

Requirements Engineering
Assignment 1: Modeling Requirements

Jasinge Don Sahas Renuja Gunasekara

CUID: 20462075

Email: 20462075@student.curtin.edu.au

Date: 17/09/2021

Words: 378

Document Size: 35 Pages

Contents

Report.....	3
Requirement Elicitation Techniques.....	4
Scope.....	6
Functional Requirements.....	6
Non-Functional Requirements.....	8
Functionality.....	9
UML Use Case Diagrams	9
Use Case Descriptions	10
UML Sequence Diagrams.....	16
Dynamic Behavior.....	22
UML Activity Diagrams	22
State Diagrams	33
References	35

Report

The Problem Statement chosen for this Assessment is the “*Problem Statement 1*” that is defined as the Industry Collaboration System. Upon initial observation, as the system is to be used by more than 50,000+ members, it was advisable to use requirement elicitation methodologies that were general and was able to gather information from a large sample to prevent the skewing of information due to the sample bias error¹. Since there is already an existing system which records collaborations between the main branch and other collaborating companies, this can be dichotomized as “Extension of an existing system” as opposed to a “re-imagining of an entire system”. Along with a “Workshop” type requirement elicitation that constantly gathers feedback along the process of development which can be accessed by all users of this system will be able to gauge the system requirements and the users’ expectations accurately.

Upon observation of the hierarchies of the users we were able to observe intuitive workloads given to people with their respective title, except for the “Top Management”, who act as the “Branch Management” of the head office of both the National Engineers Association and collaborative companies. On extension, Top Management is allowed to accept and reject any collaborations of other sub-branches as well. All user roles defined are applicable to both the National Engineers Association and Collaborative Companies registered on the system. Therefore, in this document, when the user role, “Top Management”, or “Editor” is mentioned, this is applicable to both an employee of the National Engineers Association and Collaborative Companies. The only exception to this dual nature of user roles is for the “Admin” role who are exclusively employees of the National Engineers Association.

The core aim of the system is to initiate, accept, reject and track collaborations. The way in which the two collaborative entities meet is facilitated by a rudimentary chat feature, however further meeting methodologies is not facilitated by the system by providing features such as video-conferencing and such as these mechanisms exist outside of the system, and after initial contact these collaborative groups are able to exchange alternative contact information. Therefore, it was decided against to implement such mechanisms as it would unnecessarily increase the complexity and the scope of the system.

In-depth details of this system have been extensively plotted out as part of the Appendices, with relevant assumptions and claims stated in footnotes. Inspiration gathered from certain publications have been noted in the Reference List at the end of this document.

¹ Further Requirement Elicitation Technologies mentioned below.

Requirement Elicitation Techniques

There will be a multifaceted approach to requirement elicitation techniques as the usage of a few from certain areas might not provide the most factually accurate picture. The requirement elicitation techniques used in the gathering of the information will be as follows.

1. **Data Gather from existing system:** Since there is an existing system, we will gather as much information as possible about how such collaborations have occurred with the parent organization and industry companies.²
2. **Background Reading:** Read through existing documents about the procedures on how certain companies operate and familiarize us with the culture and procedure to certain activities that take place within a collaboration.³
3. **Introspection:** The team of developers, gather around and “imagine” what the system would eventually be and lists down some features, key-issues. These features might not reflect the end-users/stakeholders’ requirements but is necessary to gauge the requirements from this vast userbase. There is also some overlap with this technique, and another known as **Stake Holder’s Analysis**.⁴
4. **Survey:** Since the National Engineers Association consists of more than 50,000 members it is highly illogical to use mechanisms such as Interviews as we will not be able to have time to gather information from a sufficiently large sample size, and when that issue arises; we also have the doubt of a possible sample bias.⁵

By creating a Survey and gathering information on the old system we can gather information from a large sample size and compile this information easily. For example, this Survey can contain questions such as, but not limited to, (these questions are from the Introspection part of the process of requirements elicitation):

On a scale of 1-10 which one of these key issues are important to you?

² Assumption is made that this current system provides some or limited functionalities to the client.

³ Assumption is made that these documents will not be extremely lengthy and time consuming to read through with useful information being able to be digested immediately to a reader.

⁴ Assumption is made that the team of software developers will be able to arrive at similar and intuitive requirements for its client.

⁵ Assumption is made that there will not be a sample bias (e.g.: group of people working in different time zone when survey is being conducted etc.)

5. **Prototyping:** Prototypes are mockups of sort, where you can draw a mockup of a certain UI and UX to pitch to a group of people/client and receive feedback from. This provides them with a view on a potential “end look” of sorts without wasting time coding something that will not be approved of.⁶
6. **Workshop:** These are continuous workshops throughout the development process to get vital feedback from stakeholders regarding current progress, such as goals, priorities, mock-ups, UI and UX elements and functionalities.⁷

⁶ Assumption is made that to prevent sample bias these mockups will be pitched to sufficiently large and important stakeholders.

⁷ Assumption is made regarding the participation of individuals at workshops to gain sufficiently valuable information.

Scope

Functional Requirements

User Story	Requirement
As a user, I must be able to browse through all records of past collaborations so that I can view all the past collaborations. ⁸	The system must keep record of all past collaborations and provide them to all requesting users. ⁹
As a user, I must be able to track all ongoing collaborations in real-time so that I can see which branches are engaged in what collaboration.	The system must allow users to view the individual stages in real-time.
As branch management ¹⁰ , I should be suggested possible partners for collaborations so I can engage with these partners if needed. ¹¹	The system must analyze the branch and suggest potential collaborations to the branch management.
As an admin, I should be able to set and change user roles so I can change the roles for users when they get promoted/demoted/change jobs. ¹²	The system must allow admins to change user roles.
As an editor, I should be able to maintain the company branch information shown on the system so I can change the outlook of my branch on this system.	The system must allow the editors to manipulate the company branch information shown.
As external relations, I should have the ability to request for collaborations so I can initiate the conversation regarding collaborations.	The system must allow external relations to request collaborations from other entities.
As a project lead, I should have the ability to discuss with the collaborators so that I can discuss the terms of the collaborations.	The system must allow the project leads to discuss with the collaborators.
As branch management, I should have the ability to accept or reject collaborations so	The system must allow the branch management to have the ability to accept or reject collaborations.

⁸ User is any person who views this system, they do not have to be logged in or registered to view all these details since it falls in line with the transparency requirement of the client.

⁹ Assumed that everyone who is a user, regardless of their role should be able to view information about all collaborations as it is one of the projects goals of visibility.

¹⁰ Entities such as Branch Management, External Relations, Project Lead, and Top Management, Editor is applicable to both collaborating companies and the National Engineers Association. Admins are exclusively employed by the National Engineers Association.

¹¹ Assumed that the person who oversees dictating collaborations is the branch management, therefore suggestions are offered to users with this role.

¹² Assumed that the admin is a tech role, who oversees setting all user roles and maintenance of the system.

that I can initiate or decline a collaboration. ¹³	
As Organization Top Leadership, I should have the ability to cancel collaborations at any time so that I can cancel collaborations that don't fall in line with the organization policy. ¹⁴	The system must allow the organization top leadership to have the ability to cancel collaborations at any time.

¹³ Assumed a certain bureaucratic hierarchy. The branch management sees suggested collaborators, they inform the external relations to initiate contact, the project leads negotiate, and finally the branch management accepts or rejects the collaboration.

¹⁴ Assumed that the Organization Top Leadership should have the ability to veto, any decisions taken by the branch management at any given time. Also, that Organization Top Leadership are also structured to be the "Branch Managers" of the parent organization since they too have been engaging in collaborations.

Non-Functional Requirements

Theme	Non-Functional Requirement
Security	The users will be prompted to a login page before the information of the system will be accessible to them.
Reliability	The system will have a Maximum Tolerable Downtime of 1 hour per month.
Usability	A user should be able to view information regarding a branch/collaborative company within 5 clicks upon logging into the system.
Performance	Each page of the system must load within 5 seconds.
Functionality	The system shall display the past collaborative history of a branch/company in the format of a table with the columns of date, collaborative partners, collaboration status.
Performance/Accuracy	The system will have updated ongoing collaboration statuses within an hour and will display it.
Security	The system will use SHA-1 encryption algorithm to store user login and password credentials on a database.
Performance/Expandability	The system should be able to have up to 1 million active users without crashing.

Functionality

UML Use Case Diagrams

Some notable assumptions when drawing this diagram.

1) User

a. Any logged in user can view all past and ongoing collaborations, but not pending collaborations. Once a pending collaboration has been accepted by either Branch Management or Top Management, they will be able to view this.

2) Top Management

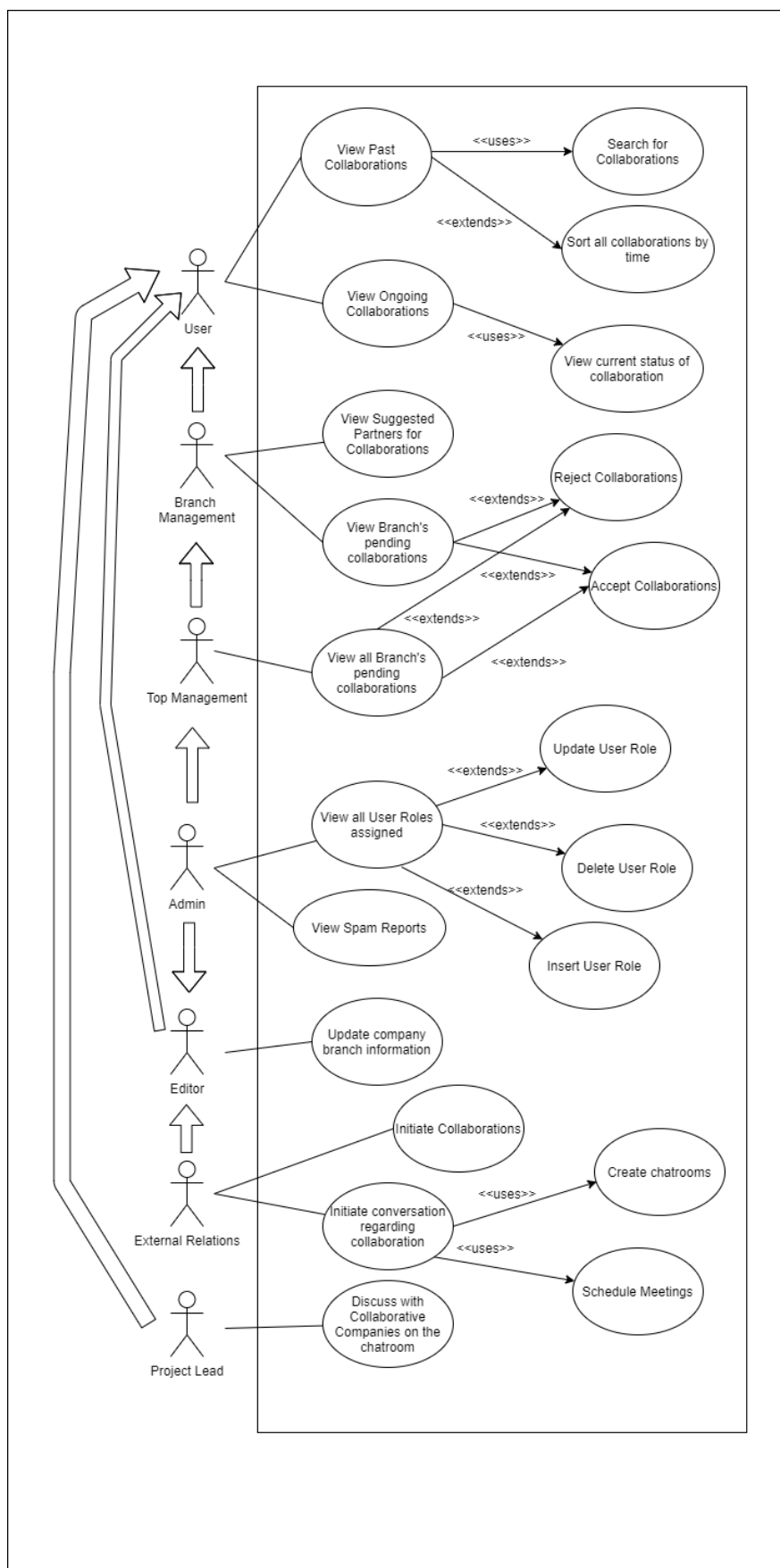
a. Top Management is assumed to be as the “Branch Management” of the Head Office, with additional powers. They can reject or accept collaborations on behalf of all branches.

3) Admin

a. The admin is given the powers to accept or reject in cases where there are technical difficulties in doing so for the management to do so. They are also given the Editor level powers in case a disgruntled ex-Editor decides to deface a branch page. He/She is not given External Relations and Project Lead powers as an Admin should not be interacting with third party entities.

4) External Relations

a. We assume that External Relations is the first point of contact for collaborative companies, they can schedule meetings and we assume we have a rudimentary chatroom capability so they can speak to one another, to keep records. Project Leads will handle the actual conversation with these companies.



Use Case Descriptions

Use Case: 001

Title: View Past Collaborations

Description: This use case describes the process where a user can browse through all the past collaborations between the National Engineers Association and its collaborators.

Primary Actor: User (Branch Management, Top Management, Admin, Editor, External Relations, Project Lead)

Preconditions: None.

Trigger: The user chooses the "Past Collaboration" option.

Postconditions: None.

STEPS

1. The User is presented all the Past Collaborations in Alphanumerical order
2. IF the User chooses on order by Chronological Order
THEN the system presents all the Past Collaborations in Chronological Order.
3. The User searches for a Collaborating Company.
4. The System retrieves information about the Users input.

ALTERNATIVES

3. The User searches for a Collaborating Company that does not match with any of the records.

3a. The System displays a "Company not found" page.

3b. RESUMES at STEP 1

Use Case: 002

Title: View Ongoing Collaborations

Description: This use case describes the process where a user can browse through all the ongoing collaborations between the National Engineers Association and its collaborators.

Primary Actor: User (Branch Management, Top Management, Admin, Editor, External Relations, Project Lead)

Preconditions: None.

Trigger: The user chooses the "Ongoing Collaboration" option.

Postconditions: None.

STEPS

1. The User is presented all the Ongoing Collaborations in Alphanumerical order
2. IF The User chooses to view more information about a particular collaboration
THEN the System displays the current status of the Collaboration.

Use Case: 003**Title: Initiate Collaborations**

Description: This use case describes the process where the External Relations first sends a request with a company to initiate a collaboration. This would be done when the Branch Management has requested the External Relations to do so (This is done outside of the scope of this system)

Primary Actor: External Relations

Precondition: The External Relations user is logged into the account successfully.

Trigger: External Relations clicks on the Initiate Collaboration option.

Postconditions: None.

STEPS

1. The External Relations chooses the Initiate Collaboration option.
2. He/She selects the relevant company/branch to send an invite to collaborate.

ALTERNATIVES

2. The branch/company is not open for any collaborations now.

2a. The system outputs a “The branch/company isn’t open for any collaborations at the moment” message.

2b. RESUME at STEP 2.

Use Case: 004**Title: Initiate Conversation Regarding Collaborations**

Description: This use case describes the process where the External Relations will create chatrooms for the Project Leads to discuss and to schedule the meetings. (That can happen on and off the system)

Primary Actor: External Relations

Precondition: The External Relations user is logged into the account successfully. There is at least one ongoing collaboration among the branches.

Trigger: External Relations clicks on the “Initiate Conversation” option.

Postconditions: None.

STEPS

1. The External Relations chooses to create a chatroom.
2. He/She chooses who they would like to have a chatroom with out of a list of all the current ongoing collaborations with their branch.
3. He/She chooses a time to schedule the meeting for the chatroom to take place.

ALTERNATIVES

1. The External Relations has created more than 5 chatrooms in the past hour.

1a. The system informs the External Relations that he/she is creating too many chatrooms.

2a. Reports to the Admins of suspicious behavior.

3a. RESUME at STEP 1 after 24 hours.

Use Case: 005

Title: Discuss with Collaborative Companies on the chatroom

Description: This use case describes the process where the Project Lead of either the National Engineers Association or the Collaborative Company contacts the other Project Lead using the chatroom functionality.

Primary Actor: Project Lead.

Precondition: The Project Lead is logged into the account successfully. There is a chatroom already created by the External Relations to discuss on.

Trigger: Project Lead chooses a chatroom that has already been created.

Postconditions: None.

STEPS

1. The Project Lead chooses a created chatroom.
2. He/she receives messages from the other party.
3. He/she sends messages from the other party.

ALTERNATIVES

1. The Project Lead has opened the chatroom outside of a scheduled time
 - 1a. System will display a message "The chatroom has not opened now"
 - 1b. Use Case will RESUME at scheduled time.
2. There are no messages to receive
 - 2a. System displays a message "No messages yet, send message?"
 - 2b. RESUMES at STEP 3.

Use Case: 006

Title: Update Company Branch Information

Description: This use case describes where the Editor will be updating the branch information.

Primary Actor: Editor, External Relations¹⁵, Admin¹⁶

Precondition: The Editor is logged into the account successfully. The Editor is editing the branch information of where the Editor is employed.

Trigger: The Editor chooses the "Edit Page" option on a branch page of either a collaborative company or National Engineers Association.

STEPS

1. The Editor is presented with fillable spaces for the branch's information such as Contact Number, About Us and Address.
2. The Editor chooses the "Save" option after he/she is done manipulating

ALTERNATIVES

2. The Editor chooses the "Save" option by leaving some fillable spaces empty
 - 2a. System highlights areas that are not filled in.
 - 2b. System requests the user to fill in the highlighted areas.
 - 2c. Use case ends.

¹⁵ Assumed that External Relations should also have control of what the Editor can do as an editor would be ideally inside the External Relations department.

¹⁶ Assumption mentioned above, has to do with possible defacing of the branch information.

Use Case: 007

Title: View all User Roles assigned

Description: This use case describes where the admin can view and manage the users and their permission levels (roles) within the system.

Primary Actors: Admin

Precondition: The Administrator has logged into the account successfully.

Trigger: The admins choose the “View User Roles” options.

STEPS

1. The admin is presented with a list of all the Users and User Roles in the system.

EXTENSIONS

- 1a. The admin chooses the “Delete User Role” option

- 1aa. The System removes the User credentials from the system

- 1ab. The System informs the admin that the User Credentials were successfully removed

- 1ac. RESUME at STEP 1

- 1b. The admin chooses the “Update User Role” option

- 1ba. The System asks the admin to choose which user role the user is.

- 1bb. The System updates the user role.

- 1bc. The System informs the admin that the User Credentials were successfully updated.

- 1c. The admin chooses the “Insert User Role” option

- 1ca. The System asks the admin to choose which user role this new user is.

- 1cb. The System generates a username and password randomly

- 1cc. The System requests the admin to copy down the credentials and pass it onwards to new user.

Use Case: 008

Title: View Suggested Partners for Collaborations.

Description: This use case describes the Branch Management viewing all Suggested Partners for Collaborations.

Primary Actors: Branch Management, Top Management

Precondition: The Branch Management is successfully logged into the account.

Trigger: They have chosen the “Suggest Collaborators” option.

STEPS:

1. The Primary Actors are shown a list of possible collaborators.

EXTENSIONS:

1. There are no such possible collaborators.

- 1a. System will display a “No Possible Collaborators” message.

- 1b. Use Case ends.

Use Case: 009**Title:** View Branch's pending collaborations**Description:** This use case describes the Branch Management viewing all the Pending Partners for the Collaborations.**Primary Actors:** Branch Management, Top Management.**Precondition:** The Branch Management is successfully logged into the account.**Trigger:** They have chosen the "View Pending Collaborations" option.**STEPS:**

1. The Primary Actors are shown all the Pending Collaborations options for their branch.

Extensions:**1a.** The Primary Actor chooses the "Accept Collaboration" option

1aa. System changes the state of the collaboration to Ongoing Collaboration.

1ab. System makes the Collaboration public.

1ac. System removes the record from Pending Collaborations.

1ad. System informs Primary Actor that the Collaboration has been accepted.

1ae. Use case resumes at STEP 1.

1b. The Primary Actor chooses the "Cancel Collaboration" option

1ba. System removes the record from the Pending Collaborations.

1bb. System informs Primary Actor that the Collaboration has been rejected.

1ae. Use case resumes at STEP 1.

Use Case: 010**Title:** View all Branch's pending collaborations**Description:** This use case describes the Top Management viewing all the Pending Partners for the Collaborations.**Primary Actors:** Top Management.**Precondition:** The Top Management is successfully logged into the account.**Trigger:** They have chosen the "View Pending Collaborations" option.**STEPS:**

1. The Primary Actors are shown all the Pending Collaborations options for all branches.

Extensions:**1a.** The Primary Actor chooses the "Accept Collaboration" option

1aa. System changes the state of the collaboration to Ongoing Collaboration.

1ab. System makes the Collaboration public.

1ac. System removes the record from Pending Collaborations.

1ad. System informs Primary Actor that the Collaboration has been accepted.

1ae. Use case resumes at STEP 1.

1b. The Primary Actor chooses the "Cancel Collaboration" option

1ba. System removes the record from the Pending Collaborations.

1bb. System informs Primary Actor that the Collaboration has been rejected.

1ae. Use case resumes at STEP 1.

Use Case: 011

Title: View all Spam Reports

Description: This use case describes the admin viewing the spam reports.

Primary Actors: Admin.

Precondition: The Admin is successfully logged into the account.

Trigger: The Admin have chosen the “View Spam” option.

STEPS:

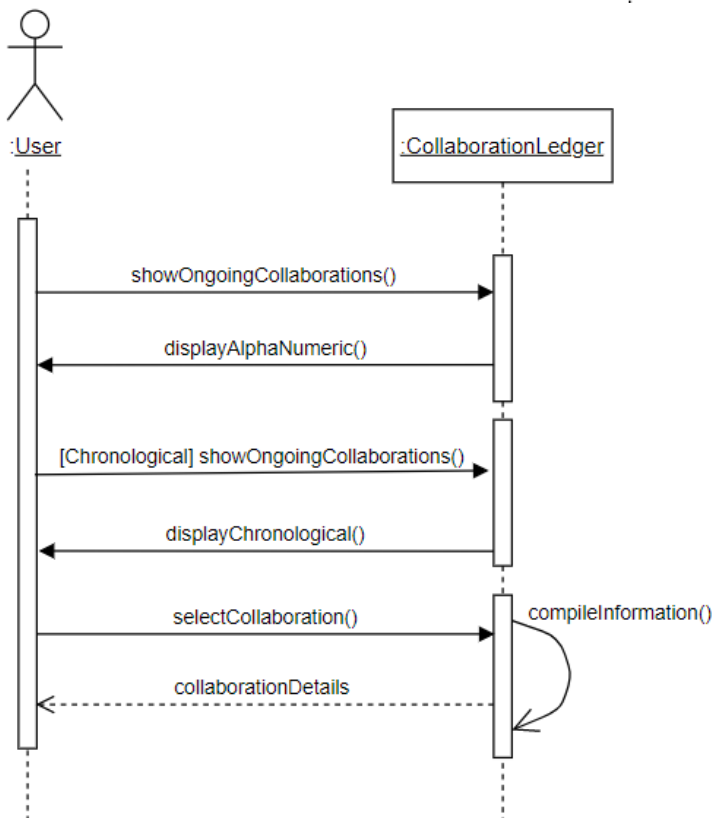
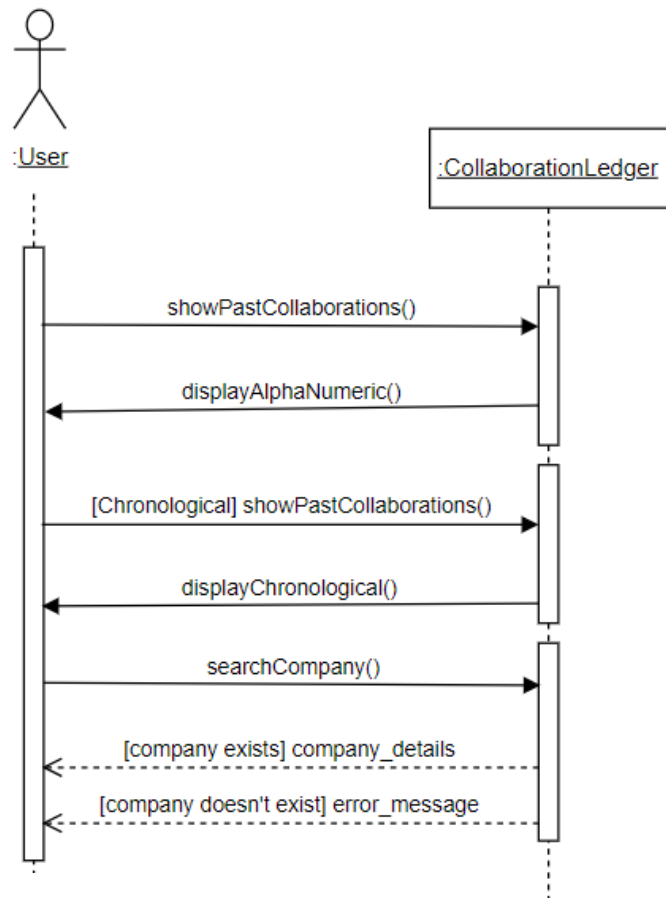
1. The admin is shown all the Spam Incidents in chronological order.

Extensions:

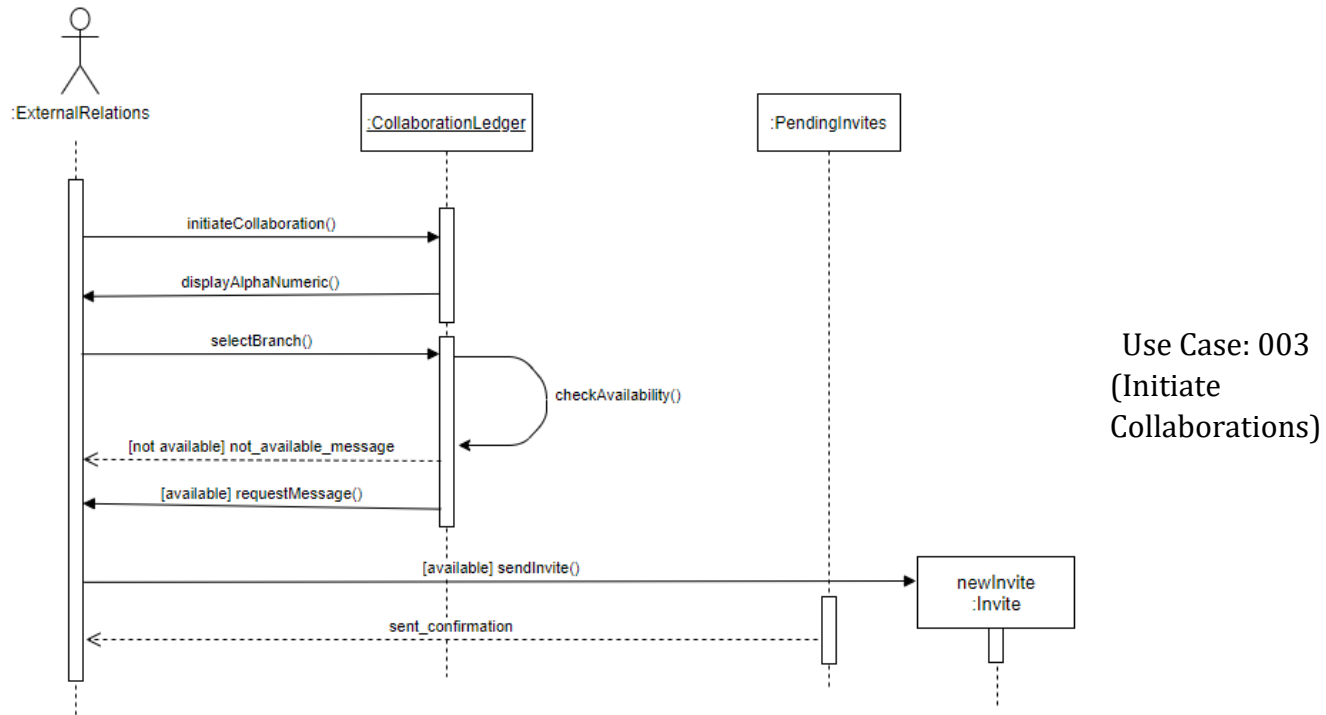
1. There are no Spam Incidents.
 - 1a. System informs the admin that there are no spam messages.
 - 1b. Use case ends.

UML Sequence Diagrams

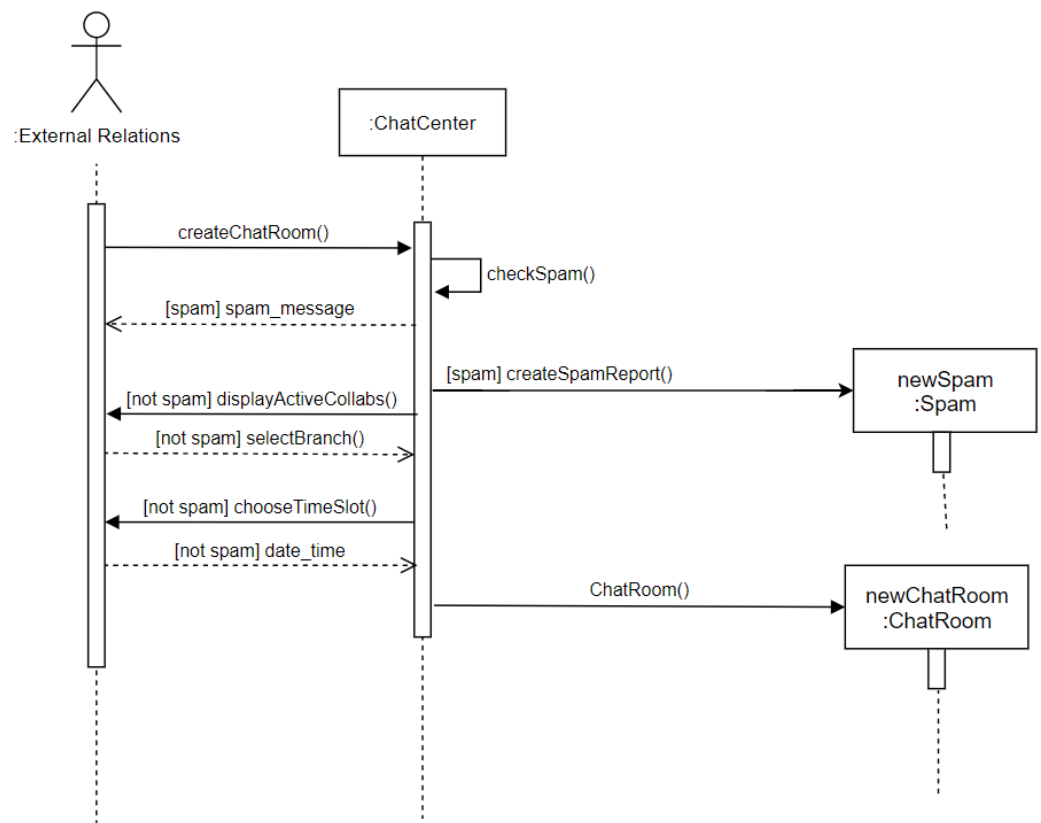
Use Case: 001 (Initiate Collaborations)

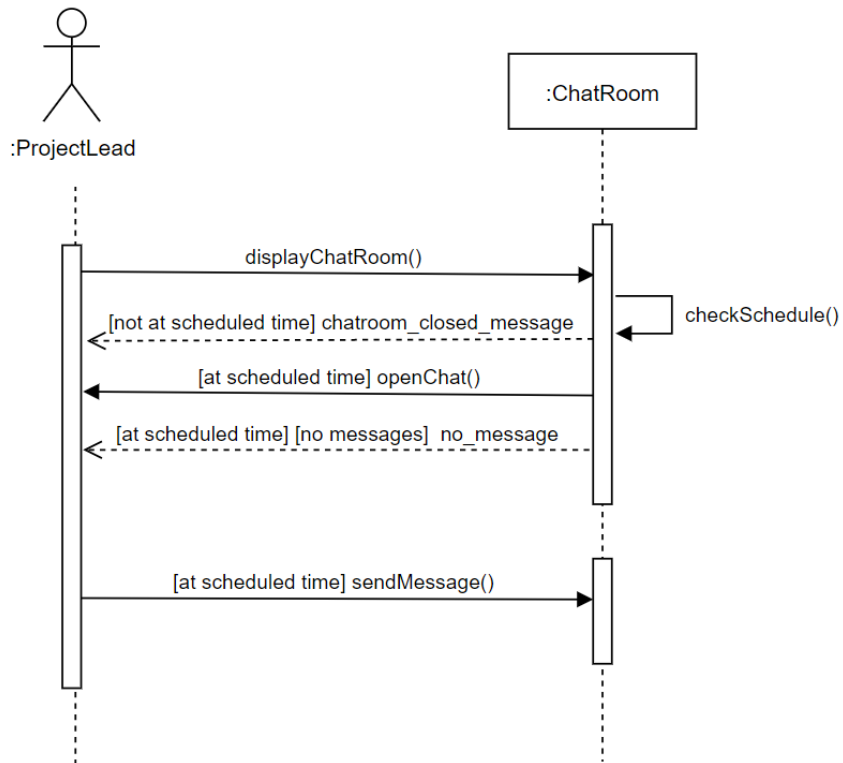


Use Case: 002 (Ongoing Collaborations)

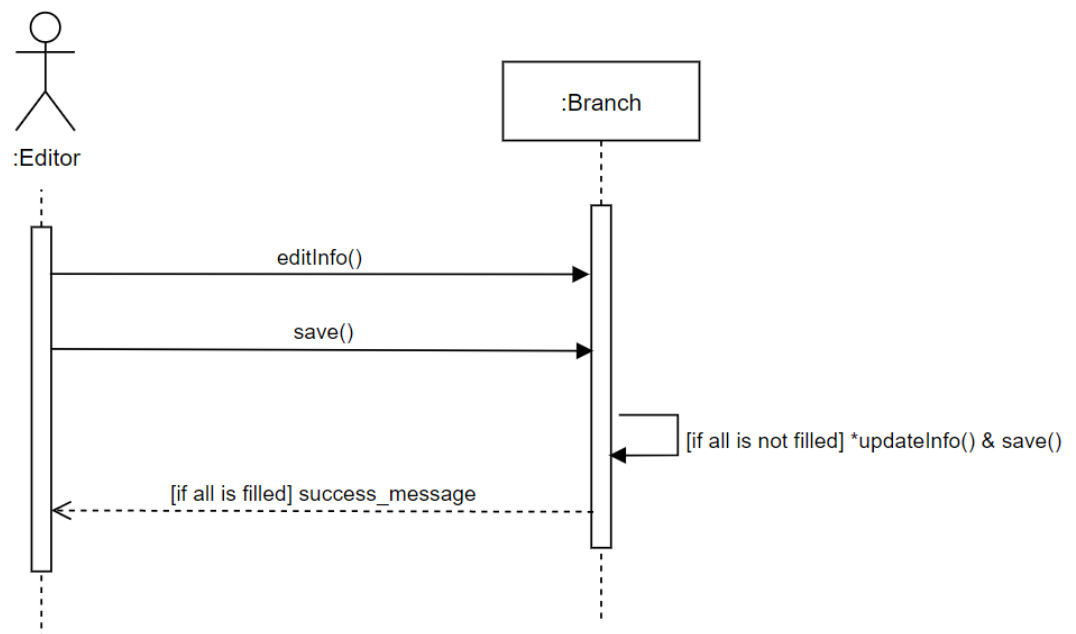


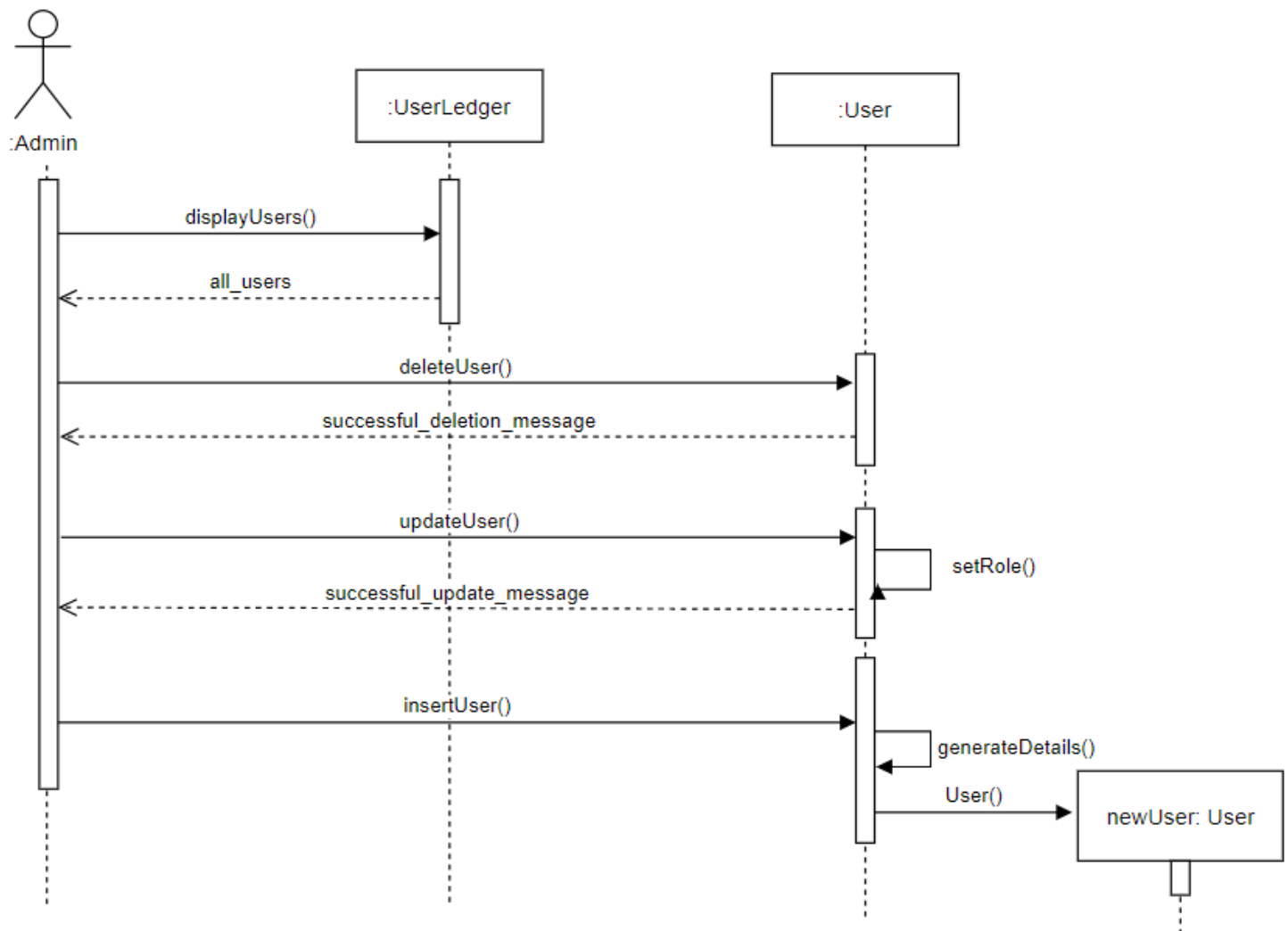
Use Case: 004 (Initiate
Conversation Regarding
Collaborations)



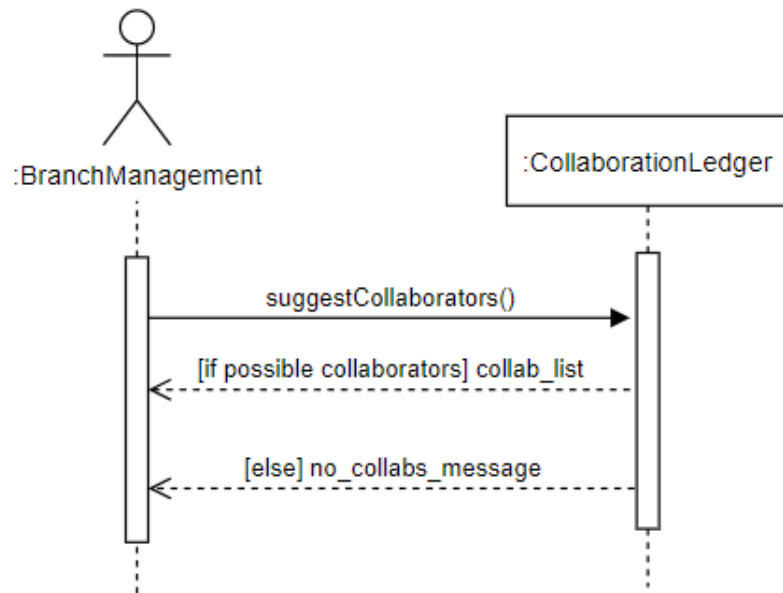


Use Case: 006 (Update Company Branch Information)

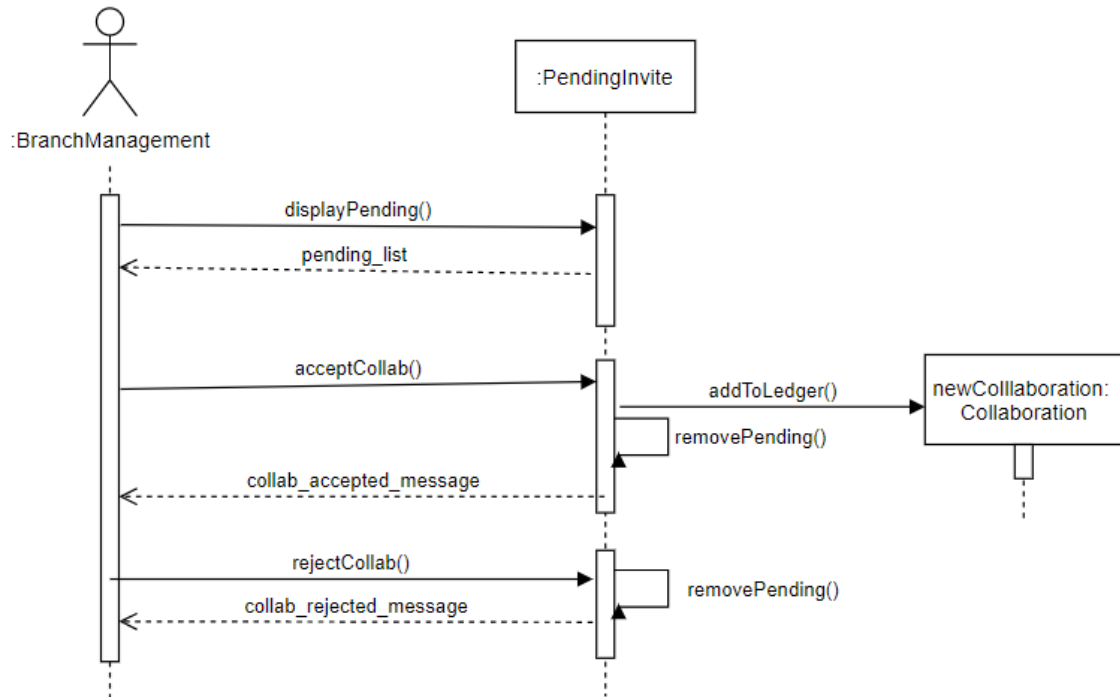




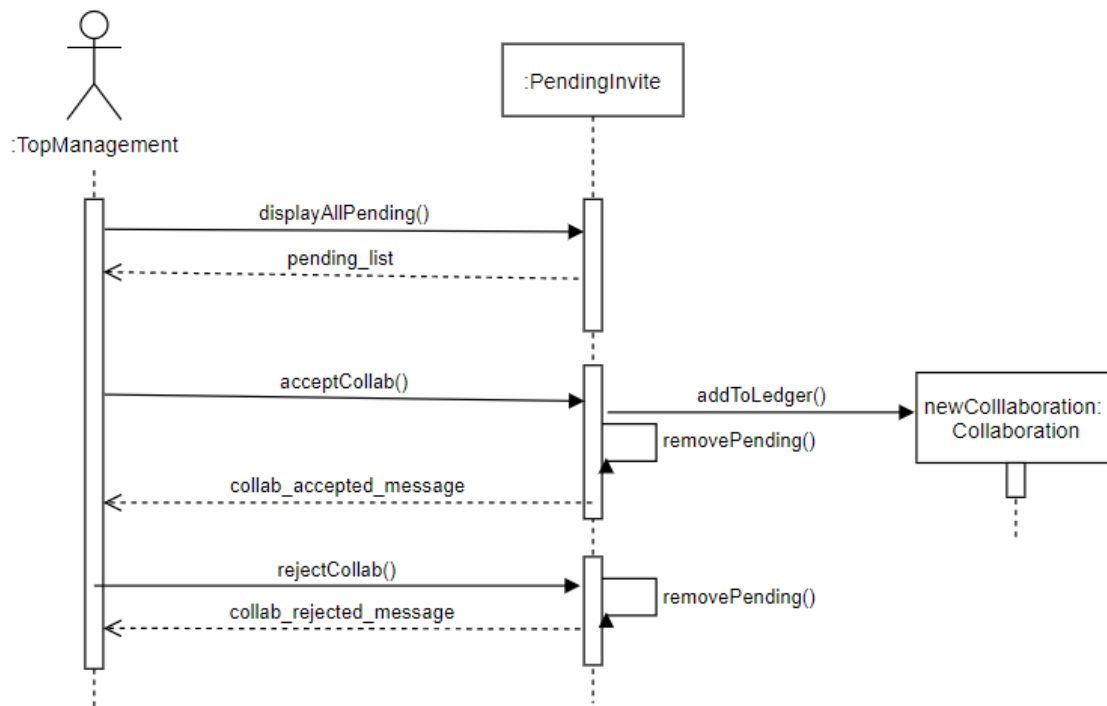
Use Case: 007 (View all User Roles Assigned)



Use Case: 008 (View Suggested Partners for Collaborations)

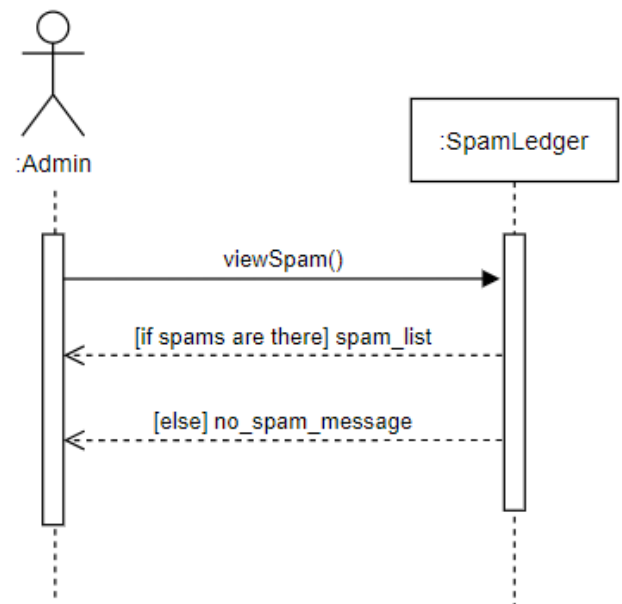


Use Case: 009 (View Branch's Pending Collaborations)



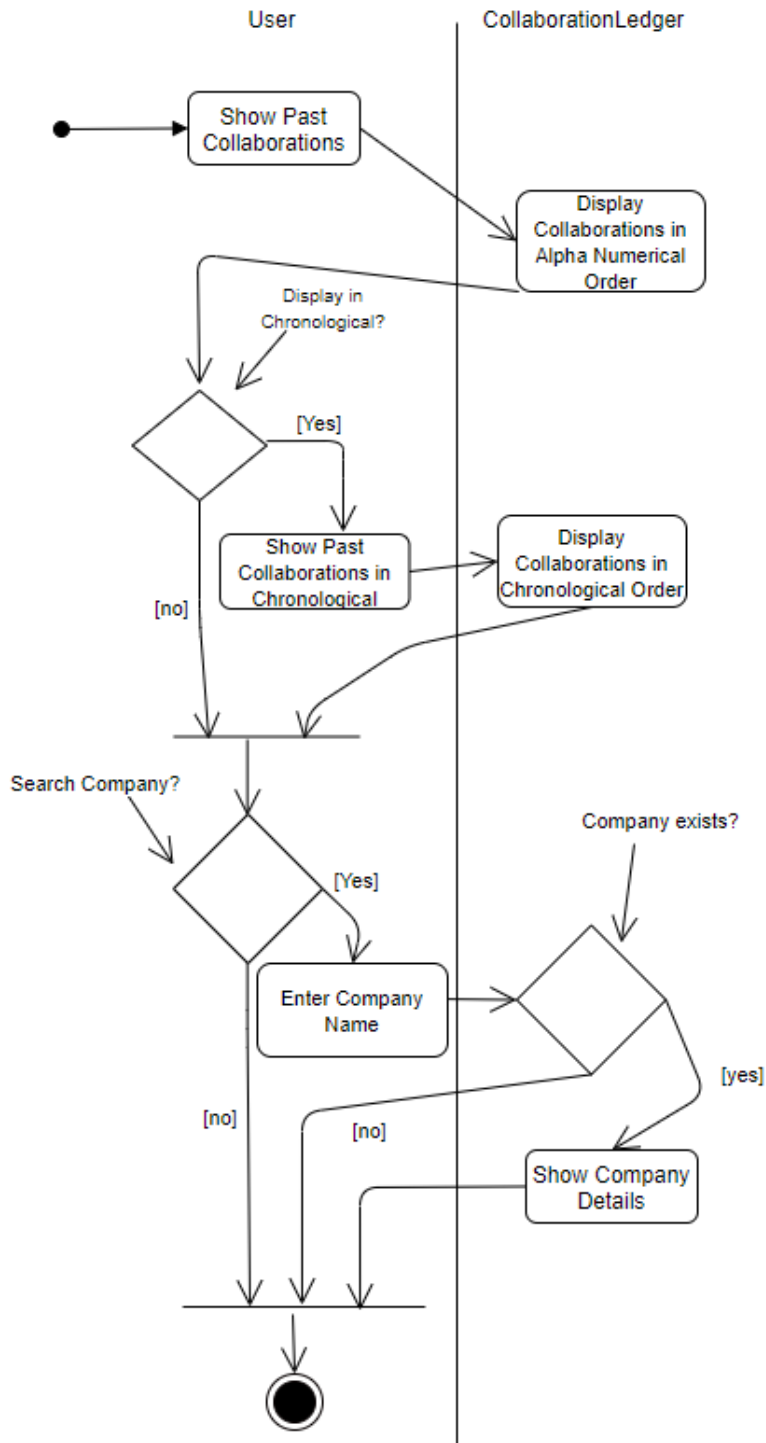
Use
Case: 010
(View All
Branch's
Pending
Collaboration)

Use Case: 011 (View All Spam Reports)



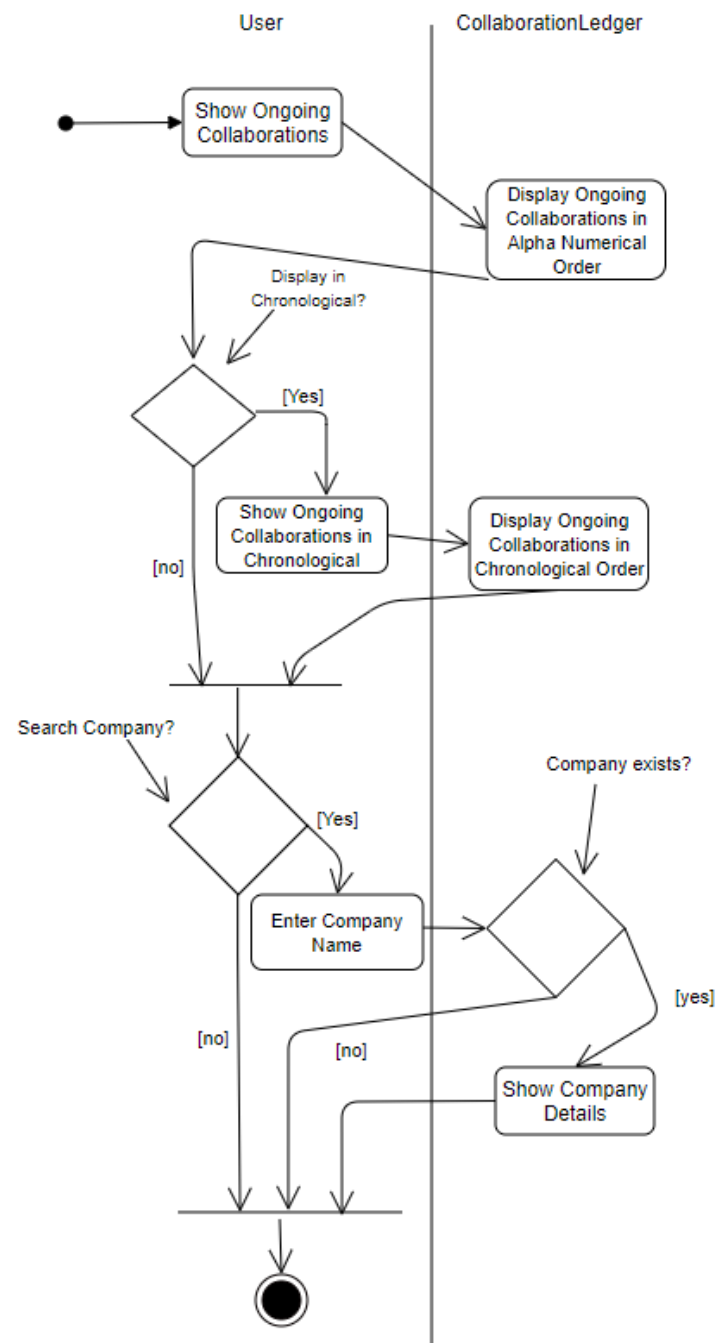
Dynamic Behavior

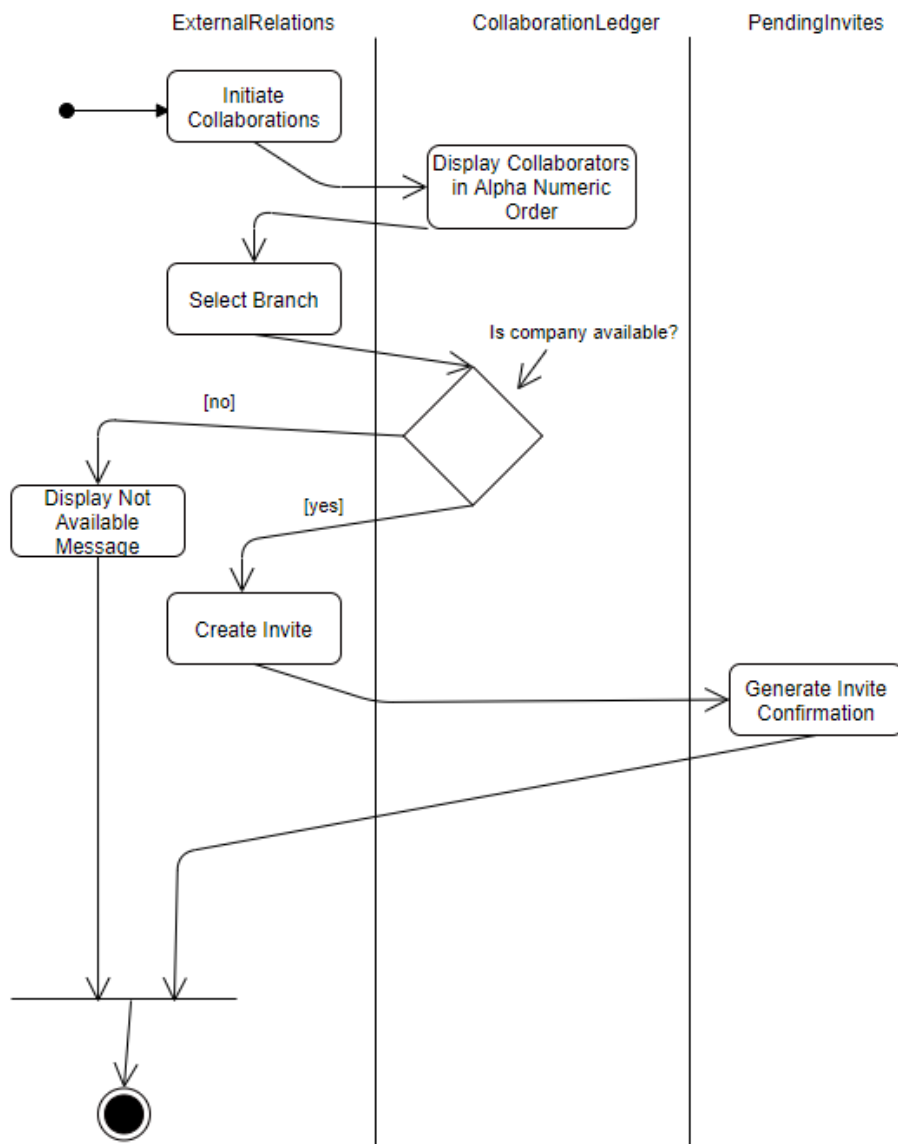
UML Activity Diagrams



Use Case 001:
Initiate Collaborations

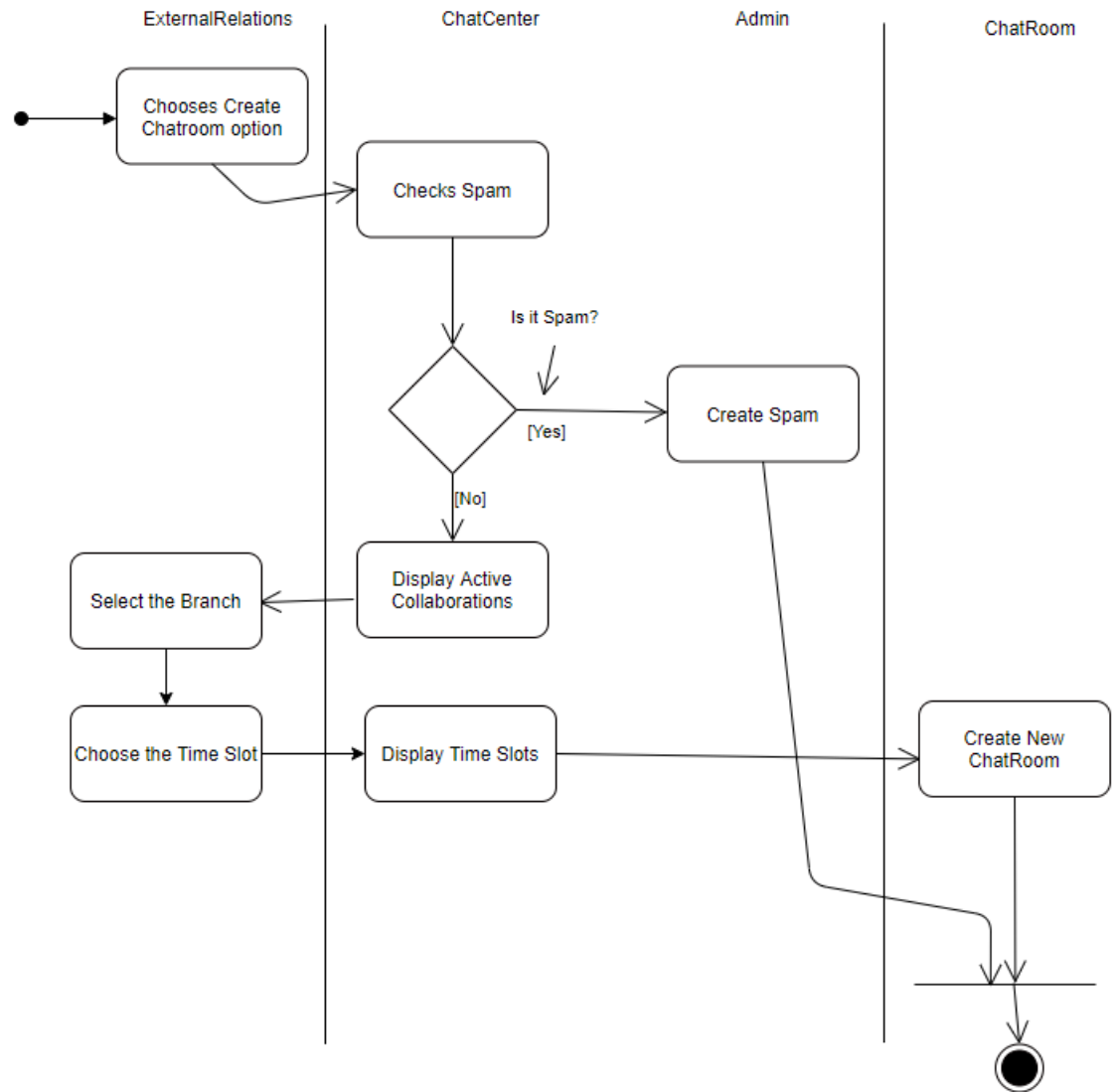
Use Case 002:
Ongoing Collaborations

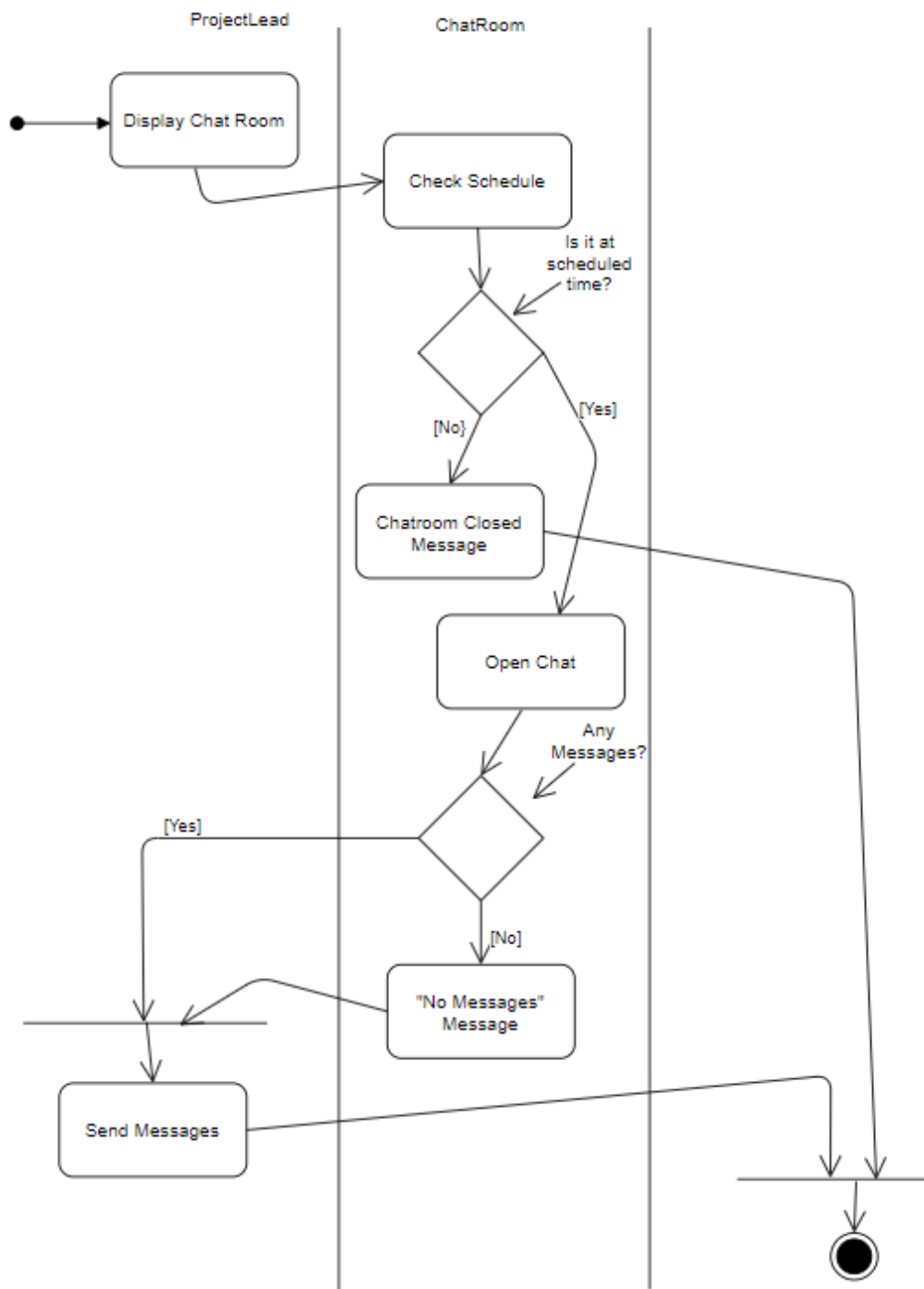




Use Case 003:
Initiate Collaborations

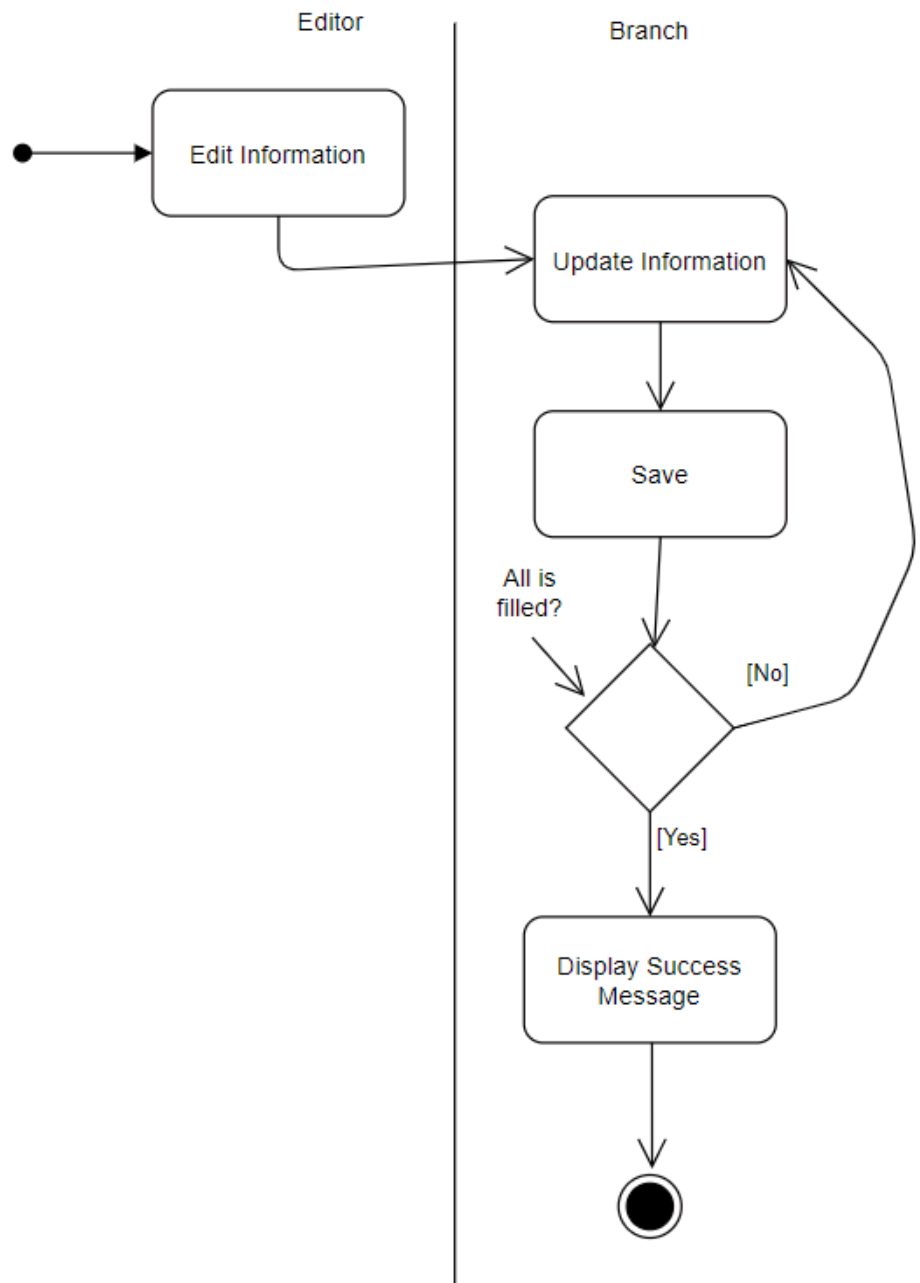
Use Case 004:
Initiate
Conversation
Regarding
Collaborations

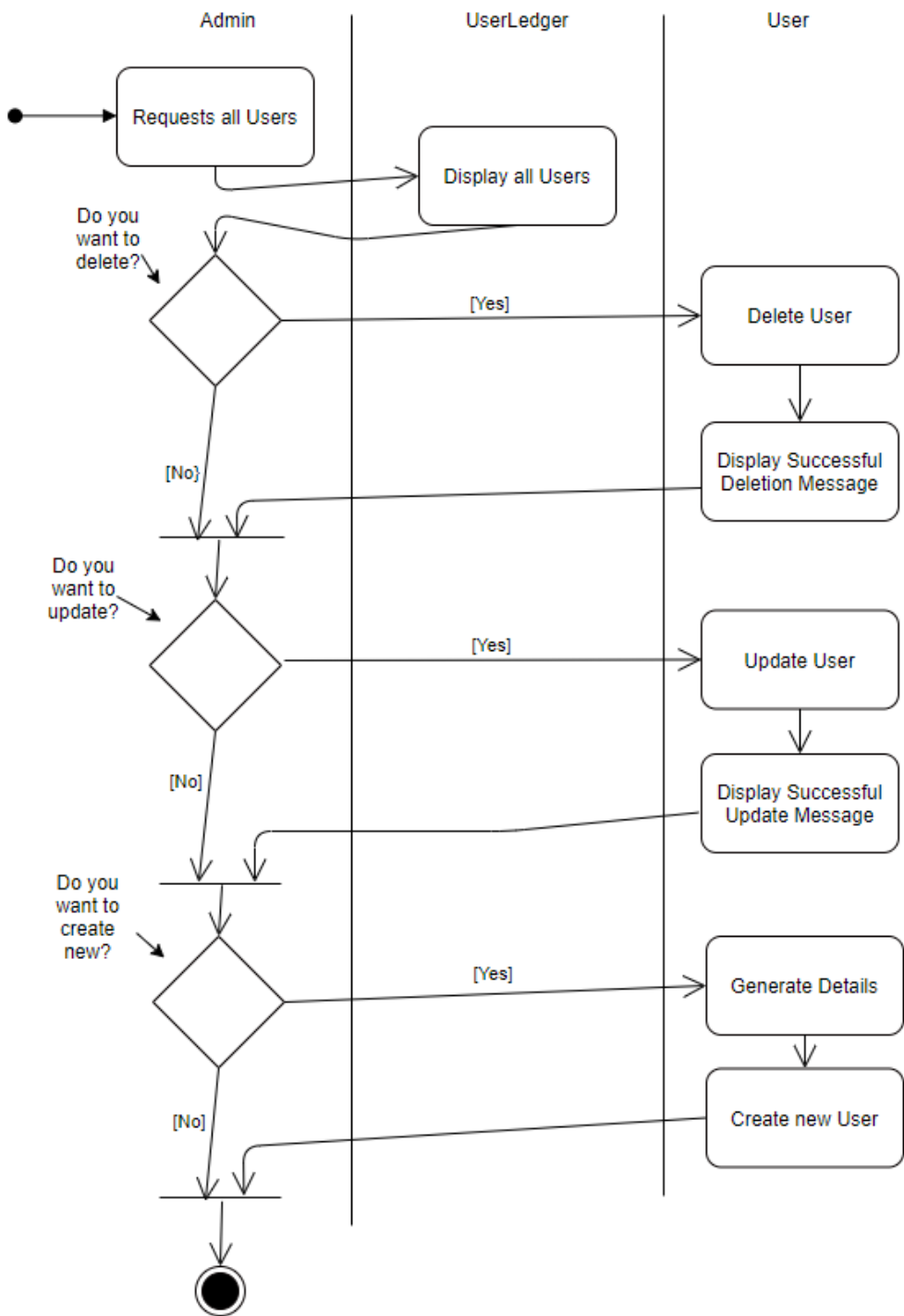




Use Case 005: Discuss with Collaborative Companies

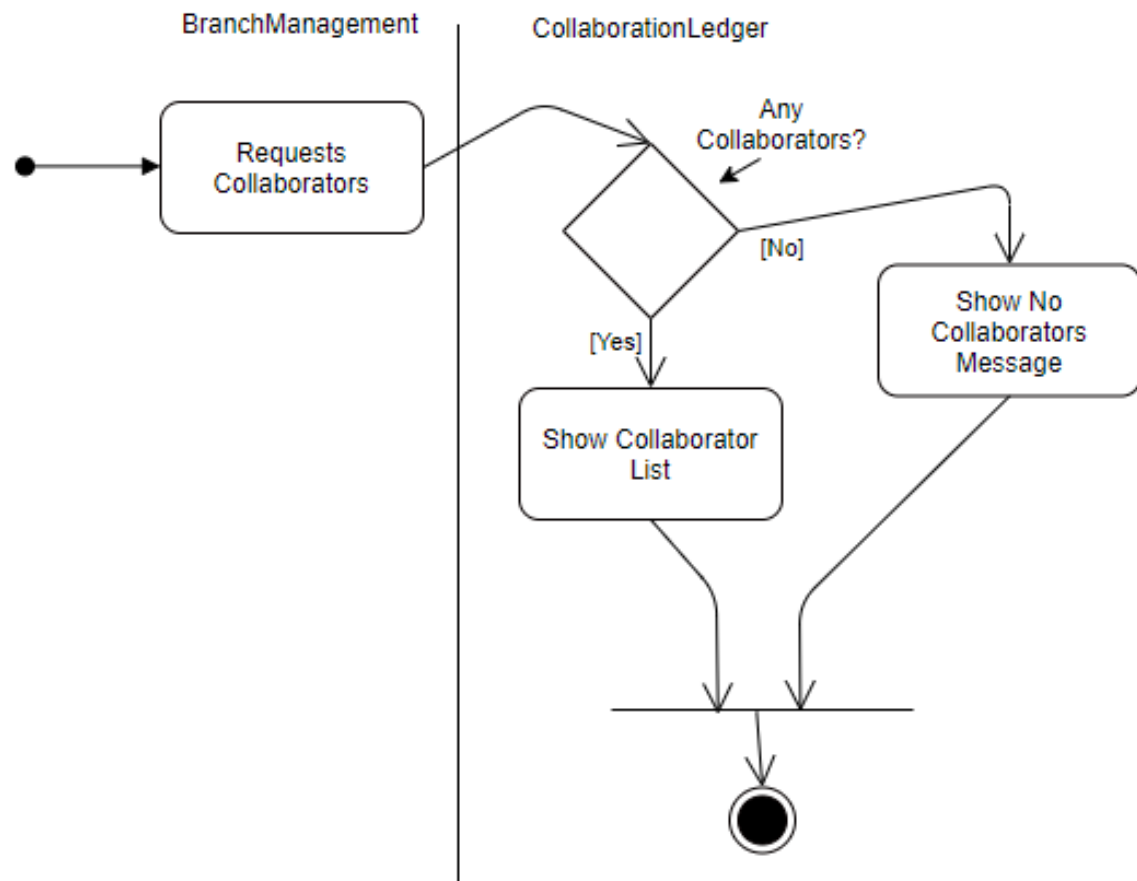
Use Case 006: Update
Company Branch Information

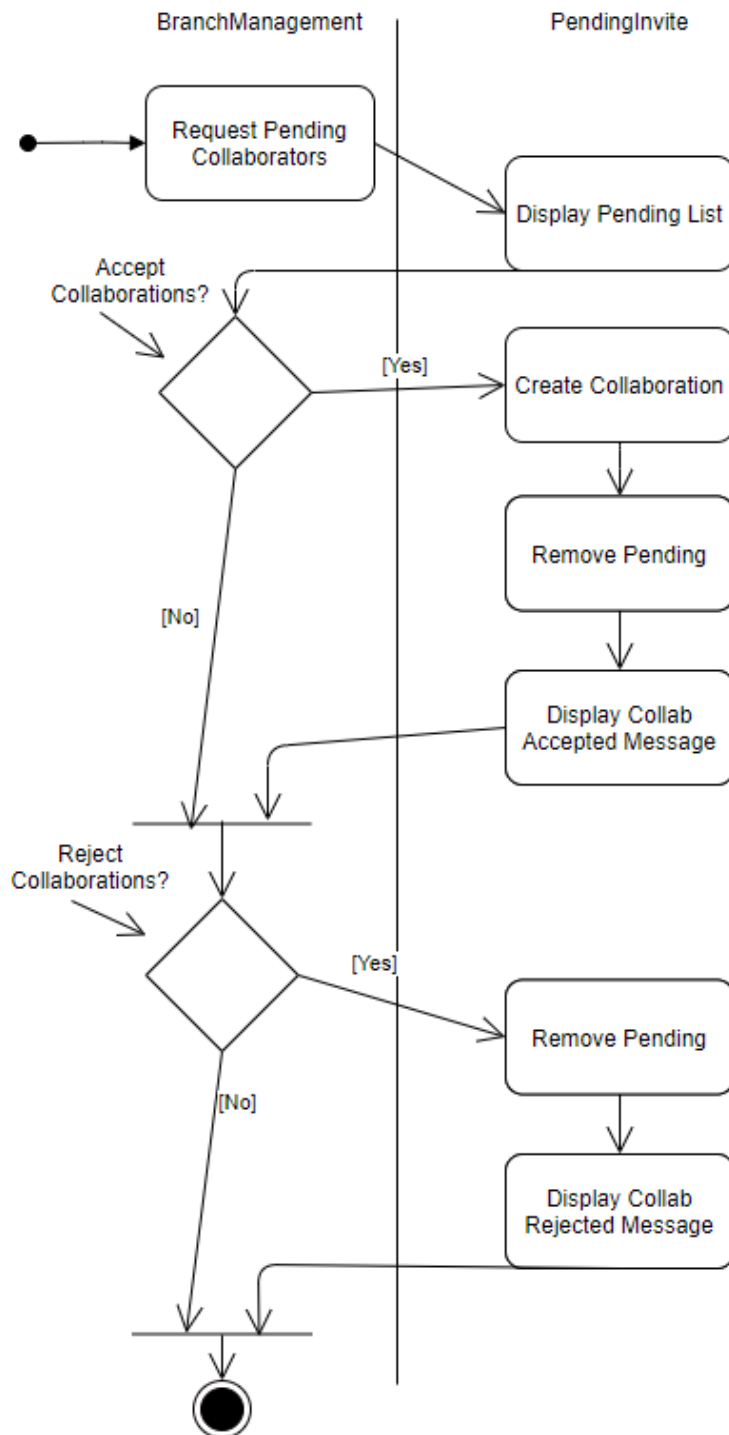




Use Case 007: View all
User Roles Assigned

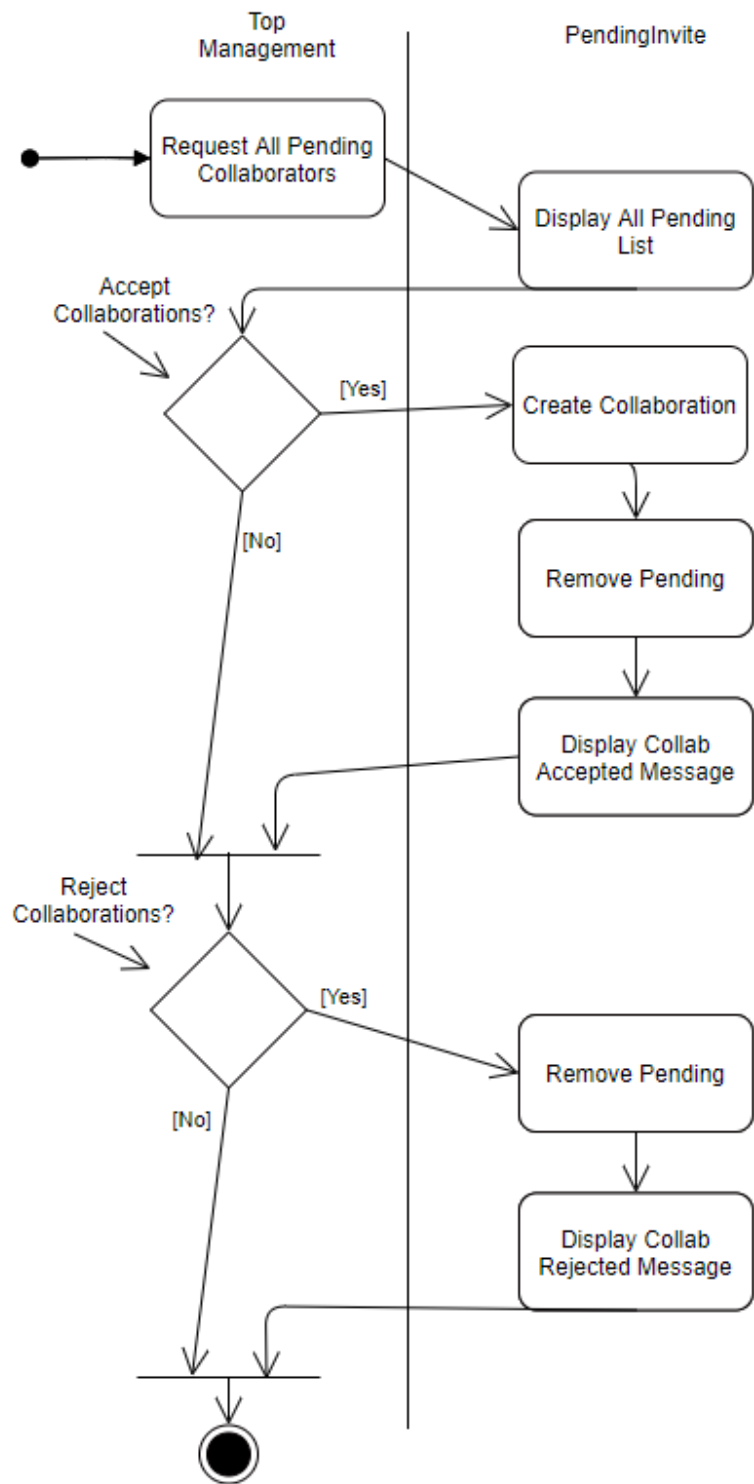
Use Case 008:
View Suggested
Partners for
Collaborations

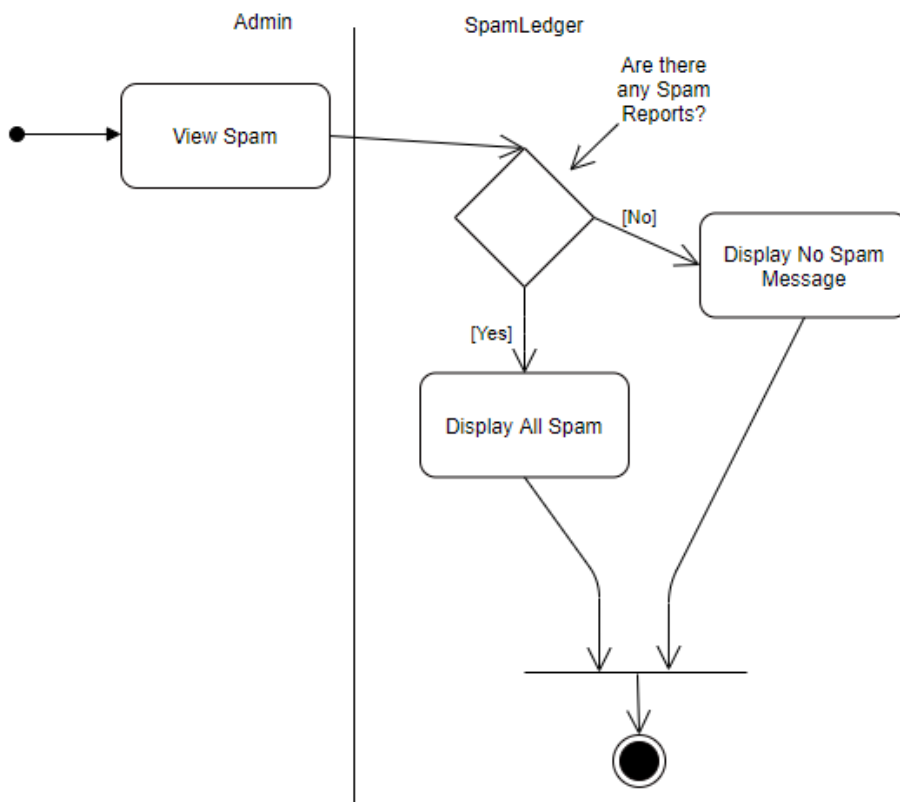




Use Case 009:
View Branch's Pending
Collaborations

Use Case 010:
View All Pending
Collaborations





Use Case 011:
View All Spam Reports

State Diagrams

The identifiable classes for this system would be

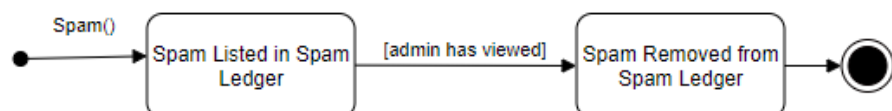
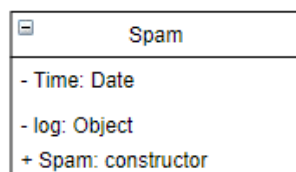
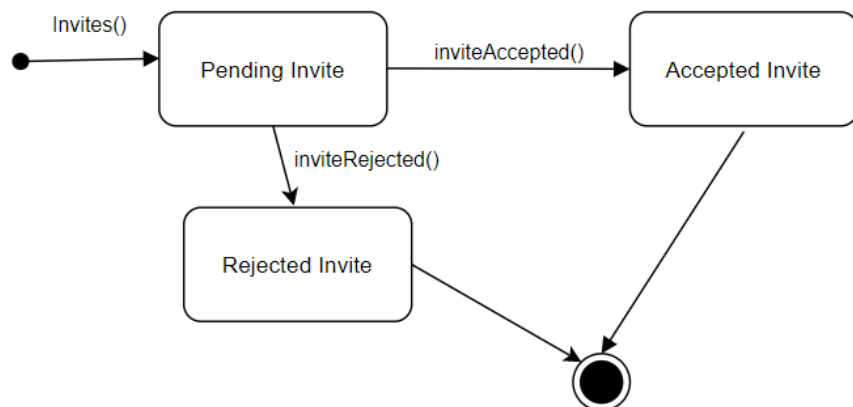
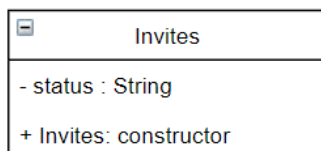
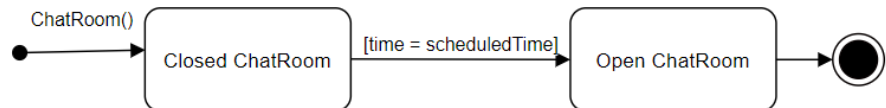
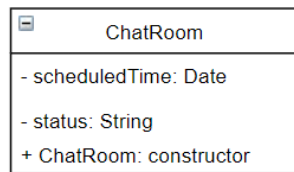
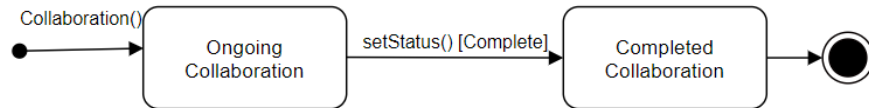
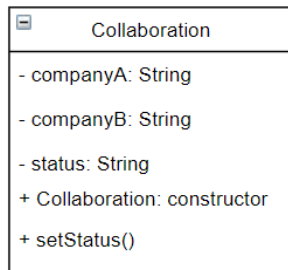
1. User
2. Admin
3. ExternalRelations
4. ProjectLead
5. Editor
6. BranchManagement
7. TopManagement
8. CollaborationLedger
9. Collaboration
10. ChatCenter
11. ChatRoom
12. PendingInvites
13. Invites
14. UserLedger
15. SpamLedger
16. Spam
17. Branch

There is no “State Change” for the classes “Admin”, “ExternalRelations”, “ProjectLead”, “Editor”, “BranchManagement”, “TopManagement” as they are all children classes of the class “User” but an Admin can dynamically cast them to different classes with the setRole() functionality, since it is not an attribute change, we will not be drawing UML State diagrams for them.

There is no “State Change” for the classes “CollaborationLedger”, “ChatCenter”, “PendingInvites”, “UserLedger”, “SpamLedger” as they are merely container classes for “Collaboration”, “ChatRoom”, “Invites”, “User”, “Spam” respectively. Upon creation of the classes, they are automatically added to the Container Classes.

Container Classes	Objects
CollaborationLedger	Collaboration
ChatCenter	ChatRoom
PendingInvites	Invites
UserLedger	Users
SpamLedger	Spam

We will only be drawing the UML State Diagrams for the Classes, “Collaboration”, “ChatRoom”, “Invites”, “Spam” as only these classes show a visible state change.



References

- Dumas, M., & ter Hofstede, A. H. M. (2001). UML Activity Diagrams as a Workflow Specification Language. *«UML» 2001 — The Unified Modeling Language. Modeling Languages, Concepts, and Tools*, 76–90. https://doi.org/10.1007/3-540-45441-1_7
- Glinz, M. (2007). On Non-Functional Requirements. *15th IEEE International Requirements Engineering Conference (RE 2007)*. Published. <https://doi.org/10.1109/re.2007.45>
- Tiwari, S., Rathore, S. S., & Gupta, A. (2012). Selecting requirement elicitation techniques for software projects. *2012 CSI Sixth International Conference on Software Engineering (CONSEG)*. Published. <https://doi.org/10.1109/conseg.2012.6349486>
- Wautelet, Y., Heng, S., Kolp, M., & Mirbel, I. (2014). Unifying and Extending User Story Models. *Advanced Information Systems Engineering*, 211–225. https://doi.org/10.1007/978-3-319-07881-6_15