

# **Unity University**

**College of Engineering, Technology and Computation Sciences**

## **Department of Computer Science**

**Communication organizer (Webook)**

### **Chapter Three**

Group Memebbers

- 1)Dagmawi Zewdu .... Uu61480r
- 2)Alazar Eshetu .....UU71208r
- 3) jema1 abdulkedir .....UU71203r
- 4)Dawit akebergne..... uu70842
- 5)Wendowesen shumete.....uu69100r

Advisor: Gulelat Yohannes

April 3 2021

### 3.1 Introduction

Requirements analysis is a process that focuses on the tasks that determine the needs or conditions to meet the new or improved product considering the possibility of conflicting requirements, analyzing, documenting, validating, and managing software system requirements.

It focuses on the tasks that determine the needs or conditions to meet the new or altered product or project, taking account of the possibly conflicting requirements of the various stakeholders, analyzing, documenting, validating, and managing software or system requirements [Wikipedia 2021].

Requirements analysis is critical to the success or failure of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design [Wikipedia 2021].

In this chapter, we will focus on system analysis on **webook**. The following core activities will be elaborated in detail

- Current and Proposed Systems overview
- Functional and Non-functional Requirements
- System Models and use case diagrams

### 3.2 Current system overview

As elaborated in chapter one there are many ways of communication in an organization. We tried to classify them based on their way of expression, the direction of movement, and formality of communication.

**Written communication** in an organization is one way of communication that can occur through letters, reports, and memos, etc. These communications are mostly written through computer software. You can print out the document you write and give it to the office that is concerned, employee, or manager.

Most organizations do everything manually. For interoffice communication, they have to write a document and give it to the concerned offices by going to their departments. For upward communications, an employee will write a document give to the manager's secretary after that the manager will review the document. For downward communication, an employee will receive a document from his manager through a manager's secretary or directly.

Some organizations implement instant messaging, emails, and other electronic communication media to send a document or an attachment.

When employees or managers receive a document, it will be documented in a department office. Thus, documenting a paper is a difficult task it needs huge effort to organize and collect to put everything in place.

To tackle this kind of problem the documents have an identification number, date, and employee name. There is also a piece of additional information that tells which department office is the recipient.

Let's look at the different forms of written communication in their current system application.

**Letters** follow some channel that guides how they are written and send to a recipient. Letters are written computer software their guidelines can be different from organization to organization. But mostly they have to write organization information, letter identification number, date, department name, employee name, title, description, receiver department, receiver employee or manager name, and greeting note

A letter is written to a particular individual (employee). But with the same description content, they can be written to multiple audiences. The only difference can be the receiver's name.

Letters can also be written for approval. That means if an employee needs approval for something, he writes a letter to a manager seeking validation. This type of letter is called **letters of approval**.

Seeking validation or approval by a letter can go through different methods. They can be linear, iterative, or both.

**Linear Approval methods**- is the approval method when letters are approved on a simultaneous level. A letter doesn't need the approval of another manager.

**Iterative Approval method** - is approval method when letters are approved step by step. one step comes from another step

**Memos** are another type of written communication. As we mentioned in chapter one unlike letters memos are an informal type of communication. They are precise and short. In most cases, memos are written by managers because a memo's main purpose is to inform a company of an event, action, and observance.

Memo writing is also different from organization to organization but mostly a memo has a header that indicates who sent it and who the intended recipients are. Pay particular attention to the title of the individual(s) in this section. Date and subject lines are also present, followed by a message that contains a declaration, a discussion, and a summary.

**Reports** are also another type of written communication. Reports come in all sizes but are typically longer than a page and somewhat shorter than a book. The type of report depends on its function. The function of the report is its essential purpose, often indicated in the thesis or purpose statement. The function will also influence the types of visual content or visual aids, representing words, numbers, and their relationships to the central purpose in graphic, representational ways that are easy for the reader to understand. The function may also contribute to parameters like report length (page or word count) or word choice and readability.

**Instant messaging** is also another type of informal communication. This communication can be done by using an instant messaging application that is designed for an organization level. Among the different instant messaging [slack](#), [hipchat](#), [stride](#), [Microsoft-team](#) and [Flowdock](#) are the popular ones

### 3.3 Proposed System Overview

**Webook** proposes a web service controlled by an organization's employees. It includes subsystems for communication. Employees can express their communication without any barrier unless if they are not connected to a computer that is connected to the server of the organization.

**Webook** mainly focuses on written methods of communication. It automates a letter, memo, report, and instant messaging system. The below description tells us how we can do that

**Letters-** in automating the letter system from letter creation, progress, and final output (validation or reviewing) process should be considered. When letters are created webook will provide different templates. The templates are written by a webook system admin. They contain texts that contain basic information to fill in the form. The user can also select a blank form. He can write edit from templates or form blank form.

Webook also let choose between letter type (approval or normal letter). if a user chooses an approval letter, he has to select approval managers and their stage to approve. The user can also choose the recipients. The recipients and the approval managers can be single or multiple.

After the letter creation, the creator of the letter can see the progress and who has seen and take a decision (approve or disapprove) of the letter. When the recipient receives a letter, he can make a decision (if it is an approval letter) and also can print out the letter.

**Memos-** in the following conditions you can write a memo on webook

- ❖ When a user approves or disapproves a letter he can write a memo that describes why he approve or disapprove the letter. Webook automatically gives text-area to fill description as feedback to the letter.
- ❖ When a user announces a message to the desired audiences.
- ❖ When a manager wants to clear out something that is a rumor

A memo can be initialized from certain conditions like when reviewing and approving a letter or from a blank document you can create and write a memo. You have to select also the recipients' user's name.

**Reports** – a manager can ask for a report from a fellow employee. Webook has a request method to ask an employee to submit a report file. It also has a submission date which is used to notify when to submit. When asking for a request file the manager also should specify what to include in the report. After requesting the requested user should submit the report by attaching a document.

**Instant messaging-** webook has a chatbox for exchanging messages. Webook also has a list of all employees to contact. Employees can also share files.

In addition to this webook identifies the type, name, and role of a user when he login to the system. This is useful to identify the pages and functionalities to display. For instance, if the user is an admin he will be directed to the admin page and can perform admin functionalities.

### 3.3.1 Functional requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team. Generally, functional requirements describe system behavior under specific conditions.

**Webook** performs the following functionalities.

- ✓ Create employees based on their role in the organization
- ✓ Edit delete registered employees
- ✓ Create, edit and delete roles of employees.
- ✓ Show all registered employee
- ✓ Create, edit, show, and delete departments
- ✓ Create, edit, and show company information.
- ✓ Create, edit and delete letter templates
- ✓ Login and create an account
- ✓ Activate and deactivate users
- ✓ Reset passwords for forgotten passwords.
- ✓ Identify users based on their role.

- ✓ Create, edit, show, and delete letters.
- ✓ For approval letter add, edit and delete approval stages.
- ✓ Create, edit, show, and delete reports and memos.
- ✓ Show progress of letters, reports, and memos.
- ✓ Chat-box for communicating.
- ✓ Name, role, and last seen status on the chat-box.
- ✓ Chat lists and history of chatting.
- ✓ Share files on the chatbot.
- ✓ announce messages.
- ✓ Show notification when receiving new messages, letters, reports, or memos.

### 3.3.2 Non-Functional Requirements

Non-functional requirements describe how a system must behave and establish constraints of its functionality. This type of requirement is also known as the system's quality attributes [Alexsoft 2021].

**Webook** performs the following non-functional requirements.

- ✚ The system should be available for all employees.
- ✚ The system will have an interface that is easy to understand easily.
- ✚ Performance should not decrease.
- ✚ Data integrity must be insured.
- ✚ Secure authentication must be enabled.
- ✚ Secured data communication must be enhanced.
- ✚ The website should work flawlessly from any web browser.
- ✚ The system should always give real-time data

### 3.4.1 Essential Use Case Modelling

A use-case model is a model of how different types of users interact with the system to solve a problem. As such, it describes the goals of the users, the interactions between the users and the system, and the required behavior of the system in satisfying these goals [utm 2007].

#### 3.4.1.1 Use Case Diagram

Use case diagrams have a building block it commonly depends on.

- **Actors:** The users that interact with a system. An actor can be a person, an organization, or an outside system that interacts with your application or system. They must be external objects that produce or consume data.
- **System:** A specific sequence of actions and interactions between actors and the system. A system may also be referred to as a scenario.

- **Goals:** The result of most use cases. A successful diagram should describe the activities and variants used to reach the goal.

In this sense, the system is a **webook**. The actors are employees, managers, and system admin. They are connected to their goal through the line and the goals are circled in the following figure.

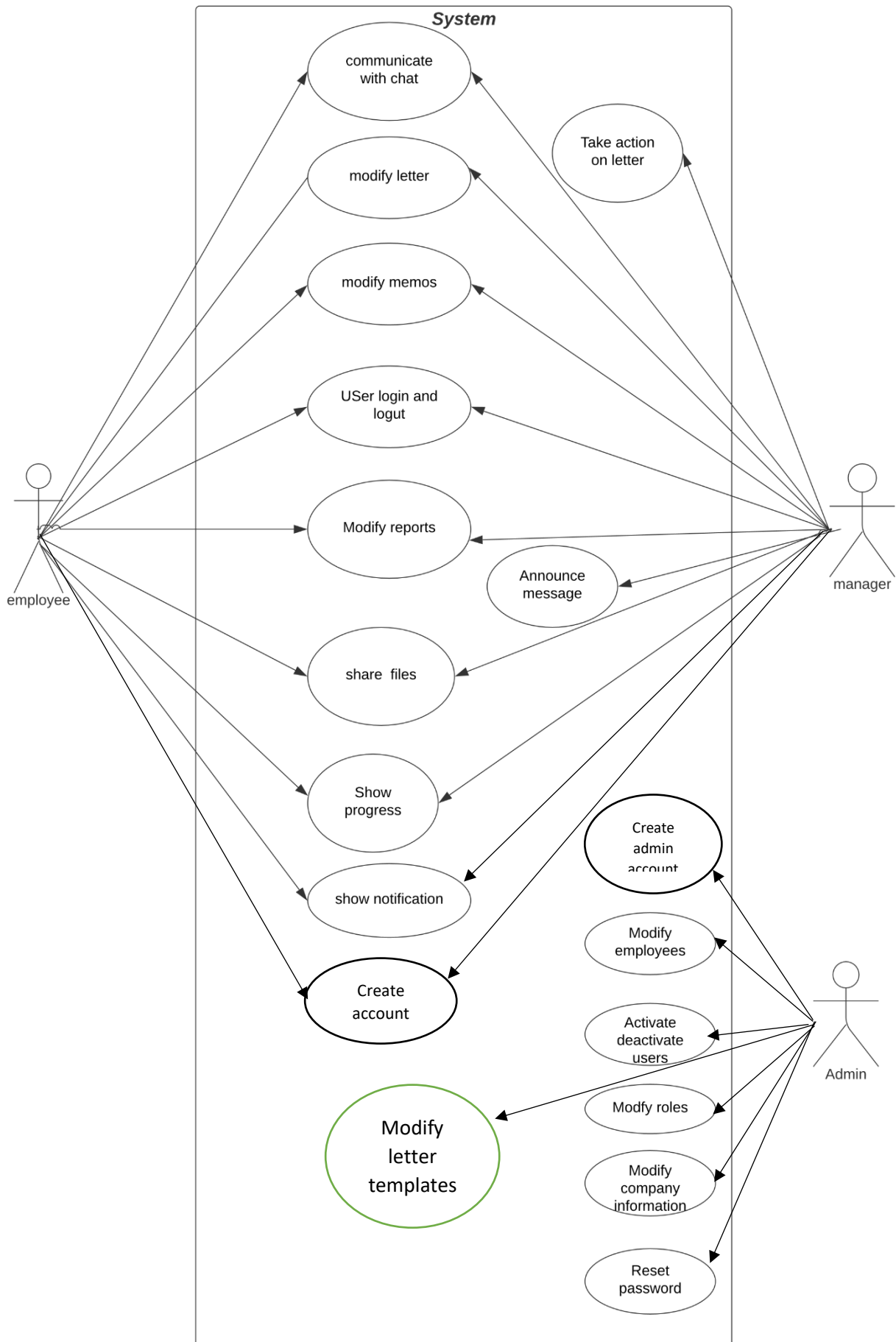




Fig 1.0 usecase diagram

### 3.4.1.2 Use Case Documentation

The Use Case Document is a business document that provides a story of how a system, and its actors, will be utilized to achieve a specific goal. An effective Use Case should provide a detailed step-by-step description of how the system will be used by its actors to achieve the planned outcome. The purpose of the Use Case is to tie the needs of the system to the design parameters of the system to ensure that the completed system achieves the goals established by the requirements. [projectmanagementdocs 2017]

**Webook** use case identifier starts with UC- and is followed by a number.

Name	<b>User login and logout</b>
Identifier	UC-01
Description	Login and logout from webook
Actor	Employees, Managers, and admins
Precondition	Users must create an account
Postcondition	Redirect to the landing page
Basic course of action	<ul style="list-style-type: none"><li>➤ A user enters his employee id.</li><li>➤ A user enters his password.</li><li>➤ A user presses the login button</li><li>➤ forget password ask admin to reset password</li></ul>

Name	<b>Create user Account</b>
Identifier	UC-02
Description	The user creates an account to use the system
Actor	Employees, Managers, and admins
Precondition	Employee or manager should be registered by the admin first
Postcondition	Redirect to a landing page
Basic course of action	<ul style="list-style-type: none"><li>➤ Users employee id</li><li>➤ Users enter their password</li><li>➤ Users enter to create an account button</li></ul>

Name	<b>Create admin account</b>
Identifier	UC-03
Description	Create an admin account this is done once only the system starts for the first time
Actor	Admin
Precondition	None
Postcondition	Redirect to enter Company information
Basic course of action	<ul style="list-style-type: none"> <li>➤ Enter admin employee id</li> <li>➤ Enter admin password</li> <li>➤ Press create account button</li> </ul>

Name	<b>Modify company information</b>
Identifier	UC-04
Description	Add company data when the system starts for the first time
Actor	Admin
Precondition	None
Postcondition	Redirect to login page
Basic course of action	<ul style="list-style-type: none"> <li>➤ Enter company name, address, and some other basic information</li> <li>➤ Enter company logo</li> <li>➤ Press submit button</li> </ul>

Name	<b>Modify employees</b>
Identifier	UC-05
Description	Admin create, edit, and delete user data
Actor	Admin
Precondition	User must be logged via the admin account
Postcondition	employee's data modified (created, updated, or deleted)
Basic course of action	<ul style="list-style-type: none"> <li>➤ Create an employee by entering basic information</li> <li>➤ Edit employee's data that are added</li> <li>➤ Delete employee</li> </ul>

	➤ Read and search employees
--	-----------------------------

Name	<b>Modify roles</b>
Identifier	UC-06
Description	Create, edit, delete and read the role of the organization
Actor	Admin
Precondition	User must be logged via an admin account
Postcondition	Organization role data modified ( created, updated, or deleted)
Basic course of action	<ul style="list-style-type: none"> <li>➤ Create a role</li> <li>➤ Set role level</li> <li>➤ Edit and delete role</li> <li>➤ Read and search role</li> </ul>

Name	<b>Modify department</b>
Identifier	Uc-07
Description	Create, edit, delete and read the departments of the organization
Actor	Admin
Precondition	User must be logged via an admin account
Postcondition	Organization departments data modified ( created, updated, or deleted)
Basic course of action	<ul style="list-style-type: none"> <li>➤ Create an employee by entering basic information</li> <li>➤ Edit employees data that are added</li> <li>➤ Delete employee</li> <li>Read and search employees</li> </ul>

Name	<b>Modify letter templates</b>
Identifier	Uc-08
Description	Create, edit, delete and read a letter template
Actor	Admin

Precondition	User must be logged via an admin account
Postcondition	Letter templates modified ( created, updated, or deleted)
Basic course of action	<ul style="list-style-type: none"> <li>• Create ,edit ,delete letter template</li> </ul>
Name	<b>Activate and deactivate employee</b>
Identifier	UC-09
Description	Activate and deactivate user account
Actor	Admin
Precondition	User must be logged via an admin account
Postcondition	User account activated or deactivated
Basic course of action	<ul style="list-style-type: none"> <li>➤ User account activated</li> <li>➤ User account deactivated</li> </ul>

Name	<b>Reset password</b>
Identifier	UC-10
Description	Reset password for forgotten passwords
Actor	Admin
Precondition	User must be logged via an admin account
Postcondition	A user account is deleted and they enter a new password
Basic course of action	<ul style="list-style-type: none"> <li>➤ Employee request to reset password</li> <li>➤ Admin Seen the request and reset the password</li> </ul>

Name	<b>Communicate with chat</b>
Identifier	UC-11
Description	Users real-time communication
Actor	Employees and managers
Precondition	A user must be logged in by manager or employee account
Postcondition	the user sends messages.
Basic course of action	<ul style="list-style-type: none"> <li>➤ User writes a message on the chatbox.</li> <li>➤ User send message</li> </ul>

Name	<b>Modify letter</b>
Identifier	UC-12
Description	User can create, edit and delete letter
Actor	Employees and managers
Precondition	A user must be logged in by manager or employee account
Postcondition	User modify letter
Basic course of action	<ul style="list-style-type: none"> <li>➤ Choose letter type</li> <li>➤ Write letter</li> <li>➤ Add approval stage for approval letters</li> <li>➤ Send letters</li> </ul>

Name	<b>Modify memos</b>
Identifier	UC-13
Description	User can create, edit and delete
Actor	Employees and Managers
Precondition	A user must be logged in by manager or employee account
Postcondition	User modifies memos
Basic course of action	<ul style="list-style-type: none"> <li>➤ Select letter to refer</li> <li>➤ Write memo</li> <li>➤ Send memos</li> </ul>

Name	<b>Write reports</b>
Identifier	UC-14
Description	Write reports
Actor	Employees and Managers
Precondition	User modify reports
Postcondition	User modifies reports
Basic course of action	<ul style="list-style-type: none"> <li>➤ Create reports</li> <li>➤ Edit and delete reports</li> <li>➤ Send reports</li> </ul>

Name	<b>Share files</b>
Identifier	UC-15

Description	Share files between users
Actor	Employees and managers
Precondition	A user must be logged in by manager or employee account
Postcondition	User sends a file.
Basic course of action	<ul style="list-style-type: none"> <li>➤ User choose file to send</li> <li>➤ User send file</li> </ul>

Name	<b>Show progress and notification</b>
Identifier	UC-16
Description	Show progress of the letter, memos, and reports whether the receiver sees and reacts to it.
Actor	Employees and managers
Precondition	A user must be logged in by a manager or employee account
Postcondition	User understands the progress
Basic course of action	<ul style="list-style-type: none"> <li>➤ Select a letter, memo, or report</li> <li>➤ Press show progress button</li> </ul>

Name	<b>Show notification</b>
Identifier	UC-17
Description	Show when a new action is performed
Actor	Employee and manager
Precondition	A user must be logged in by manager or employee account
Postcondition	Users see when a new action is performed
Basic course of action	Show notification icon and bar when the user browses

Name	<b>Take action on a letter</b>
Identifier	UC-18
Description	Show when a new action is performed
Actor	Manager
precondition	A user must be logged in by the manager

postcondition	A Letter is approved or unapproved
Basic course of action	Select a letter and take actions

Name	<b>Announce message</b>
Identifier	UC-19
Description	Show when a new action is performed
Actor	Manager
precondition	A user must be logged in by manager or employee account
postcondition	A Message is written on employees dashboard
Basic course of action	Write a message and press announce button

### 3.4.2 Essential UI Prototype

User interface (UI) prototyping is an iterative analysis technique in which users are actively involved in the mocking-up of the UI for a system. The figures below are the essential UI prototype of **Webook**.

**Webook** has multiple pages to show. But other than log in and logout pages the rest page relies on two common side-navbar. it is the admin **side navbar** and **the employee side navbar**.


Webook also have a header navbar which basically display logo of the company , contact admin option and employee information.



New to webook  
create account

Forget password

Fig 2.0 Login page



can't Create account? it means you are  
not registered to this system please  
contact admin

fig 3.0 create Account page



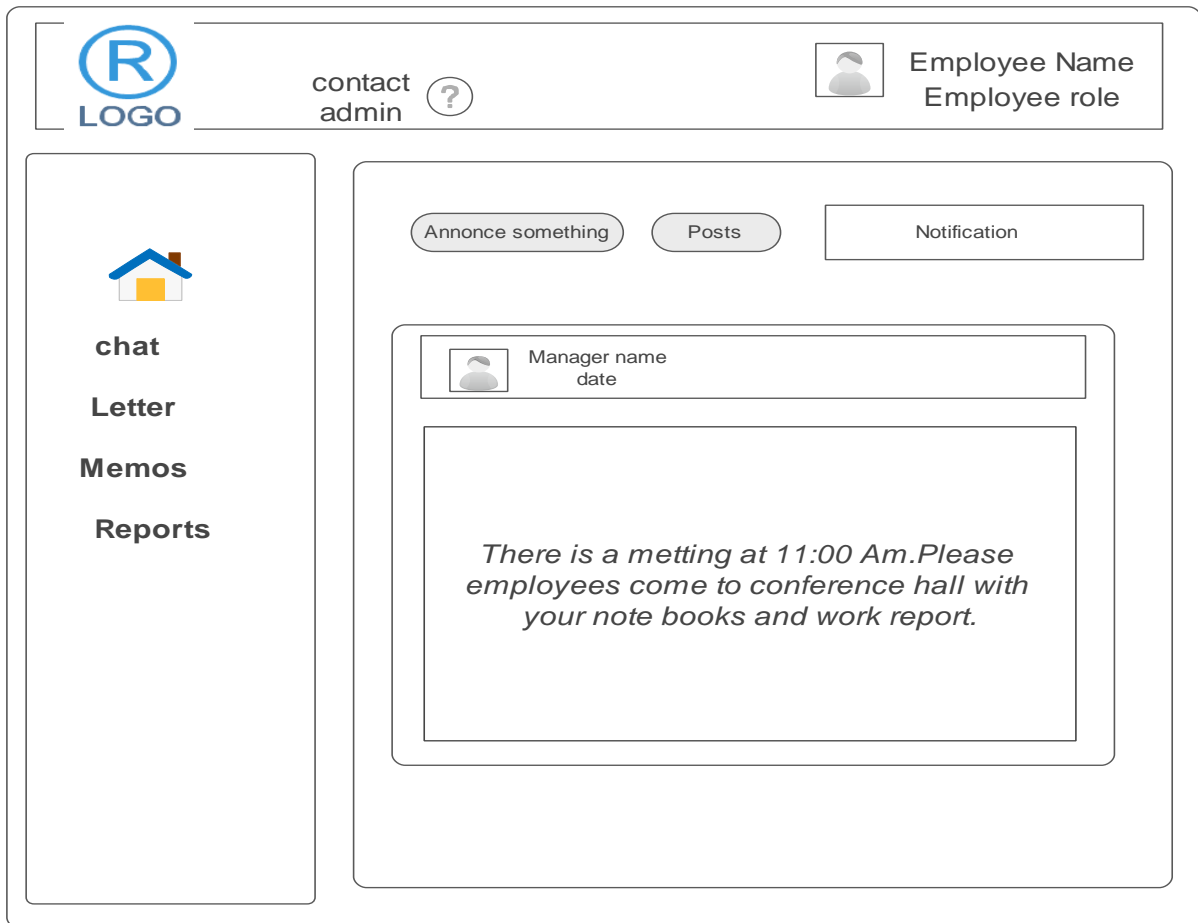


Fig 4.0 Landing page (Home page)

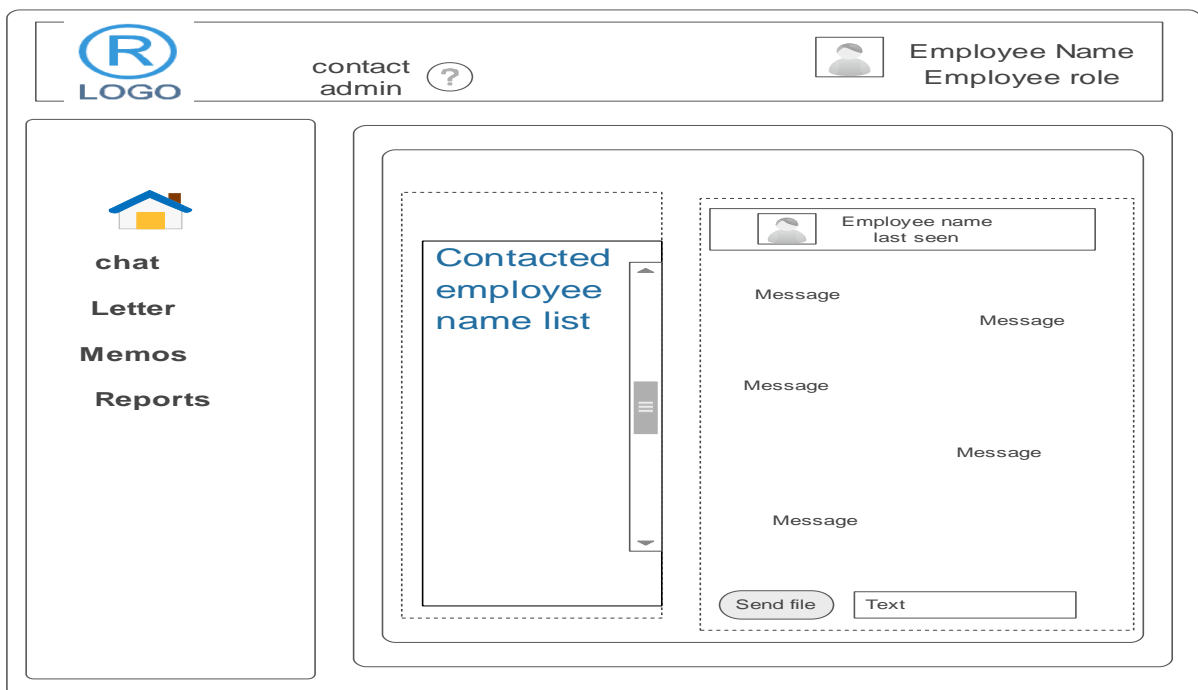





Fig 5.0 Chat page (Home page)

  
LOGO


contact  
admin

 Employee Name  
Employee role


  
chat  
Letter  
Memos  
Reports


InboxoutBox


fig 6.0 Letter page

  
LOGO

contact  
admin

 Employee Name  
Employee role

  
chat  
Letter  
Memos  
Reports

created MemosLetters to write memos


Fig 7.0 Memos page

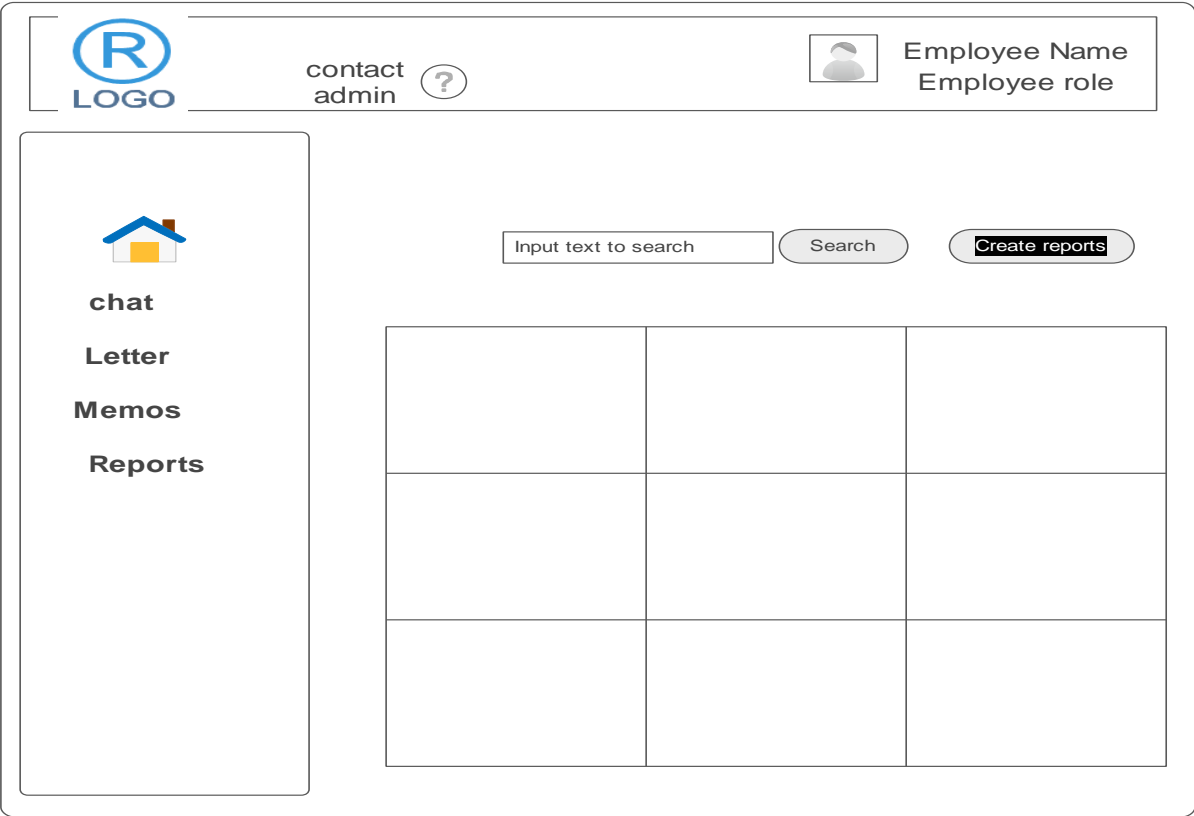


Fig 8.0 Reports page

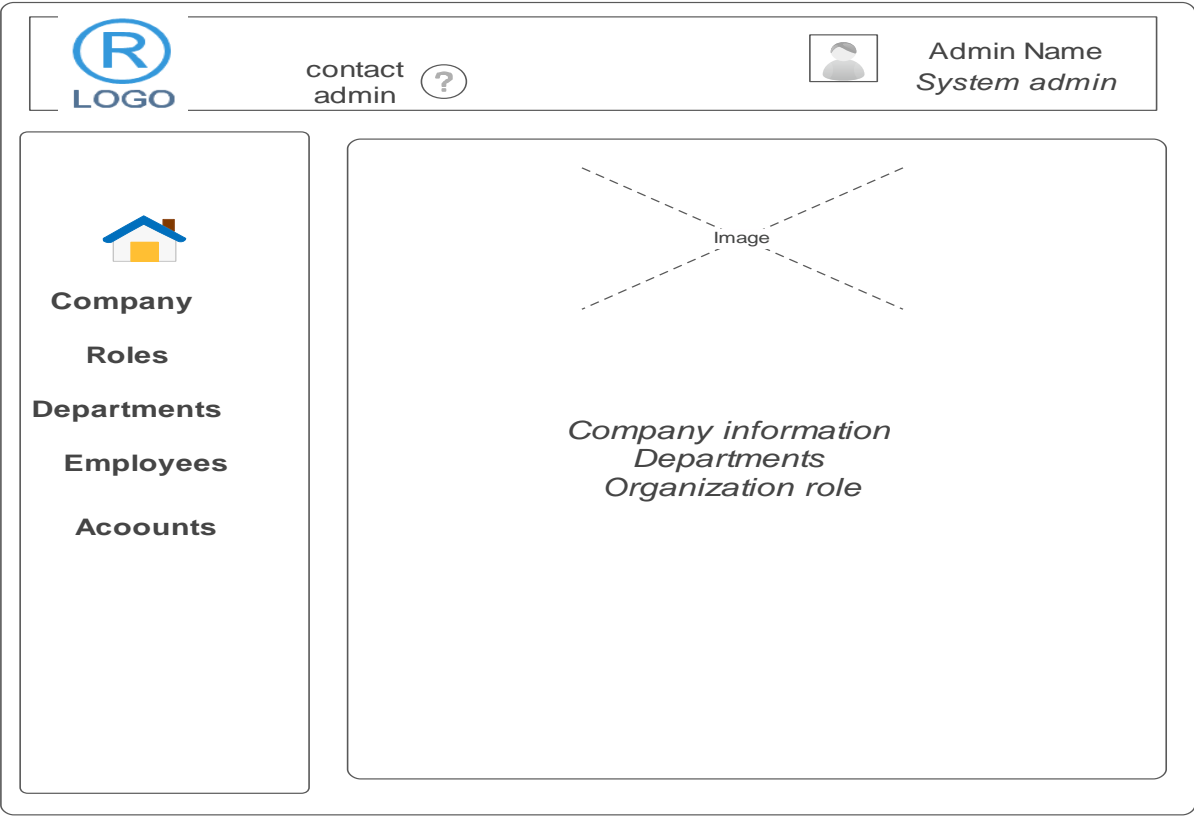


Fig 9.0 Admin page

### 3.4.3 User Interface Flow Diagram

User interface-flow diagrams - also called storyboards, interface-flow diagrams, windows navigation diagrams, and context-navigation maps - enable you to model the high-level relationships between major user s elements and thereby ask fundamental usability questions. [Agilemodeling 2016]

User interface-flow diagrams are typically used for one of two purposes. First, they are used to model the interactions that users have with your software, as defined in a single-use case. For example, a use case can refer to several screens and provides insight into how they are used. Based on this information, you can develop a user interface flow diagram that reflects the behavioral view of the single-use case. [Agilemodeling 2016] .

When **Webook** runs the first time there is no data to start and process it. So it first asks company and administrator information to control the system.

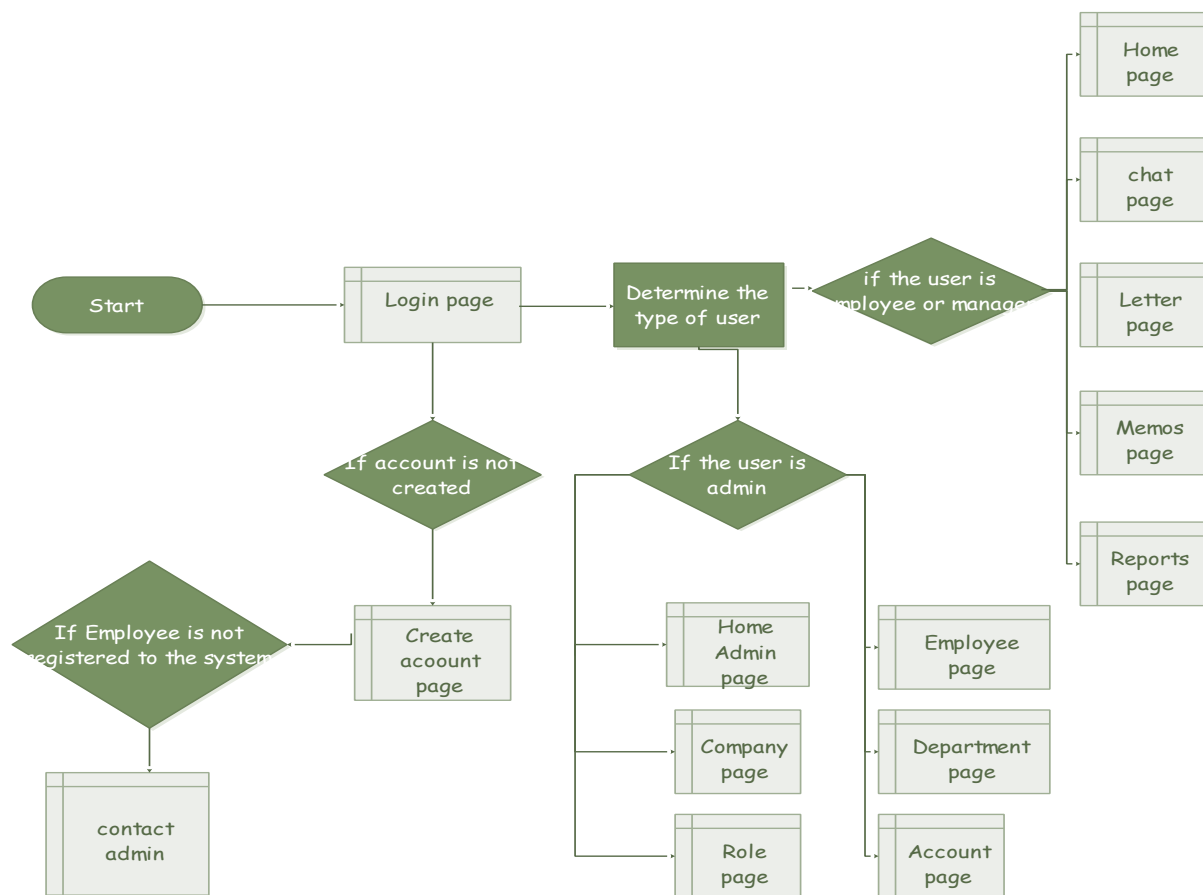
The following diagram explains how the system runs for the first time.



Fig 10.0 UI flow diagram for the first time

After the administrator inserted employee's data. Employees can start using the system by creating an account. If employees id is unknown that means they are not registered to the system. They can contact the admin to register them.

The following UI flow chart explains how the employee access **webook**



### 3.4.4 Supplementary Specifications

Supplementary specifications define the Requirement that is not easily defined in the use case Model. Requirements such as legal standards, quality aspects, supportability, and execution criteria of the system.

#### 3.4.4.1 Business rules

A business rule is intended to assert constraints that govern the business structure.

Here is a list of business rules for our system:

Name	A company must have a server
Identifier	<b>WB-01</b>
Description	Since webook is a local system the company must have a server to deploy webook. It can be locally in the organization or in the cloud (internet).

Name	A user must be an employee of the organization
Identifier	<b>WB-02</b>
Description	A user must employee of the organization. Other users like customers are not supported

Name	An employee must be registered as a user by the system admin
Identifier	<b>WB-03</b>
Description	An employee must be registered as a user by the system admin. unregistered employees can not access the system. After registering an employee can access the system by creating a user account.

Name	An Employee must be registered with employee id
Identifier	<b>WB-04</b>
Description	When an admin registers an employee, It should register with a proper id

Name	User can not delete approved letter
Identifier	<b>WB-05</b>
Description	Approved can not be deleted because they are agreements of two parties.

Name	An employee must be registered as a user by the system admin
Identifier	<b>WB-06</b>
Description	An employee must be registered as a user by the system admin. unregistered employees can not access the system. After registering an employee can access the system by creating a user account.

### 3.4.4.2 Constraints

A constraint is the limitations we have in providing a solution that may affect the way we develop a system.

ID	Description
C-01	The system requires a reliable connection with the server
C-02	Budget Constraint
C-03	Time constraint- time has a huge impact on delivering this project on time. If we are behind schedule the project manager will rush the project. This will may have an effect on project quality

### 3.4.4.3 Change case

Change cases are used to describe new potential requirements for a system or modifications to existing requirements. Change cases can be identified throughout the course of our overall development efforts. Below is a list of change cases regarding our system: -

Change case	Customers involvement in the system
Likelihood	Certain, Medium-term plan
Impact	Customers have to involve in the organization's communication.

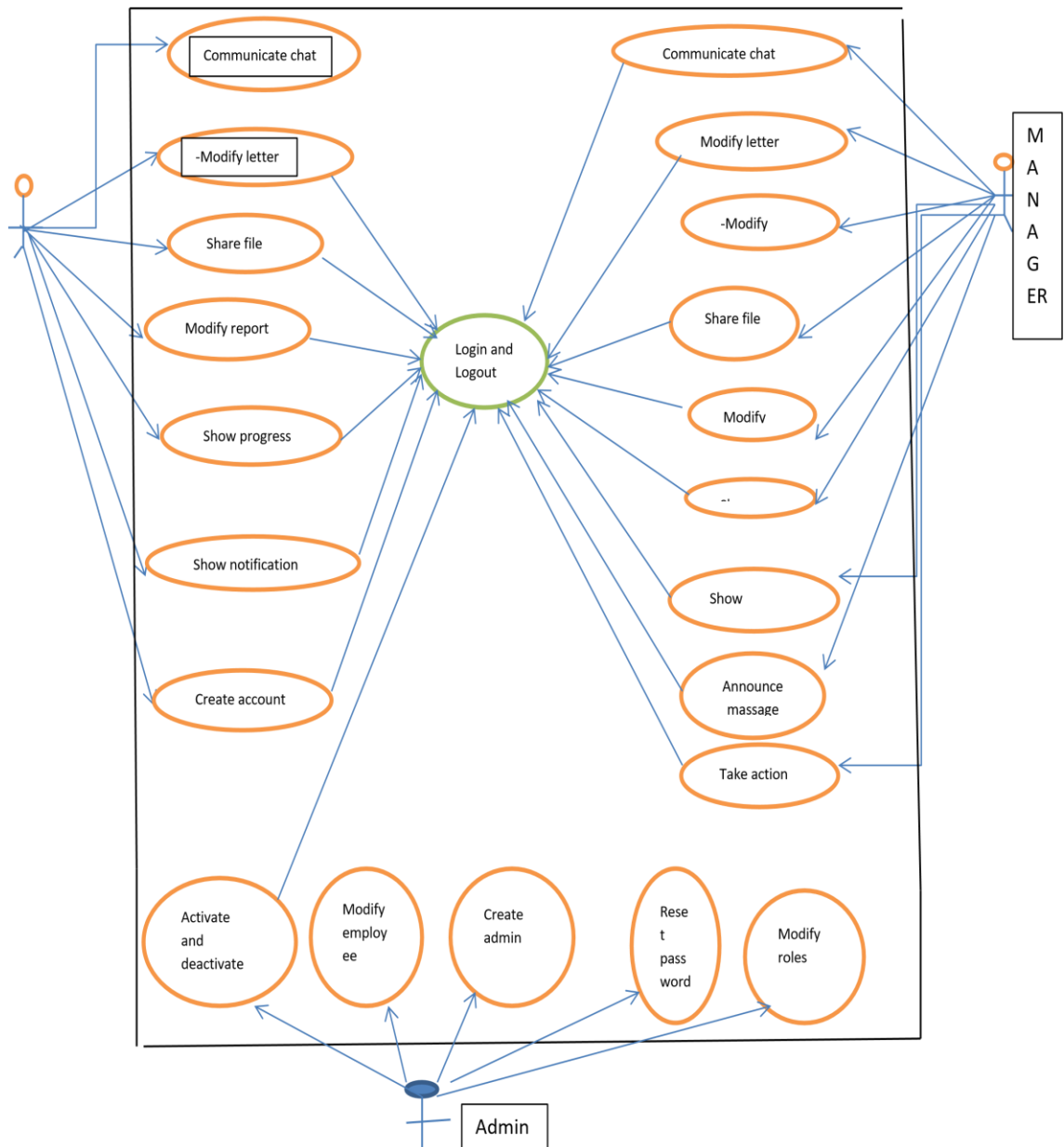
Change case	Hosting webook in a centralized server
Likelihood	Certain, Short-term plan
Impact	It will have a low impact. Because it is server-side configuration.

## 3.5 Analysis System Model

is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. System analysis is conducted to study a system or its parts to identify its objectives.

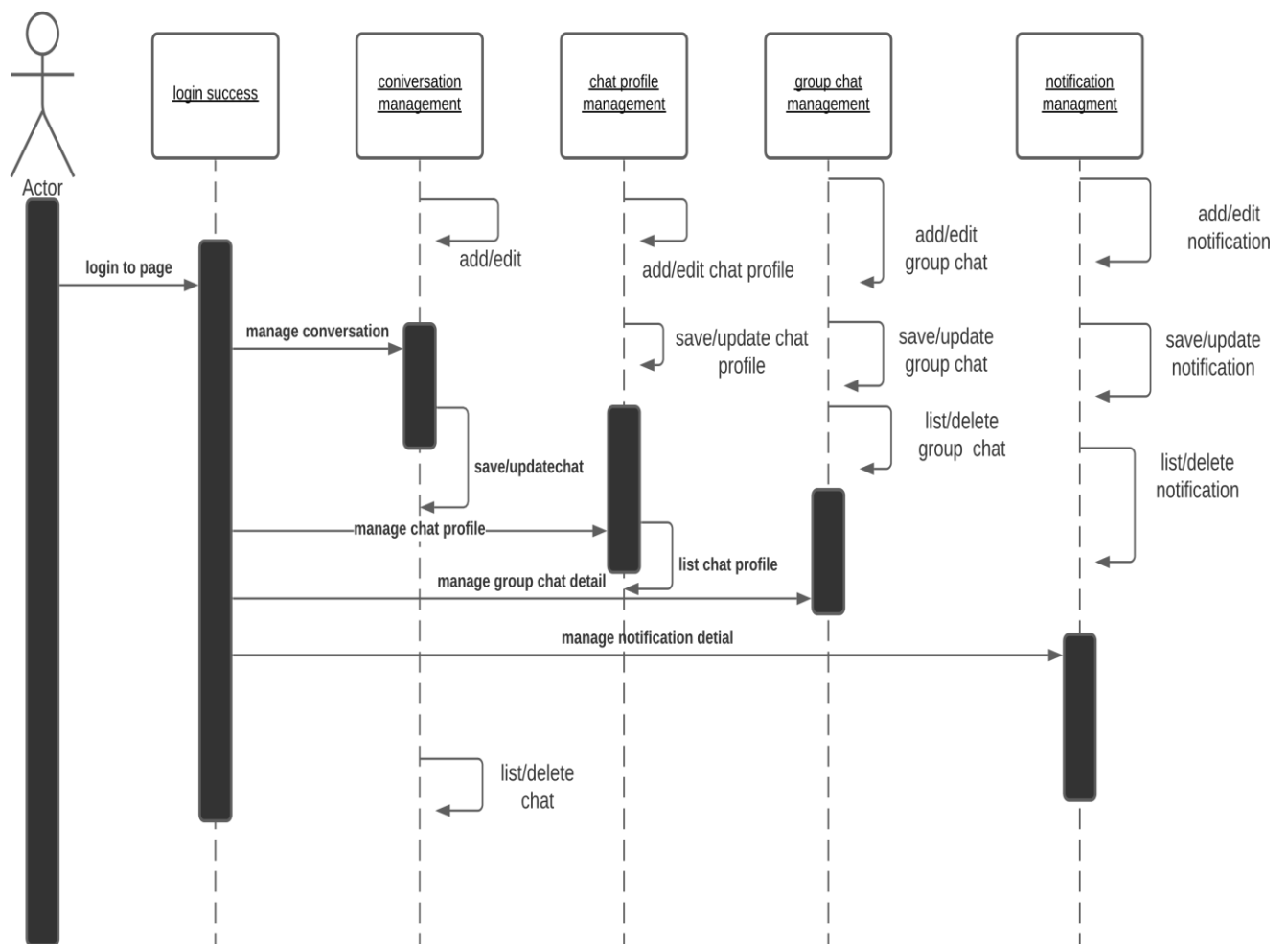
### 3.5.1 System Use Case Modeling

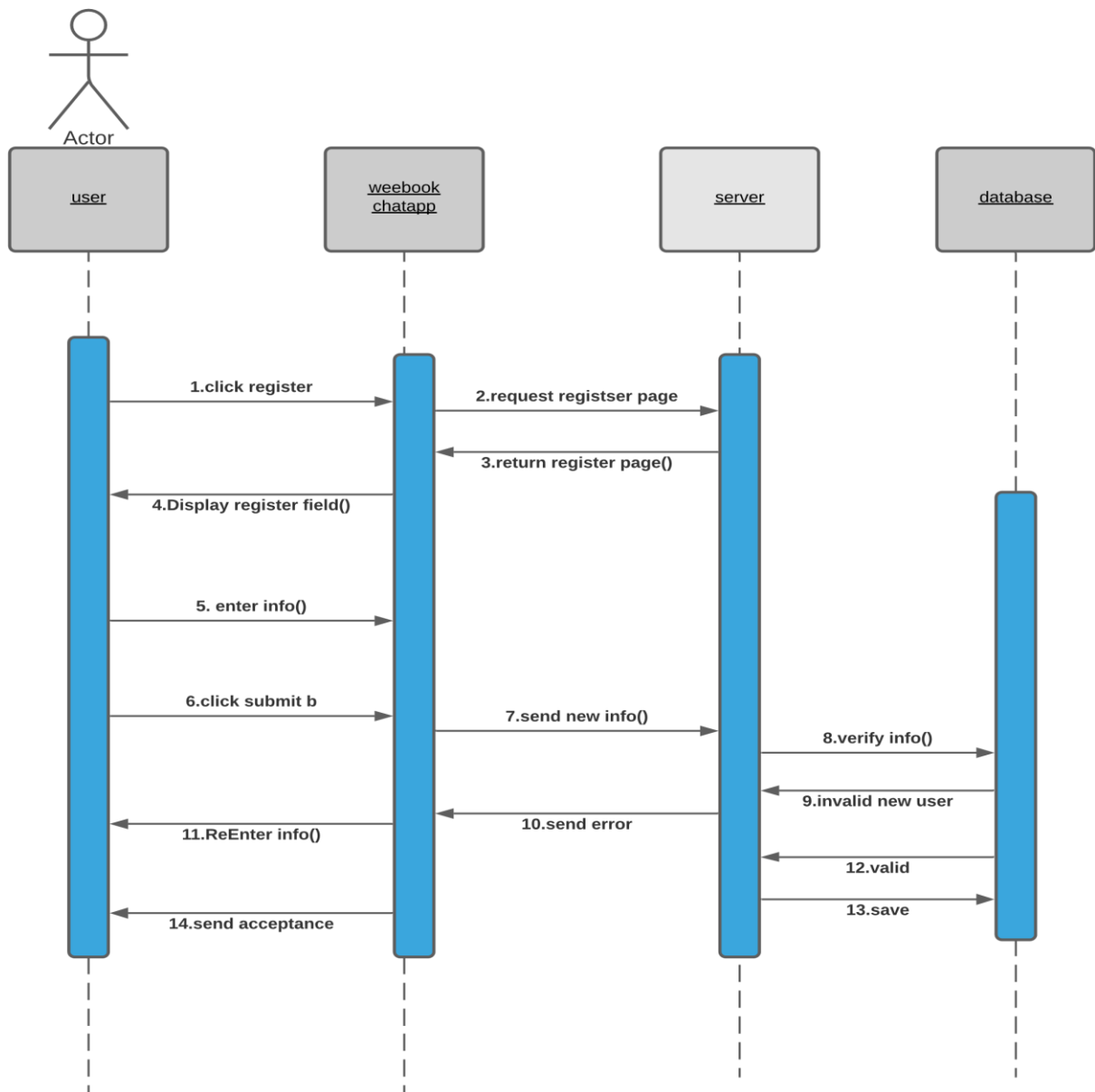
#### 3.5.1.1 System use case diagram

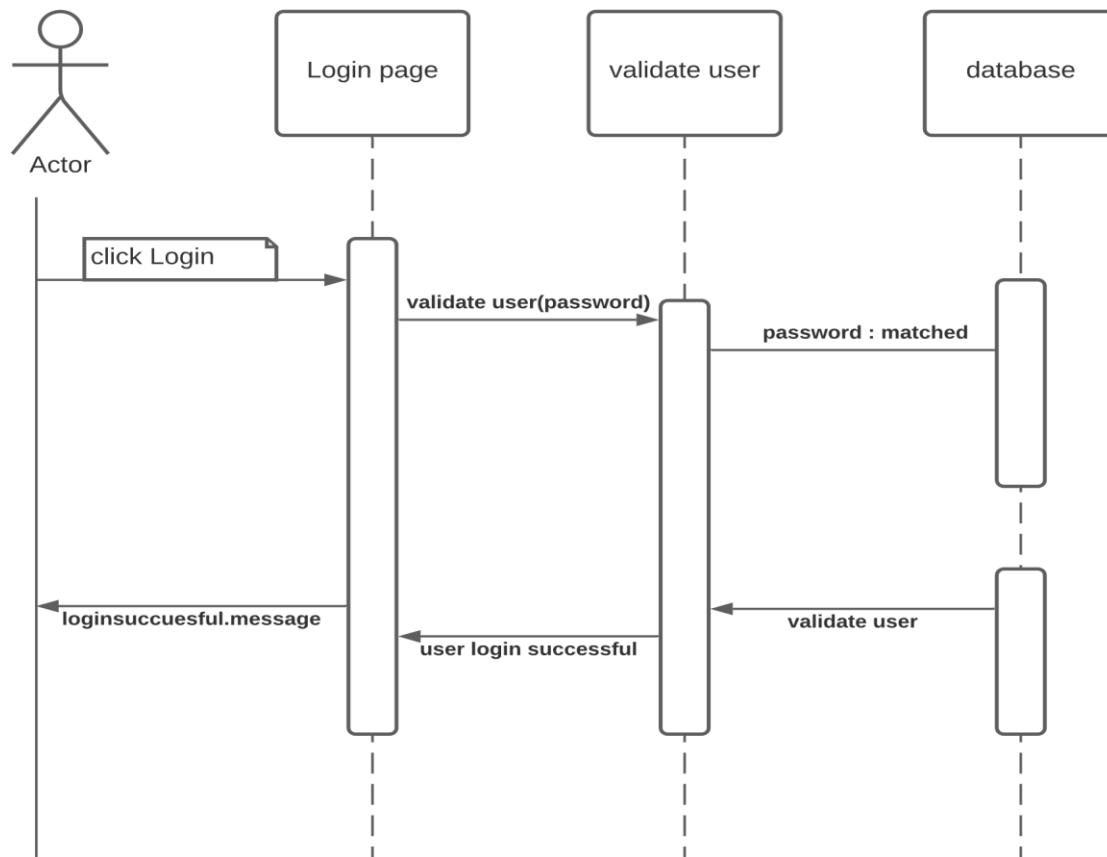


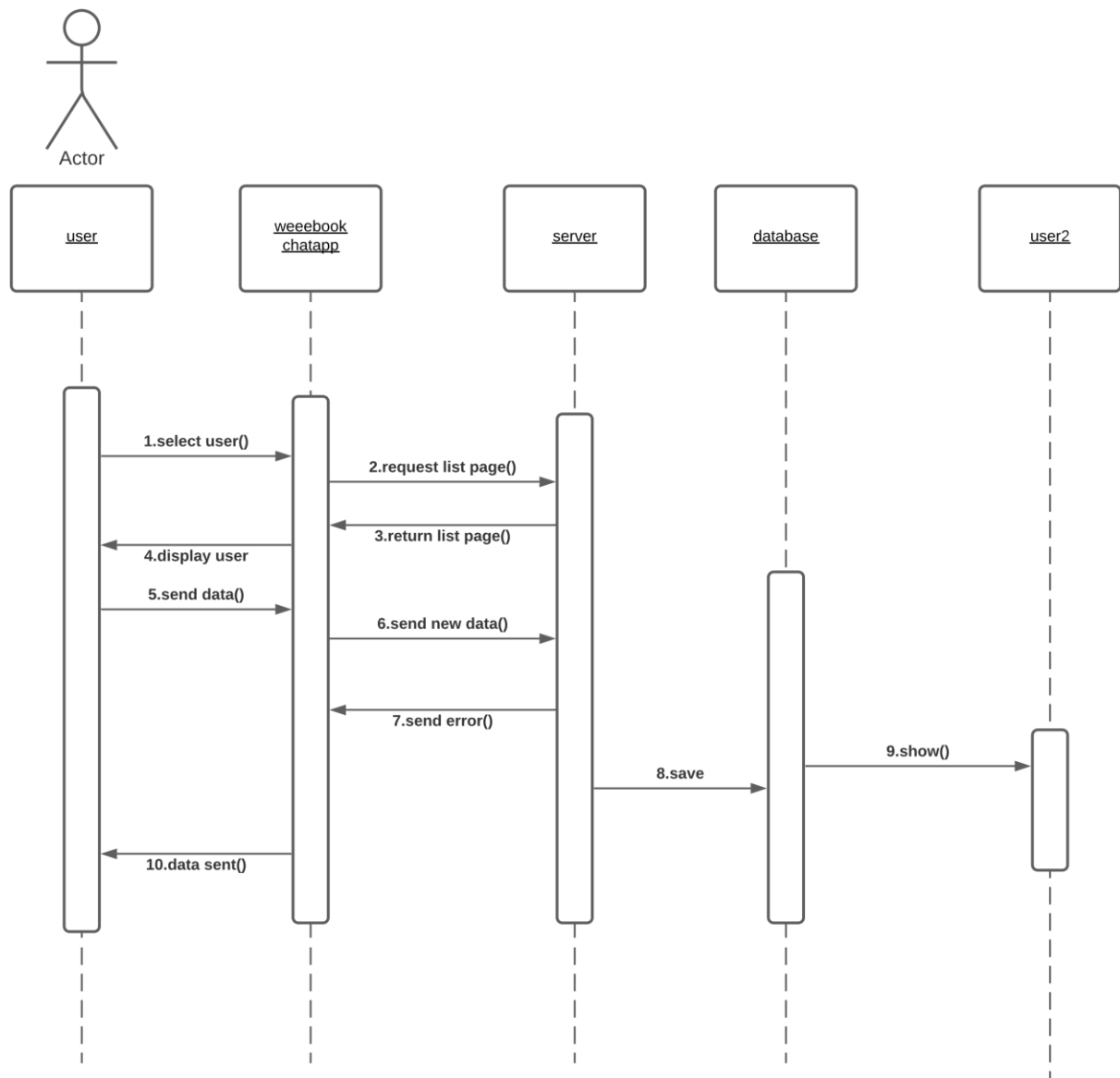
### 3.5.2 Sequence Diagram

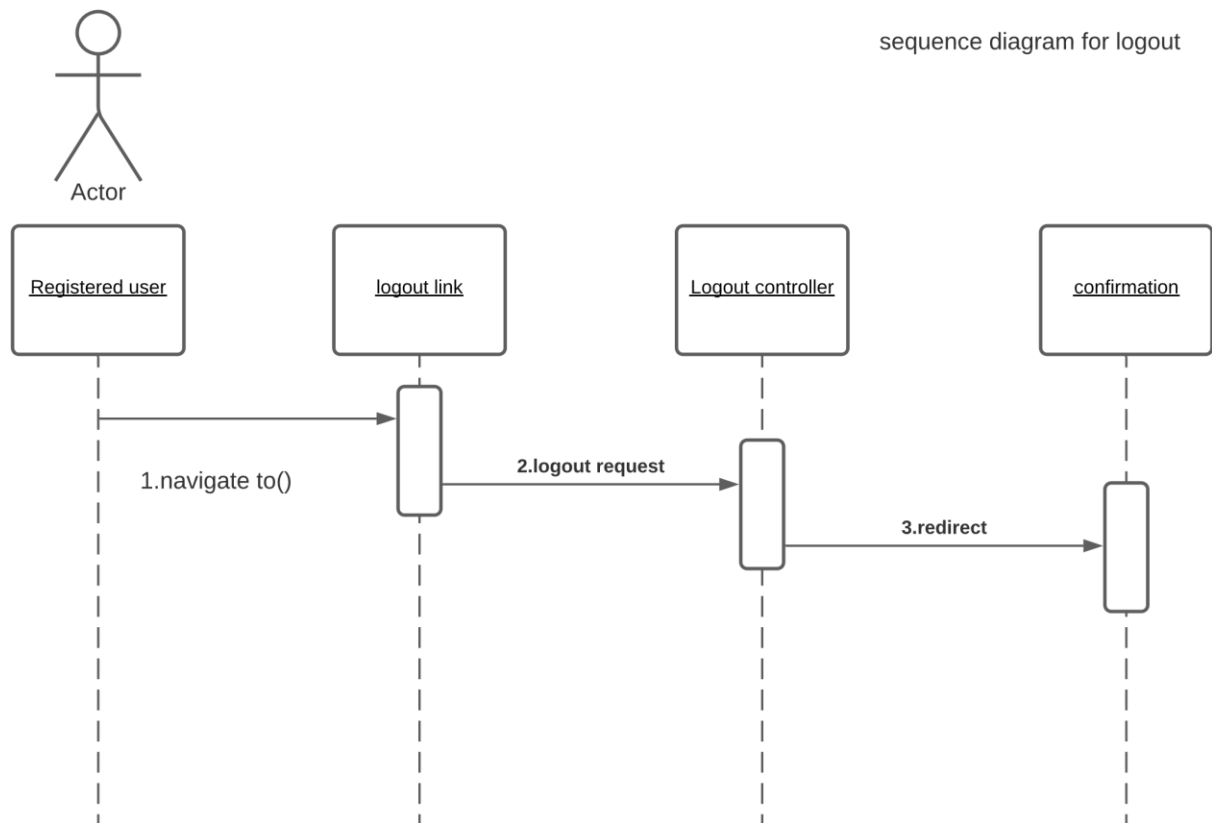




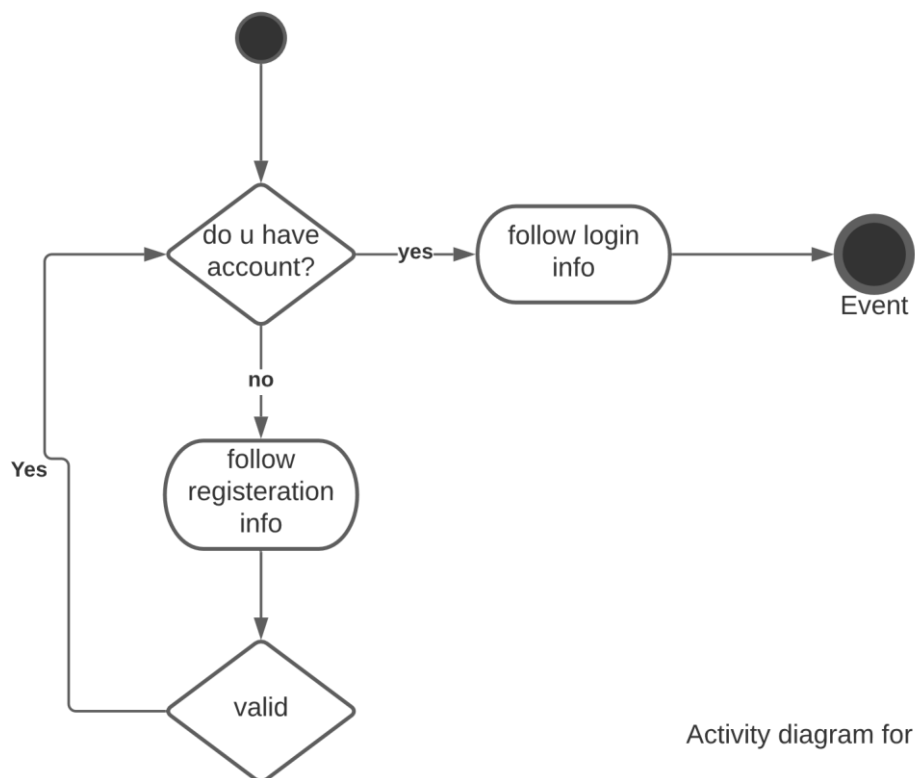




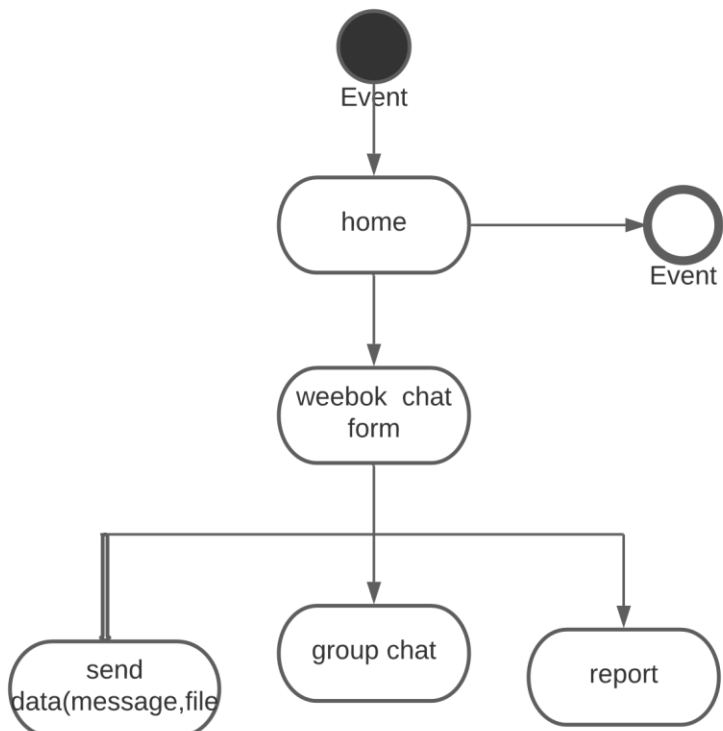
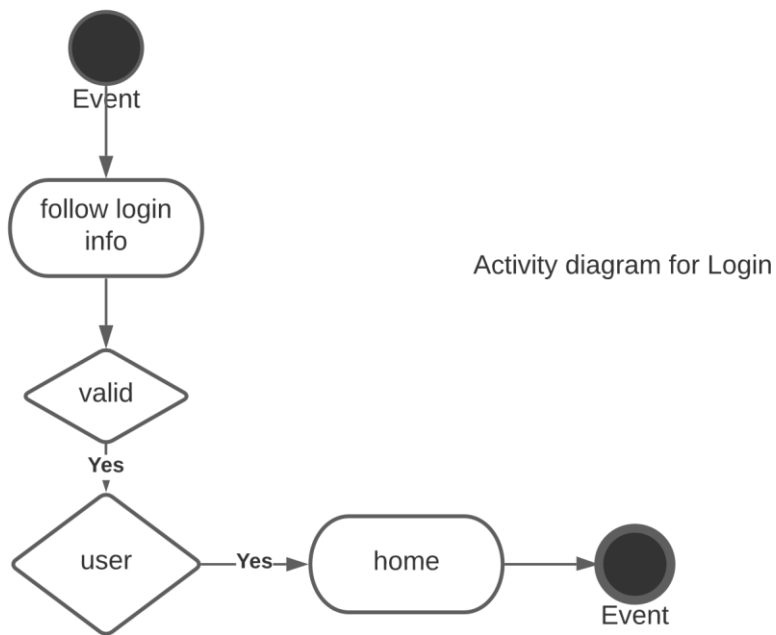


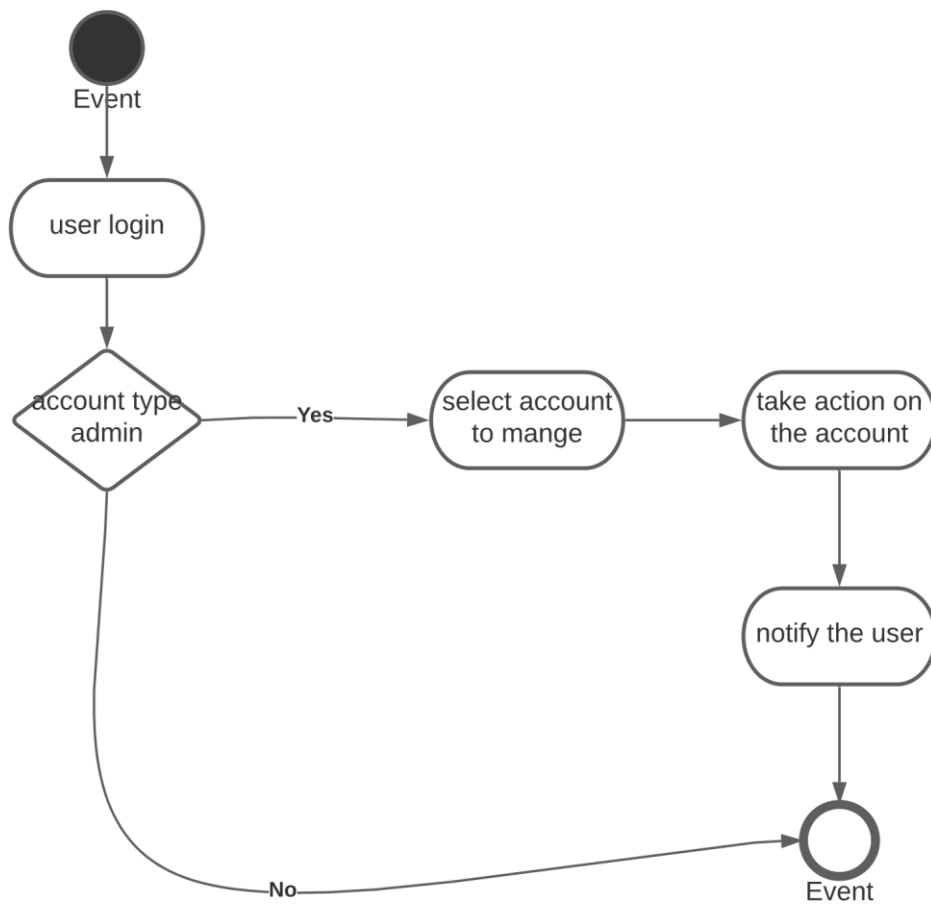


### 3.5.3 Activity Diagram



Activity diagram for Registration





## Reference

**Wikipidea(2021).**Requirements analysis.*wikipedia.org*.

[https://en.wikipedia.org/wiki/Requirements\\_analysis](https://en.wikipedia.org/wiki/Requirements_analysis)

**Alexsoft(2021).** Functional and non-functional requirements specification and types.

[altexsoft.com. https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types](https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types)

**Utm(2007).** Concept: Use-Case Model. Utm.mx .

[http://www.utm.mx/caff/doc/OpenUPWeb/openup/guidances/concepts/use\\_case\\_model](http://www.utm.mx/caff/doc/OpenUPWeb/openup/guidances/concepts/use_case_model)

**Projectmanagementdocs(2017).**Use Case Document. projectmanagementdocs.com.

<https://www.projectmanagementdocs.com/template/project-documents/use-case-document/#ixzz6qmxMVKos>

**Agilemodeling(2016).** User Interface Flow Diagrams. agilemodeling.com.

<http://agilemodeling.com/artifacts/uiFlowDiagram.html>



