**Railway Reservation**

**System**

**Group number-13**

1. **Khair Alanam (121CS0137)**
2. **Suzen Akhtar(121CS0136)**
3. **Shawon Ghosh (121CS0169)**
4. **Chodisetty Bhuvaneswari (121CS0151)**
5. **Aryaman Khuntia (121CS0177)**
6. **Amit Kumar Sah (121CS0218)**

**INDEX OF THE REPORT**

1. **Software Requirement Specification for NIRCTC**
2. **Introduction**
3. **Functional Requirements**
4. **Non- Functional Requirements**
5. **System Constraints**
6. **Diagrams:-**
7. **Class Diagram**
8. **Sequence Diagram**
9. **Activity Diagram**
10. **Use case Diagram**
11. **Screenshots of code and the user page**
12. **Conclusion**

**SRS Document for Railway Reservation System:-**

**SRS Document is the first phase of this project. So, here is**

**the explanation of all the functional requirements and**

**non-functionals requirements of this system.**

**Introduction:**

**Purpose:-**

The purpose of this document is to specify the requirements of the Railway Reservation System, which allows users to book train tickets, and manage reservations.

**Scope:-**

The system will cover ticket booking, waiting list, cancellations, seat availability, and user management.

**Functional Requirements:**

**1.User Registration and Authentication:-**

The system shall allow users to register with a valid username and password and can use their credentials for logging in.

**2. Ticket Booking:-**

* Users can search for available trains based on source, destination, and date.
* The system shall display a list of available trains matching the search criteria.
* Users can select a train and book tickets for a specified number of passengers.

**3.Ticket Cancellation:**-

User can login and can cancel their tickets very efficiently using this Railway Reservation System (NIRCTC).

**Non-Functional Requirements:**

**1. Performance Requirements:**

* The system shall be able to handle large number of concurrent users without performance degradation.
* The system shall respond to user requests within a reasonable amount of time, typically within seconds.
* The system shall be able to process ticket bookings quickly and efficiently, handling multiple bookings simultaneously.

**2. Reliability Requirements:**

* The system shall be highly reliable and must not crash or lose data.
* The system shall have a high availability rate, ensuring continuous operation for extended periods.
* The system shall have a robust fault tolerance mechanism, able to recover from hardware or software failures without significant downtime.

**3. Security Requirements:**

* The system shall protect user data from unauthorized access, modification, or disclosure.
* The system shall implement secure authentication and authorization mechanisms to verify user identities and access privileges.
* The system shall employ encryption techniques to safeguard sensitive data, such as passenger information and payment details.

**4. Usability Requirements:**

* The system shall have an intuitive and user-friendly interface, easy to navigate and understand for users of all technical backgrounds.
* The system shall provide clear instructions and prompts to guide users through various tasks, such as searching for trains, booking tickets, and managing reservations.
* The system shall be accessible to users with disabilities, providing alternative input and output methods to accommodate diverse needs.

**5. Maintainability Requirements:**

* The system shall be well-structured and documented, allowing developers to easily understand, modify, and extend its functionality.
* The system shall adhere to industry-standard coding practices and utilize modular design principles to facilitate maintenance and updates.
* The system shall have comprehensive test suites and documentation to ensure the integrity of code changes and prevent regressions.

**6. Scalability Requirements:**

* The system shall be scalable to accommodate future growth in user base and data volume without significant performance degradation.
* The system's architecture shall allow for flexible resource allocation and expansion to handle increasing demand.
* The system shall be able to integrate with additional components and services as required to meet evolving business needs.

**System Constraints:**

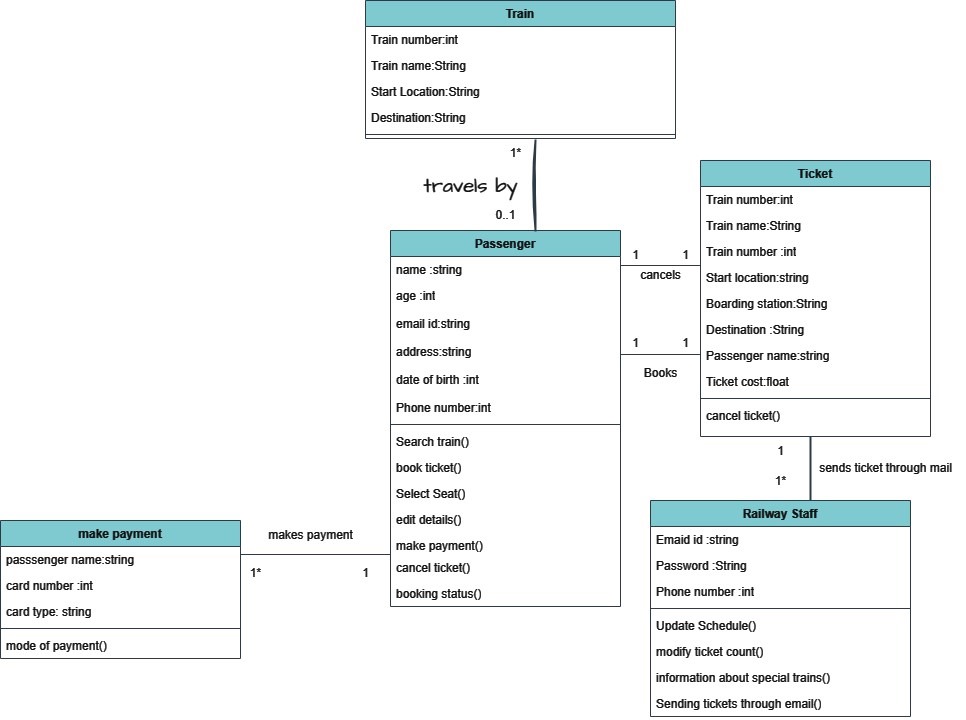
* Programming Language used- Python 3.12
* GUI Library- Tkinter, customtkinter
* Database- SQLite3
* PNR Generation- random, string
* Other Libraries- tkcalendar, time, os
* The system is compatible with:-

Operating System:- Windows, MacOS, Linux

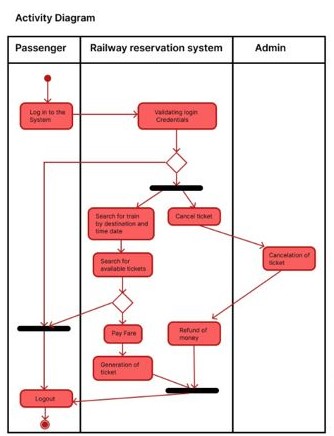
**Diagrams:-**

**Now coming to the diagrams section, the first diagram will be the Class diagram of the Railway Reservation System.**

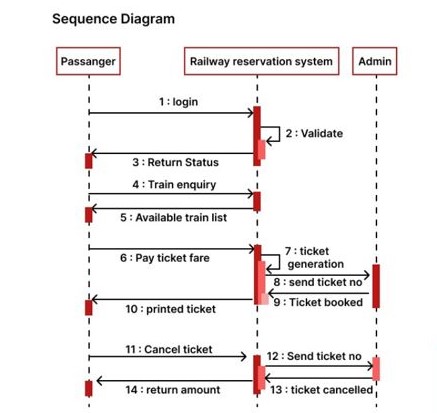
**Class Diagram:-**

****

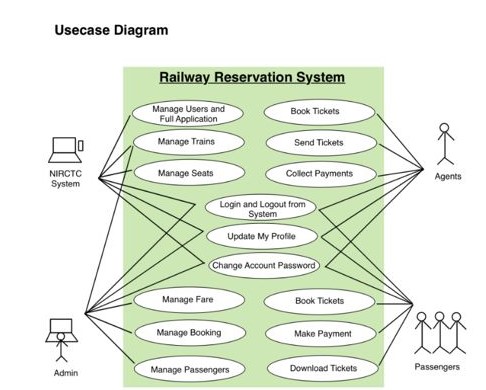
**Then after the class diagram there is the Activity diagram.**

****

**Then , here is a Sequence Diagram**

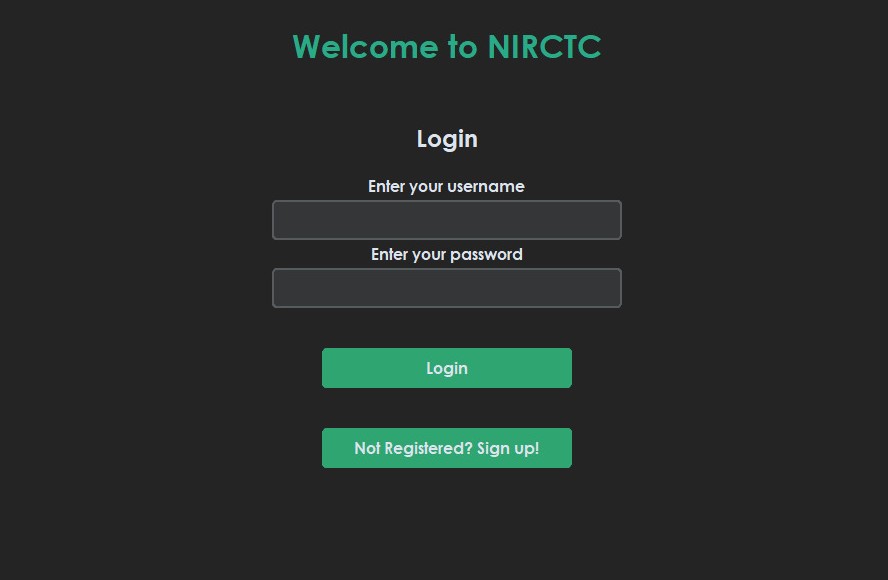
****

**Use case Diagram:-**

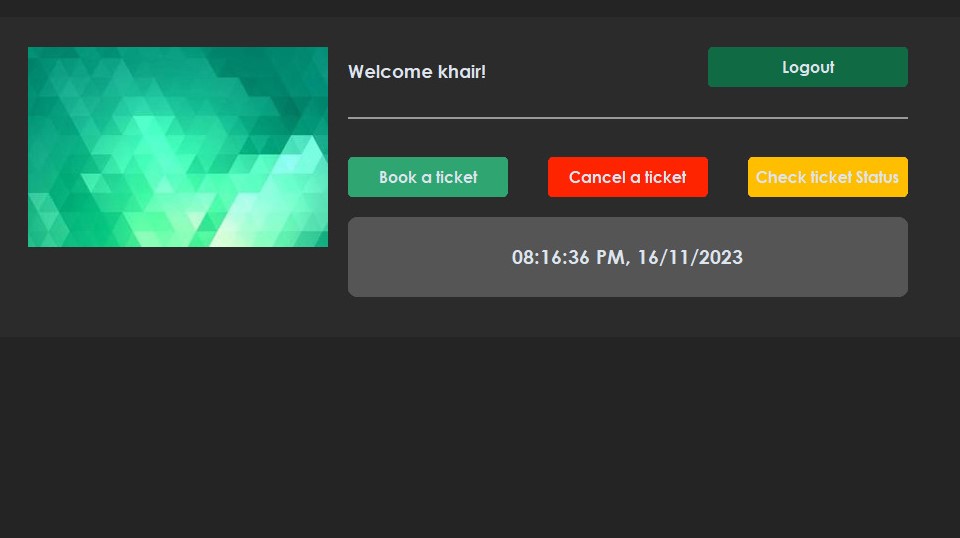
****

**User page Screenshots:-**

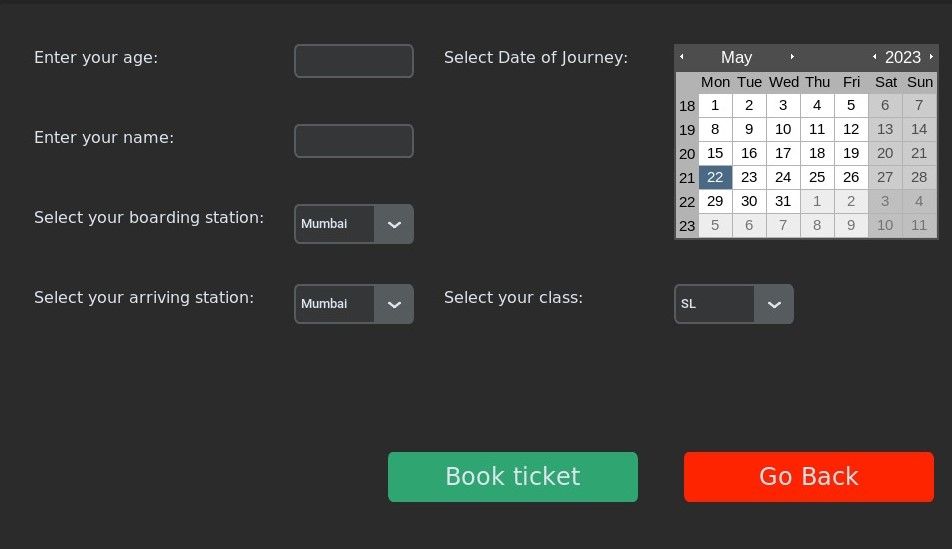
**Register and Login Page**

****

**Select any one option**

****

**Enter all the details.**

****

**Ticket Booked Successfully**

**A screenshot of a computer

Description automatically generated**

**Checking for the Ticket status using PNR number**

**A screenshot of a computer

Description automatically generated**

**Conclusion:-**

**In conclusion, the Railway Reservation System report represents a comprehensive solution designed to revolutionize the process of booking and managing train tickets. By amalgamating user-friendly interfaces, robust backend functionalities, and a secure transaction framework, this system aims to streamline ticket reservations while enhancing overall efficiency and customer satisfaction.**