Daffodil International University

Department: Software Engineering
Project Name: Online Admission System
Course Title : System Analysis And Design project
Course Code : SE231
Section : A
Teacher Initial : KBB
Group Name : SUNRISE
Member Name : Mst. Marzia khatun
ID : 193-35-2936
Member Name : Md. Saidur Rahman
ID : 193-35-2919
Date: 05 June 2021

1→Project Proposal

1.1. Overview

1.1.1. Background

As with most real world activities, there are numerous benefits to using a software system for university admissions. This system is to make the performance of certain tasks much faster than humans. This also means that these tasks can be done solely by the system, freeing up those involved to perform more important tasks. The simplicity of this project ensures the effectiveness of the system. Including that the system will enable the user to edit the data online before the last date of submission of application, as the user's status is being changed by admission staff i.e. selected and not selected. At the same time this system will also facilitate the admission staff to view the user profile and so on. The admission staff can view the user profile, can change the status of user i.e. selected or not selected, can change the last application for admission, can modify the last of admission and finally can switch themselves from the system.

1.1.2. Objectives

Online Admission System (OAS) will be a web based system which will resolve the traditional problem of maintaining students' records and admission procedure. As the numbers of web surfers are increasing day by day, a web base solution will be certainly more accurate than a traditional desktop base solution. The system will have different levels of users. The one who is not registered as a user will be able to view the basic information page. The users will be-

1. System Administrator

2. Active User

3. Inactive User

In this system here--

1. All users are able to view the basic information of this system.

2. Whoever is applying for admission, they must create a user id and password for registration.

3. After creating a user id and password they can do an application and submit an application form.

4. Registered users can edit the data online before the last date of submission of application.

5. Here administrators can view and check all student information.

1.1.3. Scope

As with most real world activities, there are numerous benefits to using a software system for college admissions. The most apparent to this project is the unification of the entire process.

Another aspect of a software system is the use of a central database. This database is the basis for all actions in the system and can be trivially updated and used to aid in all of the system's processes, meaning all of the required information is stored in one central location and thus is easily accessible. This is a far more reasonable storage method than a paper-based file system, where the time of traveling to and physically searching the records for the required information could be a burden. Human error could also be a factor in that mistakes could be made in the filing process which would not occur in a well written database system and mistakes or changes on physical records can be messy to correct.

The main objective of this system is to make the performance of certain tasks much faster than humans. This also means that these tasks can be done solely by the system, freeing up those involved

to perform more important tasks. The simplicity of this project ensures the effectiveness of the system.

Including that the system will enable the user to edit the data online before the last date of submission of application, as the user's status is being changed by admission staff i.e. selected and not selected. At the same time this system will also facilitate the admission staff to view the user profile and so on. The admission staff can view the user profile, can change the status of the user i.e. selected or not selected, can change the last of application for admission, can modify the last of admission and finally can switch themselves from the system.

1.1.4. Assumptions and Constraints

It is assumed that the user is comfortable with the computer or smartphone. The user should know how to fill up a online admission form. The user must have to register to full fill the form. The user who are registered must have good knowledge on web surfing.

The user interface is in English as a result people lacking in English skill will face difficulty in using the system. Login and password is used for identification of users and there is no facility for guests.

1.1.5. Dependencies and Risks

The user must have web access in order to use the system. So, there is a lots of threat and Vulnerability. If there occurs any damage then it will be bad for the users even there reputation can be fall. The main risk behind implementing the project is security. If somebody hacks the system then it will be a total disorder. So during development it will be one of our major concerns. Another concern is having common bugs such as the common users having the same functionalities as the user or system administrator.

1.2. Project Delivery

1.2.1 Deliverables

The following contents will be delivered with the project:

- Project Demo

- User manual along with Tutorial
- Documentation

1.2.2 Timescales

The time frame for implementing the project is given in Figure 1.2.1.

1.2.3 Work Distribution

The work distribution of the project is given in Table 1.2.1.

1.2.4 Project Resources

The resources required to finish the project is given in Table 1.2.2.

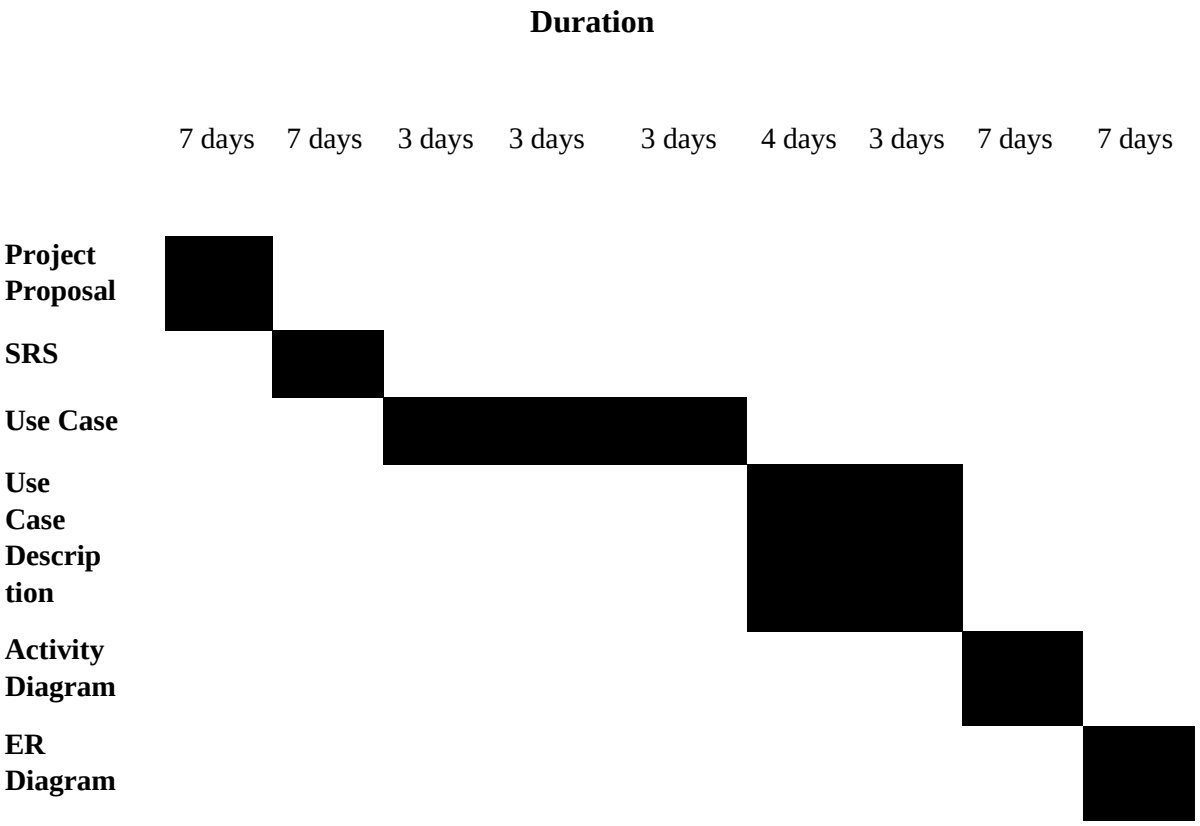


Figure 1.2.1: Time frames for project implementation

Project Proposal	Mst. Marzia Khatun	7days
Software Requirement Specification	Md. Saidur Rahman	7 days
Use Case	Md. Saidur Rahman	9 days
Use Case Description	Mst. Marzia Khatun	7 days
Activity Diagram	Md. Saidur Rahman	7 days
ER Diagram	Mst. Marzia Khatun	7 days

Table 1.2.1 Work Distribution

Hardware Requirements		
Processor	**RAM**	**Hard Disk Space**
CORE i5	8gb or higher	512 Mb or higher
Software Requirements		
Operating System		**Database**
For user no specific OS is required. The server machine must have Windows XP/Vista/10 along with .NET framework 4 and IIS.		SQL Server 2014

Table 1.2.2 Project Resources

1.3. Summary

Time is our most valuable asset. We cannot waste it when there is scope of utilizing it in a better way. Our proposed system named OAS will be developed to meet this purpose. It will save the time of the mass people wasted during the time of admission. Along with this, OAS will also help the administrator to see the previous records of a student. As a result s/he can take decisions more wisely. If the student loses his/her information s/he can retrieve it from the website. We hope that OAS is useful for student and administrator also. People can fill up admission form by staying at home. University can access easily all records. Our whole nation will be beneficial specially rural people will get benefit from this system.

1.4. References

1. How To: Write a project proposal [Online] URL:

https://www.academia.edu/10126899/Proposal_on_Online_Admission_Form

2. **https://elearn.daffodilvarsity.edu.bd/course/view.php?id=11233**

2.SRS→

Functional Requirements

FR01	Registration & Login
Description	After entering the url in any internet browser, in the software index page the user must have to register himself first. After every time of access he has to log-in first before entering the main software. All user mandatory for login. Only authorized users can login the software.
Stakeholder	Administrator, Active User, Inactive User

FR02	View Login Page
Description	All users can see the login page and enter the url or internet browser.
Stakeholder	Administrator, Active User, Inactive User

FR03	Search Information
Description	After log-in, users can search the details of the admission in the system.
Stakeholder	Administrator, Active User, Inactive User

FR04	**Fill-up Admission Form**
Description	After log-in and registration, users can fill admission forms in the system. He can fill more than one admission form.
Stakeholder	Administrator, Active User

FR05	**Attach Supporting Document**
Description	User can upload photo and Signature as attach supporting documents in admission form.
Stakeholder	Active User

FR06	**View Filled Form**
Description	After filling the admission form, users can view their filled form.
Stakeholder	Administrator, Active User

FR07	**Re-check Filled Form Records**
Description	Before the last date of application users can re check their records and also edit it.
Stakeholder	Active User

FR08	**Payment**
Description	After filled application form, user must do payment in bkash or any bank account
Stakeholder	Active User

FR09	**Edit & Delete Records**
Description	When the application date has expired, the administrator can delete and edit records in this system.
Stakeholder	Administrator

FR10	**Send a Confirmation Letter**
Description	When users' information will be alright, the administrator will give a confirmation letter to them.
Stakeholder	Administrator, Active User

FR11	**Print Application Form**
Description	After filled application form, administrator and active use can print their form.
Stakeholder	Administrator, Active User

Non-Functional Requirements

1. Active and inactive user can see web page without login and registration.
2. It can be in both languages English and Bengali.
3. Easy to use.
4. If user forget there password there should be a option to forget password.
5. Online payment system should be added.
6. Member searching option.
7. There should be a option to verify digital documents.
8. Even after login, the applicant has access to check above mentioned details.
9. There should be a option to do After login, the applicant can check the application status.
10. Applicants can log in multiple times and resume from where she/he has left.
11. Having multiple online payment options such as net banking, debit card, credit card, wallet, mobile banking etc.

Group Name: Sunrise

Topic Name: Online Admission System

Member Name: Mst. Marzia Khatun
ID: 193-35-2936
Member Name: Md Saidur Rahman
ID: 193-35-2919

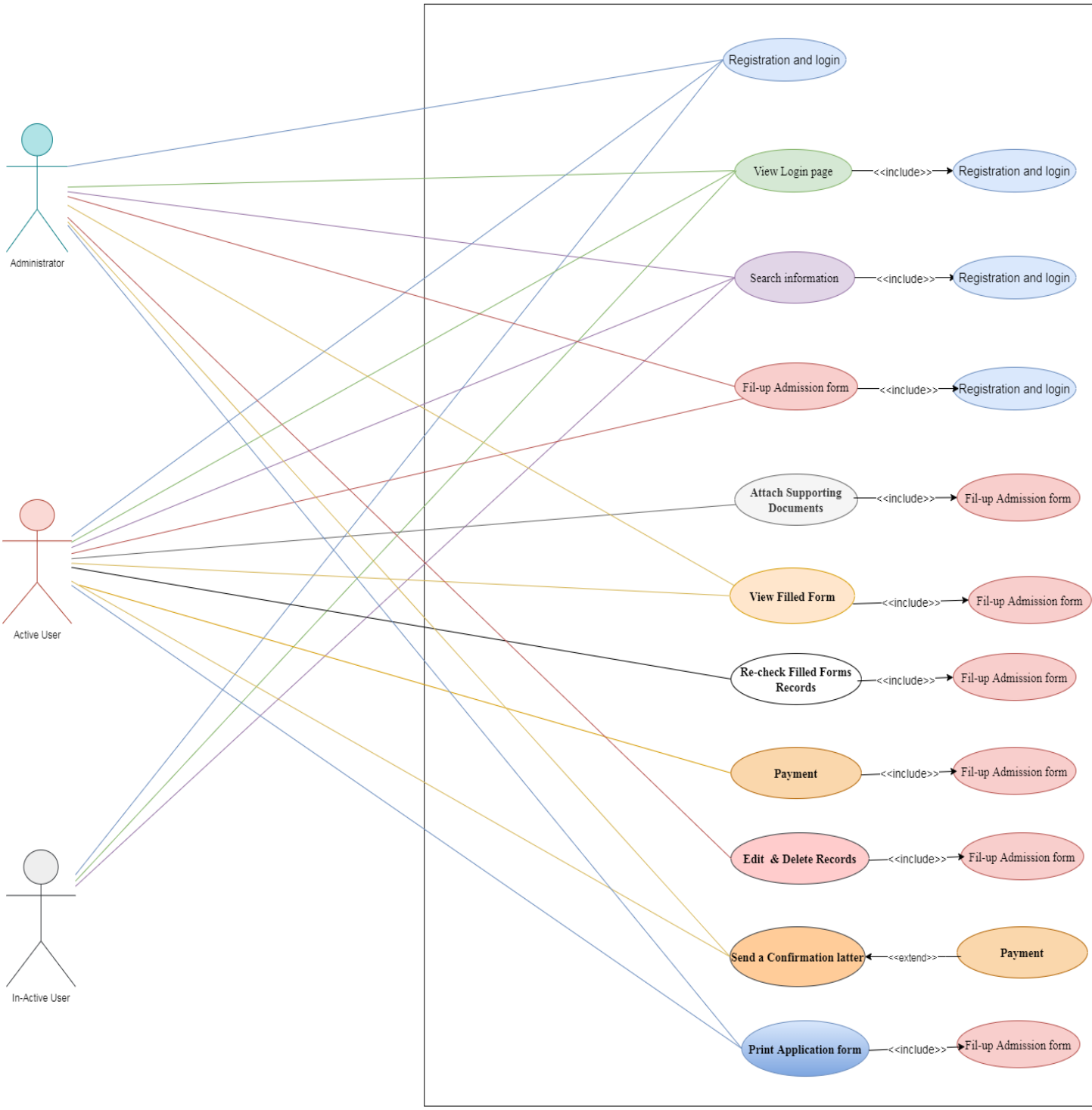


Figure: Use Case Diagram

USE CASE DESCRIPTION:-

4.

Use Case	**Registration and login**	
Goal	For full-fill the admission form user should be registered then need to login.	
Preconditions	Before login user should be registered	
Success End Condition	User can complete their registration User can login in the page	
Failed End Condition	Without registration user can't login in that system.	
Primary Actors: **Secondary Actors:**	Administrator, Active user, Inactive user.	
Trigger	Request for login.	
Description / Main Success Scenario	**Step**	**Action**
	1	User can do their registration
	2	User can login.
	3	User can see their information
	4	User can change their information
Alternative Flows	**Step**	**Branching Action**
	1a	User can visit without registration
	2a	User can't login without Registration
	3a	User can't see their information.
	4a	User can't change their information without login

Quality Requirements	Step	Requirement
	1	User should be login in 90 seconds.

Use Case	**View Login Page**	
Goal	After registration, the user can see the login page.	
Preconditions	Users Should be Registered first.	
Success End Condition	User can view login page	
Failed End Condition	User can’t login in the system	
Primary Actors: **Secondary Actors:**	Administrator, Active User In-Active User	
Trigger	Request for login	
Description / Main Success Scenario	**Ste p**	**Action**
	1	User can view the login page.
	2	User can login in that page.
Alternative Flows	**Ste p**	**Branching Action**
	2	Without registration user can not login in this page.
Quality Requirements	**Ste p**	**Requirement**
	4	User should be login within 30 seconds

Use Case	**Search Information**	
Goal	To search any information	
Preconditions	Should be registered and logged in.	
Success End Condition	Can find out any information.	
Failed End Condition	Can not find information in a very low time.	
Primary Actors: **Secondary Actors:**	Administrator, Active User, In-active user.	
Trigger	Request to search information.	
Description / Main Success Scenario	**Step**	**Action**
	1	User can search any information
	2	Users can use information.
	3	Users can update information.
Alternative Flows	**Step**	**Branching Action**
	1a	User can not find out any information within a short time.
	3a	User can not upgrade their data.
Quality Requirements	**Step**	**Requirement**
	1	User should do request within 30 seconds

Use Case	**Fill up Admission Form.**	
Goal	Full-fill admission form in online.	
Preconditions	Users should be registered and should be logged in that system.	
Success End Condition	User can fill up their admission form.	
Failed End Condition	User can not fill up their admission form	
Primary Actors: **Secondary Actors:**	Administrator, Active User.	
Trigger	Request to fill up admission form..	
Description / Main Success Scenario	**Step**	**Action**
	1	Students can fill up their admission form by staying anywhere.
	2	Students can Upgrade their information.
Alternative Flows	**Step**	**Branching Action**
	1	User can not fill up their admission form by staying anywhere..
Quality Requirements	**Step**	**Requirement**
	1	User should confirm within 60 seconds

Use Case	**Attach Supporting Details**	
Goal	Adding supporting information.	
Preconditions	Users should register first. Users should logged in system,	
Success End Condition	Users can add their supporting file.	
Failed End Condition	Users can not complete their admission form.	
Primary Actors: **Secondary Actors:**	Active User	
Trigger	Request to add others details.	
Description / Main Success Scenario	**Step**	**Action**
	1	User can attach their supporting Details.
	2	User can add others details.
Alternative Flows	**Step**	**Branching Action**
	1a	User cant complete their admission .
Quality Requirements	**Step**	**Requirement**
	4	User should done it within 30 seconds

Use Case	**View Filled Form**	
Goal	Active user's issues request directly to the system for view their filled application form.	
Preconditions	User's must do fill up admission form.	
Success End Condition	Active user can able see their form.	
Failed End Condition	Active user can't able to see their form.	
Primary Actors: **Secondary Actors:**	Active user, Administrator	
Trigger	Requesting for view form	
Description / Main Success Scenario	**Step**	**Action**
	1	User will request to system for see their application form.
	2	System captures user's name, address and other information.
	3	System automatically check users information
	4	System will give permission to users.
	5	Then user can able to see their form.
Alternative Flows	**Step**	**Branching Action**
	5a	User cannot able to see their form.
	5b	Again they will request to system.
Quality Requirements	**Step**	**Requirement**
	1	User can stay this system within 60 seconds
	2	The Invoice pop up message will stay no later than 90 seconds after it is sent by the System

Use Case	**Re-check filled form records**	
Goal	Active users issues request directly to the system for Re-check their filled application form.	
Preconditions	User's must do fill up admission form.	
Success End Condition	Active user can able re-check their form.	
Failed End Condition	Active user can't able to re-check their form.	
Primary Actors: **Secondary Actors:**	Active User Administrator	
Trigger	Request to system for re-check admission form.	
Description / Main Success Scenario	**Step**	**Action**
	1	User will request to system for re-check their application form.
	2	System captures user's name, address and other information.
	3	System automatically check users information
	4	System will give permission to users.
Alternative Flows	**Step**	**Branching Action**
	1a	User cannot able to re-check their admission application form.
	1b	Again they will request to system.
Quality Requirements	**Step**	**Requirement**
	1	User can stay this system within 60 seconds
	2	The Invoice pop up message will stay no later than 90 seconds after it is sent by the System

Use Case	**Payment**	
Goal	User's issues request directly to the system for payment.	
Preconditions	Student must do filled application form.	
Success End Condition	User's can payment successfully.	
Failed End Condition	User's can't payment successfully.	
Primary Actors: **Secondary Actors:**	Active User. Credit card company, bank, shipping service	
Trigger	Request to system for payment.	
Description / Main Success Scenario	**Step**	**Action**
	1	User's request to system for payment.
	2	System will check all valid information.
	3	System give permission to user's for payment.
	4	After payment, system will give a text message.
Alternative Flows	**Step**	**Branching Action**
	1	User can't able to do payment.
	2	He will again request to system for payment.
Quality Requirements	**Step**	**Requirement**
	1	User can stay this system within 60 seconds
	2	The Invoice pop up message will stay no later than 90 seconds after it is sent by the System

Use Case	**Edit & Delete Records**	
Goal	Administrator can do edit and Delete application form.	
Preconditions		
Success End Condition	Administrator can do edit and Delete application form.	
Failed End Condition	Administrator cannot do edit and delete application form.	
Primary Actors: **Secondary Actors:**	Administrator System	
Trigger	Request to system for edit and delete records.	
Description / Main Success Scenario	**Step**	**Action**
	1	Administrator request to the system for edit and delete user's application form.
	2	System automatically will give permission for edit and delete form.
	3	He will check all forms.
	4	He will do save all correct form.
	5	System will give a pop up message.
Alternative Flows	**Step**	**Branching Action**
	1	Administrator is not able to edit and edit form.
	2	Again he will refresh to system.
Quality Requirements	**Step**	**Requirement**
	1	User can stay this system within 60 seconds
	2	The Invoice pop up message will stay no later than 90 seconds after it is sent by the System

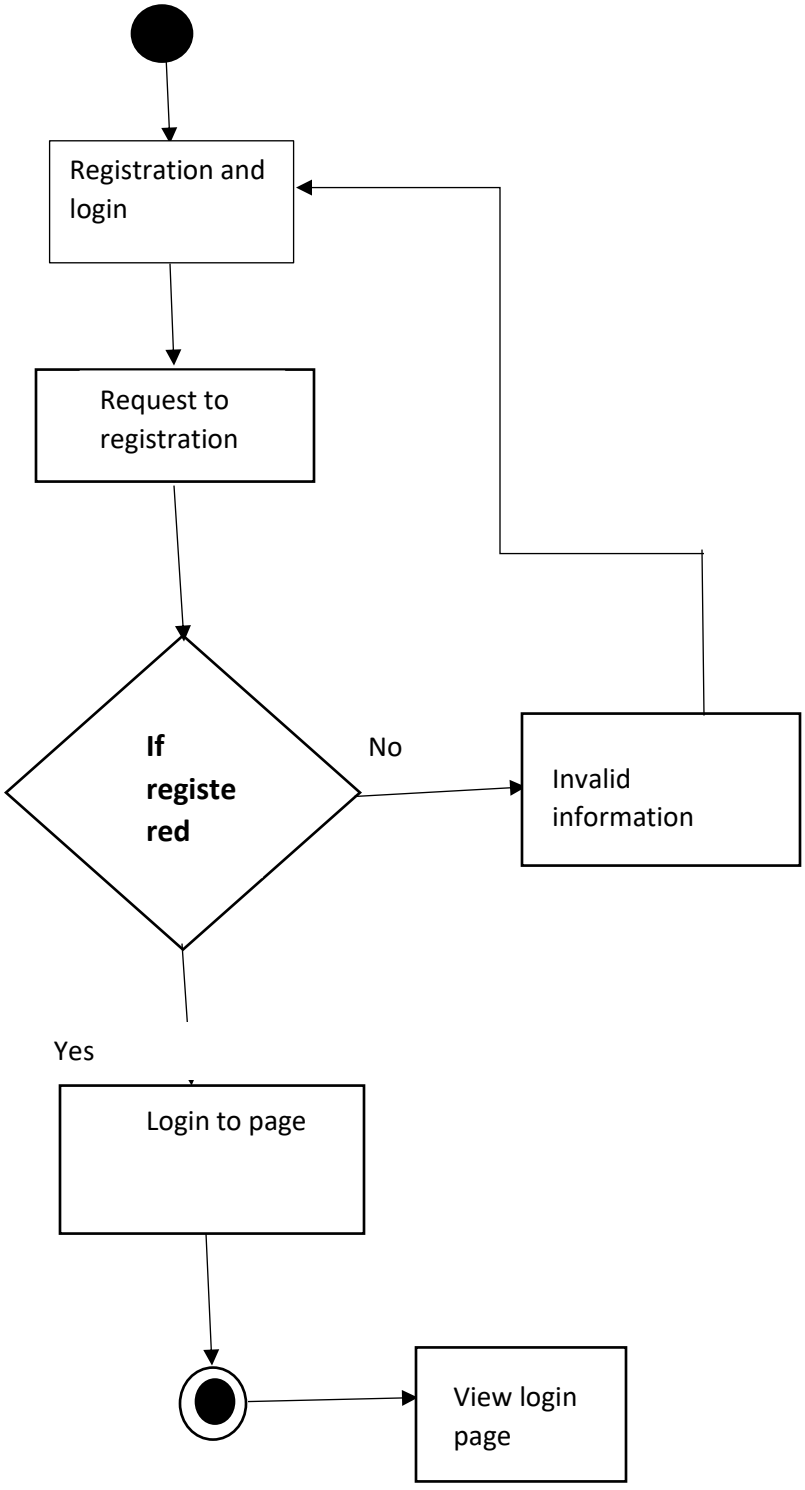
Use Case	**Send a confirmation message**	
Goal	Administrator will give confirmation message to active user.	
Preconditions	User must do application and payment.	
Success End Condition	Active user can see confirmation message.	
Failed End Condition	User don't get confirmation message.	
Primary Actors: **Secondary Actors:**	Active user, Administrator. System	
Trigger	User will give request for message.	
Description / Main Success Scenario	**Step**	**Action**
	1	User will request to system.
	2	Administrator will check all information.
	3	Administrator will give confirmation message to user.
Alternative Flows	**Step**	**Branching Action**
	1	User's do not get any confirmation message.
	2	They will wait minimum 1days.
Quality Requirements	**Step**	**Requirement**
	1	User can stay this system within 60 seconds
	2	The Invoice pop up message will stay no later than 90 seconds after it is sent by the System

Use Case	**Print Application form**	
Goal	After filled application form, administrator and active user can print their form.	
Preconditions	They will do filled application form.	
Success End Condition	Successfully print application form	
Failed End Condition	They cannot print their document.	
Primary Actors: **Secondary Actors:**	Active User, Administrator.	
Trigger	Request for print their document.	
Description / Main Success Scenario	**Step**	**Action**
	1	User do request to the system for print their form
	2	System gives permission to print form.
	2	User can able to print their form.
Alternative Flows	**Step**	**Branching Action**
	1	User cannot print their form
	2	They refresh their browser and again request to system for print their application form.
Quality Requirements	**Step**	**Requirement**
	1	User can stay this system within 60 seconds
	2	The Invoice pop up message will stay no later than 90 seconds after it is sent by the System

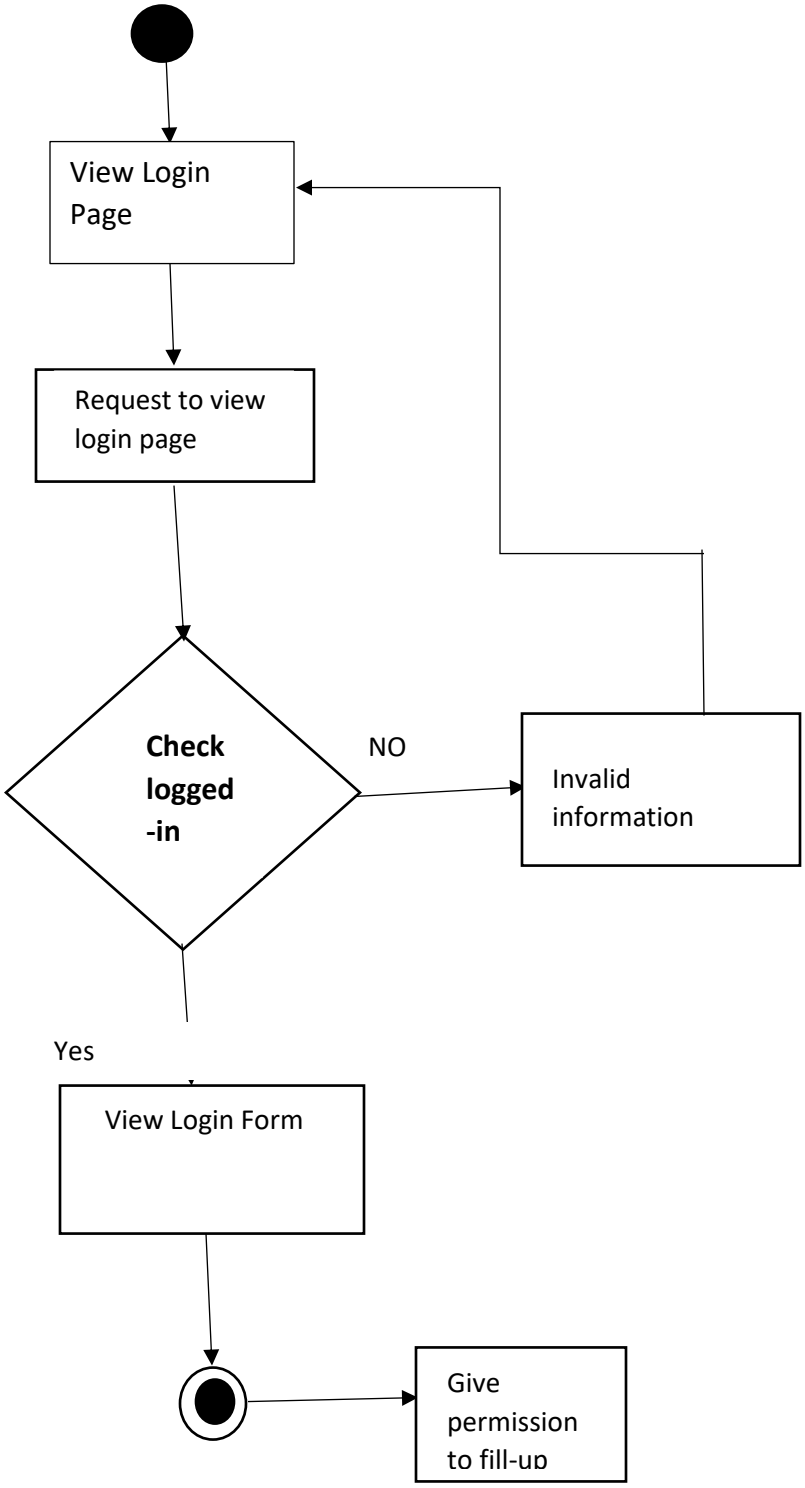
0

5→Activity Diagram…..

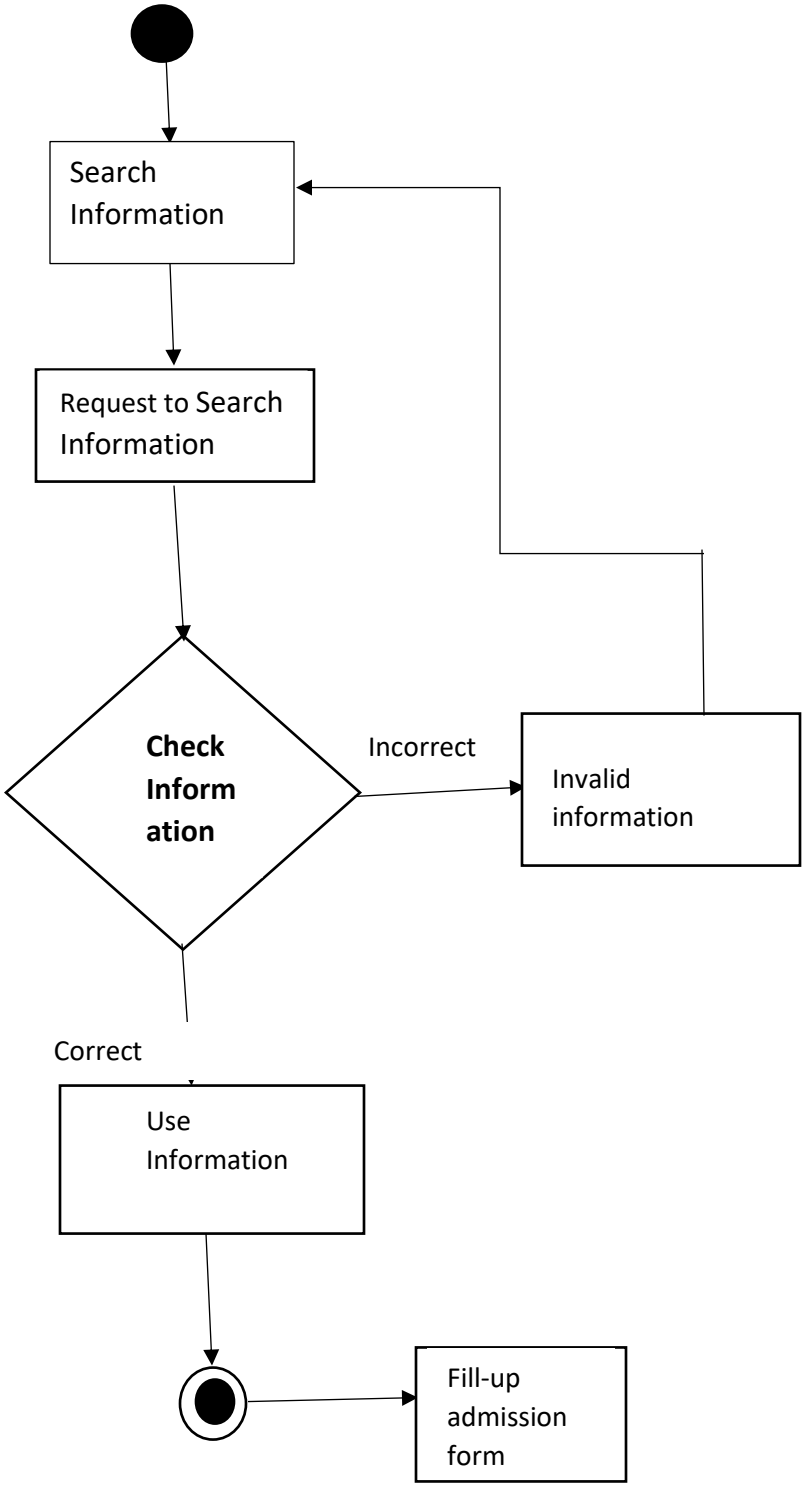
Registration and login



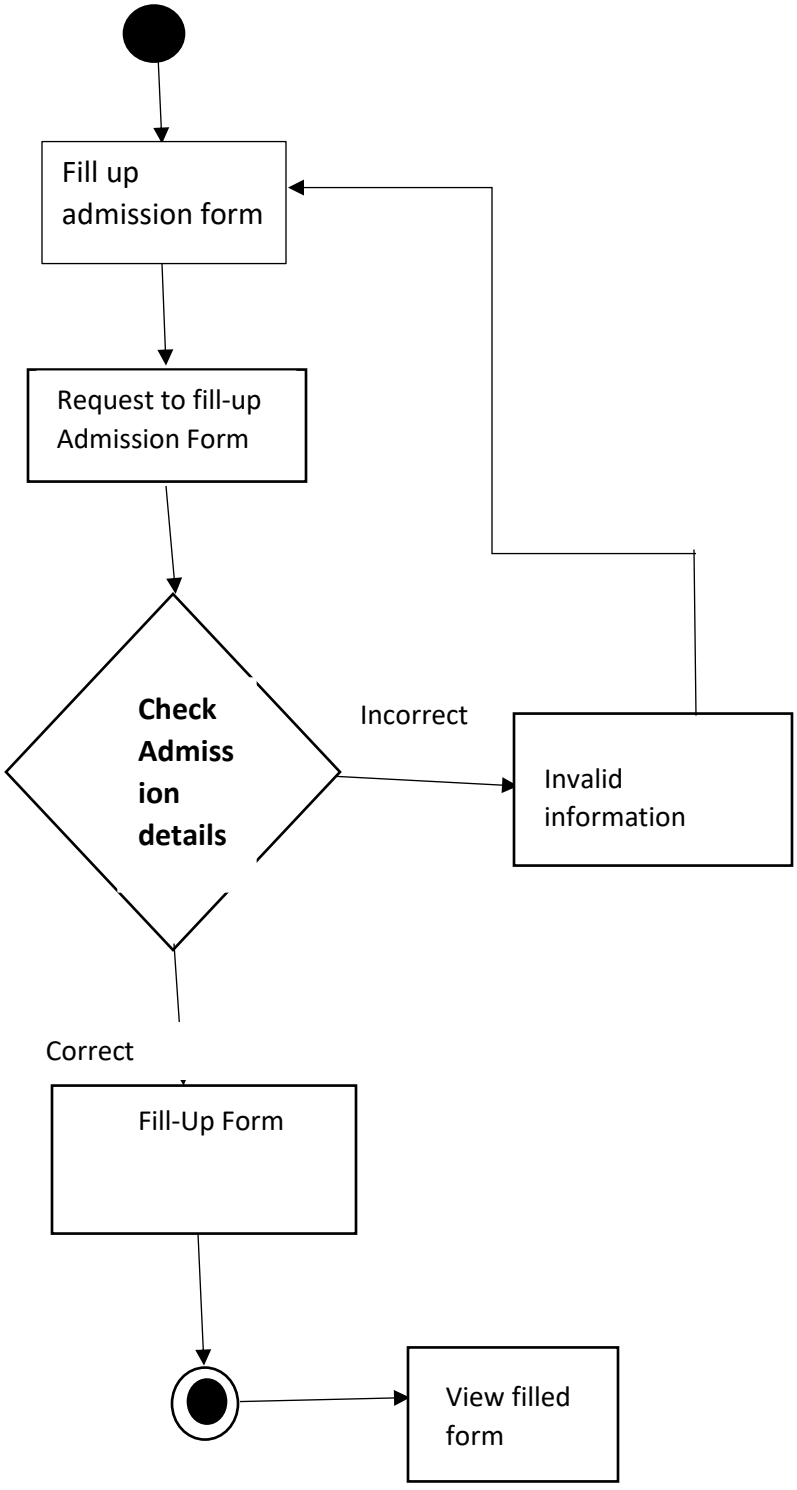
View Login Page



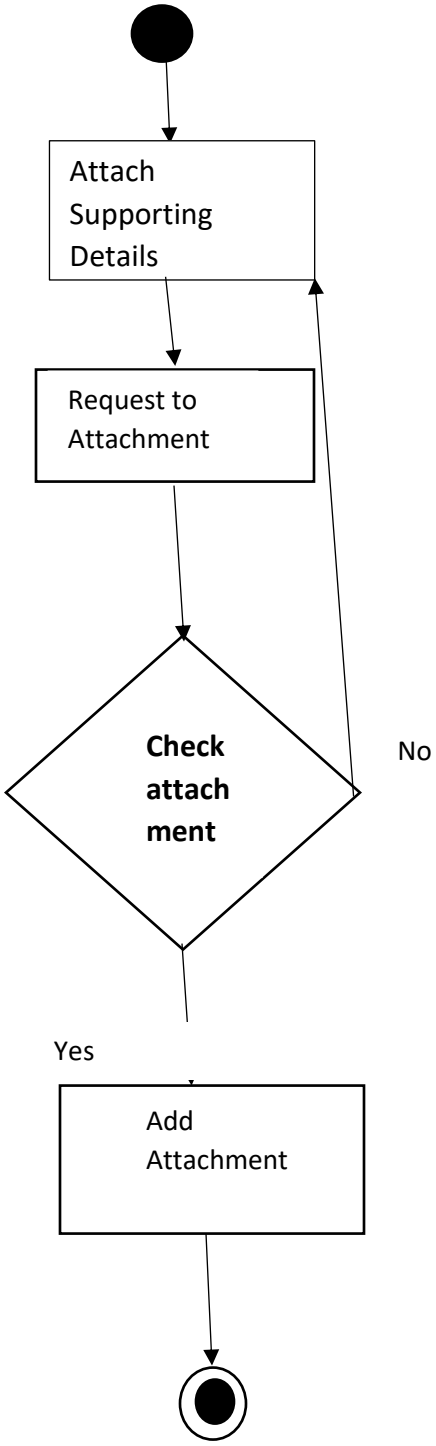
Search Information



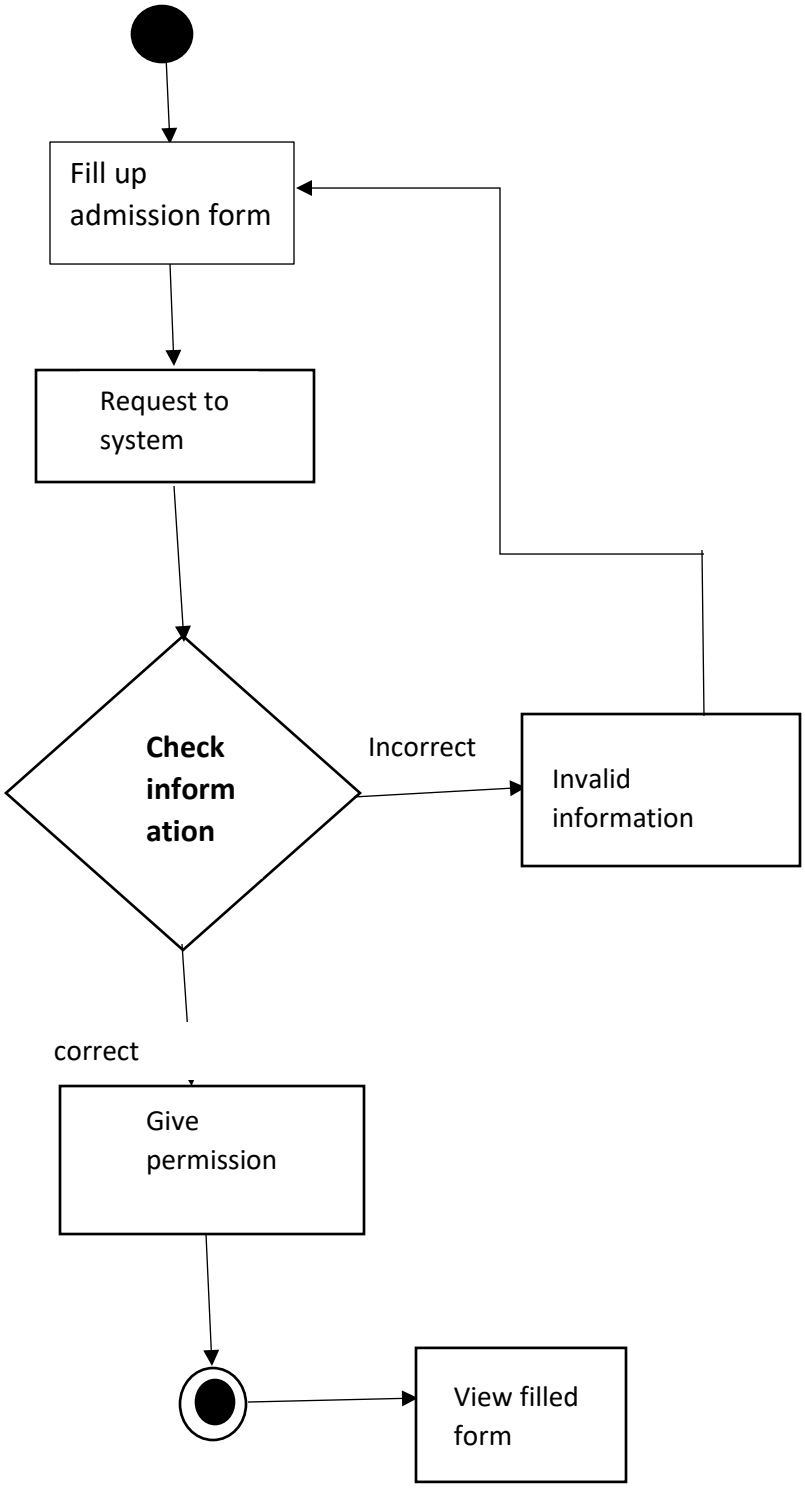
Fill Up Admission Form



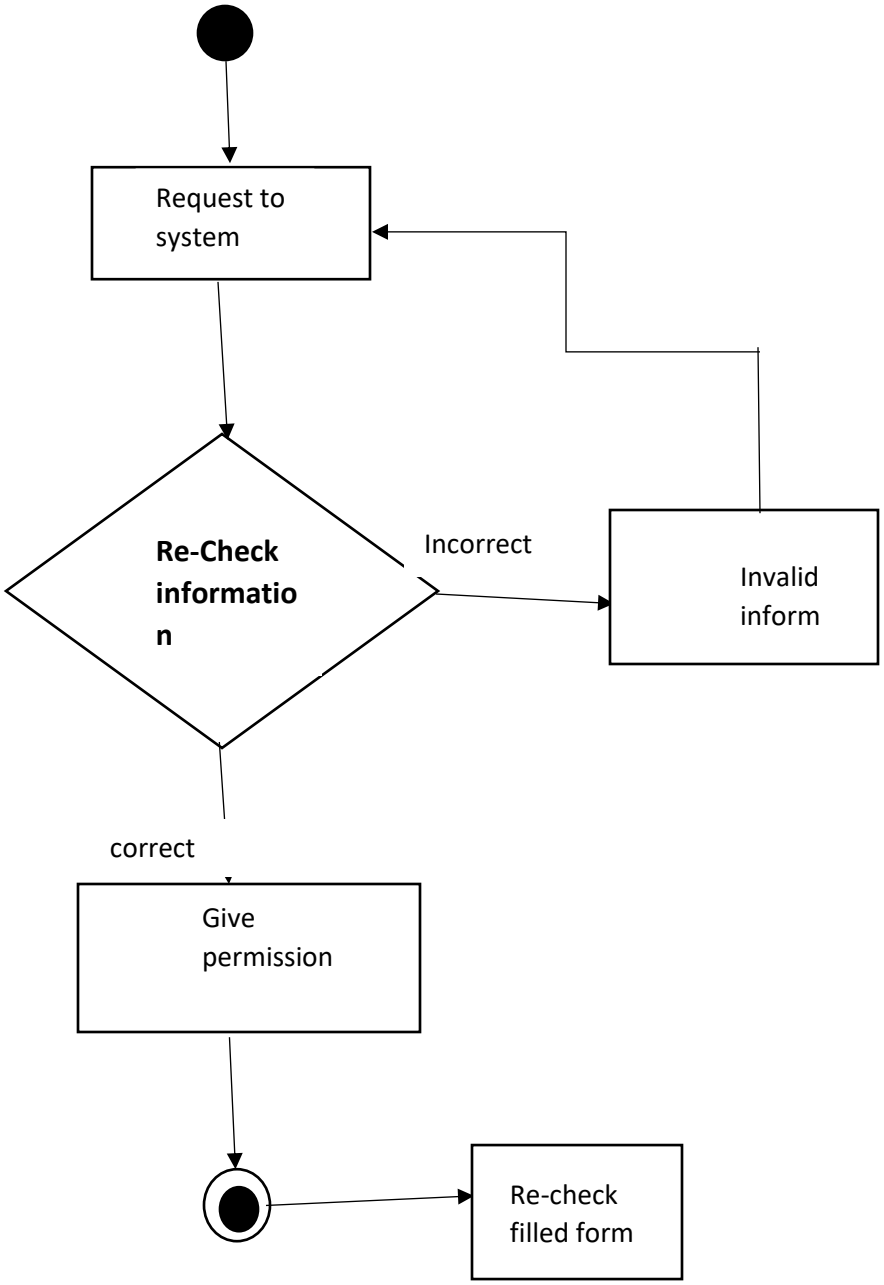
Attach Supporting Details



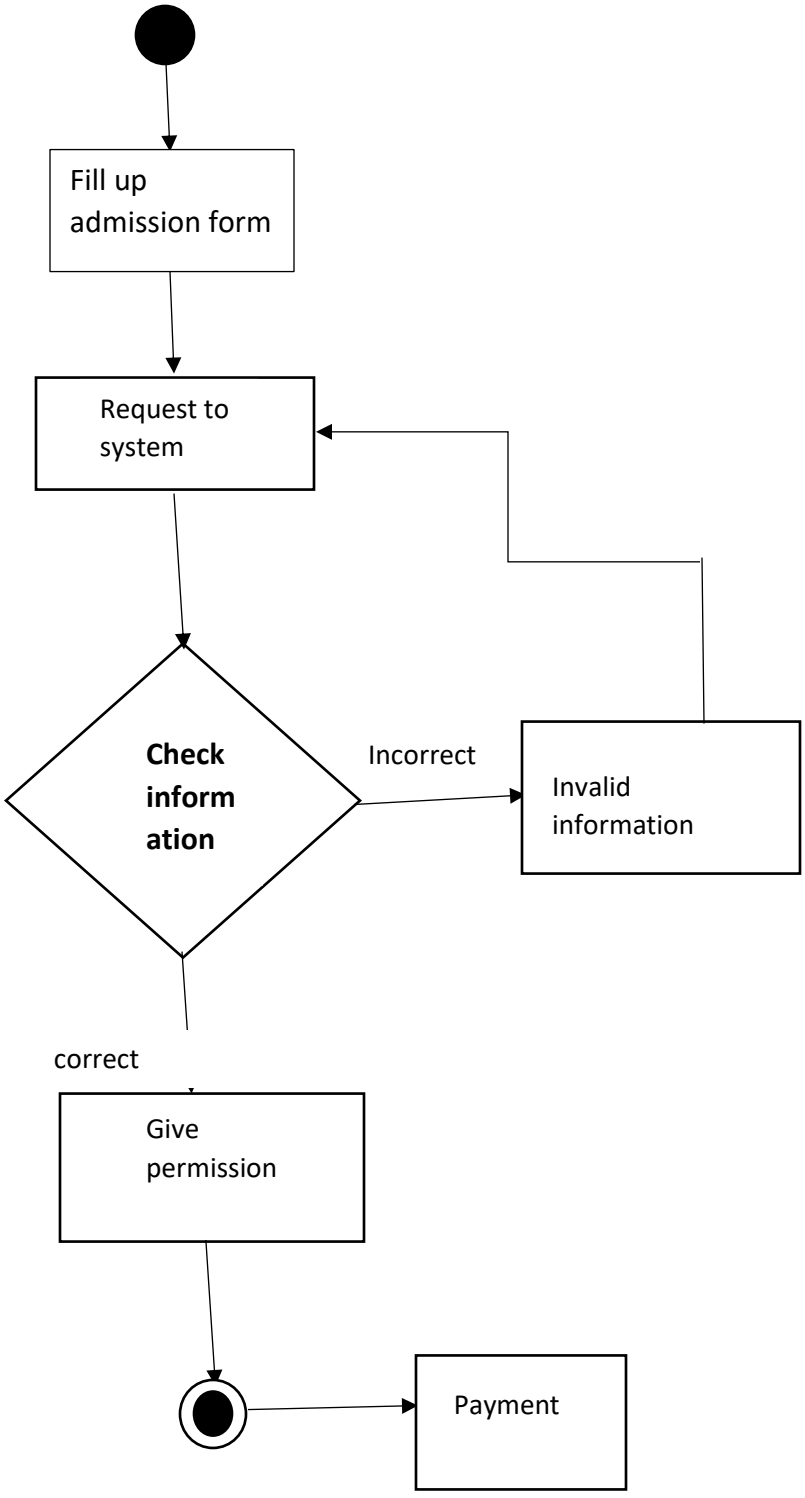
View Filled Form



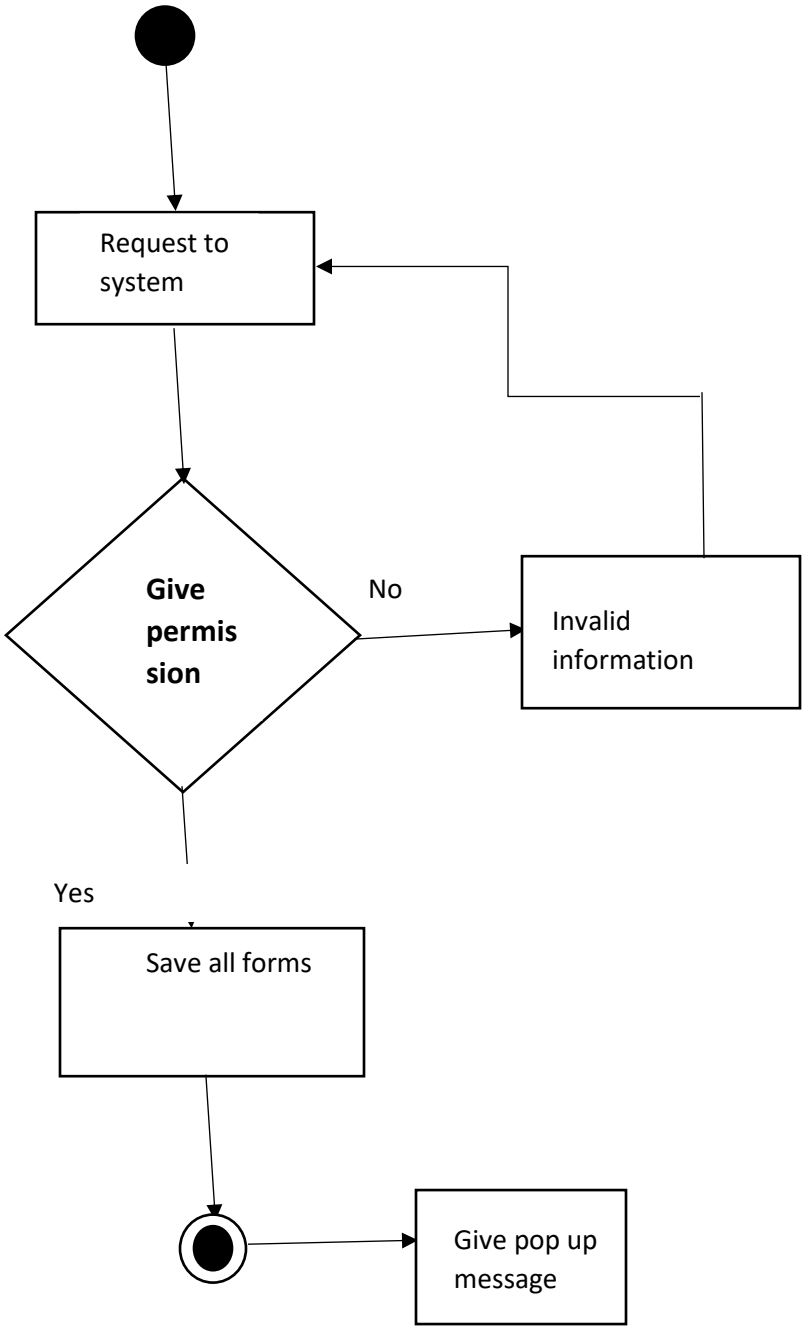
Re-check filled form records



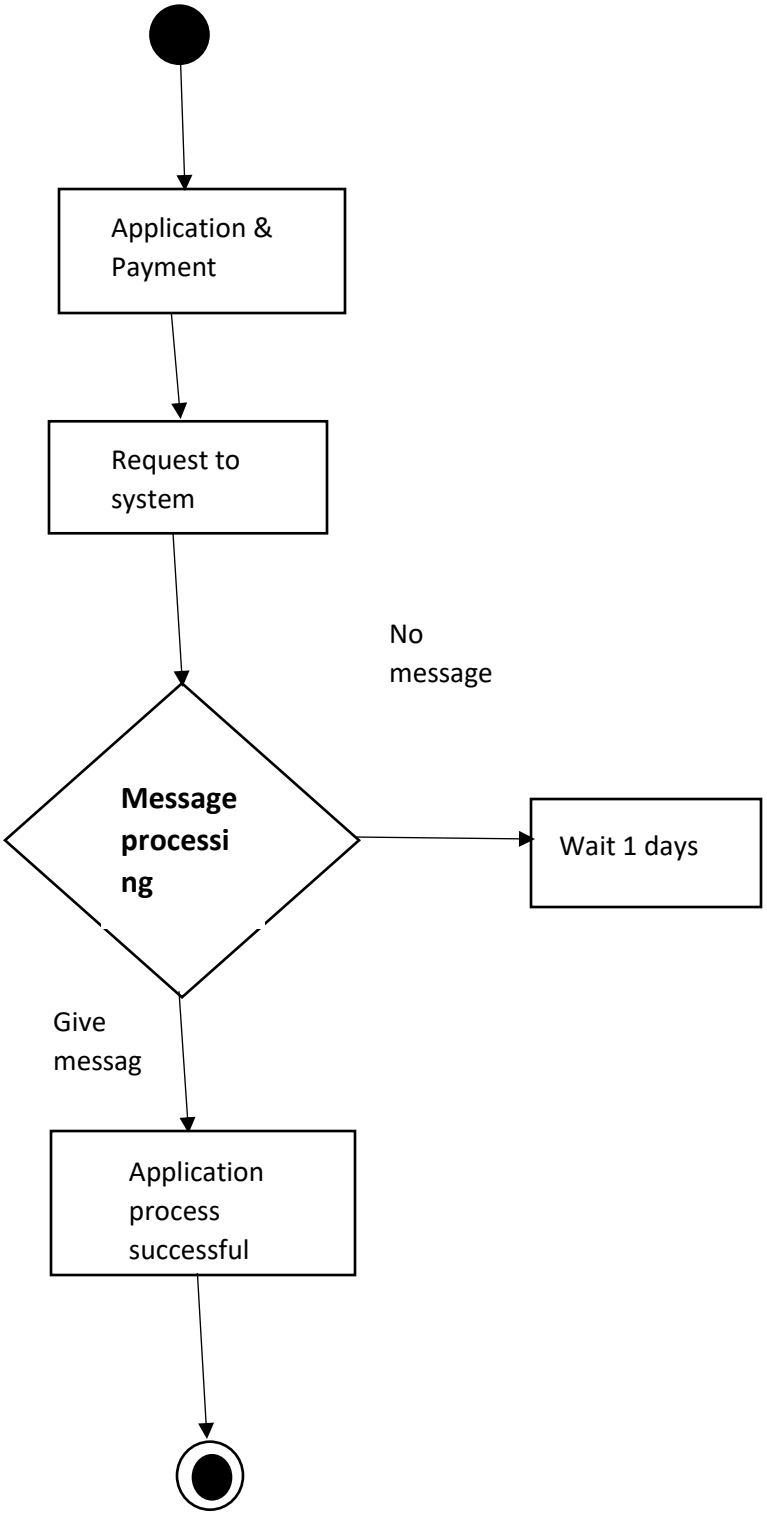
Payment


Fill up admission form
Request to system
Check inform ation
Incorrect
Invalid information
correct
Give permission
Payment

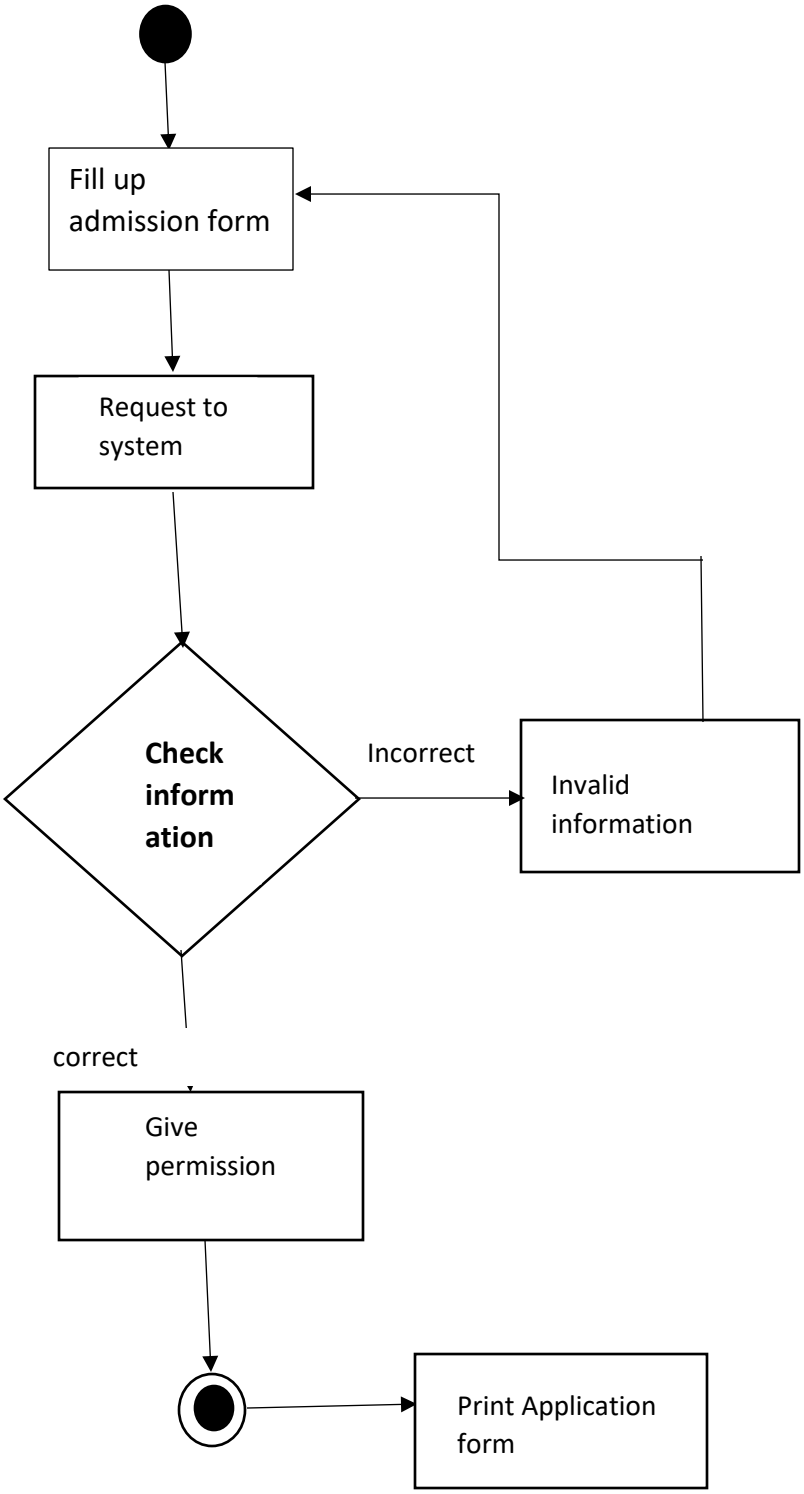
Edit & Delete Records



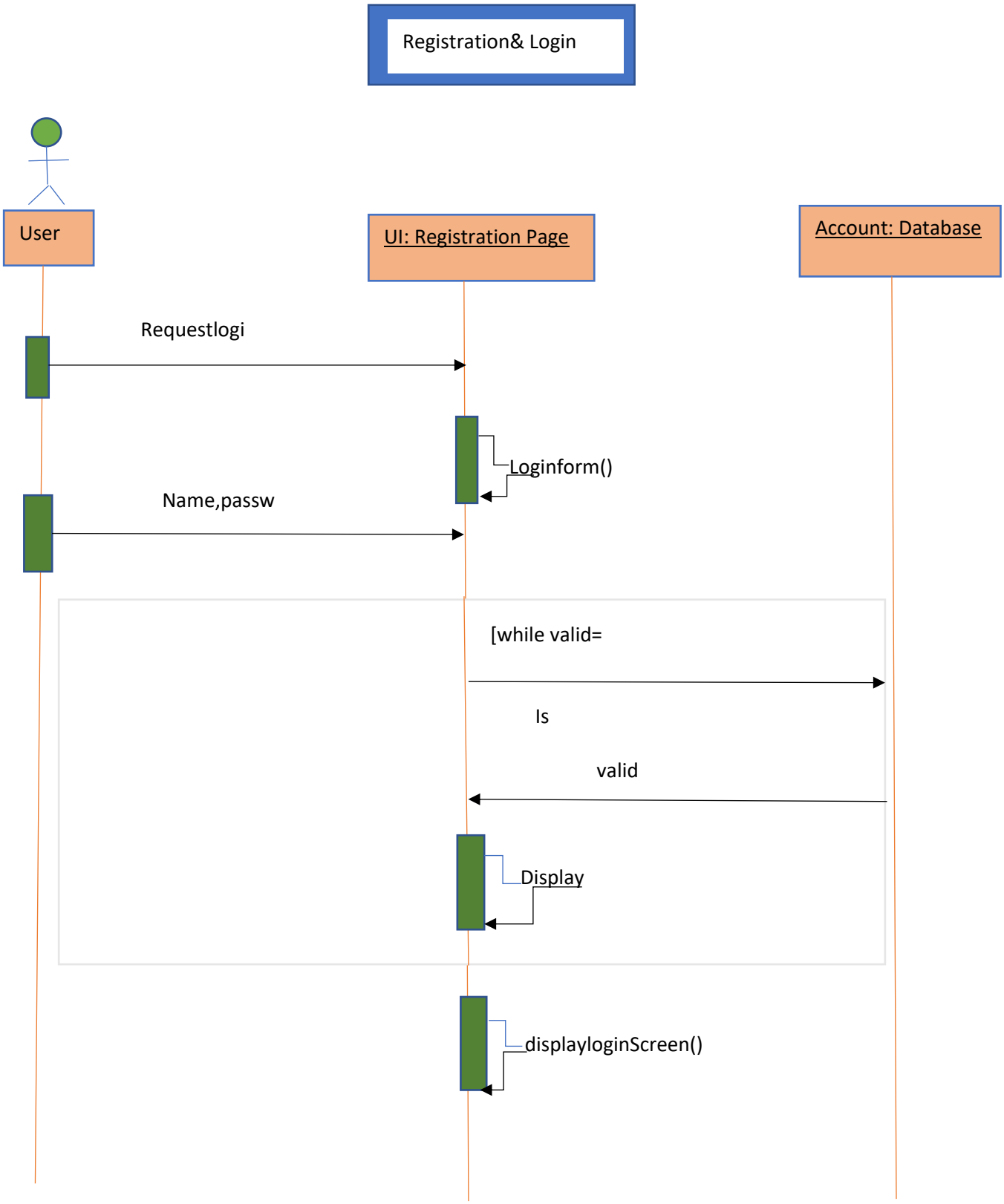
Send a confirmation message

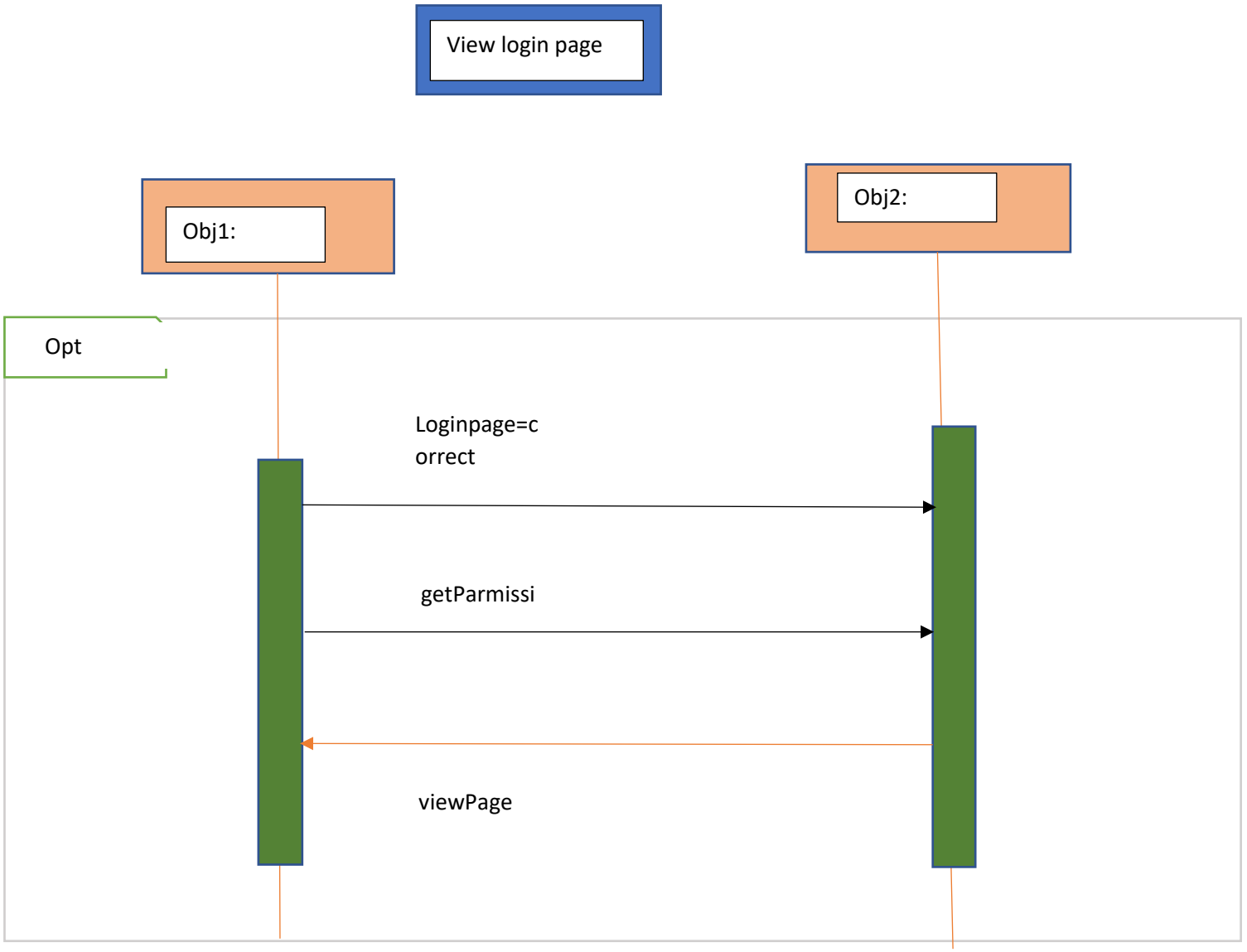


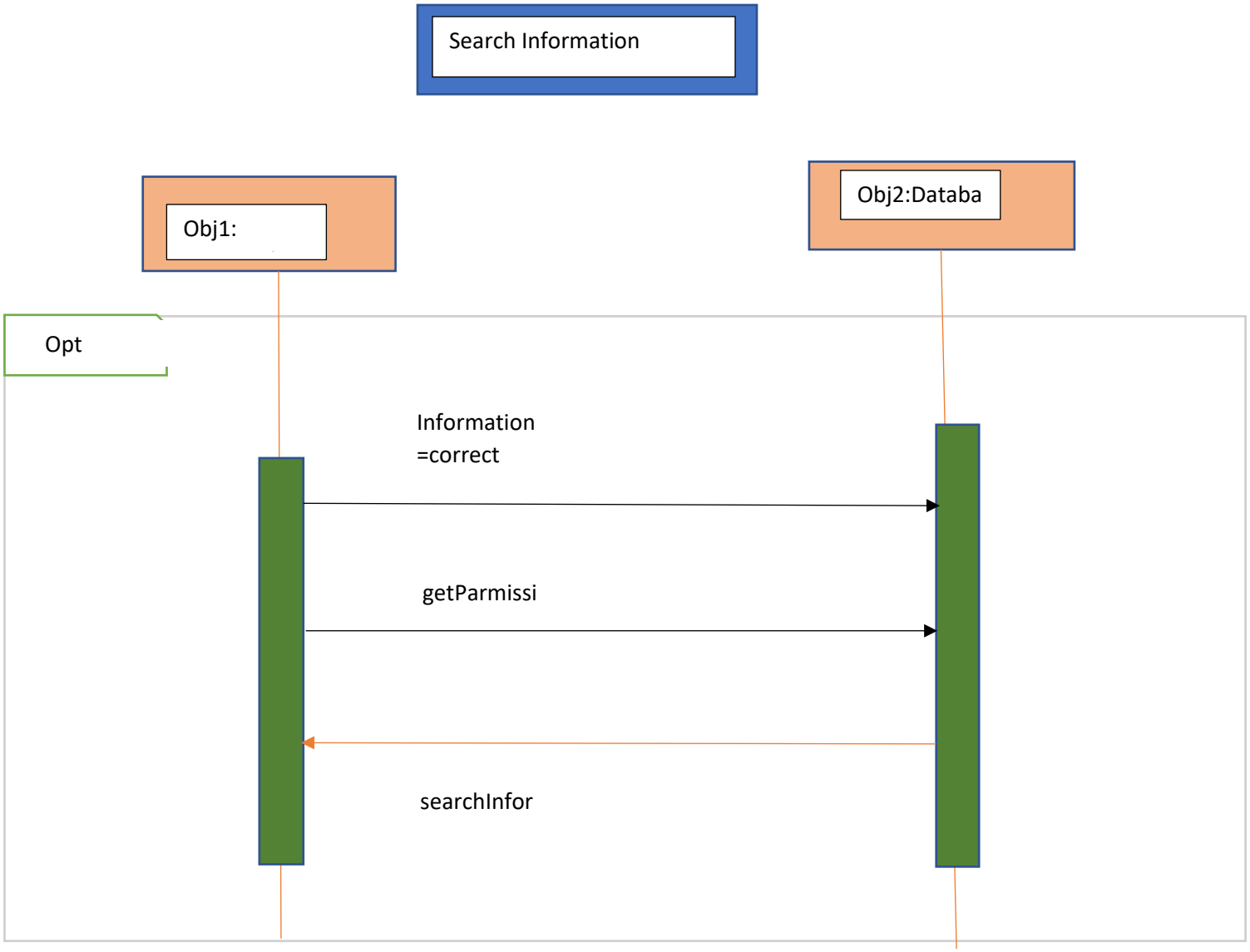
Print Application form

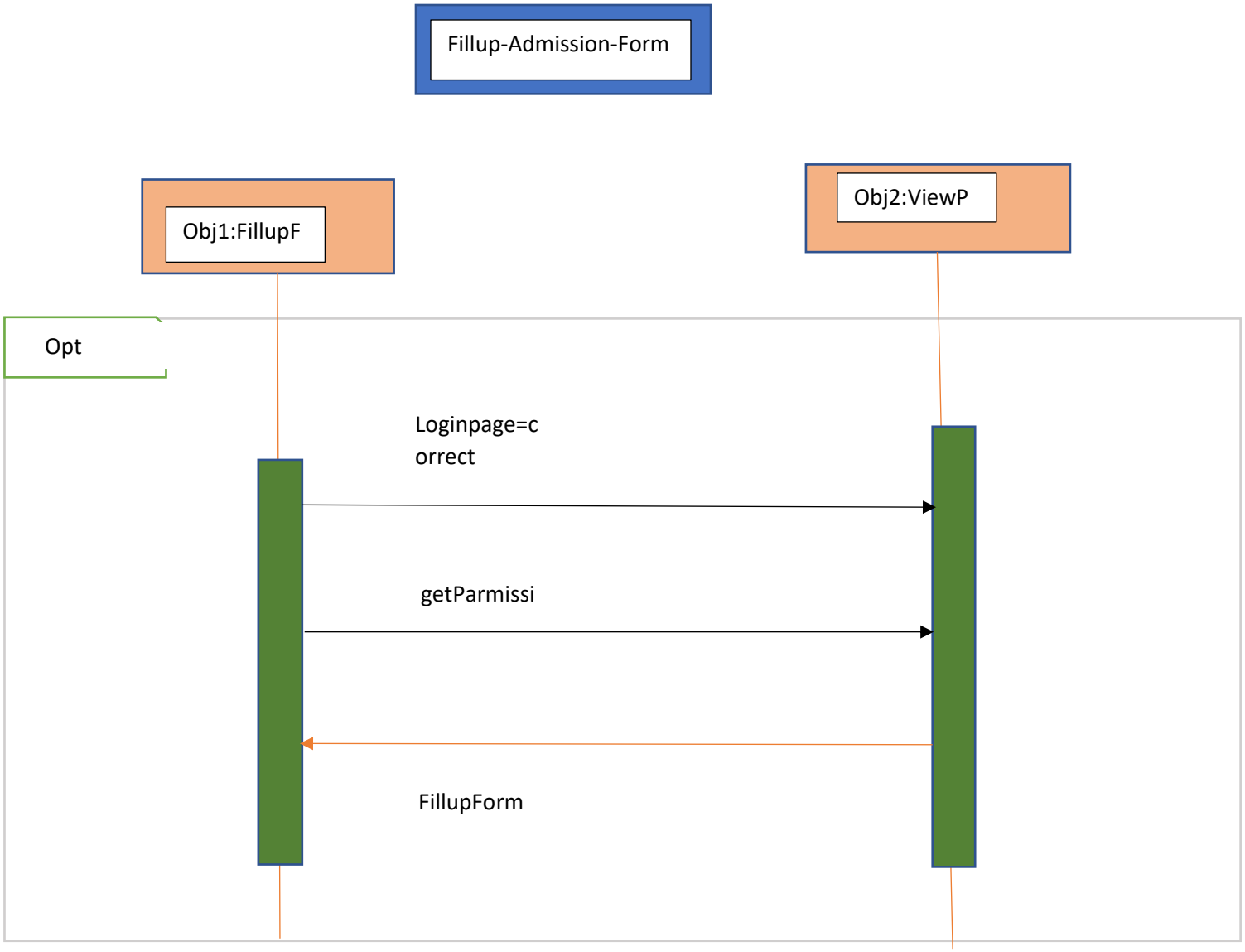


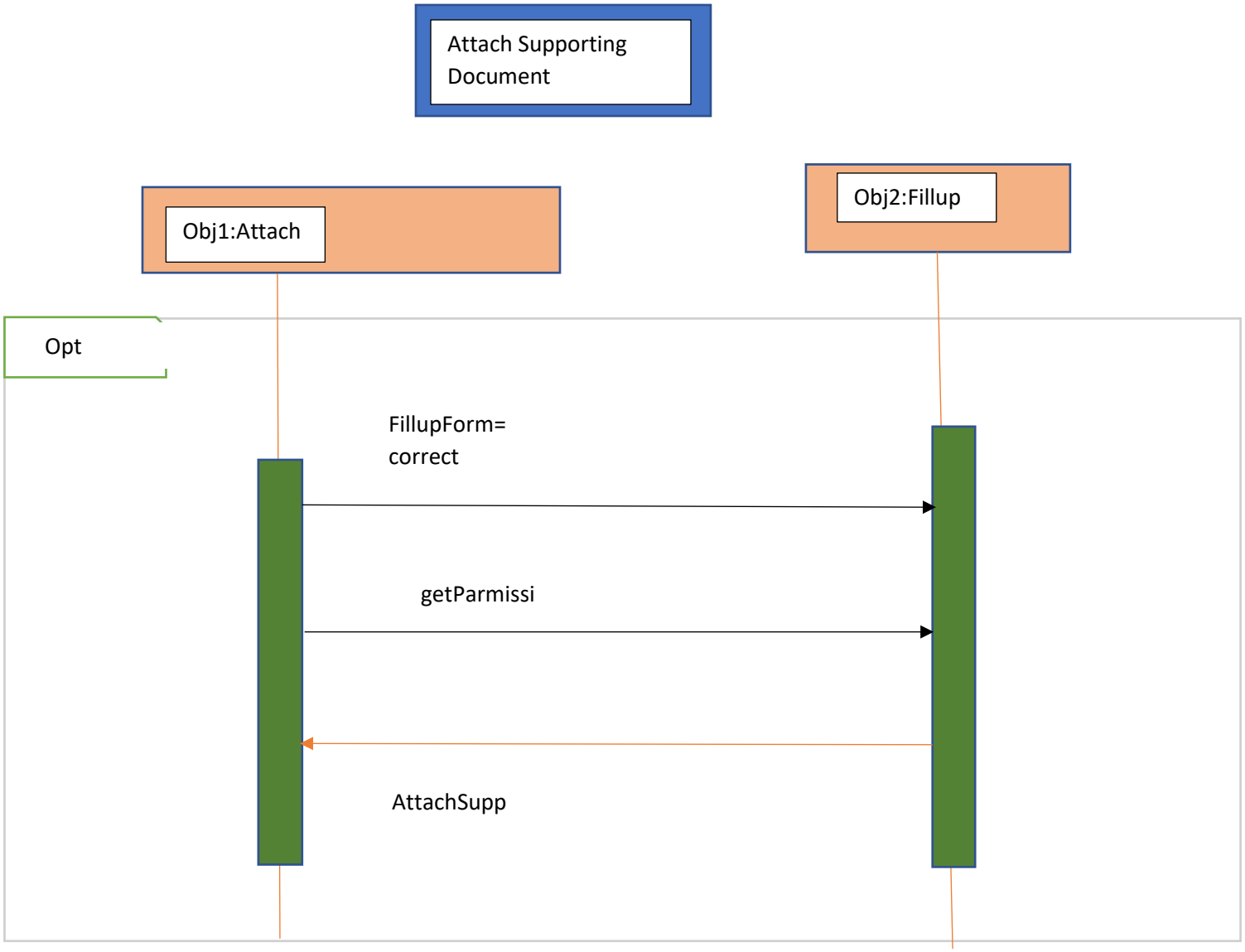
6→Sequence Diagram…..

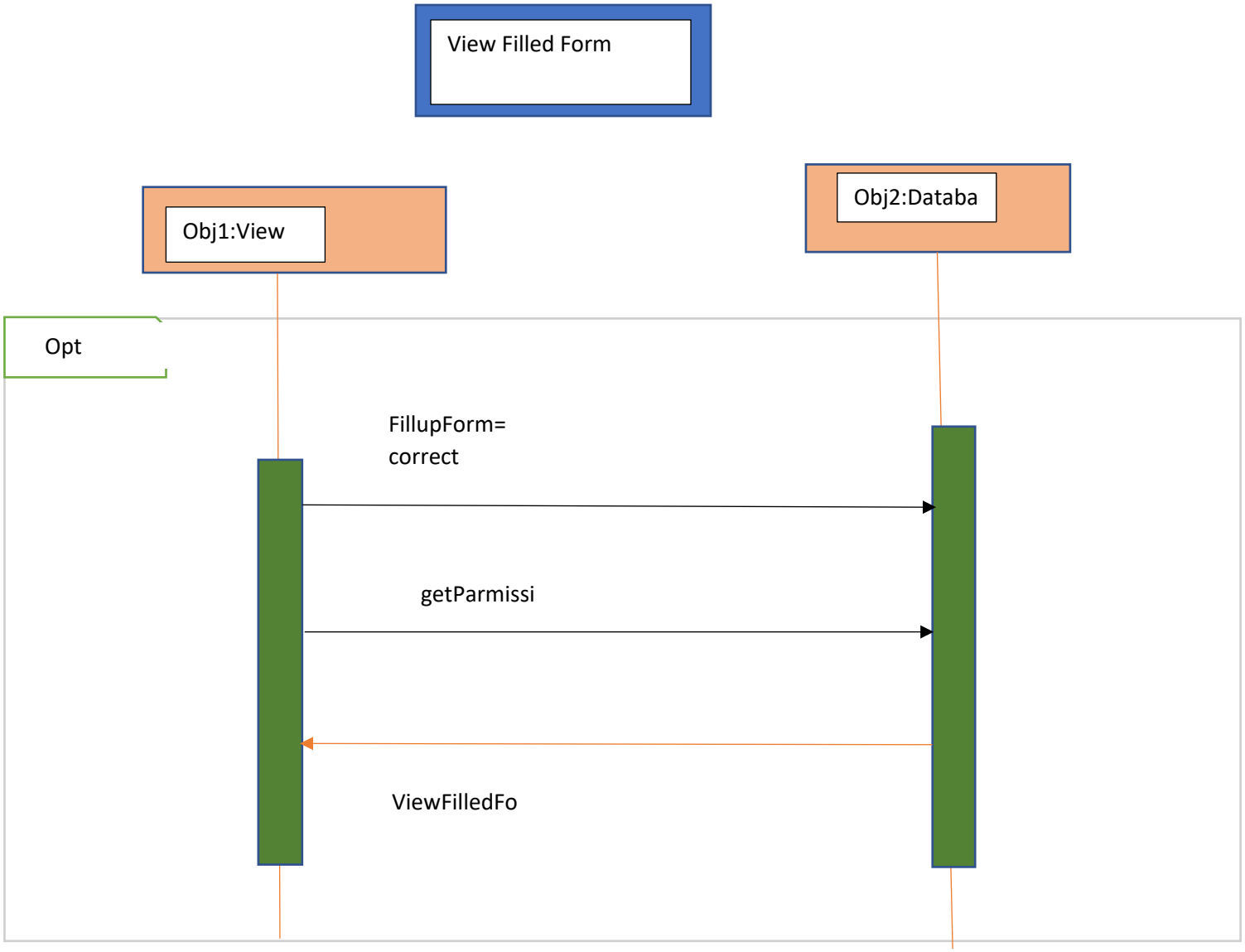

Registration& Login
User
UI: Registration Page
Account: Database
Requestlogi
Loginform()
Name,passw
[while valid=
Is
valid
Display
displayloginScreen()

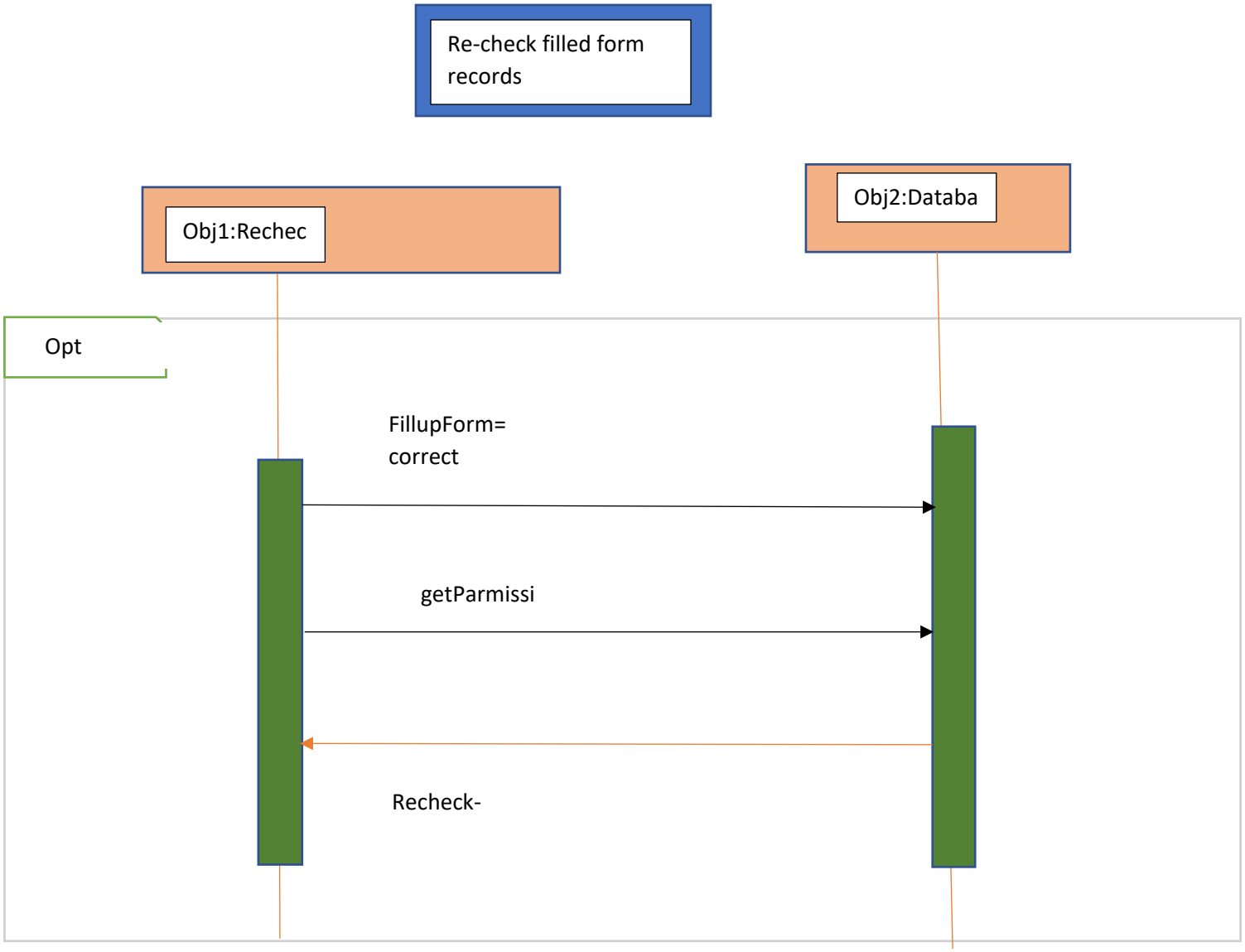

View login page
Obj1:
Obj2:
Opt
Loginpage=c
orrect
getParmissi
viewPage

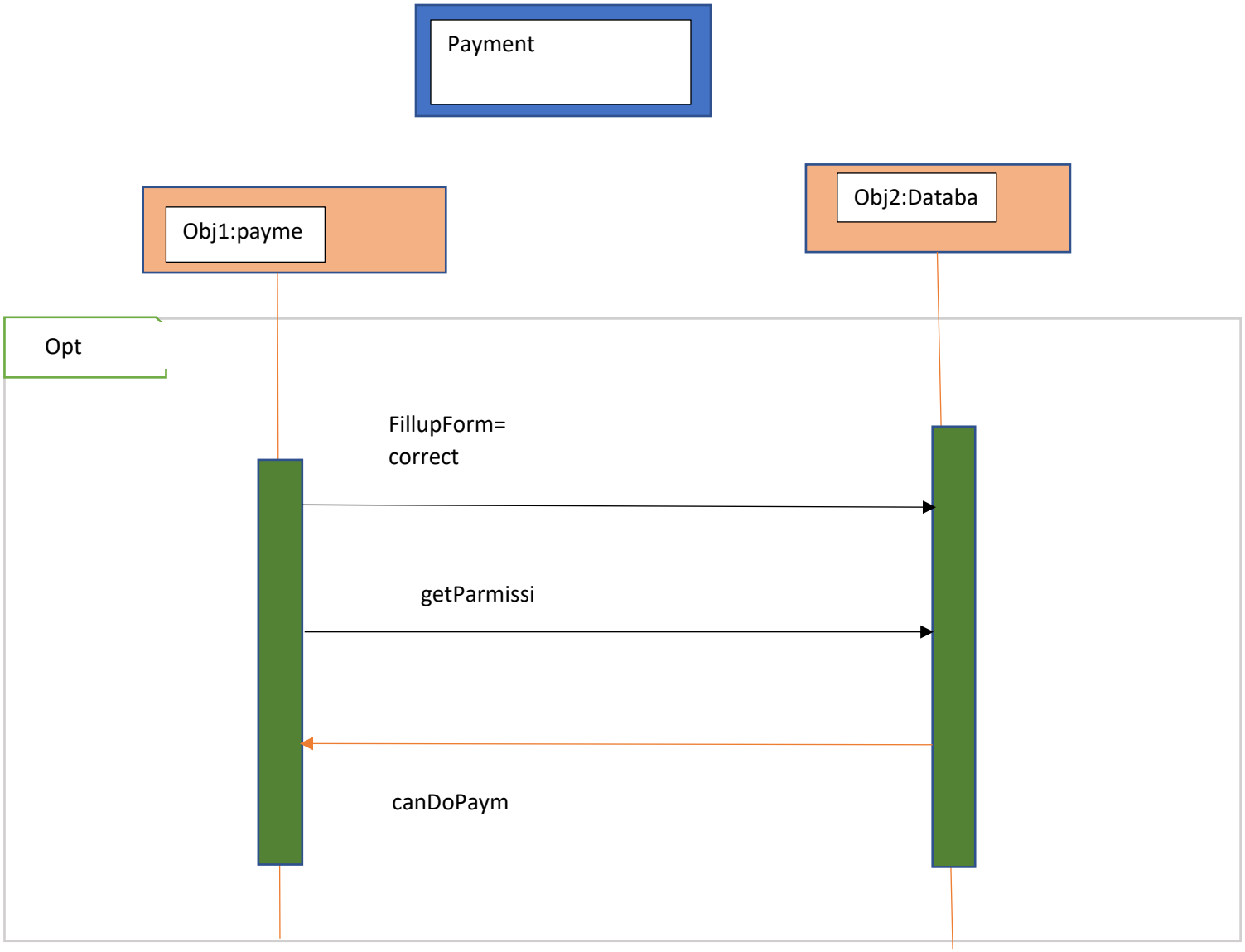

Search Information
Obj1:
Obj2:Databa
Opt
Information
=correct
getParmissi
searchInfor


Fillup-Admission-Form
Obj1:FillupF
Obj2:ViewP
Opt
Loginpage=c
orrect
getParmissi
FillupForm

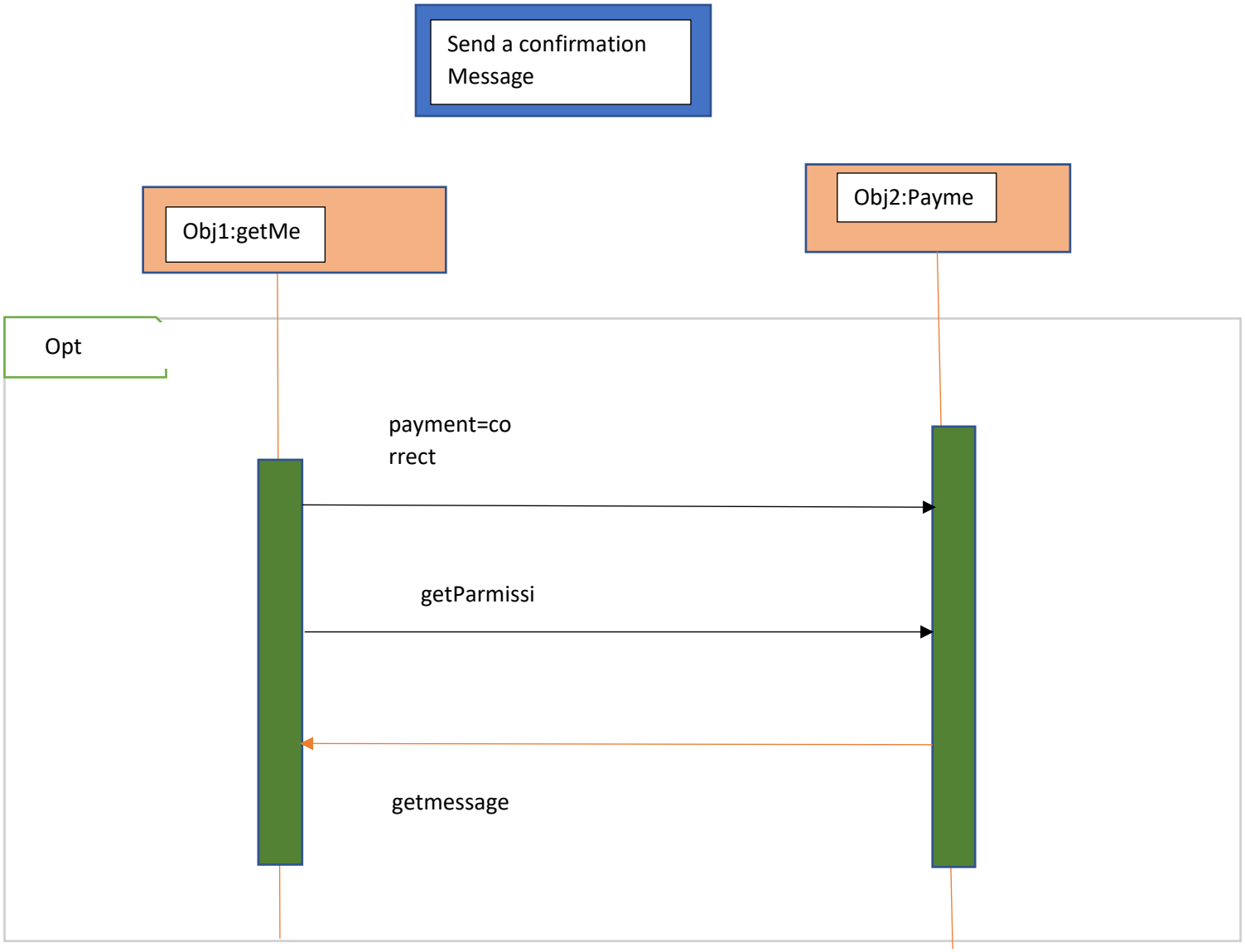

Attach Supporting Document
Obj1:Attach
Obj2:Fillup
Opt
FillupForm= correct
getParmissi
AttachSupp


View Filled Form
Obj1:View
Obj2:Databa
Opt
FillupForm=
correct
getParmissi
ViewFilledFo


Re-check filled form records
Obj1:Rechec
Obj2:Databa
Opt
FillupForm= correct
getParmissi
Recheck-


Payment
Obj1:payme
Obj2:Databa
Opt
FillupForm=
correct
getParmissi
canDoPaym


Edit & Delete Records
Obj1:edit_d
Obj2:view
Opt
FillupForm=
correct
getParmissi
edit_delete_


Send a confirmation Message
Obj1:getMe
Obj2:Payme
Opt
payment=co rrect
getParmissi
getmessage

Thank You