Diagrams

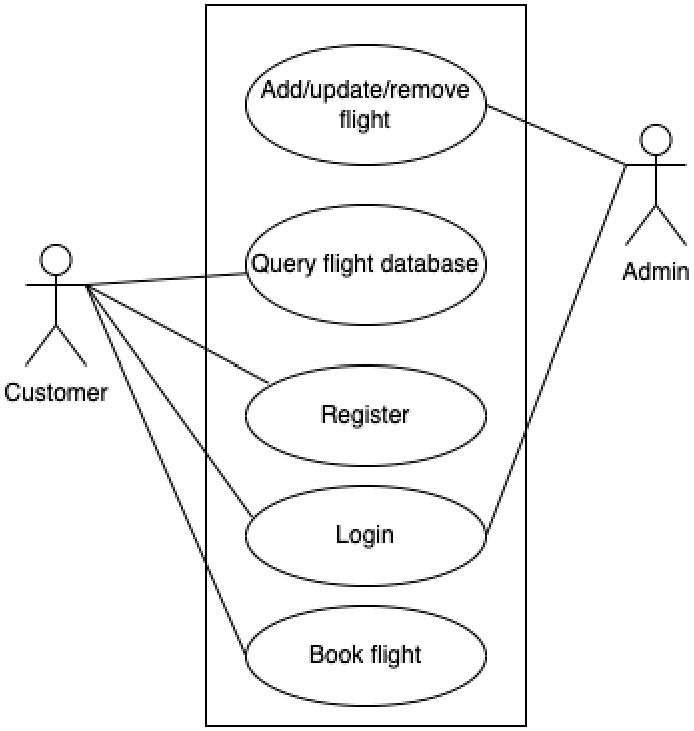


Fig 1: Use Case Diagram

Customers and travel agencies interact with the Flight Booker with separate capabilities. Travel agents can add, update, and remove flights, while customers are able to search for flights based on their arrival and departure times and locations.

Use Case Name	Query Flight Database
Scenario	Customer queries flight database
Triggering Event	Customer presses “search” button after inputting arrival and departure locations and times.
Brief Description	When the customer inputs their relevant information for a desired flight, the relevant flights that meet their requirements are displayed.
Actors	Customer
Related Use Cases	Book flight
Stakeholders	Admin: to keep available flights updated for correctness
Preconditions	Customer must exist. Customer must be logged in. Customer’s flight info must match at least one flight option.

Postconditions	Flights displayed must be correct and available for customer to book.
Flow of Activities	1. Customer enters desired flight information. 2. Database is queried, and a list of flights that meet input values is displayed. 3. Customer can select a flight and press the "book" button to choose this particular flight.
Exceeption Conditions	If the customer incorrectly inputs their locations and times, they are asked to retry. If flight info does not match any flights in database, blank page specifying that there are no available flights meeting the description is displayed.

Fig 2: Use Case Description for Flight Querying

A primary action in the Flight Booker is querying the flight database. This allows the customers to find flights that meet their requirements. By specifying the arrival and departure locations and times, the user can view all available flights that match their preferred logistics.

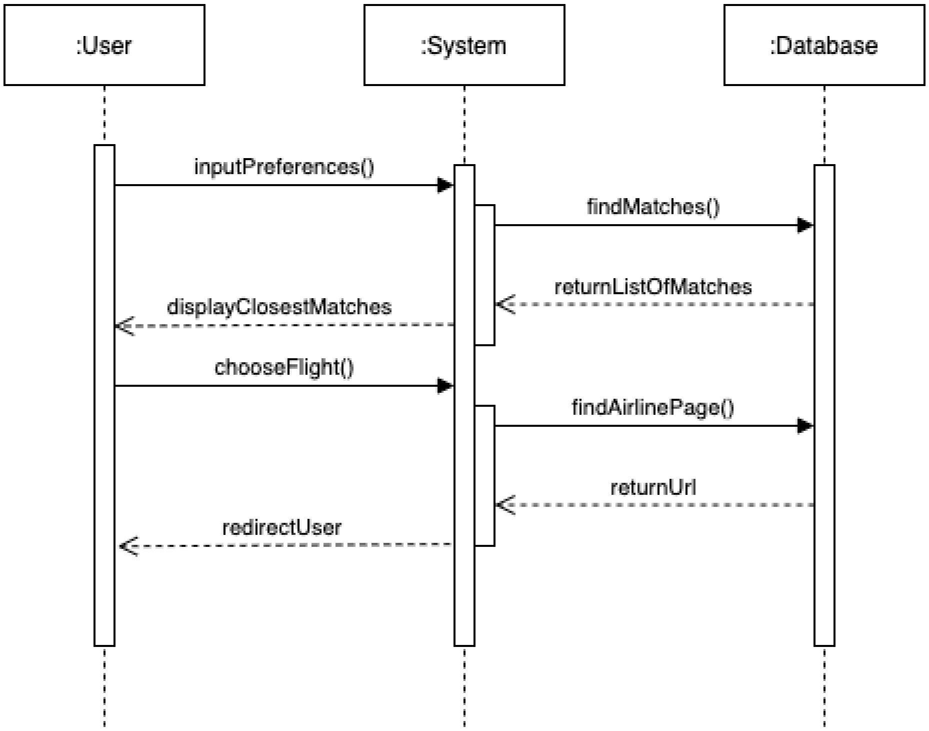


Fig 3: Sequence Diagram for Flight Querying

The user enters their arrival and departure locations into the UI. The system receives these logistics, and queries the database for potential flights that match these requirements. The database returns a table of potential matches to the system, which displays the available flights to the user. The user can then select a flight to book.

Use Case Name	Login
Scenario	Customer logs into the Flight Booker
Triggering Event	Customer presses “login” button after accessing the home page.
Brief Description	When the customer inputs their email and password, the user database is queried for their credentials. If the email and password are present, then the user proceeds to the flight query page.
Actors	Customer
Related Use Cases	Query flight database
Stakeholders	Admin: to keep users credentials updated.
Preconditions	Customer must exist. Customer must correctly enter email and password.
Postconditions	Flights must be able to be queried based on the user’s input parameters.
Flow of Activities	4. Customer clicks the “login” button. 5. Customer enters email. 6. Customer enters password. 7. Customer clicks the “login” button on the login page to proceed. 8. Credentials are verified in order for customer to continue to next page.
Exceeption Conditions	If the customer incorrectly inputs their email and password, they are asked to retry. If customer does not yet have an account, they will instead click the “register” button.

Fig 4: Use Case Description for Login

The customer must login to the Flight Booker in order to query the flight database. By entering their correct email and password, the customer can then navigate to the flight query page. Registration is a separate page, which is the appropriate choice for new users.

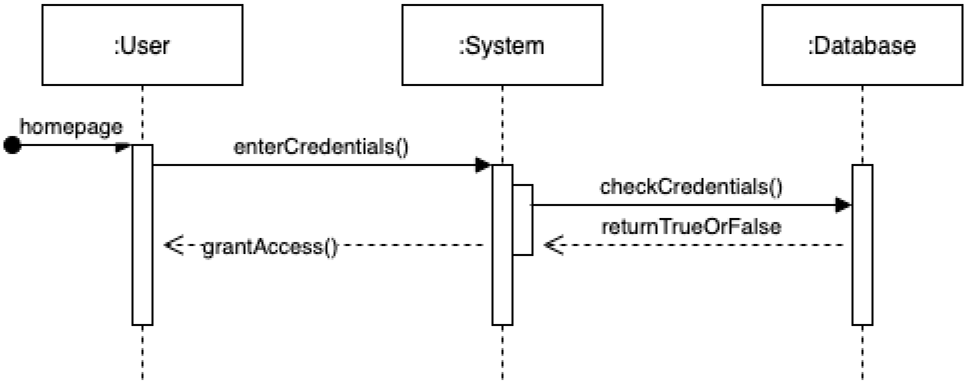


Fig 5: Sequence Diagram for Login

When a user submits their login information, the credentials are sent to the system. The system then checks these credentials against the database's table of user. The database then responds to the system confirming or denying the existence of the account. The system then either logs the user in, or asks them to reenter their credentials.

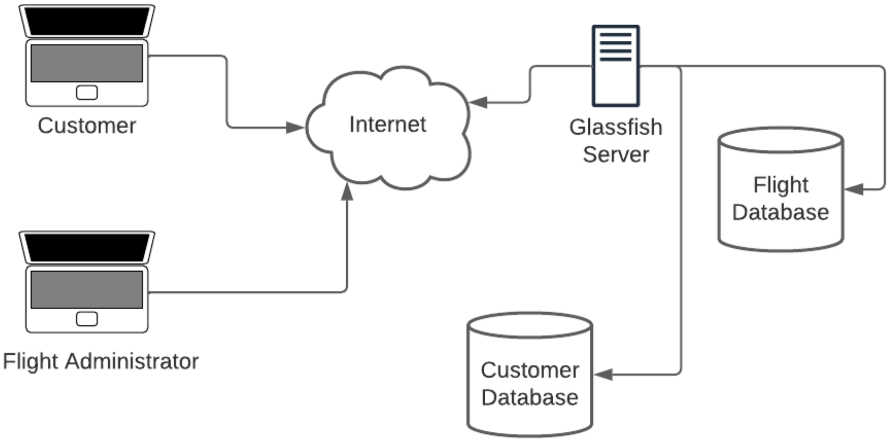


Fig 6: System Architecture

Both flight administrators and customers access the internet in order to interact with the Glassfish server. The server will then interact with the flight and customer tables in the database in order to fulfill user commands.

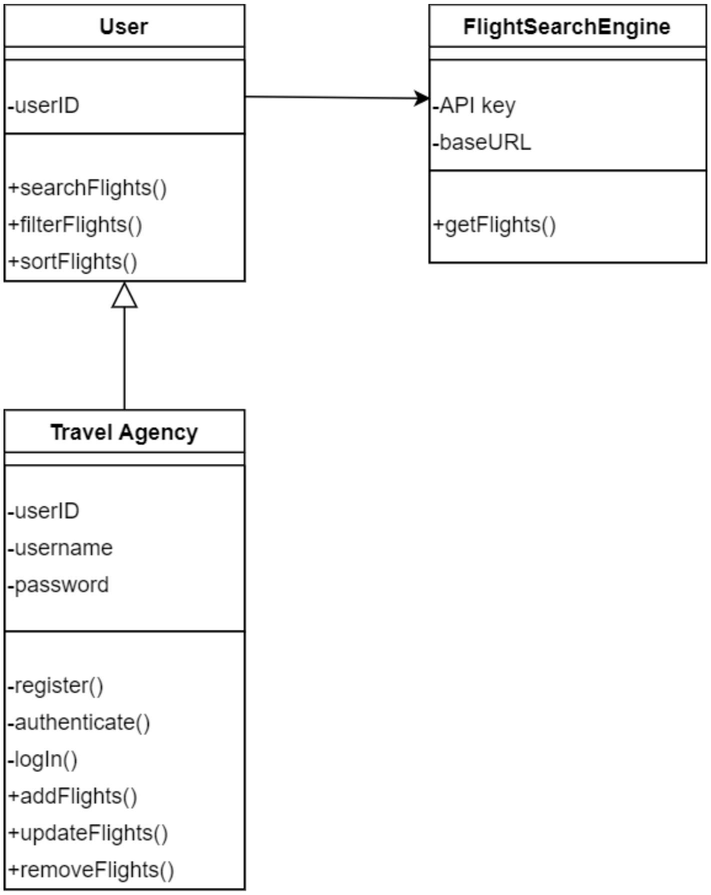
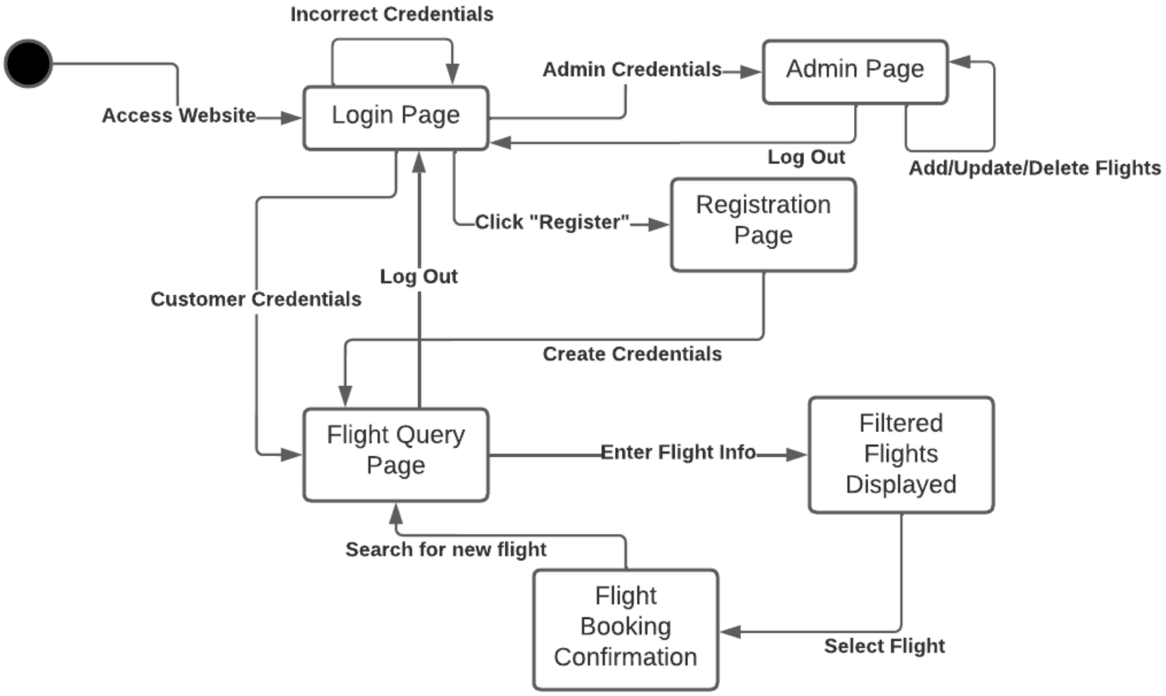


Fig 7: Class Diagram



State-Machine Diagram

The login page serves as the default state of the Flight Booker. Administrators, once logging in with their credentials, will see a different page from customers; administrators can add, update, and delete flights, whereas customers can query the flight database and place bookings.

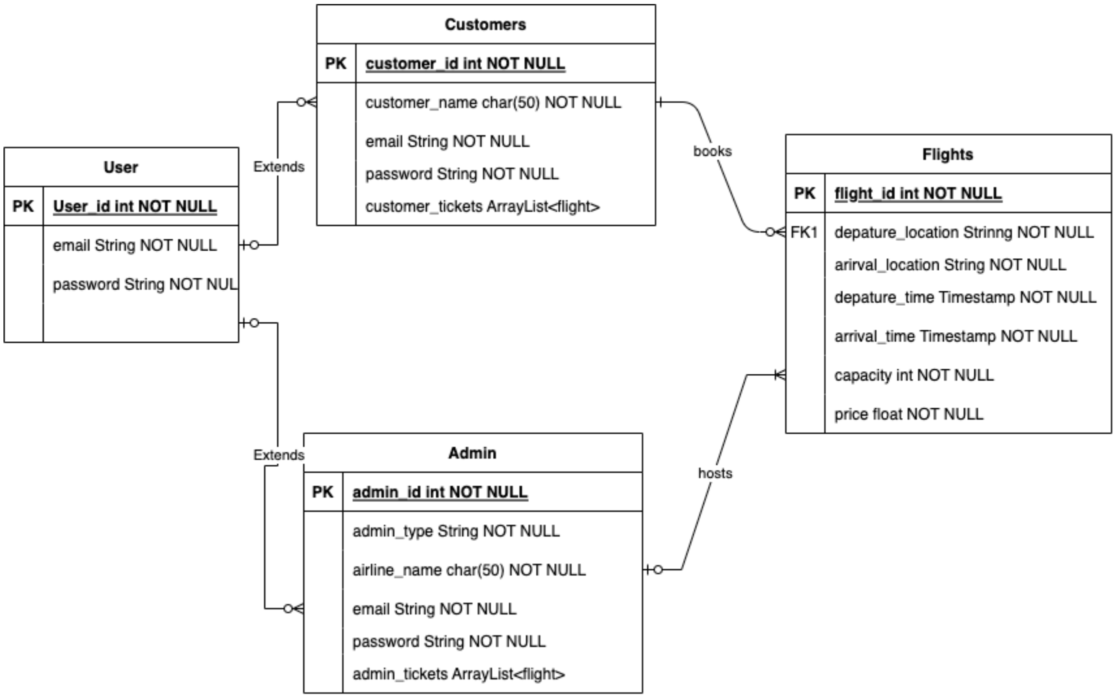


Fig 9: ER Diagram

The four tables in the database are Admin, Flights, Customers, and User. After a successful login, Admins are taken to a flight modification page, whereas Customers are directed to a flight query page.