# Software Engineering 2
## Cover Sheet

## Project Title:
Library dashboard

## Project Description:
• Admin and Reader must first log into the library dashboard.

• Admin can manage books in the library ( add, update, delete, and list).

• Each book has its own chapters, which the admin control.

• Admin can manage readers ( add, update, delete, and list).

• Reader can request a book from the gallery.

• Readers can show the requested books.

## Main functions:

Admin User:
• Login/ Logout
• Manage books (CRUD)
• Manage chapters for each book (CRUD)
• Manage Reader (CRUD)
• Accept or decline requests by the reader (CRUD)
• Show the history of reader requests for all books

2. Reader User:
• Login/ Logout
• Can view and filter books
• Send a request to the admin with the required book
• Show a history of book searches related to his account only

كلية الحاسبات والذكاء الإصطناعي
Faculty of Computers & Artificial Intelligence

جامعة حلوان
HELWAN UNIVERSITY

# Software Engineering-2 Cover Sheet

Project Title: Library dashboard

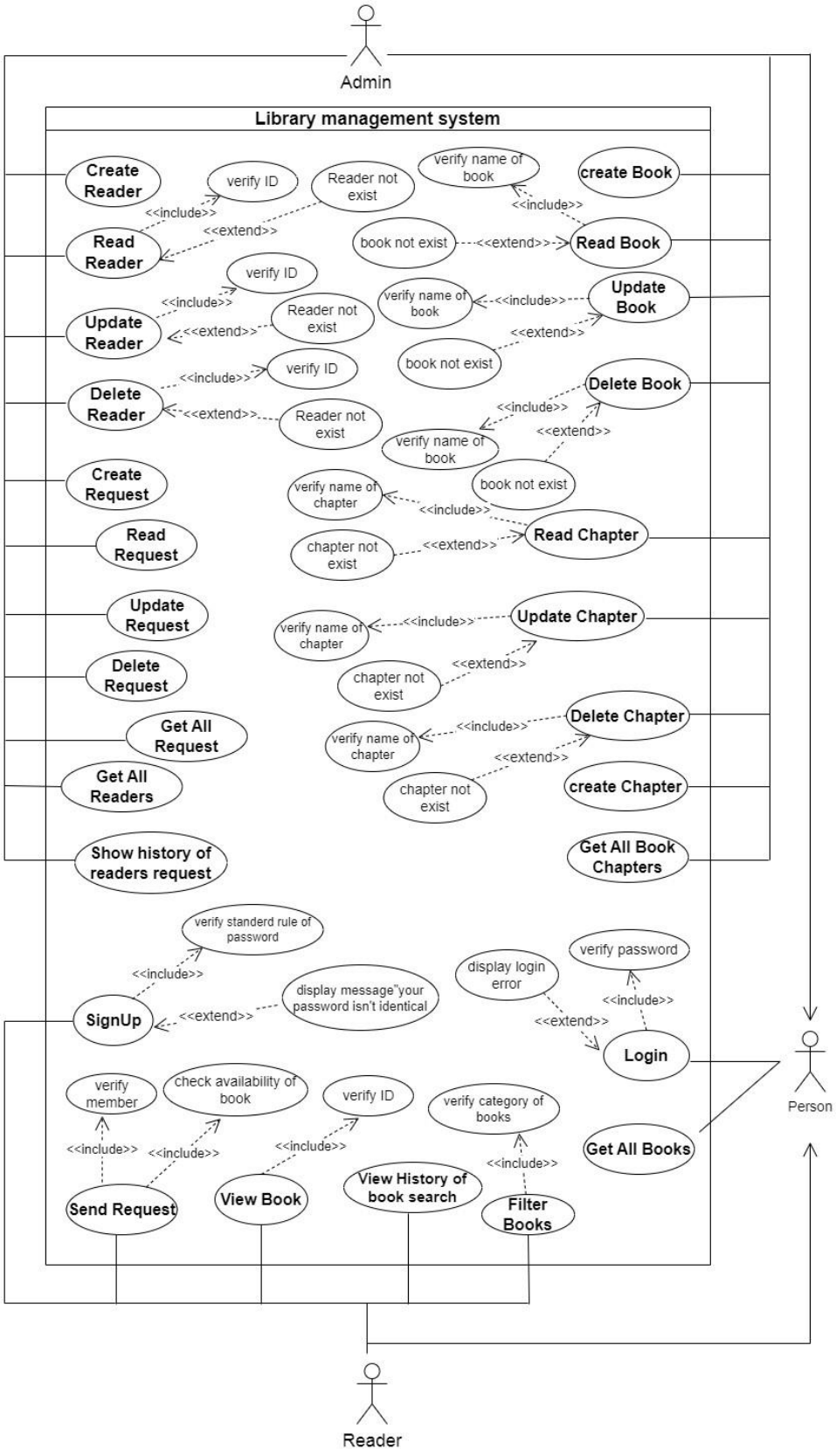
Row Number (in PDF): 67 Time: 11:00

| # | Member Name (printed in arabic) | Member Id (printed) | Handwritten signature |
|---|---|---|---|
| 1 | محمد عبدالمنعم رزق موسي | 202000782 | |
| 2 | علي محسن محمد عبدالمنعم | 202000570 | |
| 3 | سامح محمد يوسف مرسي | 202000381 | |
| 4 | محمد اشرف عبد الموجود حسن | 202000731 | |
| 5 | محمد ذكريا منصور محمود | 202000752 | |
| 6 | عمر خالد عبدالحميد عبدالحميد | 201900510 | |
| 7 | | | |

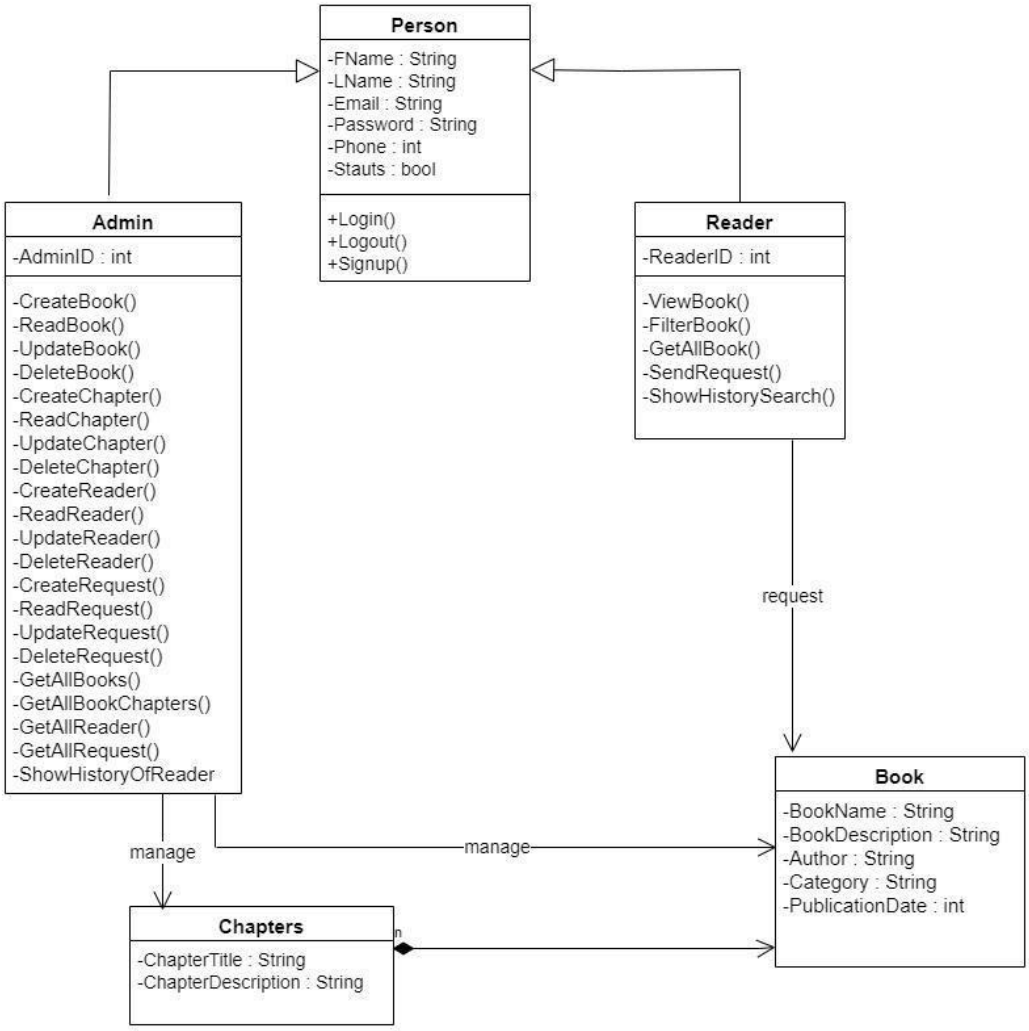
## Project Requirements(grades)

| | | |
|---|---|---|
| SRS | 2 | |
| SDD | 2 | |
| Validation | 3 | |
| OCL | 4 | |
| AOP | 4 | |
| Microservices | 1 | |
| Cloud | 4 | |

20

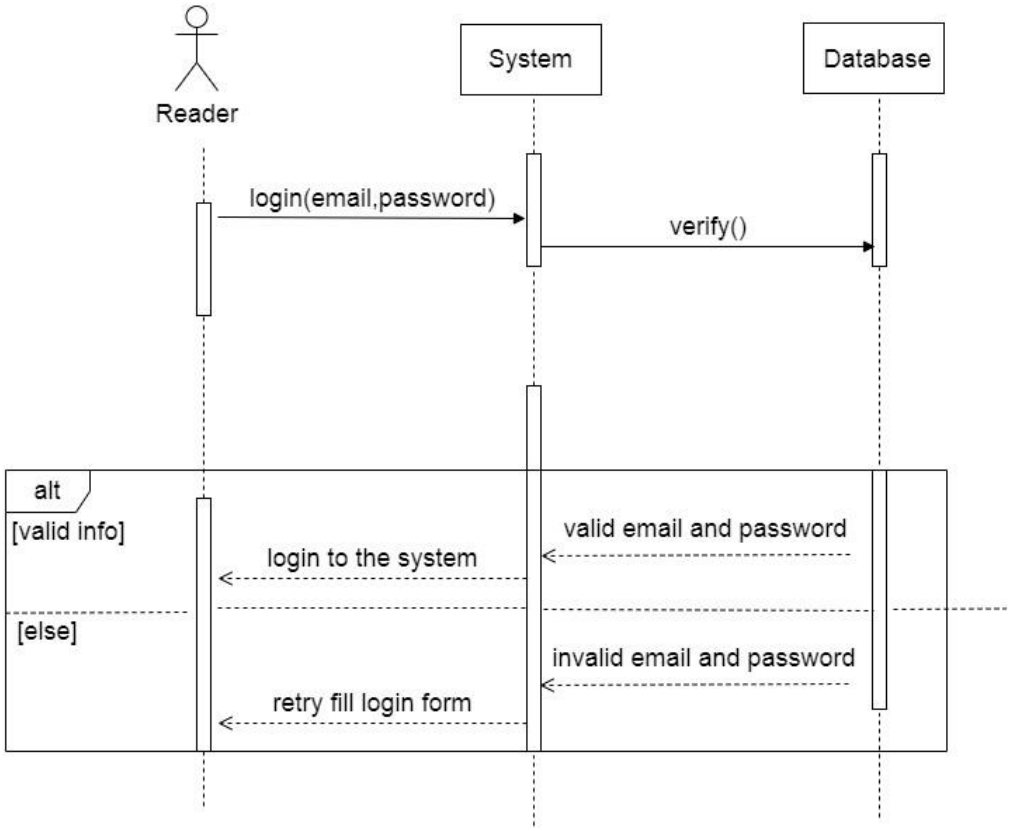



# Class diagram:

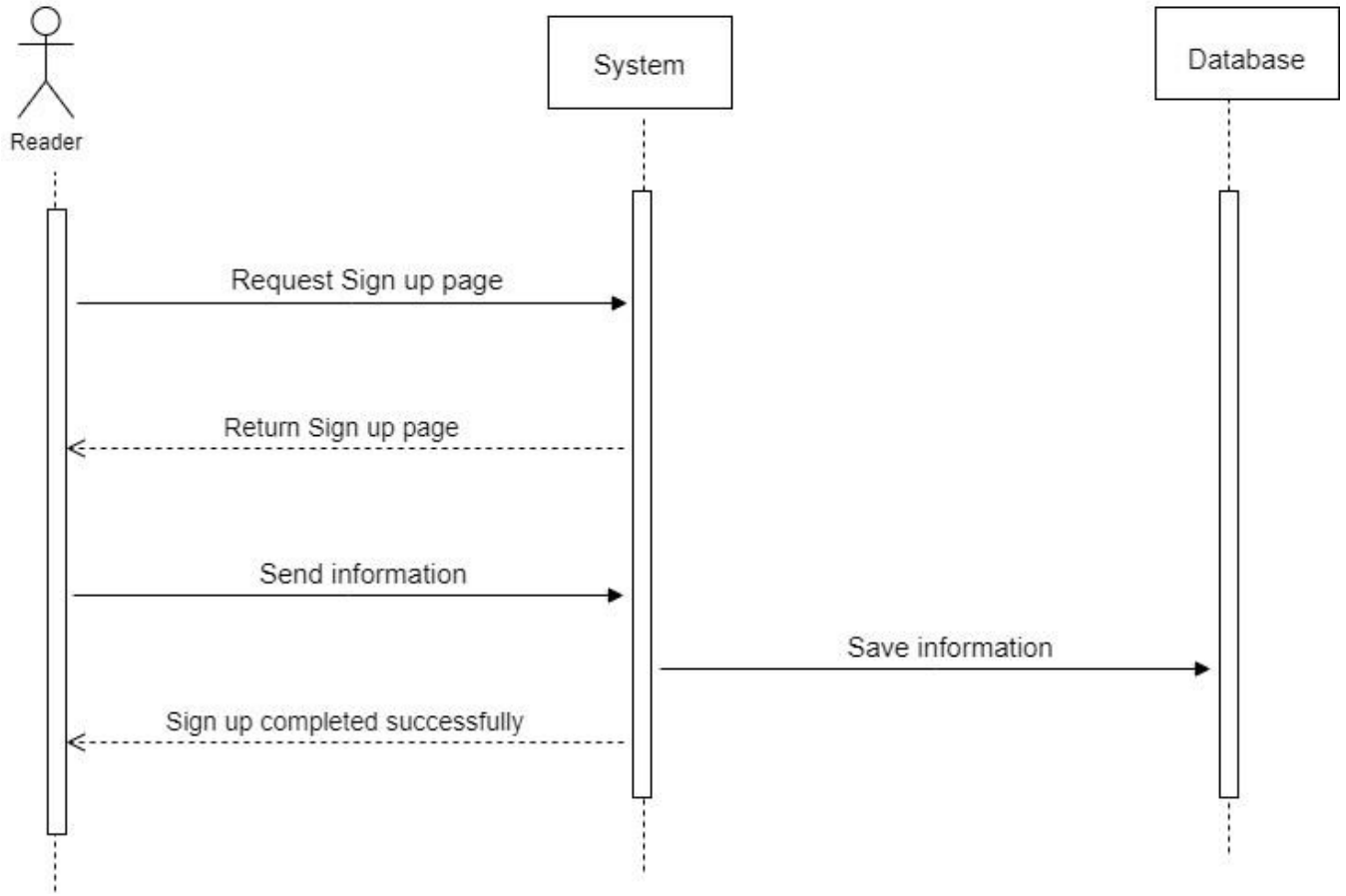



# Sequence diagram:

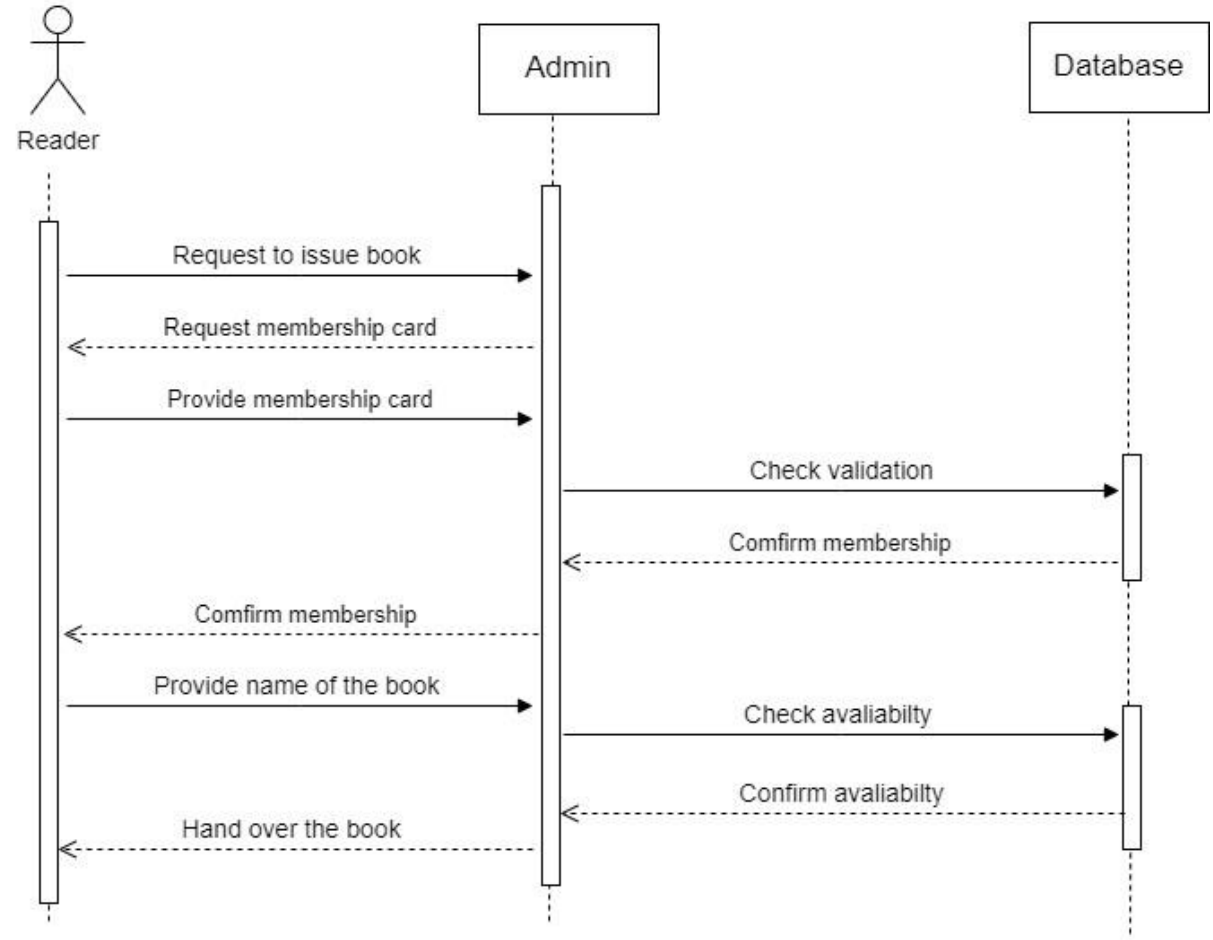
# Login for reader


Reader
System
Database
login(email,password)
verify()
alt
[valid info]
valid email and password
login to the system
[else]
invalid email and password
retry fill login form

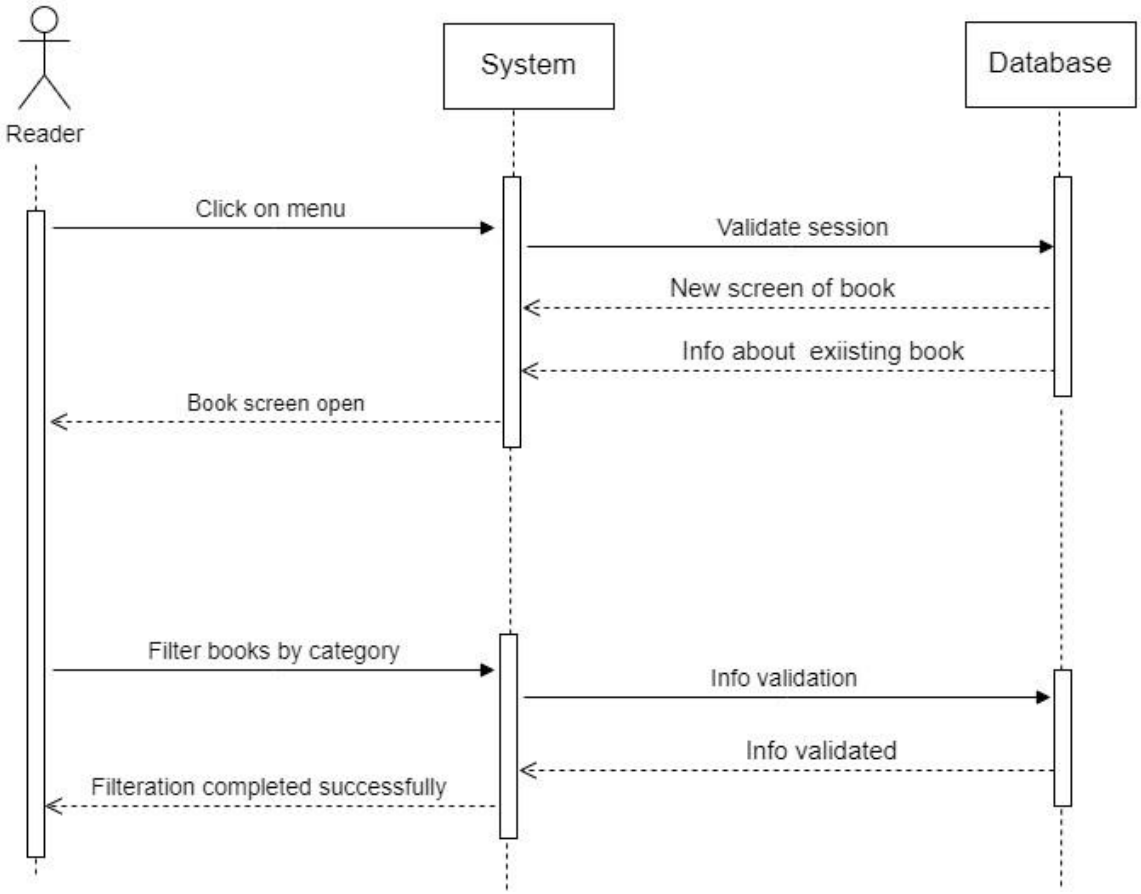
**sign up**


Reader
System
Database
Request Sign up page
Return Sign up page
Send information
Save information
Sign up completed successfully

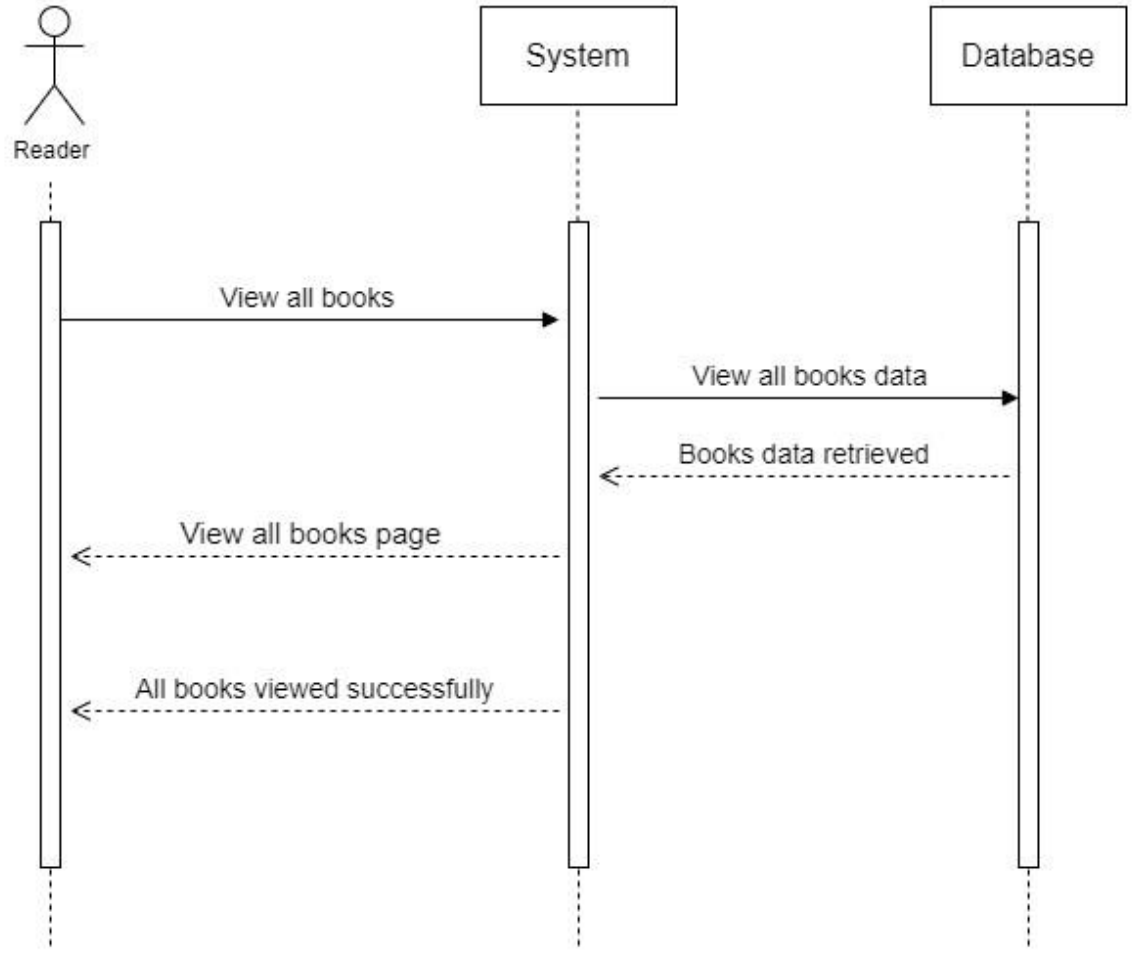
**Request book**



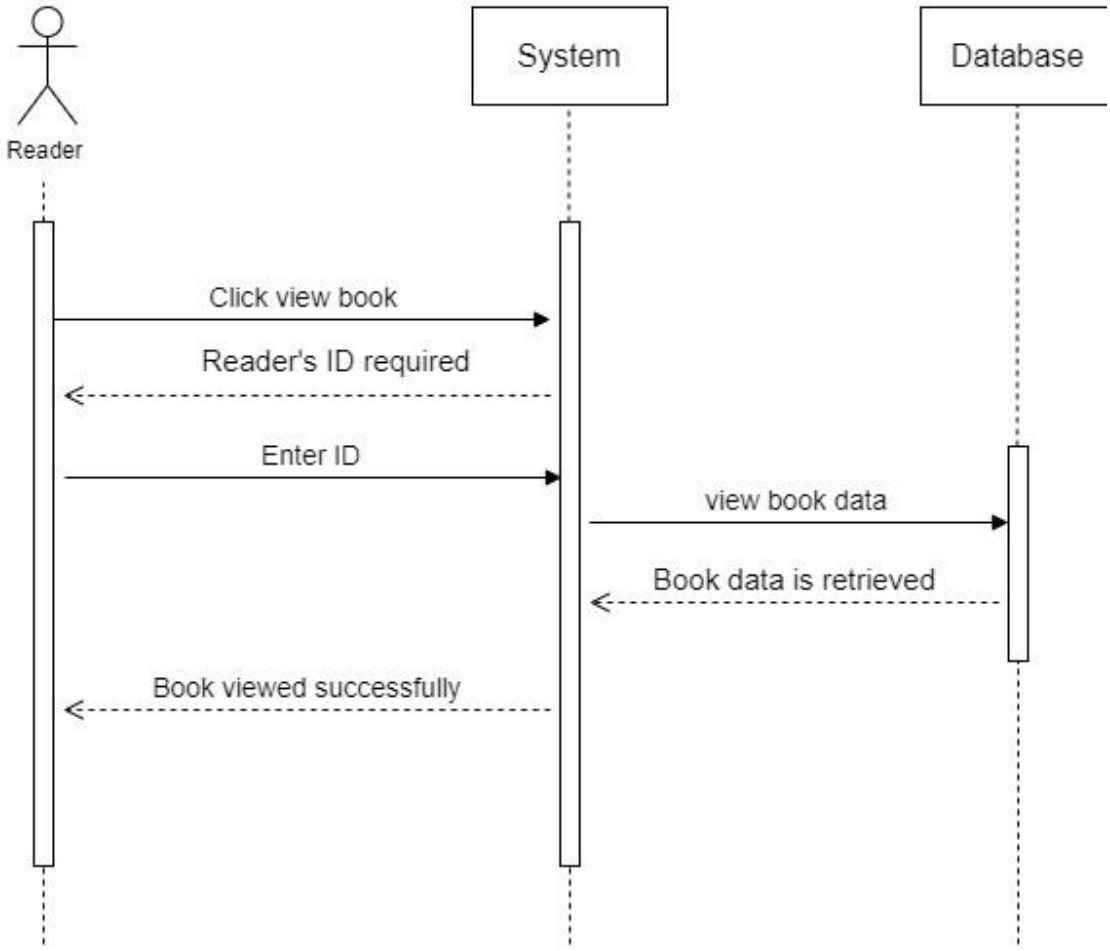
**Filter books**



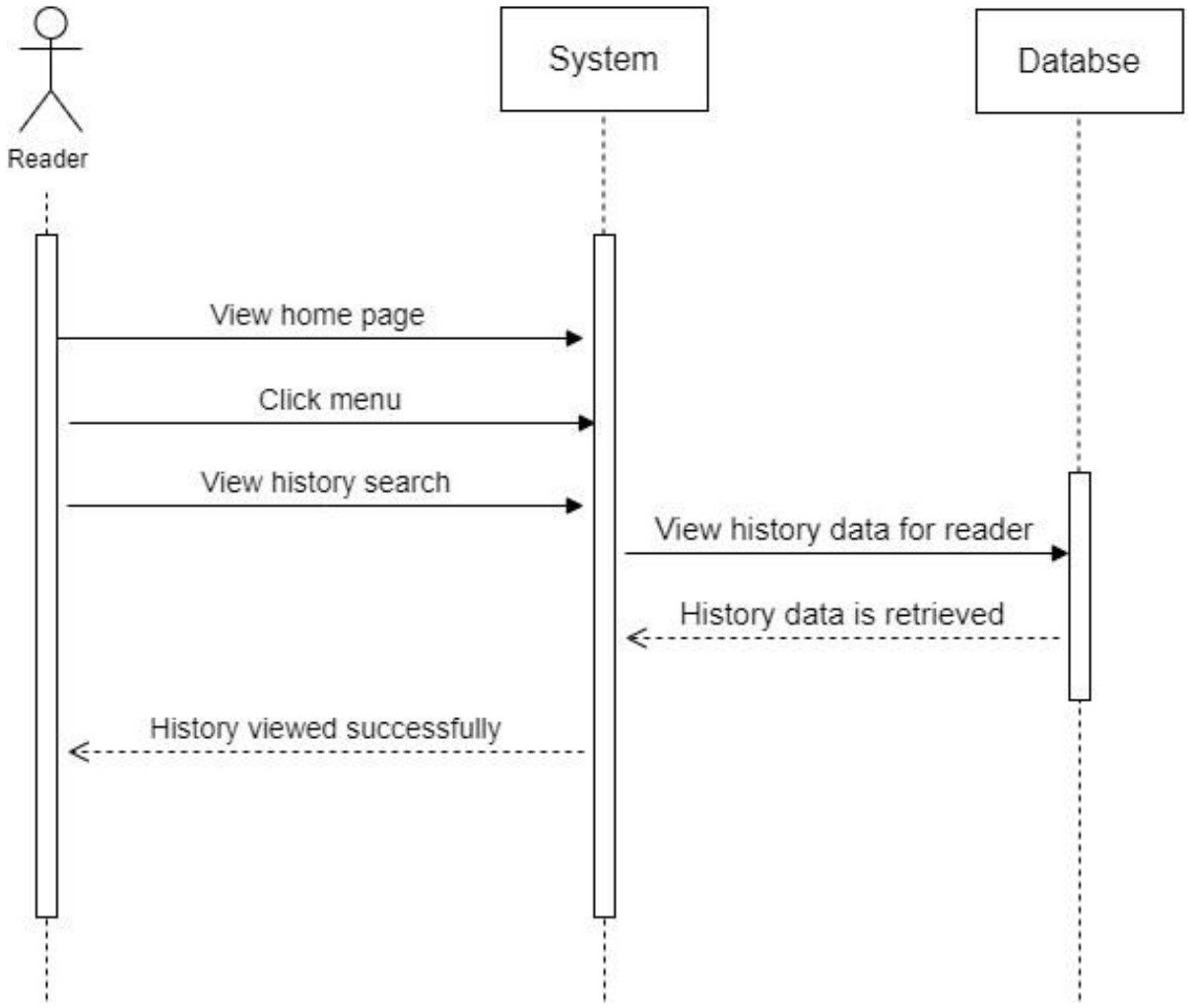
**Get all books**



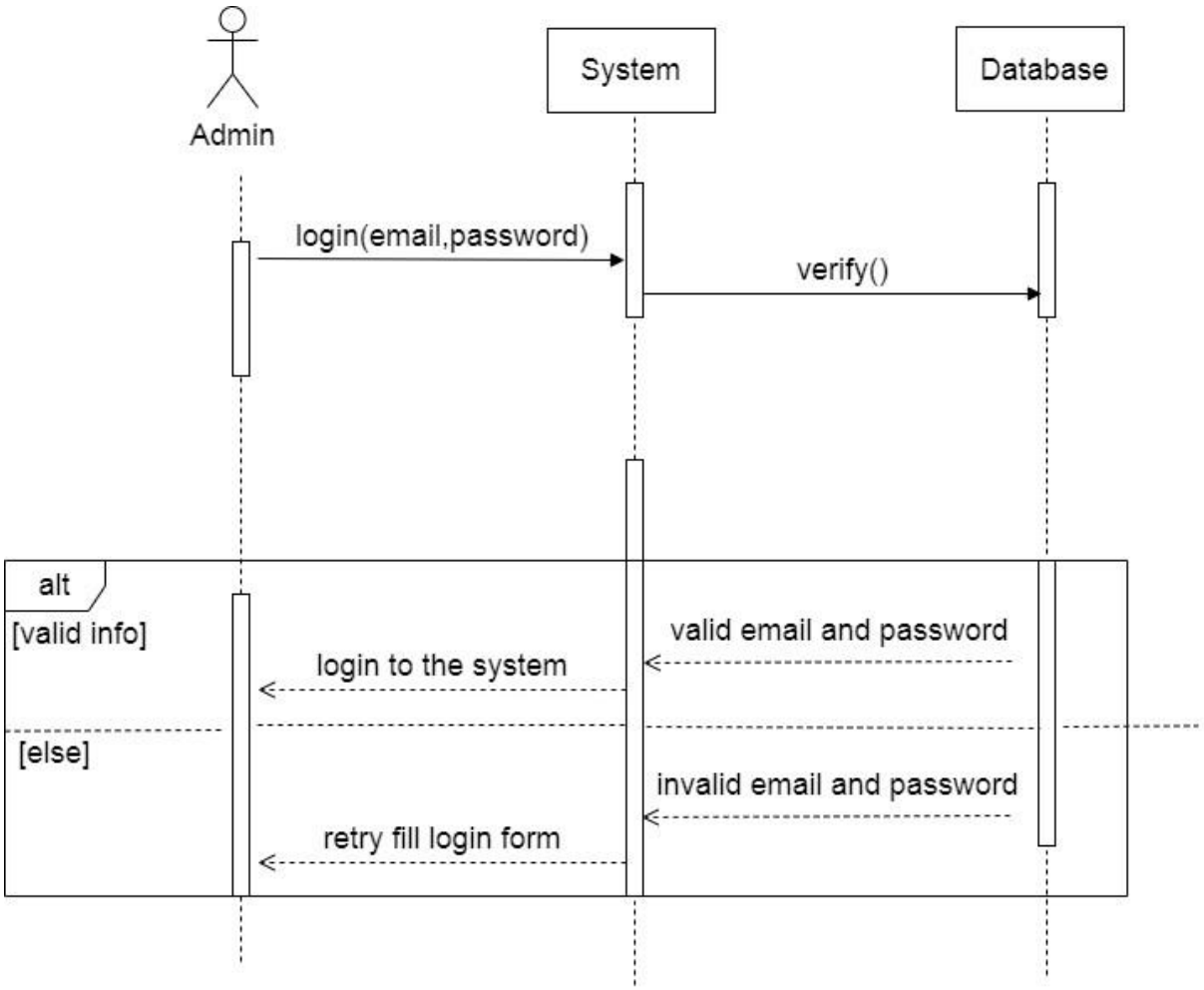
**View book**


Reader
System
Database
Click view book
Reader's ID required
Enter ID
view book data
Book data is retrieved
Book viewed successfully

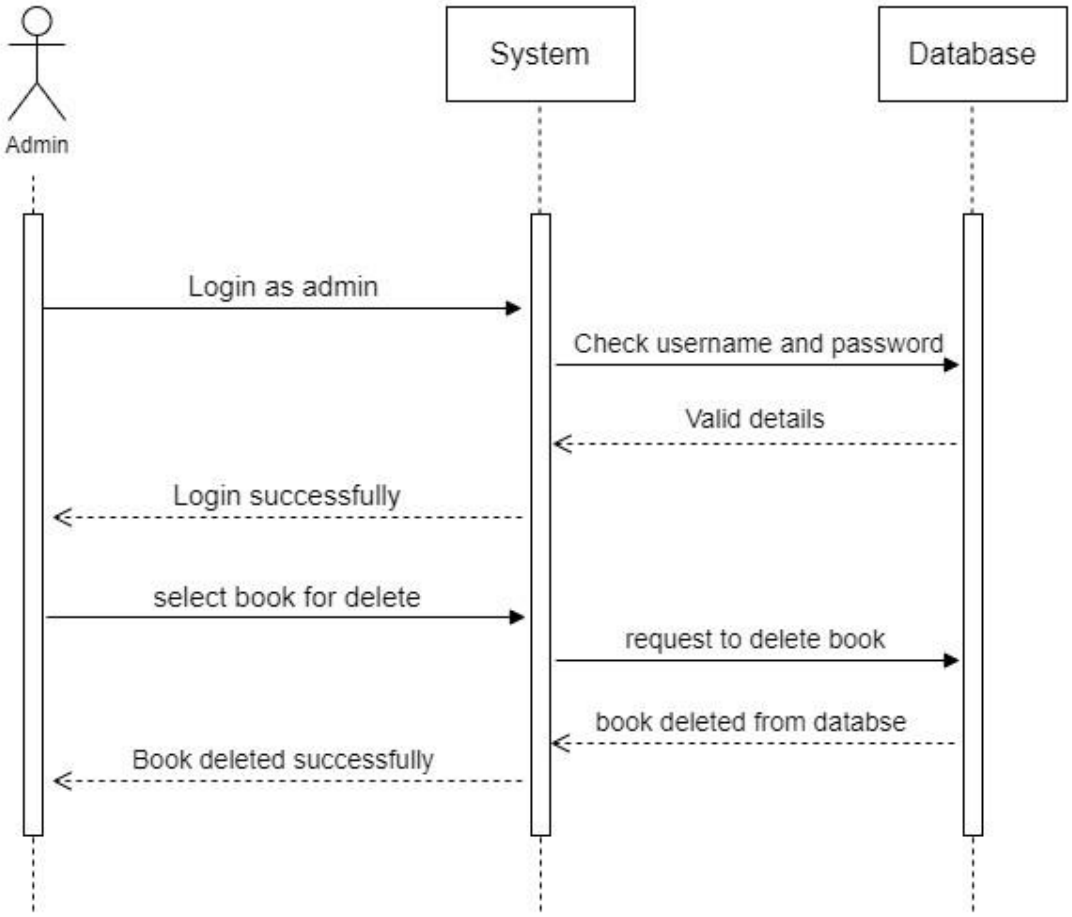
**view history**



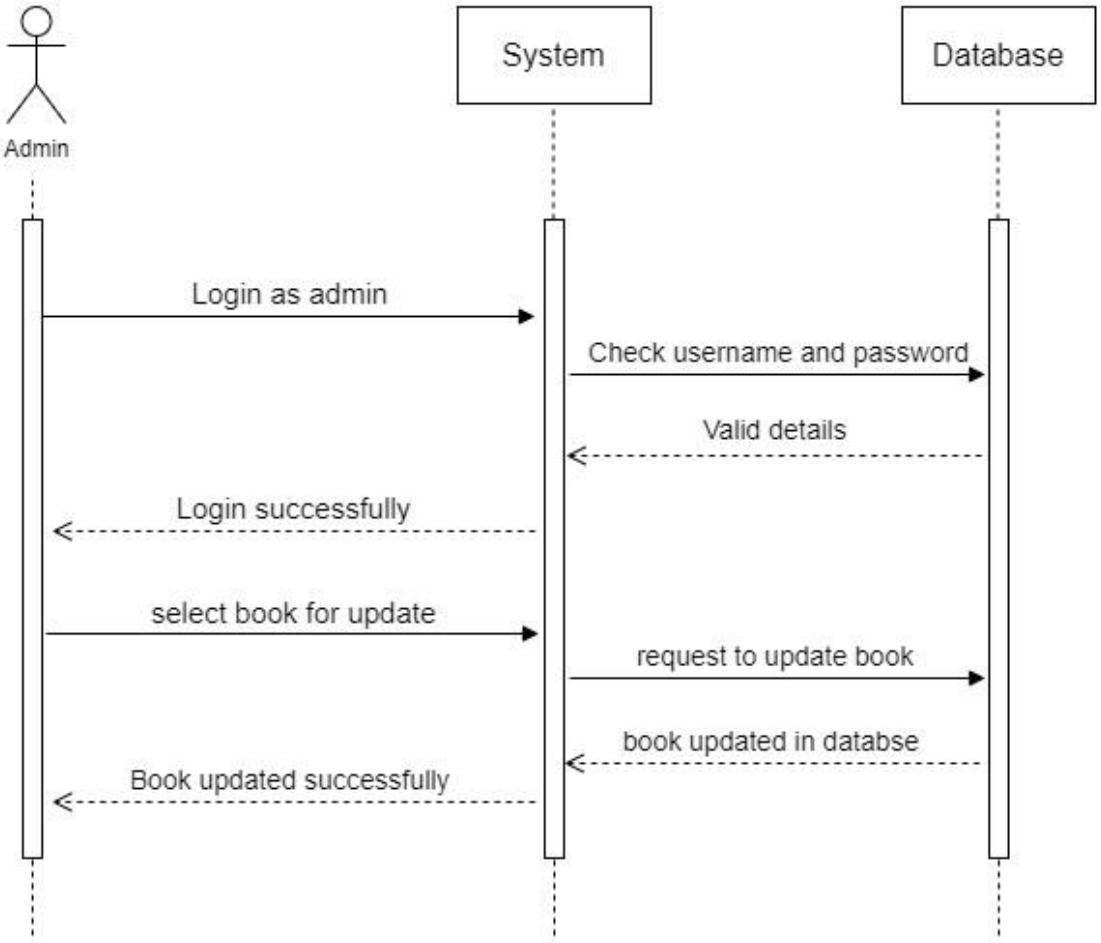

Login for admin


Admin
System
Database
login(email,password)
verify()
alt
[valid info]
valid email and password
login to the system
[else]
invalid email and password
retry fill login form

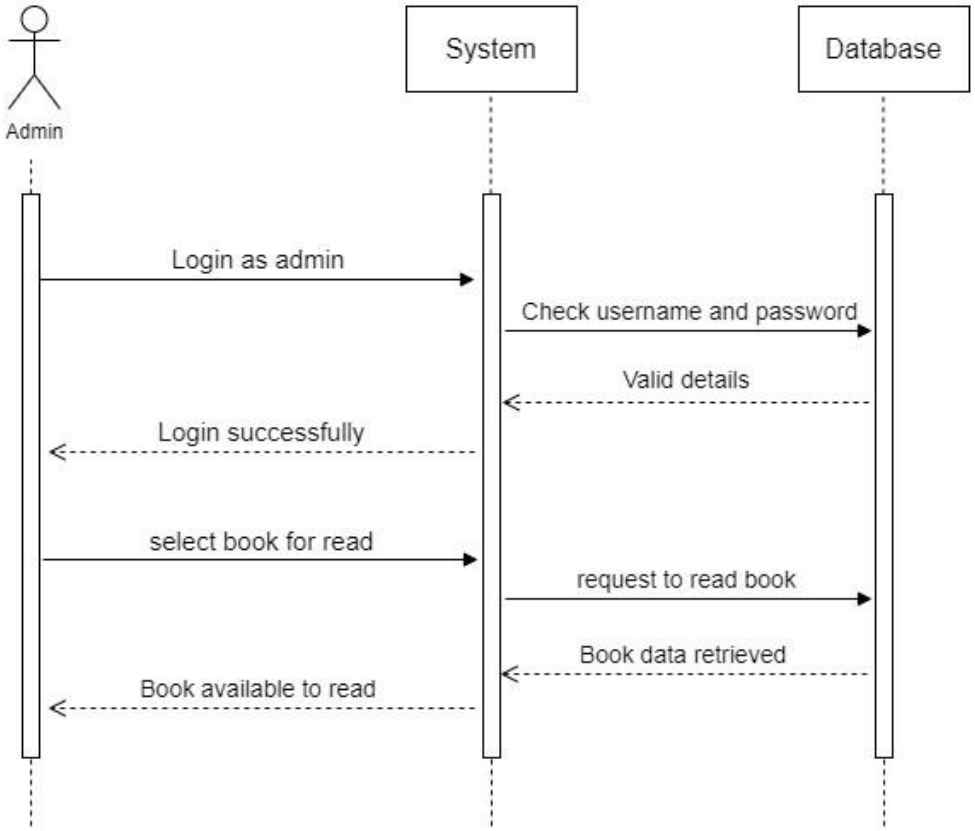
**Delete book**


Admin
System
Database
Login as admin
Check username and password
Valid details
Login successfully
select book for delete
request to delete book
book deleted from databse
Book deleted successfully

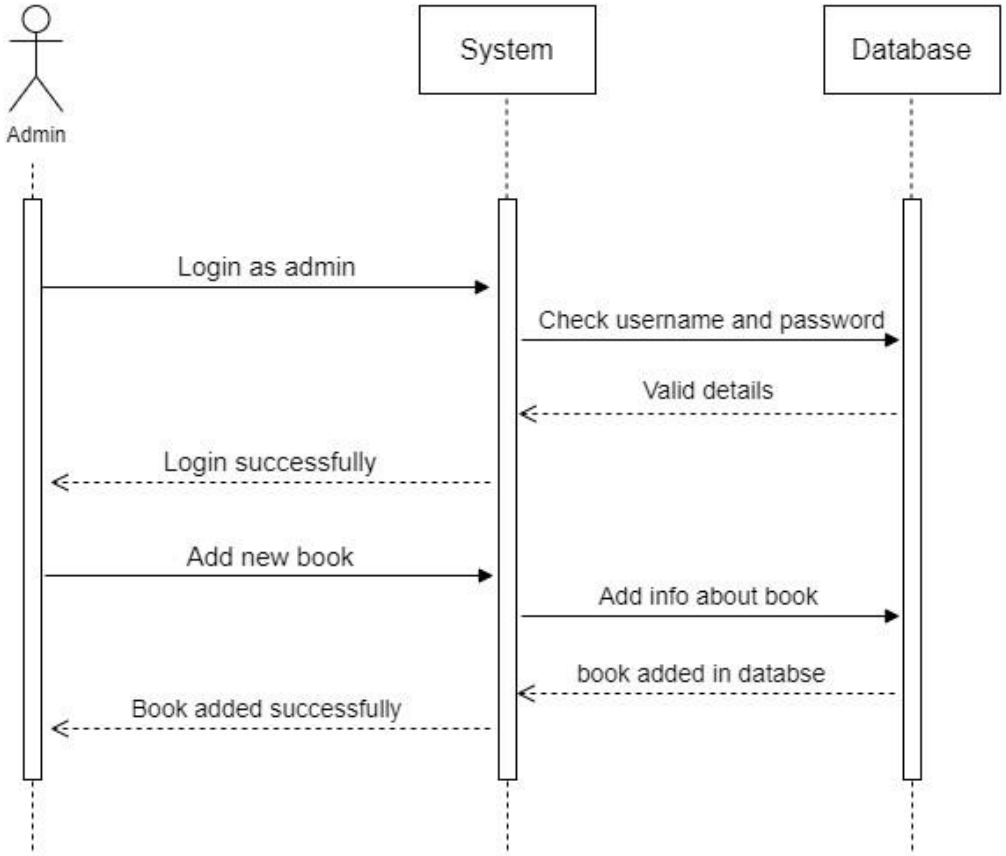
**update book**


Admin
System
Database
Login as admin
Check username and password
Valid details
Login successfully
select book for update
request to update book
book updated in databse
Book updated successfully

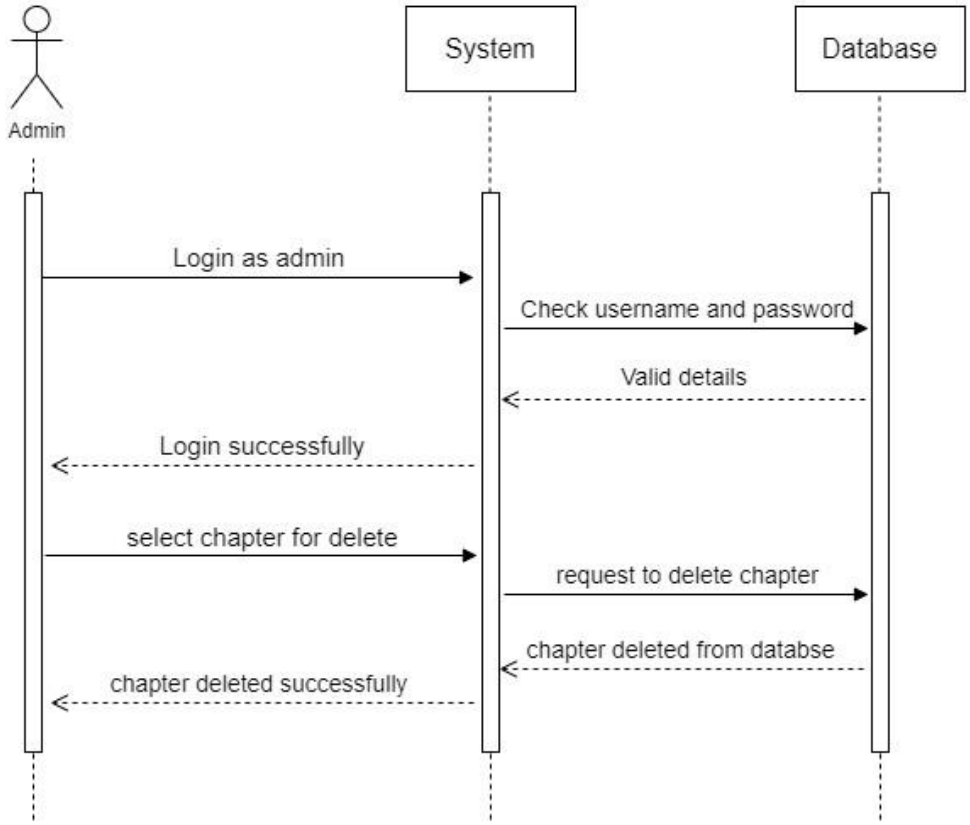
**read book**


Admin
System
Database
Login as admin
Check username and password
Valid details
Login successfully
select book for read
request to read book
Book data retrieved
Book available to read

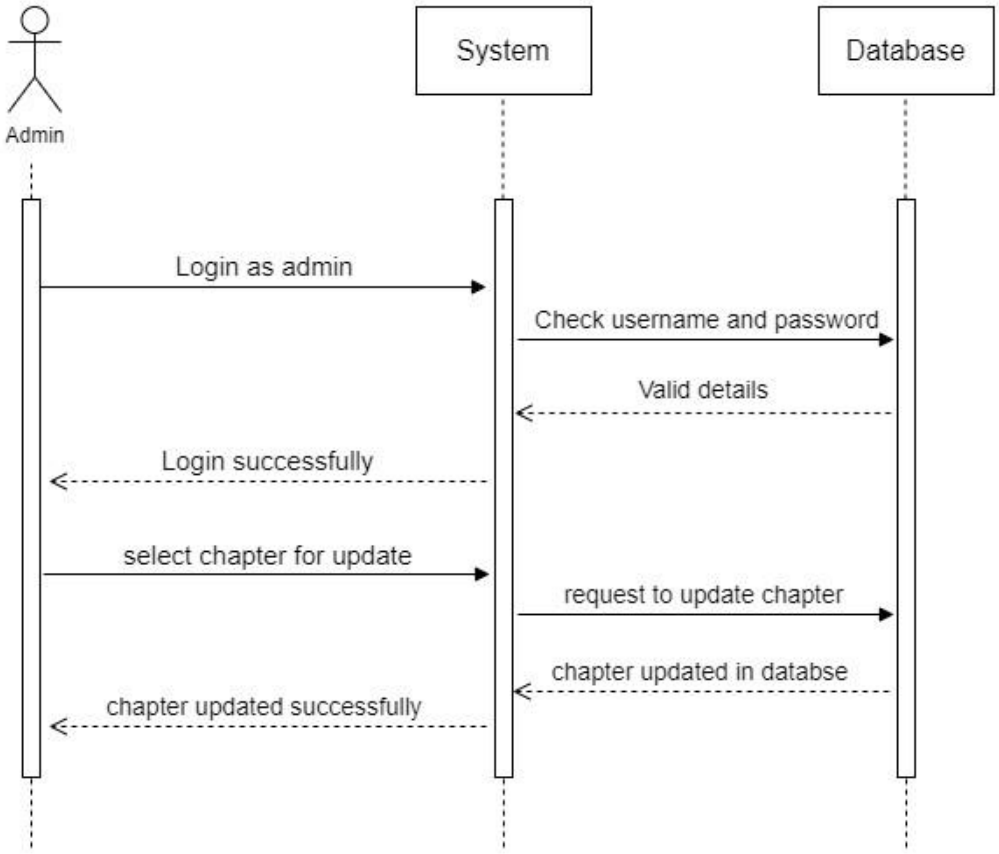
**create book**


Admin
System
Database
Login as admin
Check username and password
Valid details
Login successfully
Add new book
Add info about book
book added in databse
Book added successfully

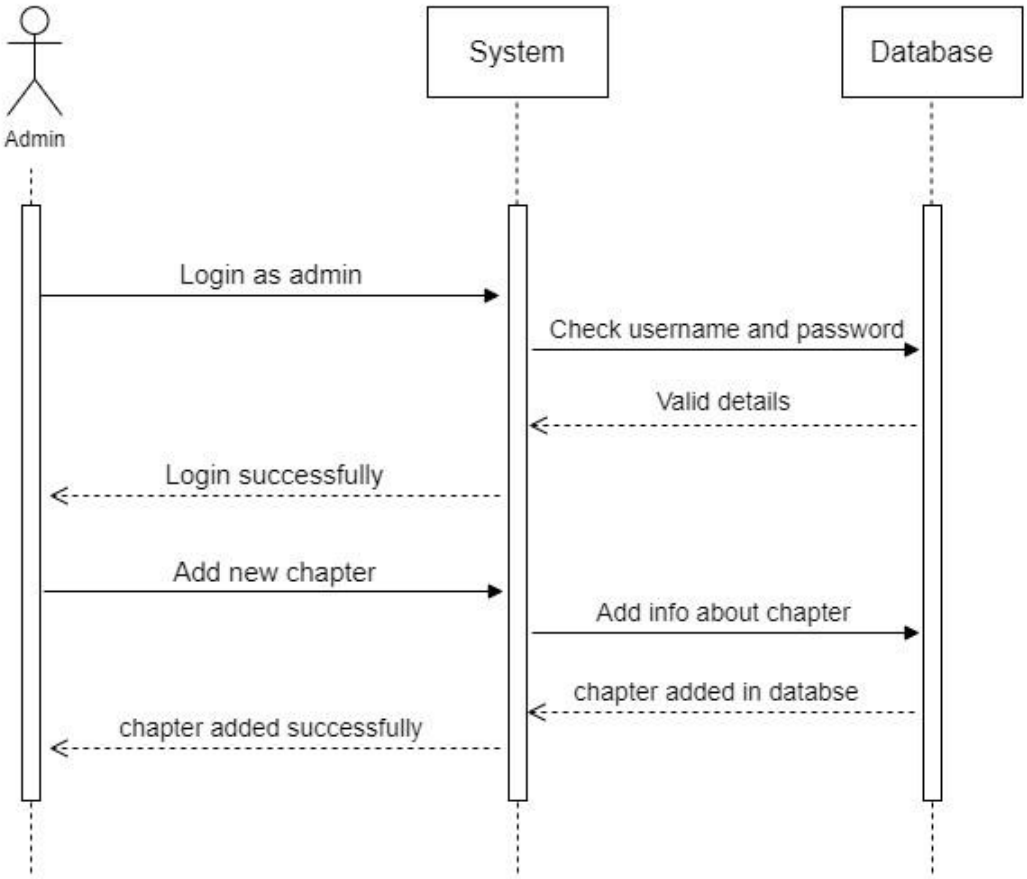
**Delete chapter**



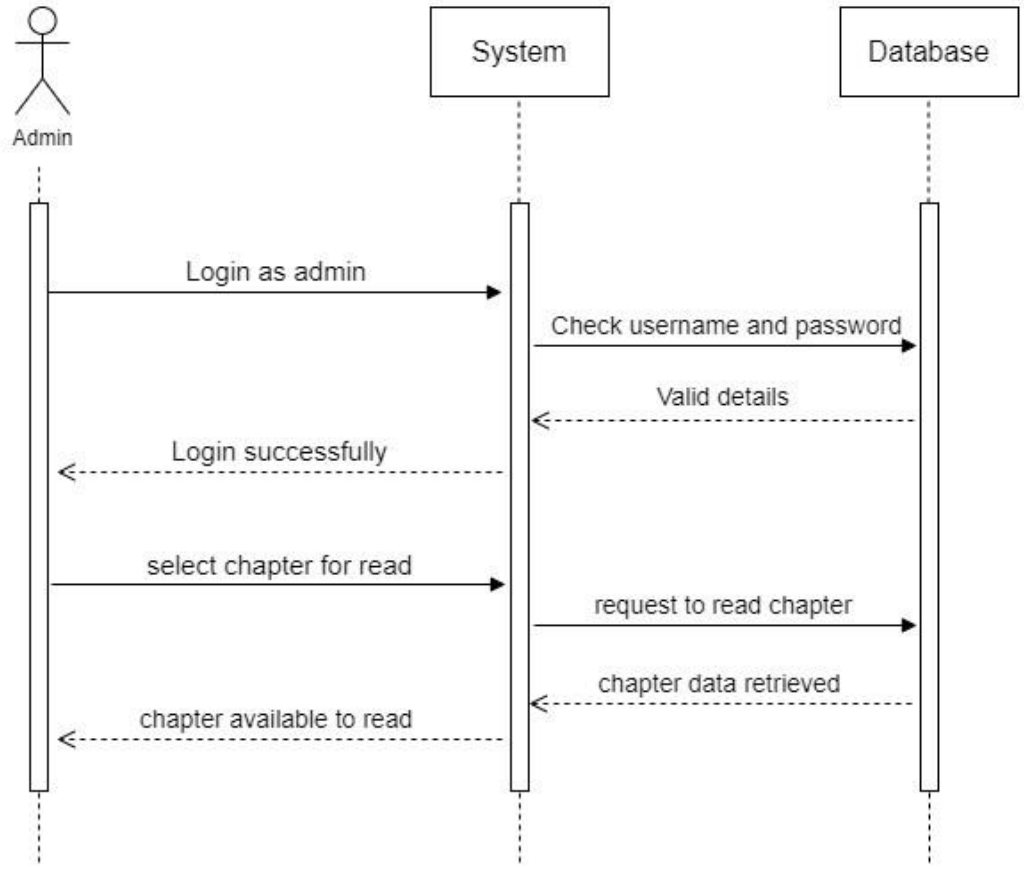
**update chapter**



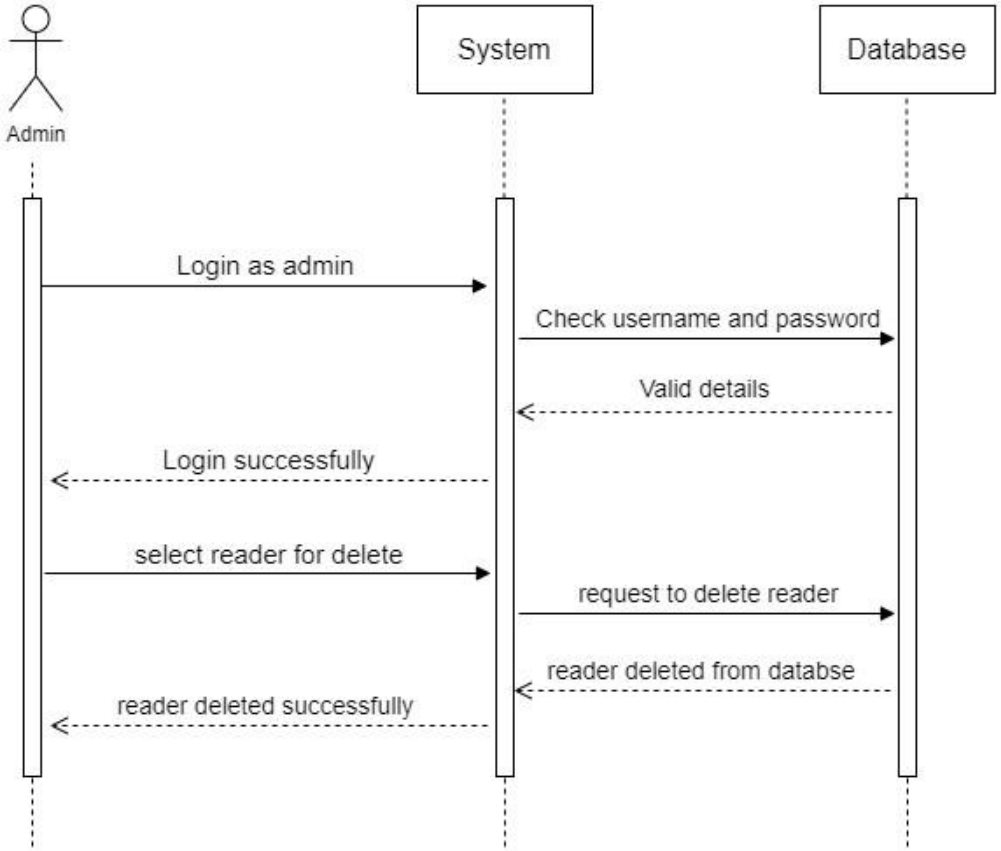
**create chapter**



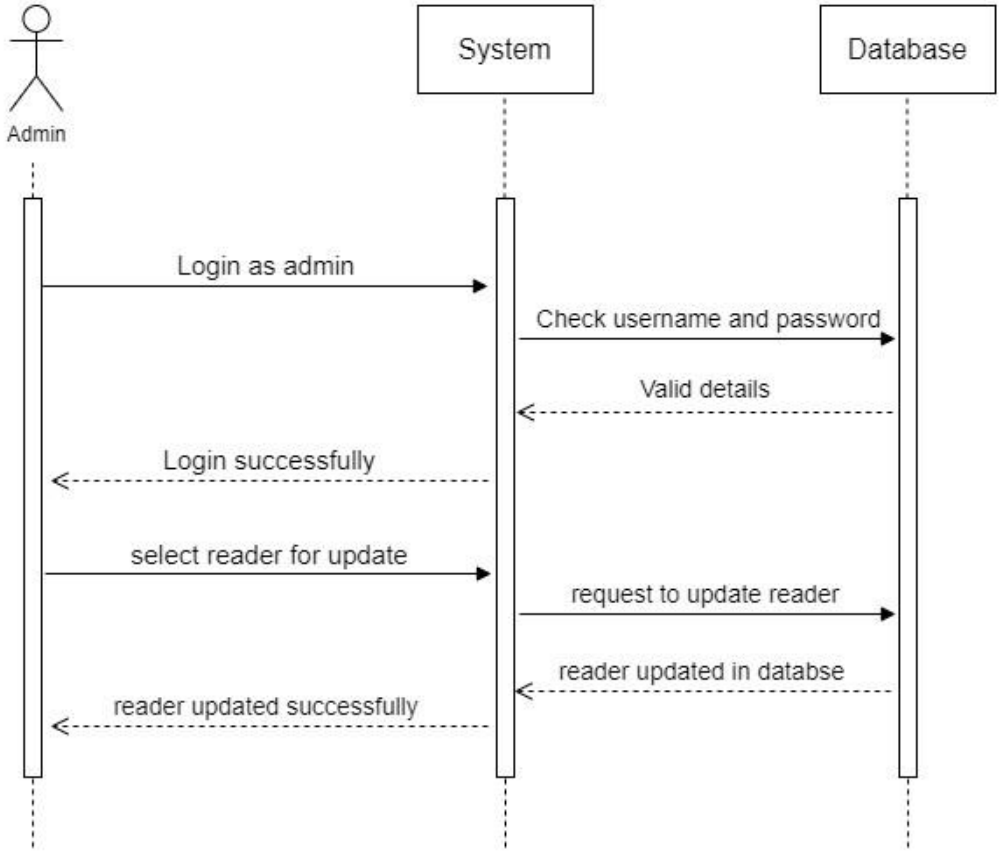
**read chapter**



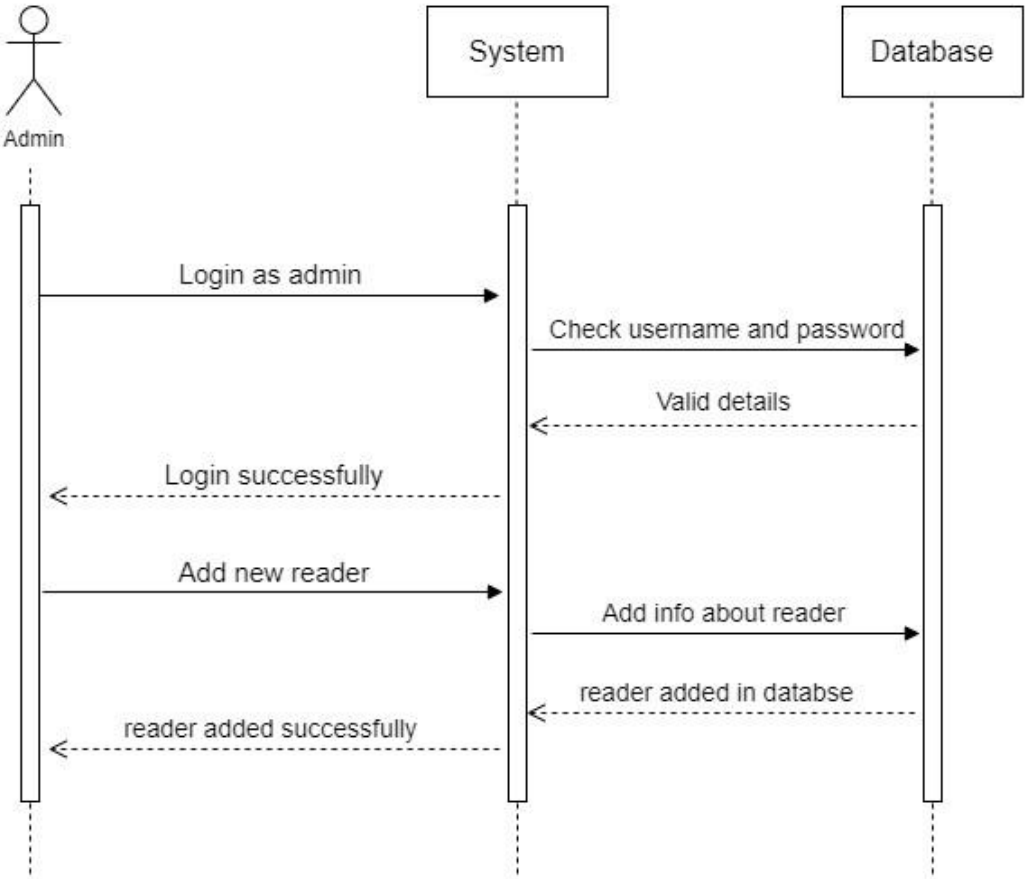
**Delete reader**



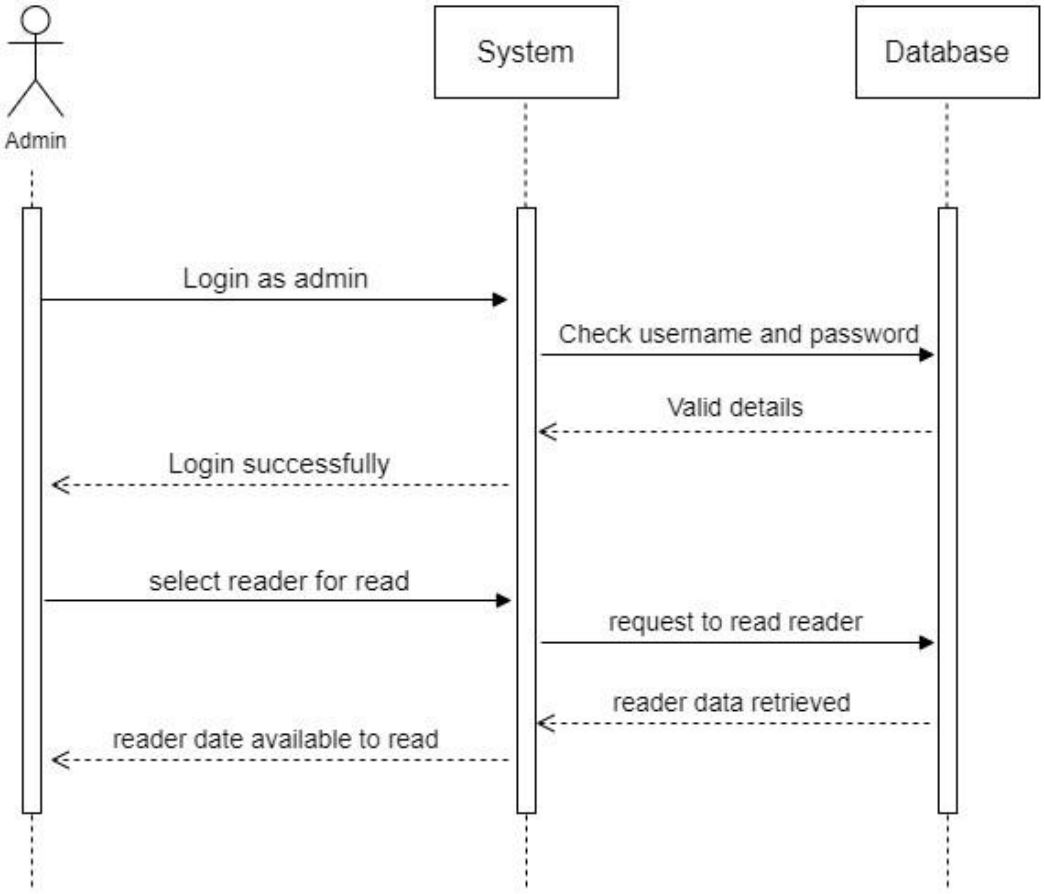
**update reader**



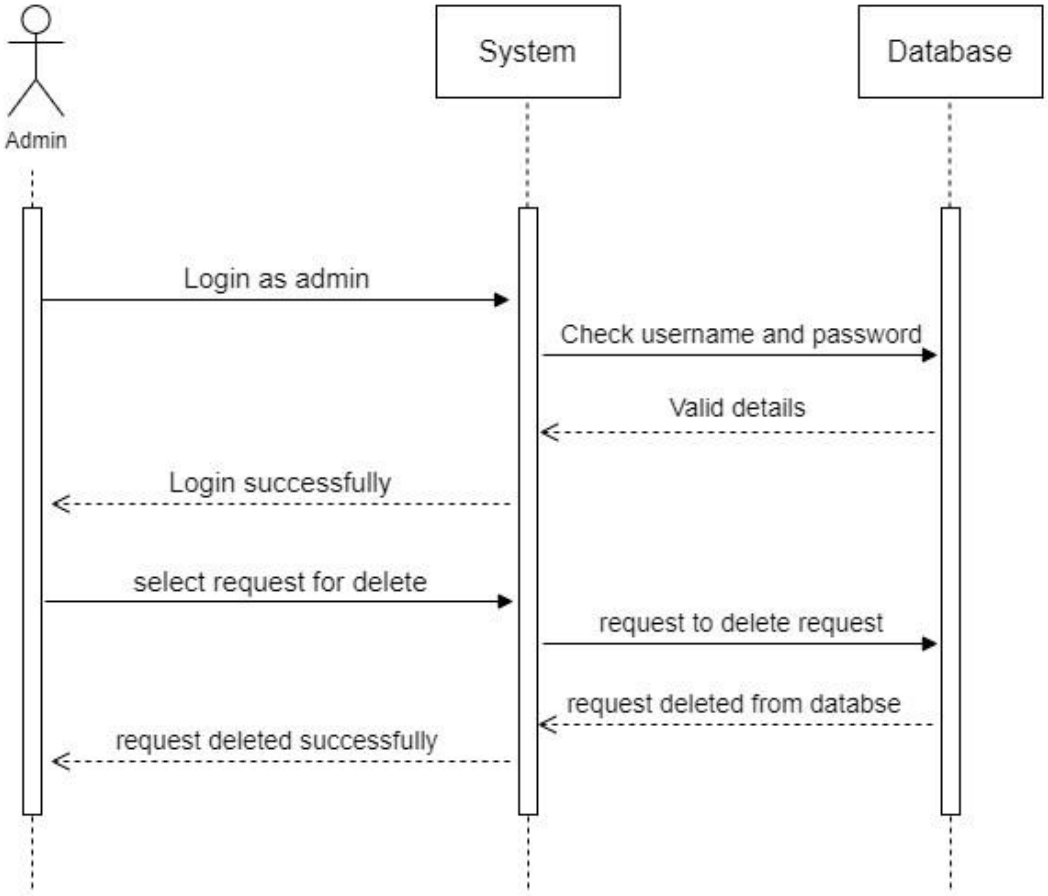
**create reader**



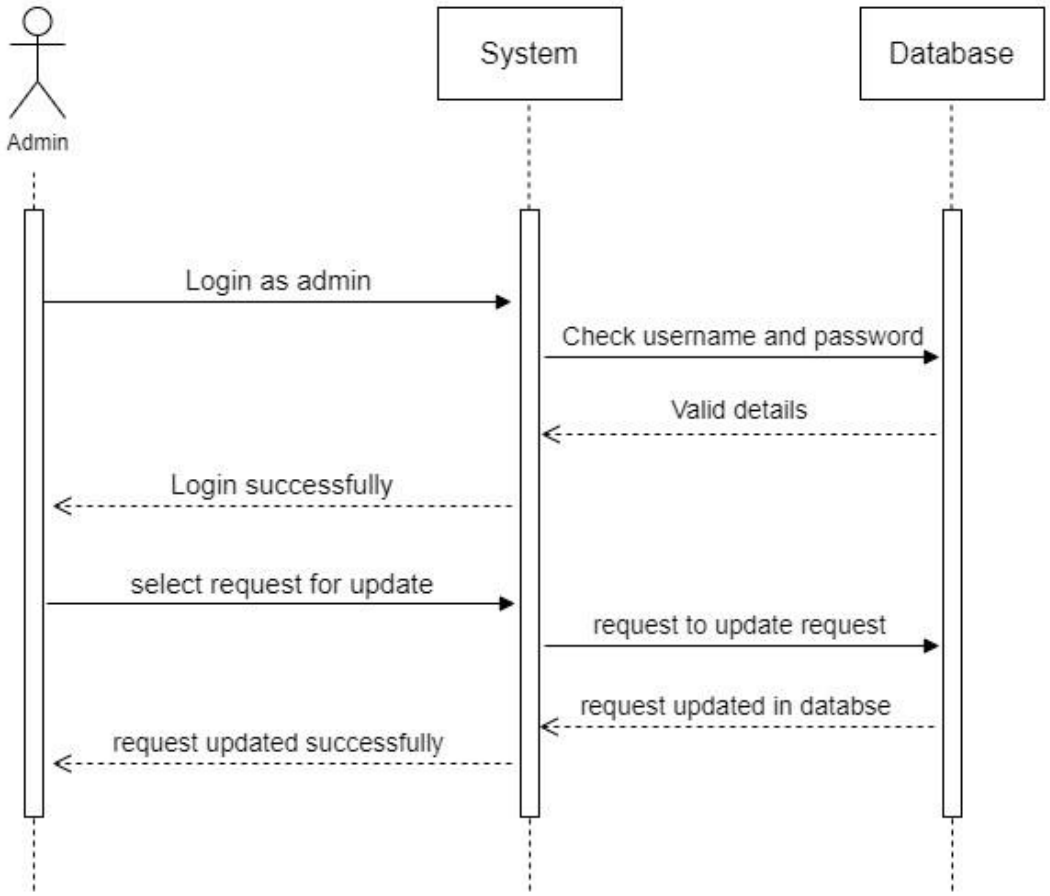
**read reader**


Admin
System
Database
Login as admin
Check username and password
Valid details
Login successfully
select reader for read
request to read reader
reader data retrieved
reader date available to read

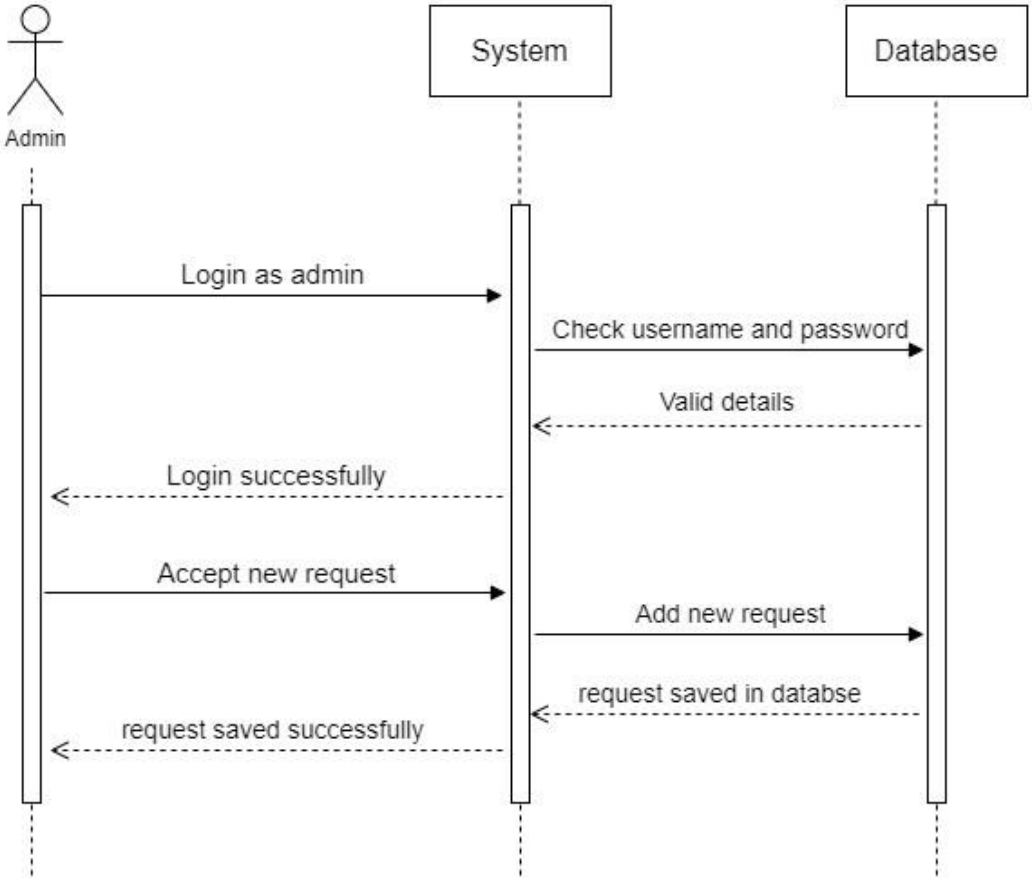
**Delete request**



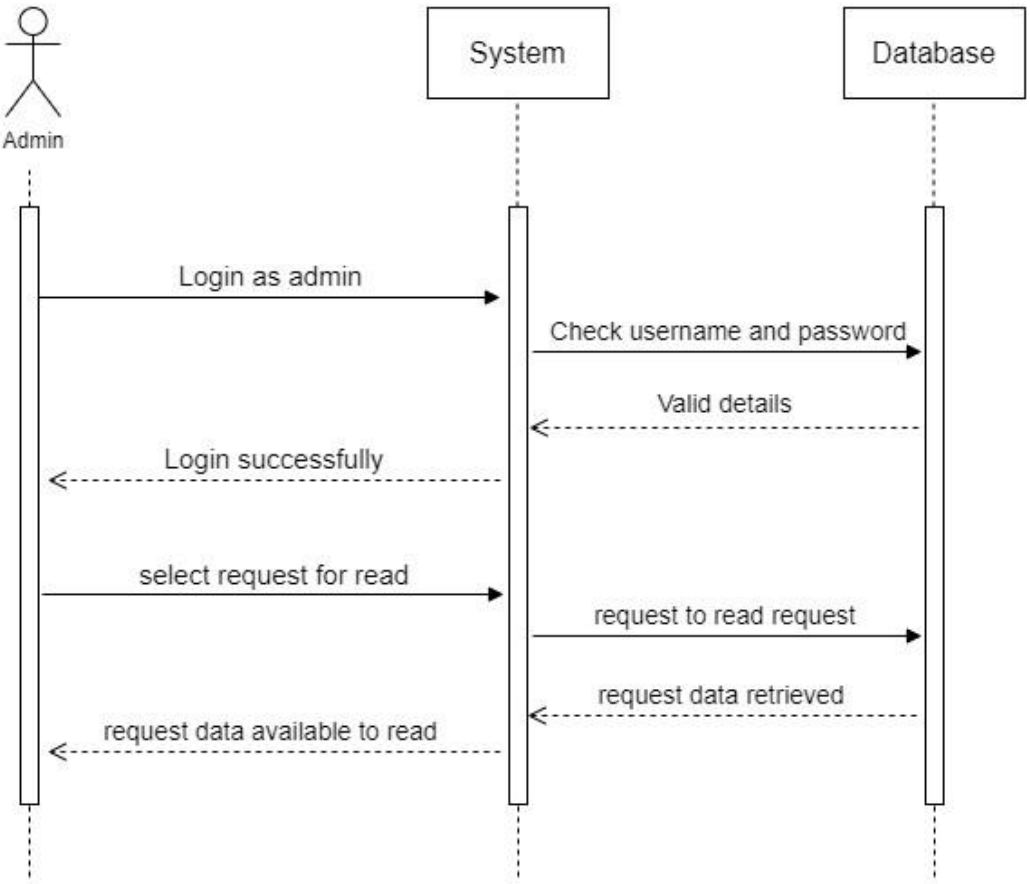
**update request**



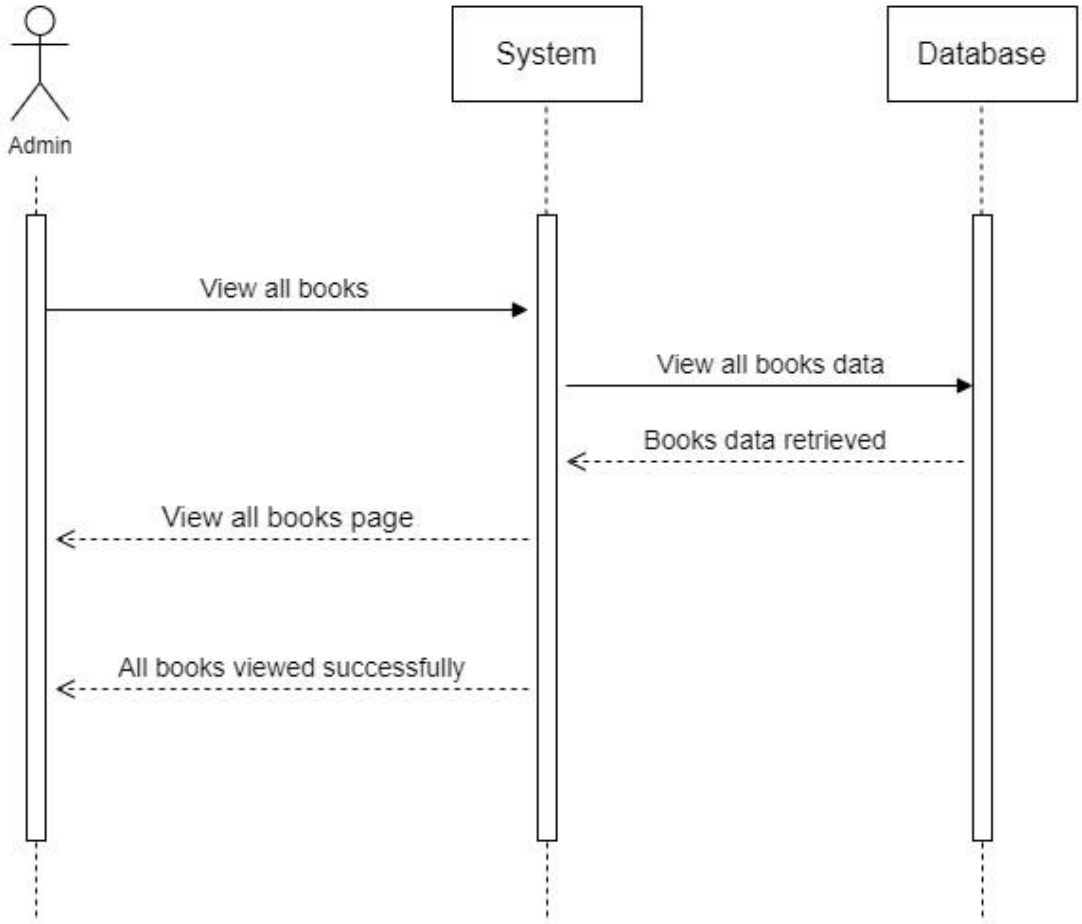
**create request**



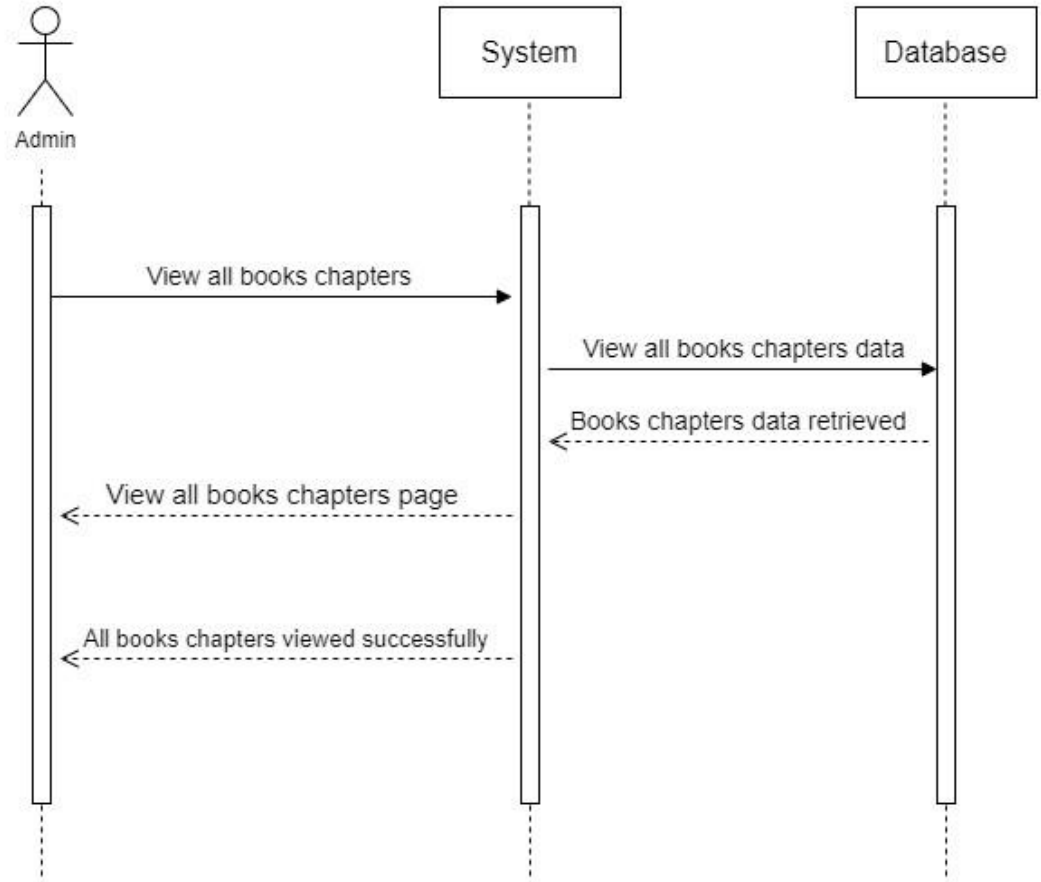
**read request**



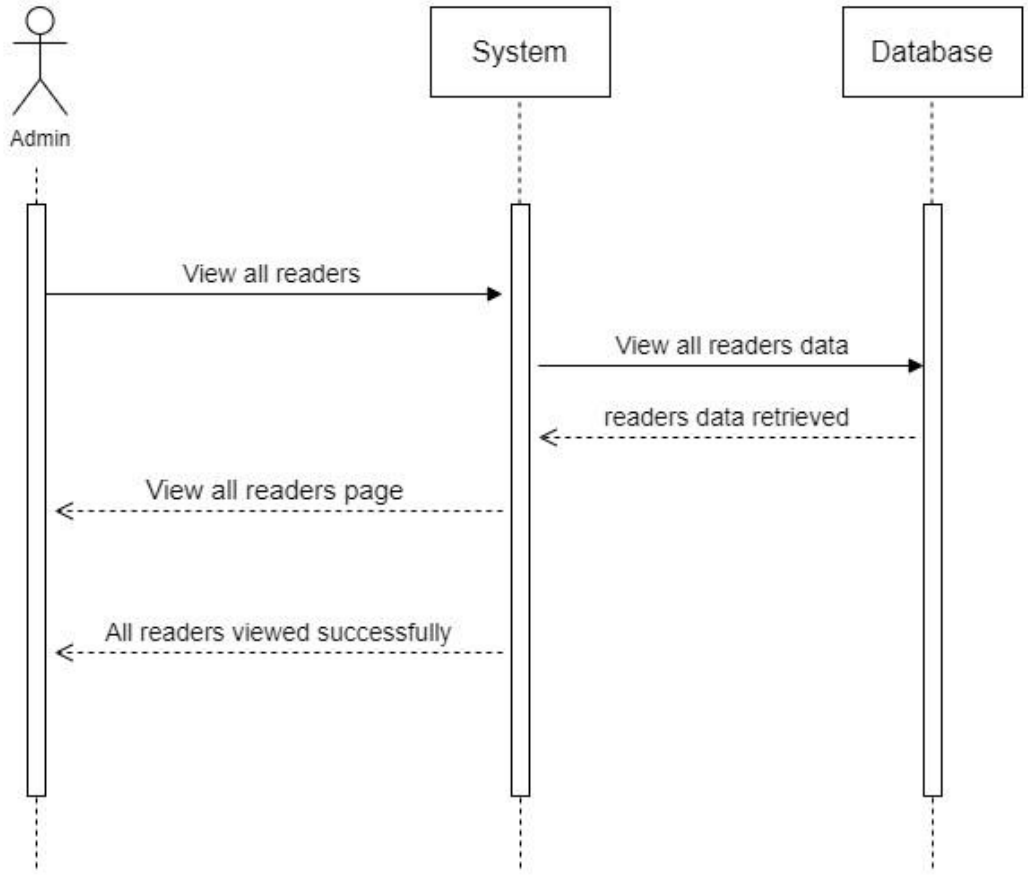
**Get all books**



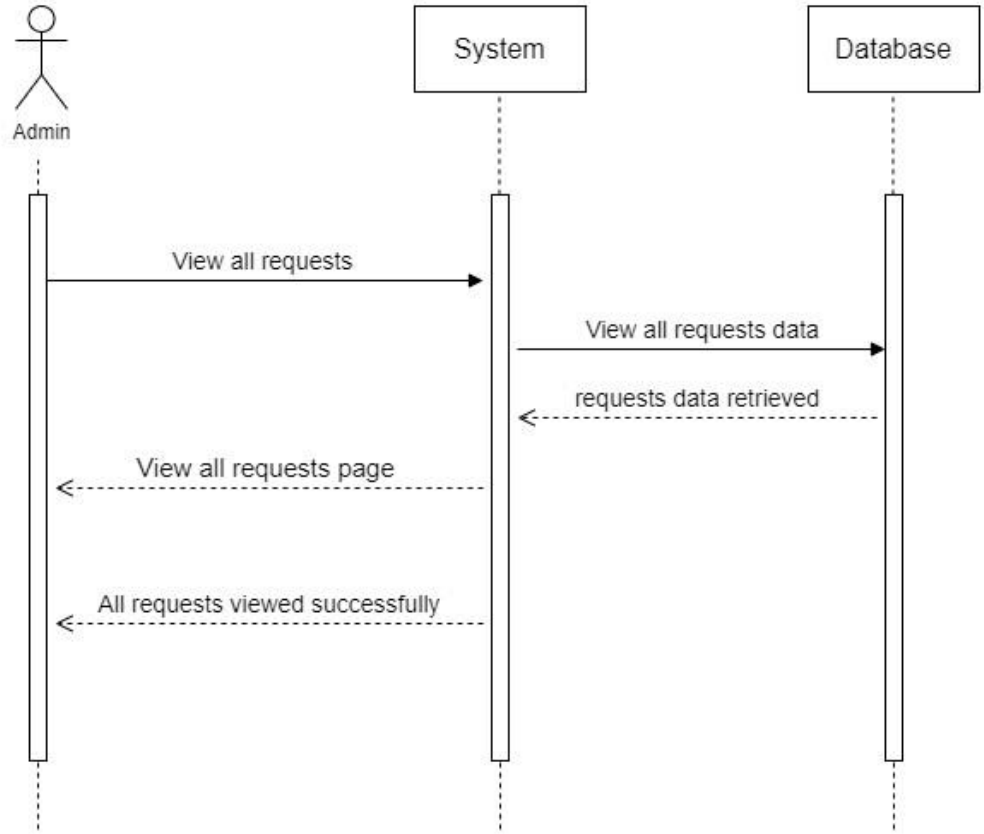
**Get all books chaprers**

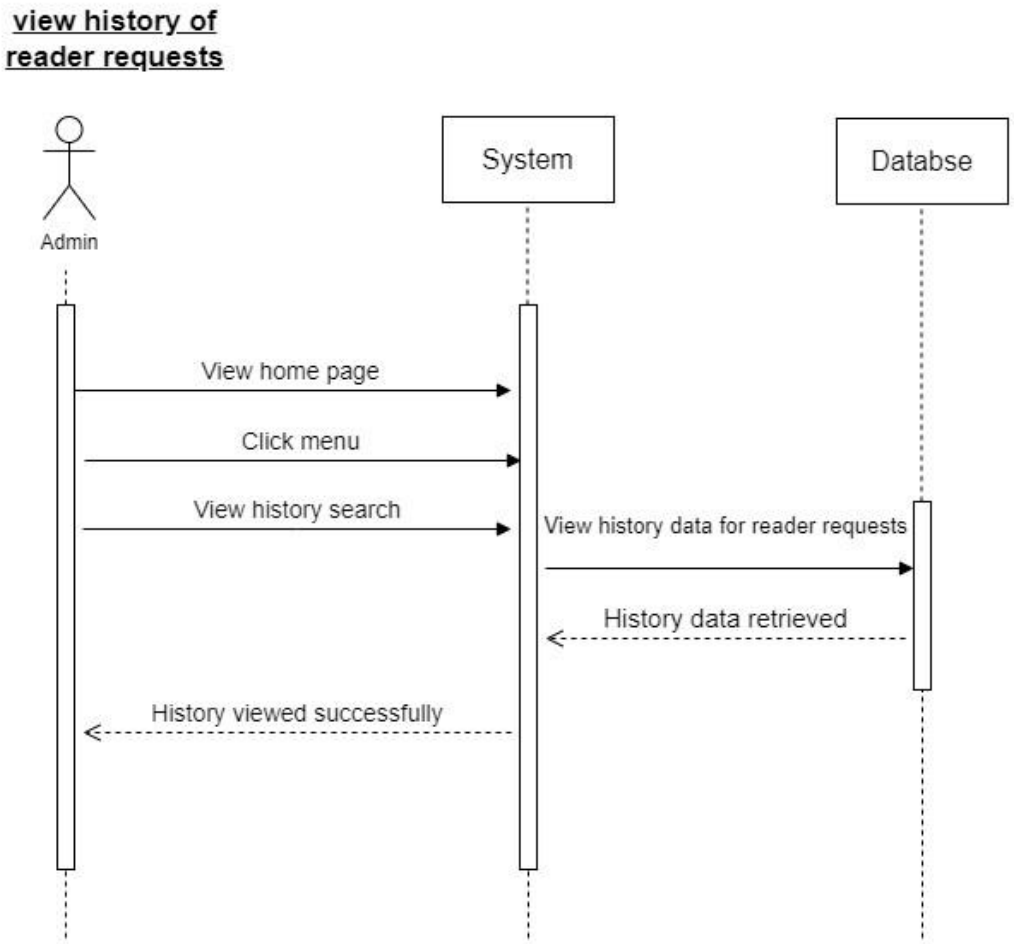


**Get all Readers**

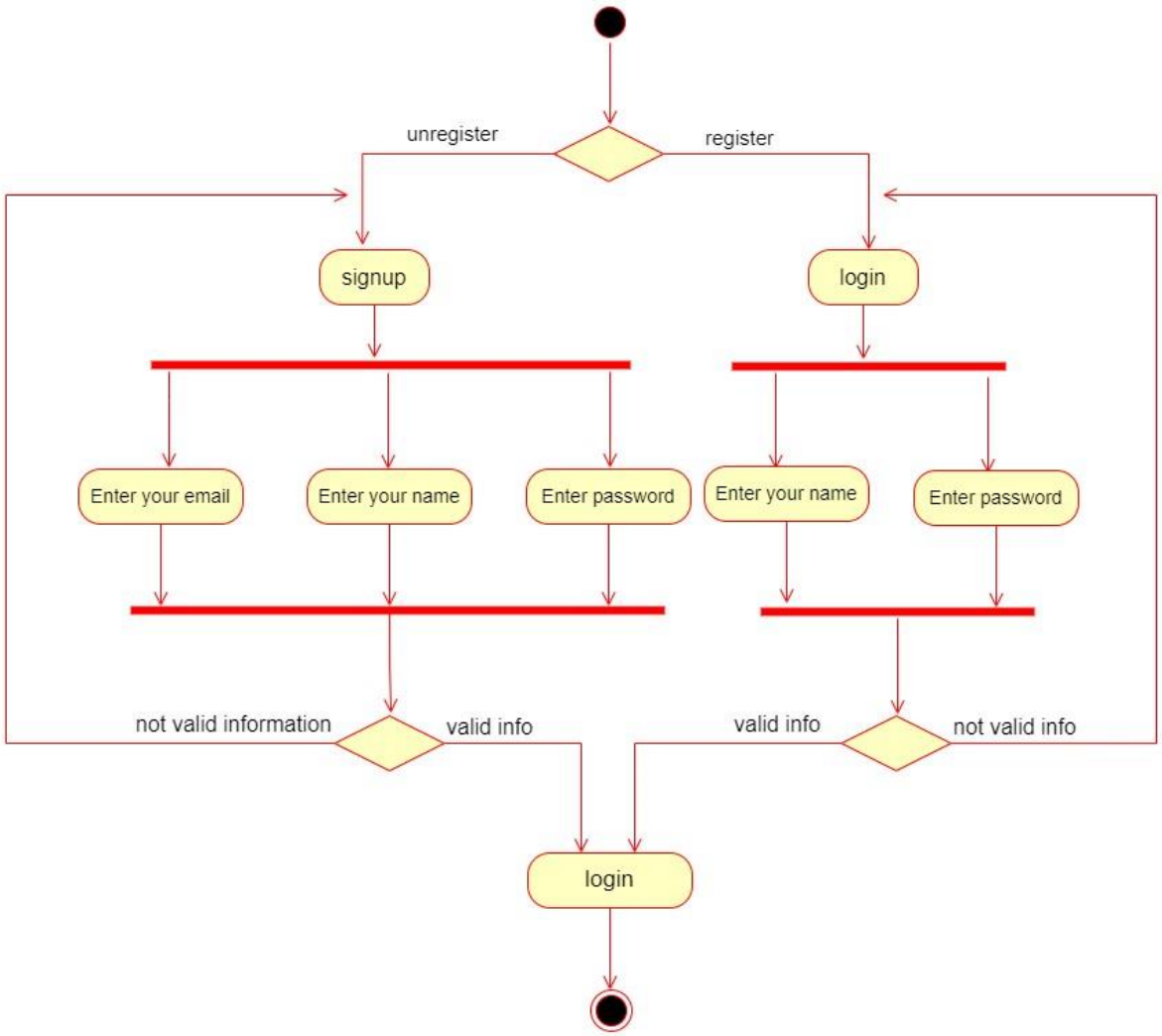


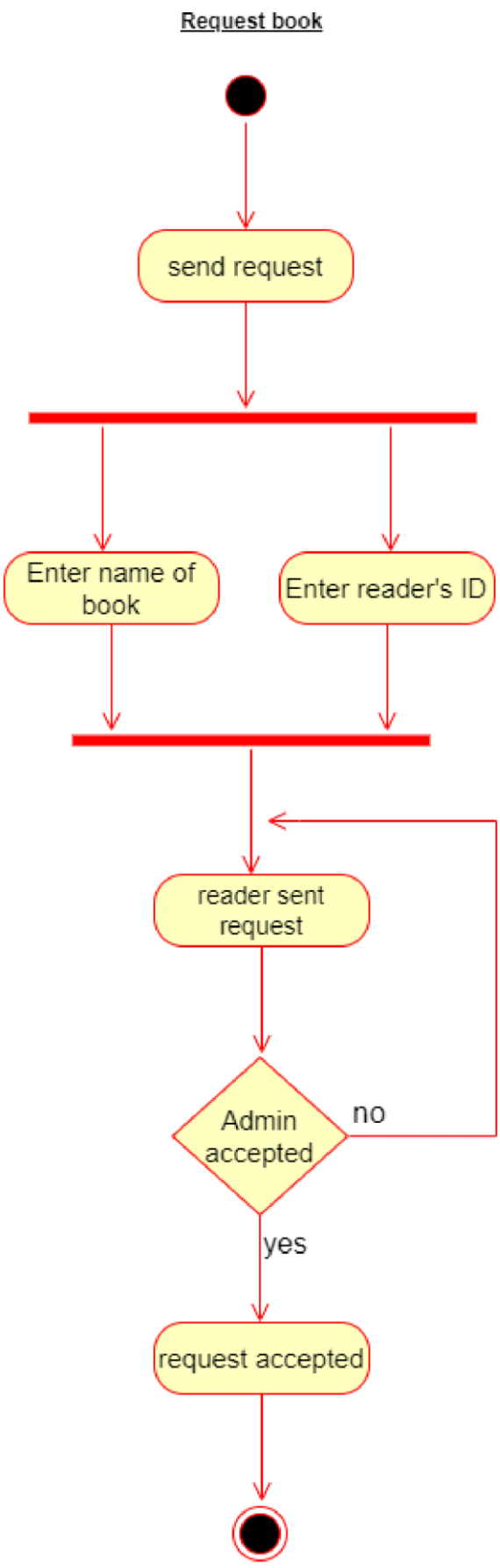
**Get all requests**





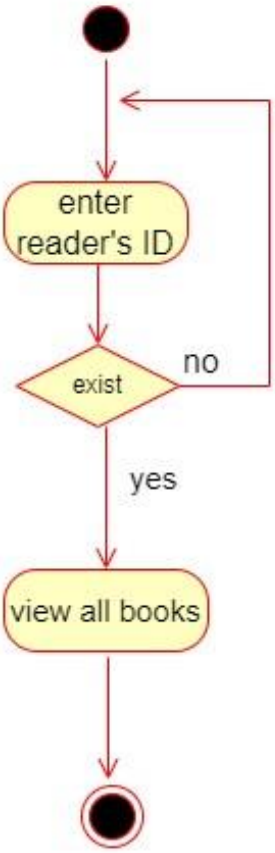

**Activity diagram** for reader:


unregister
register
signup
login
Enter your email
Enter your name
Enter password
Enter your name
Enter password
not valid information
valid info
valid info
not valid info
login


Request book
send request
Enter name of book
Enter reader's ID
reader sent request
Admin accepted
no
yes
request accepted

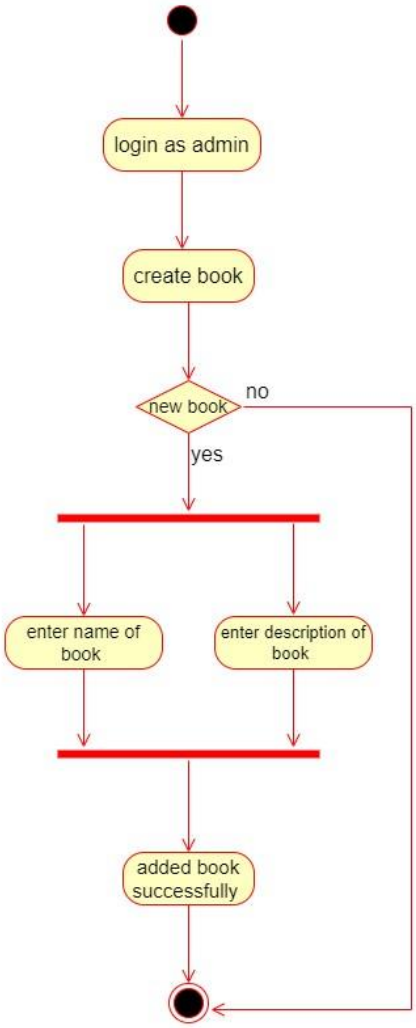
**View book**

**Get all books**

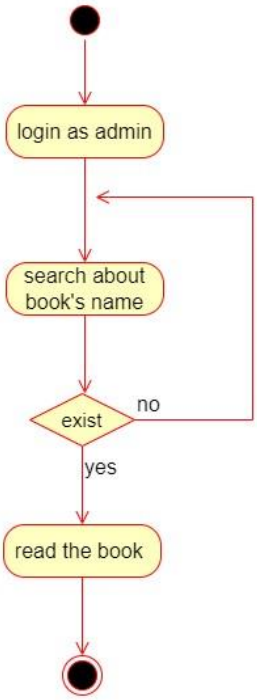



For admin

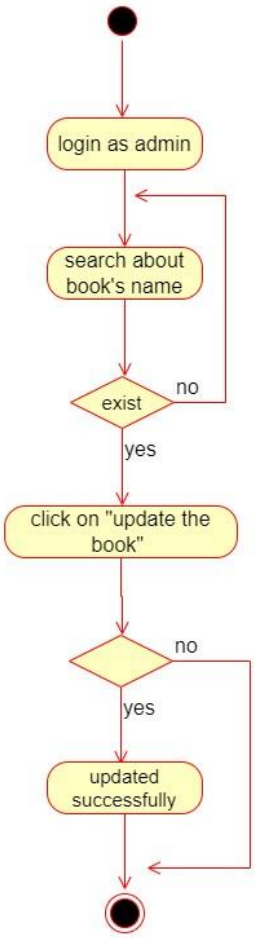
**create book**



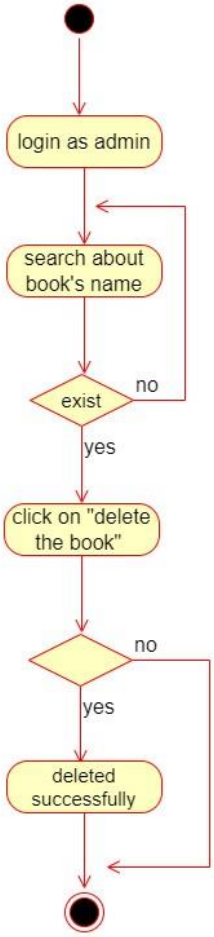

**read book**



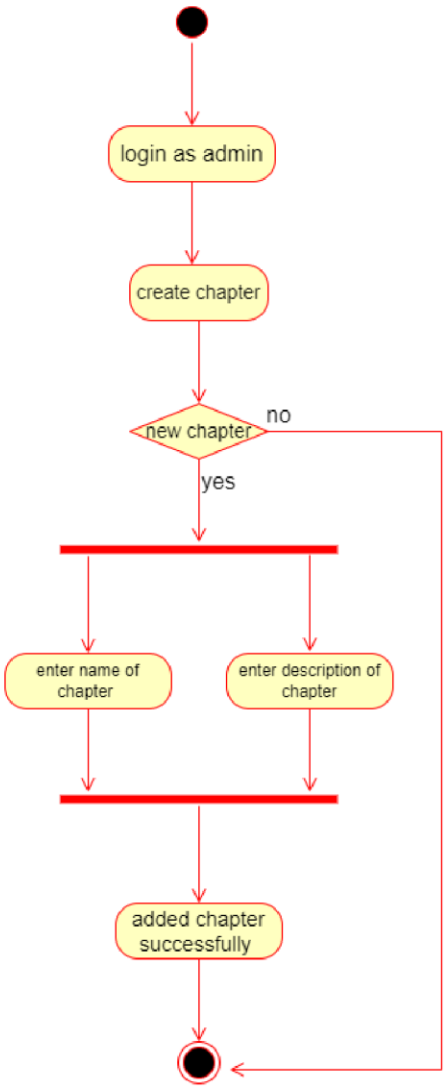
**update book**



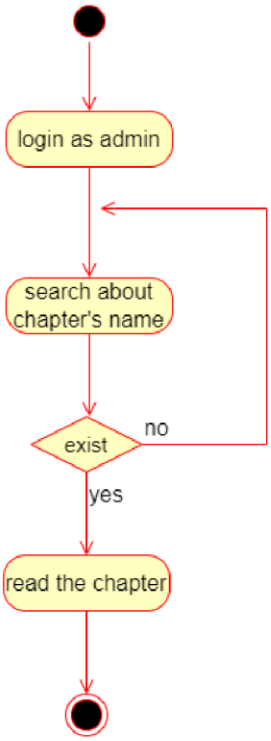

**Delete book**



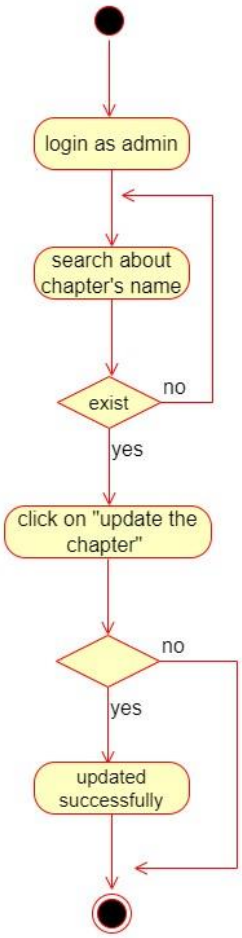
**create chapter**



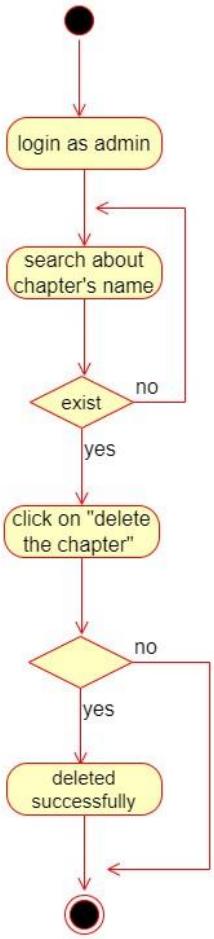

**read chapter**



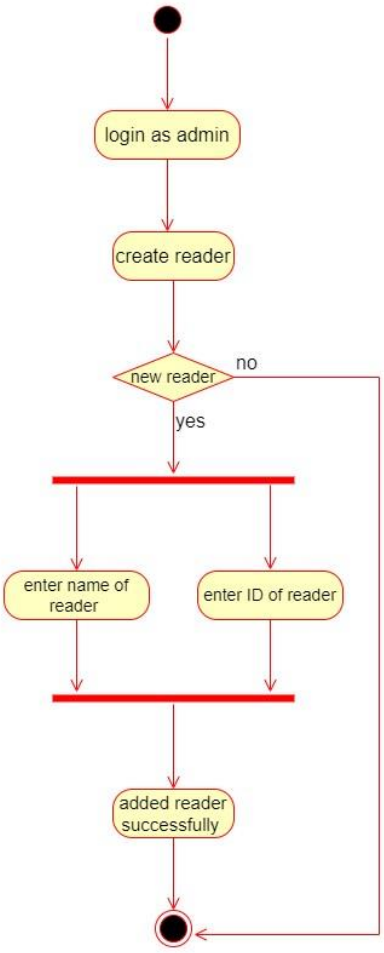
**update chapter**



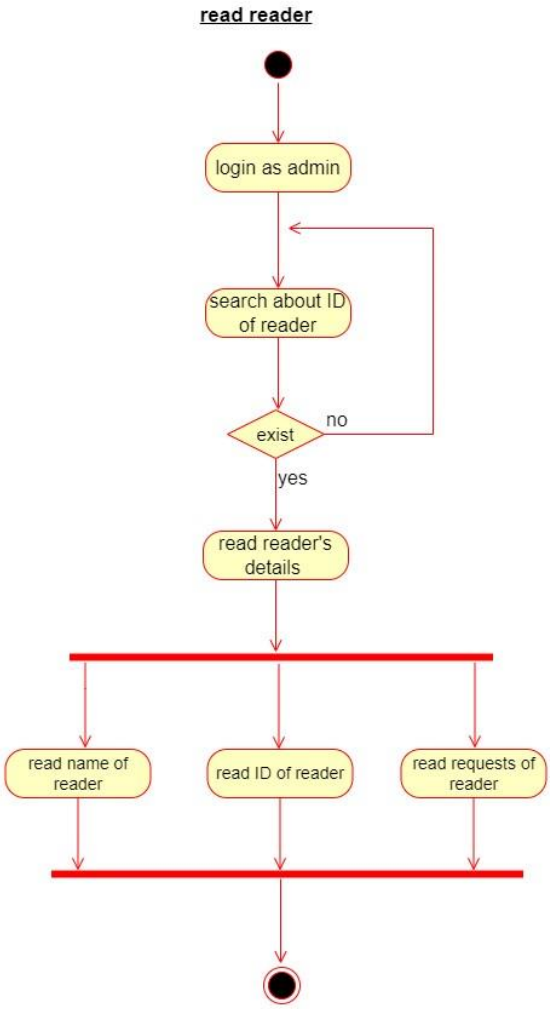

**Delete book**



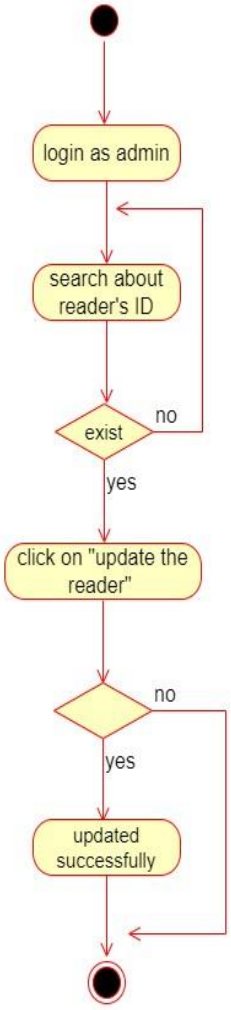
**create reader**



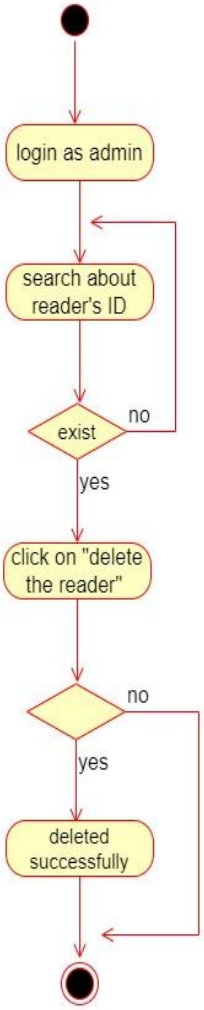

**read reader**



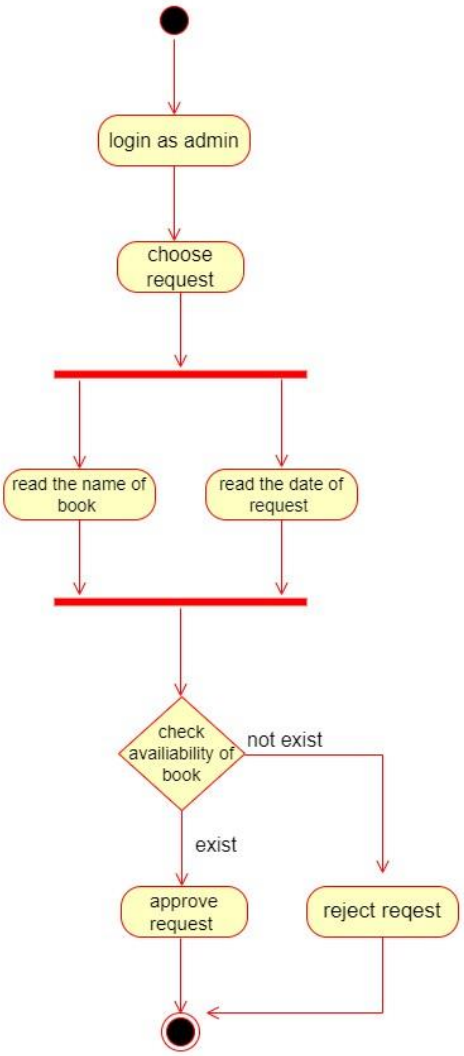
**update reader**



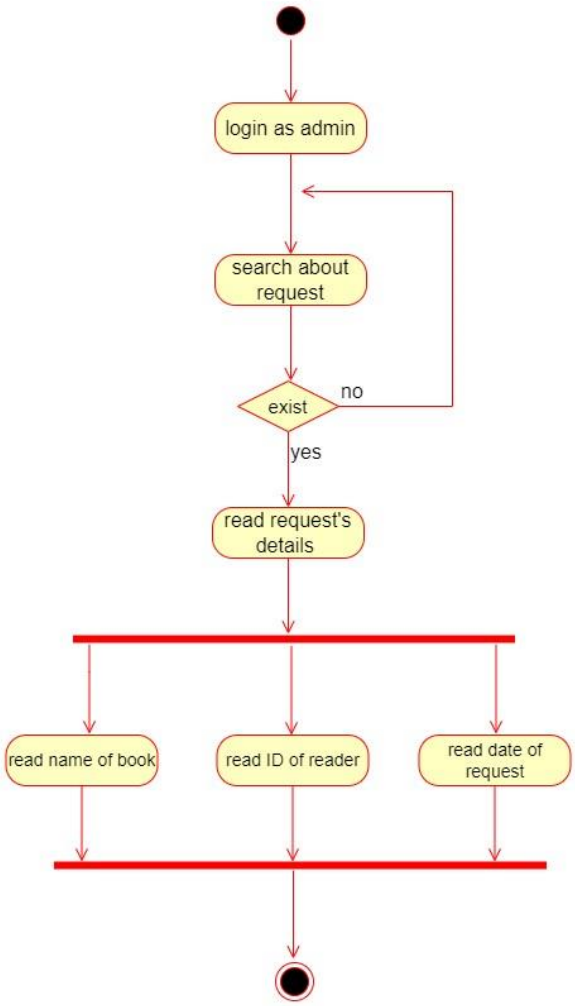

**Delete reader**

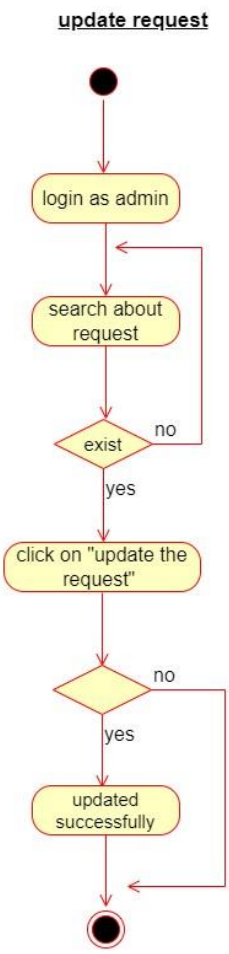


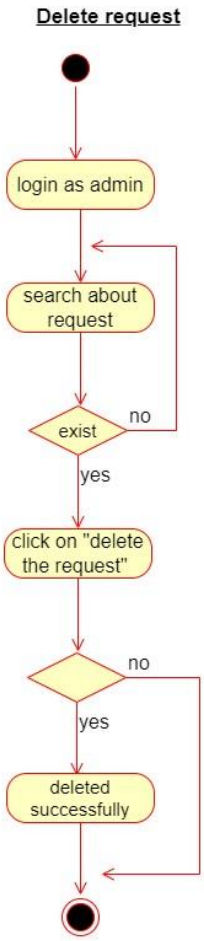
**create request**




**read request**

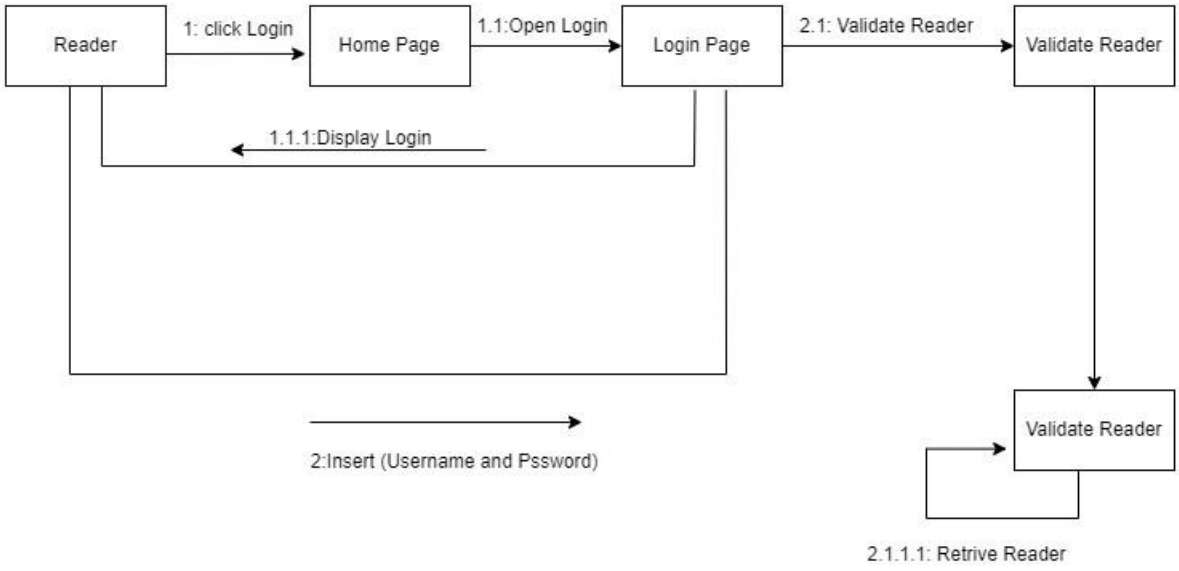




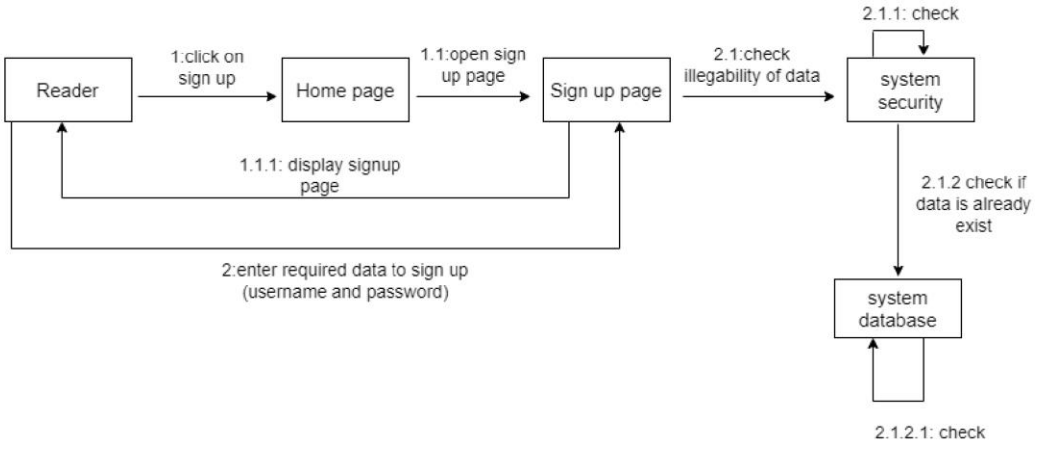




Communication diagram For reader:

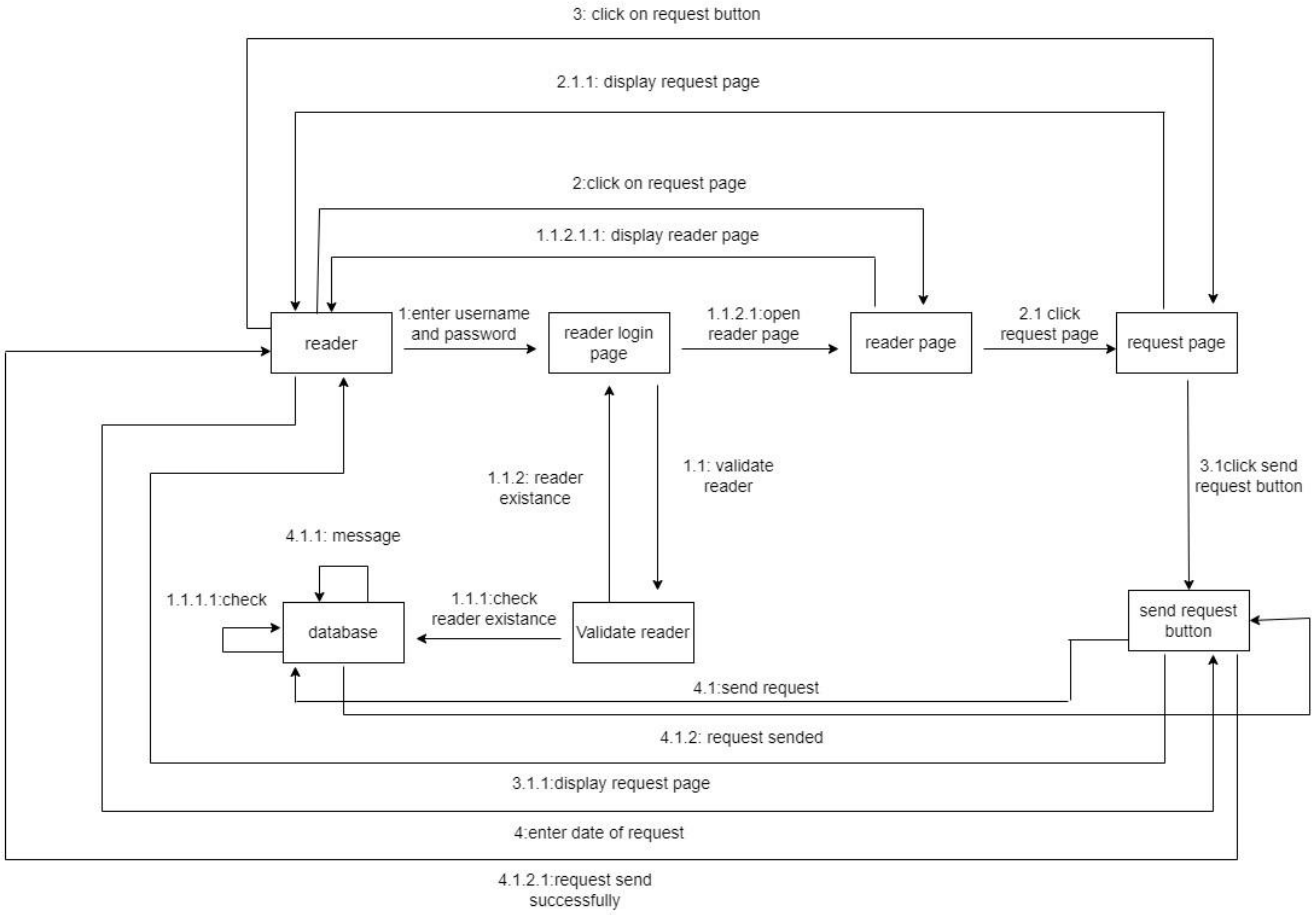
## Login



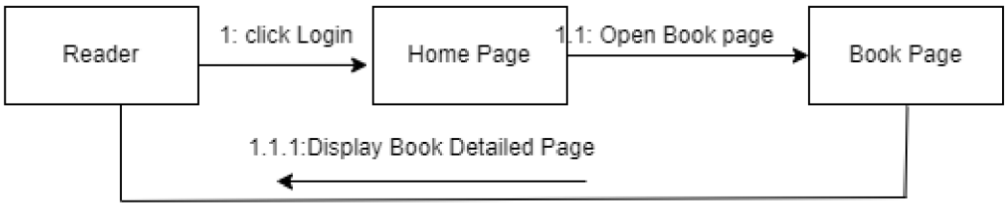

**sign up**



**send request**

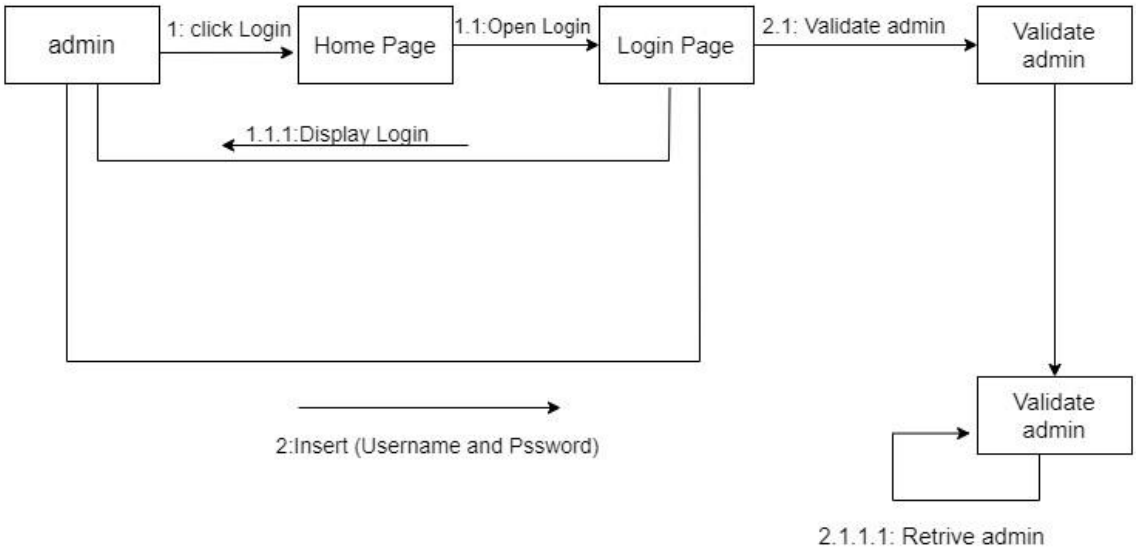



## View Book

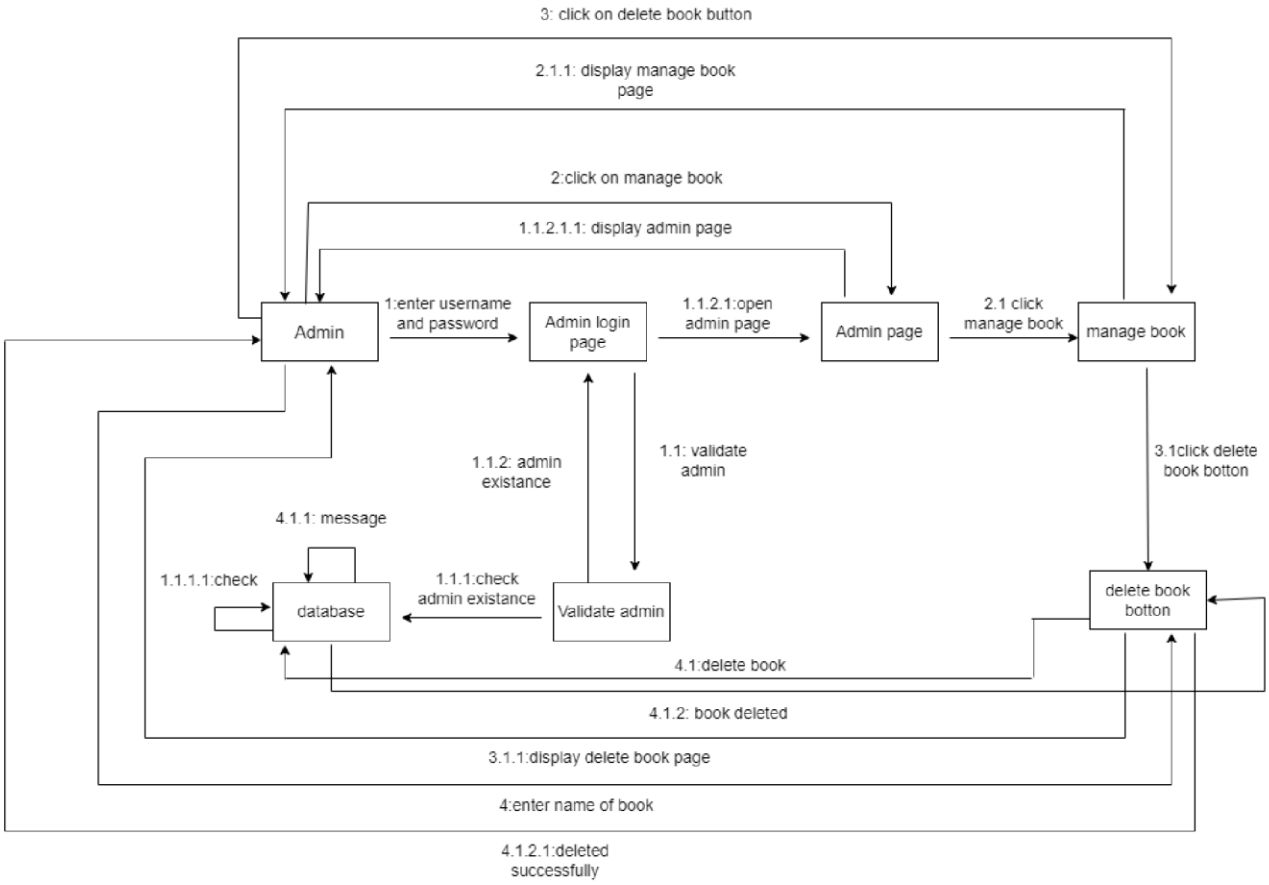


# For admin

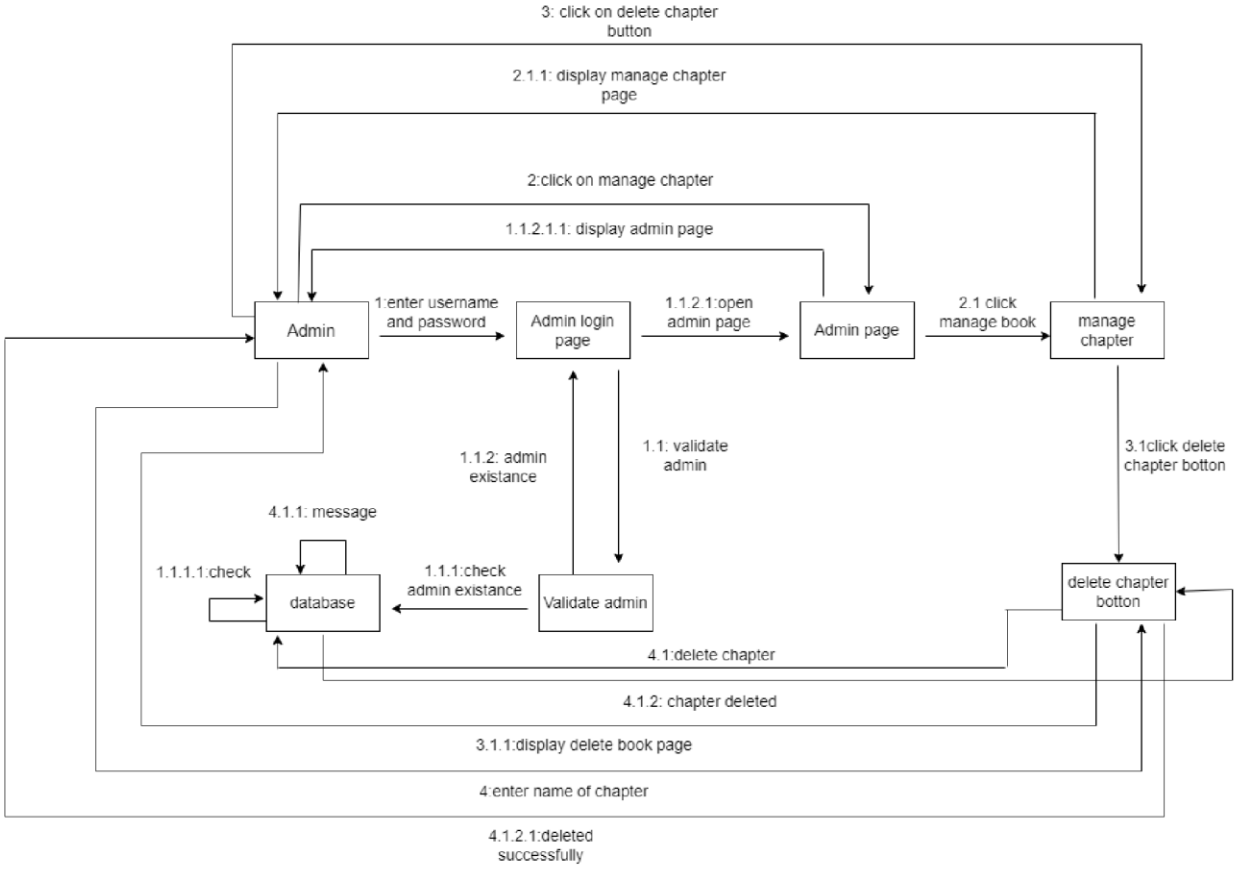
**Login**



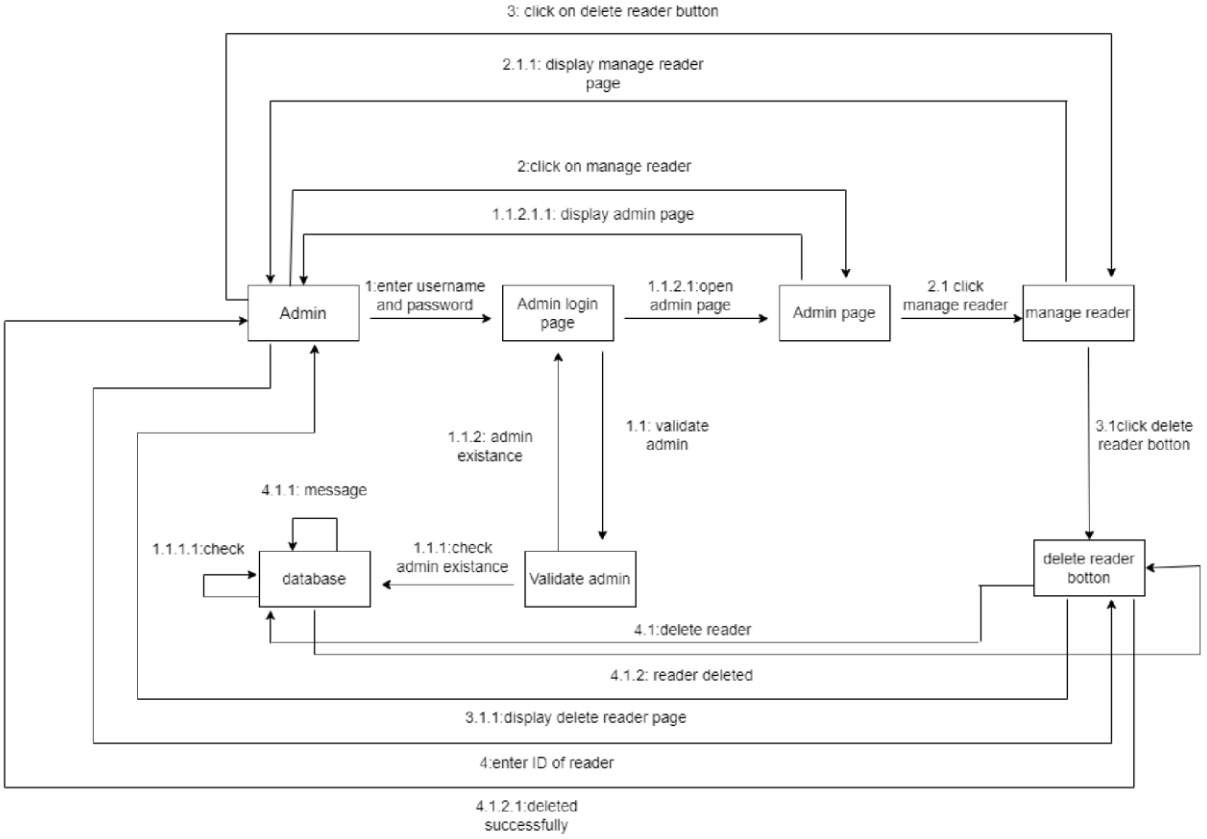
**delete book**


3: click on delete book button
2.1.1: display manage book page
2:click on manage book
1.1.2.1.1: display admin page
Admin
1:enter username and password
Admin login page
1.1.2.1:open admin page
Admin page
2.1 click manage book
manage book
1.1.2: admin existance
1.1: validate admin
3.1click delete book botton
4.1.1: message
1.1.1.1:check
database
1.1.1:check admin existance
Validate admin
delete book botton
4.1:delete book
4.1.2: book deleted
3.1.1:display delete book page
4:enter name of book
4.1.2.1:deleted successfully

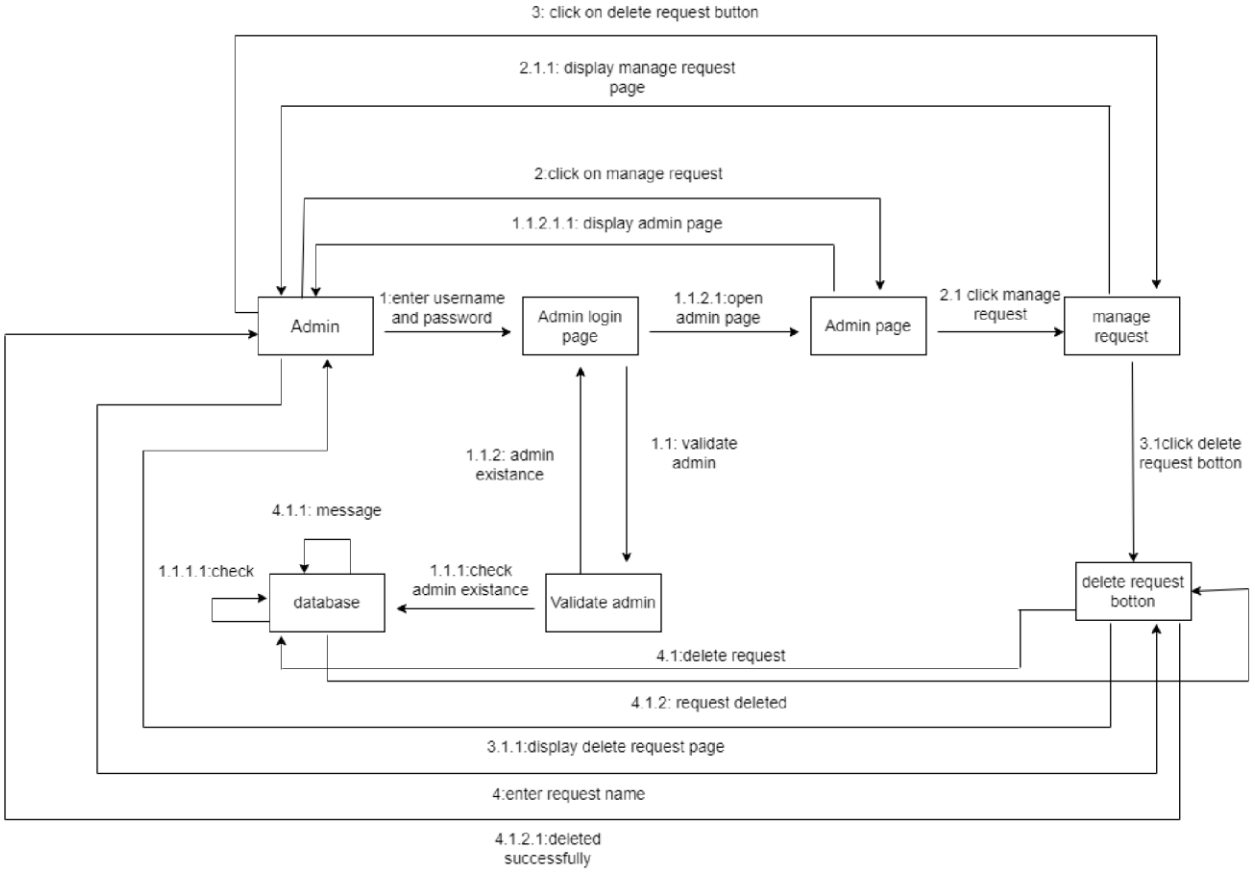
**delete chapter**



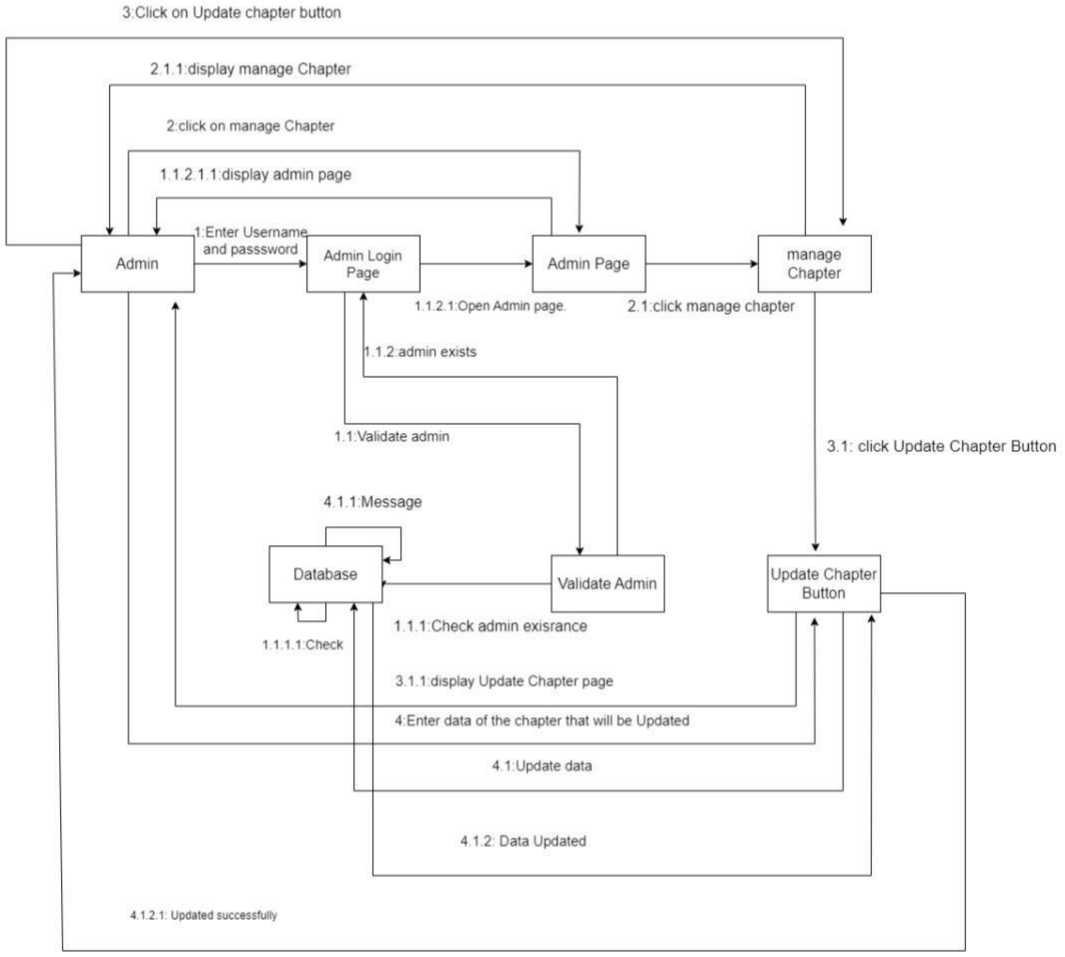
**delete reader**



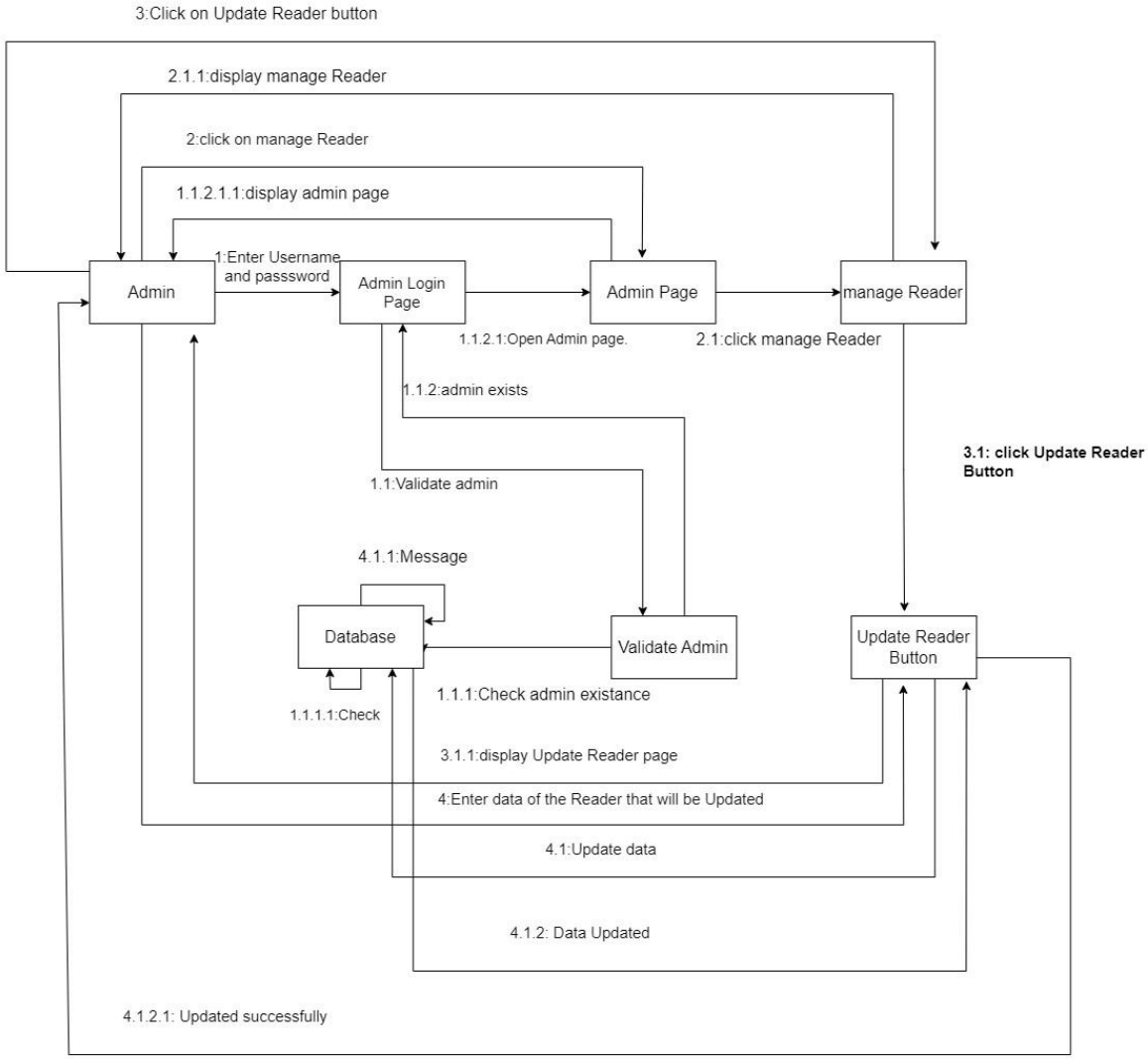
**delete request**



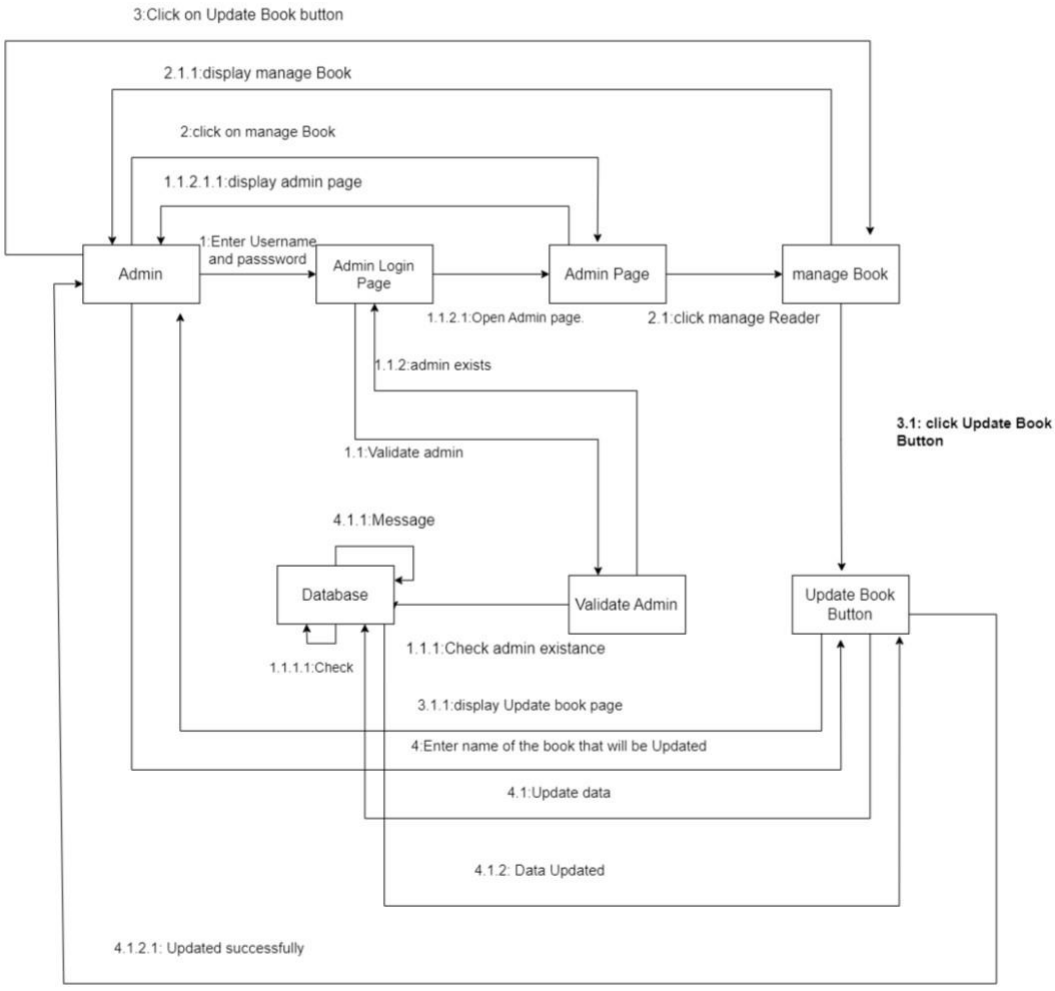
**Update Chapter**



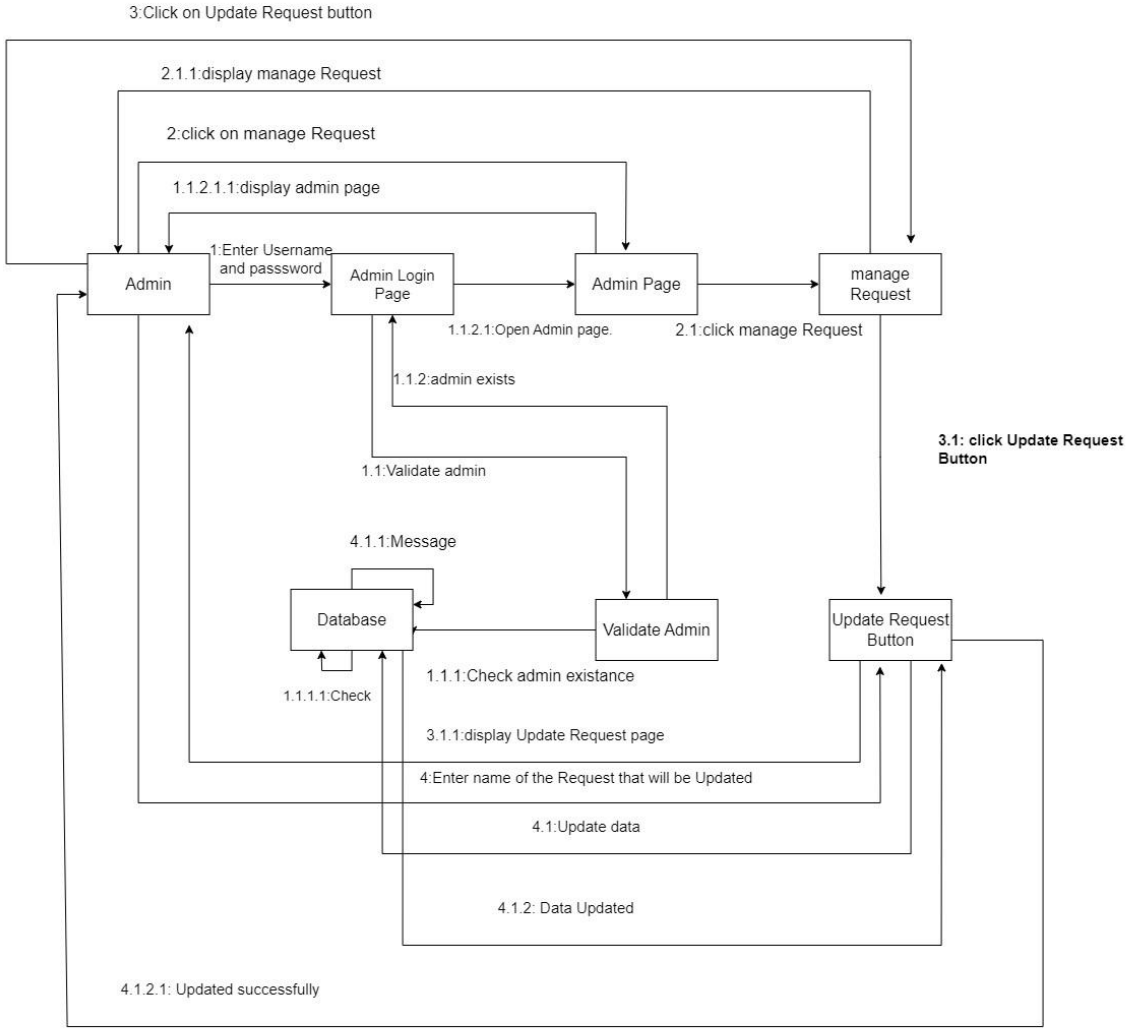
**Update Reader**



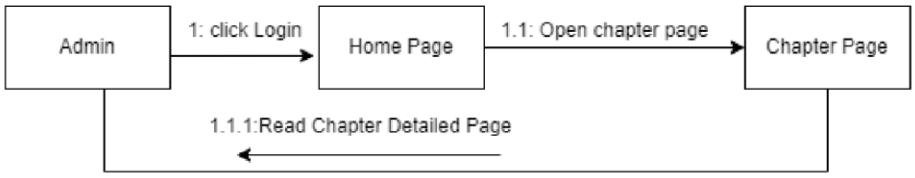
**Update Book**


3:Click on Update Book button
2.1.1:display manage Book
2:click on manage Book
1.1.2.1.1:display admin page
1:Enter Username and passsword
Admin
Admin Login Page
Admin Page
manage Book
1.1.2.1:Open Admin page.
2.1:click manage Reader
1.1.2:admin exists
3.1: click Update Book Button
1.1:Validate admin
4.1.1:Message
Database
Validate Admin
Update Book Button
1.1.1:Check admin existance
1.1.1.1:Check
3.1.1:display Update book page
4:Enter name of the book that will be Updated
4.1:Update data
4.1.2: Data Updated
4.1.2.1: Updated successfully

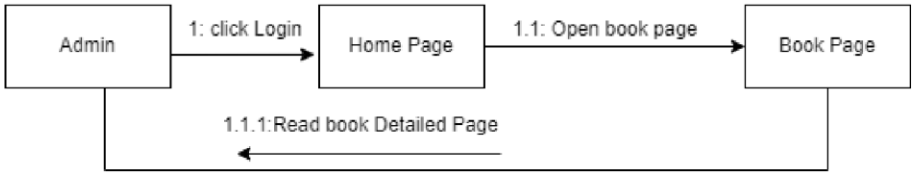
**Update Request**



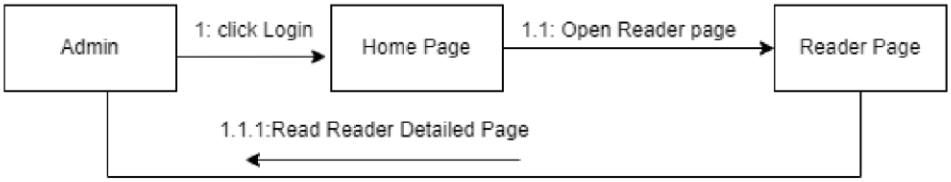
**Read Chapter**



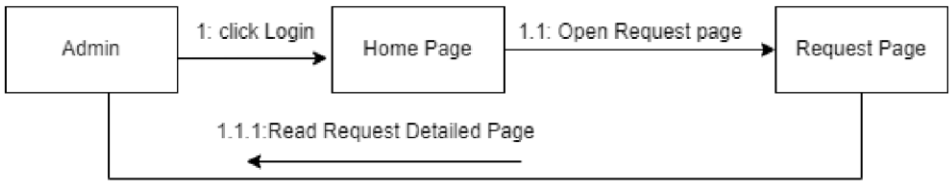

**Read Book**



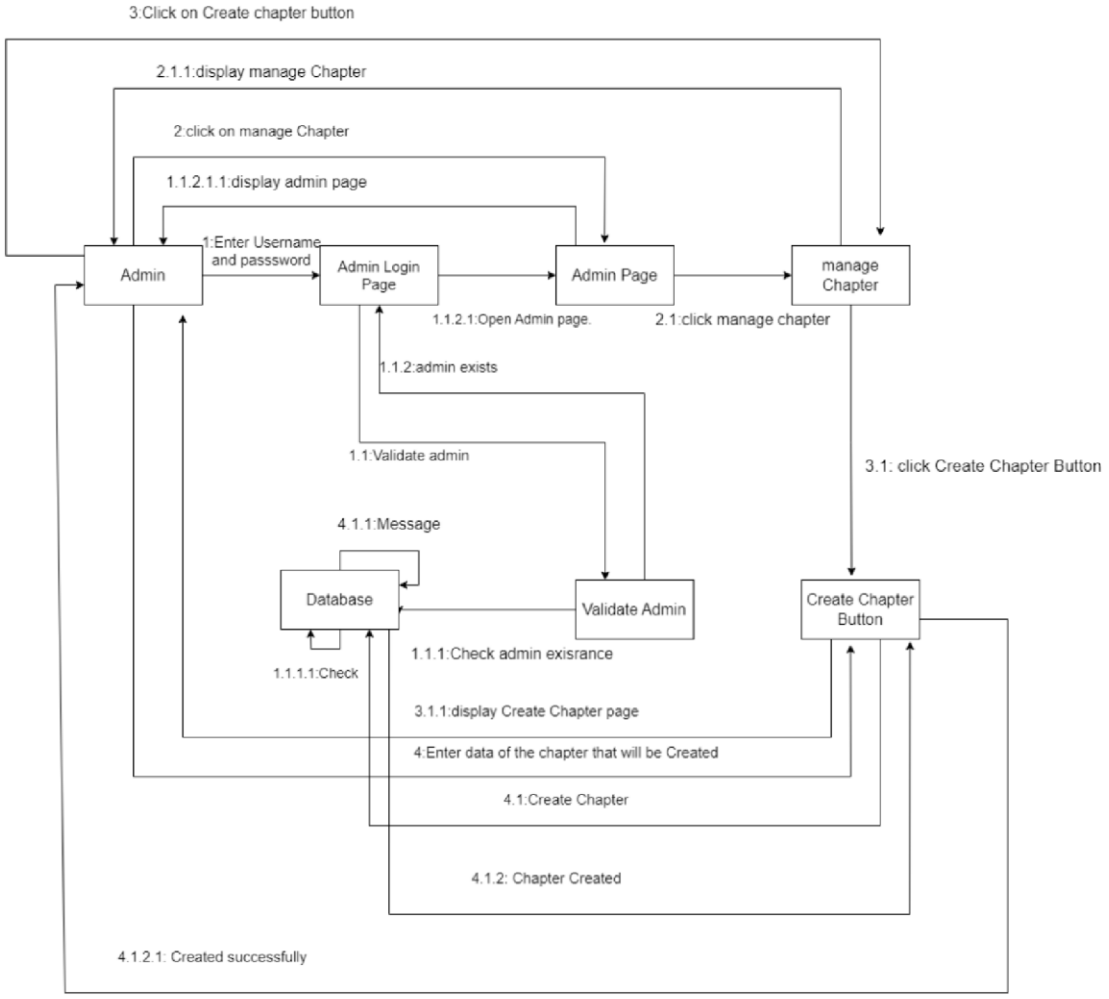
**Read Reader**



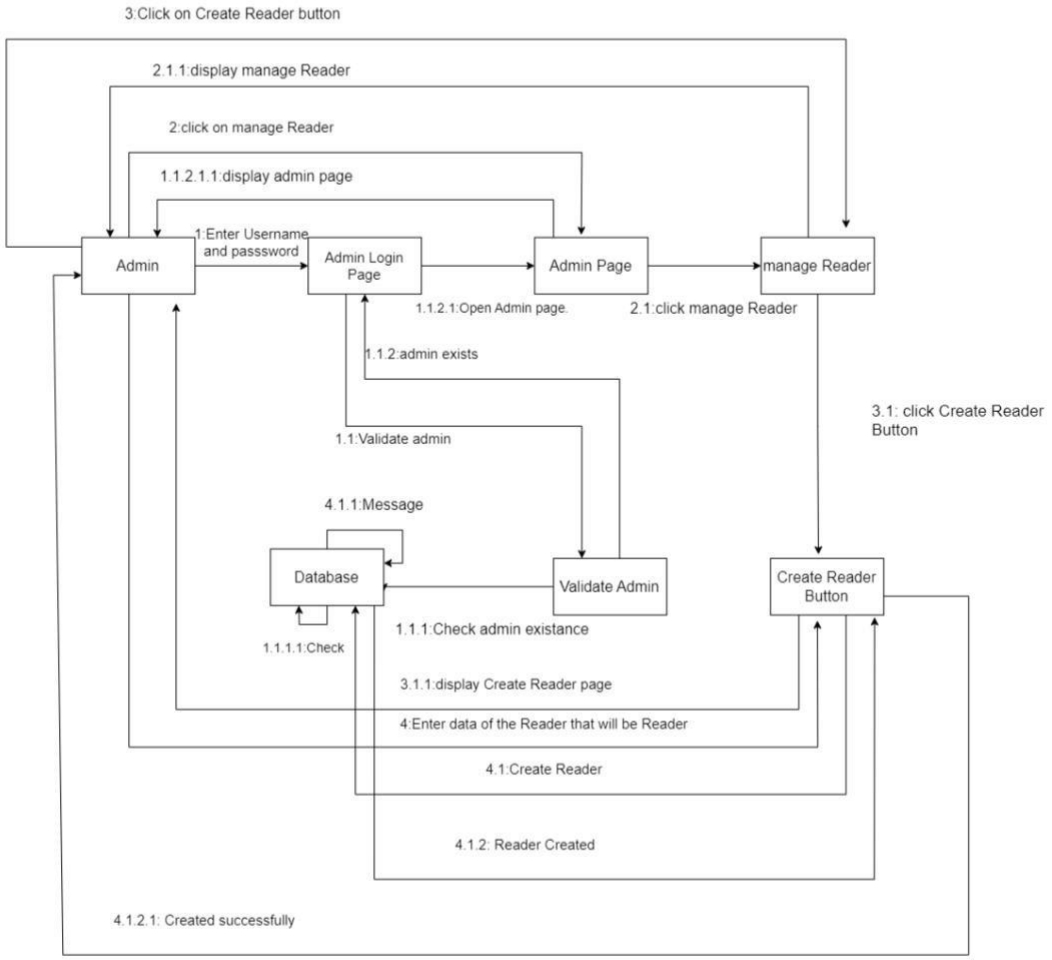

**Read Request**



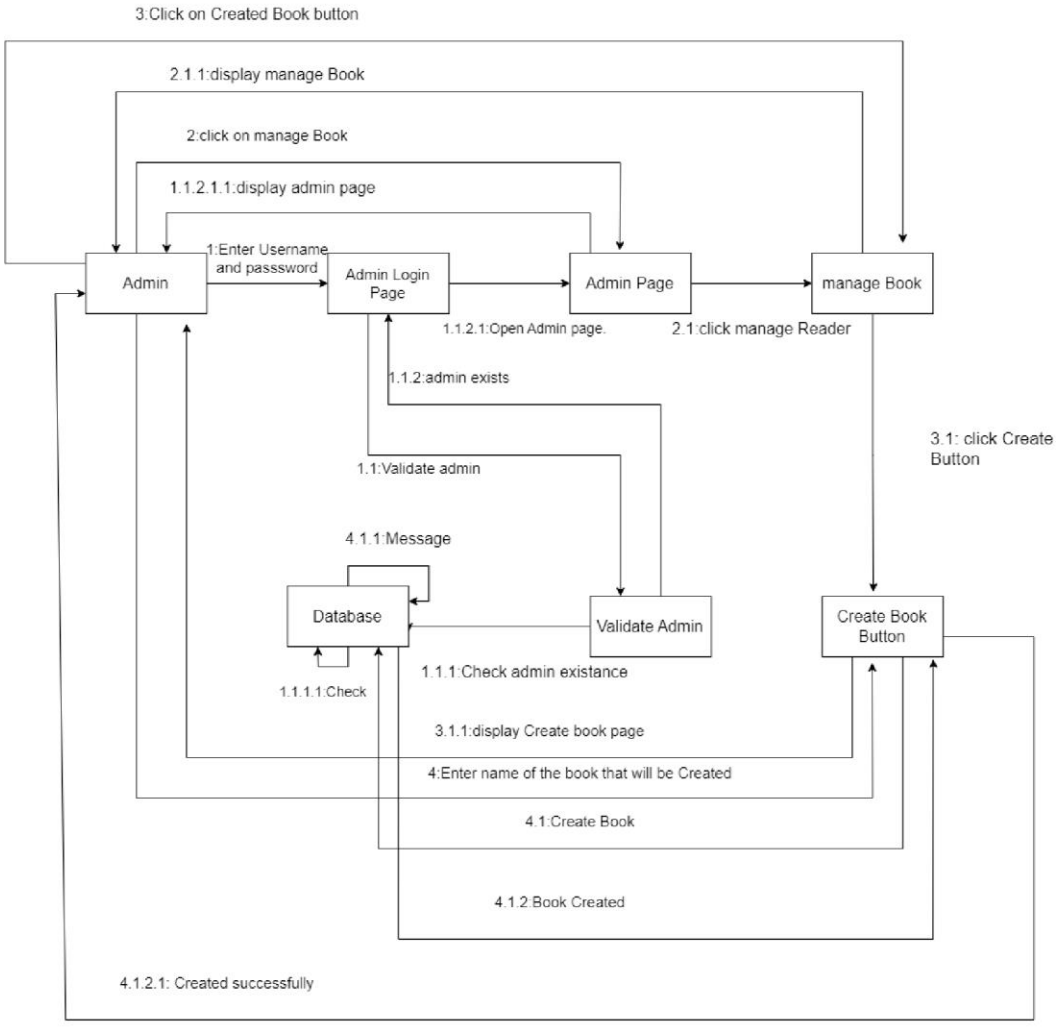
**Create Chapter**

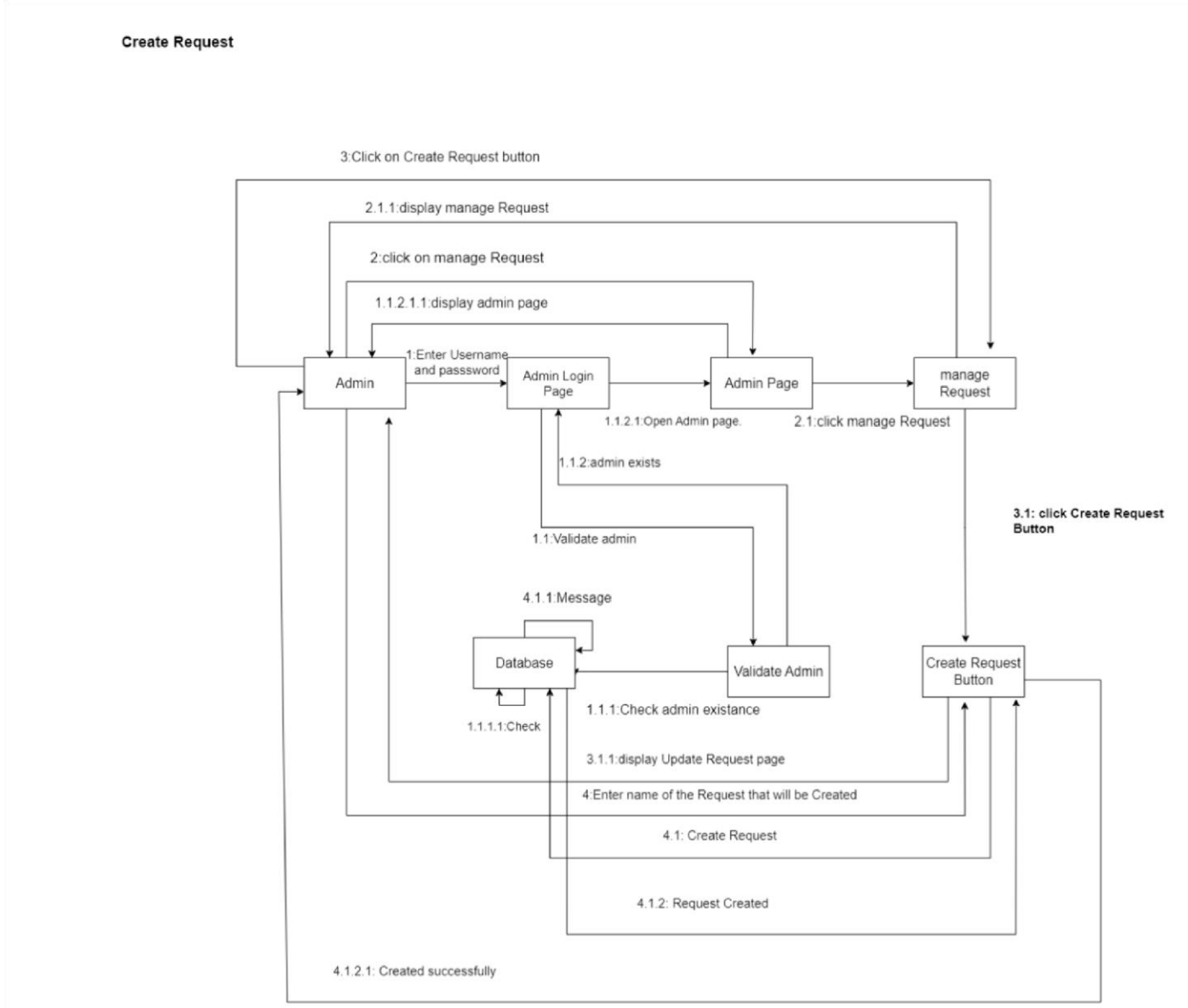


**Create Reader**



**Create Book**

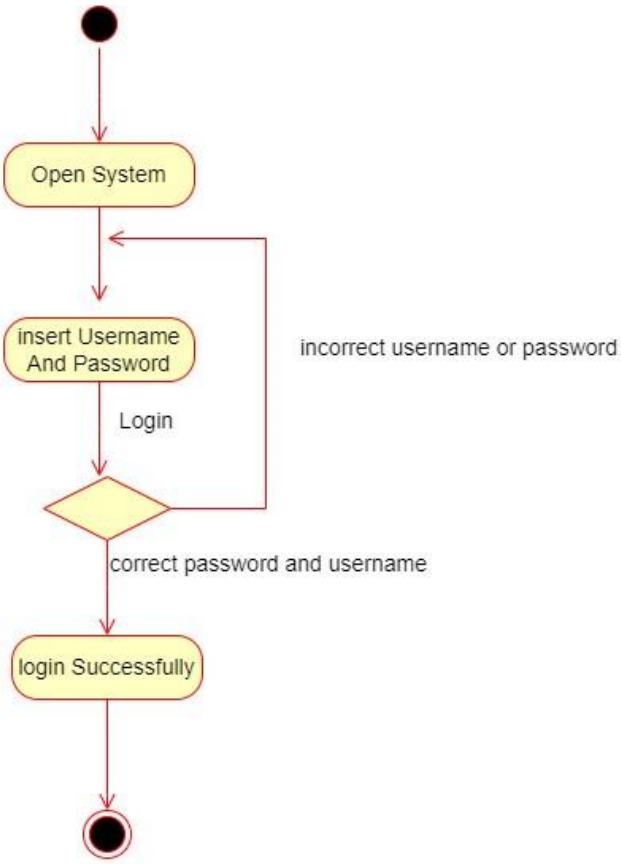

3:Click on Created Book button
2.1.1:display manage Book
2:click on manage Book
1.1.2.1.1:display admin page
1:Enter Username and passsword
Admin
Admin Login Page
Admin Page
manage Book
1.1.2.1:Open Admin page.
2.1:click manage Reader
1.1.2:admin exists
3.1: click Create Book Button
1.1:Validate admin
4.1.1:Message
Database
Validate Admin
Create Book Button
1.1.1:Check admin existance
1.1.1.1:Check
3.1.1:display Create book page
4:Enter name of the book that will be Created
4.1:Create Book
4.1.2:Book Created
4.1.2.1: Created successfully



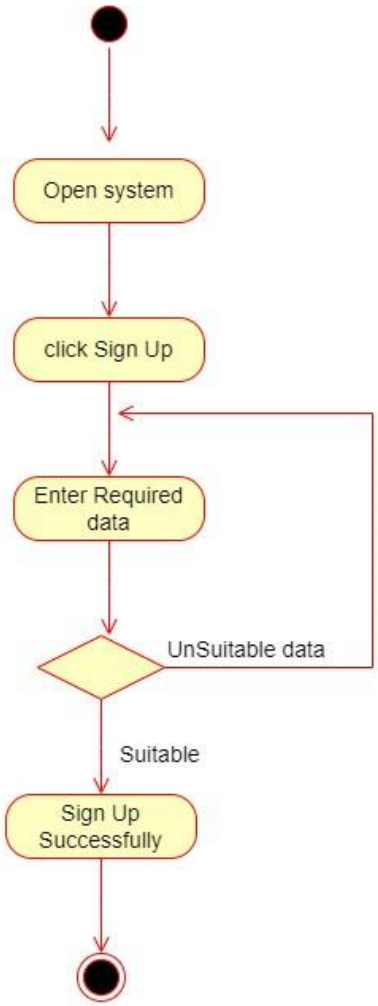

State diagram For

reader:

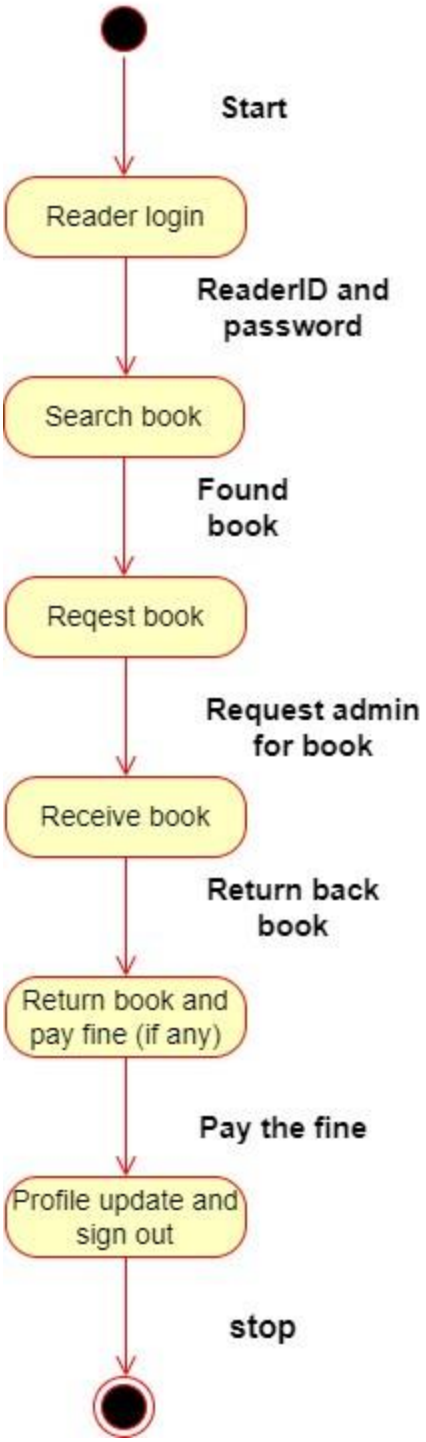
**login**

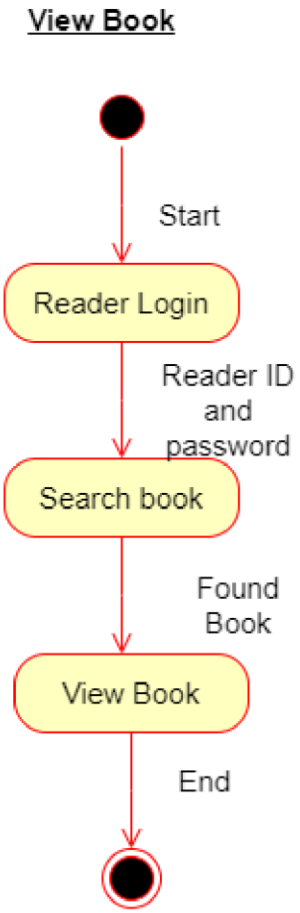



**Sign Up**



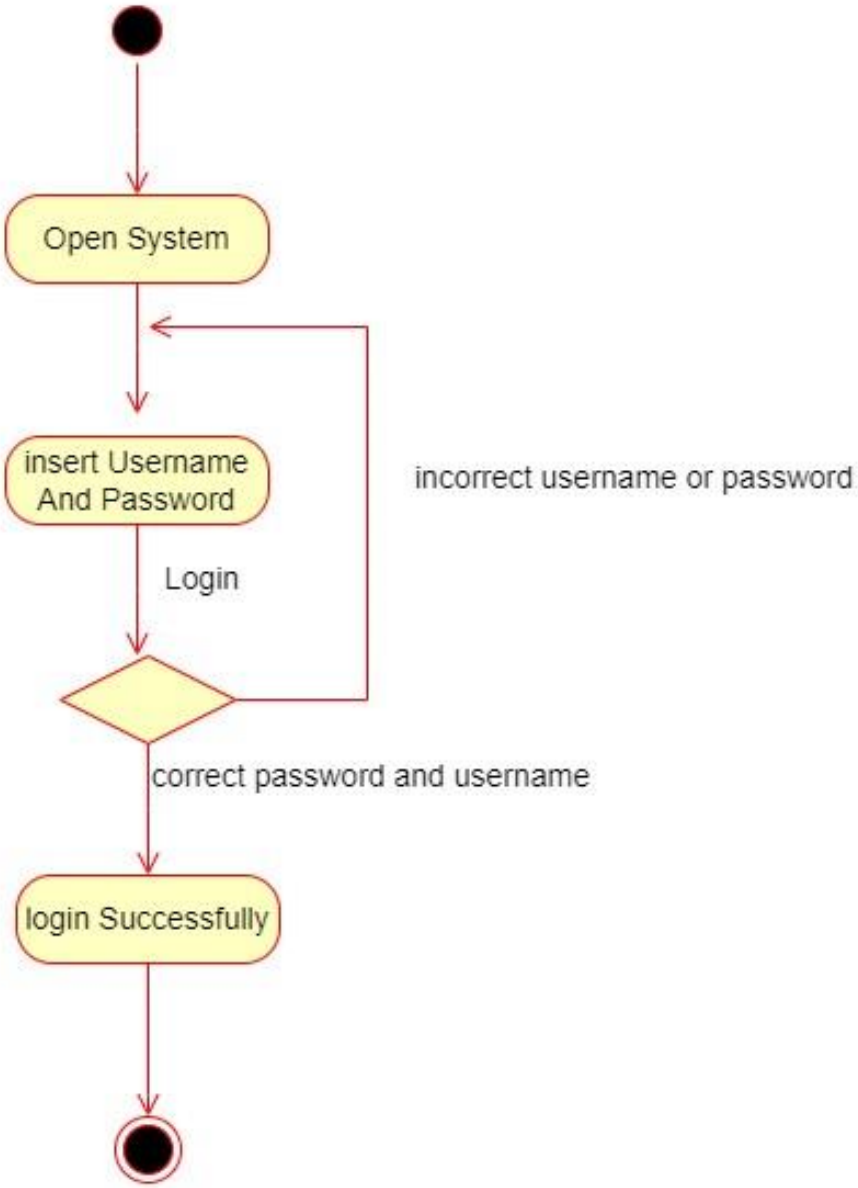
**send request**



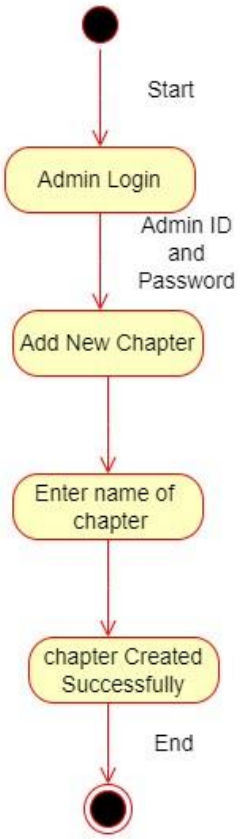

View Book
Start
Reader Login
Reader ID
and
password
Search book
Found
Book
View Book
End

For admin:

**login**

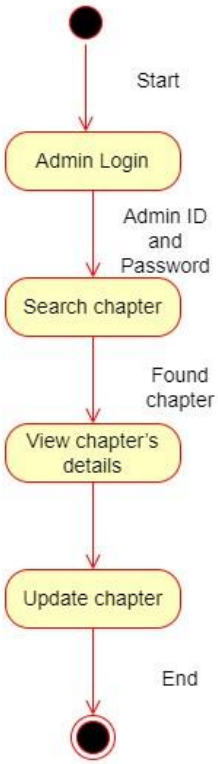

Open System
insert Username
And Password
incorrect username or password
Login
correct password and username
login Successfully

**Create chapter**

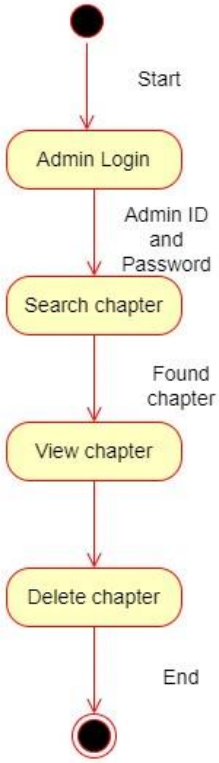



**Read chapter**

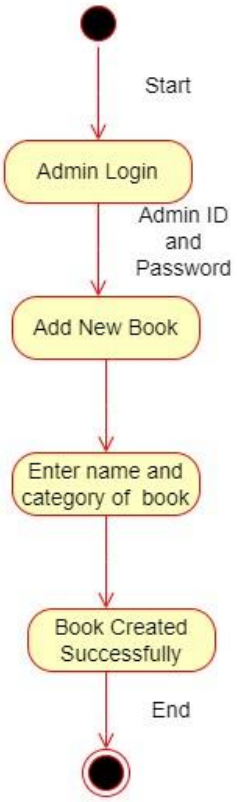
**Update chapter**



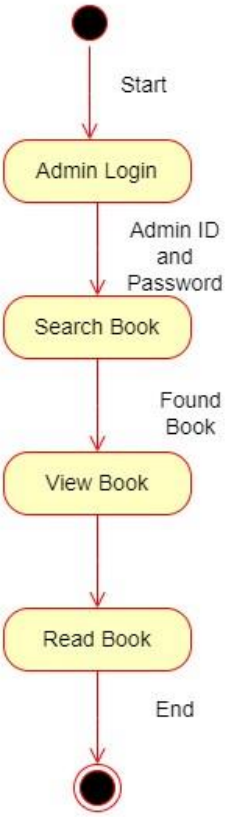

**Delete chapter**



**Create Book**

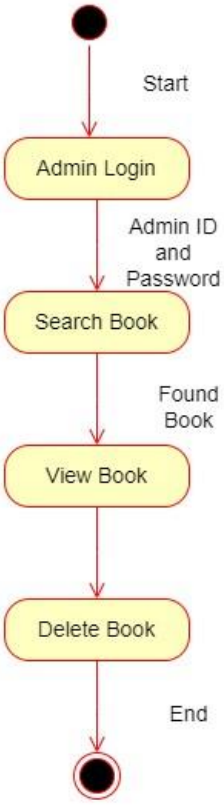



**Read book**

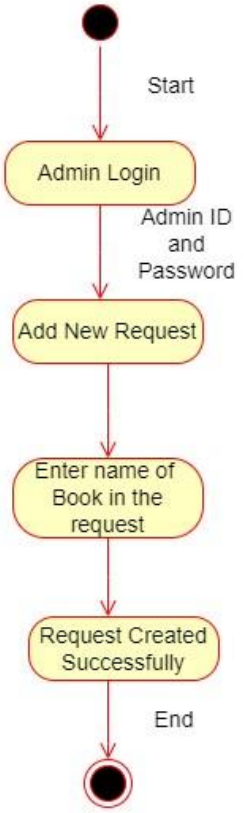


**Update book**

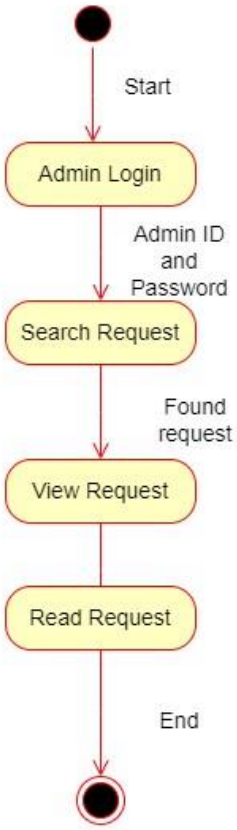
**Delete book**



**Create Request**

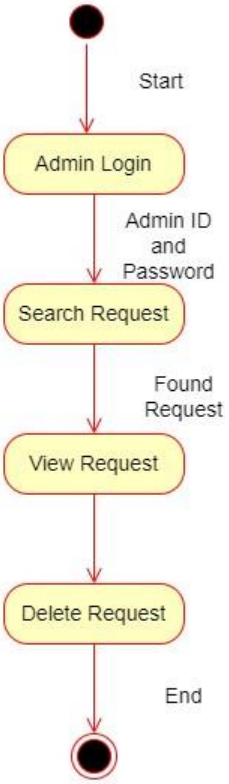



**Read Request**

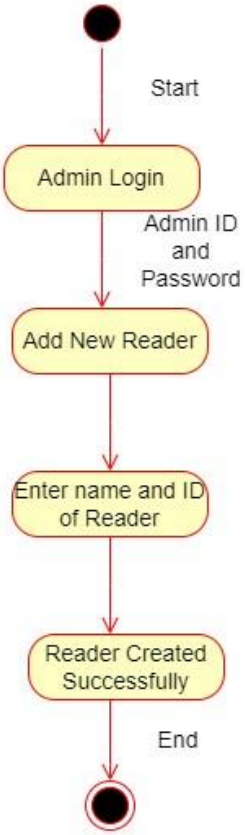


**Update request**

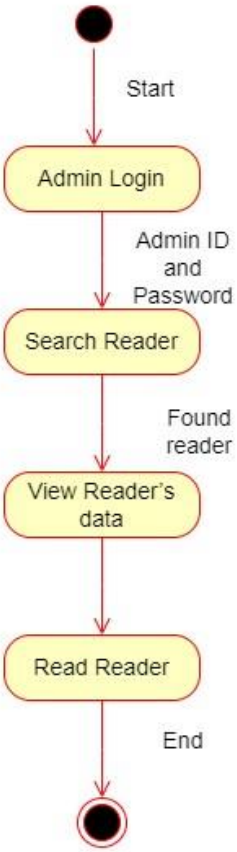
**Delete request**

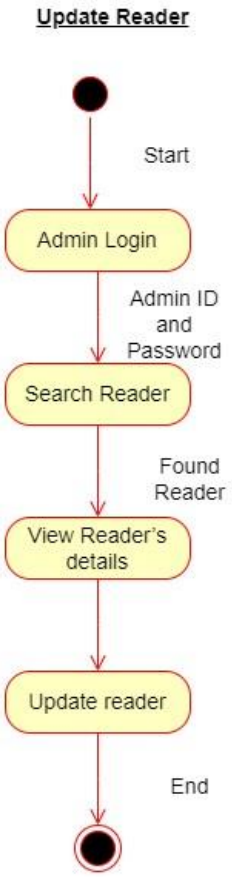


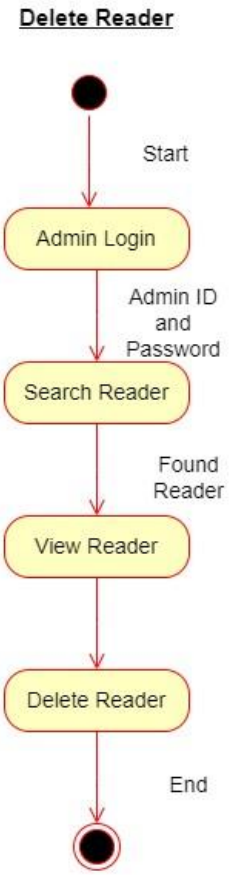
**Create Reader**



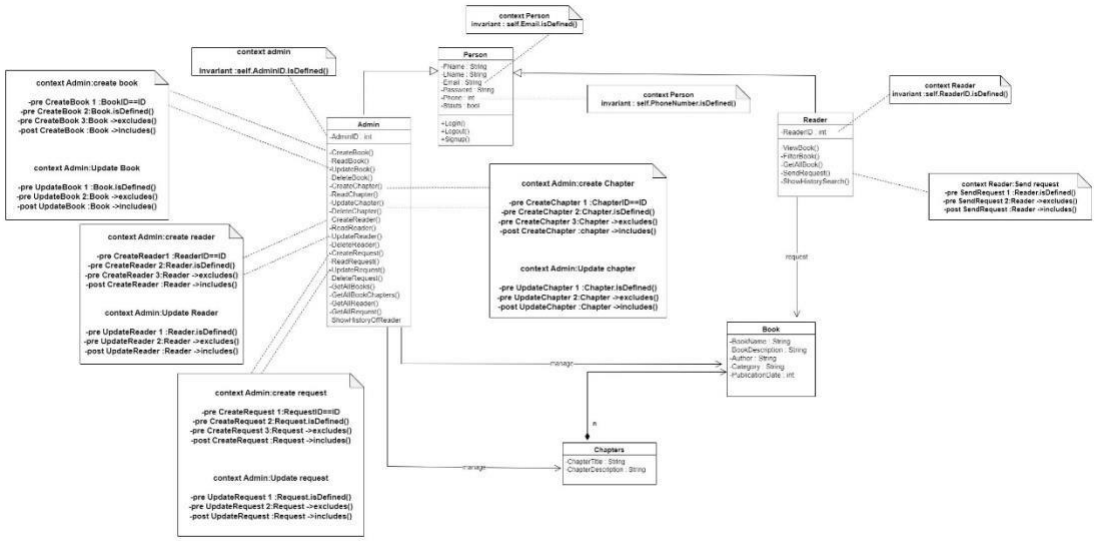

**Read Reader**




Update Reader
Start
Admin Login
Admin ID and Password
Search Reader
Found Reader
View Reader's details
Update reader
End


Delete Reader
Start
Admin Login
Admin ID and Password
Search Reader
Found Reader
View Reader
Delete Reader
End

# OCL diagram:


context Person
invariant : self.Email.isDefined()
context admin
invariant :self.AdminID.isDefined()
Person
-Fname : String
-Lname : String
-Email : String
-Password : String
-Phone : int
-Status : bool
+Login()
+Logout()
+Signup()
context Person
invariant : self.PhoneNumber.isDefined()
context Reader
invariant :self.ReaderID.isDefined()
Reader
-ReaderID : int
-ViewBook()
-FilterBook()
-GetAllBook()
-SendRequest()
-BrowseHistorySearch()
context Reader:Send request
-pre SendRequest 1 :Reader.isDefined()
-pre SendRequest 2:Reader ->excludes()
-post SendRequest :Reader ->includes()
context Admin:create book
-pre CreateBook 1 :BookID==ID
-pre CreateBook 2:Book.isDefined()
-pre CreateBook 3:Book ->excludes()
-post CreateBook :Book ->includes()
context Admin:Update Book
-pre UpdateBook 1 :Book.isDefined()
-pre UpdateBook 2:Book ->excludes()
-post UpdateBook :Book ->includes()
Admin
-AdminID : int
-CreateBook()
-ReadBook()
-UpdateBook()
-DeleteBook()
-CreateChapter()
-ReadChapter()
-UpdateChapter()
-DeleteChapter()
-CreateReader()
-ReadReader()
-UpdateReader()
-DeleteReader()
-CreateRequest()
-ReadRequest()
-UpdateRequest()
-DeleteRequest()
-GetAllBooks()
-GetAllBookChapters()
-GetAllReader()
-SendAllRequest()
-ShowHistoryOfReader
context Admin:create Chapter
-pre CreateChapter 1 :ChapterID==ID
-pre CreateChapter 2:Chapter.isDefined()
-pre CreateChapter 3:Chapter ->excludes()
-post CreateChapter :chapter ->includes()
context Admin:Update chapter
-pre UpdateChapter 1 :Chapter.isDefined()
-pre UpdateChapter 2:Chapter ->excludes()
-post UpdateChapter :Chapter ->includes()
request
context Admin:create reader
-pre CreateReader1 :ReaderID==ID
-pre CreateReader 2:Reader.isDefined()
-pre CreateReader 3:Reader ->excludes()
-post CreateReader :Reader ->includes()
context Admin:Update Reader
-pre UpdateReader 1 :Reader.isDefined()
-pre UpdateReader 2:Reader ->excludes()
-post UpdateReader :Reader ->includes()
Book
-BookName : String
-BookDescription : String
-Author : String
-Category : String
+PublicationDate : int
manage
n
Chapters
-ChapterTitle : String
-ChapterDescription : String
manage
context Admin:create request
-pre CreateRequest 1:RequestID==ID
-pre CreateRequest 2:Request.isDefined()
-pre CreateRequest 3:Request ->excludes()
-post CreateRequest :Request ->includes()
context Admin:Update request
-pre UpdateRequest 1 :Request.isDefined()
-pre UpdateRequest 2:Request ->excludes()
-post UpdateRequest :Request ->includes()