Meeting Scheduling System
Project Analysis and Development Plan
Version <1.0>

TEAM MEMBERS

Name	Student ID
AbdAllah Mahassen	akm6518
Chin Shiang Jin	sjc6393
Caralyn Harben	clh6036

Revision History

Date	Version	Description	Author
04/7/2021	<1.0>	Submission	Chin Shiang Jin, Caralyn Harben, AbdAllah Mahassen

Table of Contents

Project Analysis and Development Plan

1. Introduction

The purpose of this Software Requirement Specification (SRS) document is to collect, analyze, and define the high-level needs and features of the Meeting Scheduling System (MSS) for PennStateSoft. This document focuses on the capabilities needed by the stakeholders and the clients of this system and why these needs exist. The document incorporates the quality and security requirements in developing the software application.

1.1 Purpose

The purpose of this SRS document is to define the requirements and specifications of the Meeting Scheduling System gathered from the client representative. It also serves as the communication reference between the client, PennStateSoft, and the development team. This document also aims to support the development team by keeping the specific details of all aspects of the MSS system and provide an outline for the development plan.

1.2 Scope

The scope of this SRS document is limited to the documentation required in the development of the MSS project. Content of this SRS document includes the project purpose, scope, and objectives, specific functions available to the client and administrator account, description of the target user, assumptions and dependencies for this project, constraints for the project as well as a list of specific requirements to be fulfilled. This document also lists out the specific details of the main use cases, incorporating the quality and security requirements with associated vulnerability mapping and mitigation plan.

1.3 Definitions, Acronyms, and Abbreviations

Acronym	Definition
Admin	Administrator
HTML	HyperText Markup Language
IP	Internet Protocol
MSS	Meeting Scheduling System
SRS	Software Requirements Specification
W3C	World Wide Web Consortium
UI	User Interface
URL	Uniform Source Locator

1.4 References

No external documents were referenced by this document.
The team had a meeting with the client representative Dr. Naseem Ibrahim on Wednesday, 23rd June 2021 for clarifications of the requirements.

1.5 Overview

The rest of the document is organized in the following manner:

1.5.1 *Section 1: Introduction*
This section serves to be an introduction and overview to the SRS document. It provides a general layout of what will be discussed in other parts of the document. It aims to guide others to understand the scope of the project. It also defines important acronym terms used in the project.

1.5.2 *Section 2: Problem Description*
This section starts with the project's purpose, scope, and objective. The next subsection lists the product functions available to the client and administrator account separately. Subsection 2.3 describes the target user. Subsection 2.4 lists the assumptions and dependencies for this project. Subsection 2.5 lists the constraints to be adhered to by this project. Subsection 2.6 lists the specific requirements for this project which include the applicable standards, system requirements, performance requirements, environmental requirements, and quality requirements.

The analysis model for each of the use cases and the associated vulnerability mapping and mitigation plan are detailed out in subsection 2.7.

1.5.3 *Section 3: Team Member Log Sheet*
This section keeps track of the hours committed to this SRS document and each team member's contribution to the SRS document.

2. Problem description

2.1 Project Purpose, Scope, and Objectives

The purpose of this project is to create a web-based meeting scheduling system (MSS) for PennStateSoft, a game development company located in Pennsylvania. This MSS will keep track of the meetings schedule, meeting participants, and the room used for meeting for their employees. The system should have two types of user accounts, the client account, and the administrator account. The specific functions available to the clients and administrators are outlined in section 2.2.

This MSS application is required to maintain the meeting schedule created by PennStateSoft employees for the upcoming working week only for meeting within traditional 9-5 business hours. In addition to the specific functions outlined in section 2.2, the system is required to perform the following checks: A person cannot be attending more than one meeting in any one-hour slot; a room cannot hold more than one meeting at any one-hour slot. Besides that, this MSS needs to enable two types of rooms, the special room, and the regular room. This MSS needs to support credit card processing to reserve the special room.

2.1.1 User interfaces

This subsection describes the logical characteristics of each user interface (UI) between the MSS system and its users. In general, all the UI should display the company's logo, have a minimum font size of 12, and a good contrast between the font and the background to enhance the readability of the page content to the user.

When the user first enters the website URL of the MSS system, the system should display only the login interface. The user can choose to register a new client account using their company email address, log in as the client user, or log in as the administrator (admin) user.

The user interfaces for the client user will consist of several pages accessible by tabs. There will be a "Create Meeting" UI page for the client to create a new meeting. There will be a "Profile" page for the client to edit or update their profile information. There will be a "Manage Meeting Overview" page for the client to view the meetings created by them and edit the information of the meetings. There will also be a "My upcoming Meeting" page for the client to view all upcoming meetings they have been invited to join. There will be a "File a Complaint" page also for the user to file their complaints about the system.

The user interfaces for the admin user will consist of several pages which are accessible by tabs. This also includes a page where the admin can access the administrative functions. There will be a 'Room Management' page for the admin to add or delete rooms available for the meetings. There will be a 'Client Account Management' page for the admin to view the list of clients registered for the system, and update the client's billing information. There will be a 'Complaint Management' page for the admin to view all the unresolved complaints and respond to them. There will be a user interface for the administrator to display all the meetings in a week or a day, display all the meetings in a specific room, display all the meetings attended by a single person, or display all the meetings at a specific time slot. There will be an 'Admin Management' page for the administrator to create other administrator accounts.

2.2 Product Functions

The following sections outline the product functions of the MSS system incorporating the quality and security requirements. The product functions are categorized into two categories: function available to the normal client and function available to the administrator.

2.2.1 Functions available to Client

2.2.1.1 Create Account

The system shall allow a client (employee of the company) to create a new client account using their unique company email address. In the create account user interface page, the system shall allow the user to enter their company email address, desired username, and password. After the user clicks the create account button, the system shall perform two checks on the company email address entered. The first check will ensure the company email address entered exists in the database. The second check is to ensure that there is no client account associated with the company email address that was registered before. In both cases, if the validation fails, the system shall display an error message stating the cause of the error and stop the process of creating a new account.

Once the email address is validated, the system shall proceed to check the username entered to confirm it consists of simple text only, within the acceptable length, and not registered before. Afterward, the system shall proceed to check the password entered, to confirm it consists of alphanumeric characters between 6 to 10 characters in length. If the username or password check fails, the system shall display the error message stating the cause of the error and stop processing the request.

If the company email address, username, and password all passed the validation checks, the system shall proceed to store the new account information in the system database.

2.2.1.2 Check and Update Profile

The system shall allow the client to check and update their profile information in the profile user interface. The system shall allow the following information to be entered or updated: the client's full name, address, contact number, job title, and short biography, as well as their credit card information used for payment. Entering the profile information is not mandatory. The system shall keep all the profile information including the payment information accessible to the client and admin users only.

Besides, the system shall allow the client to change their password in the profile user interface. After the client clicks the button to change their password, the system shall show a pop-up window prompting the client to enter their old password and entering the new password twice. The system shall validate that the old password entered by the user matched the record in the database, the new password entered twice matched, and the new password consists of alphanumeric characters between 6 to 10 characters in length.

2.2.1.3 Create a meeting and book a room

The system shall allow the client to create a new meeting using the create meeting UI page. The system shall allow the client to enter the title of the meeting and select a time slot from the calendar. The system should only allow the user to select a timeslot within the traditional 9 AM – 5 PM business hours from Monday to Friday for the upcoming week. The timeslot can be selected in the number of hours, like one-hour block, two-hour block, etc. Once the client selects the time slot, the system shall display the list of rooms available for reserve. The system shall allow the client to add participants to the meeting on the same page. The function to add participants will be further described in a later section.

After the client clicks the submit meeting request button, the system shall check to ensure that the client reserved a room, stop, and prompt the client if they have not. If the room reserved by the client is the special room, the system shall proceed to the payment page for the special room which will be described in subsection 2.2.1.4. After all validation checks are passed, the system shall proceed to store the meeting information in the database. The system shall book the room reserved by the client by saving this information in the database as well.

2.2.1.4 Pay for special room

During the create meeting stage, the client may choose to reserve a special room. If they do so, when they click the submit meeting request button, the system shall bring them to the payment confirmation page, where the clients need to confirm they will be paying $100 to reserve the room using their profile payment information (stored credit card information). If there is no credit card information stored, the system shall prompt the user to enter new credit card information. After the user hit the confirm payment button, the system shall proceed to process the payment and registered the booking of the special room in the database.

2.2.1.5 Manage the participants of the meeting

As described in section 2.2.1.3, the client can add the meeting participants during the create meeting stage. Alternatively, the system shall allow the client to add the meeting participants when reviewing the meeting created by them. When adding the participants, the system shall allow the client to search for and select the participants who already registered as a client to the system. If the participant's name entered by the client is not registered, the system shall display an error message and reject the addition operation.

During the addition of participants, the system shall check if the selected participants are joining another meeting with the same time slot. If the selected participants are already joining another meeting, the system shall display an error message as well and reject the addition operation. If the selected participants passed the validation check, the system shall register the participation information in the database.

The system shall allow the client to remove the meeting participants when reviewing the meeting created by them. After the client selected participant(s) to remove and clicked the submit button, the system shall proceed to update the participation information in the database.

2.2.1.6 Display and edit all the meetings they have created.

The system shall allow the client to display and edit all the meetings they have created in the 'Manage Meetings' user interface. The system shall allow the client to edit the meeting name, select another room, select another time slot, add, or remove participants, as well as cancel the meeting. The system shall apply the same validations as the create meeting function when the client wants to select a new room, pay to reserve a special room, and add participants to the meeting.

2.2.1.7 Display all the meetings they are going to participate in

The system shall allow the client to view all the meetings they are going to participate in for the upcoming week in the 'View My Meetings' user interface. The system shall list the summary information of the meeting invites in a weekly calendar view, which displays the meeting title and room ID in the meeting time slot. The system shall enable the client to click the meeting invites and view additional details of the meeting, this includes a description of the meeting room location, the creator of the meetings, and a list of participants.

The system shall not allow the client to view other meetings they are not a participant in. The system shall not allow the client to remove themselves from the meetings as well.

2.2.1.8 File a complaint

The system shall allow the client to file a complaint through the filing complaint user interface. The system shall collect the title of the complaint and allow the client to enter the detailed descriptions of the complaint within 500 words. Once the complaint is submitted by the client, the system shall automatically generate a complaint id for client reference. The system shall keep unresolved complaint(s) in the client's filing complaint interface, and only remove them after the complaint(s) have been resolved by an admin.

2.2.2 Functions available to Admin

2.2.2.1 Add or Delete Rooms Available

The system shall allow the admin to add or delete rooms available for meetings from the room management user interface. When adding a room, the system shall allow the admin to enter the ID of the room, where the system shall validate the room ID to be unique and not used before. The system shall allow the admin to enter further details about the new room such as the room capacity, location description, and other facility descriptions of the room. The system shall allow the admin to designate the room as a special room as well. After the admin hits the add room button, the system shall perform the required validation(s) and add the room to the database if passed.

When the admin hits the delete room button to delete a selected room, the system should first check whether the room has any upcoming meeting scheduled in the room. If there is an upcoming meeting scheduled for the room, the system shall display an error message and reject the deletion request. If there are no upcoming meetings scheduled for the room, the system shall proceed to remove the room from the database.

2.2.2.2 Update Client Account Information (including update client's billing information)

The system shall allow the admin to view the list of all client accounts from the 'Client Account Management' page. The system shall allow the admin to edit the information on the client's profile, this includes updating the client's billing information.

2.2.2.3 View complaints and respond to them

The system shall allow the admin to view the list of unresolved complaints from the complaint management page. The system shall allow the admin to respond to the complaint by contacting the client via email. The system shall allow the admin to click complaint resolved only after they entered the complaint resolution remarks.

2.2.2.4 Display all meetings in a week or a day (selected timeframe)

The system shall allow the admin to view all the meetings in the selected timeframe (a day or a week through the meeting overview page. The system shall list all the titles of the meetings in the respective time slot on the calendar view of the week. The system shall allow the admin to view further details of the meeting, including the creator of the meeting, the room reserved for the meeting, and the participants of the meeting by clicking the meeting title.

The system shall allow the admin to select a day of the week to view all meetings in the day. The layout of the view by day should be like the layout of the view by week.

2.2.2.5 Display all meetings at a specific time-slot

In addition to view by week or view by day, the system shall allow the admin to click a specific time slot in the calendar view to display all the meetings occurring at that specific time. The system shall list the title of all meetings occurring at the selected time slot, where the admin shall be able to view further details of the meeting by clicking the title.

2.2.2.6 Display all meetings in a specific room

The system shall allow the admin to display all meetings in a specific room in a calendar view. The system shall allow the user to select the room from a list of all the rooms available for the meeting. The system shall list the title of the meetings for each time slot in the calendar, and allow the admin to view further details of the meeting by clicking the title of the meeting.

2.2.2.7 Display all meetings attended by a single person

The system shall allow the admin to display all meetings attended by a single person in calendar view. The system shall allow the admin to search for the client using a username or email address. The system shall list the title of the meetings for each time slot in the calendar for the specific user and allow the admin to view further details of the meeting by clicking the title of the meeting.

2.2.2.8 Create another administrator account

The system shall allow the admin to create another administrator accounts through the admin account management page. The function to create an admin account is similar to the function to create a client account. In the create account user interface page, the system shall allow the admin to enter the company email address, chosen username, and password for the new administrator. After the admin clicks the create account button, the system shall perform the two checks on the company email address entered. The first check will ensure the company email address entered exists in the database. The second check is to ensure that there is no admin account associated with the company email address that was registered before. In both cases, if the validation fails, the system shall display an error message stating the cause of the error, and stop processing the user input further.

Once the email address is validated, the system shall proceed to check the username entered to confirm it consists of simple text only, within the acceptable length, and not registered before. Afterward, the system shall proceed to check the password entered, to confirm it consists of alphanumeric characters between 6 to 10 characters in length. If the username or password check fails, the system shall display the error message stating the cause of the error and stop processing the request.

If the company email address, username, and password all passed the validation checks, the system shall proceed to store the new administrator account information in the system database.

2.3 User Description

2.3.1 User Environment

PennStateSoft is currently employing over 100 people. It is unknown whether the company wants to expand significantly over the next few years, but the system is expected to handle up to 1000 users.

The amount of time required to create a meeting or view and update a meeting should be done in less than half an hour.

All employees of PennStateSoft are working in the company's main office in Pittsburgh, PA. It is expected that they have access to the company's laptop when they want to make the booking from their working desk.

The system is expected to be hosted on the MySQL database online, and the user interface webpage is also hosted on the same web-hosting service with the database.

There is currently no requirement to integrate the MSS with other applications used by the company.

2.3.2 User Profiles

There will be two types of target users for the system, namely the client user and the administrative user. Both user groups will be the employees of the company. The administrator users are not full-time positions, rather they are some of the employees who are assigned by the IT department carrying out this part-time role. Hence the administrator users are not expected to be tech-savvy or expert in coding to use the system.

Both user groups are expected to have a college education which equips them with the common reading, writing, and comprehension abilities. Both user groups are expected to be familiar with information technology and tools such as browsing a website, using their mobile phone to surf the internet, and so on. Although there might be potential that some of the employees may be visually impaired, the system is not expected to support people with visual impairment at the current phase. However, the system will need to be able to support people with color blindness.

2.4 Assumptions and Dependencies

The following are the lists of assumptions and dependencies for this project.

1. It is assumed that the target end-users for the application are computer-literate, familiar with web-browsing activities, and able to perform form-entry operations.
2. It is assumed that the system will only need to handle up to 1000 user records in the database.
3. It is assumed that there will be only up to 100 users perform the concurrent operations with the system.
4. It is assumed that the users will safely keep their login credentials, hence there is no need to identify if a suspicious IP address is used for login.

2.5 Constraints

To meet the customer's needs, a functional version of this software must be made available to the customer by August 13th, 2021. A set budget of $20,000.00 must be honored throughout the production of this application. Further, the team producing this application will be limited to three (3) software engineers working asynchronously. Design and implementation of the application shall be done within 80 hours per individual (2 40-hour work weeks).

Finally, to ensure the best possible experience when in use, the system must be able to function in environments with slow internet speeds (see Section 2.6.2 for specific speed requirements). Even in an environment with an unstable internet connection, the system should attempt to book the meeting by waiting for the connection to stabilize once the user requests the meeting creation.

2.6 Specific Requirements

2.6.1 Applicable Standards

The system shall adhere to W3C's recommendations for Web Design and Applications. The webpage will be built using HTML, PHP, and JavaScript language. The back-end database should use MySQL. Communication between the user's browser and the webpage should use TCP/IP.

2.6.2 System Requirements

The following are the minimum specifications required to be able to use this system:

Processor	2 GHz
Memory	2 GB RAM
Operating System	Mac OS X 10.6 and newer, Windows 7 and newer, Linux (?)
Browser	Chrome 50 and 51, Safari 8, Firefox 45 and 46, Internet Explorer 11 and Edge
Internet Speed	512 kbps minimum
Additional Software	JavaScript required

2.6.3 Performance Requirements

For the best user experience possible, the system should take no longer than 5 seconds to complete a request to store a meeting upon the user clicking the "create meeting" button. Login requests should take no longer than 3 seconds to complete. To ensure that meetings are accurately booked on a first-come, first-served basis, requests should be queued and dequeued in order of the time that the request was made.

Further, failures to contact the database server should be limited to 1-2 failures per 1000 requests (a 1-2% failure rate). The system should be accessible for the duration of the workweek; if any maintenance must be done, the system should only go down on weekends to avoid any gaps in accessibility.

2.6.4 Environmental Requirements

While the system will perform best when a fast, stable connection to the internet is available, the system should be able to store requests temporarily if the internet connection drops for a short period. Once the connection is recovered, the system should attempt to finish processing the last request made.

If the system is being heavily used (for example, if a large number of users attempt to book meetings on a Monday morning), the system should be able to handle a large number of requests through double buffering (filling one buffer with requests while fulfilling requests in a separate buffer, then swapping the buffers and fulfilling the new requests received as needed). If a large amount of requests results in the page crashing, the system will attempt to save submitted requests to be completed upon recovery. Recovery time should be no more than 30 minutes.

In addition, site errors should be communicated to the user through a web page that provides an error code, a description of the error, and an option to submit the error to the administrator.

2.6.5 Quality Requirements

Correctness
The system shall be free from defects in its specification, design, and implementation, and perform the exact tasks as defined by the system specifications.

Reliability
The system must be available and operational throughout the workweek (9 a.m. - 5 p.m. Monday - Friday), with any needed maintenance scheduled on weekends. In the event of an urgent issue occurring, maintenance can also be scheduled after the workday (after 5 p.m.). Any errors with regards to database updates and requests when creating and viewing meetings shall only occur once for every 1000 requests **at most**.

Efficiency
As previously stated, the system should take no longer than 5 seconds to complete a request to store a meeting upon the user clicking the "create meeting" button. Login requests should take no longer than 3 seconds to complete. Hence the system should be able to handle concurrent usage of up to 100 users and storing records for up to 1000 users.

Integrity
To ensure that the user's data is secure and that no unauthorized changes to the database can be made, input validation should be used for all user input. Only users marked as having administrative privileges should be allowed to perform administrative requests on the system. Further, all new accounts should be sent a confirmation email to make sure that meetings can only be created by confirmed company employees.

Usability
The client should be able to use the system without training. The site itself must be easily navigable, with all necessary features being quickly accessible through a menu bar and easily readable buttons on-screen. Hints and help information should be readily available for the client and administrators. Additional training to be able to perform administrative tasks should take no longer than 2 hours to complete.

Maintainability
If maintenance must be done on the system, an error log should be made available to all system administrators for troubleshooting purposes. In addition, administrators should be able to report potential bugs to the developers through an easily accessible form. The code itself should be modular to minimize testing complexity and shorten turnaround for hotfixes.

Flexibility
The previously mentioned modularity of the code itself shall allow the system to easily be adjusted and built upon to add or remove features as the client requests. Further, the system shall be able to be used on all common operating systems described in section 2.6.2 of this document should the software need to be available on multiple platforms.

Testability
The modularity of the system should allow each "module" to be tested on its own both before and after it is integrated into the rest of the system.

Portability & Reusability

This system should be able to fulfill the needs of the customer with minimal change to the structure of the system for at least 10 years. Small additional features and maintenance shall be performed throughout the life of the system to prolong the use of the system and to maintain modernity.

2.7 Analysis model and Vulnerability Mapping

2.7.1 Extended Use Case Diagram with Misuse Management for Normal Client

The following two figures show the extended use case diagram for use cases available to the normal client. Each of the use cases is described in the followings sections 2.7.2 to 2.7.10. The figures also outline the potential threats that can be exploited by the attackers and how the system will respond in managing the misuse.

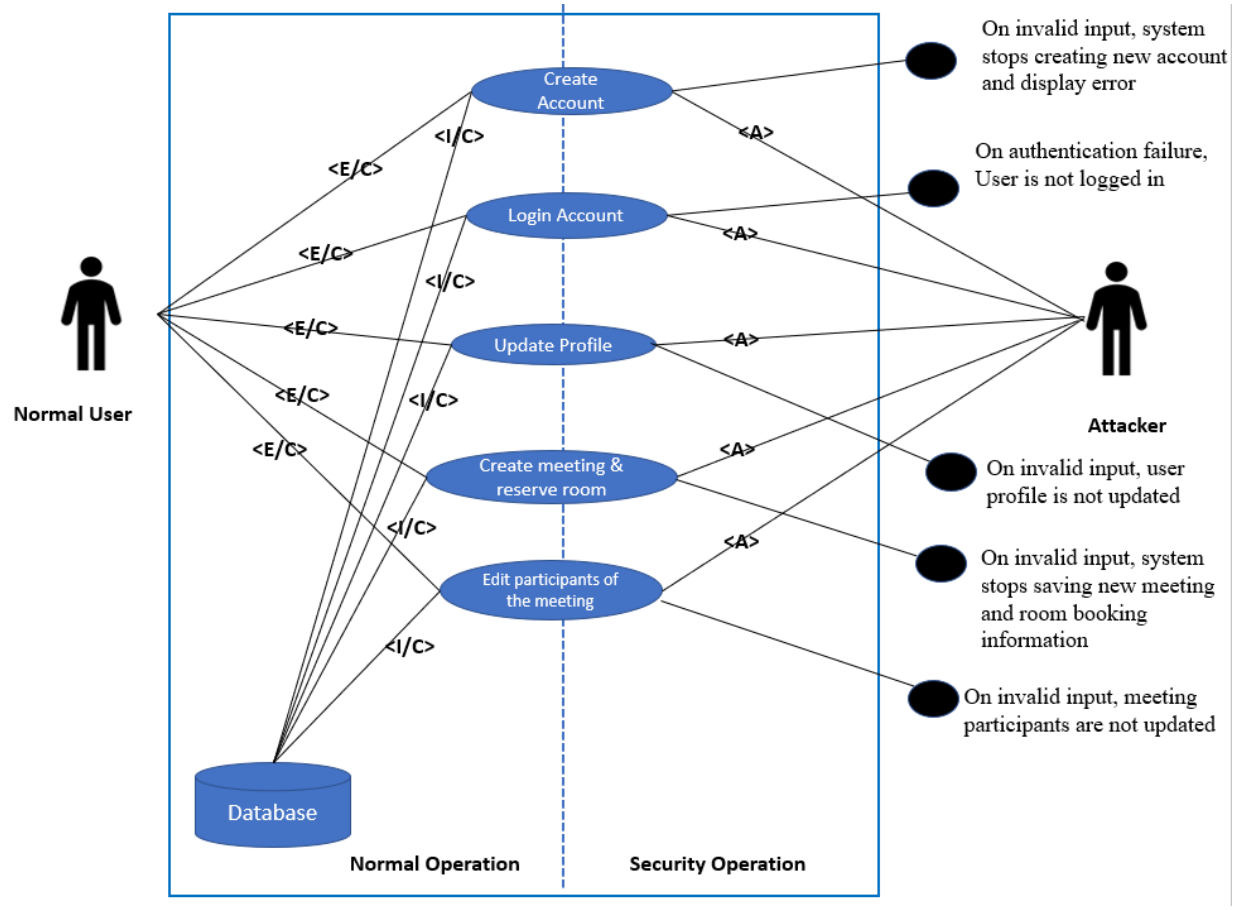

Figure 1: Extended Use Case Diagram with Misuse Management for normal user part1

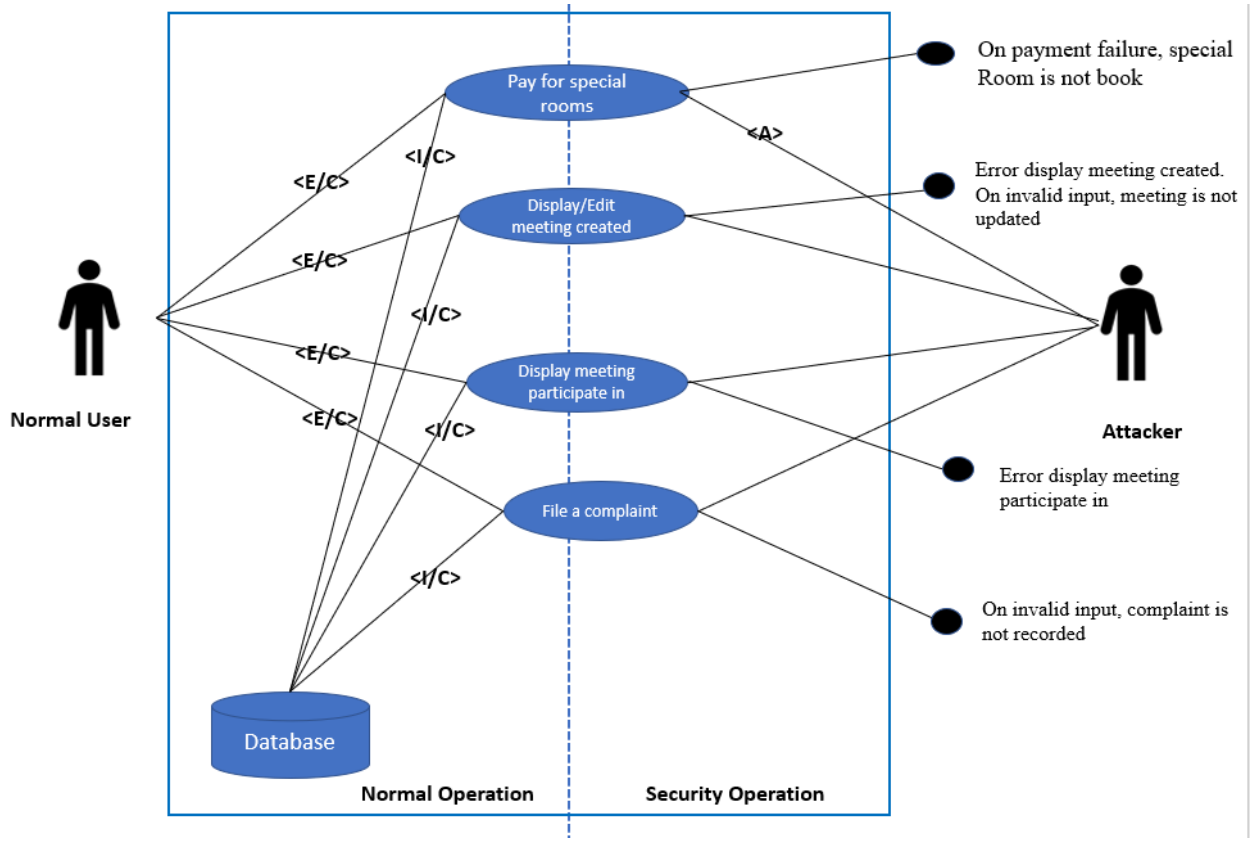

Figure 2: Extended Use Case Diagram with Misuse Management for normal user part2

2.7.2 Client Use Case – Create Account

Description	Create an account using a unique company email address
Actors	Client User
Trigger	Client navigates to create accounts page
Scenario	1. (trigger) Client navigates to create account page. 2. The client enters their company email address. 3. The system validates that the company email address entered exists in the database and was not registered before. 4. The client enters the desired username. 5. The system validates that the username entered meets the required format, is unique, and not registered before. 6. The client enters the password. 7. The system validates the password meets the security requirement format. 8. The client hits the create account button. 9. The system registered the account with the information supplied into the database.
Alternative Scenario	3a: The email address entered is not in the database: 3a1: The system informs the client that the email address entered is not in the database 3a2: The system provides a link for the user to contact the administrator 3b: The email address entered already registered before: 3b1: The system informs the client that the email address entered already registered before 3b2: The client can choose to log in, request for a password reset, or contact the administrator. 5a: The username entered already registered before 5a1: The system informs the user that the username has already been registered before 5a2: The user can choose to log in, request for a password reset, enter a new username or contact the administrator.
Fail Case	Client unable to create account
Consequence of Failure	The client is unable to use the system.
Associated Risks	The client may stop using the system
Fail Exit State	On invalid input, the system stops creating a new account, and display the warning containing the cause of the error to the users

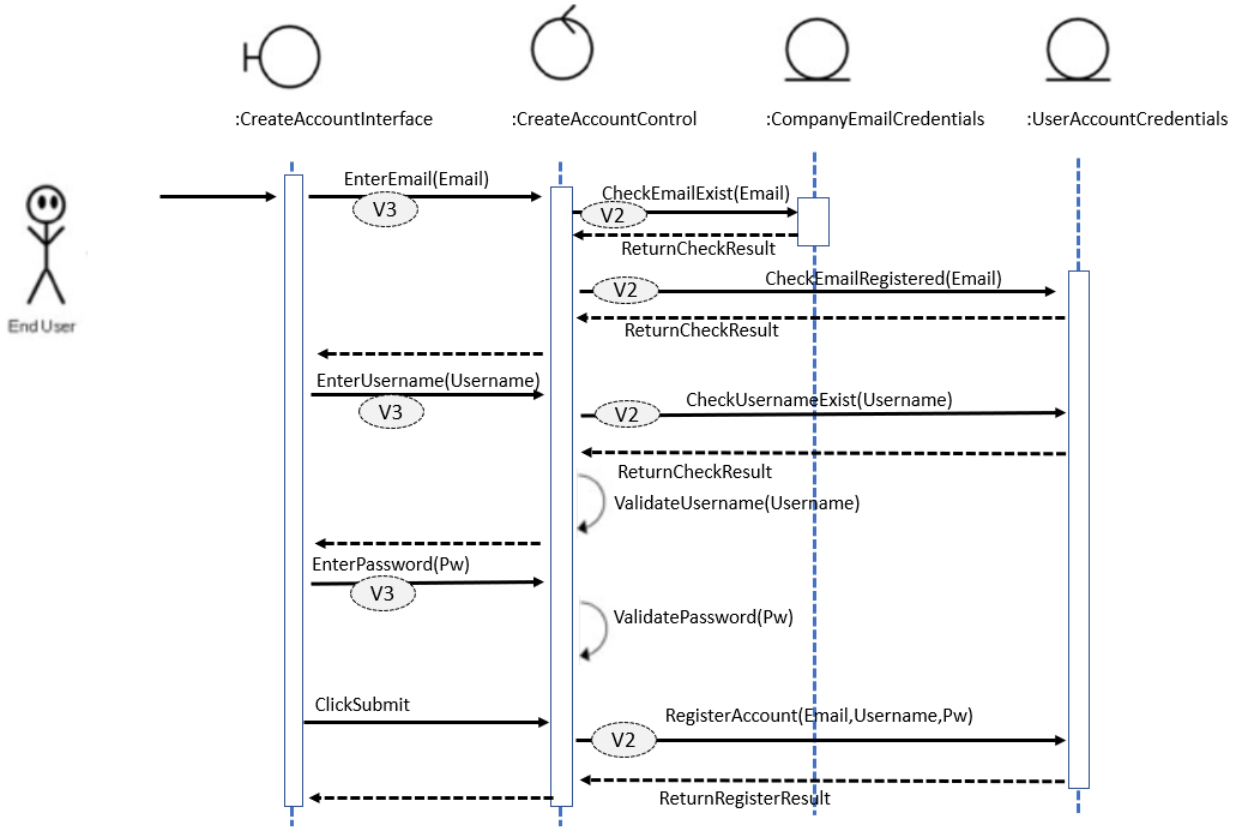

Figure 3: Extended Sequence Diagram for Use-Case Create Account

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
EnterEmail [V3]	Encryption for data transmitted.
CheckEmailExist [V2]	Encryption for data transmitted.
CheckEmailRegistered [V2]	Encryption for data transmitted.
EnterUsername [V3]	Encryption for data transmitted.
CheckUsernameExist [V2]	Encryption for data transmitted.
EnterPassword [V3]	Encryption for data transmitted.
RegisterAccount[V2]	Encryption for data transmitted.

2.7.3 Client Use Case – Login to Account

Description	Login into account
Actors	Client user
Trigger	The client navigates to the account login page
Scenario	1. The client navigates to the account login page. 2. The client enters the username of their account. 3. The client enters the password of their account. 4. The client hits the login button. 5. The system verifies that the username and password entered matched one of the registered accounts. 6. The system stores the login status. 7. The system directs the user to the main page of the application.
Alternative Scenario	5a: The system could not find a matching record for the username and password. 5a1: The system informs the user that there is no record for the username and password entered. 5a2: The client can choose to register for a new account, request for a password reset, enter new username/password information, or contact the administrator
Fail Case	The user is unable to login into the system.
Consequence of Failure	The user is unable to use the system to post questions or answer questions.
Associated Risks	Users may stop using the system and posts negative comments about the system.
Fail Exit State	The system will display error messages and redirect the user back to the account login page.

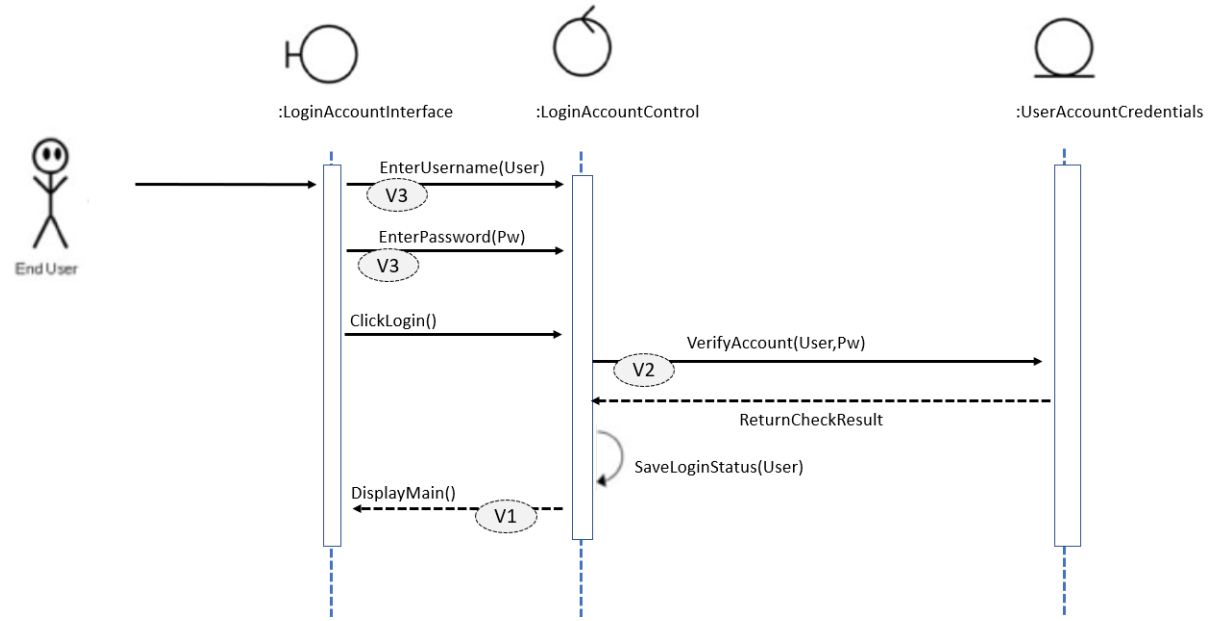

Figure 4: Extended Sequence Diagram for Use-Case Login Account

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
EnterUsername [V3]	Encryption for data transmitted.
EnterPassword [V3]	Encryption for data transmitted.
VerifyAccount [V2]	Encryption for data transmitted.

2.7.4 Client Use Case – Update Profile

Description	Update Profile
Actors	Client User
Trigger	Client navigates to update profile page
Scenario	1. The client navigates to update the profile page. 2. The system displays the profile information of the client after the authentication check. 3. The client selects the information field to edit (full name, address, contact number, job title, biography, password, credit card information). 4. The system displays the information entered by the client. 5. The client hits the submit button. 6. The system performs a validation check if applicable. 7. The system updates the information in the database
Alternative Scenario	6a: The information (example password) entered by the client failed the validation check. 6a1: The system displays a warning message. 6a2: Client will be returned to the profile page, where they can elect to select a new field to update
Fail Case	Client is unable to edit their profile information.
Consequence of Failure	The client's profile information stored in the system will be outdated.
Associated Risks	Other Clients will read the outdated information and send the meeting invites to the wrong person. The client may not able to pay for the booking of a special room.
Fail Exit State	Upon validation failure, no changes being made, and the system return to its original state.

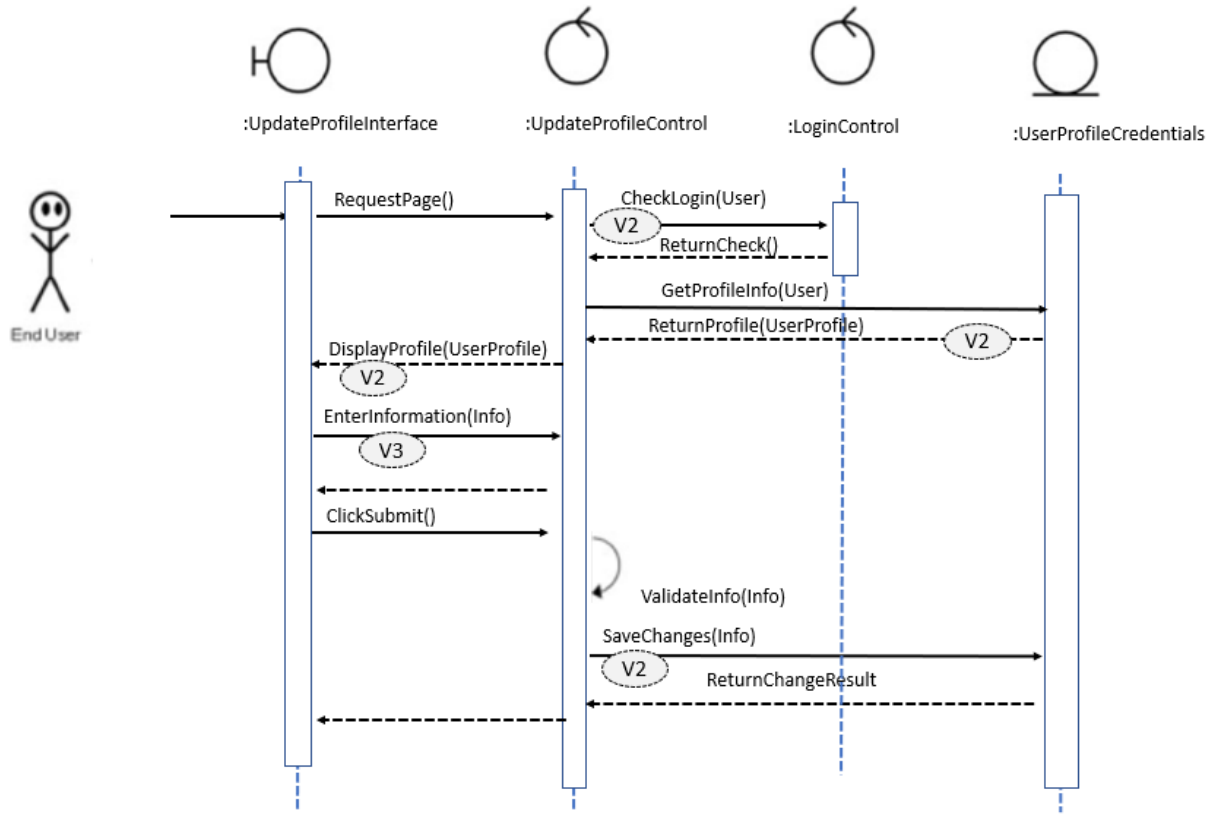

Figure 5: Extended Sequence Diagram for Use-Case Update Profile

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
ReturnProfile[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the user profile information should stop. Encryption for data transmitted.
DisplayProfile [V2]	Require authentication via CheckLogin. On authentication failure, the operation to display the user profile information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
EnterInformation[V3]	Encryption for data transmitted.
SaveChanges[V2]	Require authentication via CheckLogin. On authentication failure, the operation to save the user profile information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.5 Client Use Case – Create meeting and reserve room

Description	Create meeting and reserve room
Actors	Client User
Trigger	The client navigates to create a meeting page
Scenario	1. The client navigates to create a meeting page. 2. The system shows meeting time slots in the calendar view. 3. The client selects the meeting time slot and clicks create a meeting. 4. The system shows the meeting rooms available for the selected time slot. 5. The client selects the meeting room. 6. The system highlights the room selected. 7. The client entered the meeting title. 8. Client clicks submit to book the room and the meeting. 9. The system registered the meeting information and booking in the database.
Alternative Scenario	4a: There is no meeting room available at the selected time slot 4a1:The system prompts the client to select a different time slot
Fail Case	The client is unable to create a new meeting with the system.
Consequence of Failure	The client is unable to book a room for their meeting.
Associated Risks	Clients unable to meet with their colleagues for meetings may stop using the system.
Fail Exit State	The system will prevent the room from being booked, and return it to the create meeting page

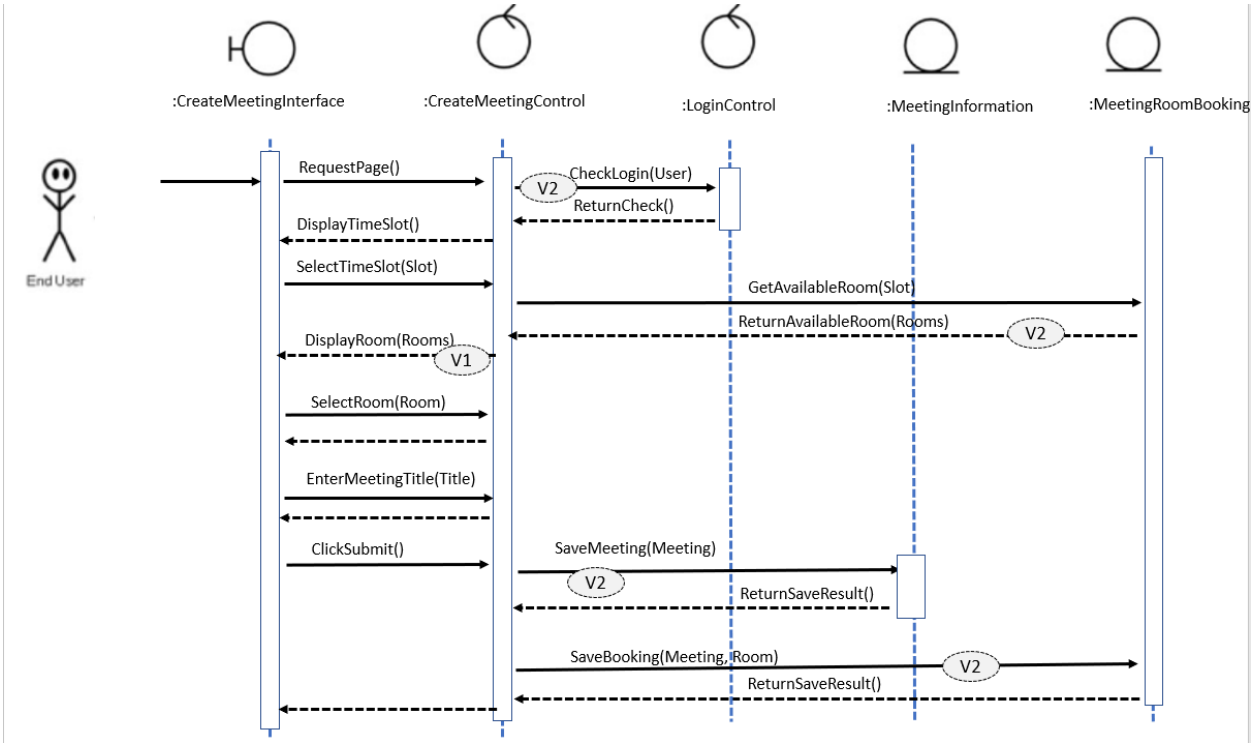

Figure 6: Extended Sequence Diagram for Use-Case Create meeting and reserve room

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
ReturnAvailableRoom[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the available rooms should stop. Encryption for data transmitted.
SaveMeeting [V2]	Require authentication via CheckLogin. On authentication failure, the operation to save a new meeting should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
SaveBooking[V2]	Require authentication via CheckLogin. On authentication failure, the operation to save the room booking information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.6 Client Use Case – Pay for special room

Description	Pay for special rooms
Actors	Client User
Trigger	The client selects a special room during create meeting stage
Scenario	1. The client selects a special room during create meeting stage 2. The system displays a pop-up window for the client to confirm the payment information. 3. The client confirms the payment information. 4. The client hits the submit button. 5. The system processes the payment information. 6. The system recorded the booking in the database
Alternative Scenario	2a. The system does not have a record of the client's payment information 2a1. The system prompts the user to enter the credit card information for payment 2a2. The client enters the credit card information 5a. The system unable to process the payment information 5a1. The system rejects the booking 5a2. The system goes back to the payment information window
Fail Case	The client is unable to book the special room as the system unable to process payment information.
Consequence of Failure	The client is unable to use the special room for the meetings.
Associated Risks	The special room(s) are not utilized. The client may stop using the system.
Fail Exit State	The system rejects the booking and goes back to the payment information window

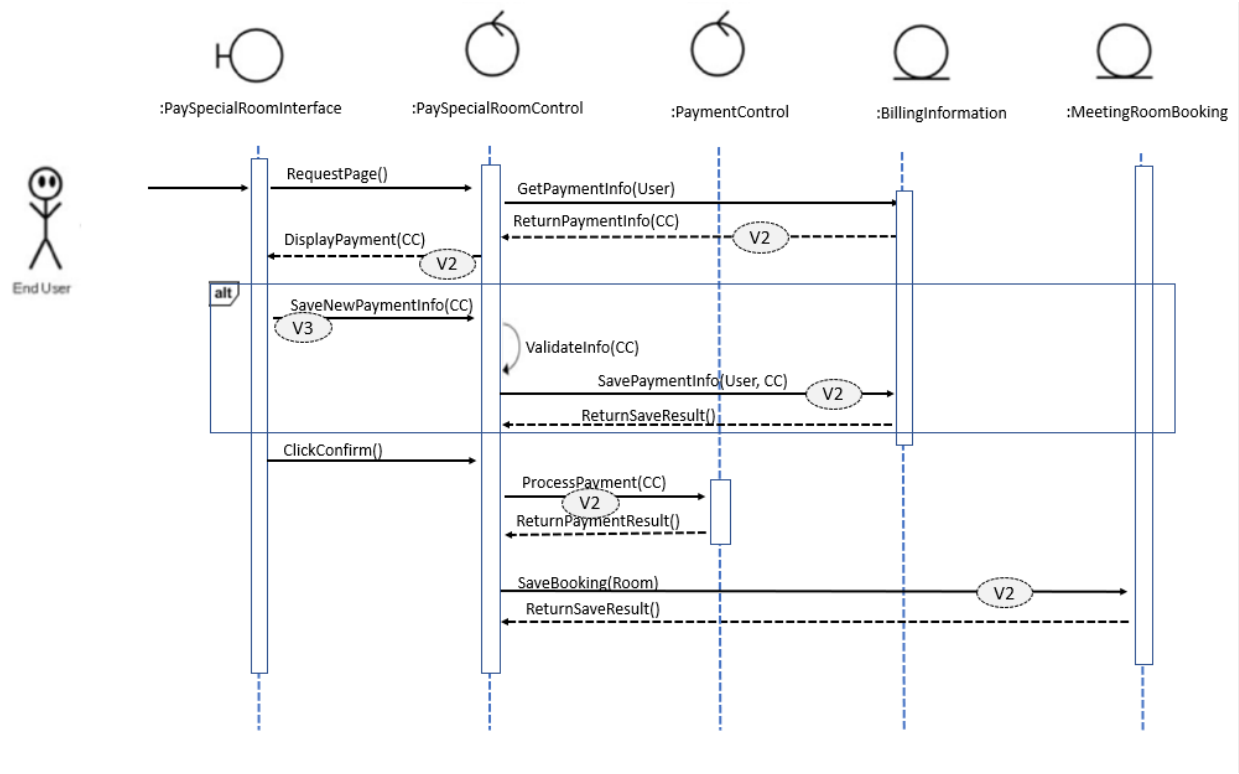

Figure 7: Extended Sequence Diagram for Use-Case Pay for Special Room

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
ReturnPaymentInfo[V2]	Require authentication. On authentication failure, the operation to retrieve the payment information should stop. Note the authentication status should be maintained from the previous create meeting page. Encryption for data transmitted.
DisplayPayment[V2]	Require authentication. On authentication failure, the operation to retrieve the payment information should stop. Note the authentication status should be maintained from the previous create meeting page. Encryption for data transmitted.
SaveNewPaymentInfo[V3]	Encryption for data transmitted.
SavePaymentInfo[V2]	Require authentication. On authentication failure, the operation to save the payment information should stop. Note the authentication status should be maintained from the previous create meeting page. Encryption for data transmitted.

ProcessPayment[V2]	Require authentication. On authentication failure, the operation to save the payment information should stop. Note the authentication status should be maintained from the previous create meeting page. Encryption for data transmitted.
SaveBooking[V2]	Require authentication. On authentication failure, the operation to save the special room booking information should stop. Note the authentication status should be maintained from the previous create meeting page. Encryption for data transmitted.

2.7.7 Client Use Case – Edit participants

Description	Edit participants of the meeting
Actors	Client User
Trigger	The client navigate to the meeting overview page
Scenario	1. (Trigger)The client navigates to the meeting overview page. 2. The system displays a list of meetings hosted by clients. 3. The client selects a meeting to edit. 4. The system displays meeting information and a list of participants. 5. The client clicks the button to add a participant. 6. The client searches for the participant using the company email address. 7. The system returns the searched result. 8. The client selects the searched participant and clicks the submit button. 9. The system checks if the participant is joining another meeting at the same time. 10. The system records the participant information in the database.
Alternative Scenario	5a. Client clicks the button to remove participant. 5a1. The system proceeds to change the participant information in the database 9a. The participant already joining another meeting at the same time. 9a1. The system displays an error message and returns to the scheduled meeting page
Fail Case	The client is unable to add participants to their meeting.
Consequence of Failure	The participant was not notified that they were invited to the meeting.
Associated Risks	The meeting fails to get the required quorum. The meeting creator and participants are dissatisfied with the system.
Fail Exit State	The system rejects adding participant return to view schedule meeting page.

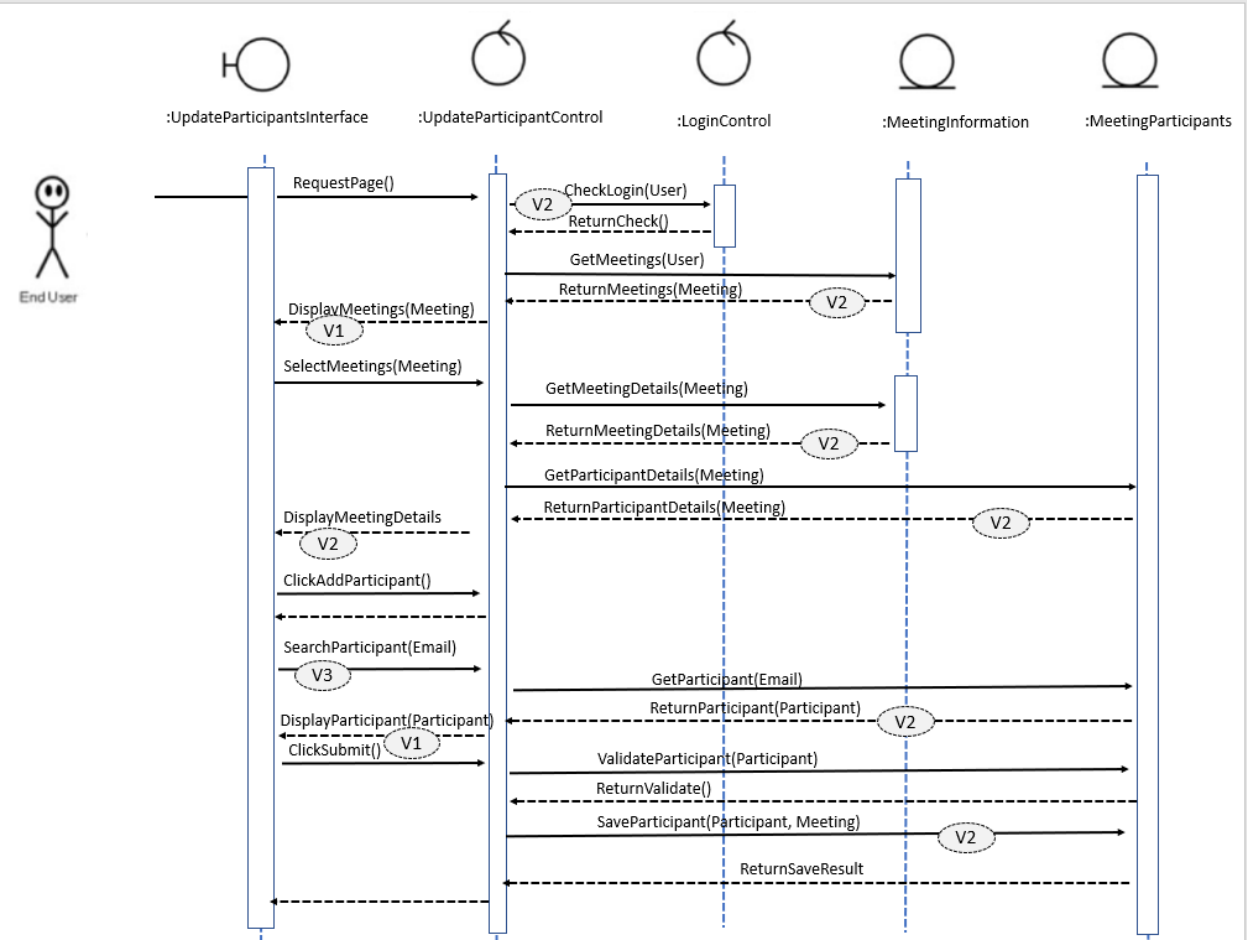

Figure 8: Extended Sequence Diagram for Use-Case Edit Participants

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
ReturnMeetings[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the meeting should stop. Encryption for data transmitted.
ReturnMeetingDetails [V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the meeting details should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
DisplayMeetingDetails [V2]	Require authentication via CheckLogin. On authentication failure, the operation to display the meeting details should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
SearchParticipant[V3]	Encryption for data transmitted.
ReturnParticipant[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the participant's name should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
SaveParticipant[V2]	Require authentication via CheckLogin. On authentication failure, the operation to save the participant in the meeting should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.8 Client Use Case – Edit a Meeting

Description	Edit a Pre-Existing Meeting
Actors	Client User
Trigger	The client clicks on the meeting to edit
Scenario	1. (trigger) The client clicks on the meeting to edit. 2. The system displays the information of the meeting (title, timeslot, room, participants). 3. The client chooses which meeting attribute they want to edit (meeting title, timeslot, room, participants). 4. For new timeslot requests, the system displays the available timeslot in the calendar view. 5. For new meeting room requests, the system displays the available room at the selected timeslot. 6. To add/remove participants, the system will direct them to the edit participants page. 7. Client selects/enters new information. 8. For new timeslot selection, the system will validate if there is room for conflict. 9. For new timeslot selection, the system will also validate if there is participation conflict. 10. The client clicks the submit button. 11. The system updates the information in the database
Alternative Scenario	7. There is room conflict, the room booked in the original timeslot not available in the new timeslot. 7a1. The system displays an error and prompts the client to select a new room. 8a. There is participation conflict, the participants in the original timeslot are not available in the new timeslot. 8a1. The system displays options for the client to select another timeslot or remove the participant from the meeting.
Fail Case	1. The system fails to discover actual conflicts. 2. The client is unable to edit the meetings.
Consequence of Failure	1. Clients can be asked to join multiple meetings. Or multiple meetings book the same room at the same time. 2. The client's meeting is not updated.
Associated Risks	Dissatisfaction among the clients, the meetings cannot proceed, they may stop using the system.

Fail Exit State	The system displays a failure message and clears the edited information, prompting the user to try again.

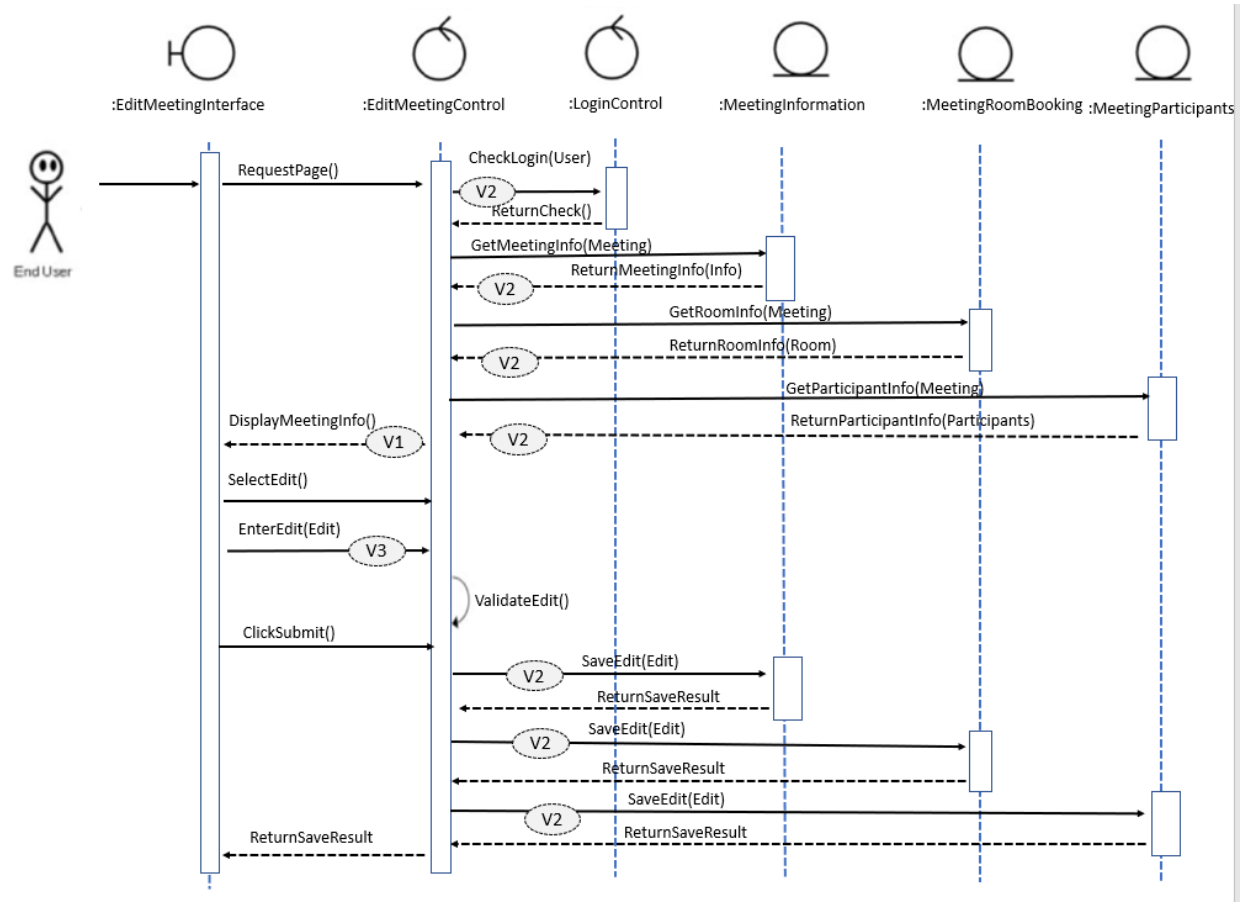

Figure 9: Extended Sequence Diagram for Use-Case Edit Meetings

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
ReturnMeetingInfo[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the meeting information should stop. Encryption for data transmitted.
ReturnRoomInfo[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the room information should stop. Encryption for data transmitted.

ReturnParticipantInfo[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the participant information should stop. Encryption for data transmitted.
EnterEdit[V3]	Encryption for data transmitted.
SaveEdit[V2]	Require authentication. On authentication failure, the operation to save the changes in meeting information, room selected, participant edit should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.9 Client Use Case – View Meeting

Description	View Meeting (User is added as a participant)
Actors	Client User
Trigger	User selects a meeting and chooses “view”
Scenario	1. (trigger) The client presses the button 2. The system retrieves meeting details from the database 3. The data is displayed on the screen
Alternative Scenario	3a. The meeting has been canceled 3b. The system displays cancellation details
Fail Case	1. The system fails to retrieve any details of the meeting 2. The system mistakenly retrieves erroneous data
Consequence of Failure	1a. The meeting information page fails to load 2a. The system displays erroneous data, directing the user to the wrong place and/or the wrong time
Associated Risks	1. Prevention of user efficiently receiving correct data (failure to complete a key task) 2. General confusion regarding dates/times/locations of meetings
Fail Exit State	1. The system directs the user to refresh the page and try again

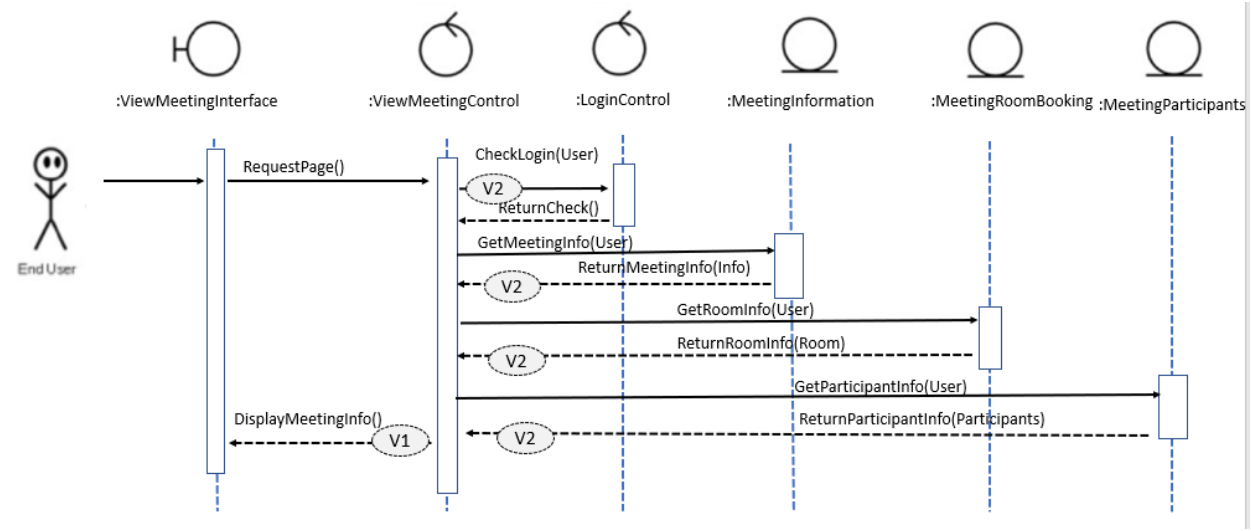

Figure 10: Extended Sequence Diagram for Use-Case View Meetings

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
ReturnMeetingInfo[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the meeting information should stop. Encryption for data transmitted. Data returned should be restricted to the minimum required only.
ReturnRoomInfo[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the room information should stop. Encryption for data transmitted. Data returned should be restricted to the minimum required only.
ReturnParticipantInfo[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the participant information should stop. Encryption for data transmitted. Data returned should be restricted to the minimum required only.

2.7.10 Client Use Case – File a Complaint

Description	File a Complaint
Actors	Client User
Trigger	User clicks the "Report Issue" button
Scenario	1. (trigger) The user clicks the button. 2. The user selects the type of issue they would like to report from a dropdown menu. 3. The user describes the issue in a text field. 4. The user clicks "submit." 5. The system stores the complaint information in the database. 6. The system sends a message containing the complaint and the submitting user's email to all administrators to notify them of the issue
Alternative Scenario	5a. System unable to store the complaint information. 5a1. System displays the error message.
Fail Case	The information fails to be stored in the database.
Consequence of Failure	The complaints by the user will not reach the administrators.
Associated Risks	Uncaught bugs are left in the system, leaving potential vulnerabilities. The client may find the system unsafe and stop the use of the system.
Fail Exit State	The system directs the user to reattempt submission or contact administrators directly via email

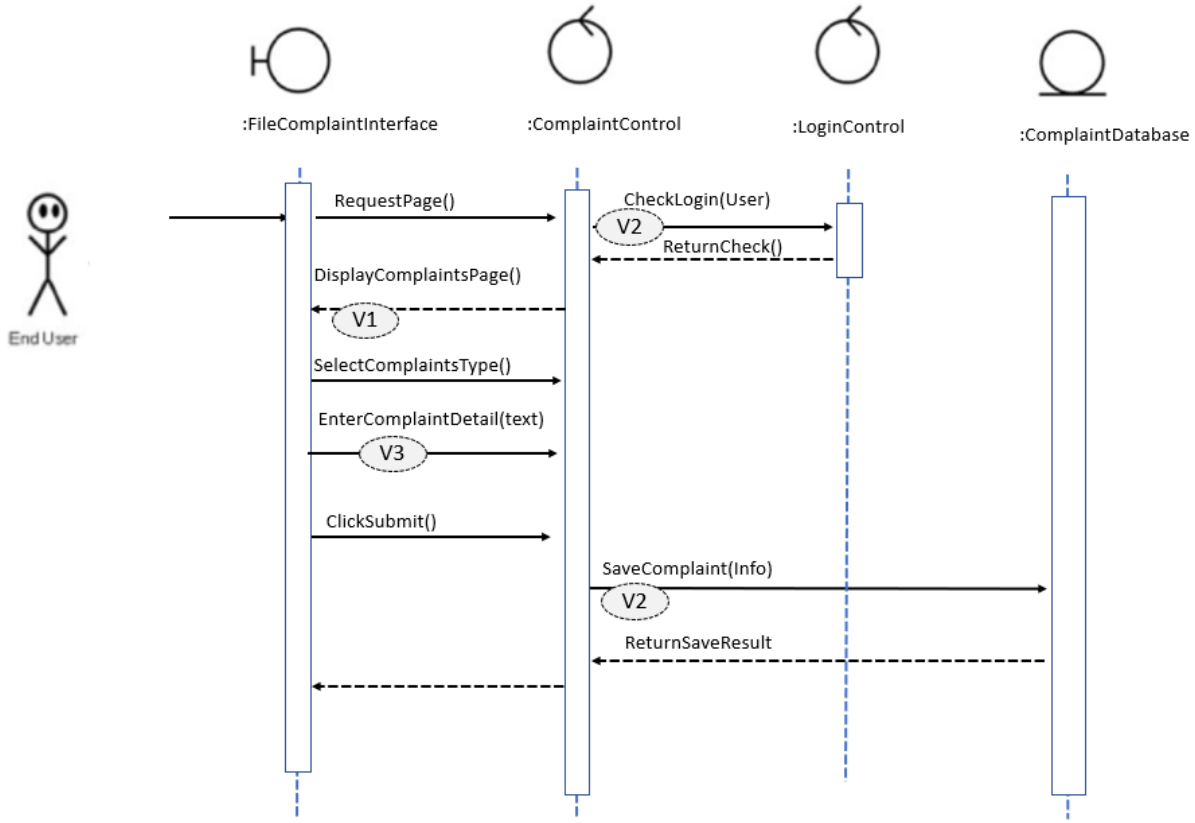

Figure 11: Extended Sequence Diagram for Use-Case File Complaint

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin[V2]	Encryption for data transmitted. Note if the client navigates to this page after login their login status should be maintained across the pages.
EnterComplaintDetail[V3]	Encryption for data transmitted.
SaveComplaint[V2]	Encryption for data transmitted.

2.7.11 Extended Use Case Diagram with Misuse Management for Administrator

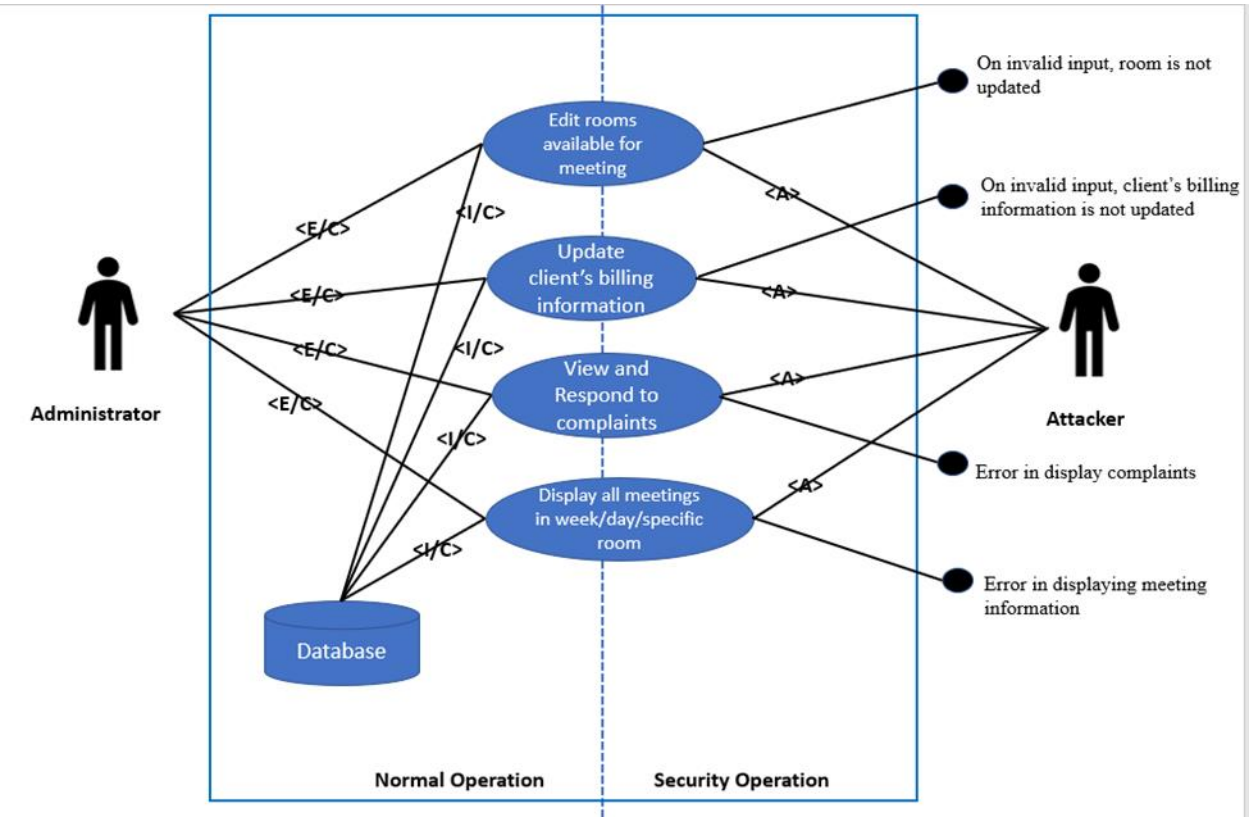

Figure 12: Extended Use Case Diagram with Misuse Management for administrator user part1

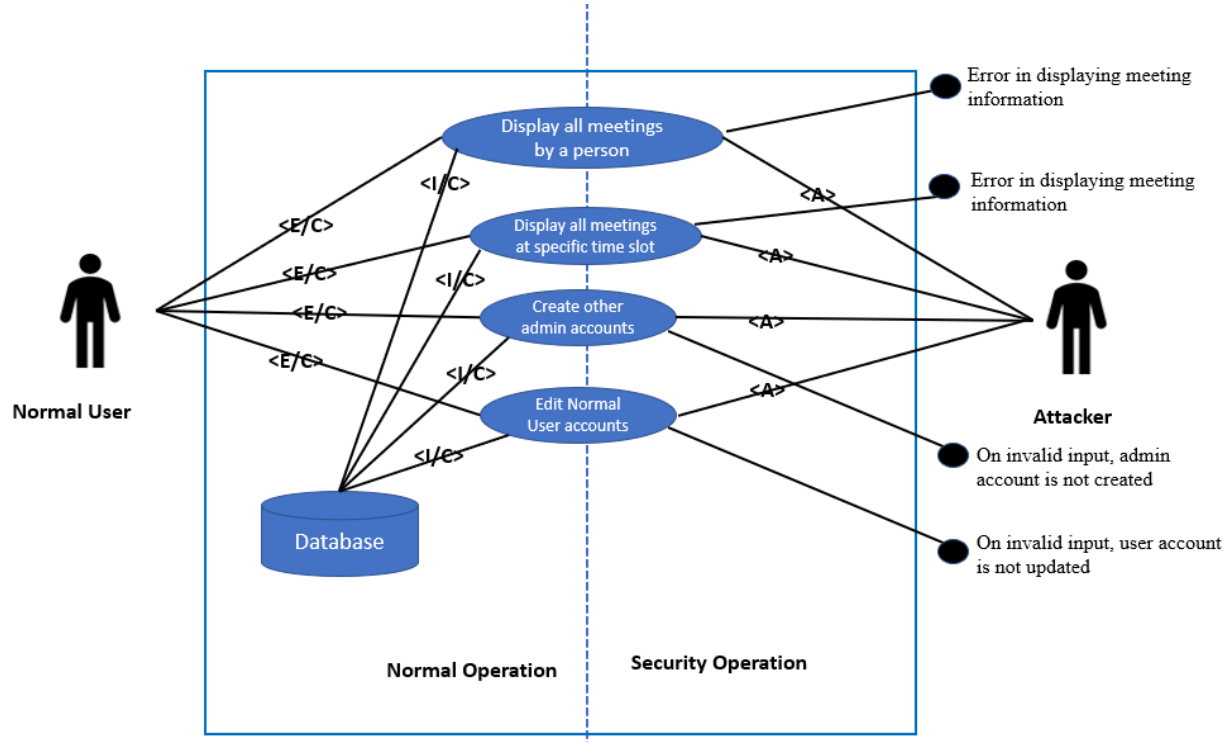

Figure 13: Extended Use Case Diagram with Misuse Management for administrator user part2

2.7.12 Administrator Use Case – Update Billing Info

Description	Update Billing Info
Actors	Administrator User
Trigger	User selects “update billing information”
Scenario	1. The user clicks the “Update Billing Info” button 2. The system asks to confirm the user’s password 3. The user enters a password 4. The system validates the password 5. The system displays the current card in the format “xxxx-xxxx-xxxx-1234” 6. The user clicks “edit” 7. The user enters their new card info (card number, exp. date, cvv) 8. The system checks if the card is valid 9. The system saves the new card information to the database
Alternative Scenario	3a. The password is not valid 3b. The system prompts the user to try again 5a. The user selects “back” 5b. The user is redirected to the home page 8a. The card is not valid 8b. The user displays an error
Fail Case	1. The system saves an invalid card 2. The system fails to validate password
Consequence of Failure	1a. The user’s payment is declined when booking a premium room 2a. The user cannot access their billing info to change it
Associated Risks	1. Faulty payment information storage could prevent a key task from being performed; the client may cease using the product.
Fail Exit State	1. The system displays a card validation failure error when the erroneous card is used 2. The system prompts the user to try editing the card again upon failure

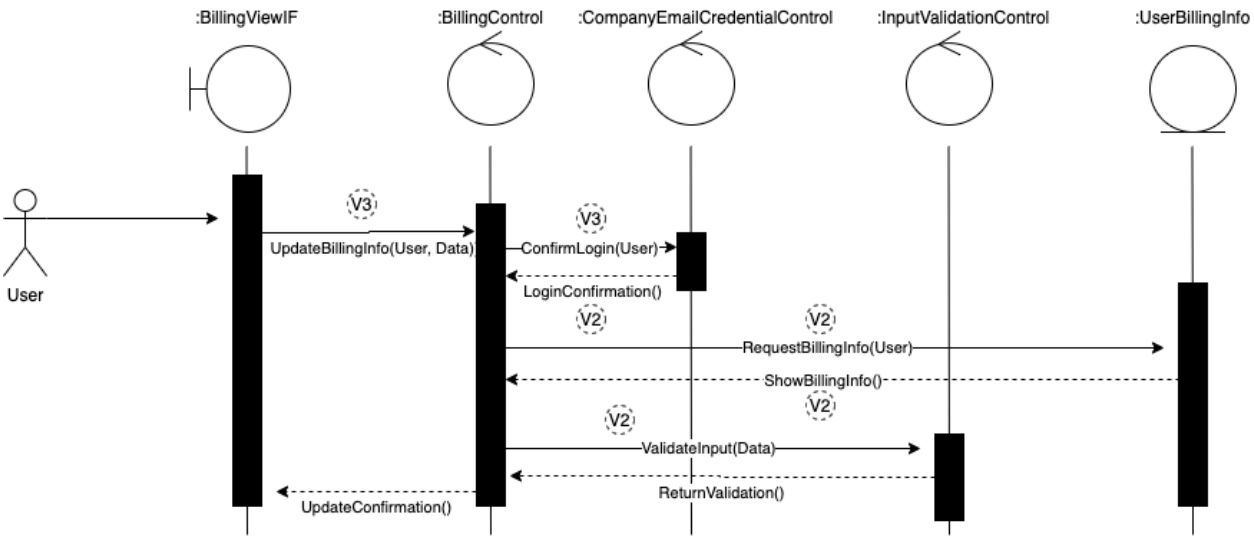

Figure 14: Extended Sequence Diagram for Use-Case Update Client Billing

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
UpdateBillingInfo[V3]	Encryption for data transmitted. Required authentication via the ConfirmLogin(). On authentication failure, the operation to update BillingInfo should stops.
ConfirmLogin[V3]	Encryption for data transmitted.
RequestBillingInfo[V2]	Encryption for data transmitted. Required authentication, authentication status from previous step should be maintained.
ShowBillingInfo[V2]	Encryption for data transmitted. Required authentication, authentication status from previous step should be maintained.
ValidateInput[V2]	Encryption for data transmitted. Required authentication, authentication status from previous step should be maintained.

2.7.13 Administrator Use Case - Add a New Meeting Room

Description	Add a Meeting Room
Actors	Administrator User
Trigger	The user presses the "Add" button
Scenario	1. (trigger) The system displays a page and prompts the user to enter room details (name/number, building, capacity) 2. The user enters the room information 3. The user clicks "create" 4. The system validates input 5. The system attempts to save data in the database 6. The system displays a confirmation message
Alternative Scenario	3a. The user cancels the room creation 3b. The system redirects the user to the admin home page 4a. The system catches invalid input 4b. The system prompts the user to edit erroneous fields
Fail Case	1. The system fails to save room data 2. Invalid input is accepted
Consequence of Failure	1a. The new room is never added 2a. The system stores erroneous data
Associated Risks	1. Standard users may be confused if an error is made in room addition, causing meeting locations to be more difficult to find
Fail Exit State	The system displays an error and directs the user to try again

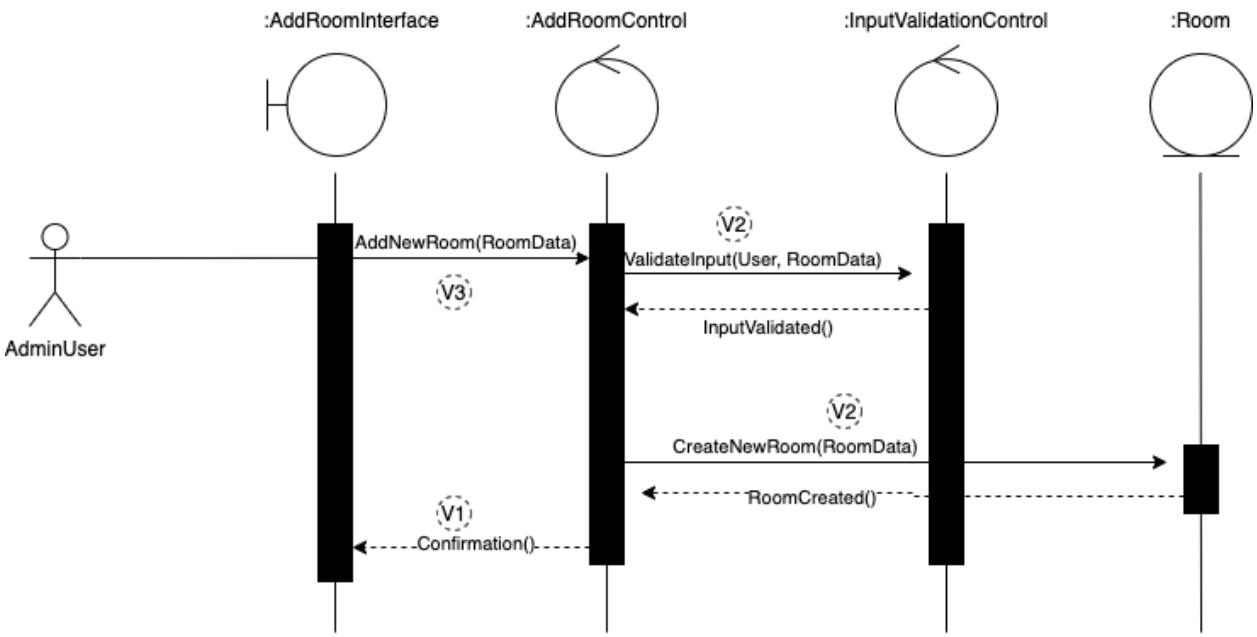

Figure 15: Extended Sequence Diagram for Use-Case Add Room

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
AddMeetingRoom[V3]	The administrator's login information should be re-verified prior to allowing a new room to be added. Further, all input should be validated prior to entering into a database to prevent malicious data from being entered (ValidateInput serves this purpose).
ValidateInput[V2]	All sensitive data being transmitted should be encrypted. Avoid excessive details from being visible.
CreateNewRoom[V2]	Any and all user data being transmitted should be encrypted.
Confirmation[V1]	There is potential for the page to not load properly.

2.7.14 Administrator Use Case - Remove a Meeting Room

Description	Delete a Meeting Room
Actors	Administrator User
Trigger	User selects a room and presses the “delete” button
Scenario	1. (trigger) The system displays the details of the selected room 2. The system asks the user if they’re sure this is the room they want to delete 3. The user selects “confirm deletion” 4. The system sends a message to meeting creators who had a meeting in the room that was deleted to inform them of the deletion 5. The system removes the room data from the database, including all scheduled meetings
Alternative Scenario	3a. The user selects “cancel” 3b. The system redirects the user to the admin home page
Fail Case	1. The room fails to send the deletion notification 2. The room fails to remove the room from the database 3. The incorrect room is deleted
Consequence of Failure	1a. Meeting creators are never informed that the room is no longer available 2a. The system shows an unavailable room as being available 3a. The system shows an available room as being unavailable, causing the administrator to have to restore the room
Associated Risks	1. Potential mass confusion regarding room availability
Fail Exit State	1. The system displays an error and directs the user to try again 2. The system sends a message to all users affected by the error

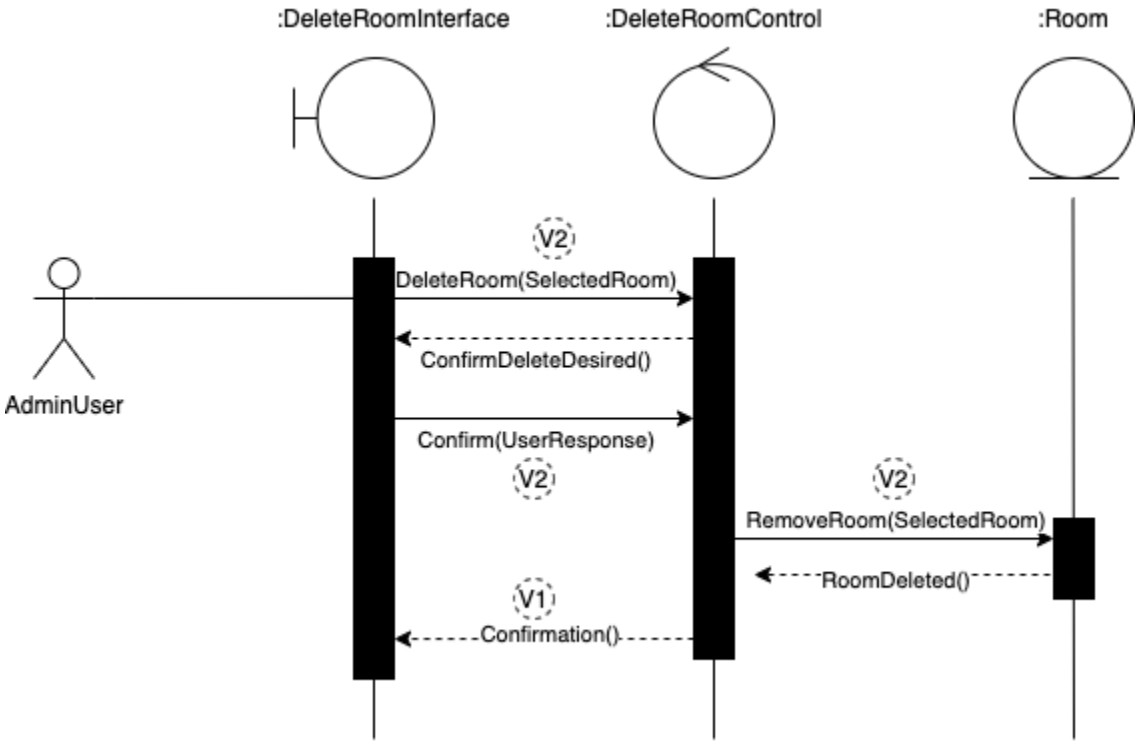

Figure 16: Extended Sequence Diagram for Use-Case delete Room

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
DeleteRoom[V2]	The administrator's login information should be re-verified prior to allowing a new room to be added. Further, any user data used in the transaction should be encrypted.
Confirm[V2]	A response (in the form of a button click) should not be an available option if the user has not been verified.
RemoveRoom[V2]	Any and all user data being transmitted should be encrypted. In addition, details of the deleted room may also be encrypted.
Confirmation[V1]	There is potential for the page to not load properly.

2.7.15 Administrator Use Case - Reply to client's complaints

Description	Reply to client's complaints
Actors	Administrator User
Trigger	An Admin clicks the "View Complaints" button
Scenario	1. (trigger) The administrator clicks on the "View Complaints" button. 2. The system returns all complaints received from various users and their basic contact info. 3. The admin selects a complaint to view. 4. The system returns more details about the complaint (descriptions entered by users, etc). 5. The administrator clicks reply and types their reply in the textbox. 6. The system sends an email containing the reply made by the administrator to the client. 7. The complaint is marked as resolved. 8. The administrator is directed back to the complaints page.
Alternative Scenario	3a. The administrator clicks "Return" when viewing the list of complaints. 5a. The administrator clicks "Return" after viewing the client's complaint. 6a. The system fails to send the email to the client. 7a. The complaint is not marked as resolved.
Fail Case	1. The system fails to send the email to the client. 2. The complaint is not marked as resolved.
Consequence of Failure	The client will not be informed if the complaint was replied to. The system will continue to show the complaint even if it was resolved.
Associated Risks	Over time, the client will think that the administrator team is lazy and doesn't care about their client's complaints. Administrators will get frustrated seeing the complaint they just replied to show up under the complaints category.
Fail Exit State	The system sends an error message regarding the email not being sent. The system will inform the administrator that the complaint was not marked as resolved.

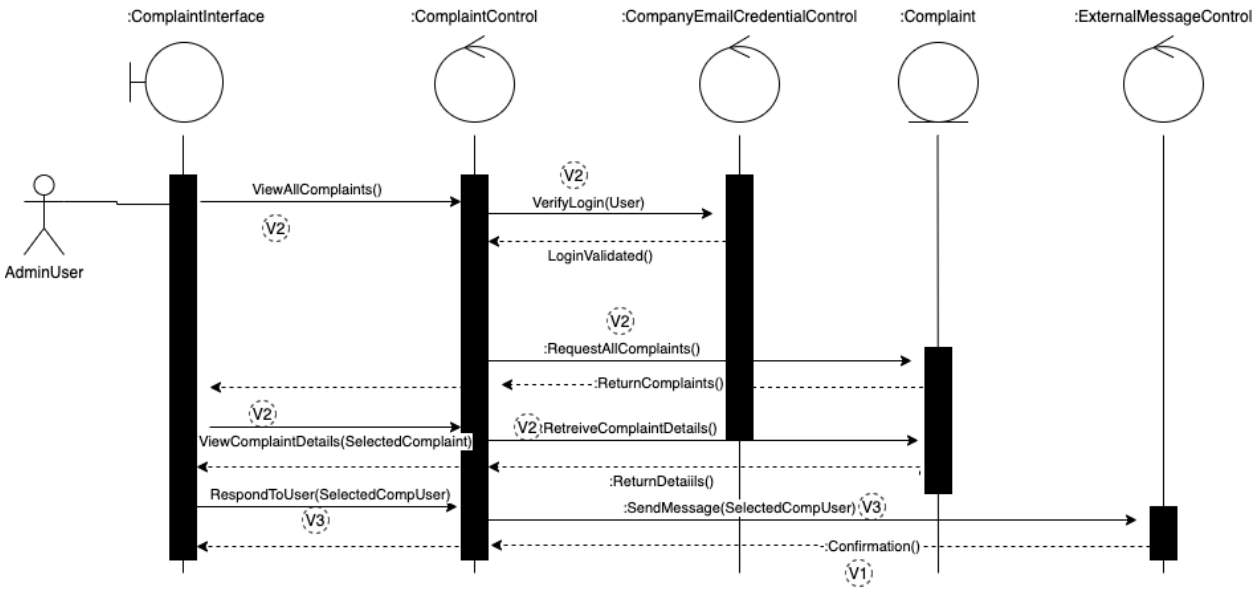

Figure 17: Extended Sequence Diagram for Use-Case delete Room

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
ViewAllComplaints[V2]	Ensure that returned user data is encrypted. Further, the administrator's login information should be re-verified prior to allowing access to the list of complaints. Necessary details should not be viewable until the administrator is verified.
VerifyLogin[V2]	Admin user data should be encrypted to prevent unauthorized access to the system.
RequestAllComplaints[V2]	As previously stated, any administrator should be verified prior to returning complaints (VerifyLogin performs this check). In addition, details pertaining to users that submitted complaints should be hidden during transmission. Only necessary data should be returned.
RespondToUser[V3]	Input should be validated to prevent against SQL and script injection attacks. Further, messages should not be able to be sent unless the user is an administrator. Finally, user data must be left encrypted.
SendMessage[V3]	Encrypt sensitive data while sending messages. Again, further check the data being sent to avoid sending any malicious messages.
Confirmation[V1]	There is potential for the page to not load properly.

2.7.16 Administrator Use Case - Update Client's account Information.

Description	Edit a client's account information
Actors	Administrator User
Trigger	The administrator clicks the "Search account" button.
Scenario	1. (trigger) The administrator account clicks the "Search account" button. 2. The administrator enters the ID or name of the client. 3. The administrator clicks "Search". 4. The system queries through the client database and displays clients that match the name or ID entered. 5. The administrator clicks on the client's profile. 6. The administrator clicks on the "Manage profile" button. 7. The system displays text fields in the user interface that allows the administrator to edit the client's profile. 8. The administrator updates the selected part of the client's profile (this include the billing information). 9. The administrator clicks "Save Changes". 10. The system records the changes into the database
Alternative Scenario	4a. The system cannot find any matching results to the administrator's request. 4a1: The system displays "No result found" message. 10a. The system does not save the changes made by the client. 10a1: The system displays "save error" message.
Fail Case	The admin unable to edit client's account
Consequence of Failure	Client information are not updated
Associated Risks	The administrator will be frustrated. The client could be dissatisfied as well.
Fail Exit State	The system will inform the administrator account that the information failed to save.

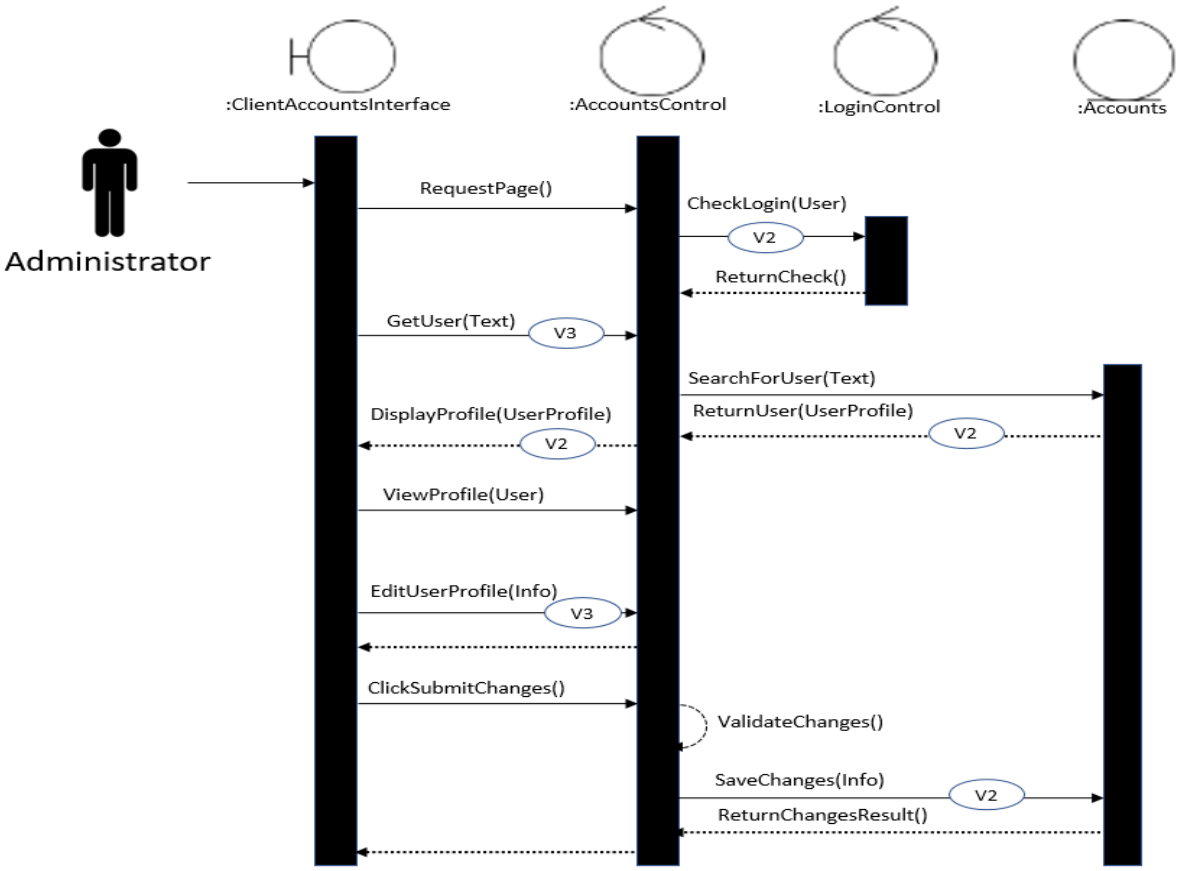

Figure 18: Extended Sequence Diagram for Use-Case Update Client's Information

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
GetUser [V3]	Encryption for data transmitted.
ReturnUser[V2]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the user profile information should stop. Encryption for data transmitted.
DisplayProfile [V2]	Require authentication via CheckLogin. On authentication failure, the operation to display the user profile information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
EditUserProfile[V3]	Encryption for data transmitted.
SaveChanges[V2]	Require authentication via CheckLogin. On authentication failure, the operation to save the user profile information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.17 Administrator Use Case - Display meetings by meeting room.

Description	Display meetings in a specific meeting room.
Actors	Administrator User
Trigger	The administrator clicks on the “Search Meetings by Meeting Room” button.
Scenario	1. (trigger) The administrator clicks on the “Search Meetings by Meeting Room” button. 2. The system prompts the administrator to type a Meeting Room ID. 3. The system queries through the database and displays the meetings for the selected meeting room ID. 4. The administrator selects a meeting to view the details. 5. The system displays the details of the meeting.
Alternative Scenario	3a. The system cannot find any results. 3a1: The system displays “No results found.”
Fail Case	The system has an error querying into the meetings database.
Consequence of Failure	The system fails to display the meeting results.
Associated Risks	The administrator is unable to get up-to-date meeting information. This may cause frustration and dissatisfaction with the systems.
Fail Exit State	The system shows the “No Result” found on the results page.

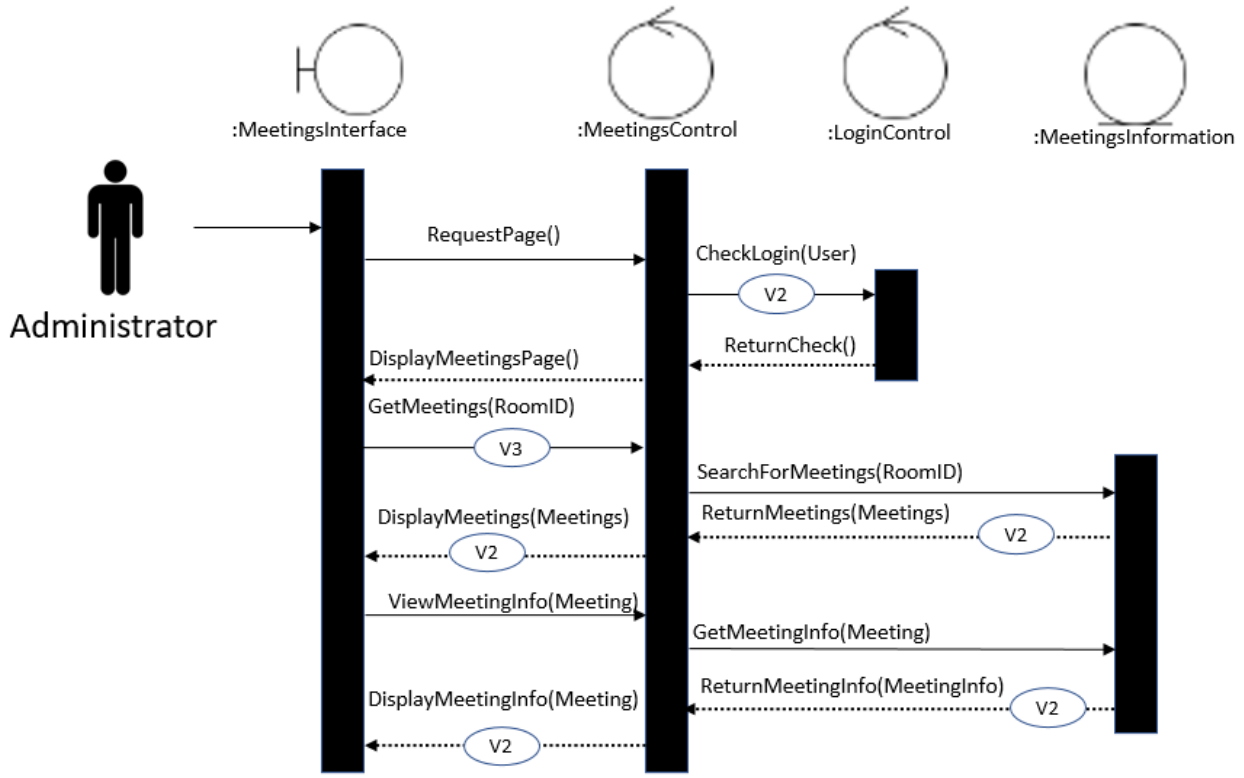

Figure 19: Extended Sequence Diagram for Use-Case Display Meeting by room

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
GetMeetings [V3]	Encryption for data transmitted.
ReturnMeetings [V2]	Encryption for data transmitted. Require authentication via CheckLogin. On authentication failure, the operation to retrieve the meetings information should stop.
DisplayMeetings [V2]	Require authentication, on authentication failure, the operation to display the meetings information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
ReturnMeetingInfo [V2]	Require authentication, on authentication failure, the operation to retrieve the specific meetings information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

DisplayMeetingInfo [V2]	Require authentication, on authentication failure, the operation to display the specific meetings information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.18 Administrator Use Case - Display meetings by timeslot.

Description	Display meetings in a specific timeslot.
Actors	Administrator User
Trigger	The administrator clicks on the "Search Meetings by Time-slot" button.
Scenario	1. (trigger) The administrator clicks on the "Search Meetings by Time-slot" button. 2. The system displays the timeslots in the calendar view. 3. The administrator clicks a timeslot on a calendar. 4. The system queries through the database and displays the meetings for the selected timeslot. 5. The administrator selects a meeting to view the details. 6. The system displays the details of the meeting.
Alternative Scenario	4a. The system cannot find any results. 4a1: The system displays "No results found."
Fail Case	The system has an error querying into the meetings database.
Consequence of Failure	The system fails to display the meeting results.
Associated Risks	The administrator is unable to get up-to-date meeting information. This may cause frustration and dissatisfaction with the systems.
Fail Exit State	The system shows the "No Result" found on the results page.

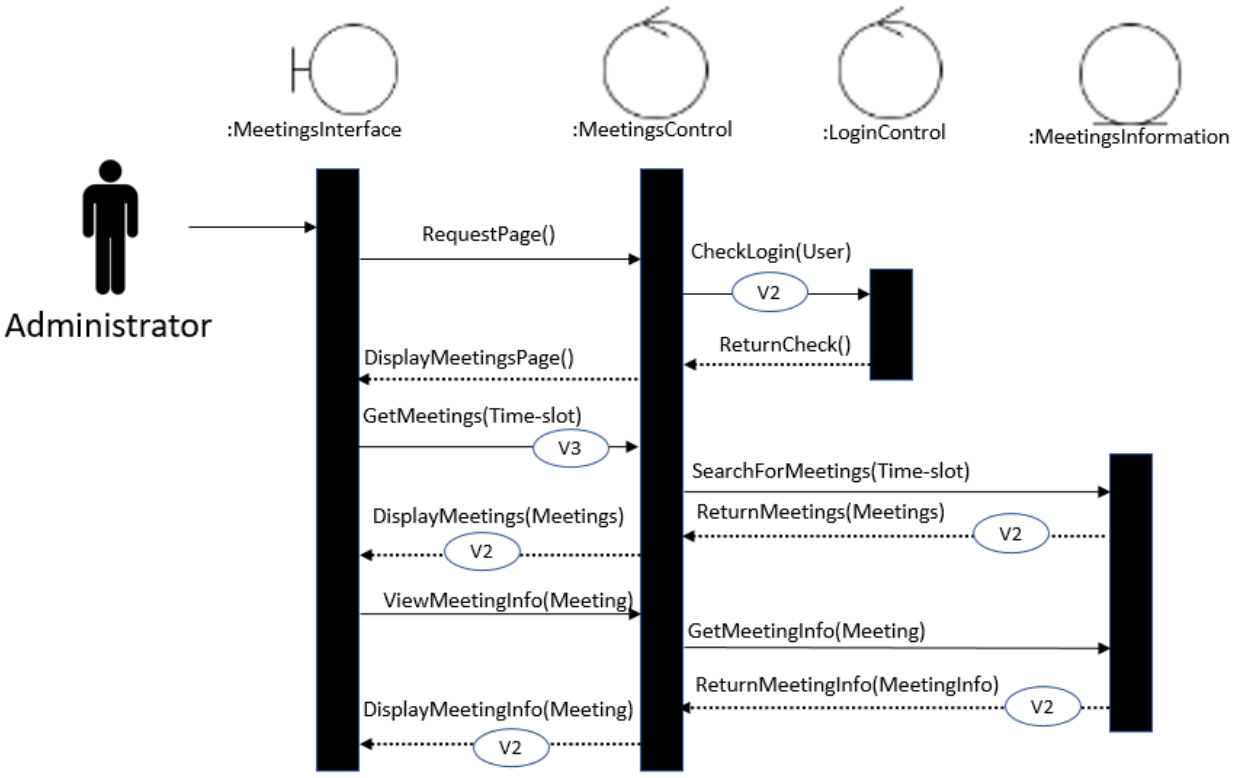

Figure 20: *Extended Sequence Diagram for Use-Case* Display meeting by timeslot

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
GetMeetings [V3]	Encryption for data transmitted.
ReturnMeetings [V2]	Encryption for data transmitted. Require authentication via CheckLogin. On authentication failure, the operation to retrieve the meetings information should stop.
DisplayMeetings [V2]	Require authentication, on authentication failure, the operation to display the meetings information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
ReturnMeetingInfo [V2]	Require authentication, on authentication failure, the operation to retrieve the specific meetings information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

DisplayMeetingInfo [V2]	Require authentication, on authentication failure, the operation to display the specific meetings information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.19 Administrator Use Case - Display meetings by meeting ID.

Description	Display meetings by meeting ID.
Actors	Administrator User
Trigger	The administrator clicks the "Search Meetings by meeting ID" button.
Scenario	1. (trigger) The administrator clicks on the "Search Meetings by meeting ID" button. 2. The administrator is prompted to type in the meeting ID 3. The administrator clicks "Display Meetings". 4. The system queries through the meetings in the meeting ID and displays the result. 5. The administrator can view a meeting's details from the queried results.
Alternative Scenario	3a. The administrator clicks the "Return" button. 4a. The system cannot find any results. 4a1: The system displays "No results found." 4b. The system has an error querying into the meetings database. 4b1: The system displays an error message regarding the exact issue.
Fail Case	The system has an error querying into the meetings database.
Consequence of Failure	The system will not retrieve any results even if there are matching results in the meetings database.
Associated Risks	There will be some frustration regarding the system's query retrievals.
Fail Exit State	The system shows the "No Result" found on the results page.

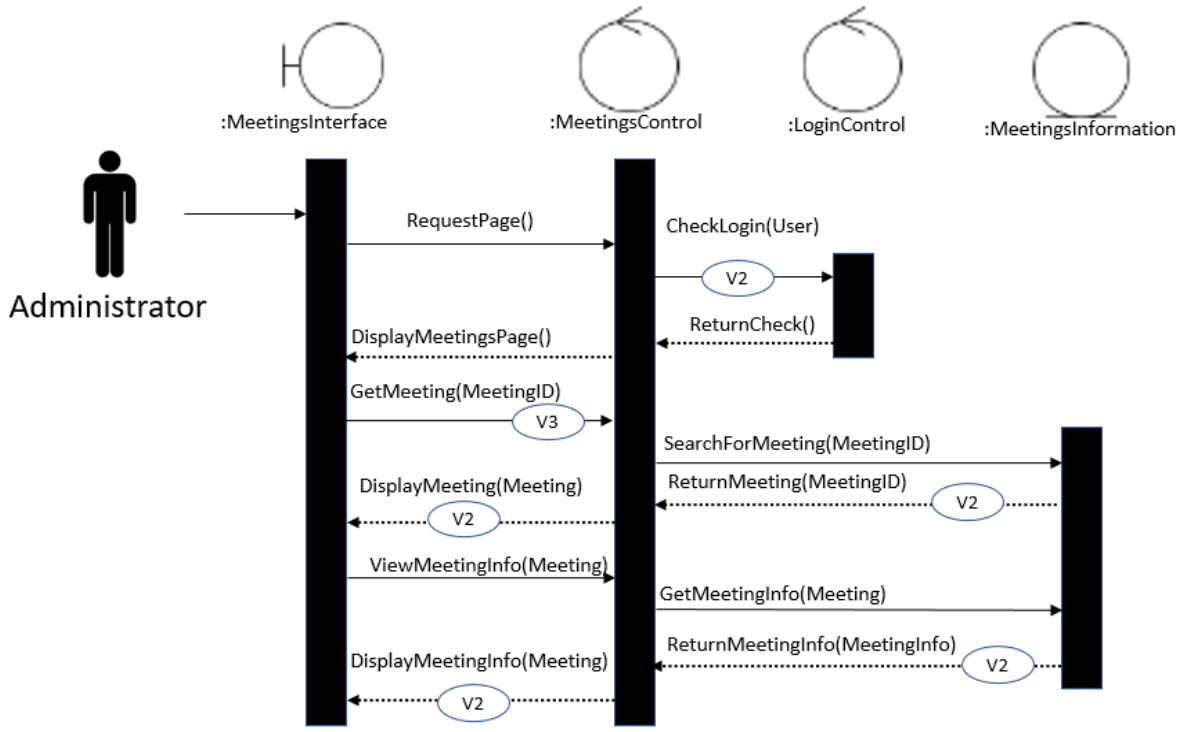

Figure 21: *Extended Sequence Diagram for Use-Case* Display meeting by timeslot

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
GetMeeting [V3]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the user profile information should stop. Encryption for data transmitted.
Return Meeting [V2]	Encryption for data transmitted.
DisplayMeeting [V2]	Require authentication via CheckLogin. On authentication failure, the operation to display the user profile information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
ReturnMeetingInfo [V2]	Encryption for data transmitted.
DisplayMeetingInfo [V2]	Require authentication via CheckLogin. On authentication failure, the operation to save the user profile information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.20 Administrator Use Case - Display meetings by a user.

Description	Display all the meetings attended by a single person
Actors	Administrator User
Trigger	The administrator clicks “Search Meetings by User”.
Scenario	1. (trigger) The administrator clicks on the “Search Meetings by User” button. 2. The administrator is prompted to type in the username or email of the client. 3. The administrator clicks “Display Meetings”. 4. The system queries through the database to return and display all meetings that a user is attending. 5. The administrator can view a meeting’s details from the queried results.
Alternative Scenario	4a. The system cannot find any results. 4a1: The system displays “No results found.”
Fail Case	The system has an error querying into the meetings database.
Consequence of Failure	The system is unable to display the meetings attended by selected clients.
Associated Risks	The administrator is unable to get up-to-date information for meetings attended by the client.
Fail Exit State	The system shows the “No Result” found on the results page.

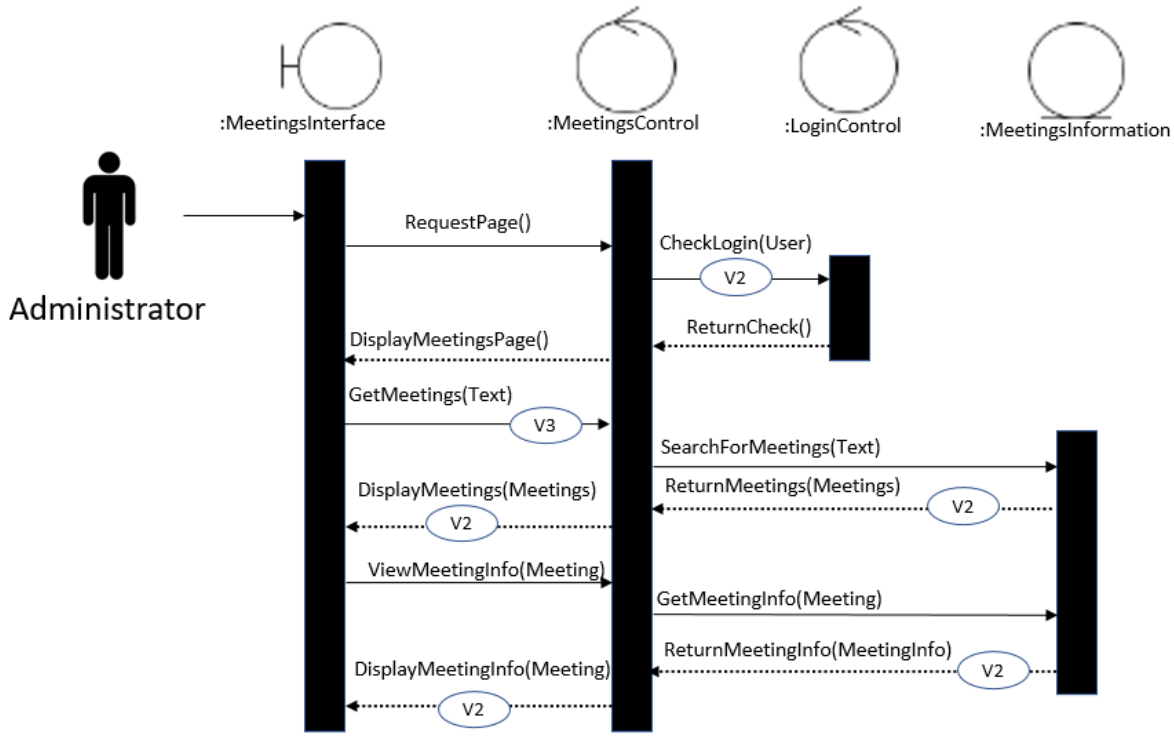

Figure 22: Extended Sequence Diagram for Use-Case Display meeting by User

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
GetMeetings [V3]	Require authentication via CheckLogin. On authentication failure, the operation to retrieve the meetings information should stop. Encryption for data transmitted.
ReturnMeetings [V2]	Encryption for data transmitted.
DisplayMeetings [V2]	Require authentication. On authentication failure, the operation to display the meetings information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
ReturnMeetingInfo [V2]	Require authentication. On authentication failure, the operation to retrieve the specific meeting information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

DisplayMeetingInfo [V2]	Require authentication. On authentication failure, the operation to display the specific meeting information should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.21 Administrator Use Case - Display meetings in a timeframe.

Description	Display meetings in a time frame (Can be week or day).
Actors	Administrator User
Trigger	The administrator clicks “Search Meetings by time-frame”.
Scenario	1. (trigger) The administrator clicks on the “Search Meetings by time-frame” button. 2. The administrator is prompted to type in the initial time and final time. 3. The administrator clicks “Display Meetings”. 4. The system queries the database to return the meetings in the selected timeframe and displays the result. 5. The administrator can view a meeting’s details from the queried results.
Alternative Scenario	4a. The system cannot find any results. 4a1: The system displays “No results found.”
Fail Case	The system has an error querying into the meetings database.
Consequence of Failure	The system is unable to display results to the administrator.
Associated Risks	The administrator is unable to get an overview of all the meetings happening in the selected timeframe.
Fail Exit State	The system shows the “No Result” found on the results page.

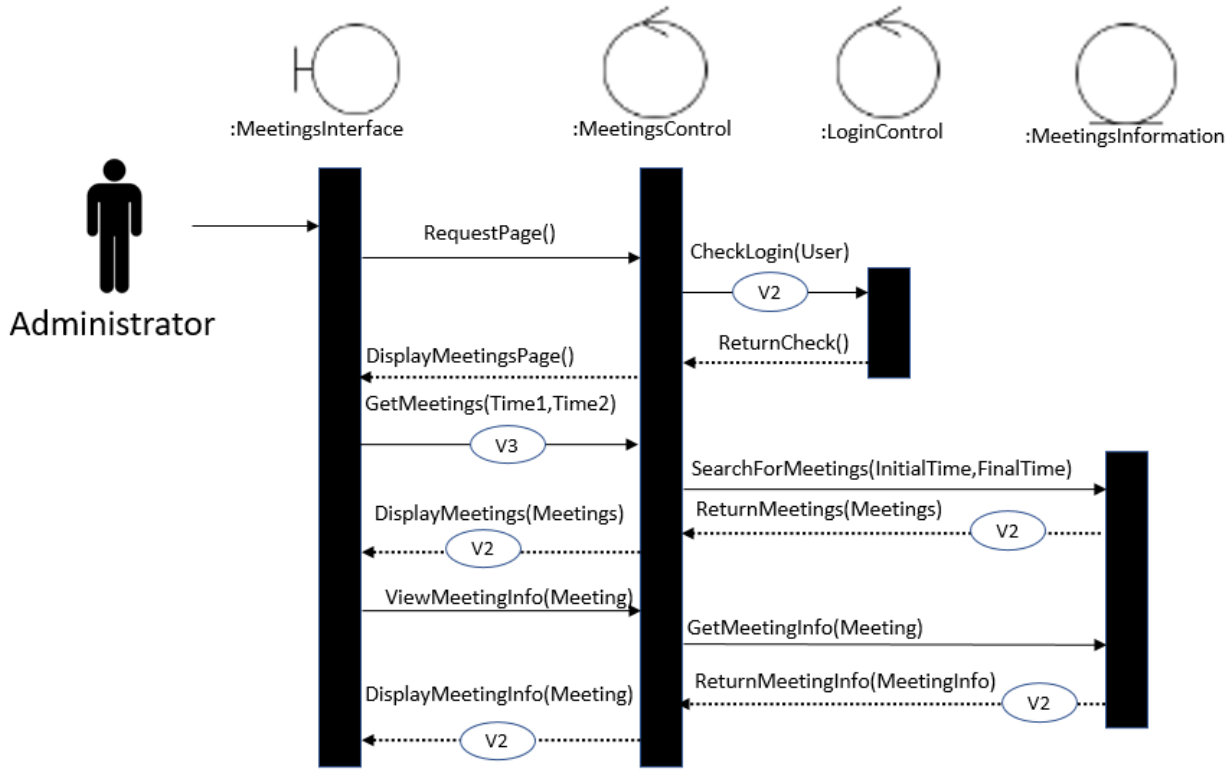

Figure 23: Extended Sequence Diagram for Use-Case Display meeting in selected timeframe

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
GetMeetings [V3]	Require authentication via CheckLogin. Encryption for data transmitted.
ReturnMeetings [V2]	Require authentication. On authentication failure, the operation to retrieve the meetings in the selected timeframe should stop. Encryption for data transmitted.
DisplayMeetings [V2]	Require authentication. On authentication failure, the operation to display all meetings in the selected timeframe should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.
ReturnMeetingInfo [V2]	Require authentication. On authentication failure, the operation to return specific meeting details should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

DisplayMeetingInfo [V2]	Require authentication. On authentication failure, the operation to display specific meeting details should stop. Note the authentication status can be maintained if the user stays on the same page. Encryption for data transmitted.

2.7.22 Administrator Use Case – Create Another Administrator Account

Description	Create an administrator account using the unique company email address
Actors	Administrator User
Trigger	Administrator navigates to Create Administrator Accounts page
Scenario	1. (trigger) The administrator navigates to the Create Administrator Accounts page. 2. The administrator enters the second administrator's company email address. 3. The system validates that the company email address entered exists in the administrator database and has not been registered before. 4. The administrator enters the second administrator's desired username. 5. The system validates that the username entered meets the required format, is unique, and not registered before. 6. The administrator enters the second administrator's password. 7. The system validates the password meets the security requirement format. 8. The administrator hits the create account button. 9. The system registered the account with the information supplied into the database.
Alternative Scenario	3a: The email address entered is not in the database: 3a1: The system informs the client that the email address entered is not in the database 3b: The email address entered already registered before: 3b1: The system informs the administrator that the email address entered was already registered before in the administrator database. 3b2: The administrator can choose to log in or request a password reset. 5a: The username entered already registered before 5a1: The system informs the user that the username has already been registered before 5a2: The user can choose to log in, request a password reset, or enter a new username. 6a: The password validation failed. 6a1: The system informs the user that the password validation failed. 6a2: The user can choose to enter a new password.

Fail Case	Administrator unable to create a new administrator account
Consequence of Failure	The new administrator is unable to use the system's administrator features.
Associated Risks	The new administrators may stop using the system.
Fail Exit State	On invalid input, the system stops creating a new administrator account and displays the warning containing the cause of the error to the users.

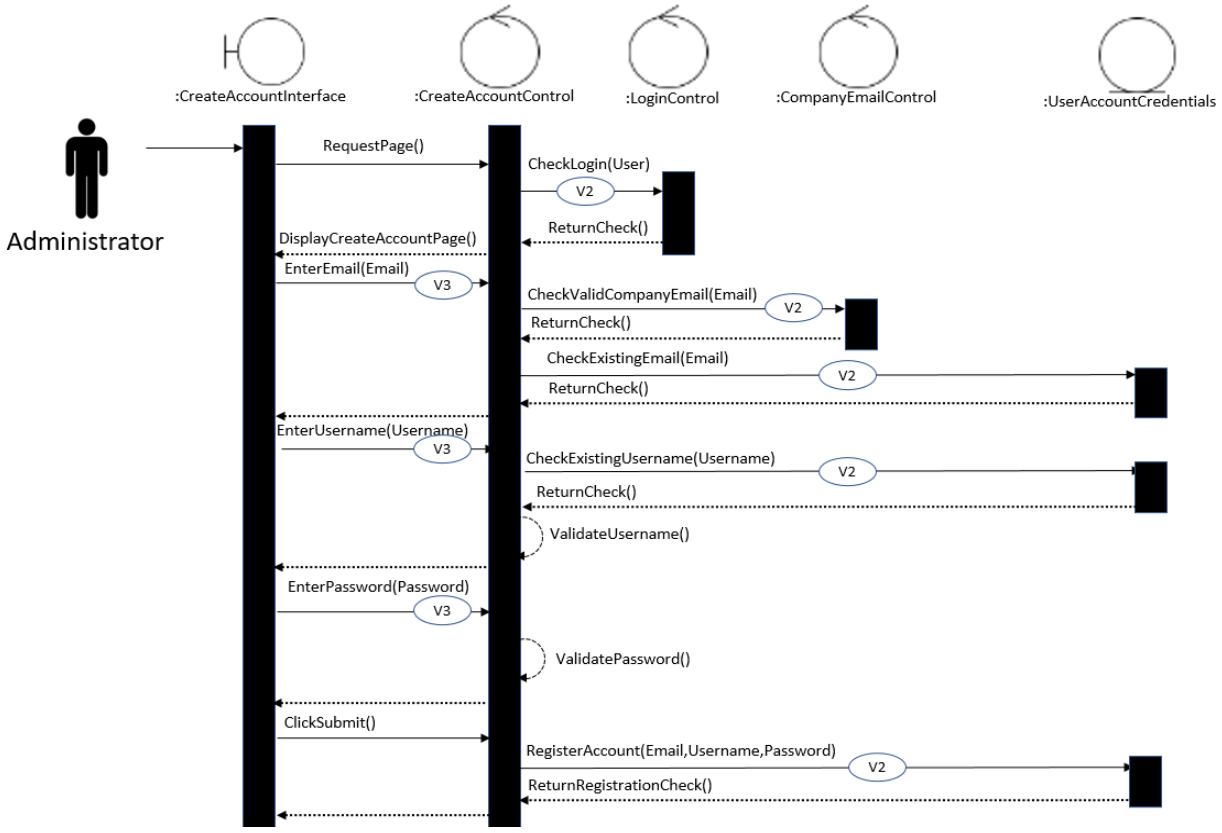

Figure 24: *Extended Sequence Diagram for Use-Case* – Create administrator account

Mitigation plan and Control for vulnerability identified in the extended sequence diagram.

Action [Vulnerability]	Mitigation Plan and Control
CheckLogin [V2]	Encryption for data transmitted.
EnterEmail [V3]	Encryption for data transmitted.

CheckValidCompanyEmail [V2]	Encryption for data transmitted.
CheckExistingEmail [V2]	Encryption for data transmitted.
EnterUsername [V3]	Encryption for data transmitted.
CheckExistingUsername [V2]	Encryption for data transmitted.
EnterPassword [V3]	Encryption for data transmitted.
RegisterAccount[V2]	Require authentication via CheckLogin. On authentication failure, the operation to register account should stop. Note the login status should be maintained if the administrator stays in the same page. Encryption for data transmitted.