Meeting coordinator

Final Report

5/8/2020

CSC 340 Ethics and Software Engineering

John Booker, Jonathon Gower, Thomas Morris

Contents

[I. Introduction 2](#_Toc39779751)

[A. Problem Statement 2](#_Toc39779752)

[B. Proposal 2](#_Toc39779753)

[II. System Description 2](#_Toc39779754)

[III. System Requirement 2](#_Toc39779755)

[A. Functional Requirements 2](#_Toc39779756)

[B. Non-functional Requirements 6](#_Toc39779757)

[IV. Use Case Diagram 7](#_Toc39779758)

[V. Class Diagram 8](#_Toc39779759)

[VI. Sequence Diagrams 9](#_Toc39779760)

[VII. Activity Diagrams 15](#_Toc39779761)

[VIII. State Diagram 21](#_Toc39779762)

[IX. Database Design 22](#_Toc39779763)

[X. Conclusion 22](#_Toc39779764)

[XI. Data Dictionary 22](#_Toc39779765)

# Introduction

## Problem Statement

Company ABC, Inc is in need of a software system to help their employees to keep a planner to record their schedule. This planner must allow their employees to coordinate meetings based on attendants’ planners and the availability of conference rooms.

## Proposal

We propose a meeting coordinator software system to help ABC, Inc to resolve the problem.

# System Description

The meeting coordinator system shall allow users to login in and then to add an event to the planner and check for conflicts with attendants’ existing events as well as if the chosen conference room is available. The system shall maintain the availability of conference rooms and update the planner of all attendants’ planner.

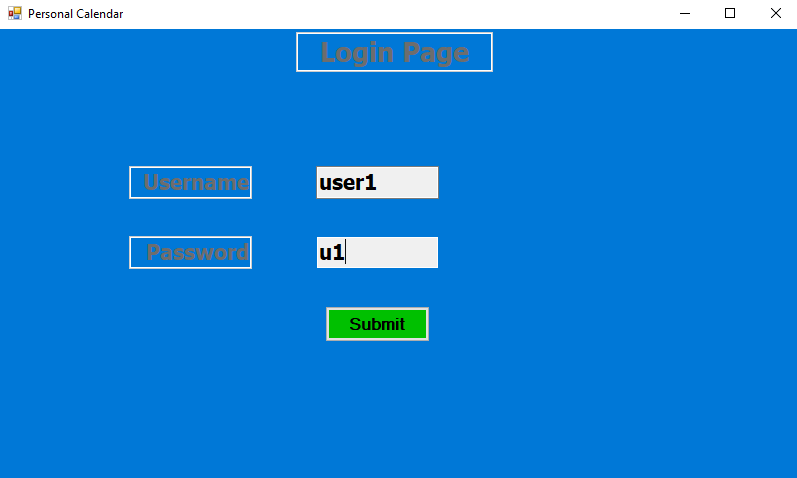
# System Requirement

## Functional Requirements

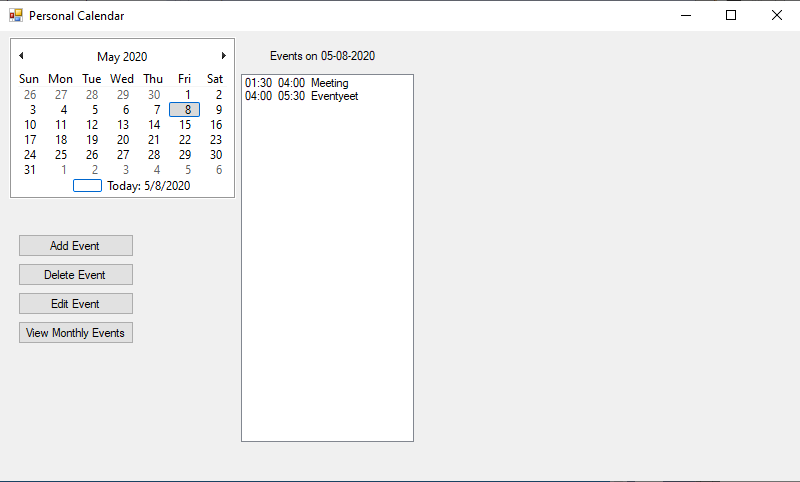
1. The system shall allow a user to keep a planner to record their schedule.
   1. The system shall display the login screen for the meeting coordinator.



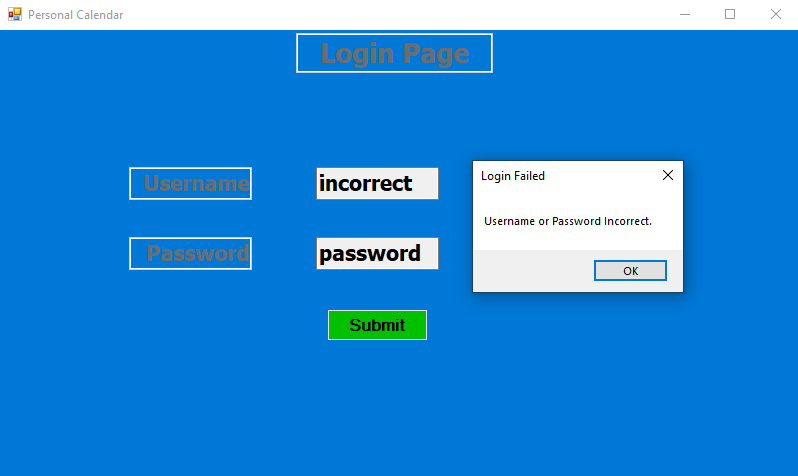
* 1. The user shall provider their company provided login data.



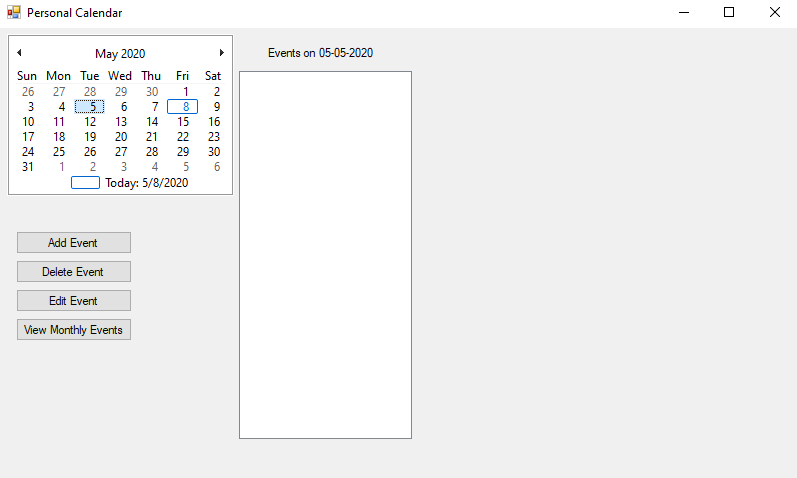
* 1. The user shall select the submit button.
     + 1. The system shall display the personal calendar screen if the username and password is correct.



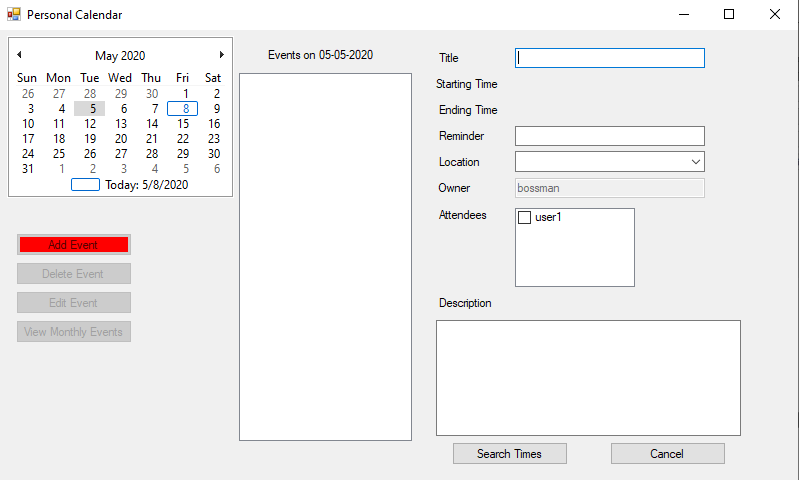
* + - 1. The system will display an error message if the username or password combination is incorrect.



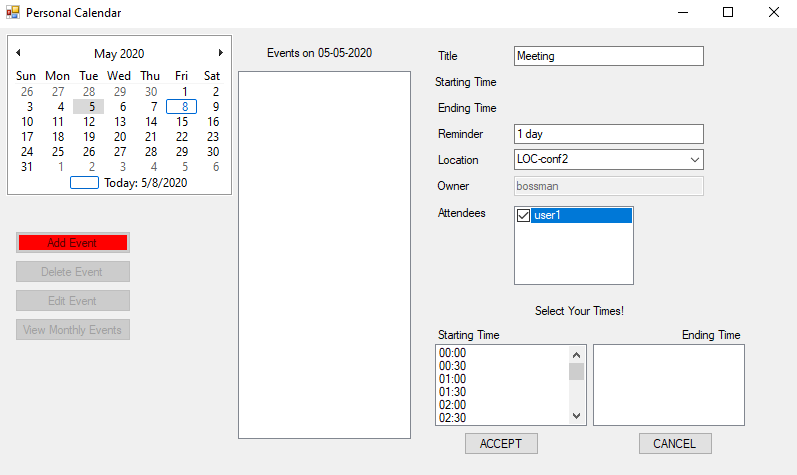
* 1. The user shall select a date from the monthly calendar to add the new event.
  2. The system shall highlight the selected date on the calendar.



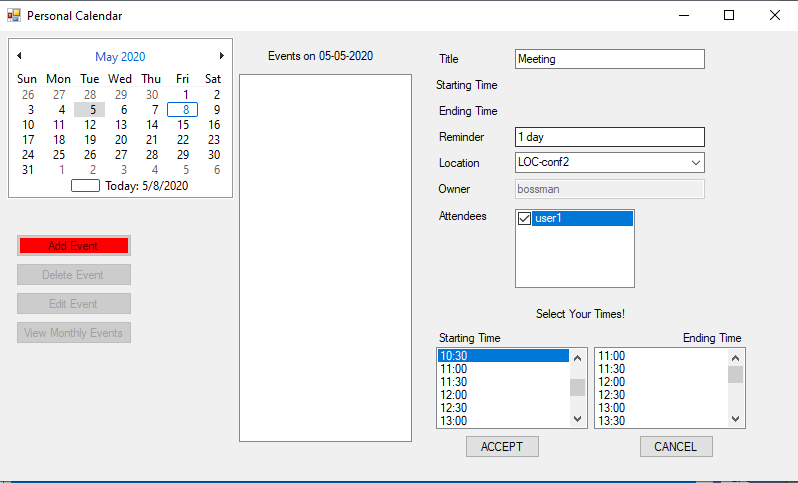
* 1. The user shall press the “Add Event” button.



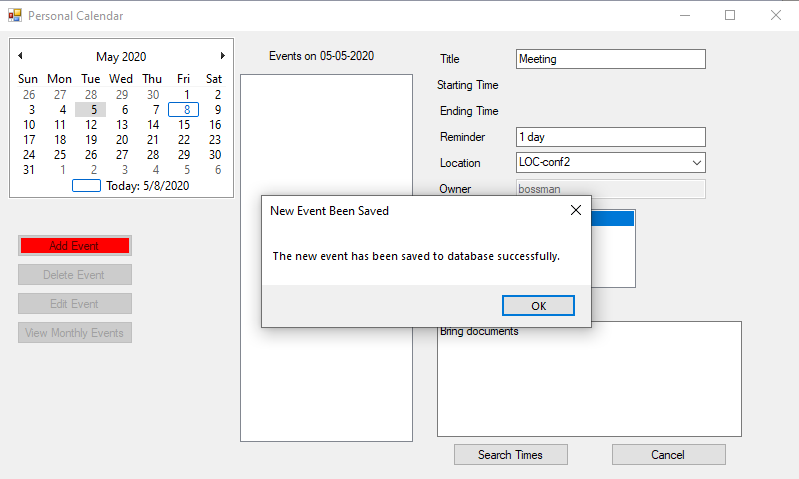
* 1. The system shall display a form to allow a user to enter the information of the new event, including event title, reminder time, location, and attendees.
  2. The user shall enter the data for the event information and press the “Search Times” button.
  3. The system shall check for starting times when the selected location and attendees are available.



* 1. The user shall select an available starting time.

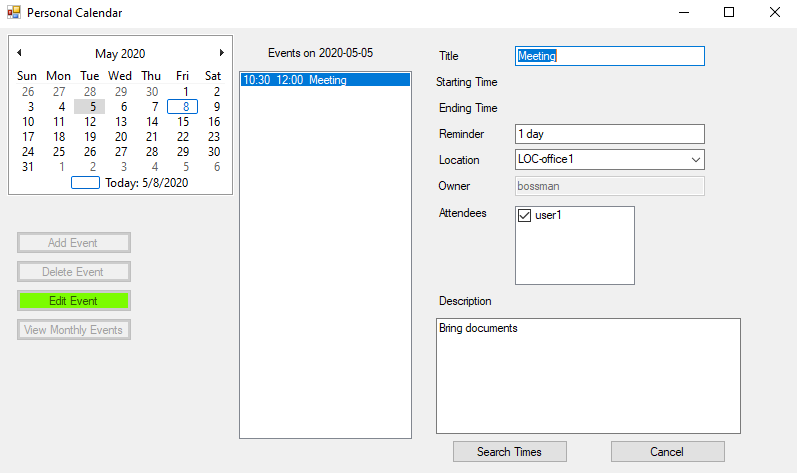


* 1. The system shall provide all available ending times.
  2. The user shall select one of the provided ending times.
  3. The user shall select the “Accept” button.
  4. The system will then attempt to save the event to the database.
     + 1. If the save is successful, then a message will appear stating the save was successful.

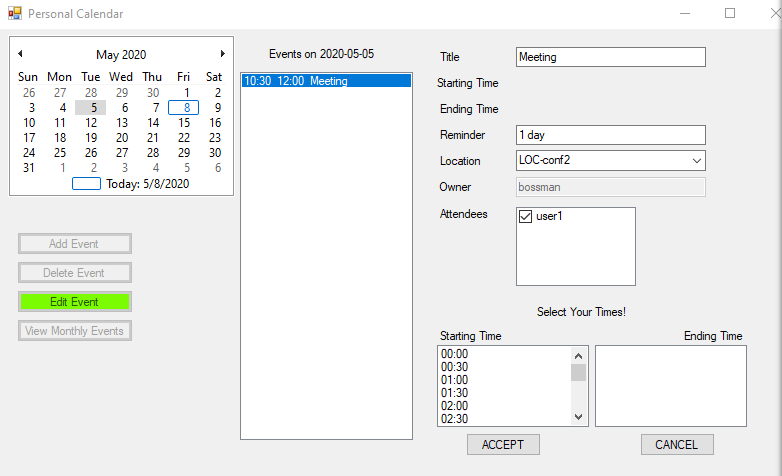


* + - 1. If there is an error with saving, the system will display an error message.

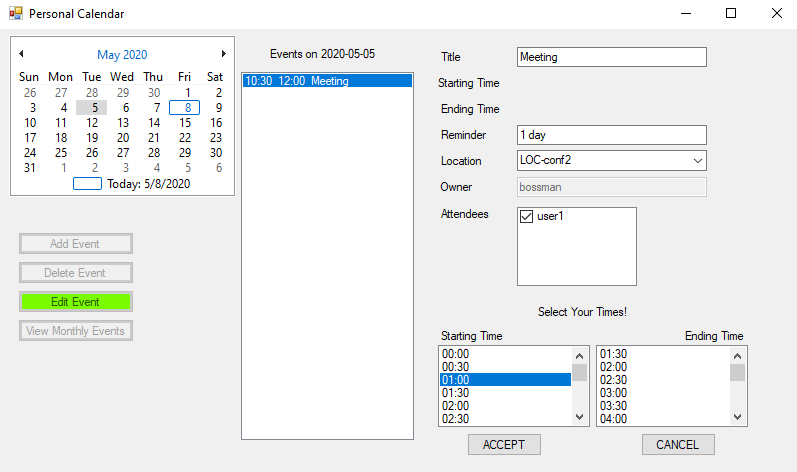
1. The system shall maintain the availability of conference rooms.
   1. The user shall select the “Add Event” or “Edit Event” button.



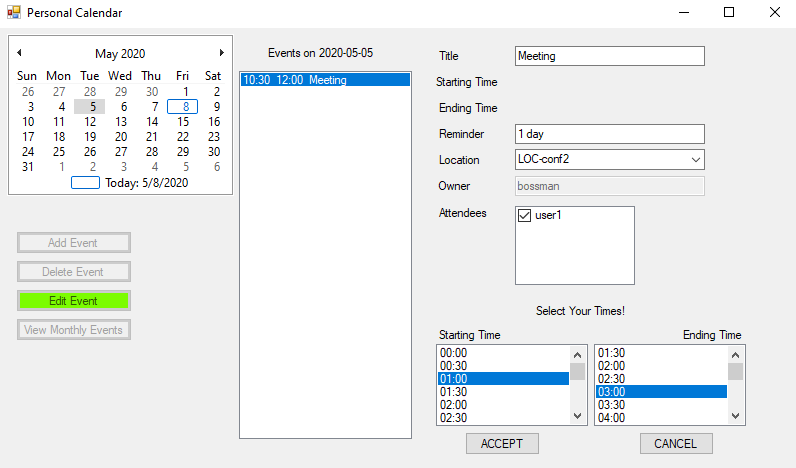
* 1. The user shall fill out the form and select the room they wish to reserve for their meeting.
  2. The user shall select the “Search Times” button.
  3. The system shall display a list of times that are available on that day for the selected attendees and during what times.



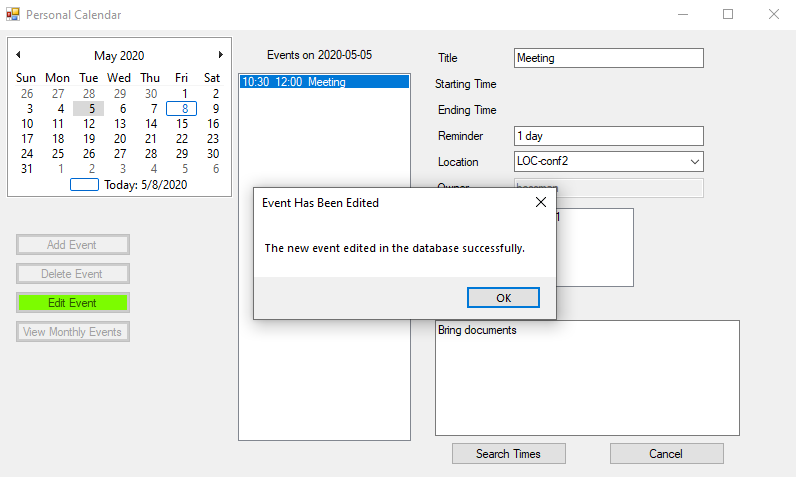
* 1. The user shall select their desired starting time.
  2. The system shall update the end times with available times.



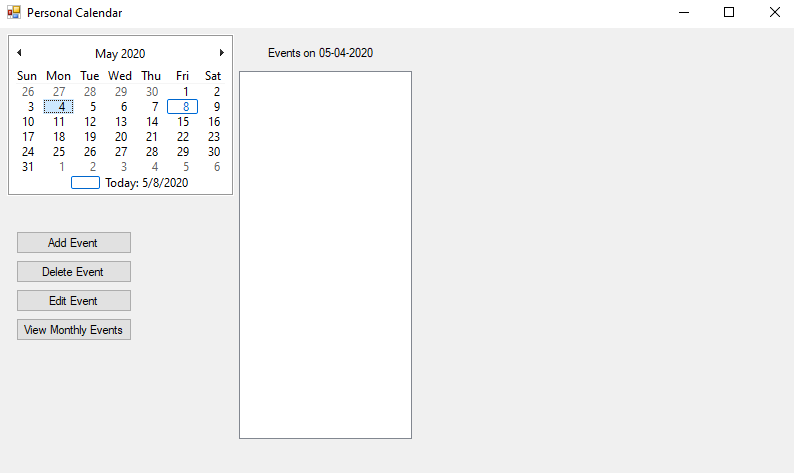
* 1. The user shall select their desired ending time for the room reservation.



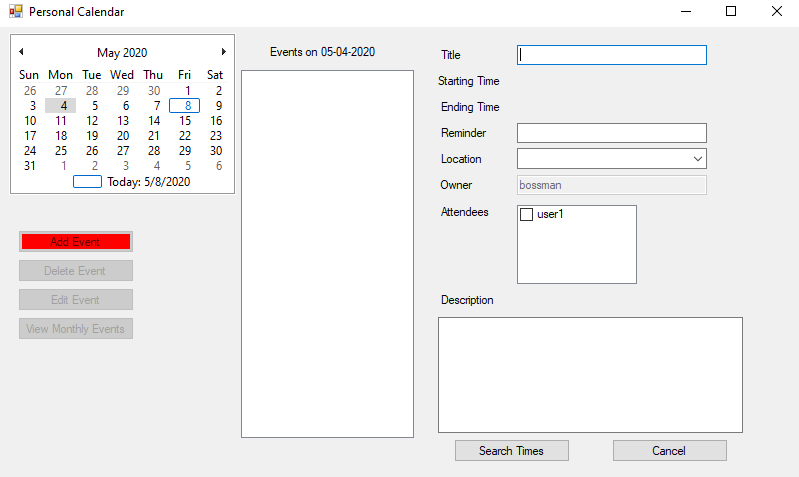
* 1. The user shall select “Accept” to confirm that room reservation.



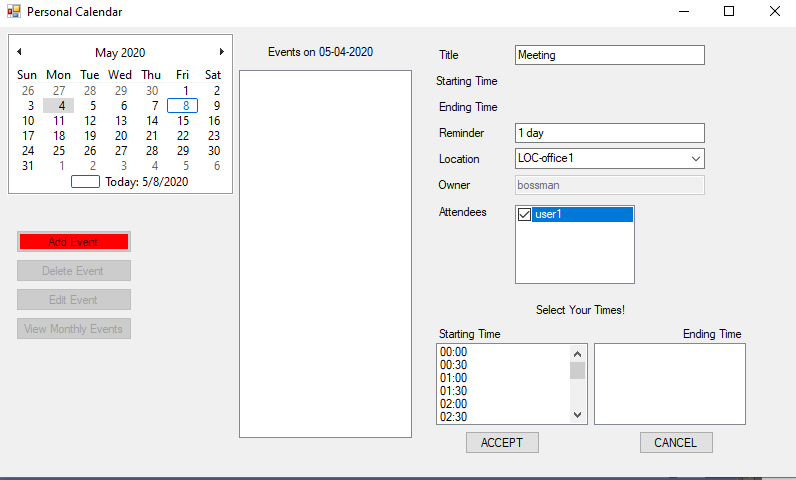
1. The system shall allow a user to coordinate meetings based on attendants’ planners and the availability of conference rooms.
   1. The user shall select a date from the monthly calendar.



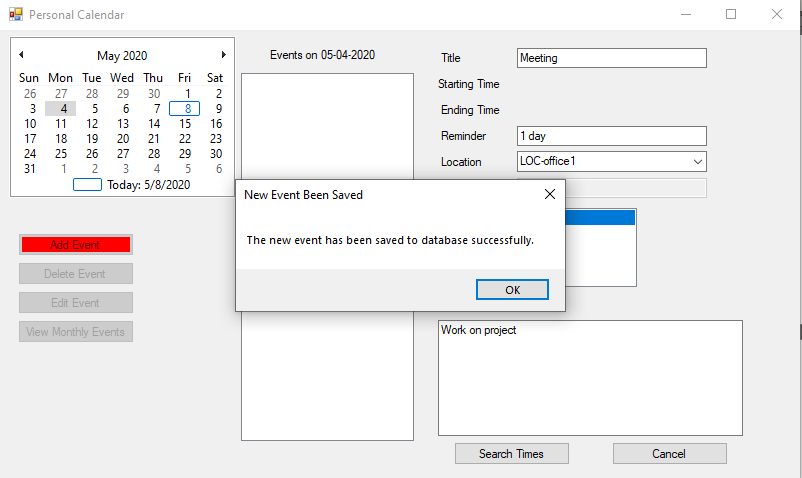
* 1. The user shall select the “Add Event” or “Edit Event” button.



* 1. The system shall display a form.
  2. The user shall enter the data for the form and select the “Search Times” button.



* 1. The user shall select their desired starting time.
  2. The system shall update with the available ending times.
  3. The user shall select their desired ending time.
  4. The user shall select the “Accept” button.
  5. The system shall check for available meeting times based on attendants’ planners and the availability of conference rooms.
     1. The system will confirm the event based on the selected date and times.

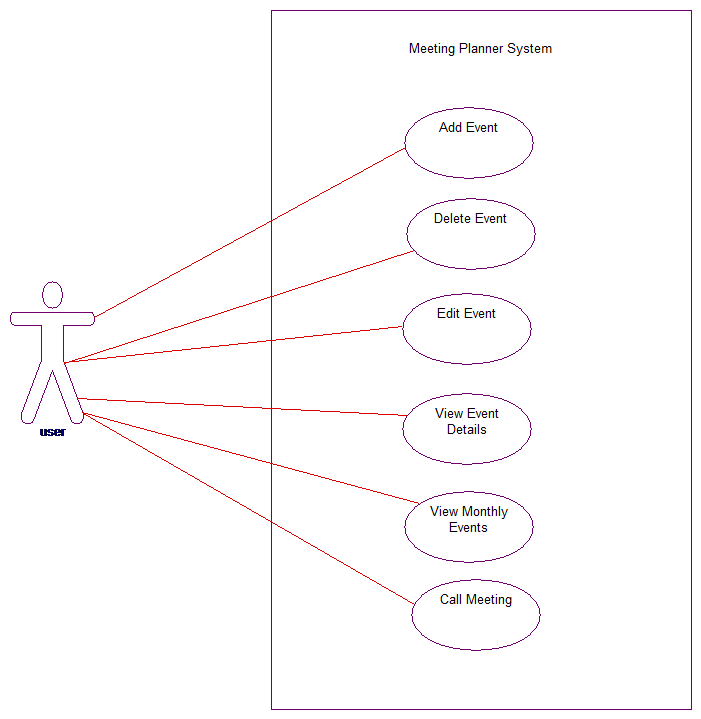


* + 1. If there is not an available time and room, the system shall display an error message and prompt the user to try a different date.

## Non-functional Requirements

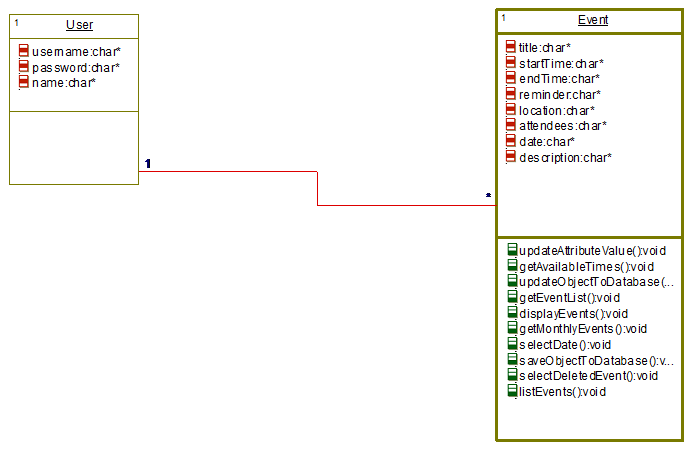
1. The system shall present information to the user in the English language.
2. The system shall use the Gregorian calendar.
3. Password must be greater than or equal to 8 characters.

# Use Case Diagram



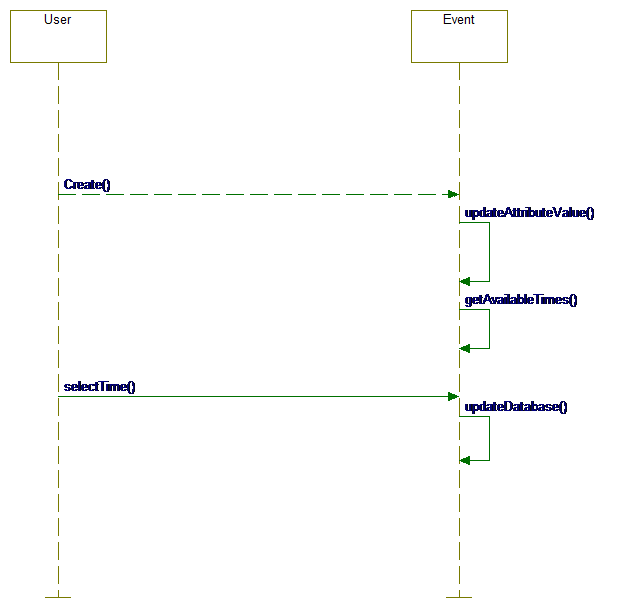
This diagram shows the expected behaviors a user is able to do in the meeting coordinator system. The user will be able to add, delete, edit, and view events. They will also be able to view monthly events and call a meeting for multiple attendees.

# Class Diagram

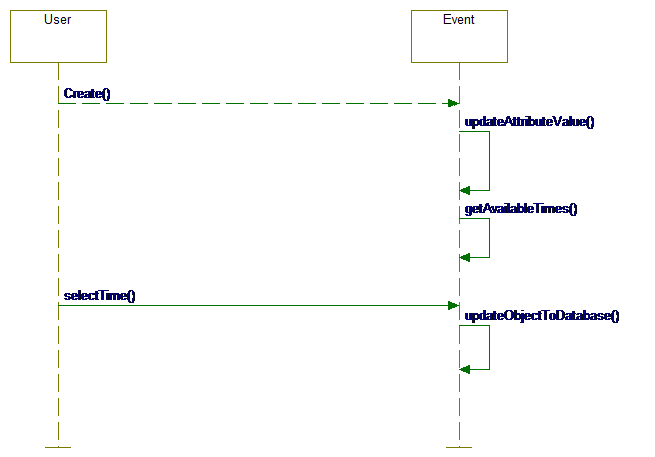


This diagram shows an overview of the system’s user and event classes. Inside of each class are the class’ operations.

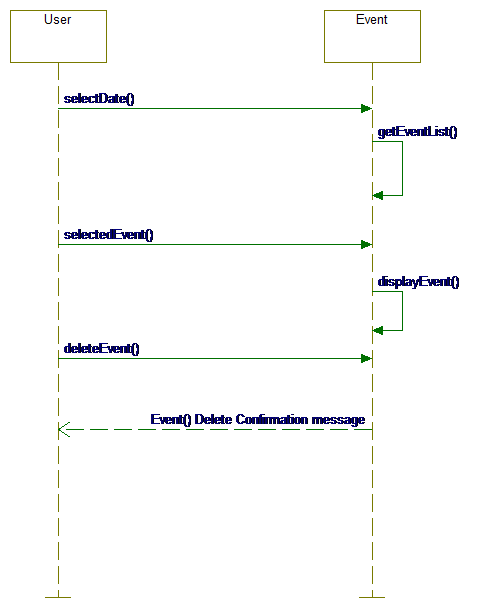
# Sequence Diagrams



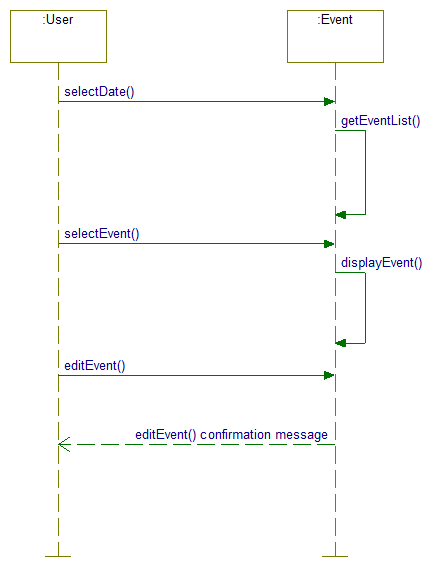
*Add Event* - This diagram shows the interactions that take place for a user to add an event to the meeting coordinator.



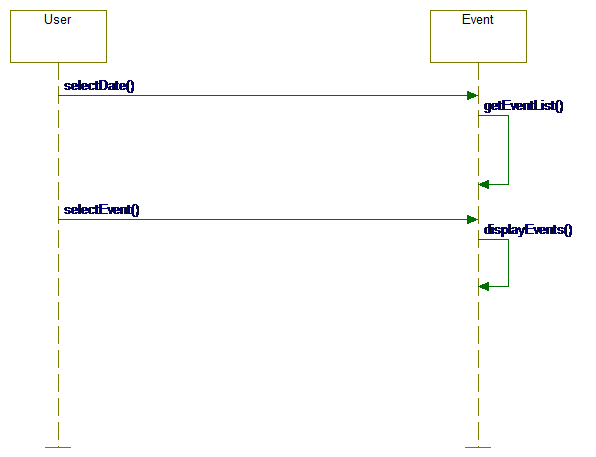
*Call Meeting* - This diagram shows the interactions that take place for a user to call a meeting and have the meeting display on the personal calendar of all users.



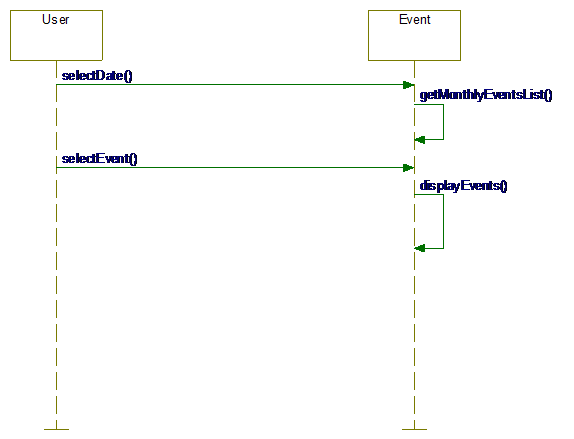
*Delete Event* - This diagram shows the interactions that take place for a user to delete an event from the users’ personal calendar and all attendee’s calendar.



*Edit Event* - This diagram shows the interactions that take place for a user to edit an event in the meeting coordinator for the user and all attendees of that event.

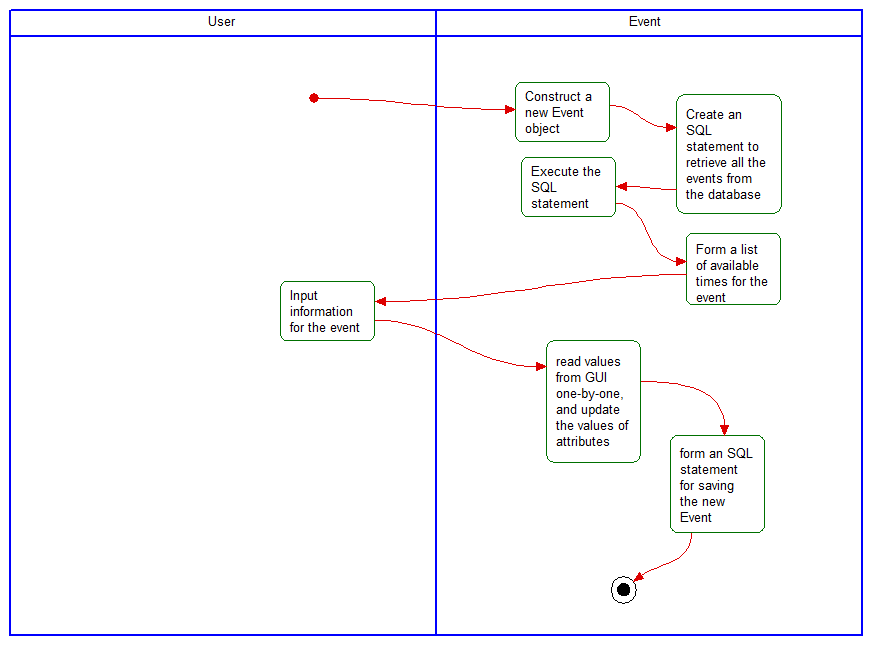


*View Event* - This diagram shows the interactions that take place for a user to view the details of an event that is on their calendar.

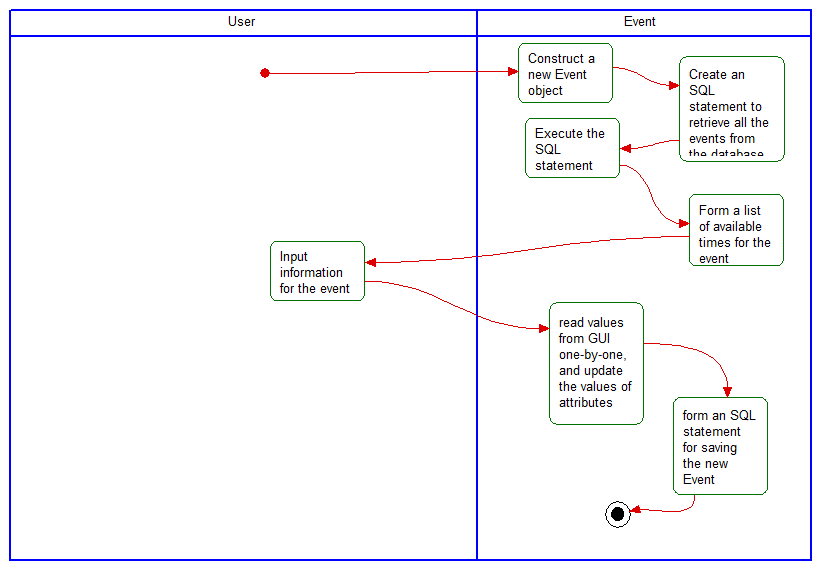


*View Monthly Event* - This diagram shows the interactions that take place for a user to view all events for the month that are on their calendar.

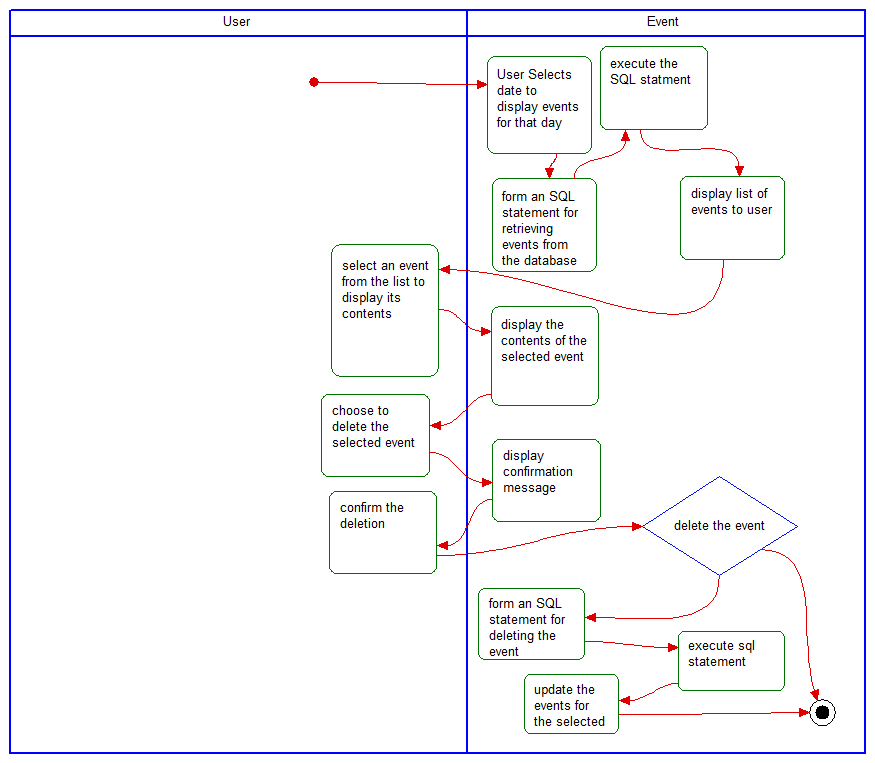
# Activity Diagrams



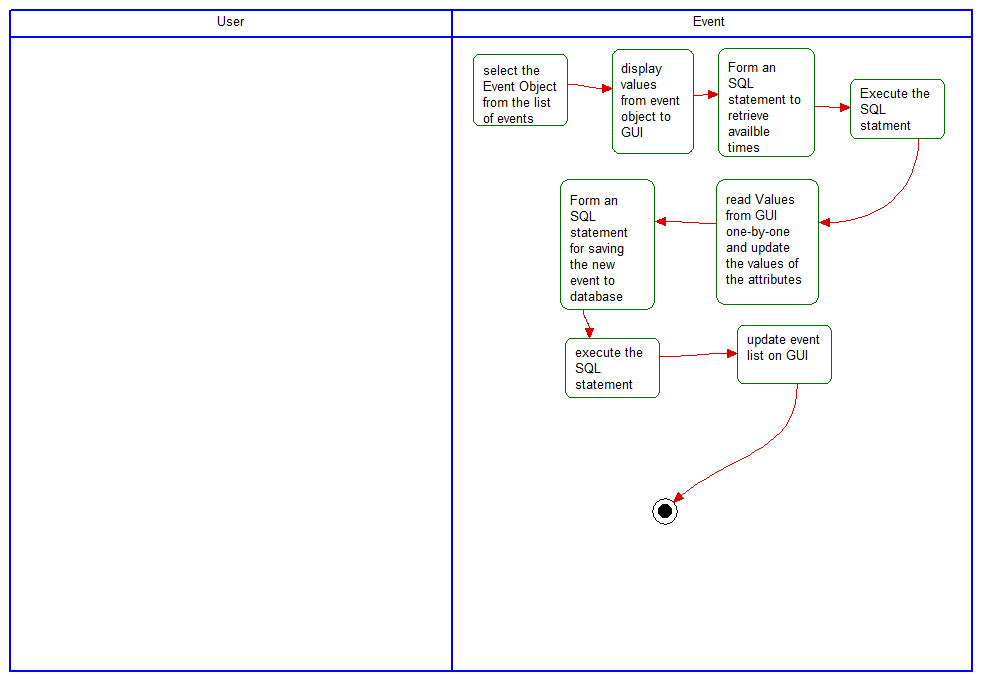
*Add Event* - This diagram shows the activity flow of the system when a user adds an event to the meeting coordinator.



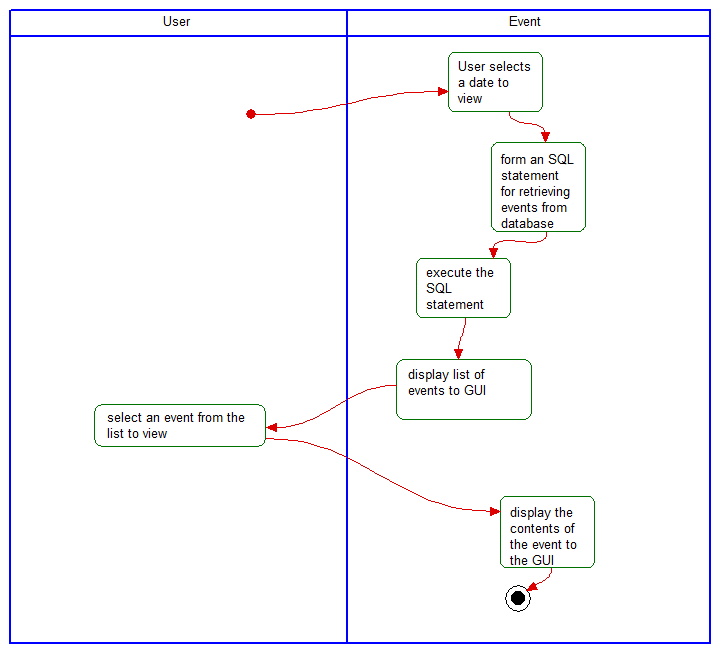
*Call Meeting* - This diagram shows the activity flows that take place to a call a meeting with the host user and the selected attendees at the selected location.



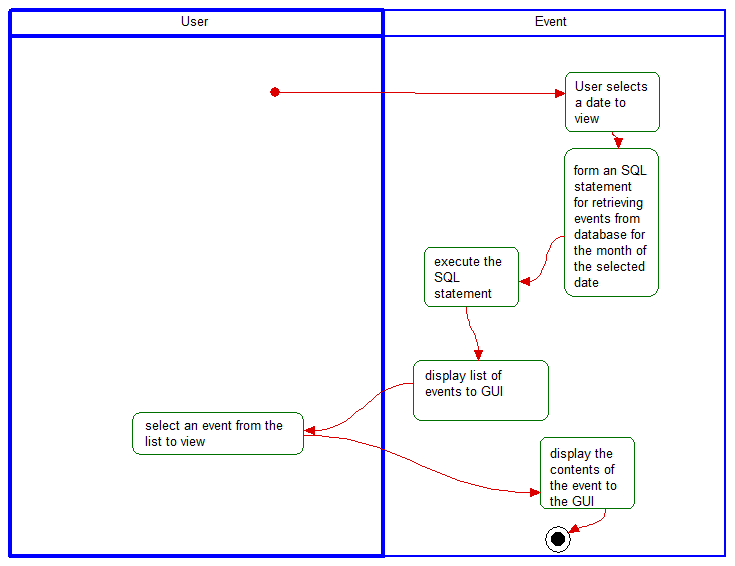
*Delete Event* - This diagram shows the activity flows that take place for a user to delete an event to from their calendar and all attendee’s calendar.



*Edit event* - This diagram shows the activity flows that take place for a user to edit an event on their calendar and all attendee’s calendar.

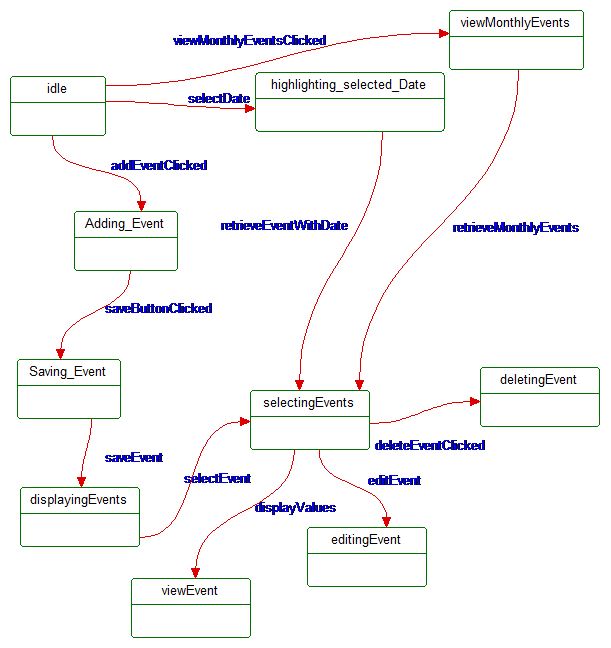


*View Event* - This diagram shows the activity flows that take place for a user to view an event that is on their calendar.



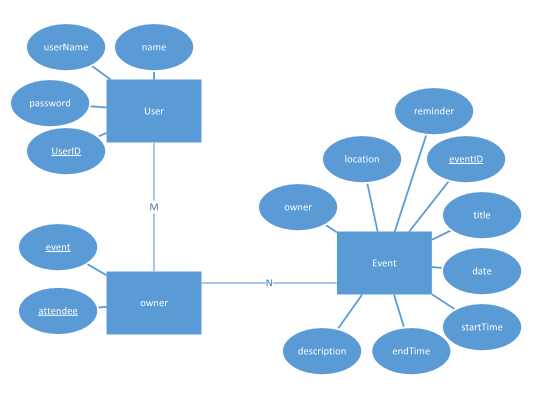
*View Monthly Events* - This diagram shows the activity flows that take place for a user to view all events for the month that are on their calendar.

# State Diagram



This diagram displays the state of the meeting coordinator system during a specific user action or system behavior.

# Database Design



User

|  |  |  |  |
| --- | --- | --- | --- |
| userID | name | userName | password |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| eventID | title | date | startTime | endTime | reminder | description | location | owner |

Event

Owner

|  |  |
| --- | --- |
| event | attendee |

# Conclusion

The meeting coordinator calendar system created allow employees at ABC, Inc to coordinate meetings based on attendants’ planners and the availability of conference rooms.

# Data Dictionary

Attendee – A different user than the host. This is a user that is added to attend a new event by the host.

Conflict – In a calendar system, this means that either the start time or end time coincides with another event or the start time takes place after the end time. This could also be because an attendee has an event that conflict with the new event or the selection location is unavailable.

Database - an organized collection of data, generally stored and accessed electronically from a computer system.

Event – added by a user. These have a date, start time, end time, title, location, and description. These are tied to a specific user. A location and attendees may be added to an event.

Gregorian calendar – a solar calendar with 12 months of 28–31 days each. A regular Gregorian year consists of 365 days, but in certain years known as leap years, a leap day is added to February.

Login data – the username and password provided to employees by the company.

User – a person that has events and is used to check the availability of events. They have a username, password, and name.

Host – The user that is logged in to the meeting coordinating system. When setting up a new event, the user that sets it up is classified as the host.

Owner – the person who created the event. See host.

State – specific instance in time