**Senior Project, Company Calendar Program**

**Jakob Fick**

**Methodist University**

**December 3, 2024Introduction**

For the final project of my Senior Project class (CSC 4900), I created a company calendar program. The program is designed to assist medium sized companies with scheduling events and communicating with employees. Many companies experience issues informing employees about meetings, deadlines, and other events. The point of this program is to create a universal calendar that everyone in the company can use.

The program will essentially function like a google calendar. Each employee will have their own schedule in the database. The program will display different events in the employee’s schedule, and will give them the ability to add, remove, and view different events. However, in this program managers and supervisors will also be able to view the scheduled of their subordinates and will be able to add and remove events from the schedules of those who work under them. The program will also allow external components to communicate with the program. For example, the company website will be able to interact with the program allowing customers schedule meetings with company representatives through the company website. When this happens, those meetings will automatically appear on the representative’s schedule in the program.

The program was designed using Python, while the database was created using the SQL programing language. Additionally, the GUI of the program was made using the Python extension Tkinter. The program had four main components: the employee calendar program, the system manager program, the customer portion (company website), and the database. The database contains the data on all employees, days, events, manager and subordinate relationships, etc. The employee calendar was the program that would allow each employee to view and edit their own calendar and that of their subordinates. This program was made up of the database controller which would access the database, the numerous GUI classes, and the main program which would control both the database controller and the GUI classes. The next program in the system is the system manager program which will be used to add and remove employees from the database. Like the previous program, this program consists of the database controller which accesses the database, numerous GUI classes, and the main program which will control both the database controller and the GUI classes. The final part of the system is the customer program, which will allow customers to schedule appointments with employees. This program consisted of the company website, and the database controller which the website will interact with. To demonstrate how the program works, let’s walk through the system operations. We will start by discussing the employee calendar program.

A screenshot of a computer

Description automatically generatedA screenshot of a calendar

Description automatically generatedThis is the log in screen of the calendar program. When an employee opens the software to view their calendar, they will see this this screen. The user will then enter the username and password that have been assigned to them and press the login button. When the button is pressed, the program will search the database for an employee with a matching username. If one is found, the program will then check if the password assigned to that employee matches the one entered. If it matches, the program will then display the ‘day screen’. If the username entered does not exist in the database, then an error message stating “Username does not exist” will be displayed. Similarly, error messages will be displayed if the program fails to connect to the database, or if the password entered does not match.

A screenshot of a computer

Description automatically generated If the username entered exists in the database, and the password matches, then the program save the database ID number of the selected employee. Next the program will open the schedule for the selected day and display it in the ‘day screen’. The current date will be selected by default. The program will search the database and retrieve information about all the events listed under the selected employee taking place on the selected day. Those events will be returned as a list. The program will then create a button in the GUI screen for each event, with the size and position of the button representing the start time, end time, and duration of the event. The ‘day screen’ will also display buttons labeled ‘next day’, ‘previous day’, ‘select date’, ‘add new event’, ‘view subordinate’, and ‘logout’. The picture above represents the day screen.

If the next day button is selected, all event buttons will be erased from the screen. Then the program will retrieve all events in the selected employees schedule taking place on the day after the selected date. Similarly, if the ‘prev day’ button is pressed the program will retrieve and display the schedule of the day before the selected day. Also, if the logout button is selected, the connection to the database will be closed and ID number of the currently selected employee will be set to null. The program will then return to the login screen to allow a different employee to log into the system.

A screenshot of a calendar

Description automatically generatedIf the ‘Add new event’ button of the day screen is pressed, then the program will display the ‘add new event screen’ which is displayed below. The user will be able to enter the date of the new event using the calendar (click on the date of the event). The user will then enter the start time of the event by using drop down menus to enter the hour, minute, and am or pm of the time the event will begin. The user will also enter the end time of the event in the same way. Finally, the use will use the text boxes to enter the name and a brief description of the event. When the user is finished, they will either press the ‘create event’ or ‘cancel’ button. If the user presses the ‘cancel’ button, the program will return to the day screen without making any changes. If the ‘create event’ button is pressed, then the information entered will be sent to the database. The program will first check if there is already an event scheduled during the time selected for the new event. If there is already an event scheduled during this time, then an error message will be displayed. Otherwise, the new event will be added to the database.

The program will then return to the day screen and will update the GUI so that the new event is displayed. As an example, let’s say the user created a new event called “Project Meeting” scheduled from 10am to 11am. The picture bellow shows what the updated day screen will look like after adding the event.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated I previously mentioned that each event is represented by a button in the day screen. If the button for an event is pressed, the program will retrieve all the information about that event from the database. Then the ‘event screen’ will be displayed which will show this information to the user. For example, when the button for the “Project Meeting” event is pressed, the example below is what will be displayed.

The ‘event screen’ will have two buttons, the ‘Remove’ button and the ‘Exit’ button. If the ‘Exit’ button is pressed, then the program will return to the day screen without making any changes. However, if the ‘Remove’ button is pressed, the program will remove the currently selected event from the database. Then the program will return to the day screen and update the GUI so that the deleted event is no longer displayed. For example, the picture below shows what will happen if the “Project Meeting” event is removed from the calendar.

A screenshot of a computer

Description automatically generated

A screenshot of a calendar

Description automatically generated

While in the day screen, if the user presses the ‘Select date’ button, then the ‘Select Date’ screen will be shown. The ‘Select Date’ screen is shown above and can be used to select the date the user would like to view. When this screen is displayed, the user will be able to choose a date by clicking on the day in the calendar. Then the user can press the “Show selected date” button. When this button is pressed, the program will search the database for all events listed under the selected employee that are taking place on the selected day. The program will then reset the day screen so that it displays all the recovered events.

A screenshot of a calendar

Description automatically generated

While in the day screen, if the user selects the ‘view subordinate’ button, the program will show the ‘select subordinate’ screen which is shown above. This screen will allow the user to select which of their employees they would like to view the schedule of. When the button is pressed, the program will search the database for all employees that have the current employee listed as their superior. These employees will be returned as a list, and a button will be shown for each employee returned. If the user presses the button for one of the employee’s listed, the program will retrieve all events listed under that employee on the selected date (the current date will be chosen by default) and then display the day screen for the subordinate selected. If the ‘cancel’ button is selected, then the program will return to the day screen without making any changes.

For example, say the current user has two subordinates: Jane Smith and Alice Johnson. The user presses the ‘View Subordinate’ button then the above screen will be displayed. Then, if the user selects the ‘Jane Smith’ button, the screen below will be displayed showing Jane Smith’s schedule on the current date.

A screenshot of a computer

Description automatically generated

The day screen displayed while viewing a subordinate will look and function the same as the normal day screen. This will allow the user to view the schedule of their subordinates. It will allow them to add and remove events from their subordinates’ schedules. The only difference is that the ‘View Subordinates’ button is replaced with the ‘Return to my Calendar’ button. When this button is pressed, the program will return to the previous day screen of the original user and the originally selected date. The picture below shows what will happen if the user presses the ‘Return to my Calendar’ button while viewing their subordinate’s schedule.

A screenshot of a computer

Description automatically generated

This summarizes the employee calendar program, which will be used by employees to view and edit their schedule. Next let’s discuss the system manager program. This part of the system will be used to add and remove employees from the system database. System managers will be the ones who will use this program and are the only ones able to access to it. Here is how the system manager program will work.

A screenshot of a calendar

Description automatically generated

A screenshot of a computer

Description automatically generatedThis is the log in screen of the system manager program. When an employee opens the software, they will see this this screen. The user will then enter the username and password that have been assigned to them and press the login button. When the button is pressed, the program will search the database for an employee with a matching username. If one is found, the program will then check if the password assigned to that employee matches the one entered, and if the employee’s role in the company is ‘System Manager’. If the passwords match and the employee’s role is ‘System Manager’ in the database, the program will then display the ‘menu’ page. If the username entered does not exist in the database, then an error message stating “Username does not exist” will be displayed. Similarly, error messages will be displayed if the program fails to connect to the database, the employee with the matching username is not a ‘System Manager’, or if the password entered does not match. The picture above demonstrates how these error messages will be displayed.

A screenshot of a computer

Description automatically generated

If an employee with a matching username is found in the database, that employee is a ‘System Manager’, and the password entered matches the one in the database then the ‘menu’ screen will be displayed. The picture above shows the ‘menu’ screen. This page allows the user to select what actions they would like to perform. The options will be displayed by three buttons. The ‘Add an Employee’ button will allow the user to add a new employee to the database. The ‘Remove an Employee’ button will let the user select an employee and remove them from the database. And the ‘logout’ button will let the user leave the program. If the ‘logout’ button is presses, the program will return to the ‘login’ screen.

If the user presses the ‘Add an Employee’ button in the ‘menu’ screen, then the ‘New employee’ screen will be displayed. In this screen, the user will be able to enter information about the new employee using textboxes, such as their first name, last name, username, and password for the employee calendar program. The user will also be able to select the company role of the new employee form a dropdown menu. Options will include things like accountant, system manager, company representative, etc. Once the user enters the information about the new employee, they can press the ‘next’ button, which will cause the program to move to the ‘select superior’ screen. The picture below shows what the ‘Add an Employee’ screen will look like.

A screenshot of a computer

Description automatically generatedAs an example, let’s assume the user is adding a new employee to the program. The new employees name is Bob Jones. Their username is [bob.jones@example.com](mailto:bob.jones@example.com). Their password is password098. And their company role is ‘programmer’. The user will enter this information and then press the next button.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated After the user presses the ‘next’ button, the ‘select superior’ screen will be displayed. This screen is demonstrated by the picture above and will allow the user to select the superior of the new employee. The user will use the textbox to enter the username of the new employee’s superior. The user will then press the ‘search’ button, and the program will search the database for an employee with a username matching the one entered. If an employee with a matching username is found, that employee will be marked as the new employee’s superior. If no matching username is found, the program will display an error message as shown below. On the other hand, if the user presses the ‘No Superior’ button, the new employee will be recorded as having no superior. After the ‘search’ button is pressed and a matching employee is found, or after the ‘no superior’ button is pressed, the program will move to the ‘select subordinate’ screen.

A computer screen shot of a computer screen

Description automatically generated

As an example, lets assume the user entered the username ‘john.doe@example.com which is the username of the employee ‘John Doe’ who will be selected as the superior of the new employee ‘Bob Jones’ that is currently being created. After John Doe is selected as the superior, the program will move to the ‘select subordinate’ screen.

A screenshot of a computer

Description automatically generated

The picture above shows the ‘Select subordinate’ screen, which will allow the user to select one of the subordinates of the new employee. The user will use the textbox to enter the username of the new employee’s subordinate. The user will then press the ‘search’ button, and the program will search the database for an employee with a username matching the one entered. If an employee with a matching username is found that employee will be marked as the new employee’s subordinate, and the program will display a message stating that the subordinate has been added to the new employee’s file. The picture below displays this message. The user will then be able to search for another subordinate in case the new employee has multiple subordinates. Once all the subordinates of the new employee have been added, the user will press the ‘Done’ button. This will cause the program to move to the ‘review employee’ screen. If the employee has no subordinates, they can simply press the done button before searching for any subordinates. The new employee will be recorded as having no subordinates and the program will move to the ‘review employee’ screen.

A screenshot of a computer

Description automatically generated

If no matching username is found, the program will display an error message as shown below. The user will be able to try again to enter the username of the new employee’s subordinate.

A computer screen shot of a computer screen

Description automatically generated

As an example, let’s assume the user entered the username of the employee ‘Alice Johnson’ as the subordinate of the new employee ‘Bob Jones’ that is currently being created. Then the user added ‘Jane Smith’ as another subordinate and then pressed the ‘done’ button.

A screenshot of a computer

Description automatically generated

After the user presses the ‘done’ button in the ‘select subordinate’ screen, the ‘review employee’ screen will be displayed. The ‘review employee’ screen is demonstrated in the picture above and will show the information of the new employee that is being added to the database. The above example of the ‘review employee’ screen shows what it would look like after entering the information for the employee ‘Bob Jones’. This screen will allow the user to review the information of the new employee and make sure everything is correct before adding them to the database. The screen will have two buttons, the ‘Confirm’ button and the ‘Cancel’ button. If the ‘Cancel’ button is pressed, the program will return to the ‘menu’ screen without making any changes. If the ‘Confirm’ button is pressed, then the program will create a new employee object in the database using the information entered, and then return to the menu screen.

The picture below shows what will happen after the user presses the ‘Confirm’ button in the ‘review employee’ screen. (The program returns to the menu).

A screenshot of a computer

Description automatically generated

If the user presses the ‘Remove an Employee’ button in the ‘menu’ screen, the program will be taken to the ‘Search for Employee’ screen. This picture below demonstrates this screen. It will allow the user to search for the employee they would like to remove from the database. The user will enter the username of the employee they wish to remove into the textbox. If the user presses the search button the program will search the database for an employee with a matching username and if one is found the ‘remove employee’ screen will be displayed. If the user presses the ‘done’ button the program will return to the ‘menu’ screen without making any changes.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

If a user enters a username into the ‘Search for Employee’ screen and presses the ‘search’ button, but no matching username is found in the database, then the program will display an error message. The picture above demonstrates what this error message will look like.

As an example, let’s assume the user entered the username ‘bob.jones@example.com’ (the username of the employee we just created named Bob Jones) into the ‘Search for Employee’ screen and presses search. The ‘remove employee’ screen will then be displayed.

A screenshot of a computer

Description automatically generated

The ‘remove employee’ screen is displayed above. This screen will display the information of the employee the user wants to remove so that the user can make sure this is the employee they had in mind. The ‘remove employee’ screen will display the selected employee’s information and have two buttons, the ‘Delete’ button and the ‘Done’ button. The above example is what would be displayed if the user searched for the new employee ‘Bob Jones’. If the user presses the ‘Done’ button, the program will return to the ‘menu’ screen without making any changes. However, if the user presses the ‘Delete’ button, the program will remove the selected employee (in this case Bob Jones) from the database and then return to the menu screen.

A screenshot of a computer

Description automatically generated

The above picture demonstrates what would happen after the user pressed the ‘Delete’ button and removed the employee ‘Bob Jones’ from the database. The program will return to the menu screen.

That sums up the functionality of the system manager program, which will be used the managers of the calendar program to add and remove new employees from the database. Finally, let’s discuss the customer program. This part of the program will be used by customers to schedule meetings with company representatives. In practice, the system is supposed to interact with the company website. However, for this project a simple Python GUI was created to represent the company website and how it will be used.

A screen shot of a computer

Description automatically generated

Somewhere in the company website there will be a button or link labeled ‘Schedule a Meeting with a representative’ as shown above. When this button/link is pressed, a page similar to the one shown below will be displayed. This page will allow the customer to enter the date they would prefer to schedule the meeting. After the user selects their preferred date, they will press the ‘Select Date’ button.

A screenshot of a computer

Description automatically generated

After the customer selects a date, the company website will send the selected date to the program. The program will then search the database for all employees with the role ‘Company Representative’. The program will search the schedules of all employees with this role and find all one-hour time slots between 8am and 5pm where there is not already an event scheduled. The program will then return all the company representatives and their available time slots to the company website. In a new page, the company website will display all available representatives and the times they are available for an hour-long meeting on the day selected. An example of this page is shown below. The customer can then either select one of the available time slots, or return to the previous page and select a different date to have the meeting.

A screenshot of a chat

Description automatically generated

If the user selects one of the available time slots, the website will send a message to the program and have it add a customer meeting event to the selected company representatives schedule at the selected time. The website will then display a message to the customer telling them that their meeting has been scheduled. An example is shown in the picture below.

A screen shot of a computer

Description automatically generated

As an example, let’s say a customer uses the company website to schedule a meeting with a company representative. They schedule the meeting with the company representative ‘Alice Johnson’ from 10:30am to 11:30am.

A screenshot of a calendar

Description automatically generated

The above picture shows the calendar of the company representative Alice Johnson before the customer schedules their meeting. The picture below shows the calendar of Alice Johnson after the customer scheduled their meeting.

A screenshot of a computer

Description automatically generated

This paper summarizes the majority of the Company Calendar Program I created for my final project. The system consists mainly of the database, the employee calendar program, the system manager program, and the company website. I have tested the system to ensure that it works as intended and fixed any bugs that I found. While the system may not be super complex or fancy, it does work and could be implemented and used by a company if desired.