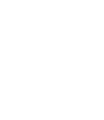


***Software Requirement and Design***

***Specifications***



***RAILWAY RESERVATION SYSTEM***

***Version: [00.01]***

|  |  |
| --- | --- |
| *Course Code* | CS2008 |
| *Instructor* | Abeeha Sattar |
| *Project Team* | Abbas Dossa [ 20k-1660 ] |
| *Submission Date* | 25th May 2022 |



Page 2 of 12

***[Instructions]***

- *No section of template should be deleted. You can write ‘Not applicable’ if a section is not applicable to your project. But all sections must exist in the final document.*

- *All comments/examples mentioned in square brackets ([]) are in the template for explanation purposes and must be replaced / removed in final document.*

- *This’ Instruction’ section should also be removed in final document.*

* Design Strategy and Database Design is not applicable for you. Any other type of diagram that we have not done, is also not applicable for you.
* Add a section under 82 for each diagram

*Table of Contents*

*1. INTRODUCTION .....................................................................................................................................................5*

1.1. Purpose of Document.........................................................................................................................5

1.2. Intended Audience ..............................................................................................................................5

*2. OVERALL SYSTEM DESCRIPTION ............................................................................................................................6*

2.1. Project Background.............................................................................................................................6

2.2. Project Scope ......................................................................................................................................6

2.3. Not In Scope ........................................................................................................................................6

2.4. Project Objectives ...............................................................................................................................6

2.5. Stakeholders ........................................................................................................................................6

2.6. Operating Environment ......................................................................................................................6

2.7. System Constraints.............................................................................................................................6

2.8. Assumptions & Dependencies ..........................................................................................................6

*3. EXTERNAL INTERFACE REQUIREMENTS ..................................................................................................................7*

3.1. Hardware Interfaces ...........................................................................................................................7

3.2. Software Interfaces .............................................................................................................................7

3.3. Communications Interfaces ...............................................................................................................7

*4. FUNCTIONAL REQUIREMENTS................................................................................................................................8*

*4.1. FUNCTIONAL HIERARCHY .............................................................................................................................8*

4.2. Use Cases............................................................................................................................................8

4.2.1. [Title of use case] ...........................................................................................................................8

*5. NON-FUNCTIONAL REQUIREMENTS........................................................................................................................9*

5.1. Performance Requirements...............................................................................................................9

5.2. Safety Requirements ..........................................................................................................................9

5.3. Security Requirements .......................................................................................................................9

5.4. User Documentation ...........................................................................................................................9

*SDS .........................................................................................................................................................................10*

***6. SYSTEM ARCHITECTURE ............................................................................................................................11***

*6.1. SYSTEM LEVEL ARCHITECTURE ...................................................................................................................11*

***7. DESIGN STRATEGY .......................................................................................................................................11***

***8. DETAILED SYSTEM DESIGN .......................................................................................................................11***

*8.1. DATABASE DESIGN .....................................................................................................................................11*

*8.2 . APPLICATION DESIGN........................................................................................................................................12*

*9. REFERENCES ......................................................................................................................................................12*

*10. APPENDICES .......................................................................................................................................................12*

***1. Introduction***

***1.1. Purpose of Document***

The purpose of this document is to make sure, that the software product built meets the requirements of the end customer. Moreover, with this document we can fill in the gap and develop a basis for agreement between customers and development organization on what the software product is expected to do. Finally, in this document we will describe everything related to the railway reservation system we have developed, its constraints, functionalities, moreover diagrams explaining the control flow of the system and much more.

***1.2. Intended Audience***

*Citizens whom are interested in travelling by train, and the administration staff whom would manage the system.*

**1.3 Definition of Terms, Acronyms and Abbreviations**

|  |  |
| --- | --- |
| ***Term*** | ***Description*** |
| PC | Personal Computer |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

***1.4 Document Convention***

Font: Arial for all the content.

Font size: 10 (Regular content/ Paragraph) 12 for headings.

***2. Overall System Description***

***2.1. Project Background***

*Pakistan has numerous railway tracks, routes and schedules; but people find it really difficult to remember the routes or either practically go and discover the route while buying a ticket on-site. Therefore, we came up with a solution in the case of a railway reservation system, where our system would have two modules, the user mode and the admin mode.*

***2.2. Project Scope***

The railway reservation system allows passengers to inquire about available trains based on their origin and destination, book and cancel tickets, and inquire about the status of their booked ticket, among other things.

Waiting list reservation.

• Online reservation.

• Ticket reservation

***2.3. Not In Scope***

***2.4. Project Objectives***

*The Corporate aim of the Railways reservation system is to commit itself to ensuring that all its activities are managed to the highest level of safety which is pragmatic and reasonably practicable to achieve.*

***2.5. Stakeholders***

Stakeholders are:

• Admin

• Custom

• Programmer

• Developer

• Management

* Person at ticket counter.

***2.6. Operating Environment***

*The software can operate on-site and on online. On-site it would operate on an windows 10 operating environment, whereas online it could work either on 64bit windows 10 pc or any updated smartphone with android or ios operating system.*

***2.7. System Constraints***

• *Software constraints*

*System would not operate on any operating system which Is less than 10. Likewise your smartphone android system should be equally updated. The online system would only be accessible by internet access.*

• *Hardware constraints*

*The systems interface might show differently on a phone and a PC.*

*For an onsite system since touch hasn’t been enabled a manual keypad would be used.*

• *Cultural constraints*

*The system would only have English as a language to demonstrate.*

• *Legal constraints*

*Tax would be added as per country’s demands on the total bill.*

• *User constraints*

*-The target audience are mature males, therefore the children wont be able to use this application since terminologies and interfaces are hard to understand.*

- The admin interface might be simpler, whereas the user interface made a bit complex but relatively learnable.

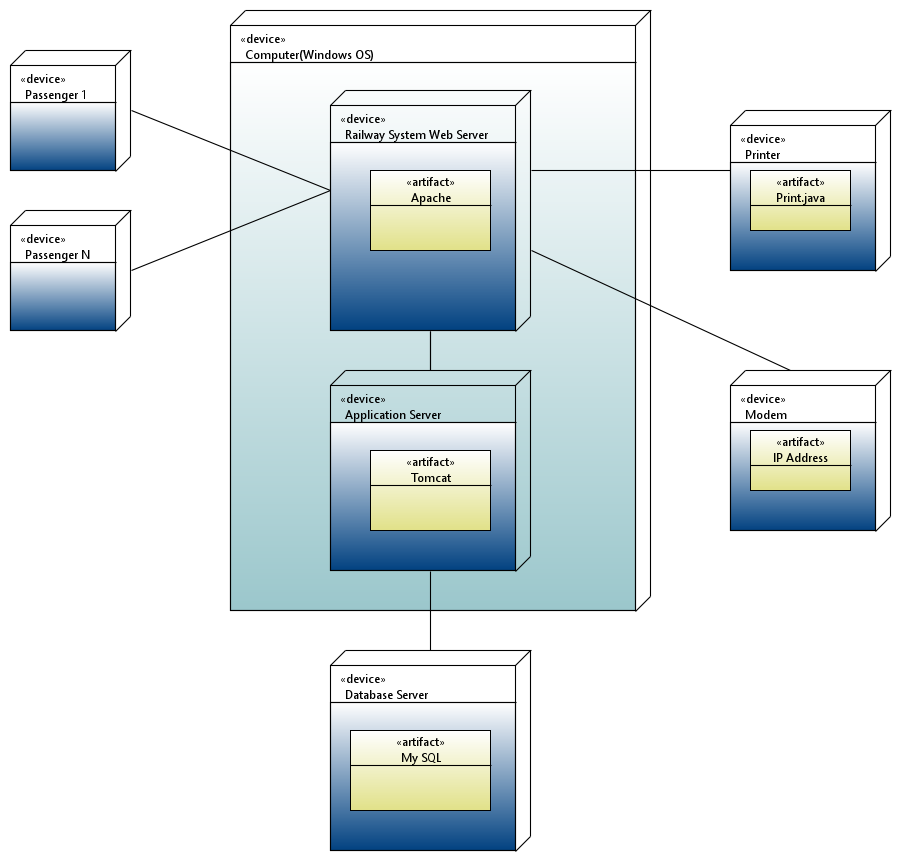
***2.8. Assumptions & Dependencies***

● User must have an android or ios based smart phone or PC with internet access. Or the user should know the basic English language to understand the system.

● Admin must have complete knowledge of the system. And should have past experience while running a management system.

● Software is dependent on internet connection to access

***3. External Interface Requirements***

****

***3.1. Hardware Interfaces***

*Some of the hardware interfaces which have been linked to our system are as follows:-*

1. *Printer: - To print the customers train ticket.*
2. *Modem: - Since the users could reserve their ticket online or on-site; they can do this by booking an on-site ticket via a model-based application which is linked with the system.*
3. *Mobile phones: - For online or e-booking of the ticket, android or ios based phones can be used.*

***3.2. Software Interfaces***

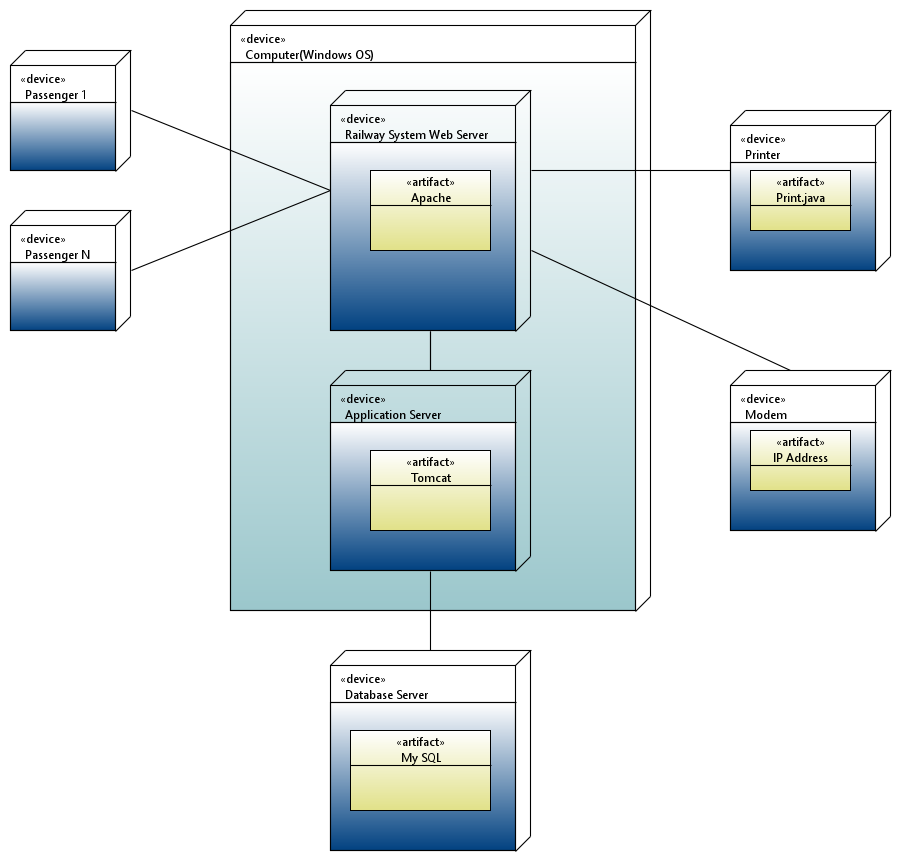
*The following software interfaces were used in our system :-*

1. *Computer ( Windows OS ) :- The train ticket reservation system was made on a 64bit computer having windows 10 as its operating system.*
2. *Database:- A database server based on SQL has been linked with the system, where all the data regarding the system is stored securely.*
3. *Server: - A railway webserver has been utilized.*
4. *Platform: - This system has been constructed on ASP .NET framework.*

*The data passing between each software components holds a different purpose, like the data passed to the database is for storage, whereas the data gathered from the server is utilized in the application for further functionalities.*

*The data shared to the database would be user details, user booked train details, user schedule etc.*

*The following deployment diagram displays the communication between the hardware and software interfaces.*

****

***3.3. Communications Interfaces***

*The multiple communication interfaces used are :*

1. *Forgot password :- if the user has forgot his password he can reset his password via a verification code sent on the users email.*
2. *Email/ SMS: - When a user books a ticket with complete payment; the user would be notified the ticket details via an email or SMS.*
3. Data encryption: - Not every data will be shown to every user. Not each data will be shown to the admin or, valuing the data of user.
4. *Security: - The admin module would be made extra secure by applying* data classification and protection mechanisms... Data classification is the conscious decision to assign a level of sensitivity to data as it is being created, amended, stored, transmitted, or enhanced.

.

***4. Functional Requirements***

***4.1. Functional Hierarchy***

*The functional hierarchy can be shown as follows:*

**ADMIN:**

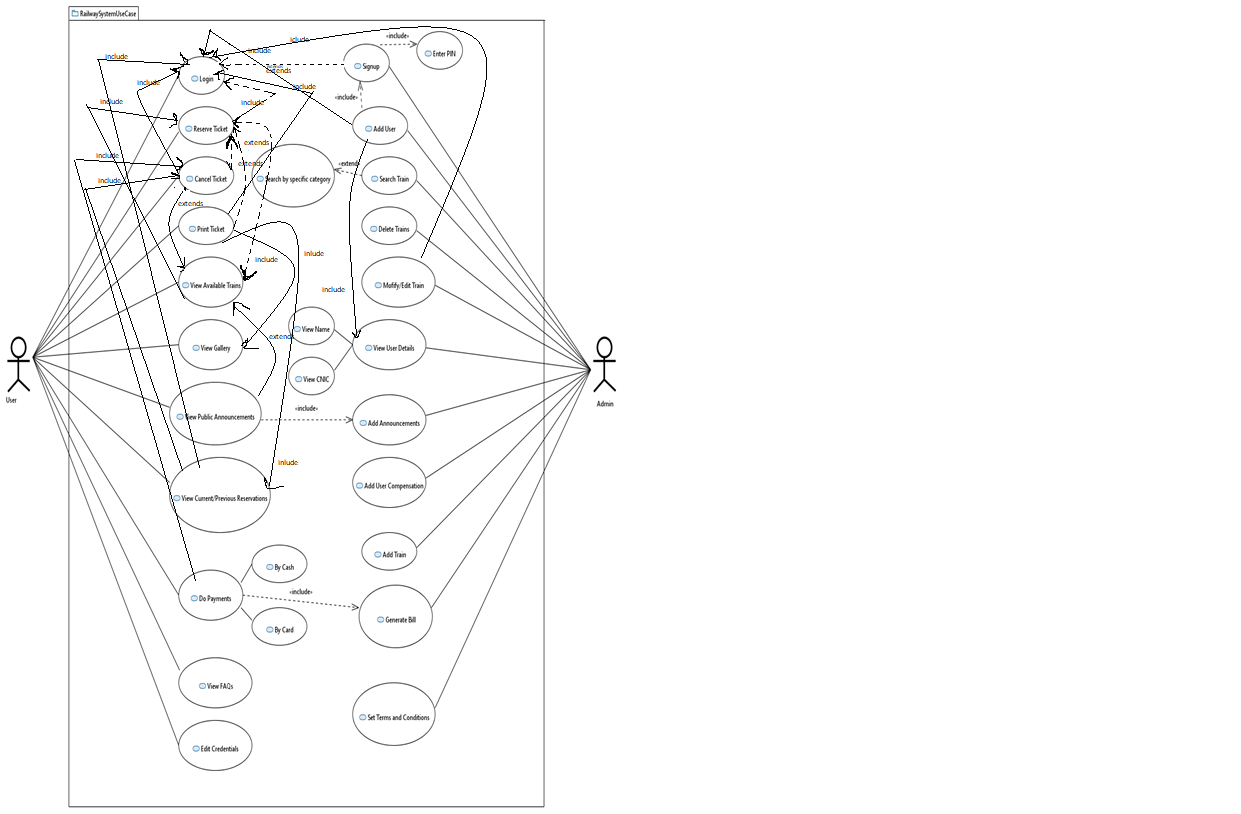
* Add Train
* Search Train
* Delete Existing Trains
* Modify Train Details
* View User’s Reservation Details
* User Creation
* Add Discount for Users
* Add New Admin (Requires Company PIN)
* Generate Bill
* Terms & Conditions/ Contact Us

**USER:**

* Reserve Ticket
* Cancellation of Ticket
* Account/Payment Details
* Print Ticket
* View Available Trains
* View Gallery
* View Public Notices
* View Current/Previous Reservations
* View FAQs
* Edit Credentials

***4.2. Use Cases***

System use case:



*From the following systems use case diagram, we have selected 4 most important use cases:*

***4.2.1. Reserve ticket***

*The user can book a ticket.*

|  |  |
| --- | --- |
|  | |
| ***Use Case name:*** | Reserve a ticket |
| ***Use Case Description:*** *Upon clicking the reserve ticket option on GUI, the user would be able to book a train according to the schedule he desires.* | |
| ***Primary actor:*** *user* | ***Other actors:*** *admin* |
| ***Stakeholders:*** | User, admin, counter man , conductor |
| ***Relationships***  ▪ ***Includes:*** *login, view available trains.*  ▪ ***Extends:*** *cancel ticket, print ticket.* | |
| ***Pre-conditions:***  ▪ User is logged in to the system | |
| **Main flow of events:**   * User selects a train * Books a train | |
| ***Alternative and exceptional flows:***  ***4.1*** Seats not available. Select another train. | |
| ***Post-conditions:***  ▪ Ticket successfully Booked | |

***4.2.2. Cancel ticket***

*The user can cancel a ticket already booked.*

|  |  |
| --- | --- |
|  | |
| ***Use Case name:*** | Cancel ticket |
| ***Use Case Description:*** *user cancels a ticket he has already booked.* | |
| ***Primary actor:*** *user* | ***Other actors:*** *admin* |
| ***Stakeholders:*** | User, admin, counter man , conductor |
| ***Relationships***  ▪ ***Includes:*** *login, view current/previous reservations*  ▪ ***Extends:*** *do payments, view available trains* | |
| ***Pre-conditions:***  ▪ User is logged in to the system | |
| **Main flow of events:**   * User selects a booked ticket * Cancel ticket * Receive compensation | |
| ***Alternative and exceptional flows:***  ***4.1*** . Ticket not booked in the first place. Invalid | |
| ***Post-conditions:***  ▪ Ticket successfully cancelled. | |

***4.2.3. Print ticket***

|  |  |
| --- | --- |
|  | |
| ***Use Case name:*** | Print ticket |
| ***Use Case Description: user prints a ticket he has already booked*** | |
| ***Primary actor:*** *user* | ***Other actors:*** *admin* |
| ***Stakeholders:*** | User, admin, counter man , conductor |
| ***Relationships***  ▪ ***Includes:*** *login, view current/previous reservations , view gallery*  ▪ ***Extends:*** *-* | |
| ***Pre-conditions:***   * User is logged into the system * train is selected | |
| **Main flow of events:**   * User selects a booked ticket * Prints ticket * Receives copy | |
| ***Alternative and exceptional flows:***  ***4.1*** . Ticket not booked in the first place. Invalid | |
| ***Post-conditions:***  ▪ Ticket successfully printed. | |

***4.2.4. view available trains***

|  |  |
| --- | --- |
|  | |
| ***Use Case name:*** | View available trains |
| ***Use Case Description: user views available trains*** | |
| ***Primary actor:*** *user* | ***Other actors:*** *admin* |
| ***Stakeholders:*** | User, admin, counter man , conductor |
| ***Relationships***  ▪ ***Includes:*** *login, reserve ticket*  ▪ ***Extends:*** *view public announcements* | |
| ***Pre-conditions:***   * User is logged into the system * The train description to view is selected. | |
| **Main flow of events:**   * User selects view trains * Goes through all available trains and checks description. | |
| ***Alternative and exceptional flows:***  ***4.1*** . NO trains available. | |
| ***Post-conditions:***  ▪ Trains successfully viewed. | |

***4.2.5. Add User***

|  |  |
| --- | --- |
|  | |
| ***Use Case name:*** | Add User |
| ***Use Case Description:*** Admin adds a new user to the system | |
| ***Primary actor:*** *admin* | ***Other actors:*** *user* |
| ***Stakeholders:*** | User, admin, counter man , conductor |
| ***Relationships***  ▪ ***Includes:*** *login, view user details*  ▪ ***Extends:*** *user compensation* | |
| ***Pre-conditions:***  ▪ Admin is logged in to the system | |
| **Main flow of events:**   * Admin selects a user * Adds a user to the system. | |
| ***Alternative and exceptional flows:***  ***4.1*** . user not added in fact deleted. | |
| ***Post-conditions:***  . User successfully added | |

***4.2.6 Modify Train***

|  |  |
| --- | --- |
|  | |
| ***Use Case name:*** | Modify train |
| ***Use Case Description:*** Admin modifies a train’s details in the system | |
| ***Primary actor:*** *admin* | ***Other actors:*** *user* |
| ***Stakeholders:*** | User, admin, counter man , conductor |
| ***Relationships***  ▪ ***Includes:*** *login, view train info , search train*  ▪ ***Extends:*** *delete train.* | |
| ***Pre-conditions:***  ▪ Admin is logged in to the system | |
| **Main flow of events:**   * Admin searches a train. * Admin modifies it | |
| ***Alternative and exceptional flows:***  ***4.1*** . No such train exists OR error in modifying train | |
| ***Post-conditions:*** Train modified Successfully  . | |

***4.2.7 Delete Train***

|  |  |
| --- | --- |
|  | |
| ***Use Case name:*** | Delete train |
| ***Use Case Description:*** Admin deletes a train from the system | |
| ***Primary actor:*** *admin* | ***Other actors:*** *user* |
| ***Stakeholders:*** | User, admin, counter man , conductor |
| ***Relationships***  ▪ ***Includes:*** *login, view train info , search train*  ▪ ***Extends:*** *add announcements* | |
| ***Pre-conditions:***   * admin is logged onto the system * train is searched | |
| **Main flow of events:**   * Admin searches a train. * Admin deletes it | |
| ***Alternative and exceptional flows:***  ***4.1*** . train added. | |
| ***Post-conditions:*** Train Deleted Successfully  . | |

***4.2.8 Add Announcements***

|  |  |
| --- | --- |
|  | |
| ***Use Case name:*** | Add announcements |
| ***Use Case Description:*** admin adds announcements in the railway system. | |
| ***Primary actor:*** *admin* | ***Other actors:*** *user* |
| ***Stakeholders:*** | User, admin, counter man , conductor |
| ***Relationships***  ▪ ***Includes:*** *login, view gallery*  ▪ ***Extends:*** *add user compensation* | |
| ***Pre-conditions:***   * admin is logged onto the system | |
| **Main flow of events:**   * Admin types an announcement * Admin adds it to the system * Announcement is seen by the users | |
| ***Alternative and exceptional flows:***  ***4.1*** . error in adding announcement | |
| ***Post-conditions:*** Announced Successfully  . | |

***5. Non-functional Requirements***

***5.1. Performance Requirements***

*Server limitations, concurrently requests being processed are performance requirements of the system.*

*Moreover, our system has a prompt user interaction interface with better speed, concurrently checked with load testin.*

***5.2. Safety Requirements***

As a public platform, it is genuinely concerning about the safety of the data of the people registered on the platform.

***5.3. Security Requirements***

Our system would automatically log out all customers after a period of inactivity. The system would not leave any cookies on the customers computer containing the users password. The systems backend servers shall only be accessible to authenticated management.

***5.4. User Documentation***

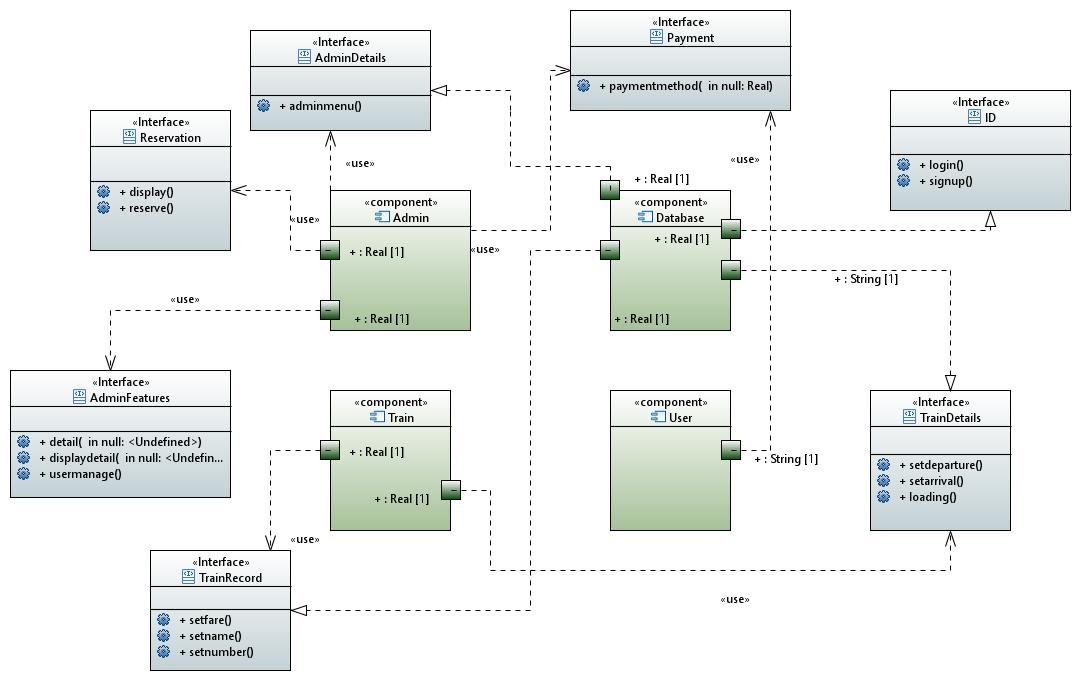
*User manual, FRID system, chips, QR Code scanner, and a training session (recorded video) will be delivered along with the system (software).*

***SDS***

***6. System Architecture***

**6.1. System Level Architecture**

COMPONENT DIAGRAM: -

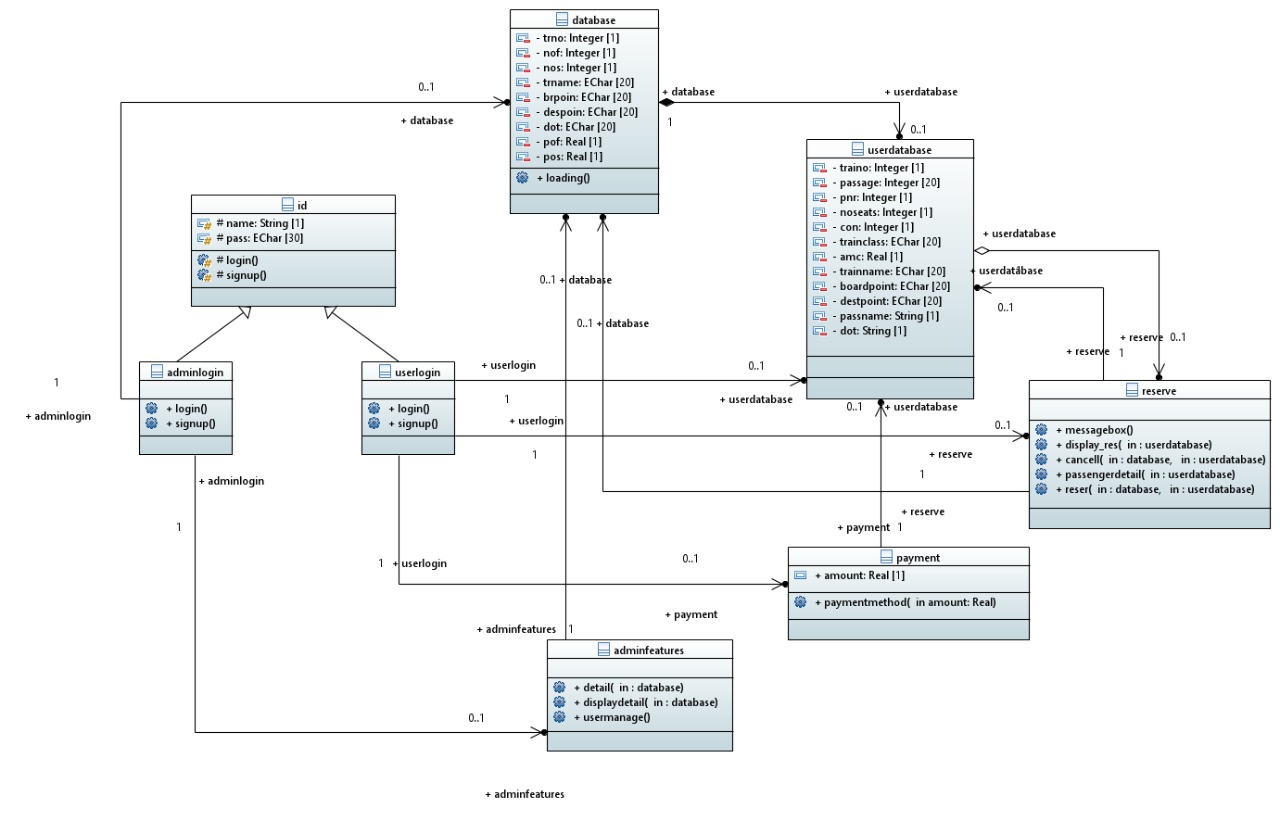
****

***7. Design Strategy***

*[NOT APPLICABLE]*

***8. Detailed System Design***

*Detailed class diagram:*

****

**8.1. Database Design**

*[NOT APPLICABLE]*

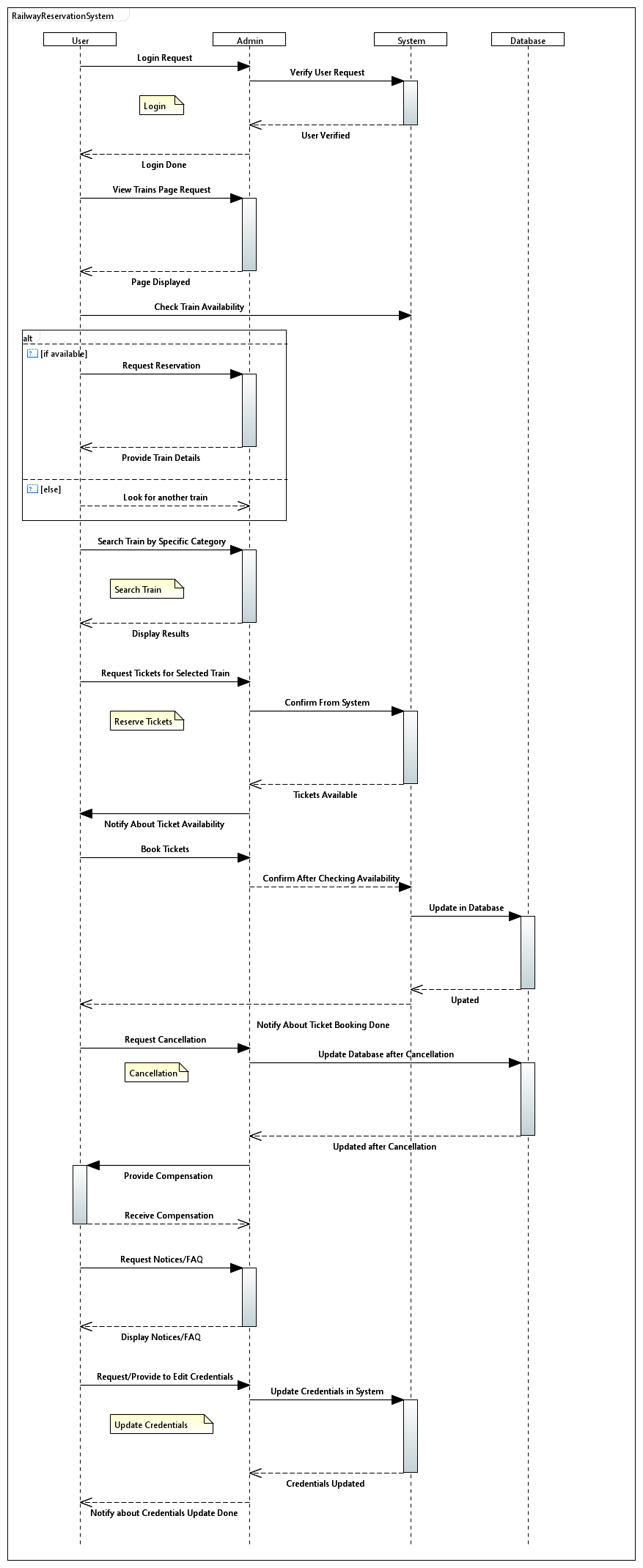
**8.2. Application Design**

*The diagrams are as follows:*

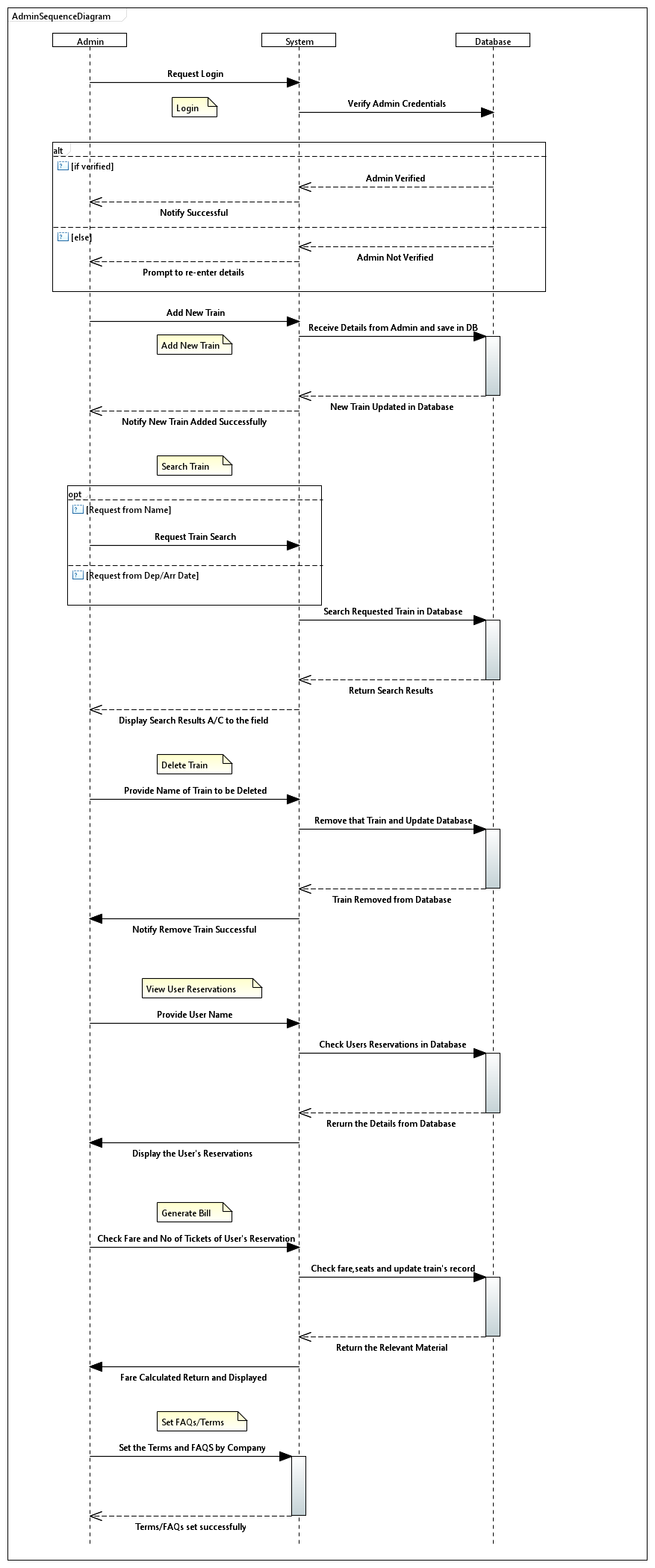
***8.2.1 Sequence Diagram***

*Since we have two modules, therefore separate diagrams are:-*

1. USER MODUE

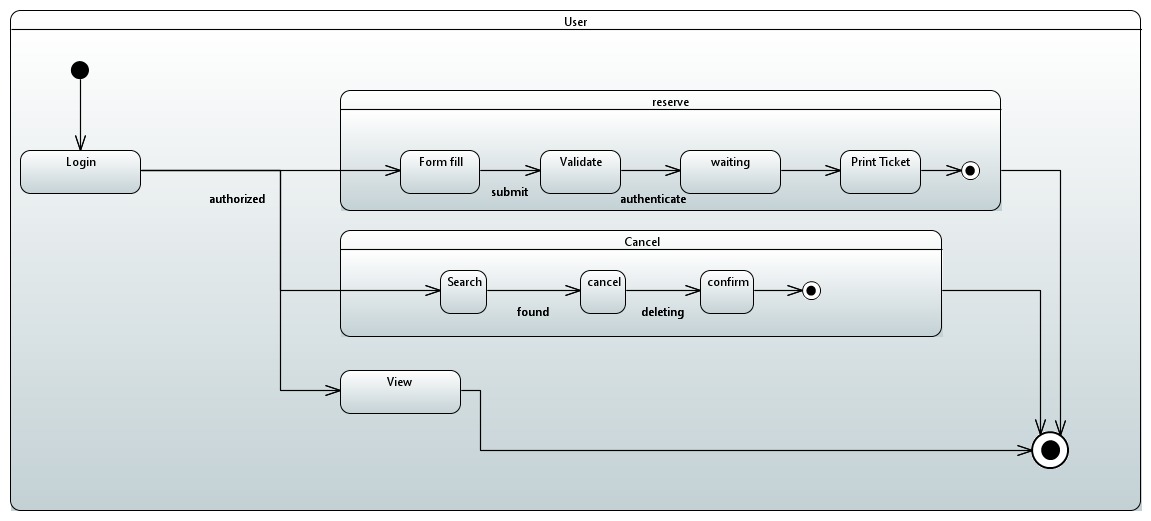
****

1. ADMIN MODULE

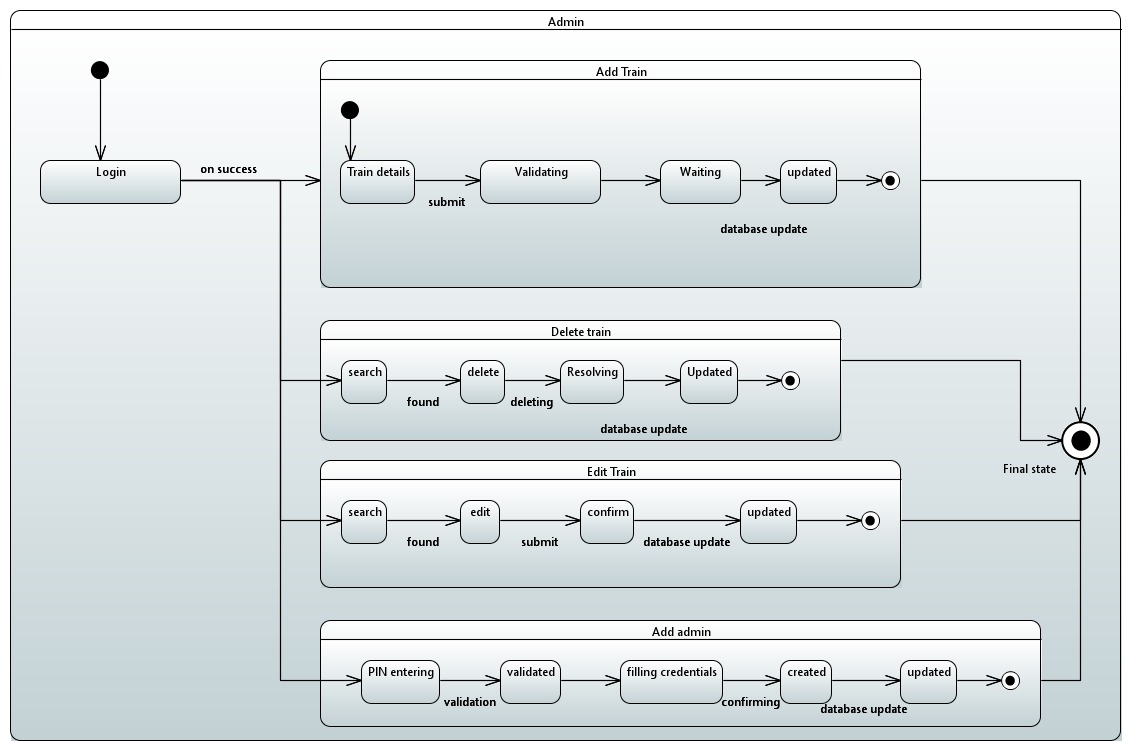
****

***8.2.2 State Diagram***

1. USER MODULE

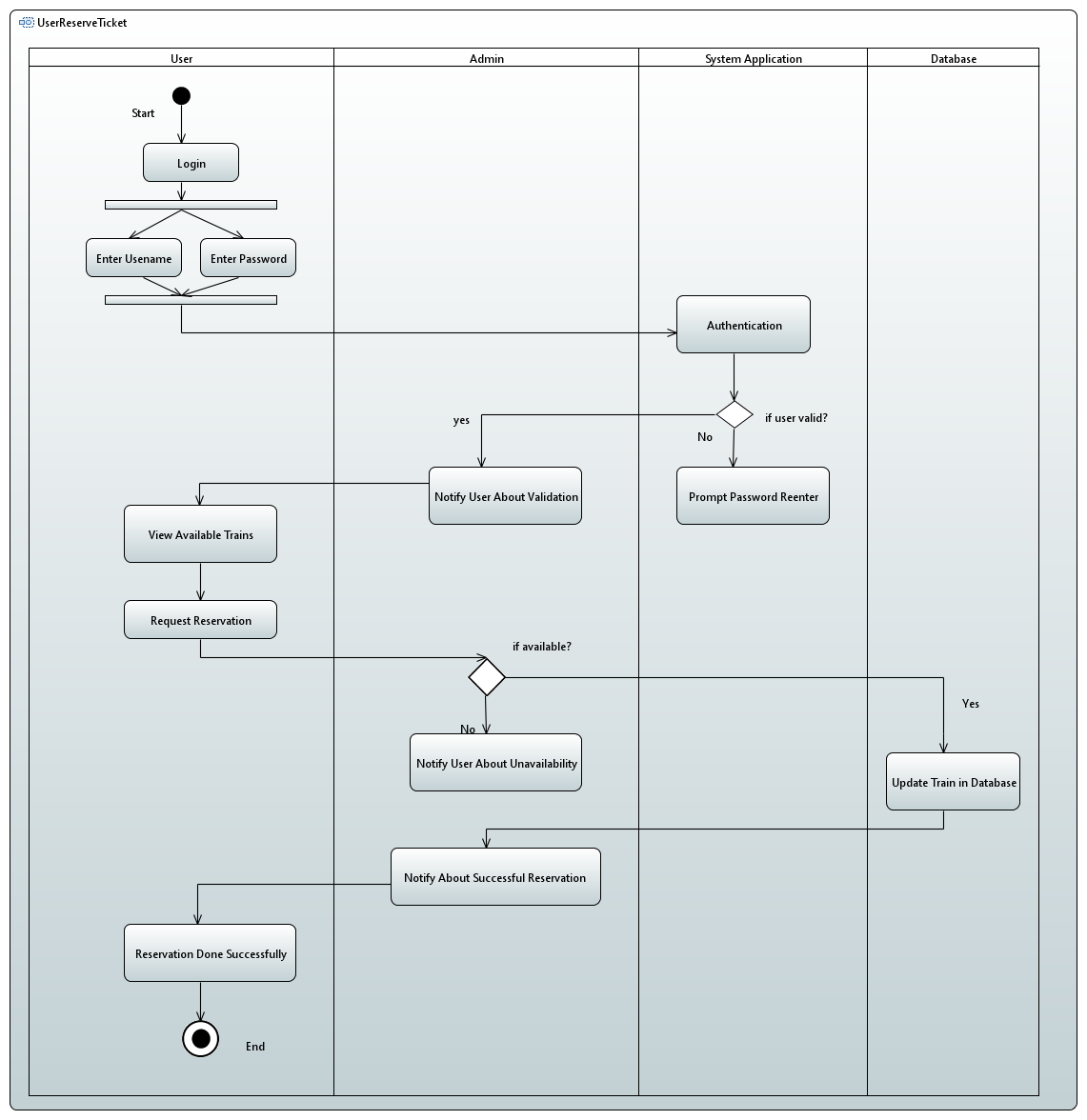
****

1. ADMIN

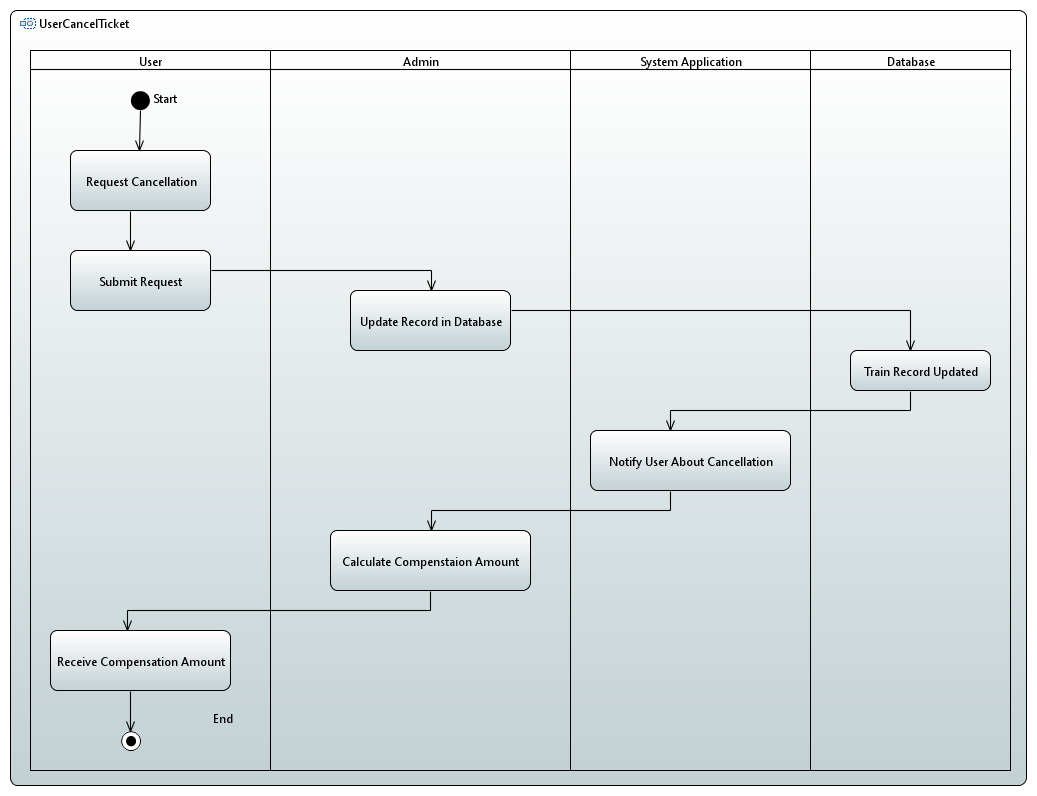
****

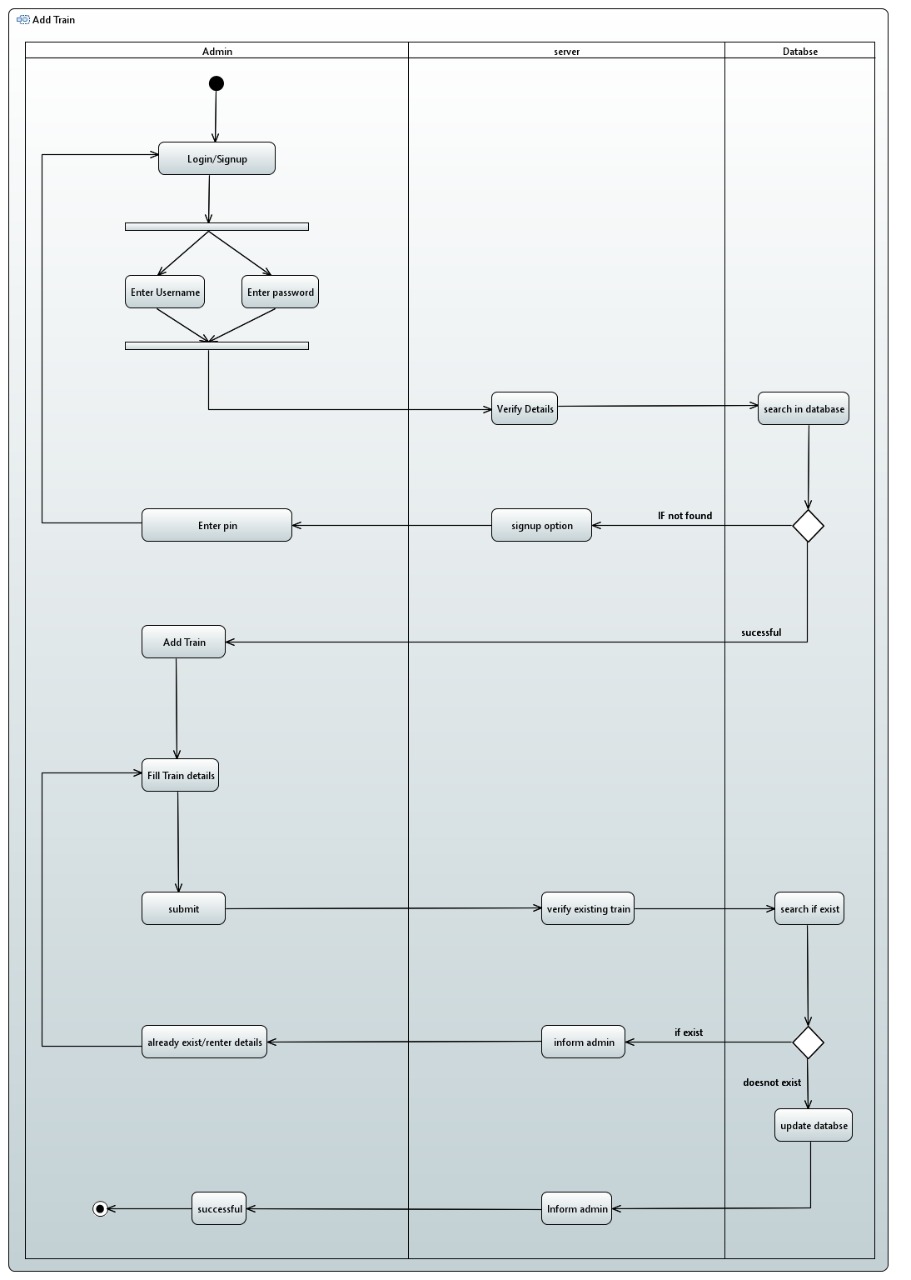
***8.2.3 Activity Diagram***

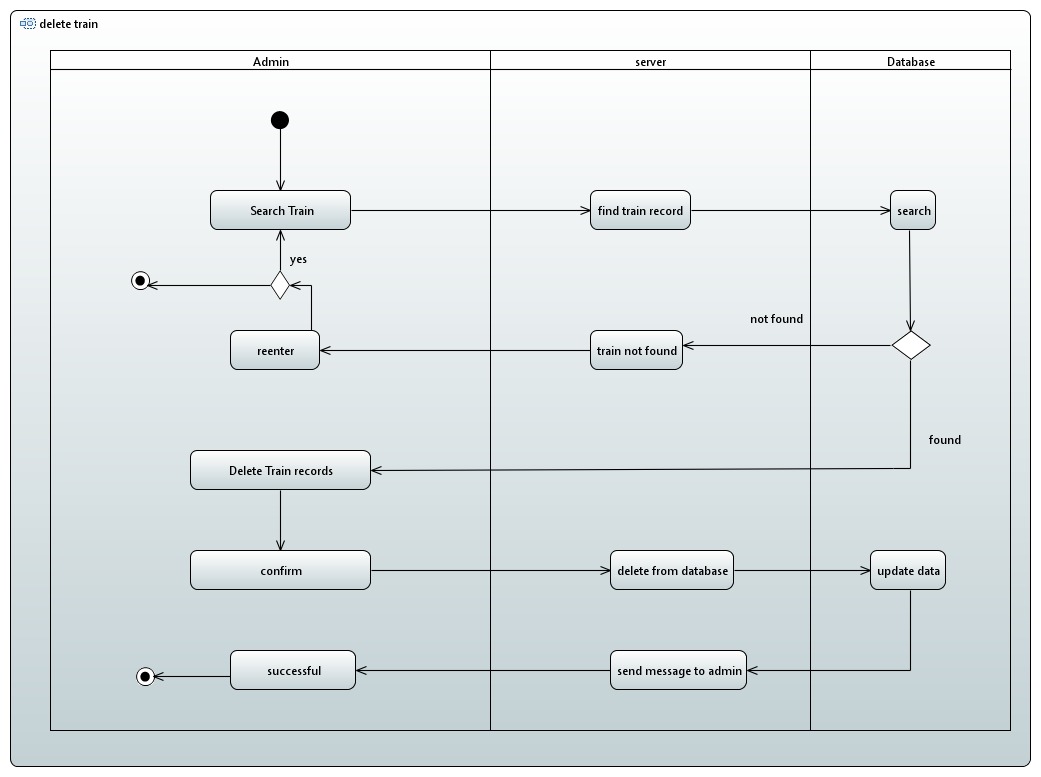
1. User reserve ticket

****

1. User cancel ticket

****

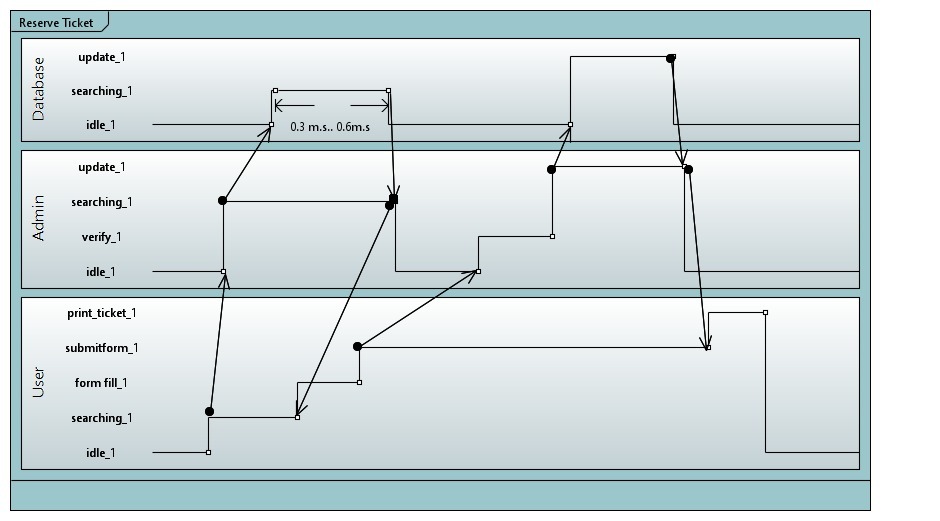
1. Admin add train****
2. Admin delete train

****

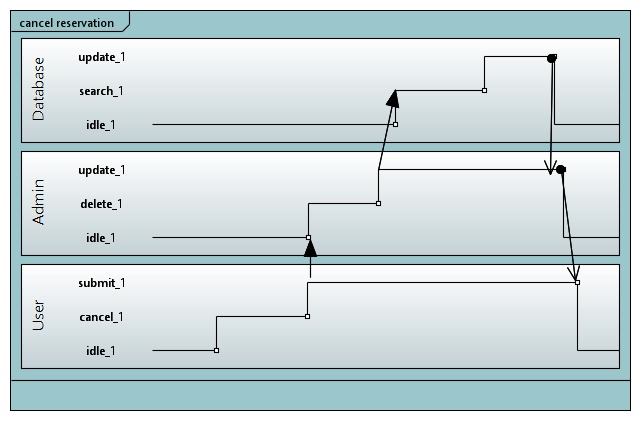
***8.2.4***

***Timing Diagram***

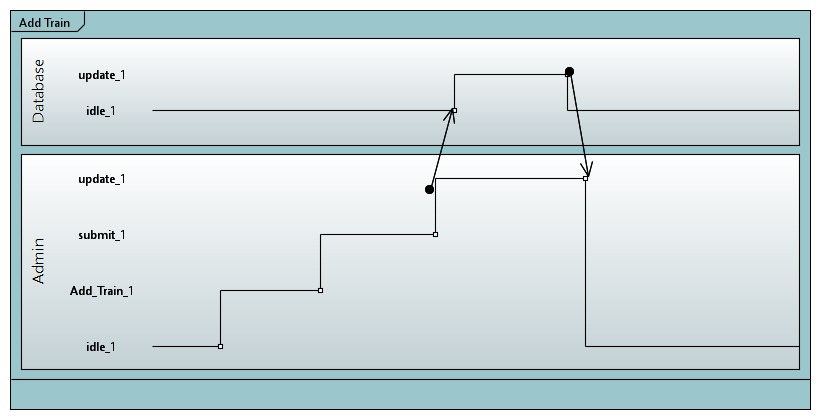
1. Reserve Ticket

****

1. Cancel ticket

****

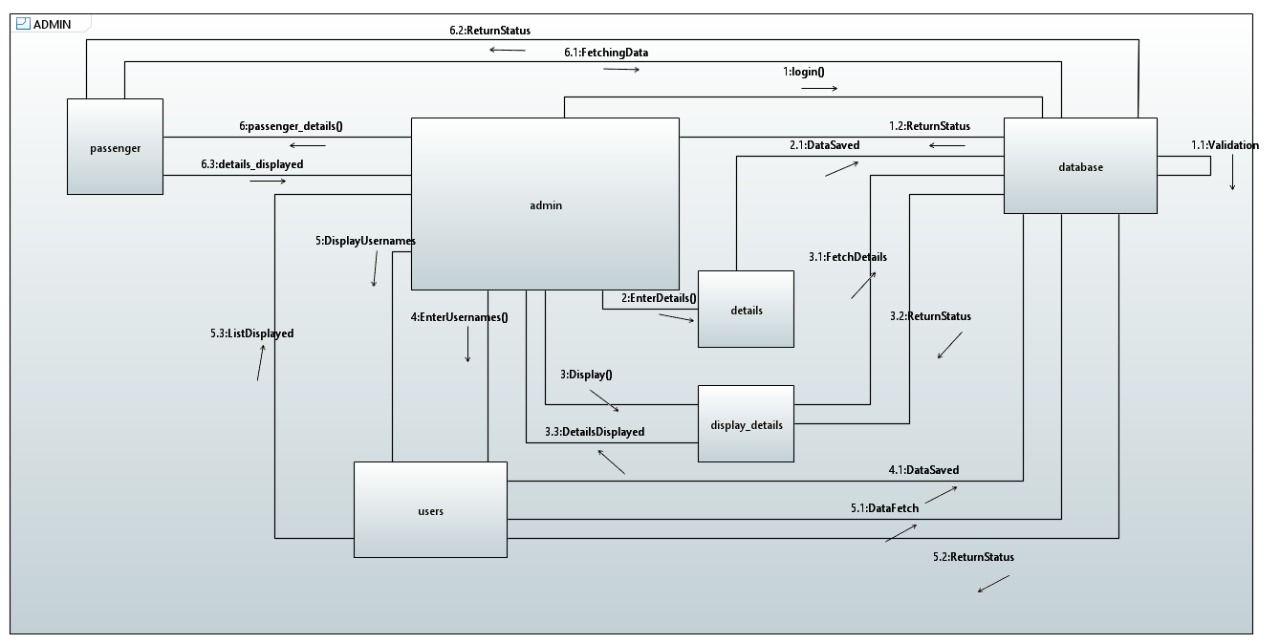
1. Add train

****

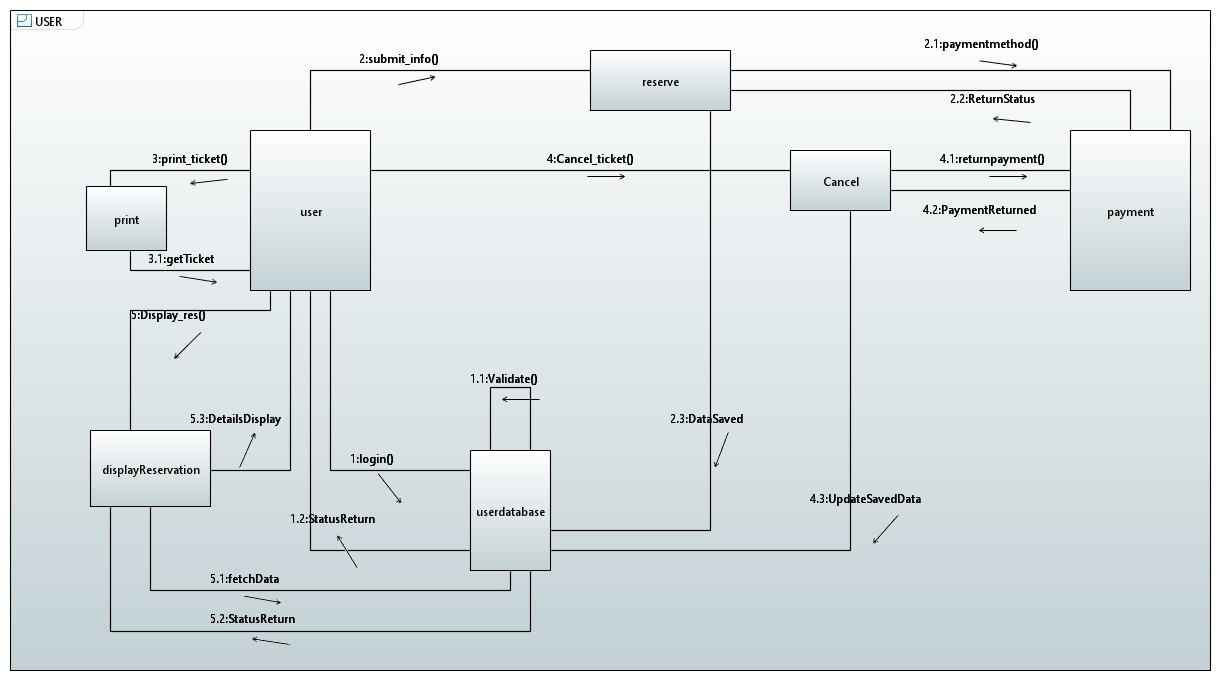
***8.2.5***

***Collaboration Diagram***

1. ***Admin***

****

1. User

****

***9. References***

References have not been taken specifically from any journals, publications except for YouTube videos to have detailed analysis being done on the model. In terms of design, it is completely genuine idea not having a working application working on similar principles or approaches

***10. Appendices***

*[not applicable]*