

# System Requirements Specification

*MyStudyGroupPlanner*

*Version 1.1*

## **Client**

Katie Hirsch

## **Team 2**

Aparna V. Kaliappan

Ying Zhang

Siqi Lin

Sean Murren

Tyler Campbell

4/14/2016

# MyStudyGroupPlanner

## System Requirement Specification

### Table of Contents

	<u>Page</u>
1. Changes in this Document	
2. Introduction	
2.1. Purpose of This Document	
2.2. References	
2.3. Purpose of the Product	
2.4. Product Scope	
3. Functional Requirements	
3.1. Use Case 1 - Log into Application	
3.2. Use Case 2 - Create Account	
3.3. Use Case 3 - Create Group	
3.4. Use Case 4 - User Leaves a Group	
3.5. Use Case 5 - User Joins a Searched for Group	
3.6. Use Case 6 - User Joins a Group by Group ID	
3.7. Use Case 7 - Remove Users from Group	
3.8. Use Case 8 - Group Creator Changes Group Permission	
3.9. Use Case 9 - Admin Views Report	
3.10. Use Case 10 - Admin Adds Information	
3.11. Use Case 11 - Admin Searches for a User	
3.12. Use Case 12 - Admin Deletes a User's Account	
3.13. Use Case 13 - Admin Deactivates a User's Account	
3.14. Use Case 14 - Admin Resets User Password	
3.15. Use Case 15 - Admin Deletes All Groups	
3.16. Use Case 16 - Sign Out of the Application	
3.17. Use Case 17 - Schedule a Meeting in a Group	
3.18. Use Case 18 - Respond to a Meeting Request	
3.19. Use Case 19 - Submit a Report	
3.20. Use Case 20 - Chat Within a Group	
3.21. Use Case 21 - User Searches for a Group	
3.22. Use Case 22 - Send Message to Group Member	
3.23. Use Case 23 - User Views the Messages Received	
3.24. Use Case 24 - Admin Creates an Account	

- 3.25. Use Case 25 - User Starts a New Search
- 3.26. Use Case 26 - View Meetings
- 3.27. Use Case 27 - View Meeting Details
- 3.28. Use Case 28 - Send Message to Whole Group
- 3.29. Use Case 29 - Leave the Chat
- 3.30. Use Case 30 - Admin Deletes Information
- 3.31. Use Case 31 - Change a Meeting
- 3.32. Use Case 32 - Join a Video Chat
- 3.33. Use Case 33 - Create Profile
- 3.34. Use Case 34 - Edit Profile
- 3.35. Use Case 35 - Delete Group
- 3.36. Use Case 36 - User Joins a Closed Group
- 3.37. Use Case 37 - Handle Closed Group Request

#### 4. Non-Functional Requirements

#### 5. User Interface

#### 6. Deliverables

#### 7. Open Issues

Appendix A - Agreement Between Customer and Contractor

Appendix B - Team Review Sign-off

Appendix C - Document Contributions

## **1. Changes in this Document**

The priorities of some use cases have been adjusted. Appendices A and B have been added.

## **2. Introduction**

### **2.1. Purpose of This Document**

The purpose of this document is to provide a detailed description for the functionalities and necessary conditions of the MyStudyGroupPlanner web application. This document is intended for all of the users of this web application, the customer, Katie Hirsch, and the members of MyStudyGroupPlanner development team.

### **2.2. References**

1. MyStudyGroupPlanner User Interface Design Document

### **2.3. Purpose of the Product**

The purpose of the MyStudyGroupPlanner web application is to help students to form or join study groups, that are created by other students, for their current courses within a university. The group creators of each group are able to schedule meetings to study both in-person and online, and the group members can decide whether or not they are attending the meeting. Once the study groups have been formed they will also have the option to hire tutors to lead them through the study sessions that they have scheduled. The study groups can negotiate hourly rates for the tutoring service.

### **2.4. Product Scope**

The MyStudyGroupPlanner web applications only supports Google Chrome Browser, and consists of numerous use cases; namely, creating an account, logging into the application, creating a group, and joining a group. For further details on the uses cases please refer to figures 1-37 below.

Figure 1. High level use case context diagram overview

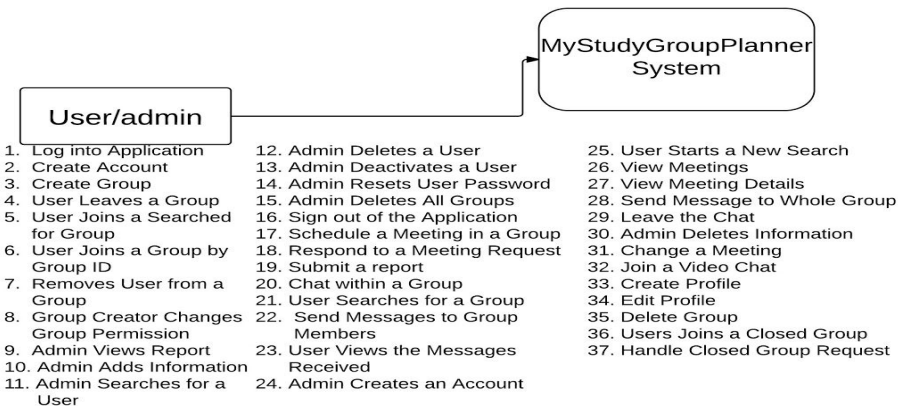


Figure 2. Use case 1, Log into application

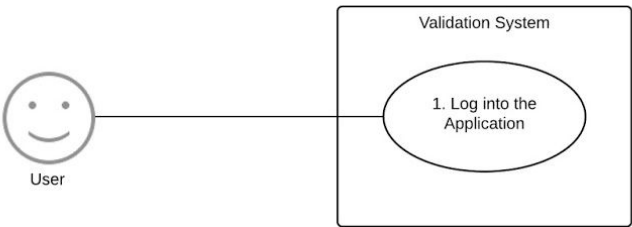


Figure 3. Use case 2, Create account

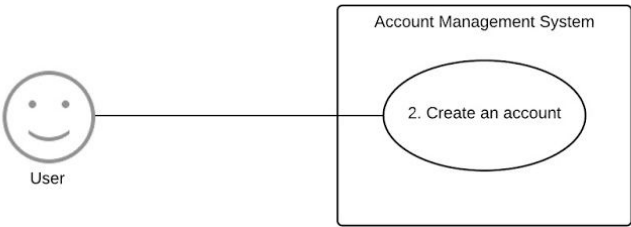
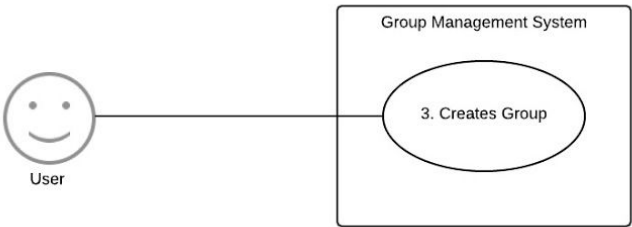
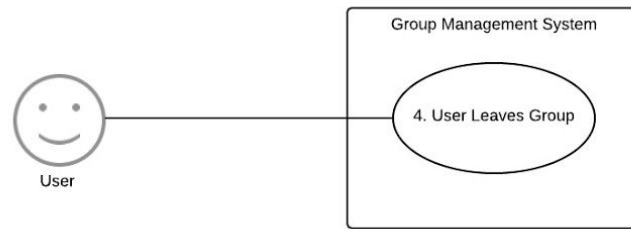


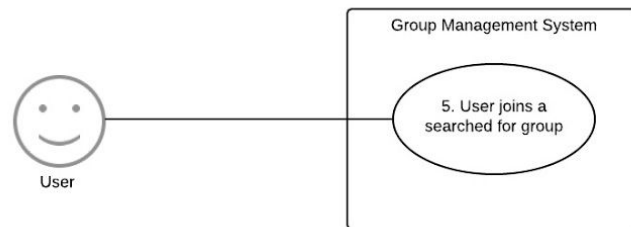
Figure 4. Use case 3, Create group



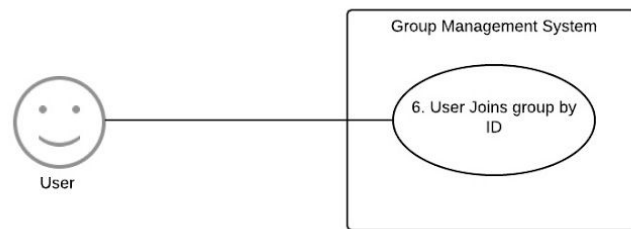
**Figure 5. Use case 4, User leaves a group**



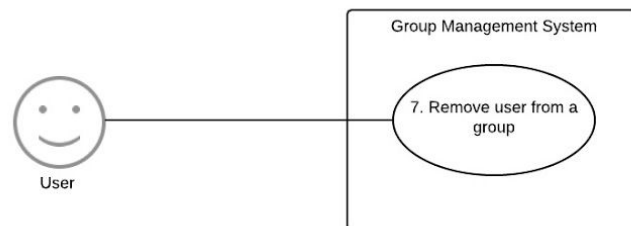
**Figure 6. Use case 5, User joins a searched for group**



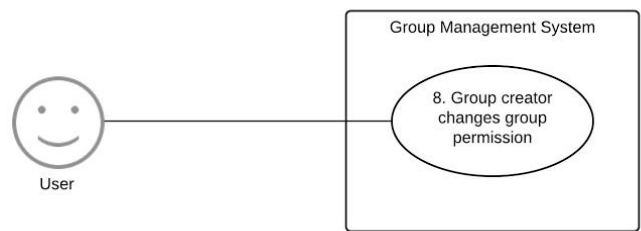
**Figure 7. Use case 6, User joins a group by group ID**



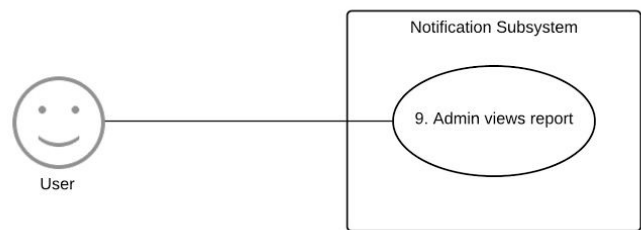
**Figure 8. Use case 7, Removes a user from a group**



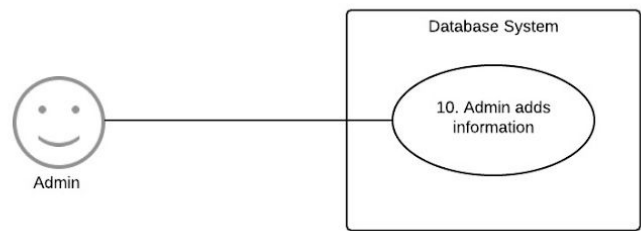
**Figure 9. Use case 8, Group creator changes group permission**



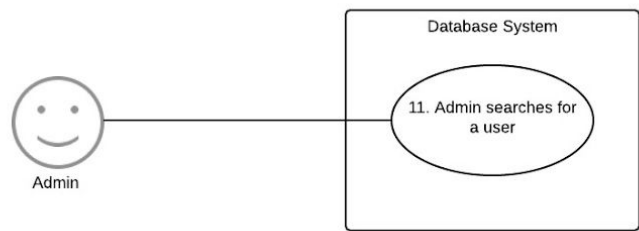
**Figure 10. Use case 9, Admin views report**



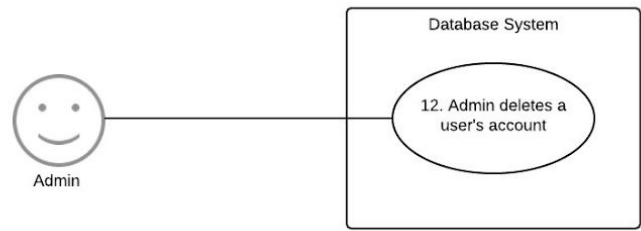
**Figure 11. Use case 10, Admin adds information**



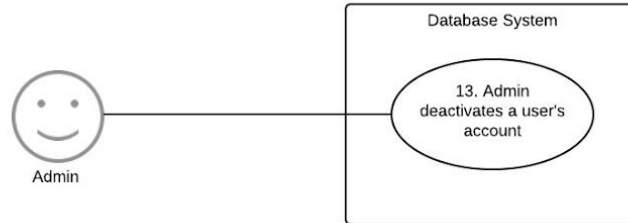
**Figure 12. Use case 11, Admin searches for a user**



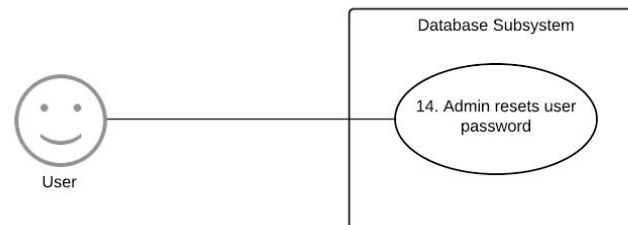
**Figure 13. Use case 12, Admin deletes a user's account**



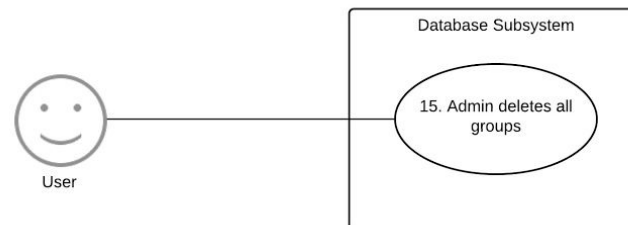
**Figure 14. Use case 13, Admin deactivates a user's account**



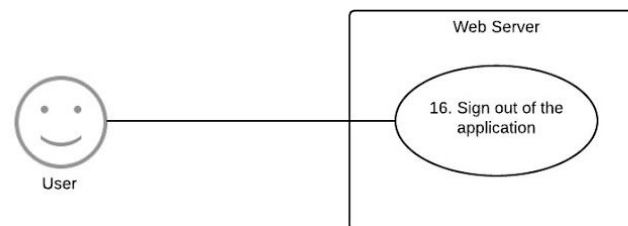
**Figure 15. Use case 14, Admin resets user password**



**Figure 16. Use case 15, Admin deletes all groups**

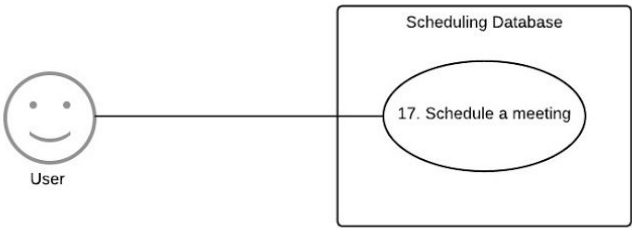


**Figure 17. Use case 16, Sign out of the application**

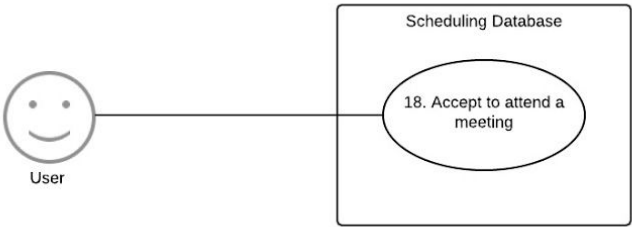




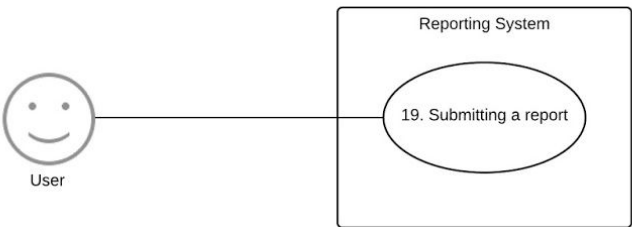
**Figure 18. Use case 17, Schedule a meeting**



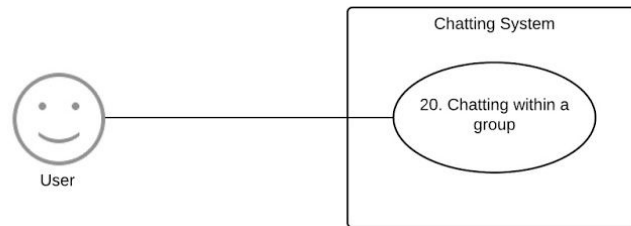
**Figure 19. Use case 18, Accept to attend a meeting**



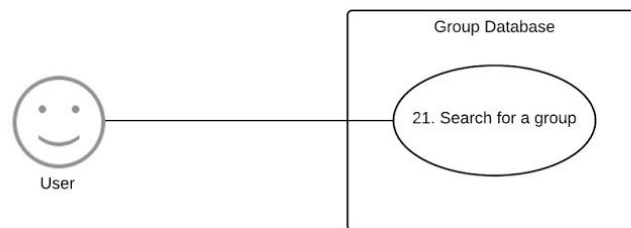
**Figure 20. Use case 19, Submit a report**



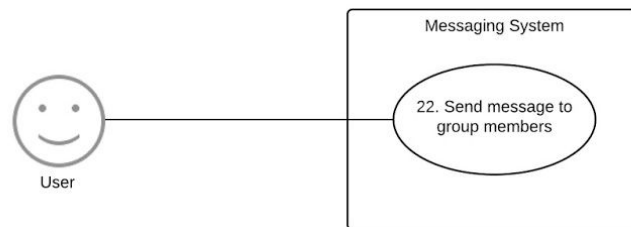
**Figure 21. Use case 20, Chat within a group**



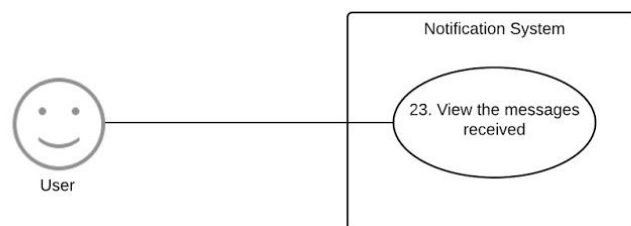
**Figure 22. Use case 21, User searches for a group**



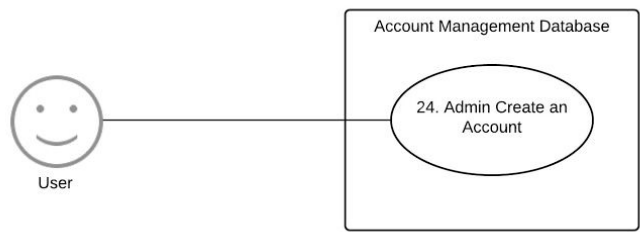
**Figure 23. Use case 22, Send messages to group member**



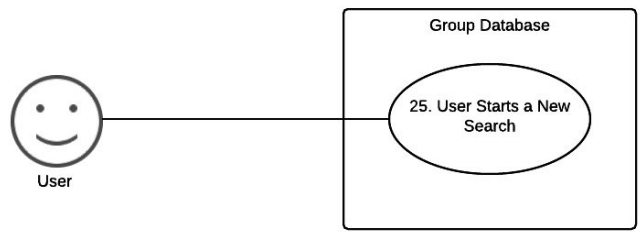
**Figure 24. Use case 23, User views the messages received**



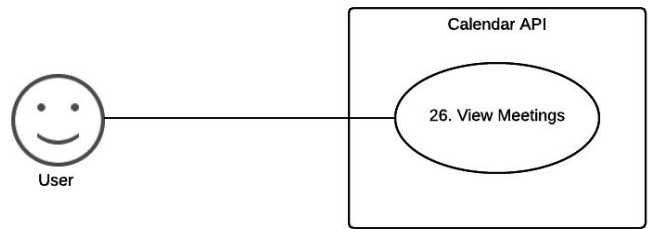
**Figure 25. Use case 24, Admin Creates an Account**



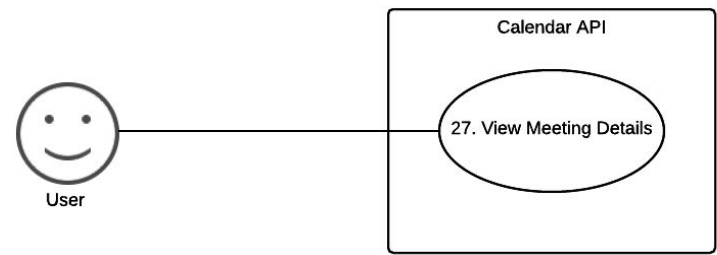
**Figure 26. Use case 25, User Starts a New Search**



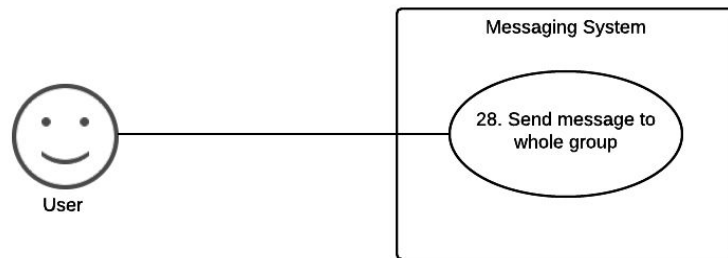
**Figure 27. Use case 26, View Meetings**



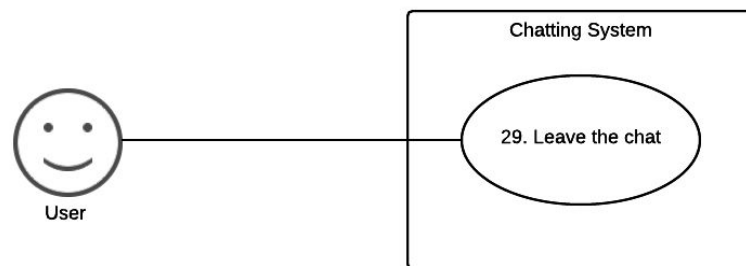
**Figure 28. Use case 27, View Meeting Details**



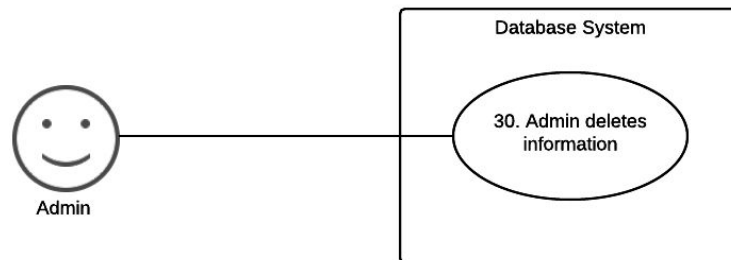
**Figure 29. Use case 28, Send Message to Whole Group**



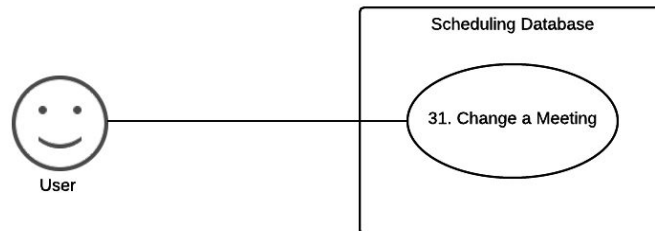
**Figure 30. Use case 29, Leave the Chat**



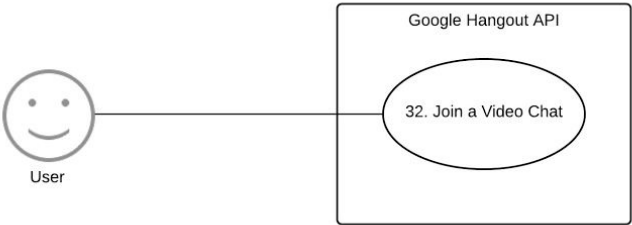
**Figure 31. Use case 30, Admin Deletes Information**



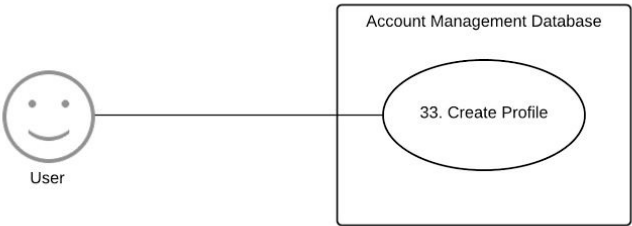
**Figure 32. Use case 31, Change a Meeting**



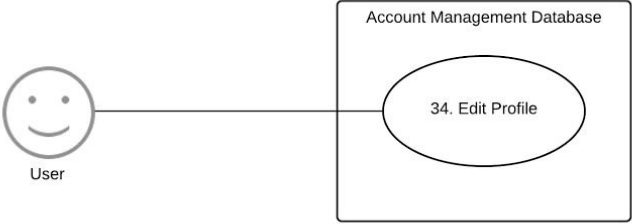
**Figure 33. Use case 32, Join a Video Chat**



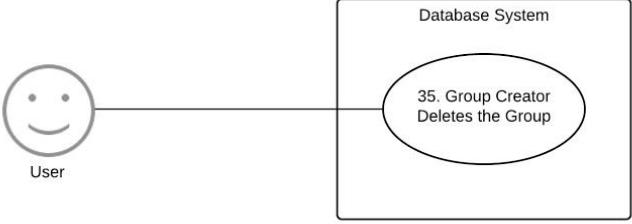
**Figure 34. Use case 33, Create Profile**



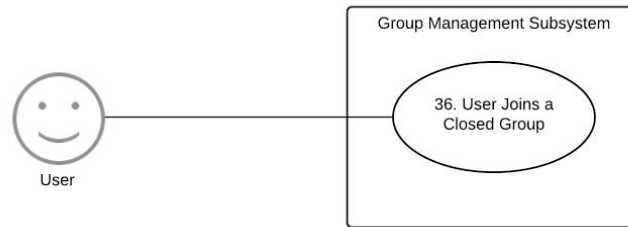
**Figure 35. Use case 34, Edit Profile**



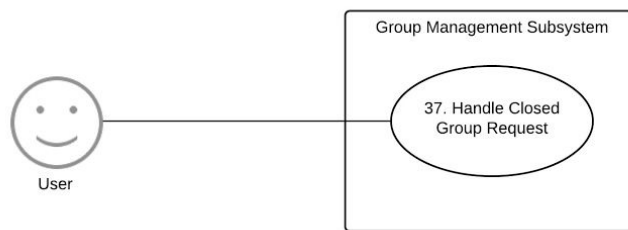
**Figure 36. Use case 35, Delete Group**



**Figure 37. Use case 36, User joins a Closed group**



**Figure 38. Use case 37, Handle closed group request**



### 3. Functional Requirements

#### 3.1. Use Case 1

<b>Number</b>	1	
<b>Name</b>	Log into Application	
<b>Summary</b>	The user/admin logs into the application	
<b>Priority</b>	1 (highest priority)	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>● The web service is up and running</li> <li>● Internet access</li> </ul>	
<b>Postconditions</b>	User/admin is logged in. Gets directed to application home page.	
<b>Primary Actor</b>	User/Admin	
<b>Secondary Actors</b>	<ul style="list-style-type: none"> <li>● Validation system</li> <li>● The main application</li> </ul>	
<b>Trigger</b>	Clicks the “Log In” button.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Types username in top box

	2	Types password in lower box
	3	Clicks “Log In” button.
	4	Validation system checks whether password is correct for the given username.
	5	Directs to the application home page
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	4a	Login Fails
	4b	System opens a popup stating error message.
	4c	User clicks the “Close” button.
	4d	Login page gets refreshed.
<b>Open Issues</b>	User forgets their password.	

### 3.2. Use Case 2

<b>Number</b>	2	
<b>Name</b>	Create Account	
<b>Summary</b>	The user creates an account	
<b>Priority</b>	1	
<b>Preconditions</b>	● The user does not already have an account	
<b>Postconditions</b>	● The user’s account is created	
<b>Primary Actor</b>	User	
<b>Secondary Actors</b>	● Validation system ● Account management system	
<b>Trigger</b>	User click the “Create account” button on the login page	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Types email in the email field (required).

	2	Types display name (required).
	3	Types First name and Last name (required).
	4	Types password (required).
	5	Confirms password (required).
	6	Clicks on the “Create Account” button
	7	Validation system checks that email has not been used by another existing user.
	8	Validation system checks that the password and confirmation password are the same.
	9	Directs to the “Create Profile” page.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	7a	The email has already been used as a username by another user.
	7b	Modal popup containing error message.
	7c	User selects another email.
	8a	Password and confirmation password don’t match.
	8b	Modal popup containing error message.
	8c	User retypes passwords.
<b>Open Issues</b>		

### 3.3. Use Case 3

<b>Number</b>	3
<b>Name</b>	Create group
<b>Summary</b>	The User creates a group
<b>Priority</b>	1 (highest priority)
<b>Preconditions</b>	● User is logged into the application



<b>Postconditions</b>	● A group is created	
<b>Primary Actor</b>	User	
<b>Secondary Actors</b>	Group management system Validation system	
<b>Trigger</b>	User clicks the “create group” button	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User selects a subject
	2	User selects a class
	3	User selects a class section
	4	User enters the maximum number of members (1-500)
	5	User selects if group is public or private
	6	User clicks the “Submit” button
	7	Validation system checks that all options are filled
	8	Validation system checks if max number of members is between 1-500
	9	User is redirected to the page of the group they just created
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	7a	popup is shown telling user the fields that still need to be filled in
	8a	popup is shown telling user max number of members is not within the 1-500 range
<b>Open Issues</b>	none	

### 3.4. Use Case 4

<b>Number</b>	4
<b>Name</b>	User leaves a group

<b>Summary</b>	A user leaves a group	
<b>Priority</b>	4	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>● A user is a member of a group</li> <li>● User is on the group page</li> </ul>	
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>● A user is no longer a member of a group</li> </ul>	
<b>Primary Actor</b>	User	
<b>Secondary Actors</b>	Group management subsystem	
<b>Trigger</b>	A user wants to leave a group	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User clicks the “Leave Group” button
	2	User selects the “yes” button on the “are you sure?” popup
	3	Group management subsystem removes the user from the group
	4	User is brought to the application main page
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	2a	User selects “cancel” button on the “are you sure?” popup
	2b	User is brought back to the group page
<b>Open Issues</b>	none	

### 3.5. Use Case 5

<b>Number</b>	5
<b>Name</b>	User joins a searched for group
<b>Summary</b>	A user joins a group
<b>Priority</b>	1 (highest)
<b>Preconditions</b>	A user has already searched for a group

<b>Postconditions</b>	A user has joined a group	
<b>Primary Actor</b>	User	
<b>Secondary Actors</b>	Group management subsystem Group permissions system	
<b>Trigger</b>	A user wants to join a group	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User finds the desired group from the search results
	2	User clicks the “join” button for their desired group
	3	User management system places user into group
	4	User is given lowest permission level for group
	5	User is redirected to the group page
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	1a	
<b>Open Issues</b>	none	

### 3.6. Use Case 6

<b>Number</b>	6
<b>Name</b>	User Joins a group by group-id
<b>Summary</b>	User joins group by group-id
<b>Priority</b>	4
<b>Preconditions</b>	User is on home page User knows the group of their desired group
<b>Postconditions</b>	User is placed into group
<b>Primary Actor</b>	User
<b>Secondary Actors</b>	Group management system Group permissions system

<b>Trigger</b>	User wants to join a group by group-id	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User clicks the “join group by id” button
	2	User enters the group-id
	3	User clicks “join” button
	3	User management system places user into group
	4	User is given lowest permission level for group
	5	User is redirected to the group page
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	3a	popup appears telling user “group-id is incorrect/ group does not exist”
<b>Open Issues</b>	none	

### 3.7. Use Case 7

<b>Number</b>	7
<b>Name</b>	Remove user from group
<b>Summary</b>	The user with removing permissions removes a different user from a group
<b>Priority</b>	4
<b>Preconditions</b>	A user has “remove member” permissions for a group A user is currently on the group page The user to be removed is currently a member of the group
<b>Postconditions</b>	User is removed from group
<b>Primary Actor</b>	User with “remove member” permissions
<b>Secondary Actors</b>	User who is a member of the group group management subsystem group permissions subsystem

<b>Trigger</b>	A user wants to remove a different user from the group	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User selects the group member they want to remove from the group members list
	2	User clicks the “remove from group” button
	3	User select “yes” button from the are you sure popup
	4	Group management system removes selected user from the group
	5	Selected user no longer shows up in the group members list
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	2a	User has incorrect permissions and the “remove from group” button is not present
<b>Open Issues</b>	none	

### 3.8. Use Case 8

<b>Number</b>	8
<b>Name</b>	Group Creator changes group permissions
<b>Summary</b>	The owner of the group modifies the permissions level of a member of the group. The permissions level can be increased or decreased.
<b>Priority</b>	5
<b>Preconditions</b>	User modifying permissions is the group’s owner User is currently on the group page User to have permissions modified is currently a member of the group
<b>Postconditions</b>	User’s permissions are modified
<b>Primary Actor</b>	User that is the group’s owner
<b>Secondary Actors</b>	group permissions subsystem
<b>Trigger</b>	The group owner wants to modify the permissions of a group member

Main Scenario	Step	Action
	1	The group owner selects a user from the group members list.
	2	The user's permissions button is clicked
	3	A popup window displays permission options for user
	4	A permission level is clicked
	5	The update button is clicked
	6	The popup window closes
	7	The group permissions subsystem updates and saves the new permission level of the user
Extensions	Step	Branching Action
	1a	After selecting a permission the group owner clicks the cancel button
	1b	The popup window closes
Open Issues	Can the group owner modify their own permissions?	

### 3.9. Use Case 9

Number	9
Name	Admin Views Report
Summary	An admin opens a report and reviews the report information
Priority	2
Preconditions	The admin is logged in The "View Report" page is open
Postconditions	A window is opened containing the report's information
Primary Actor	Admin user
Secondary Actors	Notification Subsystem

<b>Trigger</b>	The admin wants to open and look at a report	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks the view button next to a report they want to review
	2	A popup window opens and displays the report's information
	3	
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	1a	
<b>Open Issues</b>		

### 3.10. Use Case 10

<b>Number</b>	10	
<b>Name</b>	Admin Adds Information	
<b>Summary</b>	Admin can add information about classes taught in a semester, and room location availability, to the database.	
<b>Priority</b>	1 (highest priority)	
<b>Preconditions</b>	Admin must be logged into the application.	
<b>Postconditions</b>	New information is added to the application database.	
<b>Primary Actor</b>	Admin	
<b>Secondary Actors</b>	Database system	
<b>Trigger</b>	Admin clicks the "Add Information" button.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks the "Add Information" button.
	2	Database system displays the current class and room information for the semester.

	3	Admin fills in the available fields.
	4	Admin clicks the “Submit” button.
	5	Database system updates changes in database.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	4a	Admin clicks the “Cancel” button.
	5a	If the information is already in the database, displays error message.
<b>Open Issues</b>	Admin provides incorrect information about a class or room.	

### 3.11. Use Case 11

<b>Number</b>	11	
<b>Name</b>	Admin Searches for a User	
<b>Summary</b>	Admin can search for a user by username.	
<b>Priority</b>	4	
<b>Preconditions</b>	Admin must be logged into the application.	
<b>Postconditions</b>	Displays the user’s username, and provides options for resetting the user’s password, temporarily inactivating the user’s account, or permanently deleting the user’s account.	
<b>Primary Actor</b>	Admin	
<b>Secondary Actors</b>	Database system	
<b>Trigger</b>	Admin clicks the “Search for User” button.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks the “Search for User” button.
	2	Admin enters the user’s username.
	3	Database system searches for the user and displays the options for the admin.



Extensions	Step	Branching Action
	3a	If no user with that username exists, displays “User not found”.
Open Issues		

### 3.12. Use Case 12

<b>Number</b>	12	
<b>Name</b>	Admin Deletes a User’s Account	
<b>Summary</b>	Admin can permanently delete the user’s account.	
<b>Priority</b>	5	
<b>Preconditions</b>	Admin must be logged into the application. Admin must also be on the “Search for User” page, and must have searched for and found a specific user.	
<b>Postconditions</b>	User’s account gets permanently deleted.	
<b>Primary Actor</b>	Admin	
<b>Secondary Actors</b>	Database system Verification system	
<b>Trigger</b>	Admin clicks the “Delete User” button.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks the “Delete User” button.
	2	Database system will verify to make sure that the user still exists in the database.
	3	Verification system will display a confirmation popup “Are you sure you want to delete this user?”
	4	Admin clicks on the option “Yes”.
	5	Database system will delete the user’s information from the database.
	6	Admin gets redirected to the application homepage.

Extensions	Step	Branching Action
	4a	Admin clicks on the option “No”, and the popup closes.
Open Issues		

### 3.13. Use Case 13

Number	13	
Name	Admin Deactivates a User’s Account	
Summary	Admin can deactivate a user’s account.	
Priority	5	
Preconditions	Admin must be logged into the application. Admin must also be on the “Search for User” page, and must have searched for and found a specific user.	
Postconditions	User’s account gets deactivated.	
Primary Actor	Admin	
Secondary Actors	Database system	
Trigger	Admin clicks the “Inactive” button.	
Main Scenario	Step	Action
	1	Admin clicks the “Inactive” button.
	2	Database system will update the user’s account status in the database.
Extensions	Step	Branching Action
	1a	Admin wants to activate the account if it is already inactivated.
	1b	Admins clicks on the “Activate” button, and the database system updates the user’s account status in the database.
Open Issues		

### 3.14. Use Case 14

<b>Number</b>	14	
<b>Name</b>	Admin Resets User password	
<b>Summary</b>	The admin resets a user's password by deleting the current one	
<b>Priority</b>	5	
<b>Preconditions</b>	Admin must be logged in Admin must also be on the "Search for User" page Admin must have searched for and found a specific user	
<b>Postconditions</b>	A user's password is deleted	
<b>Primary Actor</b>	Admin user	
<b>Secondary Actors</b>	Database system	
<b>Trigger</b>	An admin receives a notification to reset a user's password	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks on user password text box
	2	Admin types in temporary password
	3	Admin clicks save button
	4	Database system updates user's password information
	5	Search result window closes
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	1a	Admin clicks cancel button
	1b	Search result window closes
<b>Open Issues</b>		

### 3.15. Use Case 15

<b>Number</b>	15	
<b>Name</b>	Admin Deletes all groups	
<b>Summary</b>	An admin deletes all groups from the database at the end of a semester	
<b>Priority</b>	5	
<b>Preconditions</b>	Admin is logged in Admin must also be on the "Delete" page	
<b>Postconditions</b>	All groups are deleted from the database	
<b>Primary Actor</b>	Admin user	
<b>Secondary Actors</b>	Database subsystem	
<b>Trigger</b>	An admin wants to delete all groups from the database	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks the "delete all groups" button
	2	A confirmation popup window opens
	3	Admin clicks yes button
	4	Confirmation popup window closes
	5	The database subsystem deletes all groups from the database
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	1a	Admin clicks no button in the confirmation window
	1b	Confirmation window closes
<b>Open Issues</b>	Scope of groups admin is able to delete	

### 3.16. Use Case 16

<b>Number</b>	16
---------------	----

<b>Name</b>	Sign out of the Application	
<b>Summary</b>	The user/admin signs of the application	
<b>Priority</b>	1	
<b>Preconditions</b>	The user/admin is logged in	
<b>Postconditions</b>	The user/admin successfully signs out of the application	
<b>Primary Actor</b>	user/admin	
<b>Secondary Actors</b>	Web Server	
<b>Trigger</b>	The user clicks on the “Sign out” button	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	The user/admin moves their mouse to their named located on the upper right hand side of the page
	2	The user/admin click on the “sign out” button
	3	The system logs user/admin out of the application
	4	Redirects to the main page
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>		

### 3.17. Use Case 17

<b>Number</b>	17
<b>Name</b>	Schedule a Meeting in a Group
<b>Summary</b>	The group leader schedules a meeting for the group
<b>Priority</b>	1
<b>Preconditions</b>	The group leader is on the group page

<b>Postconditions</b>	A meeting has been successfully scheduled by the group leader	
<b>Primary Actor</b>	The group leader	
<b>Secondary Actors</b>	Scheduling Database	
<b>Trigger</b>	The group leader clicks on the “Create” button	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	The group leader clicks on the “Create a meeting” button
	2	A modal popup opens
	3	The group leaders selects a date for the meeting
	4	The group leader selects a start time for the meeting
	5	The group leader selects an end time for the meeting
	6	The group leader selects a location for the meeting
	7	The group leader adds notes for the meeting
	8	The scheduling system creates the meeting for the group
	9	The system adds the meeting the group calendar
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>		

### 3.18. Use Case 18

<b>Number</b>	18
<b>Name</b>	Respond to a Meeting Request
<b>Summary</b>	The user accepts to attend the meeting scheduled by the group leader.
<b>Priority</b>	4
<b>Preconditions</b>	The user is logged in

<b>Postconditions</b>	The user has accepted to attend this meeting	
<b>Primary Actor</b>	Members of the group	
<b>Secondary Actors</b>	Scheduling Database	
<b>Trigger</b>	The user clicks on the “Yes” button	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	The user clicks on the meeting date in the calendar on the main page
	2	A modal popup opens, asking user whether or not they are going to attend the meeting
	3	The user clicks on the “Yes” button to attend the meeting
	4	The scheduling system counts of how many members are attending the meeting to ensure that it does not exceed the maximum capacity of the meeting location
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	3a	The user clicks on the “No” button to not attend the meeting
<b>Open Issues</b>		

### 3.19. Use Case 19

<b>Number</b>	19
<b>Name</b>	Submit a Report
<b>Summary</b>	The user submits a report of other group members to the admin
<b>Priority</b>	3
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>● the user is logged in</li> <li>● the user is on the group page</li> </ul>
<b>Postconditions</b>	A report has been successfully submitted to the admin

<b>Primary Actor</b>	user	
<b>Secondary Actors</b>	Reporting System	
<b>Trigger</b>	The user clicks on the “Submit” button on the Report modal popup	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Under the List of Members, the user clicks on the “Report” option besides the name of the group member that they want to report
	2	The “Report” modal popup opens
	3	The user enter the reportees’ name in the Reportee field
	4	The user selects a report type from a drop down list in the Report Type field
	5	The user writes reasons of this report in the Comment field
	6	The user clicks on the “Submit” button
	7	The reporting system sends the report to the admin
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>		

### 3.20. Use Case 20

<b>Number</b>	20
<b>Name</b>	Chat within a group
<b>Summary</b>	The users in a group can communicate on their group page
<b>Priority</b>	2
<b>Preconditions</b>	The user has joined a group
<b>Postconditions</b>	The user sends/receives a message in the group chat room
<b>Primary Actor</b>	All users in a group



<b>Secondary Actors</b>	Chatting system	
<b>Trigger</b>	A user initiates a chat by sending a message in the group chat room	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Types in a message in the group message room
	2	Clicks the “Send” button to send a message to all users in a group
	3	The message is displayed on the group chat room to all the users who are currently on the group page
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	2a	The message cannot be sent
	2b	Error message is shown to the sender
<b>Open Issues</b>		

### 3.21. Use Case 21

<b>Number</b>	21	
<b>Name</b>	User Searches for a group	
<b>Summary</b>	The user searches for a group	
<b>Priority</b>	1	
<b>Preconditions</b>	The user is logged in to the application	
<b>Postconditions</b>	Search list is displayed	
<b>Primary Actor</b>	The user who searches for the group	
<b>Secondary Actors</b>	Group database	
<b>Trigger</b>	The user clicks “Search” button	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>

	1	A modal popup displays the search criteria
	2	Enters the search criteria
	3	Enter the group code
	4	Click “Search” Button
	5	The list of search results is displayed on a new page
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	2a	Selects the subject from the dropdown menu
	2b	Selects the class from the dropdown menu
	2c	Selects the section from the dropdown menu
	4a	Click “Cancel” button
	4b	Modal is closed and resumes back to the home page
<b>Open Issues</b>		

### 3.22. Use Case 22

<b>Number</b>	22
<b>Name</b>	Send message to group member
<b>Summary</b>	The user sends a message to other group members
<b>Priority</b>	5
<b>Preconditions</b>	The user is on the group page
<b>Postconditions</b>	<ul style="list-style-type: none"> <li>● A message is sent to a user</li> <li>● The receiver of the message receives the message</li> </ul>
<b>Primary Actor</b>	The sender/receiver of the message
<b>Secondary Actors</b>	Messaging System
<b>Trigger</b>	The user clicks the “Notify User” button

Main Scenario	Step	Action
	1	A modal popup for messages overlays the group page
	2	Types in the message
	3	Clicks "Send" button
	4	The receiver receives the message
Extensions	Step	Branching Action
	3a	Unable to send the message
	3b	An error is displayed on the modal
Open Issues		

### 3.23. Use Case 23

<b>Number</b>	23	
<b>Name</b>	User Views the messages received	
<b>Summary</b>	The receiver of the message views the message on a the notification page	
<b>Priority</b>	5	
<b>Preconditions</b>	The receiver has received an message from another user	
<b>Postconditions</b>	The receiver views the message received	
<b>Primary Actor</b>	Message Receiver	
<b>Secondary Actors</b>	Notification system	
<b>Trigger</b>	The user opens up the notification center	
Main Scenario	Step	Action
	1	Clicks on the notification icon on the top right hand corner
	2	Clicks on the unread message on the notification page

	3	Views the unread message
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	2a	No unread message
	2b	Displays a list of past conversations
<b>Open Issues</b>		

### 3.24. Use Case 24

<b>Number</b>	24	
<b>Name</b>	Admin Creates an account	
<b>Summary</b>	The admin is able to create an Admin account on the application.	
<b>Priority</b>	1	
<b>Preconditions</b>	The admin does not already have an account. Admin has a username and password already given to him/her.	
<b>Postconditions</b>	The admin's account is created	
<b>Primary Actor</b>	Admin	
<b>Secondary Actors</b>	● Account management system	
<b>Trigger</b>	Admin clicks the "Create account" button on the login page	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Types special email in the email field (required).
	2	Types password (required).
	3	Clicks on the "Create Account" button
	4	Account management system verifies input
	5	Directs to the application login page.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>

<b>Open Issues</b>	
--------------------	--

### 3.25. Use Case 25

<b>Number</b>	25	
<b>Name</b>	User Starts a New Search	
<b>Summary</b>	The user searches to find another user and view his/her profile.	
<b>Priority</b>	4	
<b>Preconditions</b>	The user is logged in to the application	
<b>Postconditions</b>	Search list is displayed	
<b>Primary Actor</b>	User who is searching	
<b>Secondary Actors</b>	Group database	
<b>Trigger</b>	The user clicks the “Search” button	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	A modal popup displays the search criteria
	2	Enters the search criteria
	3	Click “Search” Button
	4	Search request processed in group database.
	5	The list of search results is displayed on a new page.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	3a	Click “Cancel” button
	3b	Modal is closed and resumes back to the home page
<b>Open Issues</b>		

### 3.26. Use Case 26

<b>Number</b>	26
---------------	----

<b>Name</b>	View meetings	
<b>Summary</b>	The user can view upcoming meetings on the calendar.	
<b>Priority</b>	2	
<b>Preconditions</b>	The user is logged in to the application. The user is either on the homepage or is in the main page for a group he/she is a member of .	
<b>Postconditions</b>	View meetings on the calendar	
<b>Primary Actor</b>	User	
<b>Secondary Actors</b>	Calendar API	
<b>Trigger</b>	The user clicks the left and right buttons on the calendar.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Users clicks the left button to view meetings in the previous month.
	2	Calendar API processes the request and changes the calendar screen to the previous month.
	3	Users clicks the right button to view meetings in the next month.
	4	Calendar API processes the request and changes the calendar screen to the next month.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>		

### 3.27. Use Case 27

<b>Number</b>	27
<b>Name</b>	View meeting details
<b>Summary</b>	The user can view the details about upcoming meetings on the calendar, such as the date, time, and location.
<b>Priority</b>	2

<b>Preconditions</b>	The user is logged in to the application. The user is either on the homepage or is in the main page for a group he/she is a member of.	
<b>Postconditions</b>	View meeting details on the calendar	
<b>Primary Actor</b>	User	
<b>Secondary Actors</b>	Calendar API Meetings Database	
<b>Trigger</b>	The user clicks on a specific date containing a meeting.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User clicks on a highlighted date.
	2	Calendar API processes the request.
	3	Meetings database collects the data for the meeting.
	4	Details for the meeting are displayed under the calendar.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	1a	User clicks on a date with no meeting. "No meetings on this date" will be displayed.
<b>Open Issues</b>		

### 3.28. Use Case 28

<b>Number</b>	28
<b>Name</b>	Send message to whole group
<b>Summary</b>	The user can send a message to all members in the group.
<b>Priority</b>	4
<b>Preconditions</b>	The user is on the group page, and has the permission to send group messages.
<b>Postconditions</b>	All members in the group will receive the message.
<b>Primary Actor</b>	The sender of the message
<b>Secondary</b>	Messaging System

<b>Actors</b>		
<b>Trigger</b>	The user clicks the “Send group message” button	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	A modal popup for messages overlays the group page
	2	Types in the message
	3	Clicks “Send” button
	4	All members receive the message.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	3a	Unable to send the message
	3b	An error is displayed on the modal
<b>Open Issues</b>		

### 3.29. Use Case 29

<b>Number</b>	29	
<b>Name</b>	Leave the chat	
<b>Summary</b>	User can leave the chat after he/she is done.	
<b>Priority</b>	5	
<b>Preconditions</b>	The user has joined a group and is on the group page.	
<b>Postconditions</b>	The user leaves the chat.	
<b>Primary Actor</b>	User leaving the chat	
<b>Secondary Actors</b>	Chatting system	
<b>Trigger</b>	User leaves the group page.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User clicks the button to return to the application’s homepage.



	2	Chatting system updates that the user has left.
	3	User is redirected to the homepage.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>		

### 3.30. Use Case 30

<b>Number</b>	30	
<b>Name</b>	Admin Deletes Information	
<b>Summary</b>	Admin can delete information in the database about classes taught in a semester, and room location availability.	
<b>Priority</b>	4	
<b>Preconditions</b>	Admin must be logged into the application.	
<b>Postconditions</b>	Information gets deleted from the database.	
<b>Primary Actor</b>	Admin	
<b>Secondary Actors</b>	Database system	
<b>Trigger</b>	Admin clicks the “Delete Information” button.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks the “Delete Information” button.
	2	Database system displays the current class and room information for the semester.
	3	Admin clicks on the database item to delete.
	4	Admin clicks the “Delete” button.
	5	Database system updates changes in database.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	4a	Admin clicks the “Cancel” button.

<b>Open Issues</b>	Admin accidentally deletes the wrong piece of information about a class or room.
--------------------	--

### 3.31. Use Case 31

<b>Number</b>	31	
<b>Name</b>	Change a Meeting	
<b>Summary</b>	The group creator changes a meeting's time/date/location.	
<b>Priority</b>	4	
<b>Preconditions</b>	The group leader is on the group page	
<b>Postconditions</b>	The meeting's details are changed by the group leader. A notification of the change is sent as a message to all group members.	
<b>Primary Actor</b>	The group leader	
<b>Secondary Actors</b>	Scheduling Database Validation system	
<b>Trigger</b>	The group leader clicks on the "Edit meeting" button	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	The group leader clicks on the "Edit meeting" button
	2	A modal popup opens
	3	The group leader changes some of the fields and clicks "Submit".
	4	The validation system checks that the fields have reasonable input.
	5	The scheduling system creates the meeting for the group
	6	The system adds the meeting the group calendar
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	4a	User entered a end date/time that was smaller than the start date/time.
	4b	An error message is displayed.

<b>Open Issues</b>	
--------------------	--

### 3.32. Use Case 32

<b>Number</b>	32	
<b>Name</b>	Join a video chat	
<b>Summary</b>	The users in a group can join a video chat through Google Hangouts.	
<b>Priority</b>	5	
<b>Preconditions</b>	The user has joined a group and is on the group's page.	
<b>Postconditions</b>	The user is part of a video chat on Google Hangouts.	
<b>Primary Actor</b>	User joining the video chat	
<b>Secondary Actors</b>	Google Hangouts API	
<b>Trigger</b>	The user clicks the Google Hangouts link.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	The user clicks the Google Hangouts link on the group page.
	2	The Google Hangouts API processes the user's request.
	3	The user can begin video chatting.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	2a	The connection fails to be created.
	2b	Error message is shown to the user.
<b>Open Issues</b>		

### 3.33. Use Case 33

<b>Number</b>	33
<b>Name</b>	Create Profile
<b>Summary</b>	The user creates a profile.

<b>Priority</b>	3	
<b>Preconditions</b>	The user has created an account.	
<b>Postconditions</b>	The user's profile is created.	
<b>Primary Actor</b>	User	
<b>Secondary Actors</b>	<ul style="list-style-type: none"> <li>● Validation system</li> <li>● Account management system</li> </ul>	
<b>Trigger</b>	User clicks the "Create Profile" button after creating an account.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Types list of classes that he/she is taking that semester (required).
	2	Types a brief summary about himself/herself (optional).
	3	Clicks on the "Create Profile" button
	4	Directs to the application login page.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>		

### 3.34. Use Case 34

<b>Number</b>	34
<b>Name</b>	Edit Profile
<b>Summary</b>	The user edits their profile.
<b>Priority</b>	4
<b>Preconditions</b>	The user has already created a profile.
<b>Postconditions</b>	The user's profile gets edited.
<b>Primary Actor</b>	User
<b>Secondary Actors</b>	<ul style="list-style-type: none"> <li>● Account management database</li> </ul>

<b>Trigger</b>	User clicks the “Edit Profile” button after clicking on their name on the application header.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User clicks on his/her name on the application header.
	2	User makes changes to the profile.
	3	User clicks on the “Update” button.
	4	These changes are updated in the database.
	5	User returns to the homepage.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	3a	User clicks on the “Cancel” button. Changes are not saved.
<b>Open Issues</b>		

### 3.35. Use Case 35

<b>Number</b>	35	
<b>Name</b>	Delete group	
<b>Summary</b>	Group creator deletes the group.	
<b>Priority</b>	4	
<b>Preconditions</b>	Group creator must be on the page of the group to be deleted.	
<b>Postconditions</b>	The group gets deleted from the system.	
<b>Primary Actor</b>	Group creator user	
<b>Secondary Actors</b>	Database system Verification system Notification system	
<b>Trigger</b>	Admin clicks the “Delete Group” button.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks the “Delete Group” button.

	2	Database system will verify to make sure that the group still exists in the database.
	3	Verification system will display a confirmation popup “Are you sure you want to delete this group?”
	4	Admin clicks on the option “Yes”.
	5	Notification system will send a message to all group members about the deletion of the group.
	5	Database system will delete the group’s information from the database.
	6	User gets redirected to the application homepage.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	4a	User clicks on the option “No”, and the popup closes.
<b>Open Issues</b>		

### 3.36. Use Case 36

<b>Number</b>	36	
<b>Name</b>	User joins a Closed group	
<b>Summary</b>	A user joins a searched for group that is closed.	
<b>Priority</b>	4	
<b>Preconditions</b>	User has already searched for groups.	
<b>Postconditions</b>	User joins the closed group.	
<b>Primary Actor</b>	User	
<b>Secondary Actors</b>	Group management subsystem	
<b>Trigger</b>	User clicks the “Request” button next to a closed group on the search page.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	User decides on a closed group in the search results.

	2	User clicks the “Request” button.
	3	The subsystem sends a message to the group’s creator.
	4	Acceptance/rejection notification sent to the user.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>	none	

### 3.37. Use Case 37

<b>Number</b>	37	
<b>Name</b>	Handle closed group request	
<b>Summary</b>	Group creator must accept/reject a request to join the closed group.	
<b>Priority</b>	5	
<b>Preconditions</b>	Group creator has opened the request message.	
<b>Postconditions</b>	Group creator accepts/rejects the request.	
<b>Primary Actor</b>	Group creator user	
<b>Secondary Actors</b>	Group management subsystem	
<b>Trigger</b>	Creator clicks the “Accept” or “Reject” button.	
<b>Main Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Creator reads the message and fills in the optional message box.
	2	Creator clicks the “Accept” button and clicks “Send”.
	3	Requesting user gets added to the group.
	4	Acceptance notification sent to the user.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	2a	Group creator clicks the “Reject” button and clicks “Send”.
	2b	Rejection notification sent to the user.

<b>Open Issues</b>	none
--------------------	------



#### 4. Non-Functional Requirements

Requirement number	Item	Priority 1(highest) to 5 (highest)
1	The database should be secured	1
2	Reliable environment	1
3	Application is user-friendly	2
4	All code should follow a coding standard	3
5	Use the spiral software development process	1
6	The web service should always be available	1
7	Passwords are encrypted	2
8	Must use the AngularJS framework	2
9	Must use the MVC architecture	4
10	Must maintain page state when a page is refreshed	1
11	Guaranteed to work in Google Chrome	2
12	Application supports a minimum resolution of 1024x768	2

#### 5. User Interface

See “User Interface Design Document for MyStudyGroupPlanner”.

#### 6. Deliverables

Hard copies of each of the following:

- Systems Requirement Specification

- System Design Document
- UI Design Document
- Testing Report
- Code Inspection Report

Electronic copies via email of the following:

- Systems Requirement Specification
- System Design Document
- UI Design Document
- Testing Report
- Code Inspection Report

## **7. Open Issues**

## **Appendix A - Agreement Between Customer and Contractor**

The customer and the development team agree that the MyStudyGroupPlanner application performs all of the high-priority functionalities described in this System Requirements Specification. Use cases with lower priority will be addressed during future development of this application.

In the case that there are any future changes to the requirements outlined in this document, the development team will make these changes and provide the customer with updated hardcopy and softcopy versions to be read, approved, and signed.

### **Customer Comments:**

### **Customer Signature**

---

Print Name

---

Date

---

Signature

**Development Team Signatures**

\_\_\_\_\_  
Print Name

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature

\_\_\_\_\_  
Print Name

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature

\_\_\_\_\_  
Print Name

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature

\_\_\_\_\_  
Print Name

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature

\_\_\_\_\_  
Print Name

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature

## Appendix B - Team Review Sign-off

All members of the MyStudyGroupPlanner development team have reviewed this document and agree on its content and format. Any disagreements about this document are documented below.

### Development Team Signatures

---

Print Name

---

Date

---

Signature

Comments:

---

Print Name

---

Date

---

Signature

Comments:

---

Print Name

---

Date

---

Signature

Comments:

---

Print Name

---

Date

---

Signature

Comments:

---

Print Name

---

Date

---

Signature

Comments:

## Appendix C - Document Contributions

- **Group**
  - Use cases 1-2
  - Non-functional Requirements
- **Tyler Campbell**
  - Use case 3-7
- **Aparna Kaliappan**
  - Wrote use cases 10-13, 24-37
  - Created use case diagrams 10-13, 25-31
  - Appendices A and B
- **Sean Murren**
  - Section 5: Deliverables
  - Use cases 8-9, 14-15
- **Ying Zhang**
  - Introduction
  - Use Cases 16 - 19
  - Use case diagrams 1, 2, 16-24, 32-37
- **Siqi (Sandy) Lin**
  - Use Case 19 - 24
  - Non-functional Requirement 1-7