Project Name: Online Library

Team information (typed not handwritten, except for attendance and grade):

	ID	Member Name	Handwritten Signature
1	20210561	عبد الله مصطفى محمود الشامى	
2	20210930	مريم وليد مصطفى شعلان	
3	20210942	مصطفى محمد زنون محمد	
4	20210948	مصطفى وليد سعيد السيد	
5	20210389	زينب ياسر محمد امام	
6	20210765	محمد حمادة محمد اسماعيل	

Project requirements (grading criteria)

SRS	
SDD	
Validation	
OCL	
AOP	
microservice	

- **System Requirements**

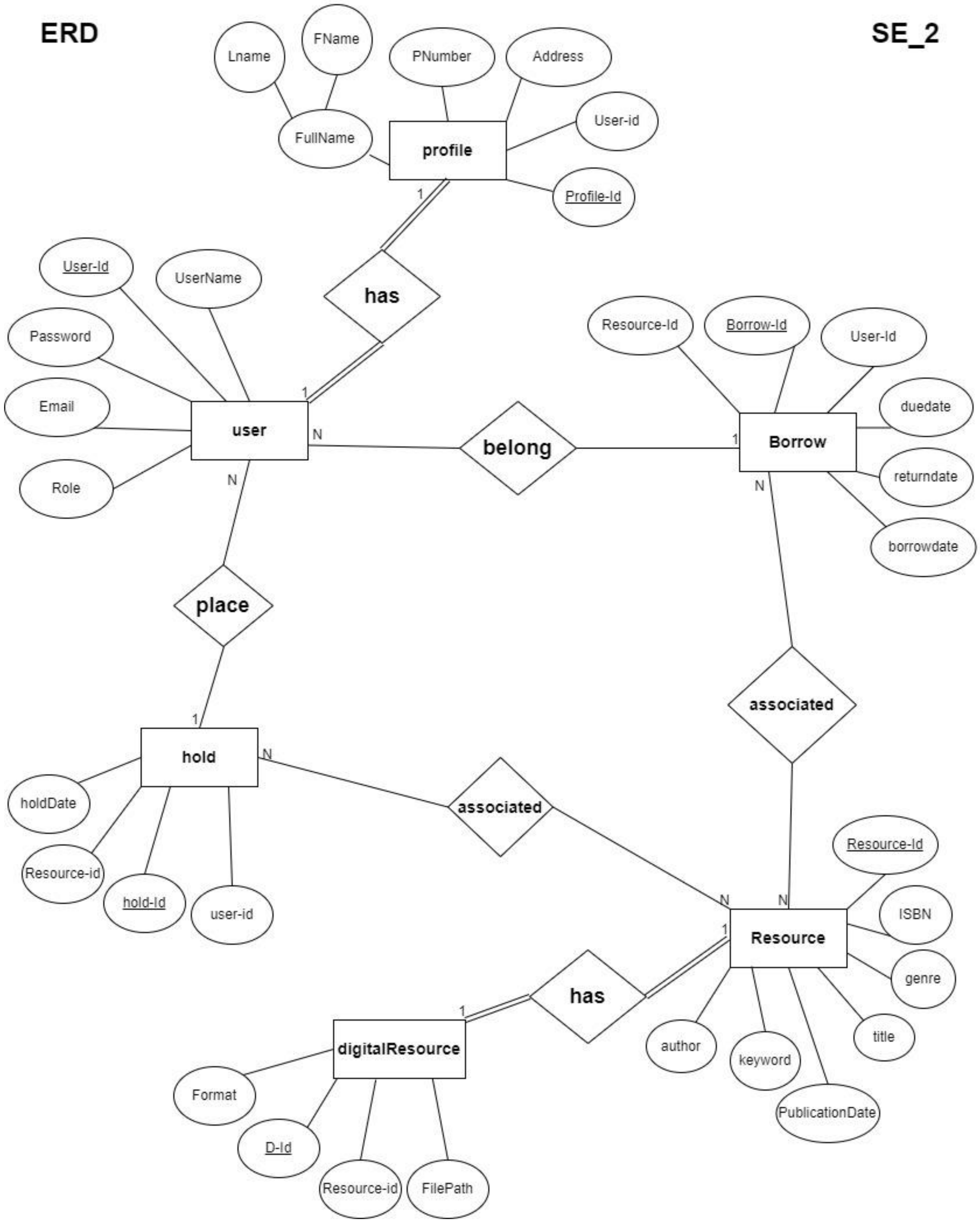
1) functional requirements:

- **User Authentication: Users should be able to create accounts, log in, and log out securely.**
- **Search Functionality: Users should be able to search for books by title, author, genre, or keyword.**
- **Book Availability: The system should display whether a book is available for borrowing or not.**
- **Book Reservation: Users should be able to reserve books that are currently checked out.**
- **Book Checkout and Return: Users should be able to check out books and return them within a specified period.**
- **User Profile Management: Users should be able to update their profiles, including contact information and password.**
- **Admin Panel: Administrators should have access to an admin panel to manage books, users, and system configurations.**
- **Notification System: Users should receive notifications about overdue books, reservation confirmations, etc.**
- **Book Recommendations: The system should provide personalized book recommendations based on user preferences and borrowing history.**
- **Rating and Review: Users should be able to rate and review books they have read.**

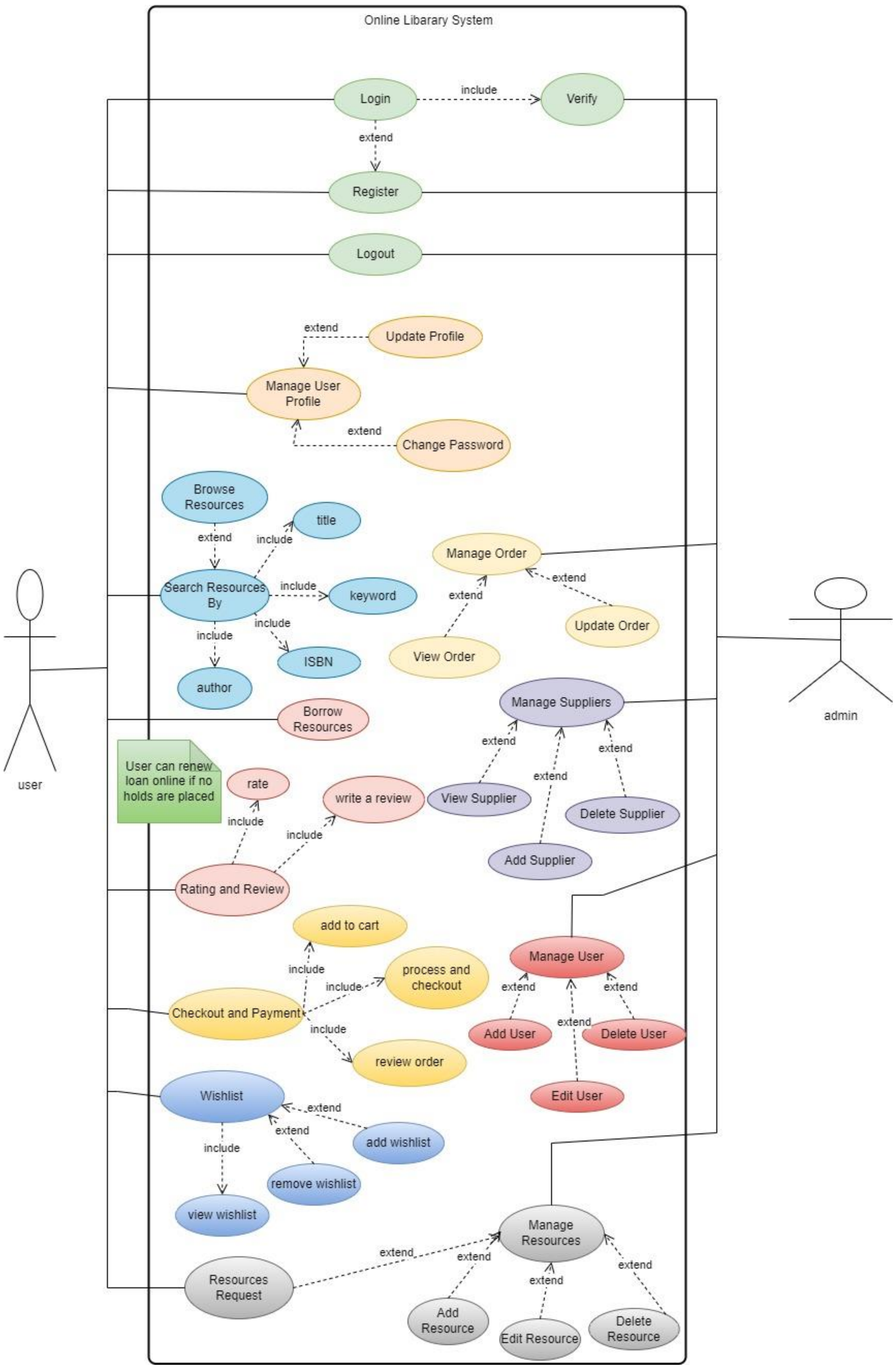
2) Non-functional requirements

- **Performance: The system should be able to handle a large number of simultaneous users without significant performance degradation.**
- **Scalability: The system should be designed to scale easily as the number of users and books increases.**
- **Security: The system should implement robust security measures to protect user data and prevent unauthorized access.**
- **Reliability: The system should be highly reliable, with minimal downtime and data loss.**
- **Usability: The user interface should be intuitive and easy to use, even for users with limited technical knowledge.**
- **Accessibility: The system should be accessible to users with disabilities, following accessibility standards such as WCAG.**
- **Compatibility: The system should be compatible with a wide range of devices and browsers.**
- **Data Integrity: The system should ensure the integrity of data, preventing errors such as duplicate records or data corruption.**
- **Backup and Recovery: Regular backups of data should be performed, and mechanisms for data recovery in case of system failure should be in place.**
- **Regulatory Compliance: The system should comply with relevant laws and regulations regarding data protection and privacy, such as GDPR.**

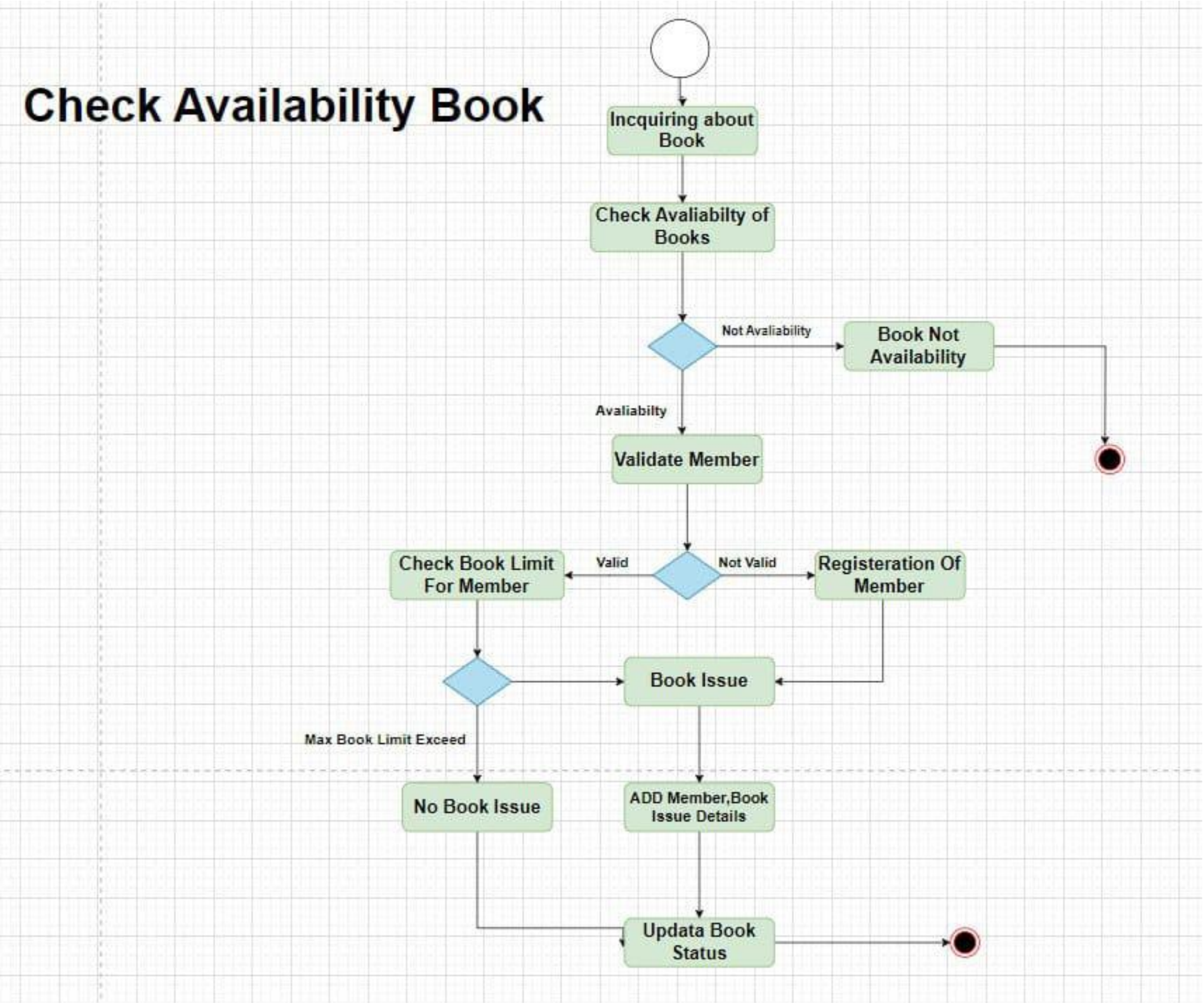
1) ERD:


ERD
SE_2
Lname
FName
PNumber
Address
FullName
profile
User-id
Profile-Id
1
has
User-Id
UserName
Password
Email
Role
user
1
N
N
belong
Resource-Id
Borrow-Id
User-Id
duedate
returndate
borrowdate
Borrow
1
N
place
1
hold
N
associated
associated
holdDate
Resource-id
hold-Id
user-id
Resource-Id
ISBN
genre
title
N
N
1
Resource
has
1
digitalResource
author
keyword
PublicationDate
Format
D-Id
Resource-id
FilePath

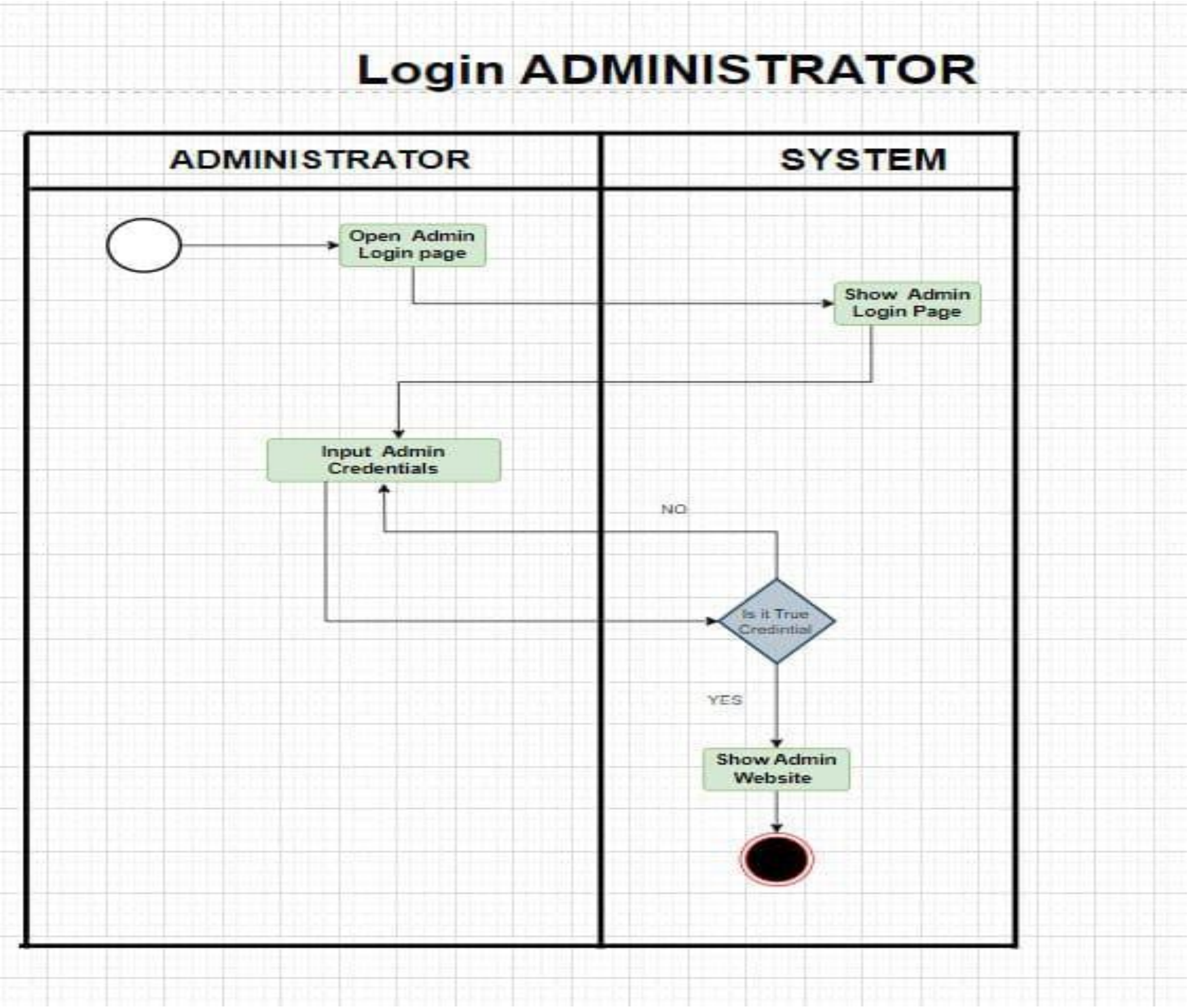
2) use case diagram:



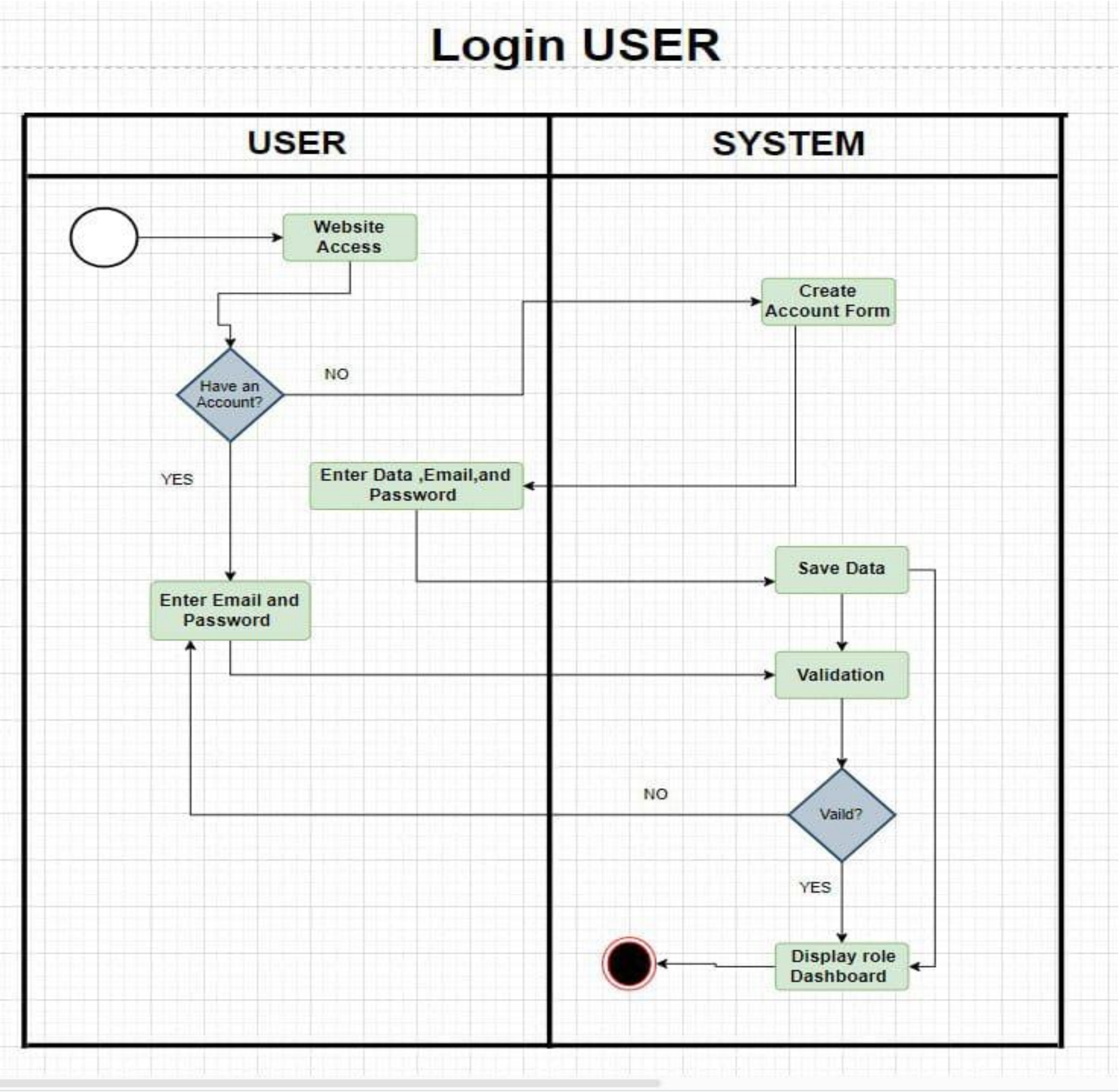
- **Activity diagram (check availability book):**



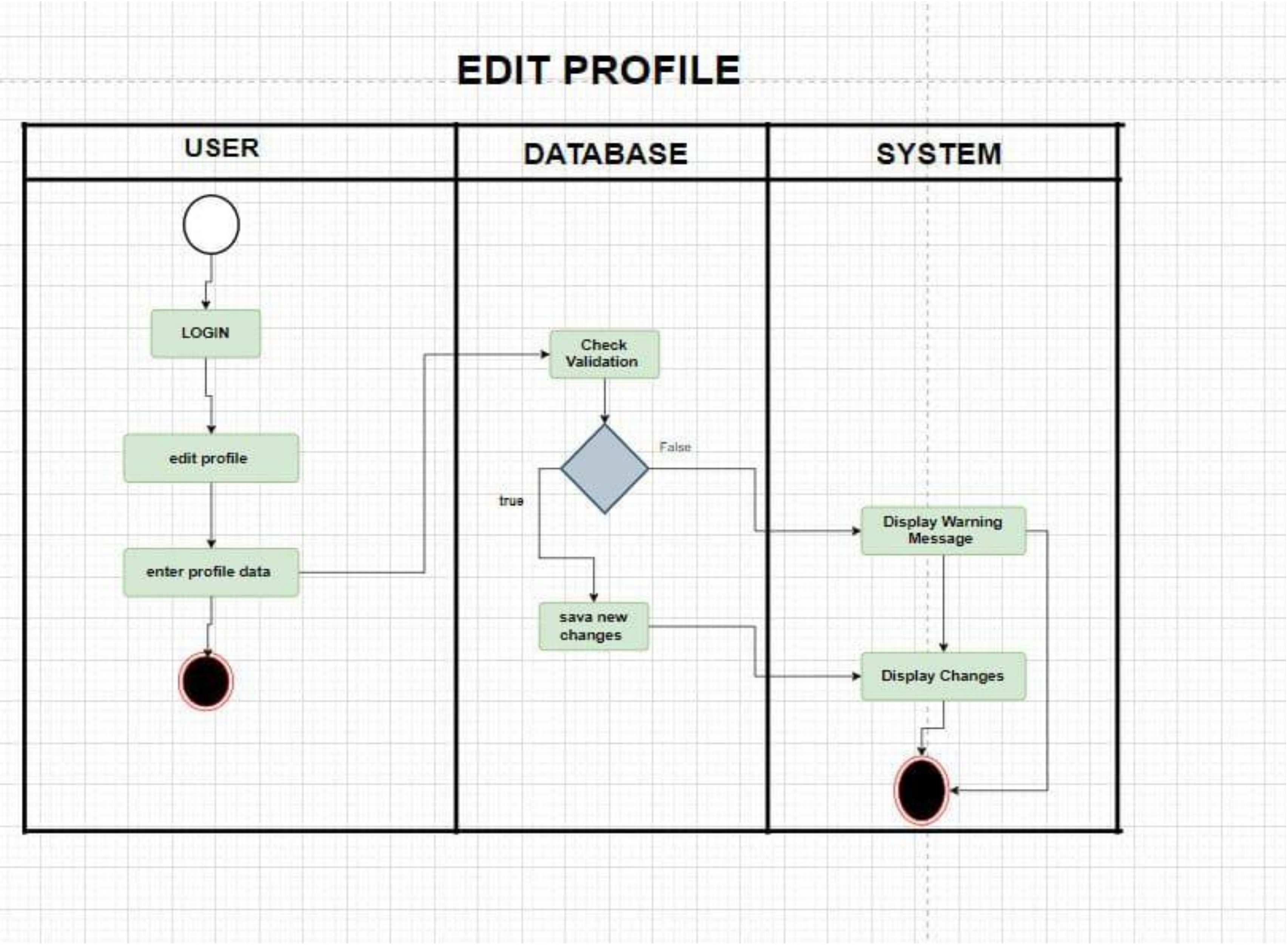
- **Activity diagram (Login administrator)**



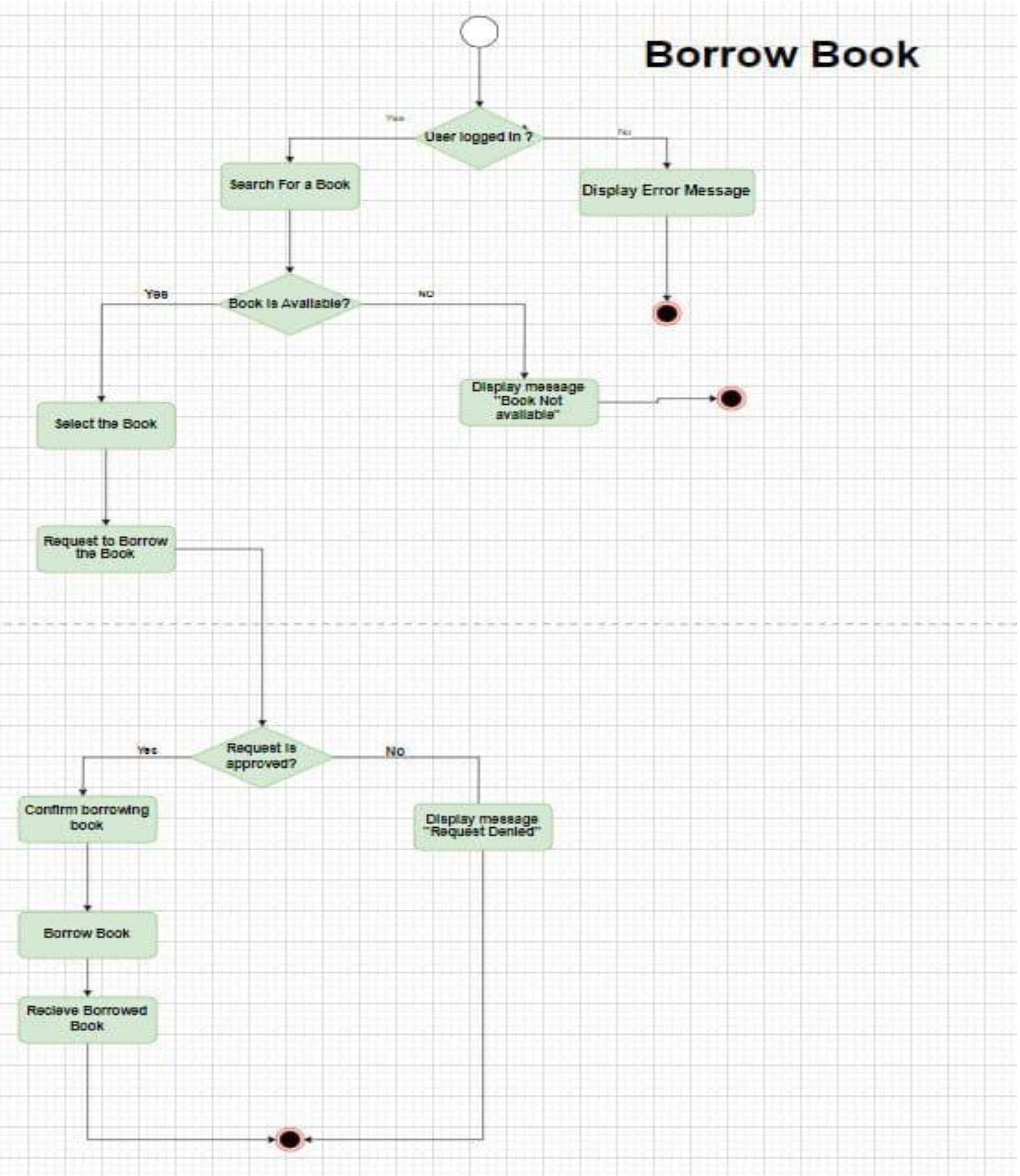
- **Activity diagram (Login User)**



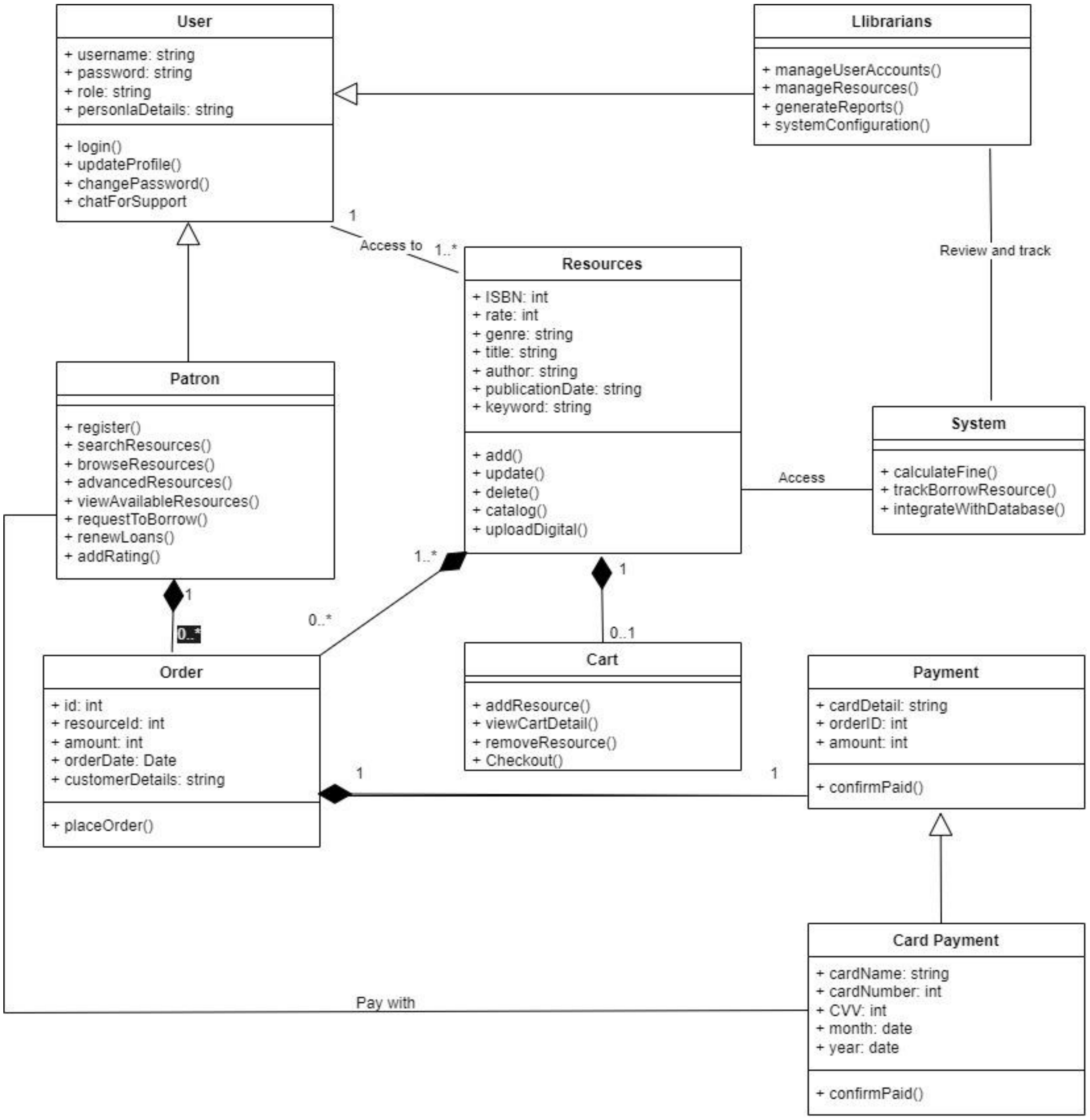
- **Activity diagram (Edit Profile)**



- **Activity diagram (borrow book)**

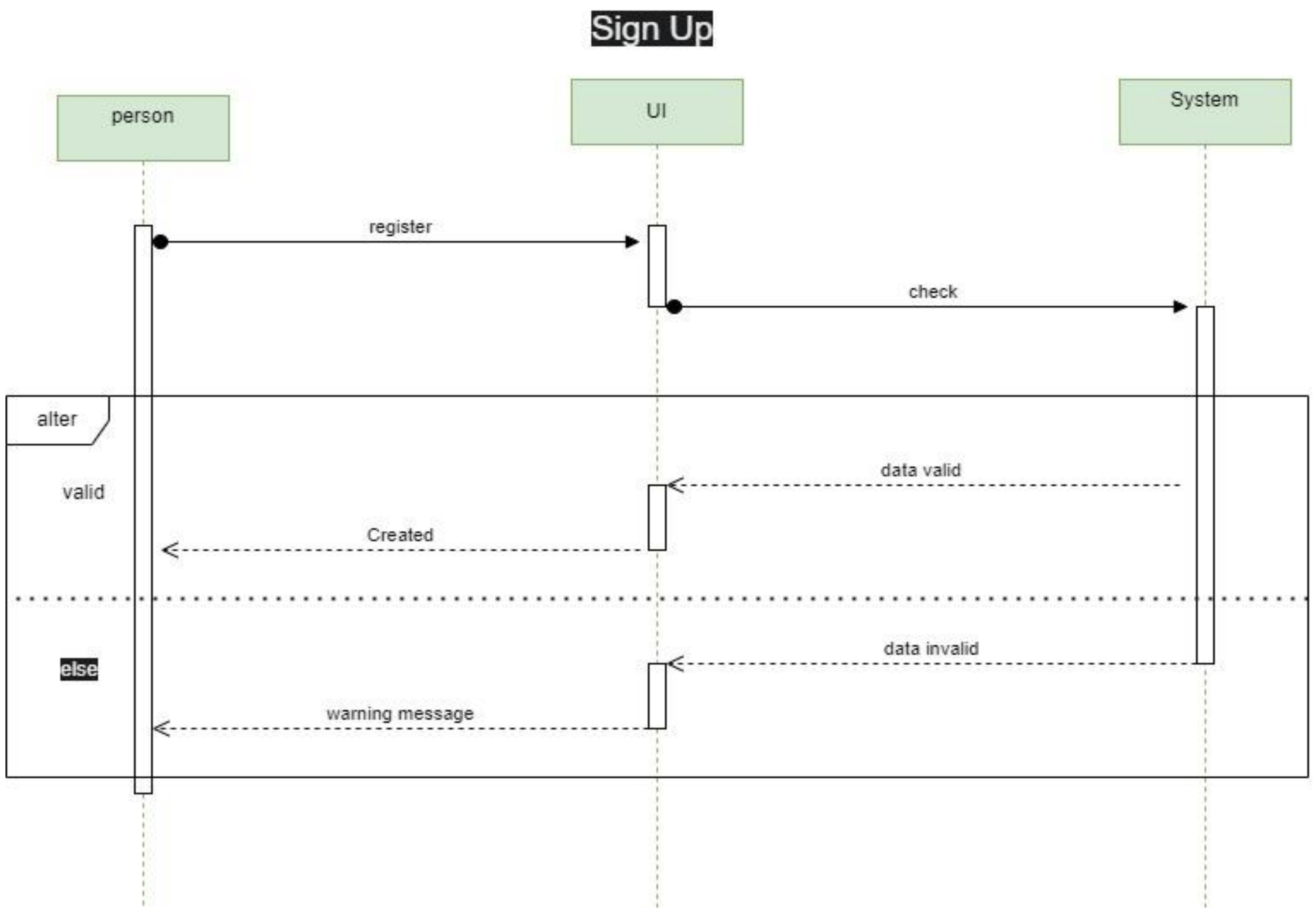


4) Class diagram :

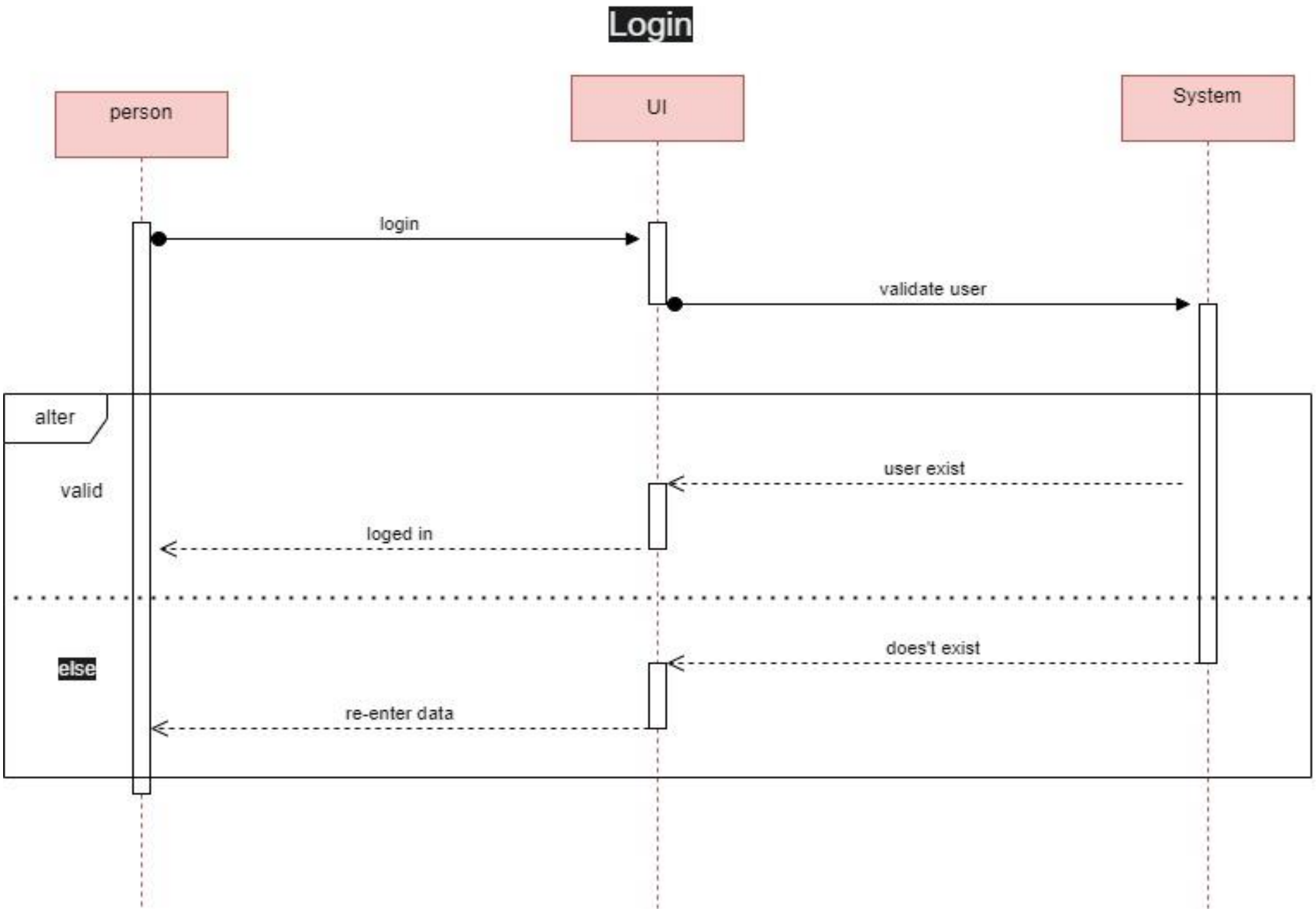


5) Sequence diagram:

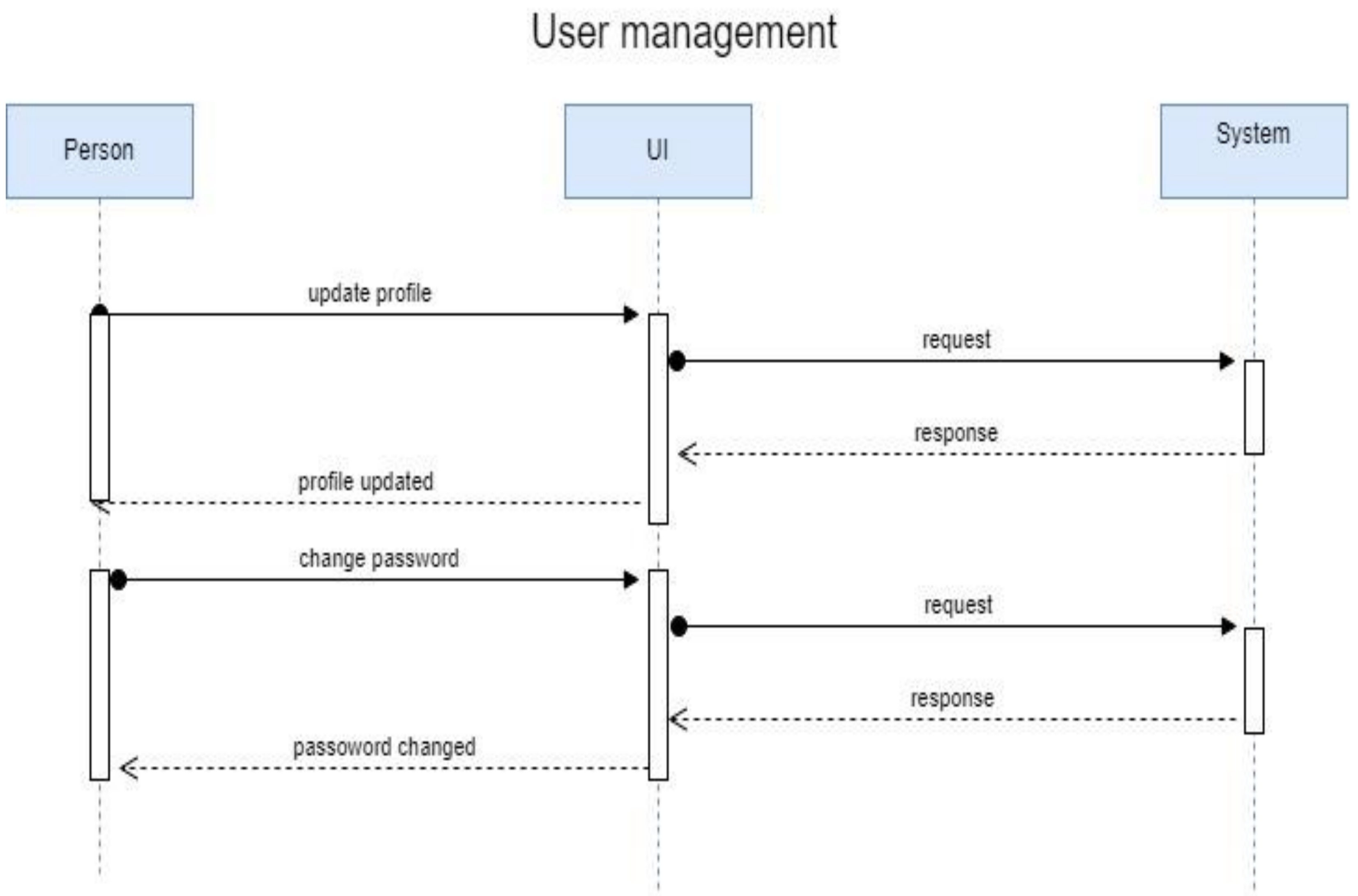
- **Sign up sequence diagram**



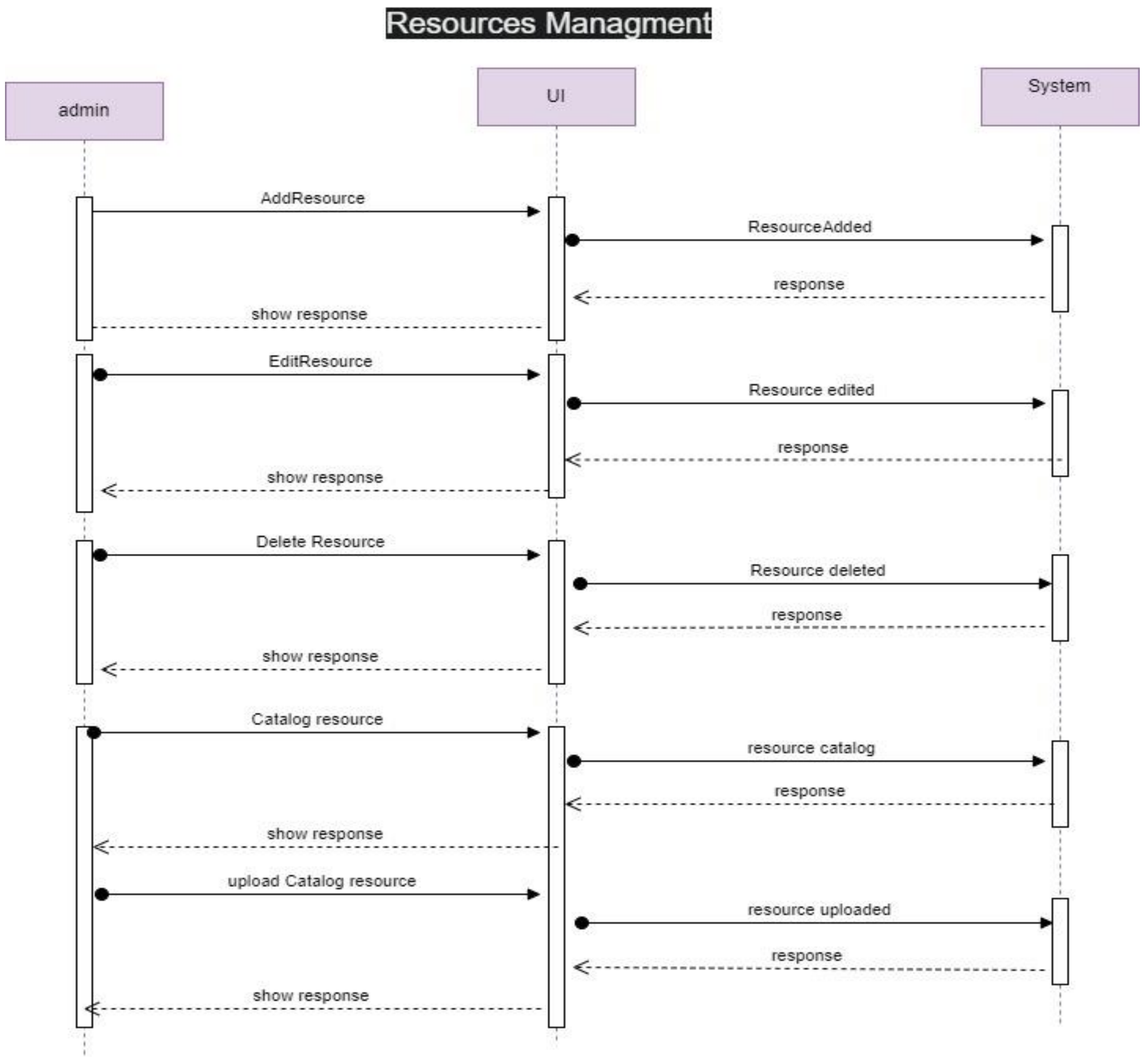
- **Sequence diagram for Log in**



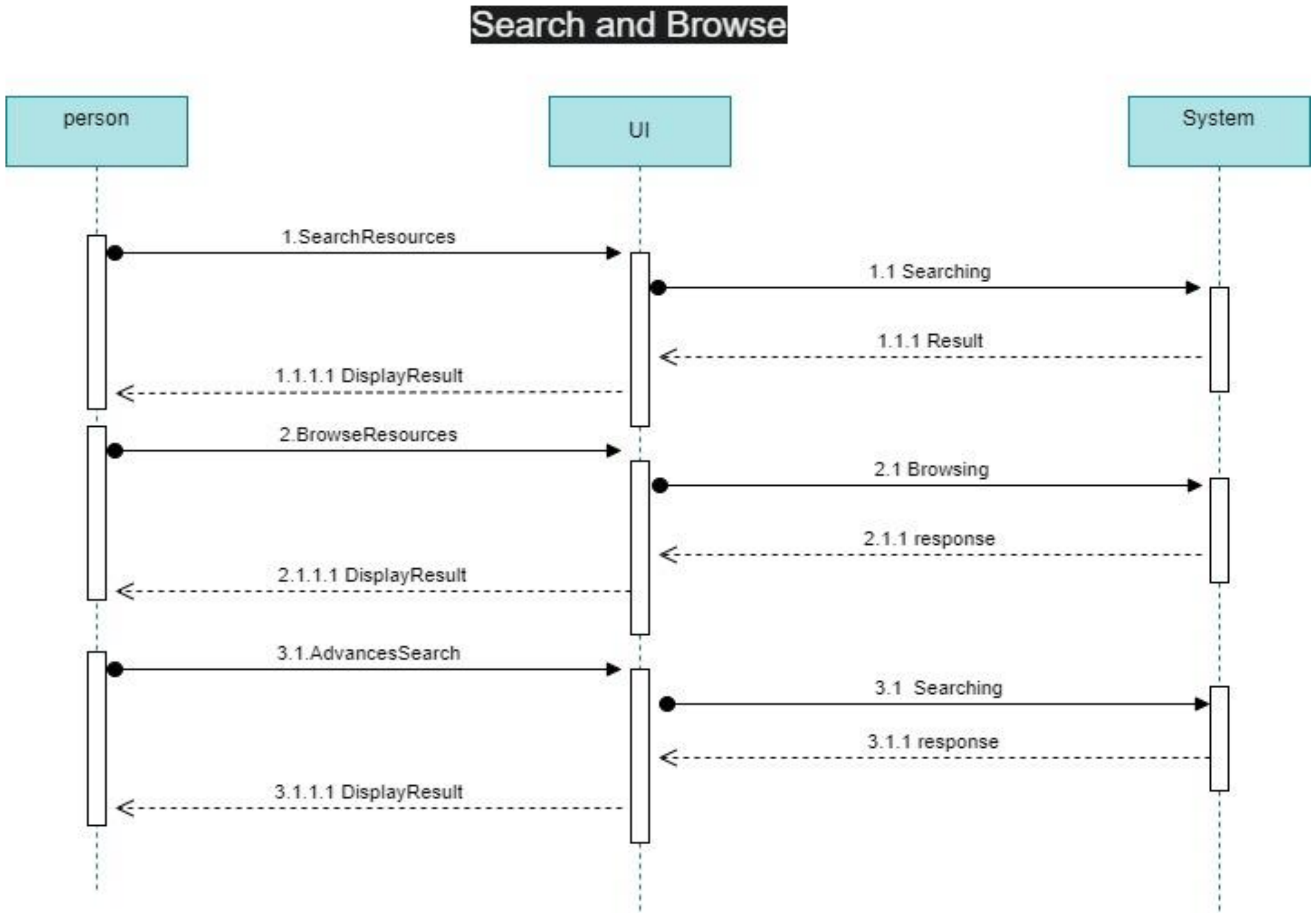
- **Sequence diagram for user management**



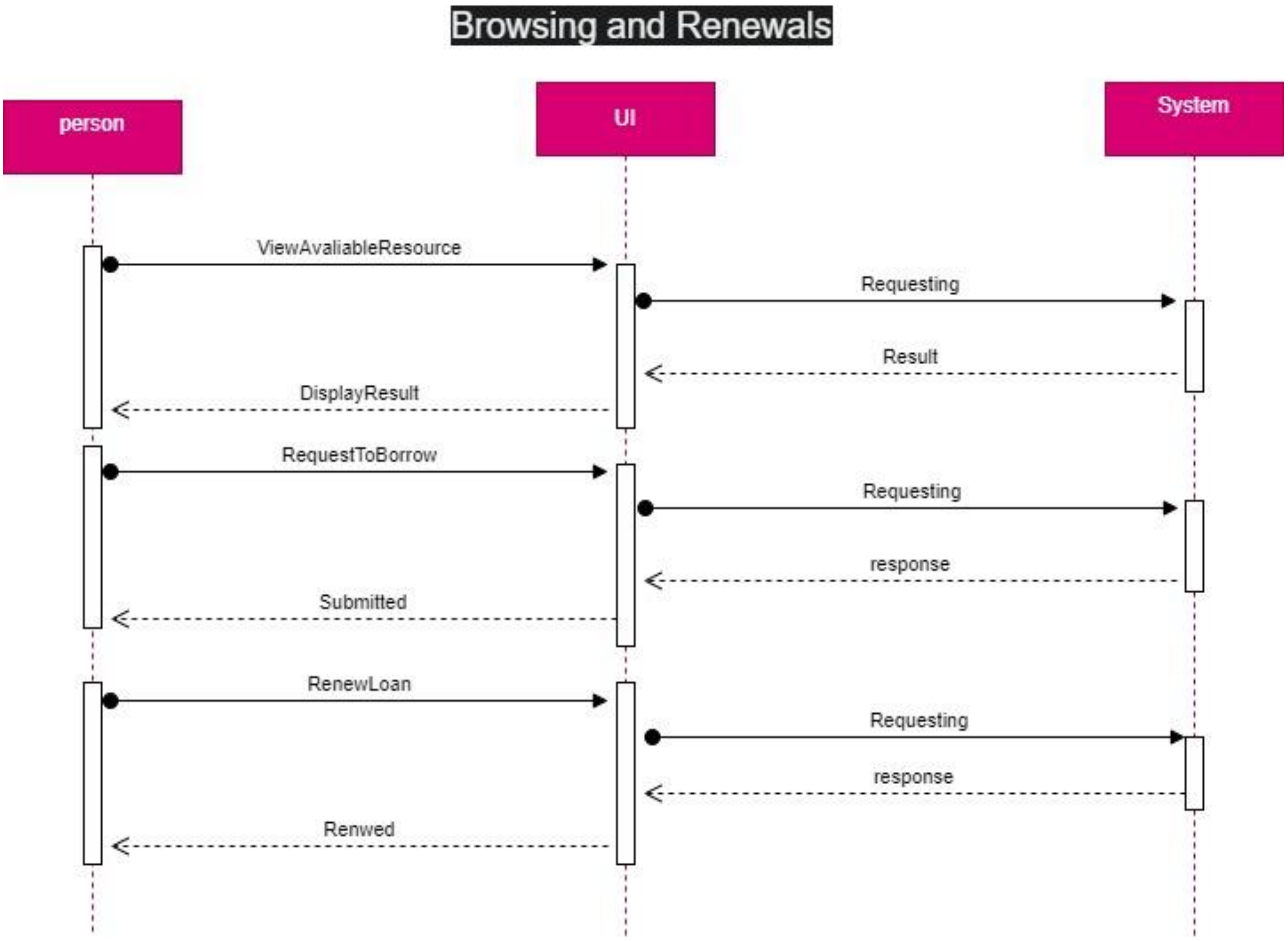
- **Sequence diagram for resource management**



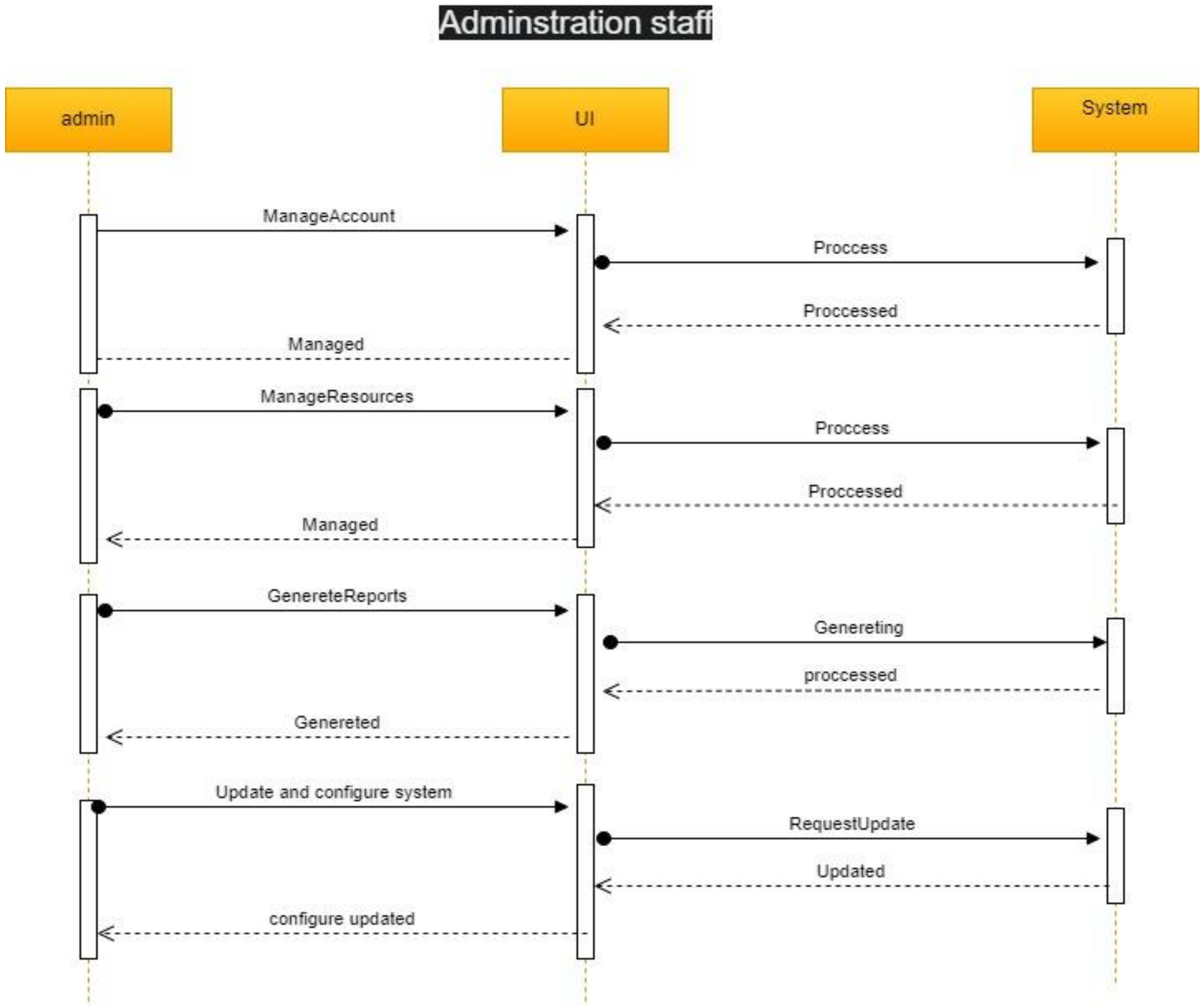
- **Sequence diagram Search and browse**



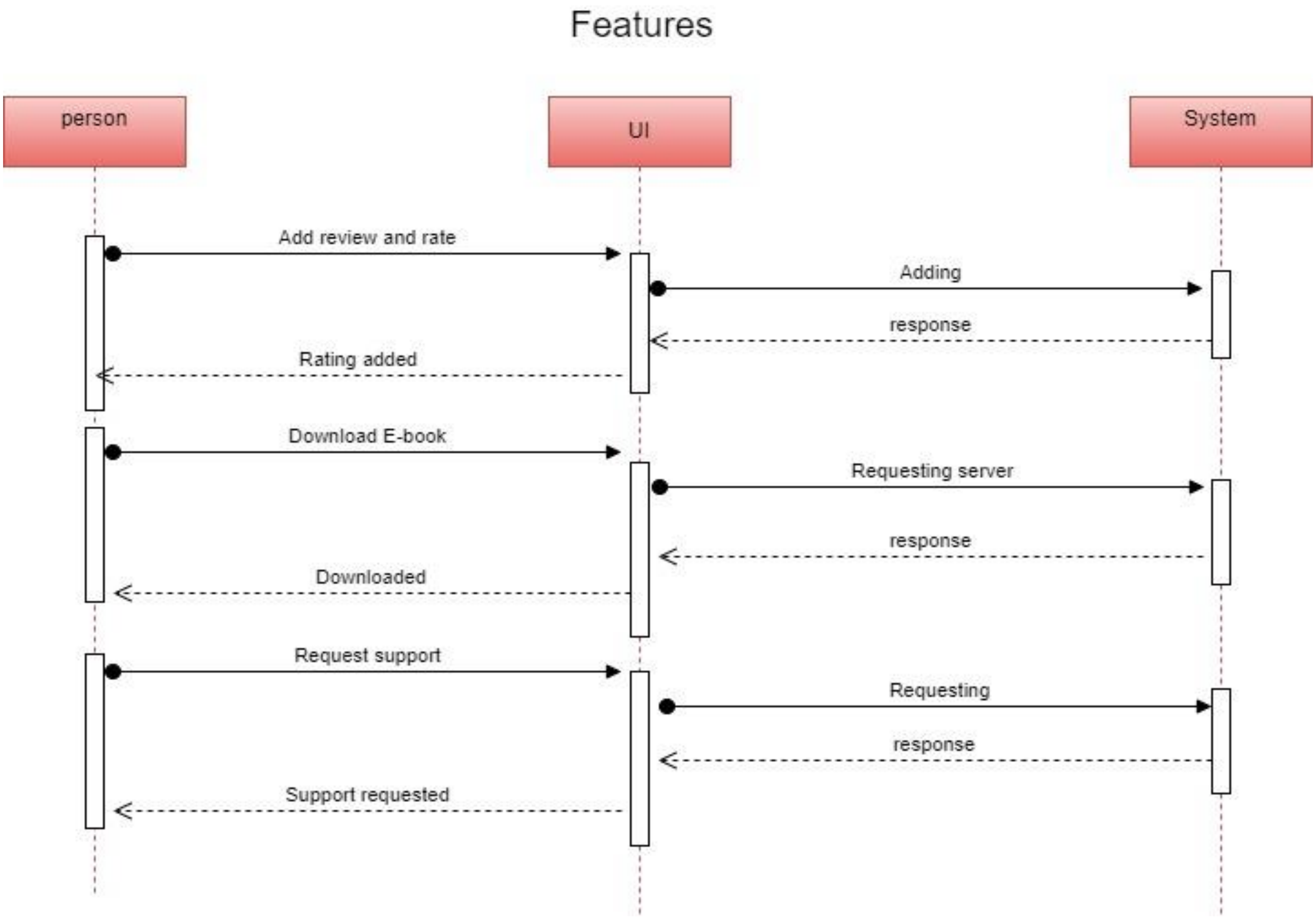
- **Sequence diagram for Browsing and Renewals**



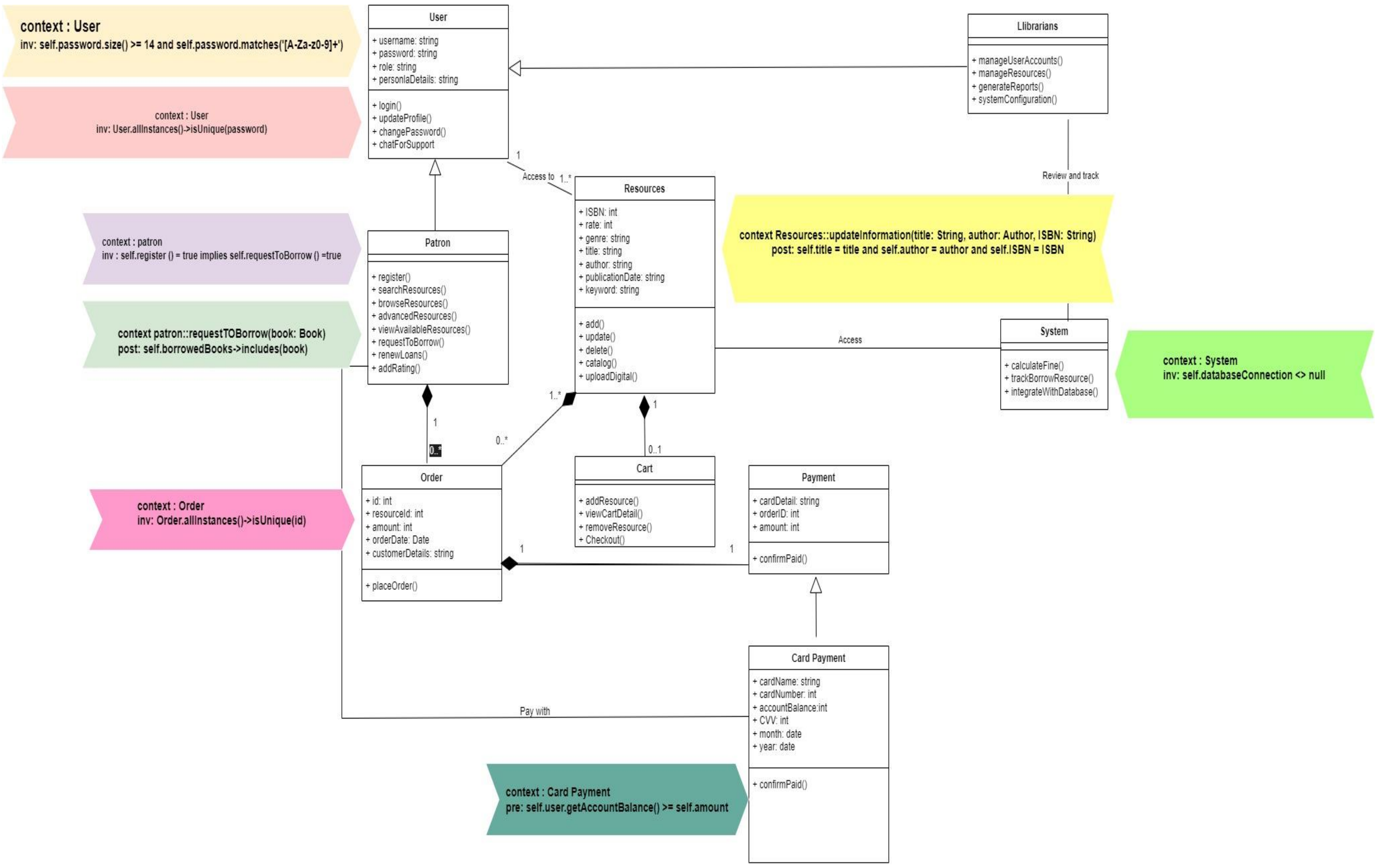
- ## Sequence diagram for Administration staff



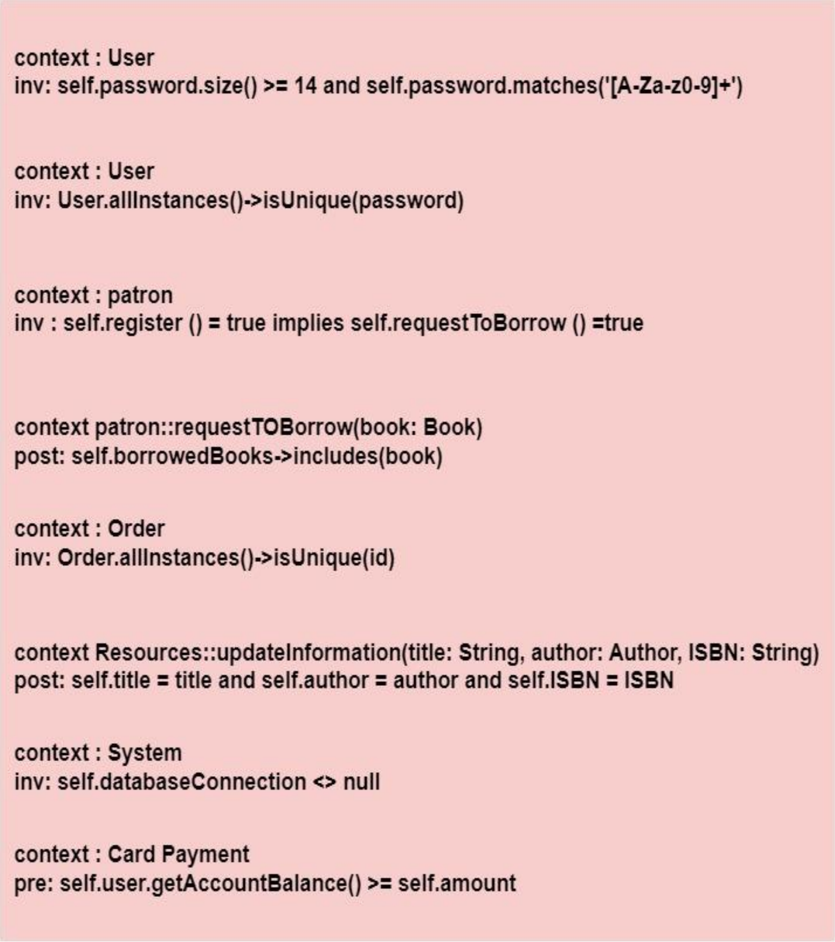
- **Sequence diagram for Features**



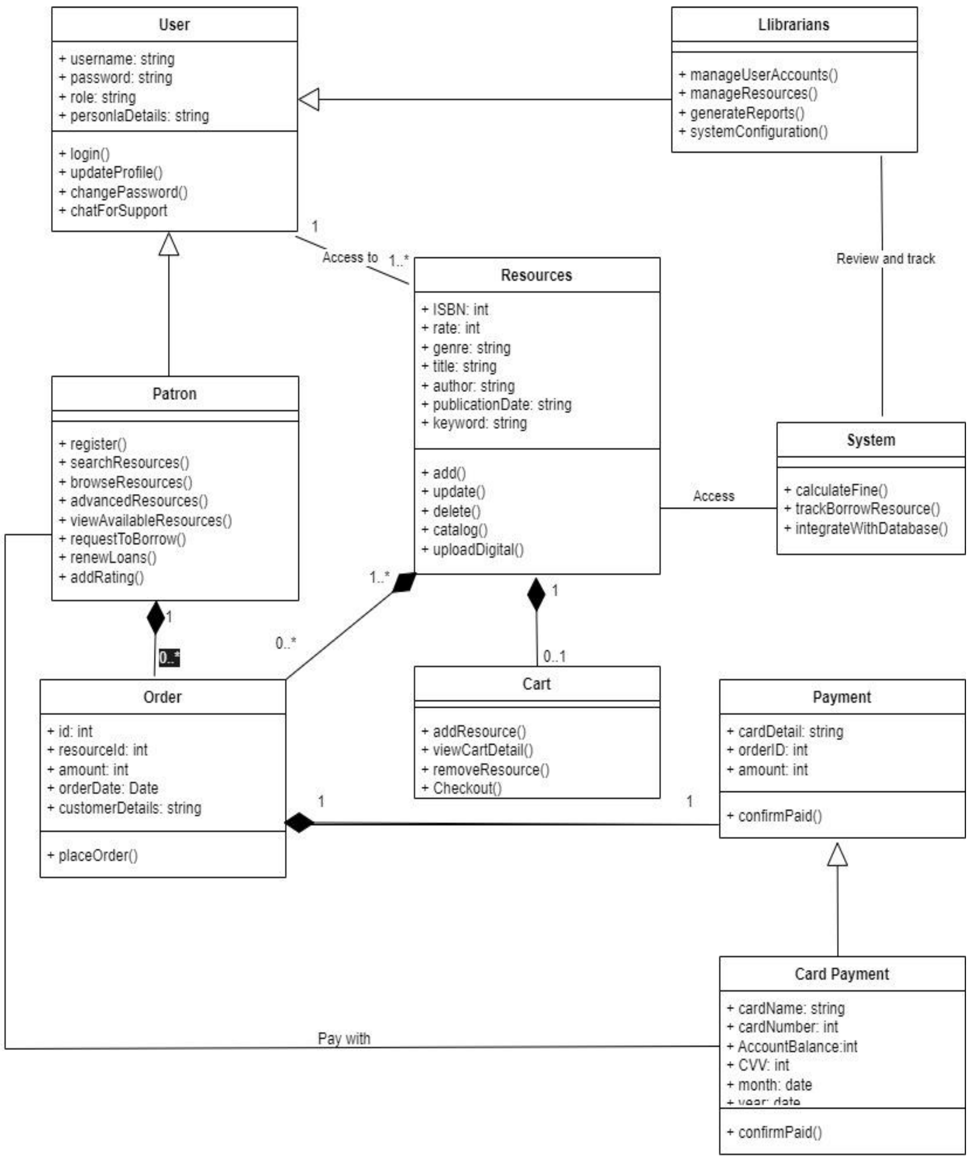
6) OCL on class diagram



- **OCL in another shape**



```
context : User
inv: self.password.size() >= 14 and self.password.matches('[A-Za-z0-9]+')

context : User
inv: User.allInstances()->isUnique(password)

context : patron
inv : self.register () = true implies self.requestToBorrow () =true

context patron::requestTOBorrow(book: Book)
post: self.borrowedBooks->includes(book)

context : Order
inv: Order.allInstances()->isUnique(id)

context Resources::updateInformation(title: String, author: Author, ISBN: String)
post: self.title = title and self.author = author and self.ISBN = ISBN

context : System
inv: self.databaseConnection <> null

context : Card Payment
pre: self.user.getAccountBalance() >= self.amount
```


User
+ username: string
+ password: string
+ role: string
+ personlaDetails: string
+ login()
+ updateProfile()
+ changePassword()
+ chatForSupport
Llibrarians
+ manageUserAccounts()
+ manageResources()
+ generateReports()
+ systemConfiguration()
1
Access to
1..*
Review and track
Resources
+ ISBN: int
+ rate: int
+ genre: string
+ title: string
+ author: string
+ publicationDate: string
+ keyword: string
+ add()
+ update()
+ delete()
+ catalog()
+ uploadDigital()
Patron
+ register()
+ searchResources()
+ browseResources()
+ advancedResources()
+ viewAvailableResources()
+ requestToBorrow()
+ renewLoans()
+ addRating()
System
+ calculateFine()
+ trackBorrowResource()
+ integrateWithDatabase()
Access
1..*
1
1
0..*
0..*
0..1
Order
+ id: int
+ resourceId: int
+ amount: int
+ orderDate: Date
+ customerDetails: string
+ placeOrder()
Cart
+ addResource()
+ viewCartDetail()
+ removeResource()
+ Checkout()
Payment
+ cardDetail: string
+ orderID: int
+ amount: int
+ confirmPaid()
1
1
Card Payment
+ cardName: string
+ cardNumber: int
+ AccountBalance:int
+ CVV: int
+ month: date
+ year: date
+ confirmPaid()
Pay with