Project title	**Airline Reservation System**
Author(s)	**Maniu Maria-Diana**
Group	**30421**

1. Task Description

Overview

The Airline Reservation System is a Java-based application that allows users to manage flights, passengers, tickets, and seat reservations. The system uses a graphical user interface (GUI) built with Java Swing, integrating a PostgreSQL database for backend storage.

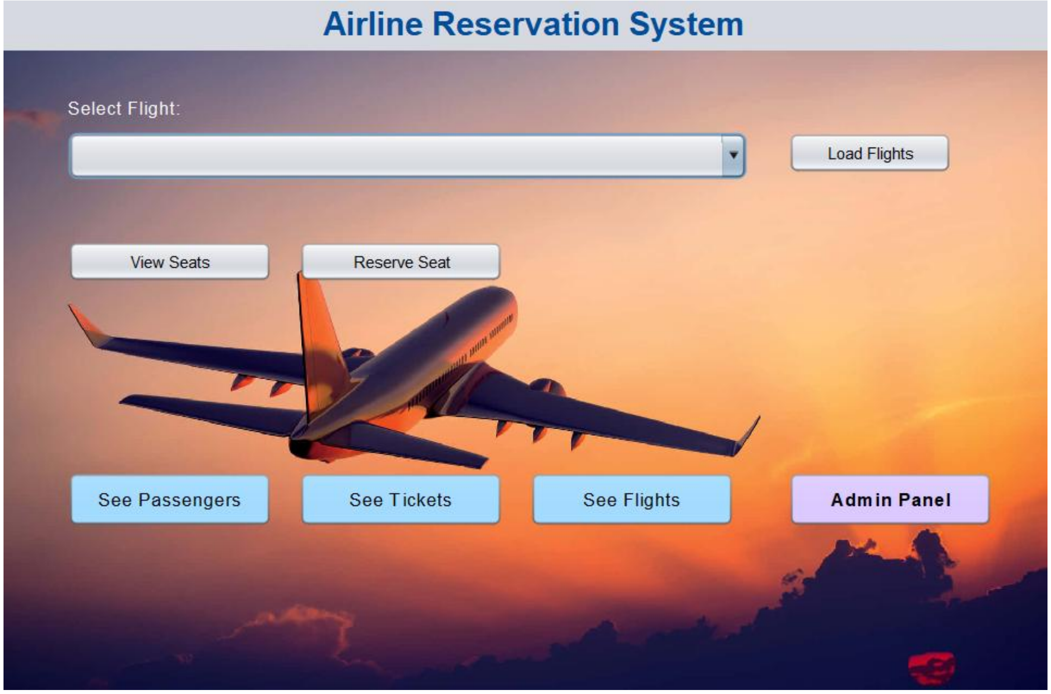
Main Menu

The main menu offers the following options:

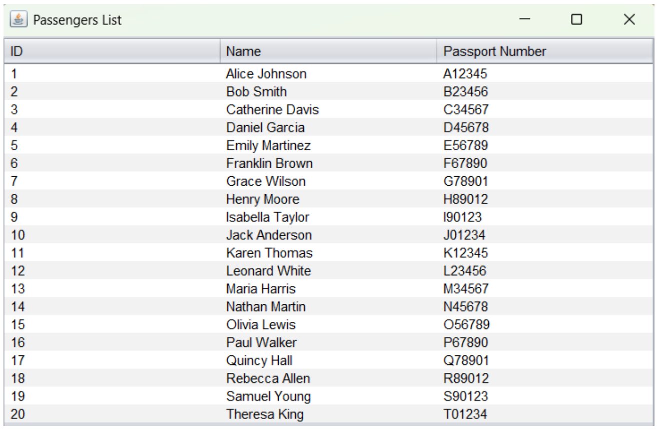
- **Book Flight**: Select a flight, view available seats, and book a ticket (reserve a seat).
- **View Flights**: Displays a list of all flights (both available and not available).
- **View Tickets:** Displays all booked tickets.
- **View Passengers**: Displays a list of the passengers stored in the database.
- **Admin Panel**: Accessible via password, allows flight management.
- **Load Flights**: loads the data from the database.

GUI Elements

- **Background Image**: Aviation-themed background for aesthetic appeal.
- **Buttons**: Navigate between functionalities, styled consistently with hover effects.
- **Tables**: Display flight, passenger, ticket, and seat data.
- **Input Dialogs**: Collect user input for booking or admin tasks.

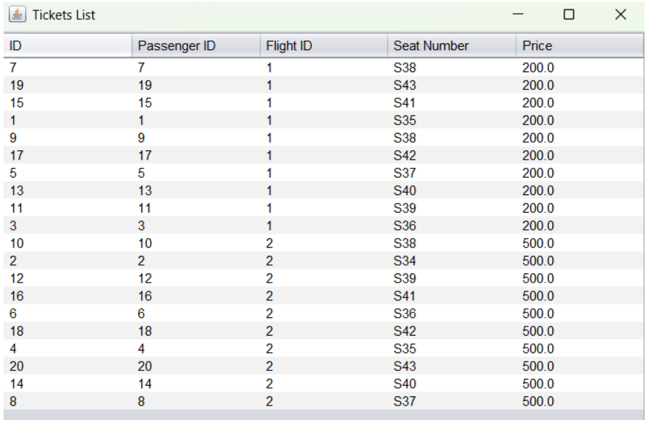


The Passengers table



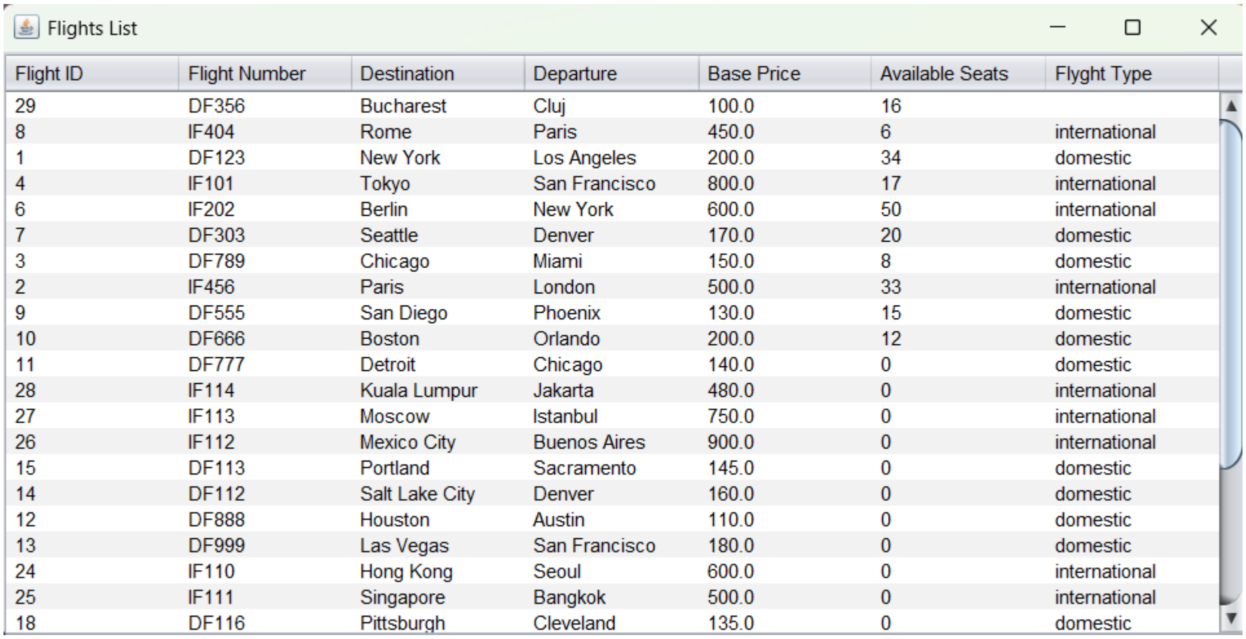
ID	Name	Passport Number
1	Alice Johnson	A12345
2	Bob Smith	B23456
3	Catherine Davis	C34567
4	Daniel Garcia	D45678
5	Emily Martinez	E56789
6	Franklin Brown	F67890
7	Grace Wilson	G78901
8	Henry Moore	H89012
9	Isabella Taylor	I90123
10	Jack Anderson	J01234
11	Karen Thomas	K12345
12	Leonard White	L23456
13	Maria Harris	M34567
14	Nathan Martin	N45678
15	Olivia Lewis	O56789
16	Paul Walker	P67890
17	Quincy Hall	Q78901
18	Rebecca Allen	R89012
19	Samuel Young	S90123
20	Theresa King	T01234

The Tickets table



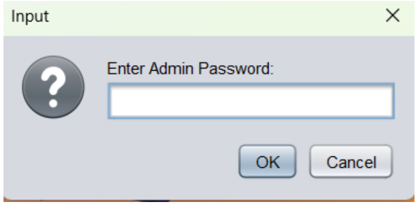
ID	Passenger ID	Flight ID	Seat Number	Price
7	7	1	S38	200.0
19	19	1	S43	200.0
15	15	1	S41	200.0
1	1	1	S35	200.0
9	9	1	S38	200.0
17	17	1	S42	200.0
5	5	1	S37	200.0
13	13	1	S40	200.0
11	11	1	S39	200.0
3	3	1	S36	200.0
10	10	2	S38	500.0
2	2	2	S34	500.0
12	12	2	S39	500.0
16	16	2	S41	500.0
6	6	2	S36	500.0
18	18	2	S42	500.0
4	4	2	S35	500.0
20	20	2	S43	500.0
14	14	2	S40	500.0
8	8	2	S37	500.0

The Flights table

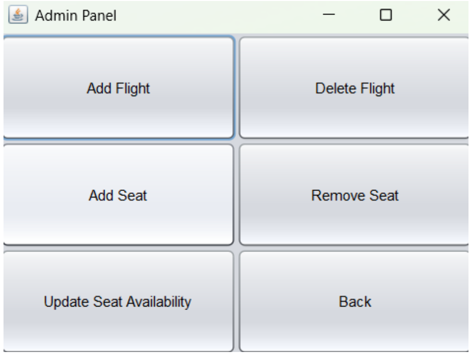


Flight ID	Flight Number	Destination	Departure	Base Price	Available Seats	Flyght Type
29	DF356	Bucharest	Cluj	100.0	16	
8	IF404	Rome	Paris	450.0	6	international
1	DF123	New York	Los Angeles	200.0	34	domestic
4	IF101	Tokyo	San Francisco	800.0	17	international
6	IF202	Berlin	New York	600.0	50	international
7	DF303	Seattle	Denver	170.0	20	domestic
3	DF789	Chicago	Miami	150.0	8	domestic
2	IF456	Paris	London	500.0	33	international
9	DF555	San Diego	Phoenix	130.0	15	domestic
10	DF666	Boston	Orlando	200.0	12	domestic
11	DF777	Detroit	Chicago	140.0	0	domestic
28	IF114	Kuala Lumpur	Jakarta	480.0	0	international
27	IF113	Moscow	Istanbul	750.0	0	international
26	IF112	Mexico City	Buenos Aires	900.0	0	international
15	DF113	Portland	Sacramento	145.0	0	domestic
14	DF112	Salt Lake City	Denver	160.0	0	domestic
12	DF888	Houston	Austin	110.0	0	domestic
13	DF999	Las Vegas	San Francisco	180.0	0	domestic
24	IF110	Hong Kong	Seoul	600.0	0	international
25	IF111	Singapore	Bangkok	500.0	0	international
18	DF116	Pittsburgh	Cleveland	135.0	0	domestic

The Admin Pannel



Enter Admin Password:



2. Class Discovery

CRC Card 1

Card name	**Main**		
Superclasses	None		
Subclasses	None		
Responsibilities	Start the application	Collaborators	GUI

CRC Card 2

Card name	**GUI**		
Superclasses	None		
Subclasses	BackgroundPannel		
Responsibilities	Show the interface Handles the menus and dialogs	Collaborators	Main Flight DatabaseConnection AdminPannel

CRC Card 3

Card name	**Flight**		
Superclasses	None		
Subclasses	DomesticFlight, InternationalFlight		
Responsibilities	Abstract base class for flights	Collaborators	DatabaseConnection AdminPannel GUI

CRC Card 4

Card name	**DomesticFlight**		
Superclasses	Flight		
Subclasses	None		
Responsibilities	Start the application	Collaborators	Flight

CRC Card 5

Card name	**InternationalFlight**		
Superclasses	Flight		
Subclasses	None		
Responsibilities	Calculates the total price of the ticket with the international fee	Collaborators	Flight

CRC Card 6

Card name	**Passenger**		
Superclasses	None		
Subclasses	None		
Responsibilities	Represents passenger details	Collaborators	DatabaseConnection GUI

CRC Card 7

Card name	**Seat**		
Superclasses	None		
Subclasses	None		
Responsibilities	Represents seat details	Collaborators	Flight DatabaseConnection GUI

CRC Card 8

Card name	**Ticket**		
Superclasses	None		
Subclasses	None		
Responsibilities	Represents ticket details	Collaborators	Flight DatabaseConnection GUI

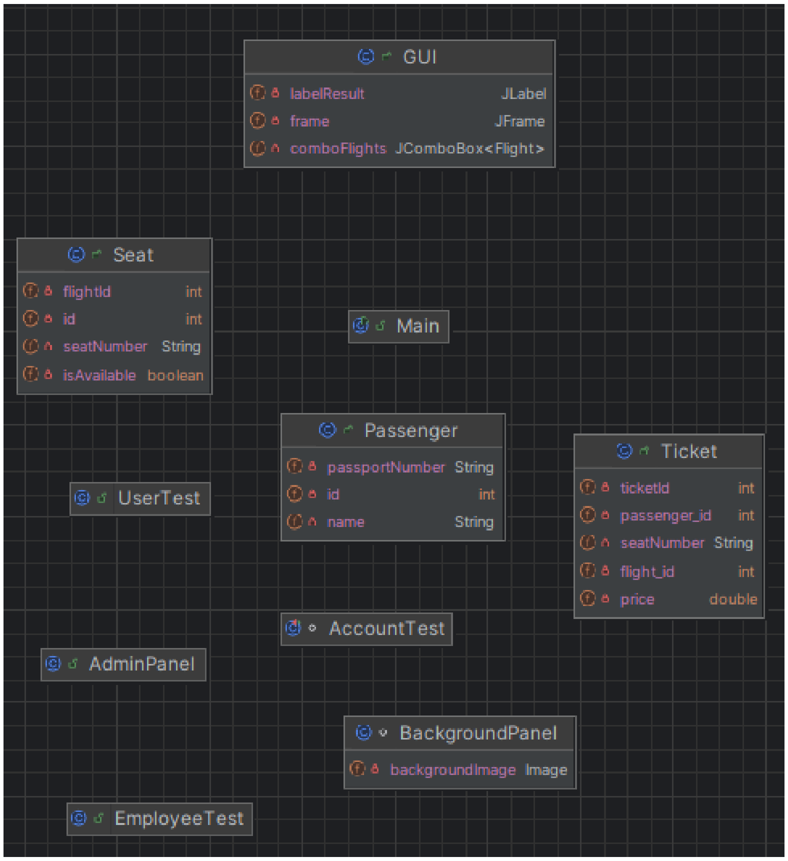
CRC Card 9

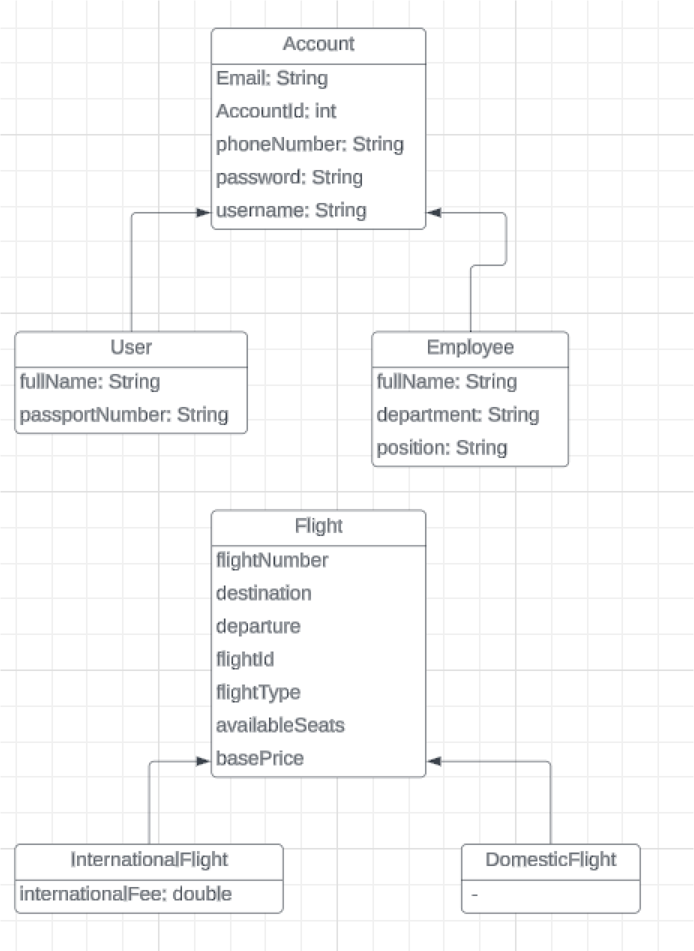
Card name	**DatabaseConnection**		
Superclasses	None		
Subclasses	None		
Responsibilities	Manages database interactions: fetch, add, update, delete for flights, tickets, seats etc.	Collaborators	Flight Seat Ticket Passenger

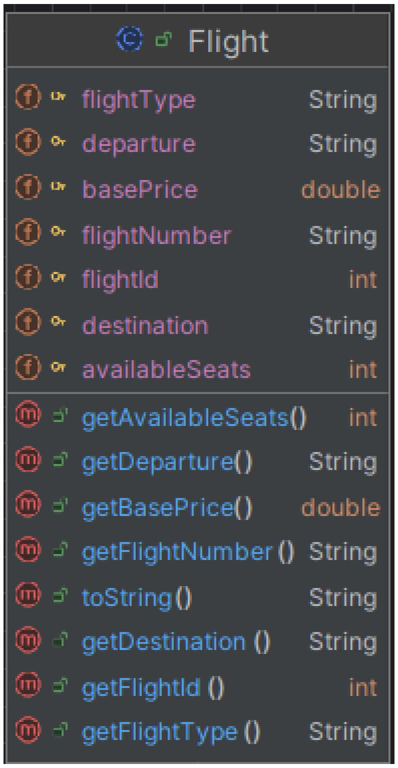
CRC Card 10

Card name	**AdminPannel**		
Superclasses	None		
Subclasses	None		
Responsibilities	Provides administrative functionalities: add, edit and delete flights	Collaborators	Flight Seat GUI

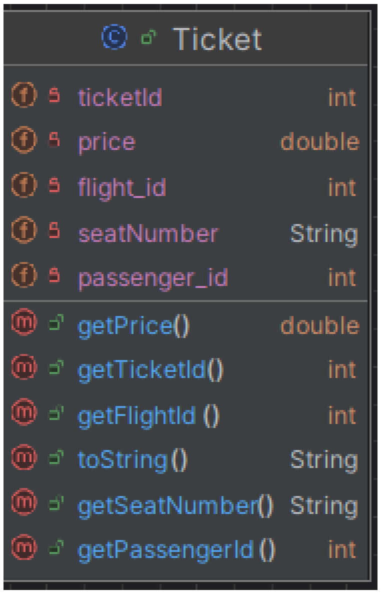
3. Class Diagram



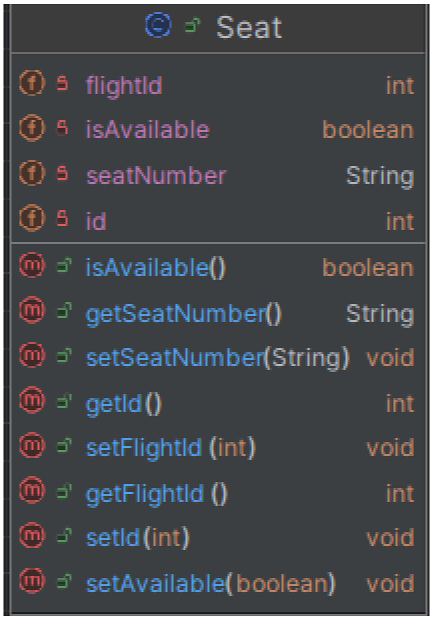




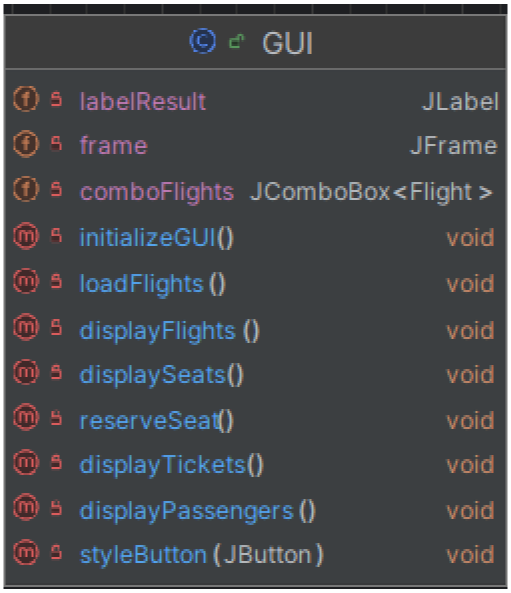
```
Flight
flightType              String
departure               String
basePrice               double
flightNumber            String
flightId                int
destination             String
availableSeats          int
getAvailableSeats()     int
getDeparture()          String
getBasePrice()          double
getFlightNumber ()      String
toString ()             String
getDestination ()       String
getFlightId ()          int
getFlightType ()        String
```



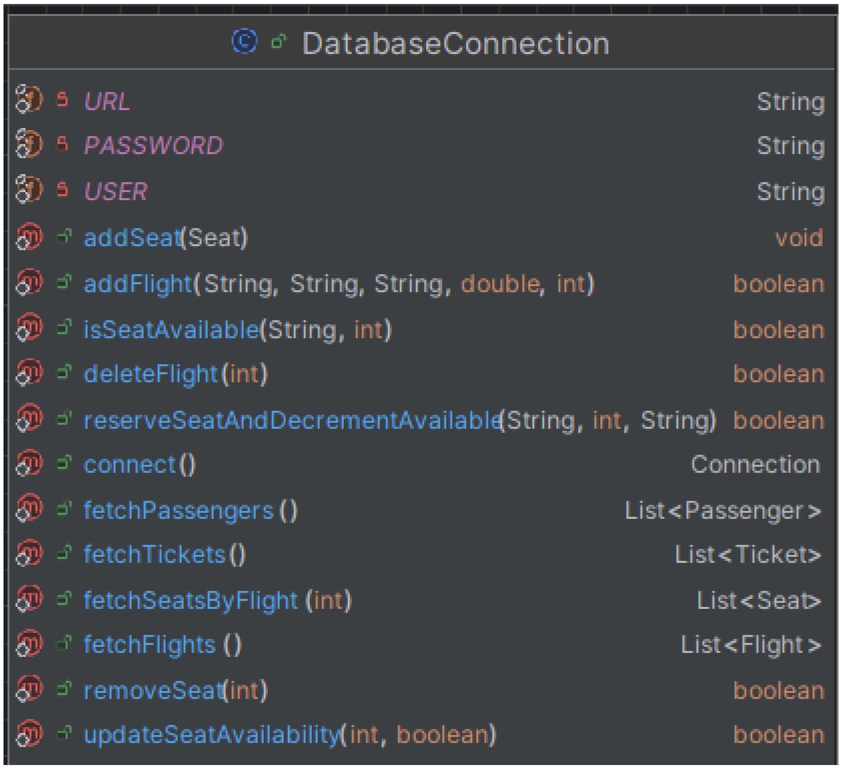
```
Ticket
ticketId                int
price                   double
flight_id               int
seatNumber              String
passenger_id            int
getPrice()              double
getTicketId()           int
getFlightId ()          int
toString ()             String
getSeatNumber()         String
getPassengerId ()       int
```



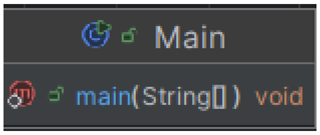
```
Seat
flightId                    int
isAvailable                 boolean
seatNumber                  String
id                          int
isAvailable()               boolean
getSeatNumber()             String
setSeatNumber(String)       void
getId()                     int
setFlightId (int)           void
getFlightId ()              int
setId(int)                  void
setAvailable(boolean)       void
```



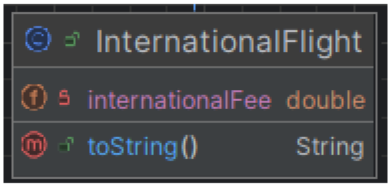
```
GUI
labelResult                 JLabel
frame                       JFrame
comboFlights     JComboBox<Flight>
initializeGUI()             void
loadFlights ()              void
displayFlights ()           void
displaySeats()              void
reserveSeat()               void
displayTickets()            void
displayPassengers ()        void
styleButton (JButton)       void
```



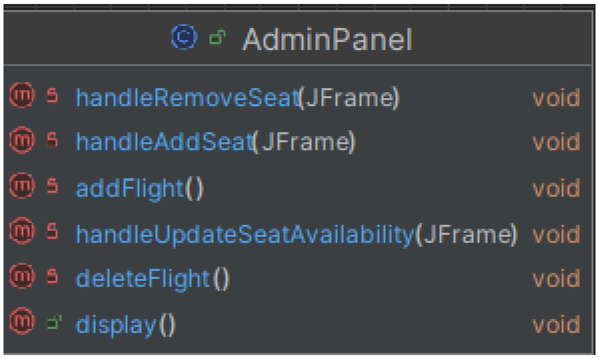
```
DatabaseConnection
URL                                                        String
PASSWORD                                                   String
USER                                                       String
addSeat(Seat)                                              void
addFlight(String, String, String, double, int)             boolean
isSeatAvailable(String, int)                               boolean
deleteFlight(int)                                          boolean
reserveSeatAndDecrementAvailable(String, int, String)      boolean
connect()                                                  Connection
fetchPassengers ()                                         List<Passenger>
fetchTickets()                                             List<Ticket>
fetchSeatsByFlight (int)                                   List<Seat>
fetchFlights ()                                            List<Flight>
removeSeat(int)                                            boolean
updateSeatAvailability(int, boolean)                       boolean
```



```
Main
main(String[] )  void
```

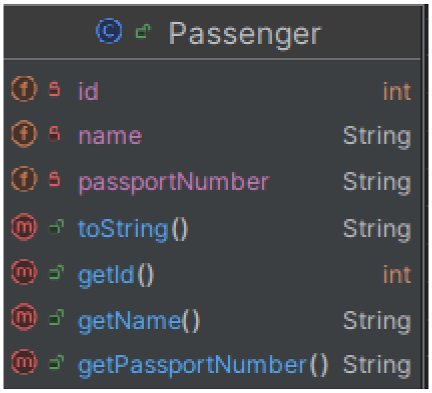


```
InternationalFlight
internationalFee   double
toString()         String
```



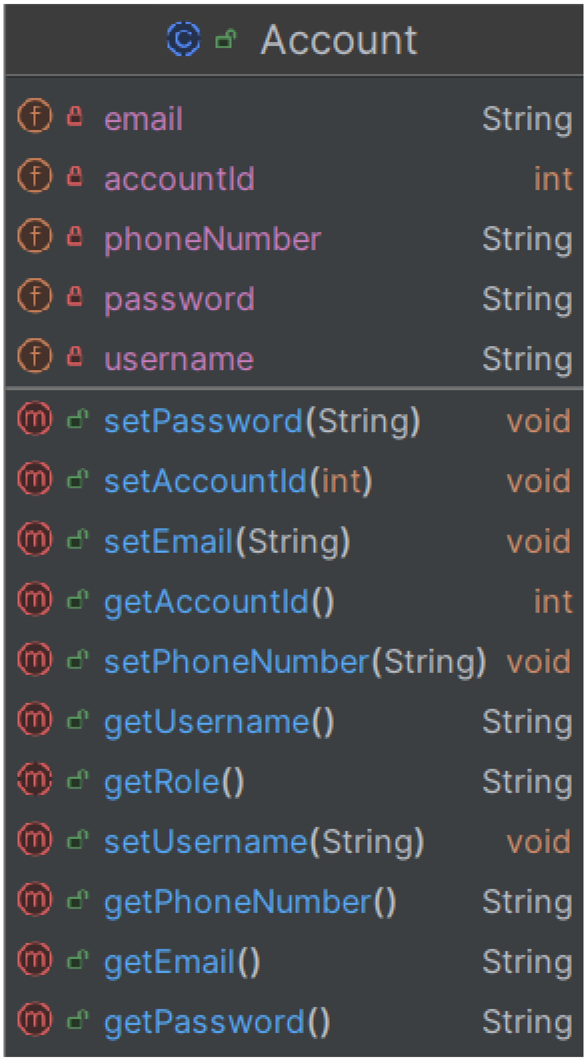
AdminPanel

handleRemoveSeat(JFrame)	void
handleAddSeat(JFrame)	void
addFlight()	void
handleUpdateSeatAvailability(JFrame)	void
deleteFlight()	void
display()	void



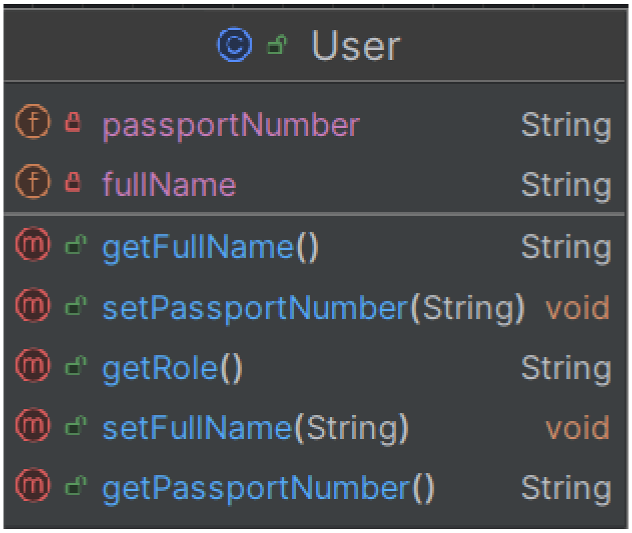
Passenger

id	int
name	String
passportNumber	String
toString()	String
getId()	int
getName()	String
getPassportNumber()	String



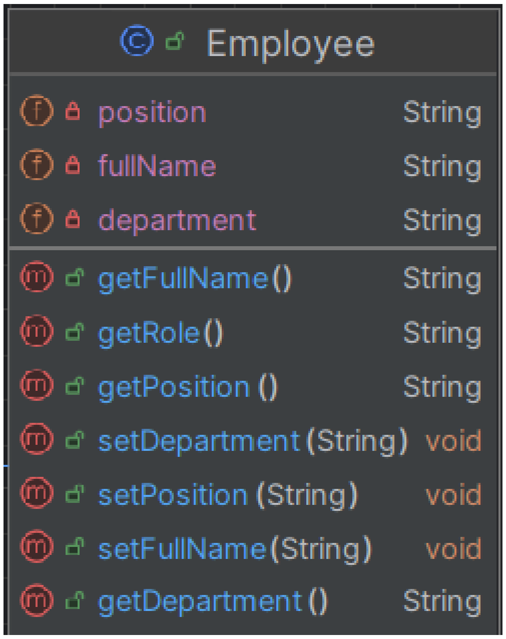
Account

email	String
accountId	int
phoneNumber	String
password	String
username	String
setPassword(String)	void
setAccountId(int)	void
setEmail(String)	void
getAccountId()	int
setPhoneNumber(String)	void
getUsername()	String
getRole()	String
setUsername(String)	void
getPhoneNumber()	String
getEmail()	String
getPassword()	String



User

passportNumber	String
fullName	String
getFullName()	String
setPassportNumber(String)	void
getRole()	String
setFullName(String)	void
getPassportNumber()	String



Employee

position	String
fullName	String
department	String
getFullName()	String
getRole()	String
getPosition ()	String
setDepartment (String)	void
setPosition (String)	void
setFullName(String)	void
getDepartment ()	String