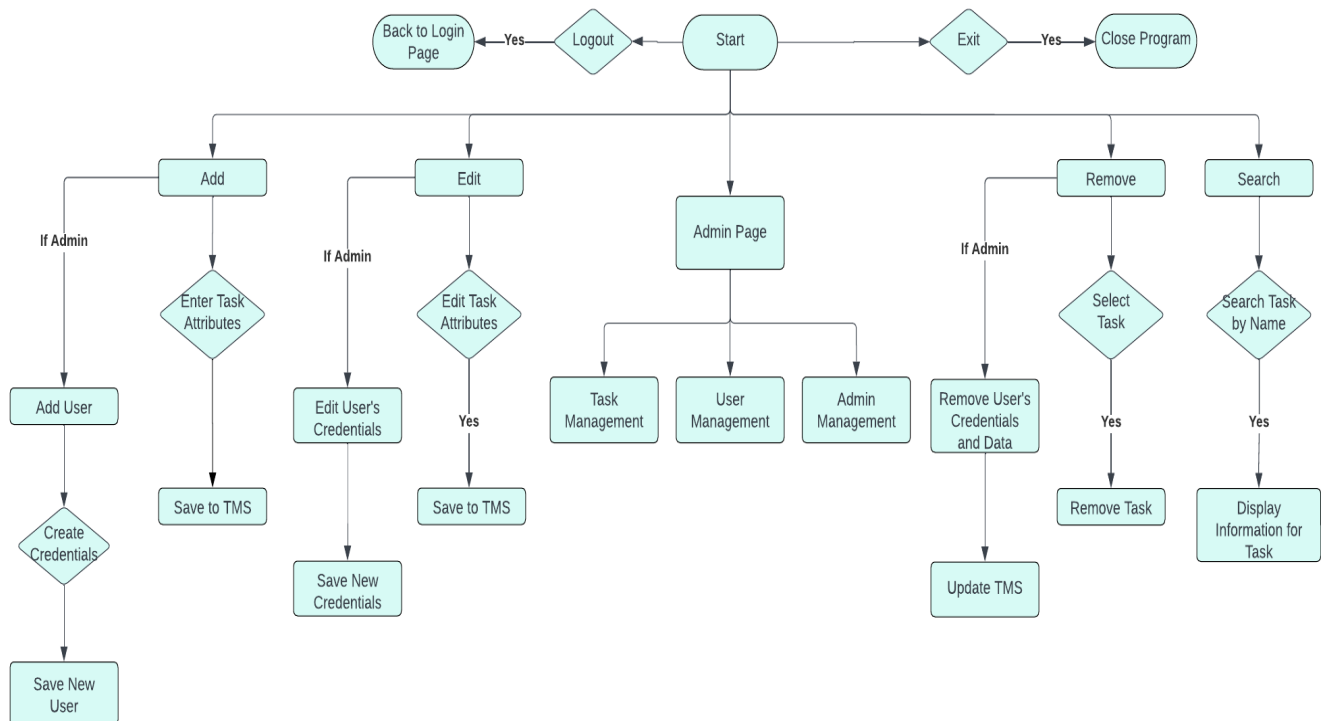


Figure 2: Main Page Flowchart

Graphical User Experience Design

Add Task

Abstract: The add feature allows the user/admin to add a task of preference. Users will be asked to include its time, title, duration, and a general description for it to qualify as a valid task. If any of the features are not valid, the user will need to re-enter the item. If it is a valid entry, the task will be saved to the task management system. Once again, the user is also allowed to exit the page when they are first asked to add a task, which would ultimately close the program.

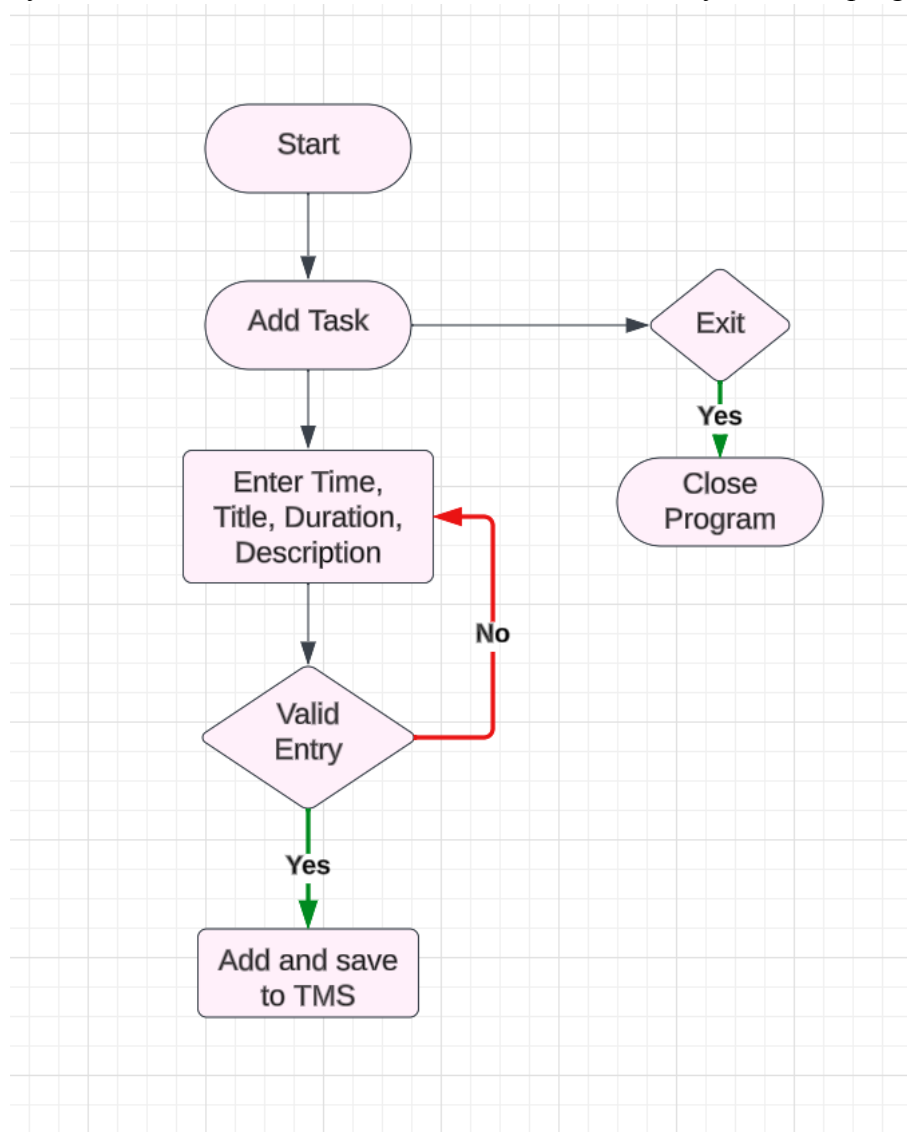


Figure 3: Add Task Flowchart

Graphical User Experience Design

Edit Task

Abstract: The edit feature allows the user to search for any task that is a part of their schedule and change specific information about it. Using the exact title for the task, they can then edit the task's date, start time, description, and duration. The user can also simply leave this option by exiting and closing the program.

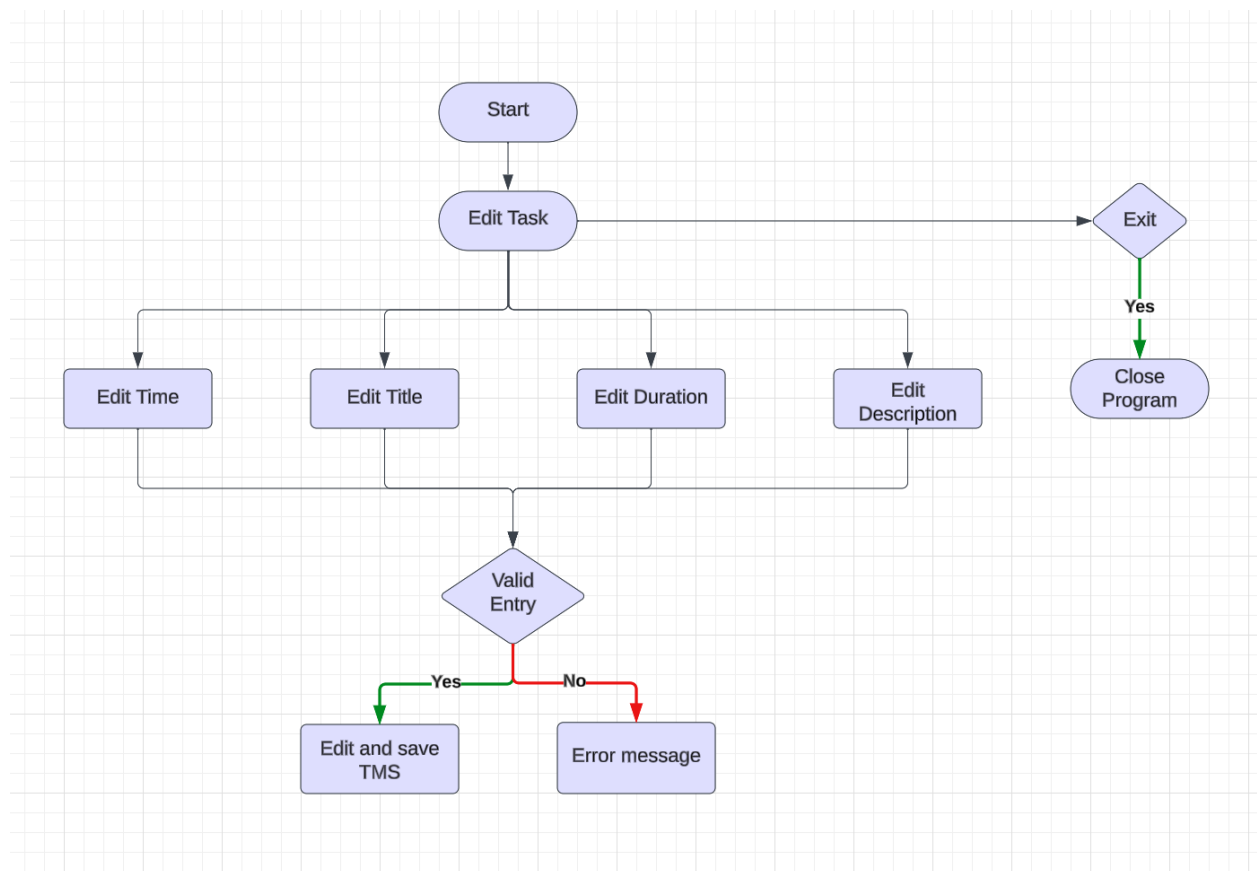


Figure 4: Edit Task Flowchart

Graphical User Experience Design

Remove Task

Abstract: The remove feature allows the user to discard a task they do not need or no longer want. The user is faced with the option to remove the task, and then asked to confirm this decision or not. If the user says no, it circles back to which task the user would like to remove. If the user confirms their decision to remove the task, it will be deleted from the task management system. The user can also simply leave this option by exiting and closing the program.

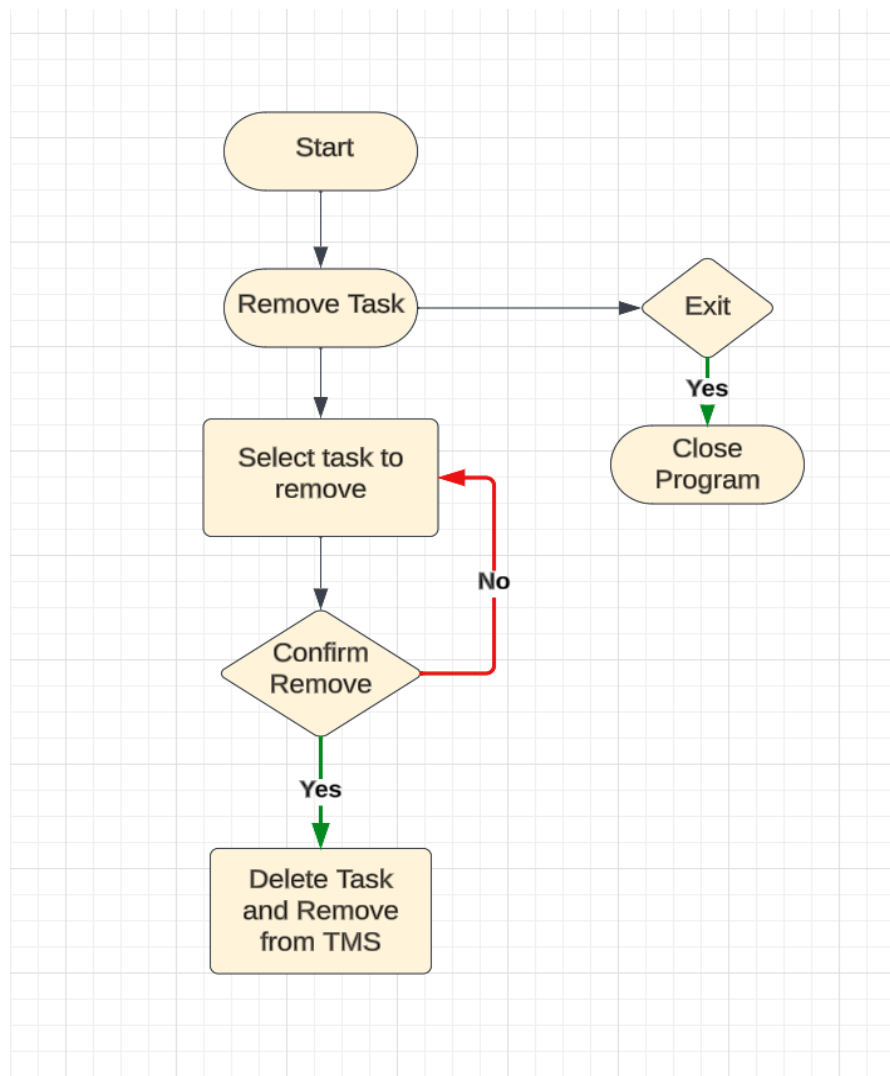


Figure 5: Remove Task Flowchart

Graphical User Experience Design

Search Task

Abstract: The search feature allows the user to search for any task that is a part of their individual schedule. The user can search through all tasks with a given title, duration or start time and will be offered a list of all tasks that match those requirements. The user can also simply leave this option by exiting and closing the program.

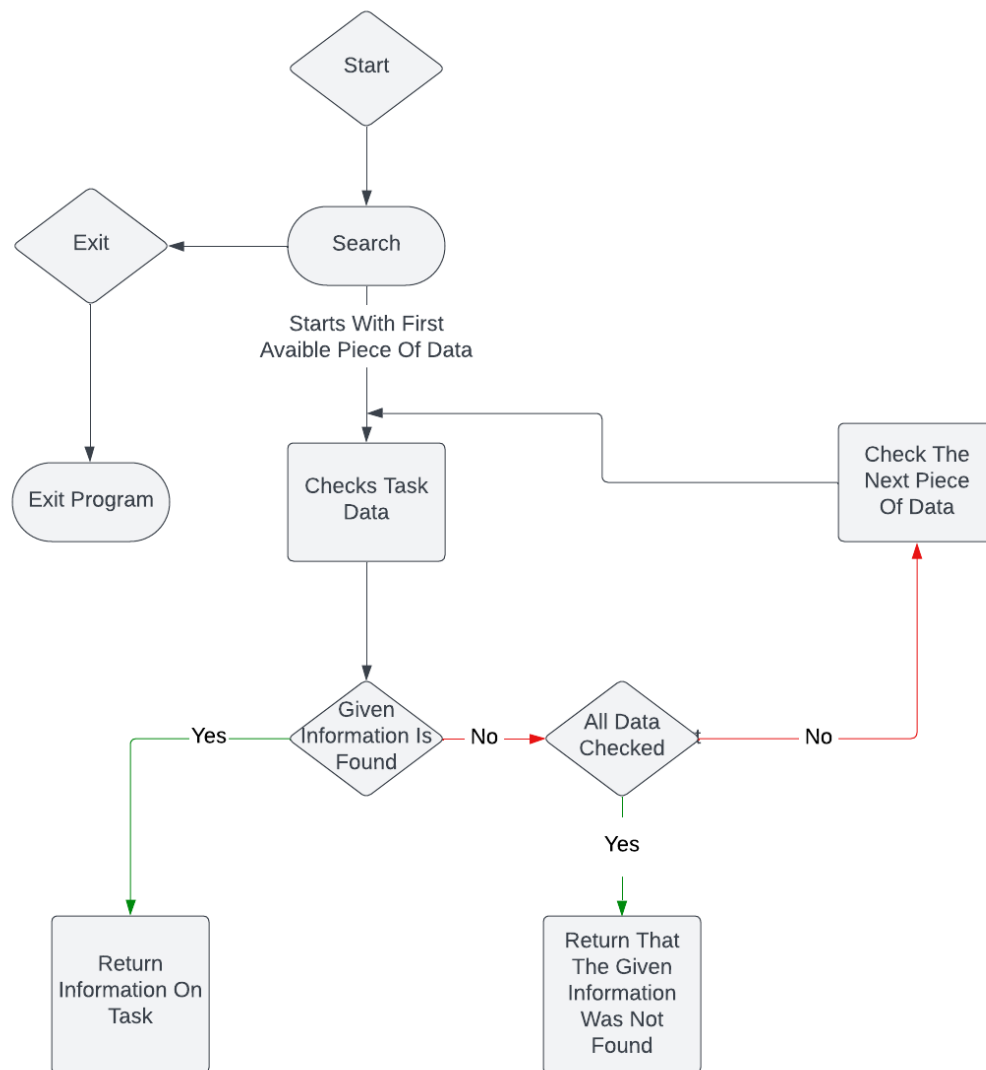


Figure 6: Search Task Flowchart

Graphical User Experience Design

Calendar

Abstract: The calendar is a unique window that takes all tasks from each user's schedule and displays their calendar. The amount of time the user can see is determined by first entering a starting date, followed by entering if they want to see their task for the upcoming week, month, or year. The user can also simply leave this option either closing the program or exiting the page back to the user's respective page.

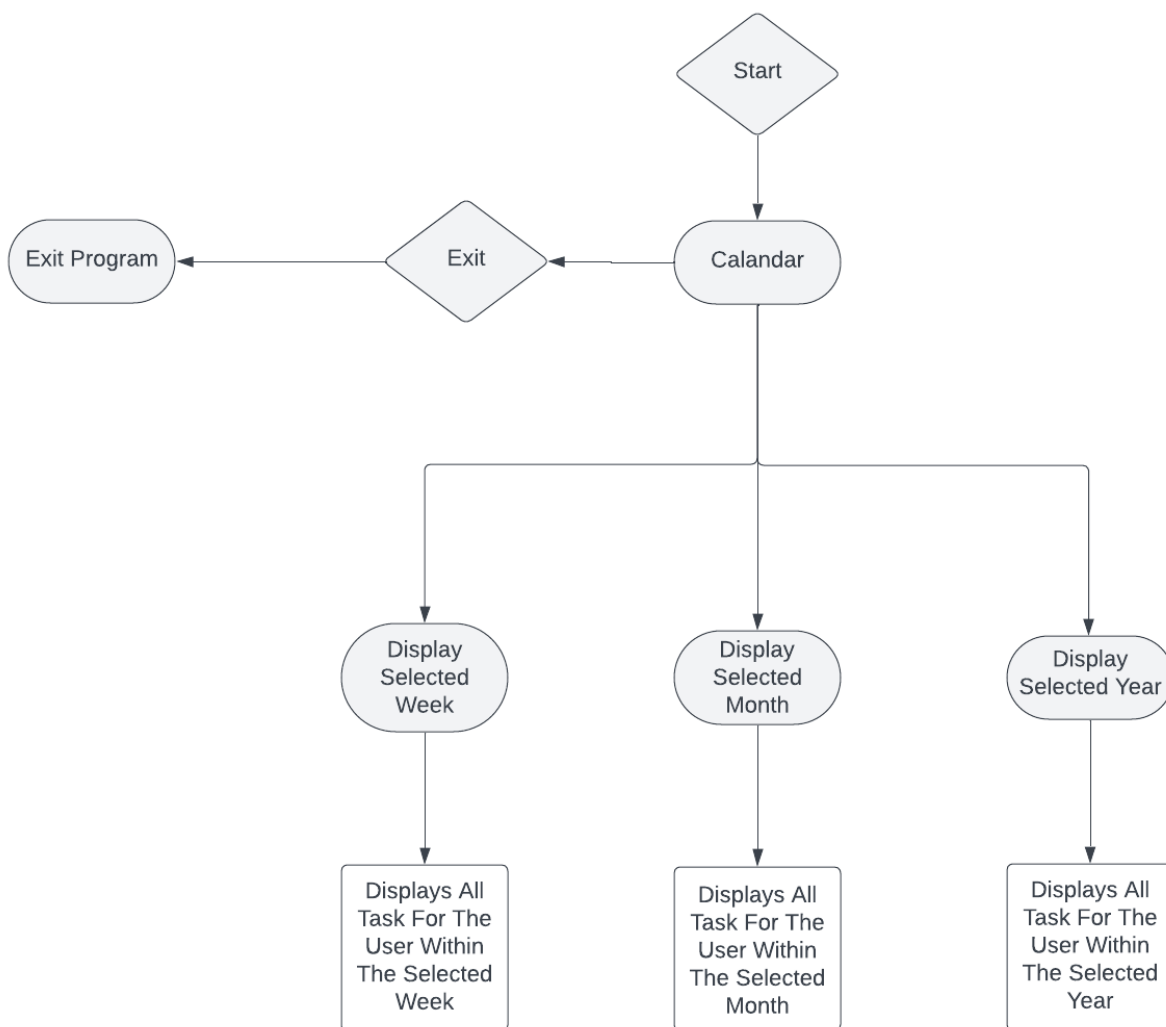


Figure 7: Calendar Flowchart

Graphical User Interface Design

Navigation Bar

Abstract: The navigation bar gives the admin user access to each of the three actions they can complete, user management, admin management, and task management. Clicking on each of the buttons will allow the user to enter a screen to complete one of the tasks. All of the tasks have been previously defined, hence the yellow note that they all connect to. Finally, the user is allowed to log out of the system at the bottom right of the interface.

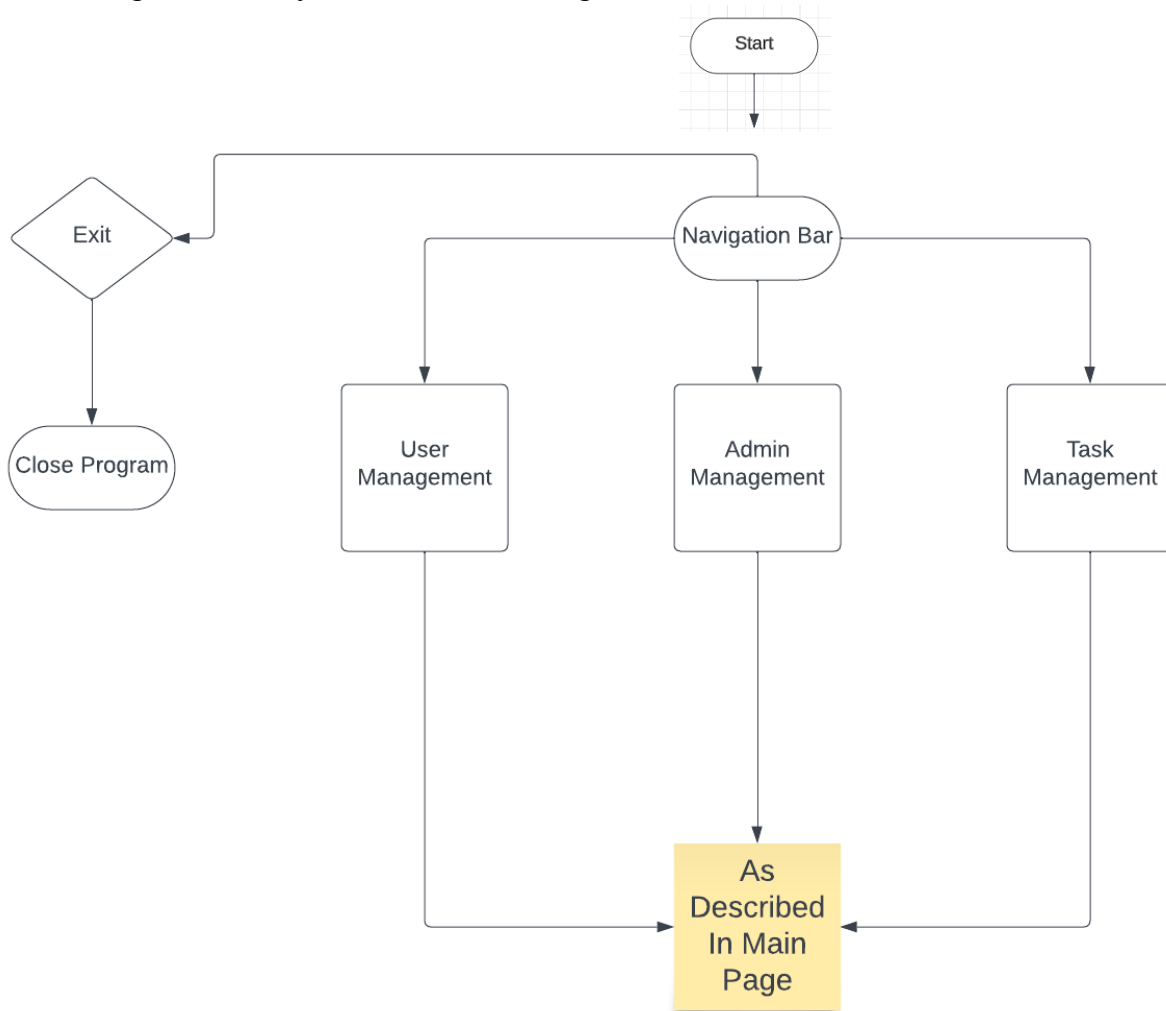


Figure 8: Navigation Bar Flowchart

Graphical User Interface Design [4]

Login Page

Abstract: The login page contains a banner with the name of our project Task Management System. It has two images on each side resembling tasks and management. The word Login is displayed big at the top and following it there's the username label and its entry bar then the password label and its entry bar. The submit button for entering the credentials and then the exit button in the corner to quit.

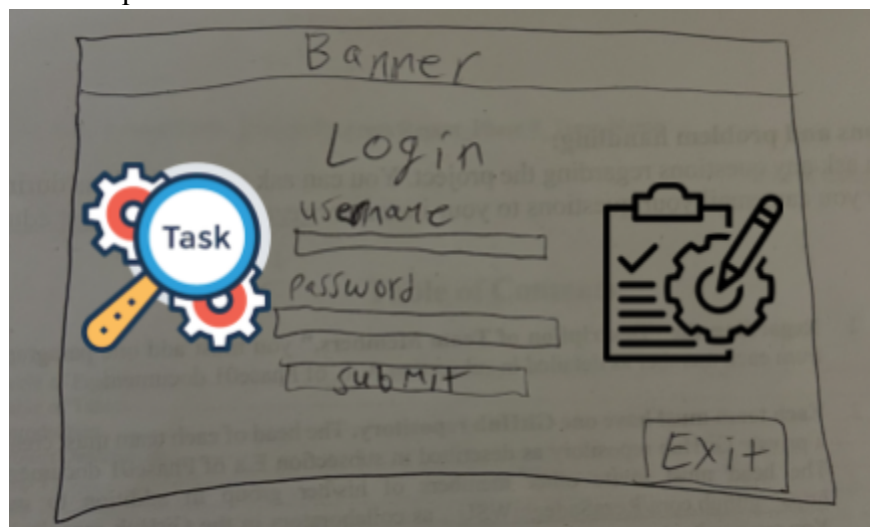


Figure 9: Login Page Layout Drawing

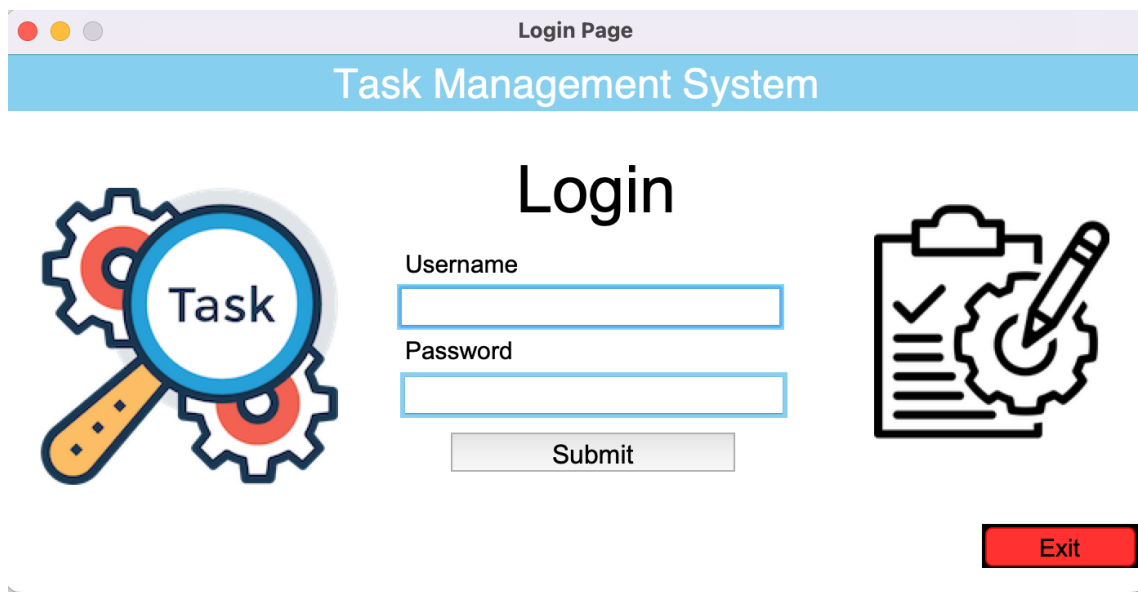


Figure 10: Login Page Layout

Graphical User Interface Design

Login Page Code

```
#Login Page - TMS
import tkinter as tk
import tkmacosx as tkm
from PIL import ImageTk, Image
window=tk.Tk()
window.title("Login Page")
window.geometry('700x330')
window.resizable(False,False)
window.configure(bg='white')

task_image = ImageTk.PhotoImage(file="/Users/nickmuratore/Downloads/TASK.png")
clipboard_image = ImageTk.PhotoImage(file="/Users/nickmuratore/Downloads/clipboard.png")

task_label = tk.Label(window, image=task_image, bg='white')
task_label.place(x=20, y=80)
clip_label = tk.Label(window, image=clipboard_image, bg='white')
clip_label.place(x=530, y=90)

banner = tk.Label(window, text="Task Management System", bg='#89CFF0', fg='White', font=("Helvetica", 25))
banner.pack(side=tk.TOP, fill=tk.X)

label = tk.Label(text='Login',bg='white',font=('Arial',40))
label.place(x=310, y=55)

userEnt = tk.Entry(window,width = 25,bg='white')
userEnt.configure(highlightbackground='#89CFF0')
userEnt.place(x=240, y=141)

userLabel = tk.Label(text='Username',bg='white',font=('Arial',15))
userLabel.place(x=242,y=116)

passEnt = tk.Entry(window,width = 25,bg='white')
passEnt.configure(highlightbackground='#89CFF0')
passEnt.place(x=242, y=195)

passLabel = tk.Label(text='Password',bg='white',font=('Arial',15))
passLabel.place(x=242,y=169)

submitButton = tk.Button(text='Submit',
                           font=('Arial',16),
                           bg='white',
                           width=16)
submitButton.place(x=270, y=229)
submitButton.configure(highlightbackground='white')

ExitButton = tkm.Button(text='Exit',font=('Arial',15),bg='#FF3131',command=window.destroy)
ExitButton.place(x=598,y=288)
ExitButton.configure(highlightbackground='black')

window.mainloop()
```

Graphical User Interface Design

User Page

Abstract: The user page of the task management system consists of the title, description, date (month, date year format), duration, and the start time for the given task, which are all shown on the left side of the interface. On the right side, the user has access to the add, edit, search and remove commands, and is also able to access a calendar to view their tasks. Finally, the user is allowed to log out of the system in the bottom right of the interface.

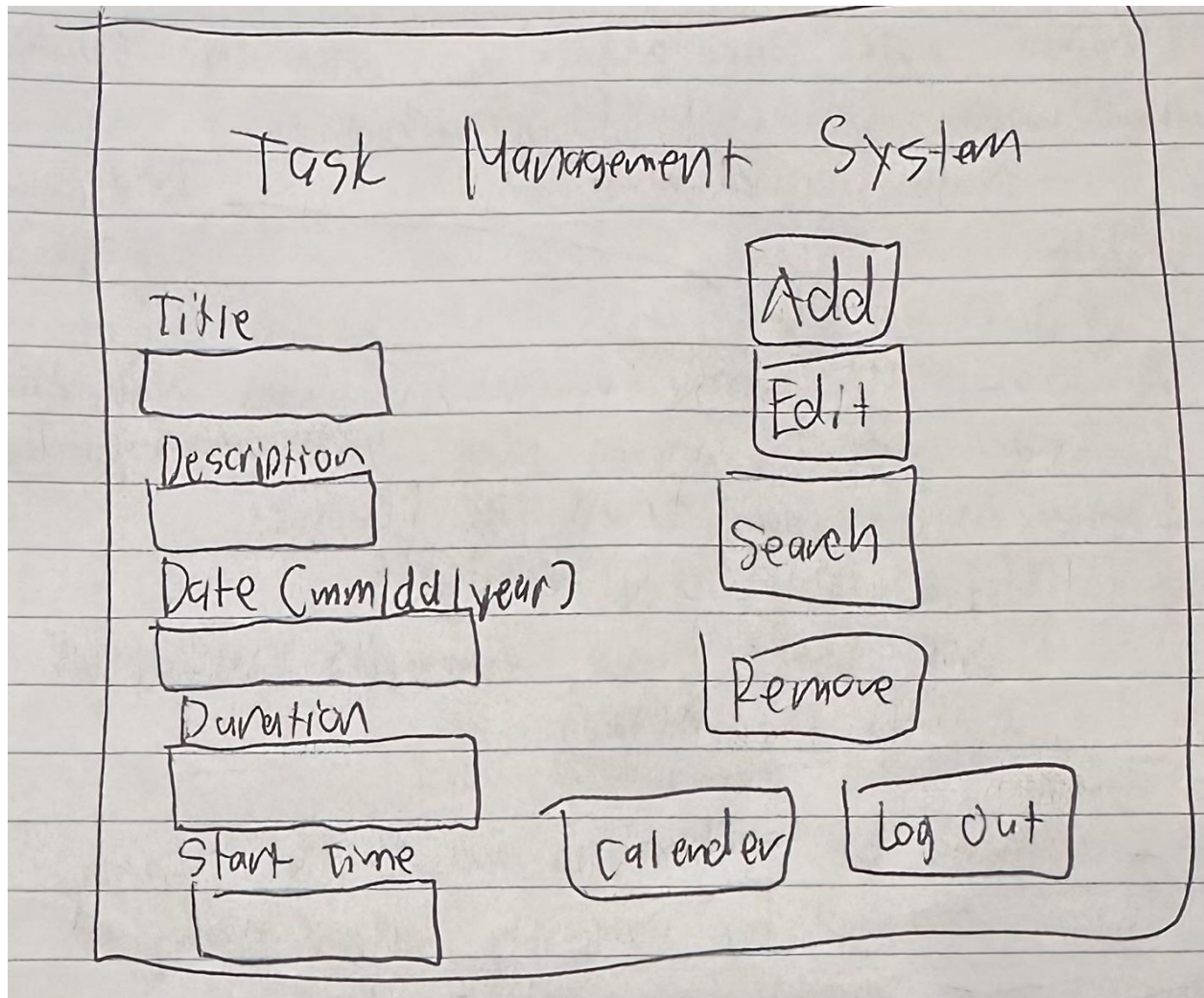


Figure 11: User Page Layout Drawing

Task Management

| | | |
|----------------------------------------|----------------------|--------------------------------------------------------------------|
| Title | <input type="text"/> | <div>Add</div> <div>Edit</div> <div>Search</div> <div>Remove</div> |
| Description | <input type="text"/> | |
| Date (mm/dd/year) | <input type="text"/> | |
| Duration | <input type="text"/> | |
| Start Time (Military) | <input type="text"/> | |
| <div>Calender</div> <div>Log Out</div> | | |

Figure 12: User Page Layout

Graphical User Interface Design

User Page Code

```

import tkinter as tk
import tkmacosx as tkm
window=tk.Tk()
window.title("User Task Management")
window.geometry('400x450')
window.resizable(False,False)
window.configure(bg="#D58A94")

banner = tk.Label(window, text="Task Management", bg='#D58A94', fg='white', font=("Helvetica", 25))
banner.pack(side=tk.TOP, fill=tk.X)

titleEnt = tk.Entry(window,width = 25)
titleEnt.place(x=10, y=100)

titleLabel = tk.Label(text='Title',font=('Arial',15))
titleLabel.place(x=10,y=70)

desEnt = tk.Entry(window,width = 25)
desEnt.place(x = 10, y=160)

desLabel = tk.Label(text='Description',font=('Arial',15))
desLabel.place(x=10,y=130)

dateEnt = tk.Entry(window,width = 25)
dateEnt.place(x=10, y=220)

dateLabel = tk.Label(text='Date (mm/dd/year)',font=('Arial',15))
dateLabel.place(x=10,y=190)

durEnt = tk.Entry(window,width = 25)
durEnt.place(x = 10, y=280)

durLabel = tk.Label(text='Duration',font=('Arial',15))
durLabel.place(x=10,y=250)

startLabel = tk.Label(text='Start Time (Military)',font=('Arial',15))
startLabel.place(x=10,y=310)

startEnt = tk.Entry(window,width = 25)
startEnt.place(x = 10, y=340)

addButton = tk.Button(text='Add',
                      font=('Arial',12),
                      width=16)
addButton.place(x=230, y=120)

```

```
editButton = tk.Button(text='Edit',
                        font=('Arial',12),
                        width=16)
editButton.place(x=230, y=170)

searchButton = tk.Button(text='Search',
                          font=('Arial',12),
                          width=16)
searchButton.place(x=230, y=220)

removeButton = tk.Button(text='Remove',
                          font=('Arial',12),
                          width=16)
removeButton.place(x=230, y=270)

ExitButton = tk.Button(text='Log Out',font=('Arial',15),bg='red',command=window.destroy)
ExitButton.place(x=225,y=390)

CalenderButton = tk.Button(text='Calender',font=('Arial',15),bg='blue',command=window.destroy)
CalenderButton.place(x=50,y=390)

window.mainloop()
```

Graphical User Interface Design

Admin User Navigation Bar Page

Abstract: The admin user navigation bar page consists of everything from the normal user page interface, but with a few more options. The admin has access to the user management, where someone can add a task's title, description, date (month, day, year format), duration, and start time, as well as all the user commands of add, edit, search, and remove. The admin user can also access the admin management section, where they can change their user and password (must be at least 8 characters), remove users from the task management system by removing their username and password, and add new users by creating new usernames and passwords.

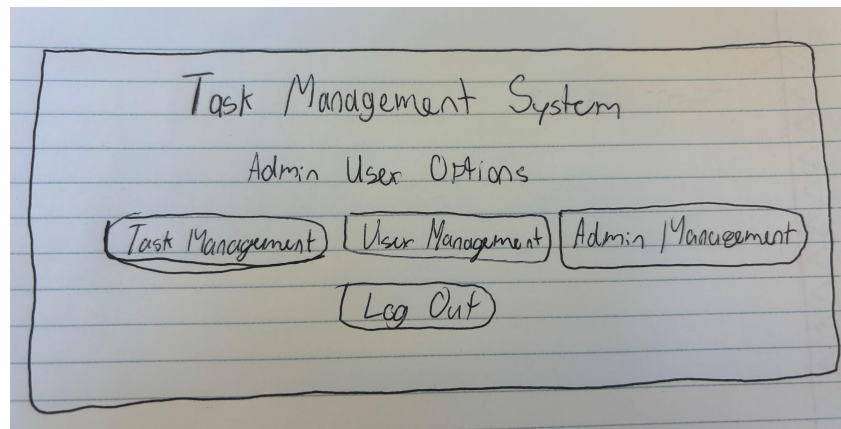


Figure 13: Admin User Navigation Bar Layout Drawing

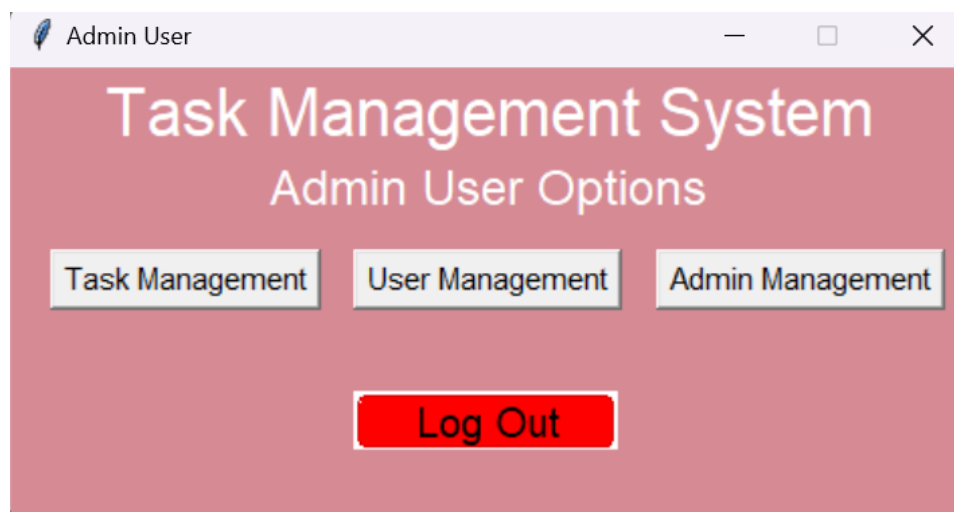


Figure 14: Admin User Navigation Bar Page

Graphical User Interface Design

Admin User Navigation Bar Code

```
#Admin User Nav Bar - TMS
import tkinter as tk
import tkmacosx as tkm
window=tk.Tk()
window.title("Admin User")
window.geometry('475x225')
window.resizable(False,False)
window.configure(bg="#D58A94")

banner = tk.Label(window, text="Task Management System", bg='#D58A94', fg='white',
font=("Helvetica", 25))
banner.pack(side=tk.TOP, fill=tk.X)

sub_banner = tk.Label(window, text="Admin User Options", bg='#D58A94', fg='white',
font=("Helvetica", 18))
sub_banner.pack(side=tk.TOP, fill=tk.X)

Task_Btn = tk.Button(text='Task Management',font=('Arial',11))
Task_Btn.place(x=20, y=90)

User_Btn = tk.Button(text='User Management',font=('Arial',11))
User_Btn.place(x=170, y=90)

Admin_Btn = tk.Button(text='Admin Management',font=('Arial',11))
Admin_Btn.place(x=320, y=90)

ExitButton = tkm.Button(text='Log Out',font=('Arial',15),bg='red',command=window.destroy)
ExitButton.place(x=170,y=160)

window.mainloop()
```


Graphical User Interface Design

Admin User's Task Management Page

Abstract: The user page of the task management system consists of the title, description, date (month, date year format), duration, and the start time for the given task, which are all shown on the left side of the interface. On the right side, the user has access to the add, edit, search, and remove commands, and is also able to access a calendar to view their tasks. Finally, the user is allowed to log out of the system at the bottom right of the interface.

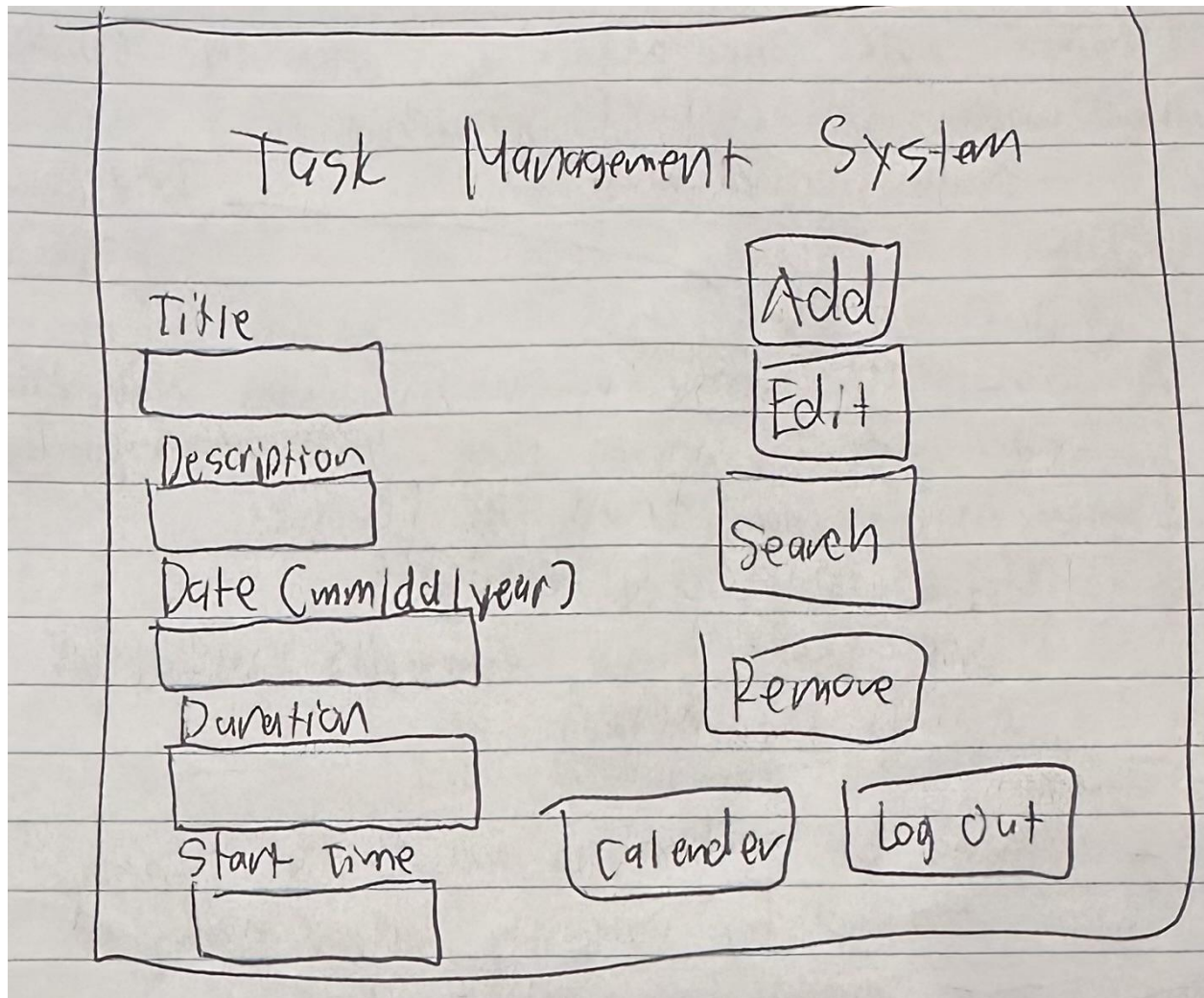


Figure 11: Admin User's Task Management Layout Drawing

Task Management

| | | |
|-----------------------|--|---------|
| Title | | Add |
| Description | | |
| Date (mm/dd/year) | | |
| Duration | | |
| Start Time (Military) | | |
| | | Edit |
| | | Search |
| | | Remove |
| Calender | | Log Out |

Figure 15: Admin User's Task Management Page Layout

Graphical User Interface Design

Admin User's Task Management Page Code

```
import tkinter as tk
import tkmacosx as tkm
window=tk.Tk()
window.title("Admin User Task Management")
window.geometry('400x450')
window.resizable(False,False)
window.configure(bg="#D58A94")

banner = tk.Label(window, text="Task Management", bg='#D58A94', fg='white',
font=("Helvetica", 25))
banner.pack(side=tk.TOP, fill=tk.X)

titleEnt = tk.Entry(window,width = 25)
titleEnt.place(x=10, y=100)

titleLabel = tk.Label(text='Title',font=('Arial',15))
titleLabel.place(x=10,y=70)

desEnt = tk.Entry(window,width = 25)
desEnt.place(x = 10, y=160)

desLabel = tk.Label(text='Description',font=('Arial',15))
desLabel.place(x=10,y=130)

dateEnt = tk.Entry(window,width = 25)
dateEnt.place(x=10, y=220)

dateLabel = tk.Label(text='Date (mm/dd/year)',font=('Arial',15))
dateLabel.place(x=10,y=190)

durEnt = tk.Entry(window,width = 25)
durEnt.place(x = 10, y=280)

durLabel = tk.Label(text='Duration',font=('Arial',15))
durLabel.place(x=10,y=250)

startLabel = tk.Label(text='Start Time (Military)',font=('Arial',15))
startLabel.place(x=10,y=310)
```

```
startEnt = tk.Entry(window,width = 25)
startEnt.place(x = 10, y=340)

addButton = tk.Button(text='Add',
                      font=('Arial',12),
                      width=16)
addButton.place(x=230, y=120)

editButton = tk.Button(text='Edit',
                      font=('Arial',12),
                      width=16)
editButton.place(x=230, y=170)

searchButton = tk.Button(text='Search',
                        font=('Arial',12),
                        width=16)
searchButton.place(x=230, y=220)

removeButton = tk.Button(text='Remove',
                        font=('Arial',12),
                        width=16)
removeButton.place(x=230, y=270)

ExitButton = tk.Button(text='Log Out',font=('Arial',15),bg='red',command=window.destroy)
ExitButton.place(x=225,y=390)

CalenderButton =
tk.Button(text='Calender',font=('Arial',15),bg='blue',command=window.destroy)
CalenderButton.place(x=50,y=390)

window.mainloop()
```

Graphical User Interface Design

Admin User's User Management Page

Abstract: The admin's user management page is where the admin can access their specific abilities that a normal user cannot. On this screen, a manager can enter a username, and use it to either change their given password. They can also remove the given username, password, and all task information associated with that user.

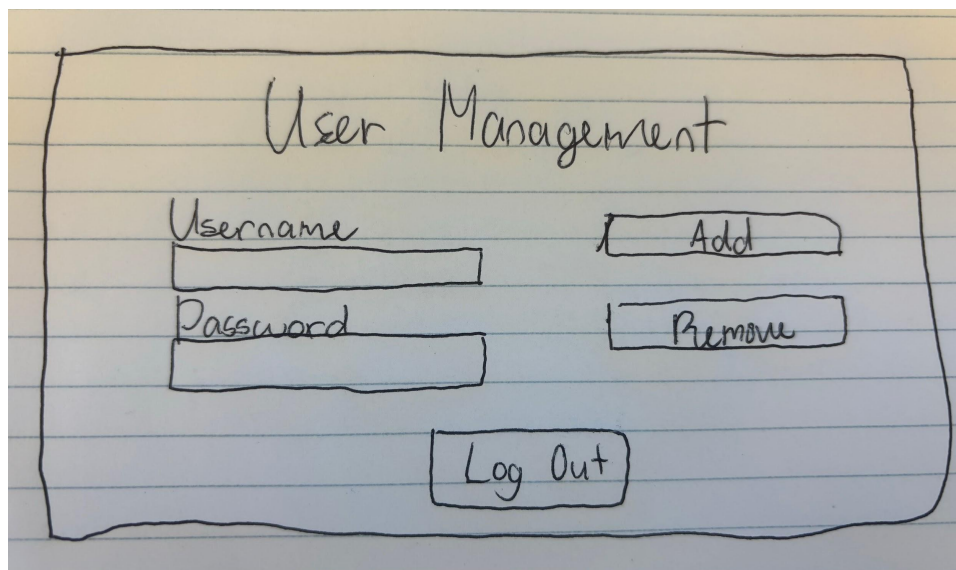


Figure 16: Admin User's User Management Page Layout Drawing

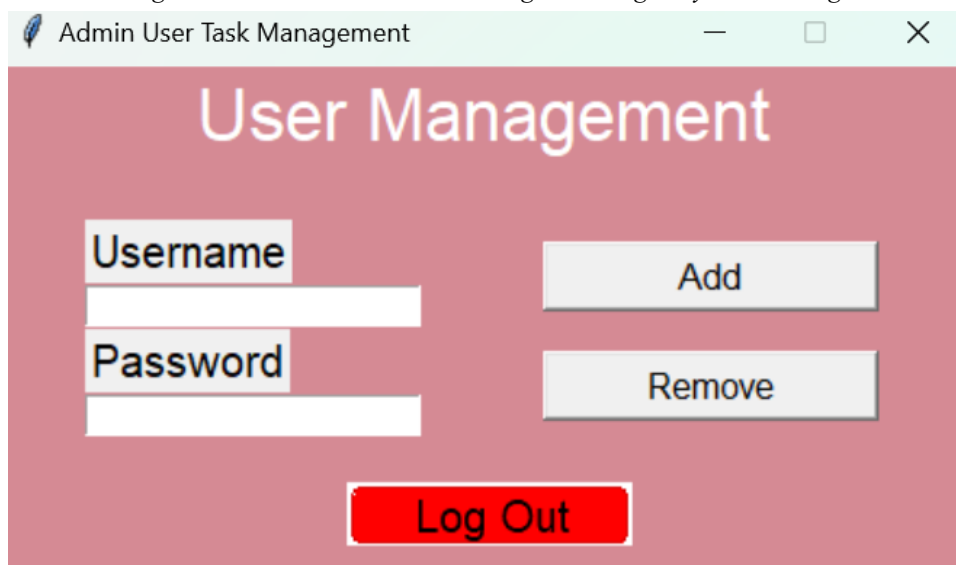


Figure 17: Admin User's User Management Page Layout

Graphical User Interface Design

Admin User's User Management Page Code

```
#Admin User Page - TMS
import tkinter as tk
import tkmacosx as tkm
window=tk.Tk()
window.title("Admin User Task Management")
window.geometry('445x250')
window.resizable(False,False)
window.configure(bg="#D58A94")

banner = tk.Label(window, text="User Management", bg='#D58A94', fg='white', font=("Helvetica", 25))
banner.pack(side=tk.TOP, fill=tk.X)

addUserEnt = tk.Entry(window,width = 25)
addUserEnt.place(x=40, y=100)

addUserLabel = tk.Label(text='Username',font=('Arial',15))
addUserLabel.place(x=40,y=70)

removeUserEnt = tk.Entry(window,width = 25)
removeUserEnt.place(x = 40, y=150)

removeUserLabel = tk.Label(text='Password',font=('Arial',15))
removeUserLabel.place(x=40,y=120)

addUserButton = tk.Button(text='Add',
                           font=('Arial',12),
                           width=16)
addUserButton.place(x=250, y=80)

removeUserButton = tk.Button(text='Remove',
                              font=('Arial',12),
                              width=16)
removeUserButton.place(x=250, y=130)

ExitButton = tkm.Button(text='Log Out',font=('Arial',15),bg='red',command=window.destroy)
ExitButton.place(x=160,y=190)

window.mainloop()
```

Graphical User Interface Design

Admin User's Admin Management Page

Abstract: This is the page specific to the admin and the admin only. This is where they can manage the task management system. They can use this page to change their name and password. The admin user can leave this page using the log-out option.

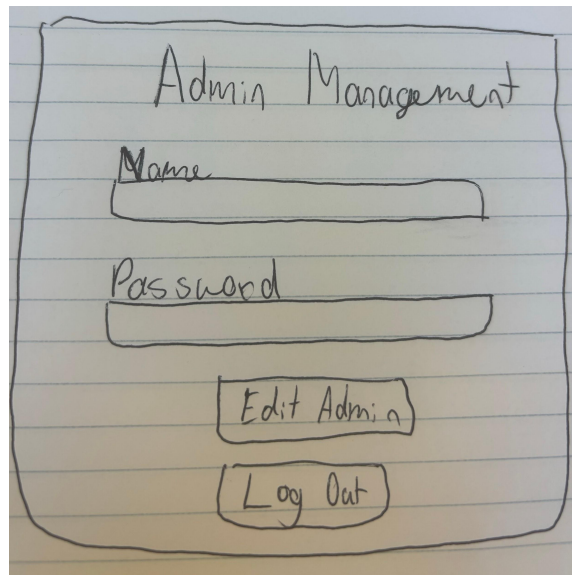


Figure 18: Admin User's Admin Management Page Layout Drawing

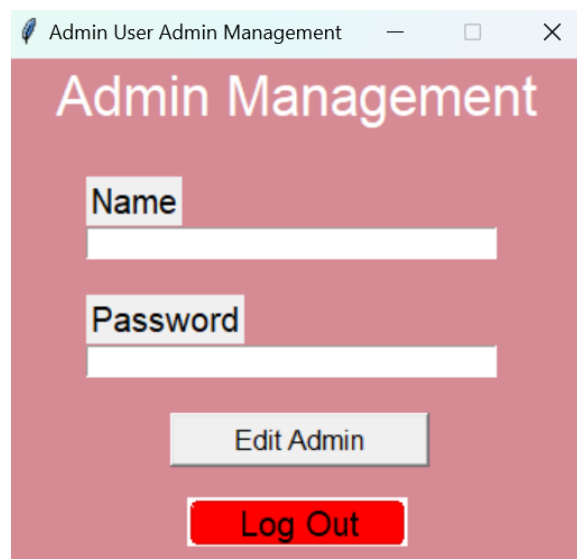


Figure 19: Admin User's Admin Management Page

Graphical User Interface Design

Admin User's Admin Management Page Code

```

import tkinter as tk
import tkmacosx as tkm
window=tk.Tk()
window.title("Admin User Admin Management")
window.geometry('350x330')
window.resizable(False,False)
window.configure(bg="#D58A94")

banner = tk.Label(window, text="Admin Management", bg='#D58A94', fg='white',
font=("Helvetica", 25))
banner.pack(side=tk.TOP, fill=tk.X)

nameEnt = tk.Entry(window,width = 40)
nameEnt.place(x=50, y= 100)

nameLabel = tk.Label(text='Name',font=('Arial',15))
nameLabel.place(x=50,y=70)

passwordEnt = tk.Entry(window,width = 40)
passwordEnt.place(x=50, y= 170)

passwordLabel = tk.Label(text='Password',font=('Arial',15))
passwordLabel.place(x=50,y=140)

editAdminButton = tk.Button(text='Edit Admin',
                             font=('Arial',12),
                             width=16)
editAdminButton.place(x=100, y=210)

ExitButton = tkm.Button(text='Log Out',font=('Arial',15),bg='red',command=window.destroy)
ExitButton.place(x=110,y=260)

window.mainloop()

```

Graphical User Interface Design

Calendar Page

Abstract: The calendar page showcases the various tasks through a weekly, monthly, or yearly format. In the top left, the starting date of the tasks is entered, and in the top right, three options are given to the user of how they want their tasks viewed. When one of the options is selected, all tasks that someone has will be shown on the calendar display within the given period.

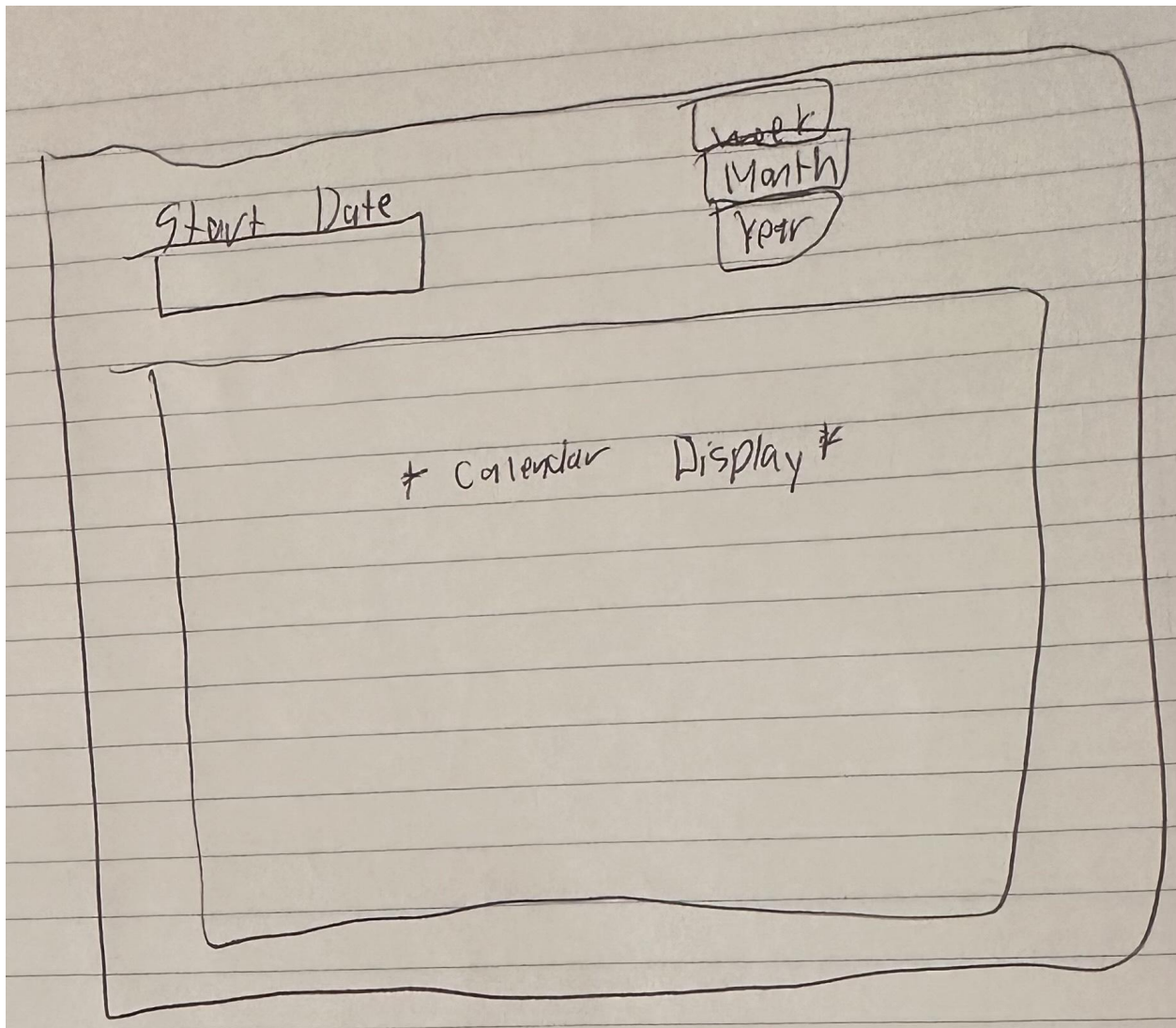


Figure 20: Calendar Page Layout Drawing

Calendar Page

Task Management System

Date (mm/dd/year)

Week

Month

Year

Calendar Display Will Go Here

Close Window

Figure 21: Calendar Page Layout

Graphical User Interface Design

Calendar Page Code

```
#Calendar Page - TMS
import tkinter as tk
import tkmacosx as tkm
window=tk.Tk()
window.title("Calendar Page")
window.geometry('800x600')
window.resizable(False,False)
window.configure(bg="#D58A94")

banner = tk.Label(window, text="Task Management System", bg='#D58A94', fg='white', font=("Helvetica", 25))
banner.pack(side=tk.TOP, fill=tk.X)

dateEnt = tk.Entry(window,width = 25)
dateEnt.place(x = 10, y=80)

dateLabel = tk.Label(text='Date (mm/dd/year)',font=('Arial',15))
dateLabel.place(x=10,y=50)

addButton = tk.Button(text='Week',
                      font=('Arial',12),
                      width=16)
addButton.place(x= 200, y=50)

editButton = tk.Button(text='Month',
                      font=('Arial',12),
                      width=16)
editButton.place(x=400, y=50)

searchButton = tk.Button(text='Year',
                        font=('Arial',12),
                        width=16)
searchButton.place(x=600, y=50)

displayBox = tk.Label(window, text = 'Calendar Display Will Go Here')
displayBox.place(x = 0, y = 110)

ExitButton = tkm.Button(text='Close Window',font=('Arial',15),bg='red',command=window.destroy)
ExitButton.place(x=650,y=550)

window.mainloop()
```

Graphical User Interface Design

Internal Connection Flow Chart

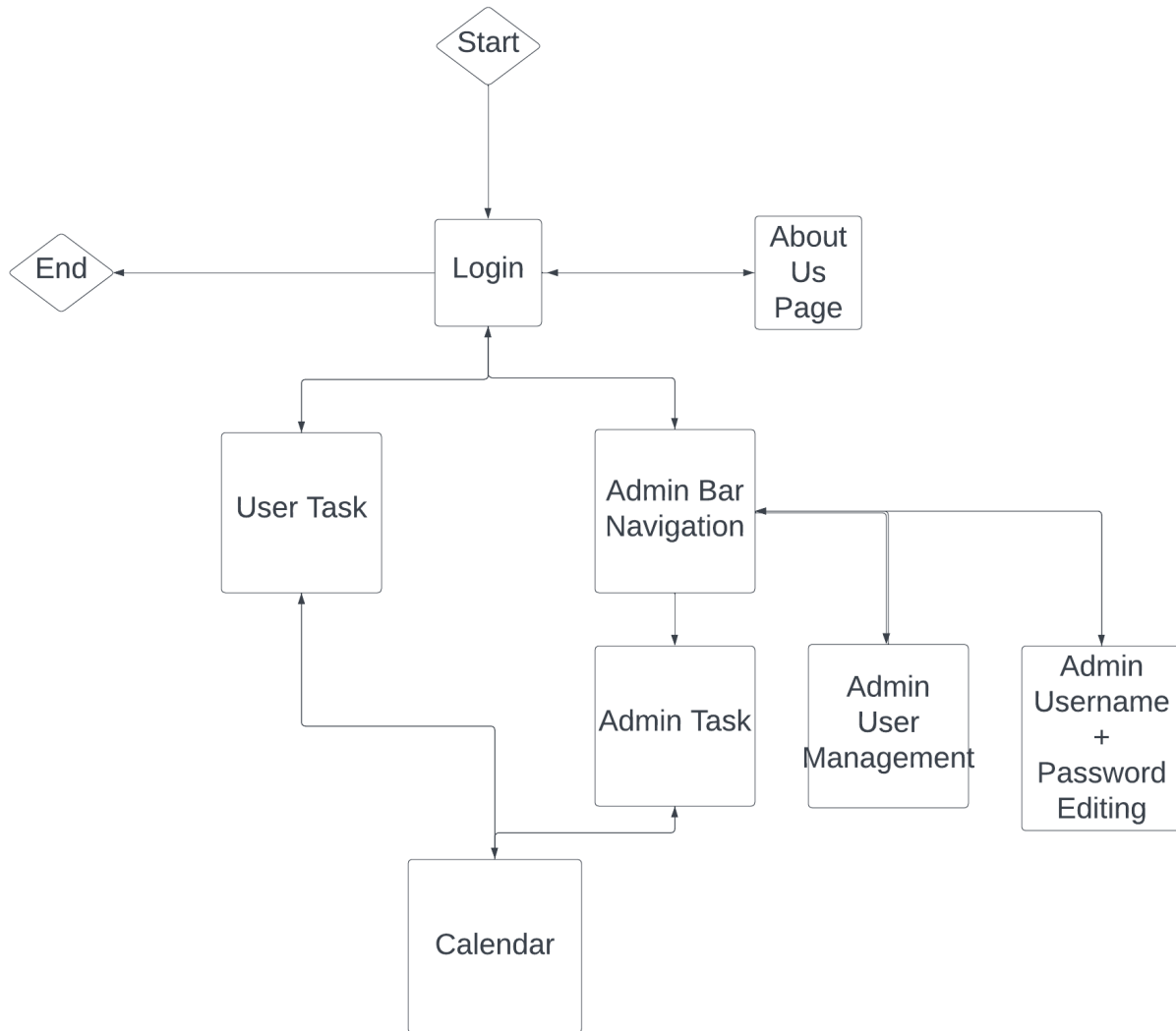


Figure 22: Management System Layout

Graphical User Interface Design

Task Management System Code

```
#Libraries
import tkinter as tk
import tkmacosx as tkm
from PIL import ImageTk, Image
import os
import csv

#Creating the CSV files
os.chdir('C:\\Users\\chris\\OneDrive - Marist College\\Desktop\\Intro to Programming\\Final Project
Phase 03\\TMS Phase 03 CSV')
#Password

passwordFile = 'Passwords.csv'
with open(passwordFile, 'a') as file:
    field = ['Username', 'Password']
    rows = [['AdminUser', 'TMSADMIN23!']]
    csvwriter = csv.writer(file)
    csvwriter.writerow(field)
    csvwriter.writerows(rows)

#Creating the variables of admin username and admin password to compare for later in the program
adminUsernameFinal = 'AdminUser'
adminPasswordFinal = 'TMSADMIN23!'

#Task
taskFile = 'Task.csv'
with open(taskFile, 'a') as file:
    field = ['User', 'Title', 'Description', 'Date', 'Start Time', 'Duration']
    csvwriter = csv.writer(file)
    csvwriter.writerow(field)

global track_username
track_username = ""

#Page Functions
def loadLogIn():
    #Login Page - TMS
    window1=tk.Tk()
    window1.title("Login Page")
    window1.geometry('400x340')
```

```

window1.resizable(False,False)
window1.configure(bg='#D58A94')

#task_image = ImageTk.PhotoImage(file="/Users/nickmuratore/Downloads/TASK.png")
#clipboard_image = ImageTk.PhotoImage(file="/Users/nickmuratore/Downloads/clipboard.png")

#task_label = tk.Label(window1, image=task_image, bg='white')
#task_label.place(x=20, y=80)

#clip_label = tk.Label(window1, image=clipboard_image, bg='white')
#clip_label.place(x=530, y=90)

banner = tk.Label(window1, text="Task Management System", bg='#D58A94', fg='white',
font=("Helvetica", 25))
banner.pack(side=tk.TOP, fill=tk.X)

label = tk.Label(text='Login Page', bg='#D58A94', fg= 'white', font=('Arial',30))
label.pack(side=tk.TOP, fill=tk.X)

userEnt = tk.Entry(window1,width = 35,bg='white')
userEnt.configure(highlightbackground='#89CFF0')
userEnt.place(x=90, y=148)

userLabel = tk.Label(text='Username',bg='white',font=('Arial',15))
userLabel.place(x=90,y=116)

passEnt = tk.Entry(window1,width = 35,bg='white')
passEnt.configure(highlightbackground='#89CFF0')
passEnt.place(x=90, y=210)

passLabel = tk.Label(text='Password',bg='white',font=('Arial',15))
passLabel.place(x=90,y=179)

def logInDirections():
    global track_username
    if(str(userEnt.get()) == adminUsernameFinal and str(passEnt.get()) == adminPasswordFinal):
        track_username = str(userEnt.get())
        window1.destroy()
        loadNavBar()
    elif(str(userEnt.get()) == 'user' and str(passEnt.get()) == 'pass'):
        track_username = str(userEnt.get())
        window1.destroy()
        loadUserPage()
    else:
        print('No')

```

```

submitButton = tk.Button(text='Submit',font=('Arial',16),bg='white',width=16, command =
logInDirections)
submitButton.place(x=100, y=239)
submitButton.configure(highlightbackground='white')

about_us_pageButton = tk.Button(text='About Us',font=('Arial',9),bg='white',width=8, command =
lambda : (window1.destroy(), about_us_page()))
about_us_pageButton.place(x=310, y=300)
about_us_pageButton.configure(highlightbackground='white')

ExitButton = tk.Button(text='Exit',font=('Arial',15),bg='#FF3131', command = window1.destroy)
ExitButton.place(x=135,y=300)
ExitButton.configure(highlightbackground='black')

window1.mainloop()

#User Page
def loadUserPage():
    window2=tk.Tk()
    window2.title("User Task Management")
    window2.geometry('800x450')
    window2.resizable(False,False)
    window2.configure(bg="#D58A94")

    print(track_username)
    banner = tk.Label(window2, text="Task Management", bg="#D58A94", fg='white', font=("Helvetica",
25))
    banner.pack(side=tk.TOP, fill=tk.X)

    titleEnt = tk.Entry(window2,width = 25)
    titleEnt.place(x=10, y=100)

    titleLabel = tk.Label(text='Title',font=('Arial',15))
    titleLabel.place(x=10,y=70)

    desEnt = tk.Entry(window2,width = 25)
    desEnt.place(x = 10, y=160)

    desLabel = tk.Label(text='Description',font=('Arial',15))
    desLabel.place(x=10,y=130)

    dateEnt = tk.Entry(window2,width = 25)
    dateEnt.place(x=10, y=220)

    dateLabel = tk.Label(text='Date (mm/dd/year)',font=('Arial',15))
    dateLabel.place(x=10,y=190)

    durEnt = tk.Entry(window2,width = 25)

```

```
durEnt.place(x = 10, y=280)
```

```
durLabel = tk.Label(text='Duration',font=('Arial',15))
durLabel.place(x=10,y=250)
```

```
startLabel = tk.Label(text='Start Time (Military)',font=('Arial',15))
startLabel.place(x=10,y=310)
```

```
startEnt = tk.Entry(window2,width = 25)
startEnt.place(x = 10, y=340)
```

```
def add():
    u = str(track_username)
    t = str(titleEnt.get())
    d = str(desEnt.get())
    da = str(dateEnt.get())
    du = str(durEnt.get())
    s = str(startEnt.get())
    taskFile = 'Task.csv'
    with open(taskFile, 'a') as file:
        field = [u, t, d, da, du, s]
        csvwriter = csv.writer(file)
        csvwriter.writerow(field)
    file.close()
```

```
addButton = tk.Button(text='Add',font=('Arial',12),width=16, command=add)
addButton.place(x=230, y=120)
```

```
def edit():
    u = str(track_username)
    t = str(titleEnt.get())
    d = str(desEnt.get())
    du = str(durEnt.get())
    s = str(startEnt.get())
    with open('contact.txt', 'r') as file:
        contact = file.read()
    lines = contact.split(sep='\n')
    for c in lines:
        cc = c.split(sep=',')
        if(cc[0] == oldN):
            data = contact.replace(c, oldN + ',' + str(oldC))
            file = open('contact.txt', 'w')
            file.write(data)
    file.close()
```

```
editButton = tk.Button(text='Edit',font=('Arial',12),width=16)
editButton.place(x=230, y=170)
```

```

def search():
    u = str(track_username)
    t = str(titleEnt.get())
    d = str(desEnt.get())
    da = str(dateEnt.get())
    du = str(durEnt.get())
    s = str(startEnt.get())
    with open('Task.csv', 'r') as file:
        content = csv.reader(file)
        for lines in content:
            try:
                if lines[0] == u and lines[1] == t and lines[2] == d and lines[3] == da and lines[4] == s and
lines[5] == du:
                    lblTitleofDisplay=tk.Label(text='Searched Task',font=('Arial',15))
                    lblTitleofDisplay.place(x=400,y=70)
                    lblDisplay=tk.Label(text=(f {lines[0]}, {lines[1]}, {lines[2]}, {lines[3]}, {lines[4]},
{lines[5]}'),font=('Arial',8))
                    lblDisplay.place(x=400,y=100)
            except IndexError:
                None
    file.close()

searchButton = tk.Button(text='Search',font=('Arial',12),width=16,command=search)
searchButton.place(x=230, y=220)

def remove():
    u = str(track_username)
    t = str(titleEnt.get())
    d = str(desEnt.get())
    du = str(durEnt.get())
    s = str(startEnt.get())
    with open('contact.txt', 'r') as file:
        contact = file.read()
        lines = contact.split(sep='\n')
        for c in lines:
            cc = c.split(sep=',')
            if(cc[0] == n):
                data = contact.replace(c, "")
                file = open('contact.txt', 'w')
                file.write(data)
    file.close()

removeButton = tk.Button(text='Remove',font=('Arial',12),width=16,command=remove)
removeButton.place(x=230, y=270)

ExitButton = tk.Button(text='Log Out',font=('Arial',15),bg='red',command= lambda:
(window2.destroy(), loadLogIn()))
ExitButton.place(x=225,y=390)

```



```

CalenderButton = tkm.Button(text='Calender',font=('Arial',15),bg='blue', command = lambda:
(window2.destroy(), loadCalendarPage1()))
CalenderButton.place(x=50,y=390)

window2.mainloop()

#Admin Nav Bar
#Admin User Nav Bar - TMS
def loadNavBar():
    window3=tk.Tk()
    window3.title("Admin User")
    window3.geometry('475x225')
    window3.resizable(False,False)
    window3.configure(bg="#D58A94")

    banner = tk.Label(window3, text="Task Management System", bg='#D58A94', fg='white',
font=("Helvetica", 25))
    banner.pack(side=tk.TOP, fill=tk.X)

    sub_banner = tk.Label(window3, text="Admin User Options", bg='#D58A94', fg='white',
font=("Helvetica", 18))
    sub_banner.pack(side=tk.TOP, fill=tk.X)

    Task_Btn = tk.Button(text='Task Management',font=('Arial',11), command = lambda:
(window3.destroy(), loadAdminTaskMan()))
    Task_Btn.place(x=20, y=90)

    User_Btn = tk.Button(text='User Management',font=('Arial',11), command = lambda:
(window3.destroy(), loadAdminUserMan()))
    User_Btn.place(x=170, y=90)

    Admin_Btn = tk.Button(text='Admin Management',font=('Arial',11), command = lambda:
(window3.destroy(), loadAdminAdminMan()))
    Admin_Btn.place(x=320, y=90)

    ExitButton = tkm.Button(text='Log Out',font=('Arial',15),bg='red',command= lambda:
(window3.destroy(), loadLogIn()))
    ExitButton.place(x=170,y=160)

    window3.mainloop()

#Admin User - Task Management
#Admin User Page - TMS
def loadAdminTaskMan():
    window4=tk.Tk()
    window4.title("Admin User Task Management")
    window4.geometry('400x450')

```

```

window4.resizable(False,False)
window4.configure(bg="#D58A94")

banner = tk.Label(window4, text="Task Management", bg='#D58A94', fg='white', font=("Helvetica",
25))
banner.pack(side=tk.TOP, fill=tk.X)

titleEnt = tk.Entry(window4,width = 25)
titleEnt.place(x=10, y=100)

titleLabel = tk.Label(text='Title',font=('Arial',15))
titleLabel.place(x=10,y=70)

desEnt = tk.Entry(window4,width = 25)
desEnt.place(x = 10, y=160)

desLabel = tk.Label(text='Description',font=('Arial',15))
desLabel.place(x=10,y=130)

dateEnt = tk.Entry(window4,width = 25)
dateEnt.place(x=10, y=220)

dateLabel = tk.Label(text='Date (mm/dd/year)',font=('Arial',15))
dateLabel.place(x=10,y=190)

durEnt = tk.Entry(window4,width = 25)
durEnt.place(x = 10, y=280)

durLabel = tk.Label(text='Duration',font=('Arial',15))
durLabel.place(x=10,y=250)

startLabel = tk.Label(text='Start Time (Military)',font=('Arial',15))
startLabel.place(x=10,y=310)

startEnt = tk.Entry(window4,width = 25)
startEnt.place(x = 10, y=340)

def add():
    u = str(track_username)
    t = str(titleEnt.get())
    d = str(desEnt.get())
    da = str(dateEnt.get())
    du = str(durEnt.get())
    s = str(startEnt.get())
    taskFile = 'Task.csv'
    with open(taskFile, 'a') as file:
        field = [u, t, d, da, du, s]
        csvwriter = csv.writer(file)

```

```

        csvwriter.writerow(field)
    file.close()

addButton = tk.Button(text='Add',font=('Arial',12),width=16,command=add)
addButton.place(x=230, y=120)

editButton = tk.Button(text='Edit',
                        font=('Arial',12),
                        width=16)
editButton.place(x=230, y=170)

searchButton = tk.Button(text='Search',
                          font=('Arial',12),
                          width=16)
searchButton.place(x=230, y=220)

removeButton = tk.Button(text='Remove',
                          font=('Arial',12),
                          width=16)
removeButton.place(x=230, y=270)

#Task Data Function For Admin

ExitButton = tk.Button(text='Return To Nav Bar',font=('Arial',15),bg='red',command= lambda:
(window4.destroy(), loadNavBar()))
ExitButton.place(x=195,y=390)

CalenderButton = tk.Button(text='Calender',font=('Arial',15),bg='blue', command = lambda:
(window4.destroy(), loadCalendarPage2()))
CalenderButton.place(x=30,y=390)

window4.mainloop()

#Admin User - User Management
#Admin User Page - TMS
def loadAdminUserMan():
    window5=tk.Tk()
    window5.title("Admin User Task Management")
    window5.geometry('445x250')
    window5.resizable(False,False)
    window5.configure(bg="#D58A94")

    banner = tk.Label(window5, text="User Management", bg="#D58A94", fg='white', font=("Helvetica",
25))
    banner.pack(side=tk.TOP, fill=tk.X)

    addUserEnt = tk.Entry(window5,width = 25)

```

```

addUserEnt.place(x=40, y=100)

addUserLabel = tk.Label(text='Username',font=('Arial',15))
addUserLabel.place(x=40,y=70)

removeUserEnt = tk.Entry(window5,width = 25)
removeUserEnt.place(x = 40, y=150)

removeUserLabel = tk.Label(text='Password',font=('Arial',15))
removeUserLabel.place(x=40,y=120)

addUserButton = tk.Button(text='Add',
                           font=('Arial',12),
                           width=16)
addUserButton.place(x=250, y=80)

removeUserButton = tk.Button(text='Remove',
                              font=('Arial',12),
                              width=16)
removeUserButton.place(x=250, y=130)

#User Data Function For User

ExitButton = tk.Button(text='Return To Nav Bar',font=('Arial',15),bg='red', command= lambda:
(window5.destroy(), loadNavBar()))
ExitButton.place(x=140,y=190)

window5.mainloop()

#Admin user - Admin Management
#Admin User Page - TMS
def loadAdminAdminMan():
    window6=tk.Tk()
    window6.title("Admin User Admin Management")
    window6.geometry('350x330')
    window6.resizable(False,False)
    window6.configure(bg="#D58A94")

    banner = tk.Label(window6, text="Admin Management", bg='#D58A94', fg='white', font=("Helvetica",
25))
    banner.pack(side=tk.TOP, fill=tk.X)

    nameEnt = tk.Entry(window6,width = 40)
    nameEnt.place(x=50, y= 100)

    nameLabel = tk.Label(text='Name',font=('Arial',15))
    nameLabel.place(x=50,y=70)

```

```

passwordEnt = tk.Entry(window6,width = 40)
passwordEnt.place(x=50, y= 170)

passwordLabel = tk.Label(text='Password',font=('Arial',15))
passwordLabel.place(x=50,y=140)

editAdminButton = tk.Button(text='Edit Admin',
                             font=('Arial',12),
                             width=16)
editAdminButton.place(x=100, y=210)

ExitButton = tk.Button(text='Return To Nav Bar',font=('Arial',15),bg='red',command= lambda:
(window6.destroy(), loadNavBar()))
ExitButton.place(x=90,y=260)

window6.mainloop()

#Calendar Page
#Calendar Page - TMS
def loadCalendarPage1():
    window7=tk.Tk()
    window7.title("Calendar Page")
    window7.geometry('800x600')
    window7.resizable(False,False)
    window7.configure(bg="#D58A94")

    banner = tk.Label(window7, text="Task Management System", bg='#D58A94', fg='white',
font=("Helvetica", 25))
    banner.pack(side=tk.TOP, fill=tk.X)

    dateEnt = tk.Entry(window7,width = 25)
    dateEnt.place(x = 10, y=80)

    dateLabel = tk.Label(text='Date (mm/dd/year)',font=('Arial',15))
    dateLabel.place(x=10,y=50)

    weekButton = tk.Button(text='Week',font=('Arial',12),width=16)
    weekButton.place(x= 200, y=50)

    monthButton = tk.Button(text='Month',font=('Arial',12),width=16)
    monthButton.place(x=400, y=50)

    yearButton = tk.Button(text='Year',font=('Arial',12),width=16)
    yearButton.place(x=600, y=50)

    displayBox = tk.Label(window7, text = 'Calendar Display Will Go Here')
    displayBox.place(x = 10, y = 110)

```

```
ExitButton = tk.Button(text='Return To Task Management',font=('Arial',15),bg='red', command =
lambda: (window7.destroy(), loadUserPage()))
ExitButton.place(x=520,y=550)
```

```
window7.mainloop()
```

```
def loadCalendarPage2():
```

```
    window8=tk.Tk()
    window8.title("Calendar Page")
    window8.geometry('800x600')
    window8.resizable(False,False)
    window8.configure(bg="#D58A94")
```

```
    banner = tk.Label(window8, text="Task Management System", bg='#D58A94', fg='white',
font=("Helvetica", 25))
    banner.pack(side=tk.TOP, fill=tk.X)
```

```
    dateEnt = tk.Entry(window8,width = 25)
    dateEnt.place(x = 10, y=80)
```

```
    dateLabel = tk.Label(text='Date (mm/dd/year)',font=('Arial',15))
    dateLabel.place(x=10,y=50)
```

```
    weekButton = tk.Button(text='Week',font=('Arial',12),width=16)
    weekButton.place(x= 200, y=50)
```

```
    monthButton = tk.Button(text='Month',font=('Arial',12),width=16)
    monthButton.place(x=400, y=50)
```

```
    yearButton = tk.Button(text='Year',font=('Arial',12),width=16)
    yearButton.place(x=600, y=50)
```

```
    displayBox = tk.Label(window8, text = 'Calendar Display Will Go Here')
    displayBox.place(x = 10, y = 110)
```

```
    ExitButton = tk.Button(text='Return To Task Management',font=('Arial',15),bg='red', command =
lambda: (window8.destroy(), loadAdminTaskMan()))
    ExitButton.place(x=520,y=550)
```

```
    window8.mainloop()
```

```
def about_us_page():
```

```
    window9 = tk.Tk()
    window9.title("About Us Page")
    window9.geometry('800x600')
    window9.resizable(False,False)
```

```

window9.configure(bg='pink')

# Main project description
main_description_label = tk.Label(window9, text="About Our Project",bg='pink', font=('Helvetica', 22,
'bold'))
main_description_label.pack(pady=10)

main_description_text = "      Our project is a Task Management System (TMS) that allows users to
manage their tasks
by adding, removing, editing, and searching their tasks. For each task, they enter a title, description,
date, and duration to which their tasks will be stored in a csv database. Hope you enjoy our project
and find it useful. Let's meet the creators of TMS, otherwise known as MC Squared!"

main_description = tk.Label(window9, text=main_description_text, bg='pink', font=('Helvetica', 12),
justify='left')
main_description.place(x=45, y=75)

# Nick
nick_label = tk.Label(window9, text="Nick",bg='pink', font=('Helvetica', 16, 'underline'))
nick_label.place(x=50,y=200)

nick_placeholder_label = tk.Label(window9, text="Senior\nCybersecurity",bg='pink',font=('Helvetica',
11))
nick_placeholder_label.place(x=28,y=500)

# Andrew
andrew_label = tk.Label(window9, text="Andrew", bg='pink', font=('Helvetica', 16, 'underline'))
andrew_label.place(x=250,y=200)

andrew_placeholder_label = tk.Label(window9, text="Freshmen\nSoftware
Development",bg='pink',font=('Helvetica', 11))
andrew_placeholder_label.place(x=200,y=500)

# Ben
ben_label = tk.Label(window9, text="Ben",bg='pink', font=('Helvetica', 16, 'underline'))
ben_label.place(x=485,y=200)

ben_placeholder_label = tk.Label(window9, text="Freshmen\nSoftware
Development",bg='pink',font=('Helvetica', 11))
ben_placeholder_label.place(x=415,y=500)

# Chris
Chris_label = tk.Label(window9, text="Chris",bg='pink', font=('Helvetica', 16, 'underline'))
Chris_label.place(x=675,y=200)

Chris_placeholder_label = tk.Label(window9, text="Freshmen\nGame Design &
Programming",bg='pink',font=('Helvetica', 11))

```

```
Chris_placeholder_label.place(x=588,y=500)
```

```
ExitButton = tk.Button(text='Return To Log In',font=('Arial',15),bg='red', command = lambda:  
(window9.destroy(), loadLogIn()))
```

```
ExitButton.place(x=20,y=20)
```

```
window9.mainloop()
```

#On Opening

#Way To Check If It's Been Previously Called

#How to create and check location on users laptop?

#Some condition in the csv file?

#Check For Admin user

loadLogIn()

Virtual Environment [5]

Link To Video Of Code Running In Virtual Environment

<https://drive.google.com/drive/folders/1qI3AcKppAlemQWo13VqlGnrQaR795cU7?usp=sharing>

Libraries [6]

Tkinter

Tkmacosx

PIL

References [7]

<https://www.lucidchart.com>

<https://www.geeksforgeeks.org/>

<https://pypi.org/>

<https://stackoverflow.com/>