Flight Sharing Platform

BEAUGIRAUD Paul
PORCEL Lauréna

Summary

- System definition..... p.3
- System features details..... p.4
- The Users..... p.6
- Business Objects..... p.7
- Technical proposition..... p.8
- Navigation diagram..... p.10
- Mock-ups..... p.11
- Backlog..... p.17

System definition

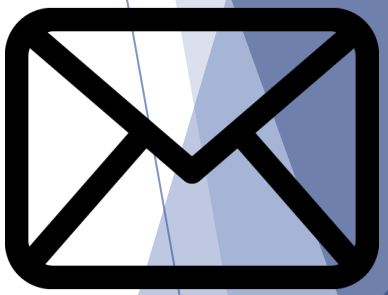
The system to developed is a website for a flight sharing platform in order to put a pilot in touch with passengers. The site will have an account system with 2 distinct roles: the pilot and the passenger. Without being registered, a passenger will be able to search for a flight with its characteristics among a list of flights added by the pilots. Finally, reservations will be accepted through a mail system and reminders will be send the day before flights.

System features details (1)

- ***Consult a flight:*** all the site users can search for a flight and see all flight information by selecting the departure aerodrome and the period sought.
- ***Register:*** as a pilot or passenger by completing the registration form.
- ***Edit profile: b***y adding other necessary information to the account or by modifying it.
- ***Planning a flight:*** The pilot can plan flights by manually adding the useful information (departure and destination aerodrome, flight duration, number of places available, price per passenger, flight sheet with trip details).
- ***Book a flight: i***dentified passengers can book a plane by selecting the desired number of seats.
- ***Flight viewing*** pilots and passengers can view their flights respectively plan or book.

System features details (2)

- ***Automatic mail sending:*** An email is automatically sent to the pilot following the reservation of a plane by a passenger. This email tells him the number of seats reserved for each reservation.
- ***Confirmation of flight:*** The pilot can confirm or not a booked flight.
- ***Reminder mail:*** A reminder of public and private information is sent the day before each flight to all participants.



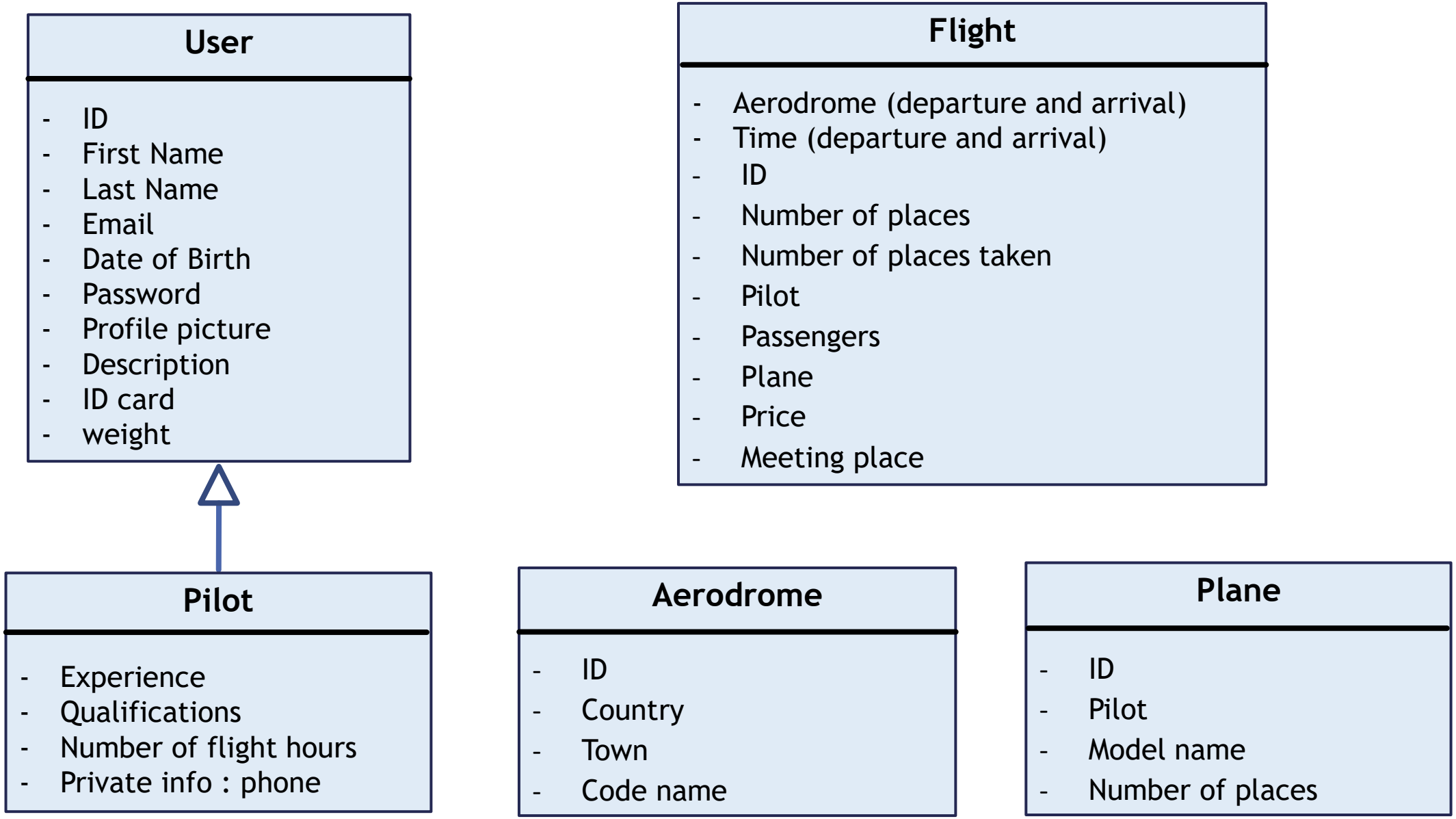
The Users

The pilot will be able to add **account information** concerning his personal life, the aircraft used (experience, qualifications, number of hours of flight, etc.). He can also add **public information**: flight planification and details (with the flight sheet), **private information** (meeting place, telephone), and modify them.
Finally, a pilot can **accept reservation request.**

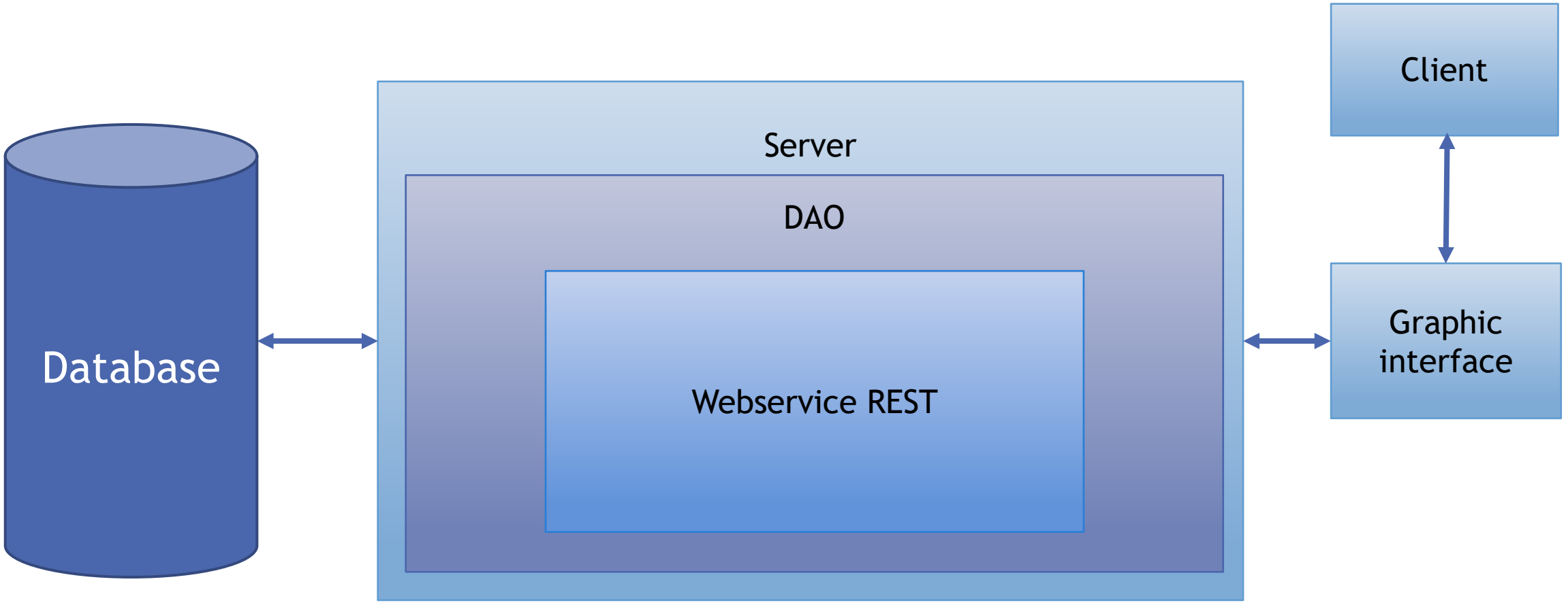
The non-registered can **search for a flight** and view the details of all the flights found. But pilot and passenger can also search and book a flight of other pilots. A registration form must be completed to complete and finish the reservation.

The passenger can **edit his profile** by adding new information (photo, description, weight, ID,…) and modify them. He can **book the chosen flight** and indicate the number of seats desired. Finally, he can **check the list** of his reserved flights.

Business objects



Technical proposition



DAO Interfaces

AerodromeDao

```
- List<Aerodrome>getAerodromes()
- Aerodrome getAerodrome(int aerodromeID);
- User postAerodrome();
- User putAerodrome();
- boolean deleteAerodrome();
- List<Flight> getFlightsDeparture();
- List<Flight> getFlightsArrival();
```

PilotDao

```
- List<Pilot> getPilots();
- Pilot getPilots(String pilotID);
- User postPilot();
- User putPilot();
- boolean deletePilot();
- List<Flight> getFlightsAdded(String
  pilotName);
- List<Plane> getPlanes(String pilotName);
```

FlightDao

```
- List<Flight>getFlights();
- Flight getFlight(int flightID);
- User postFlight();
- User putFlight();
- boolean deleteFlight();
- List<User> getPassengers();
```

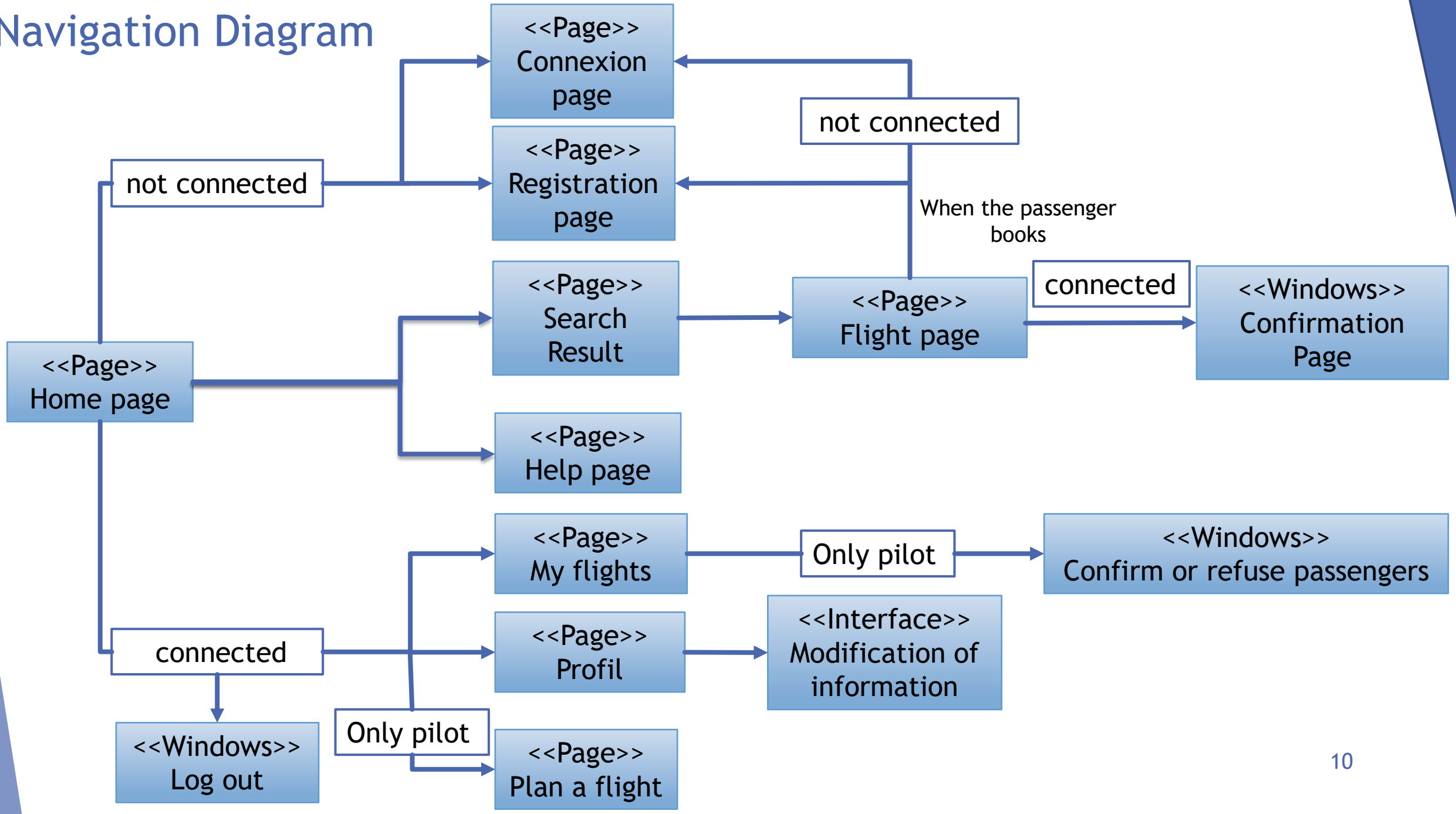
PlaneDao

```
- List<Plane> getPlanes()
- Plane getPlane(String planeID);
- User postPlane();
- User putPlane();
- boolean deletePlane();
```

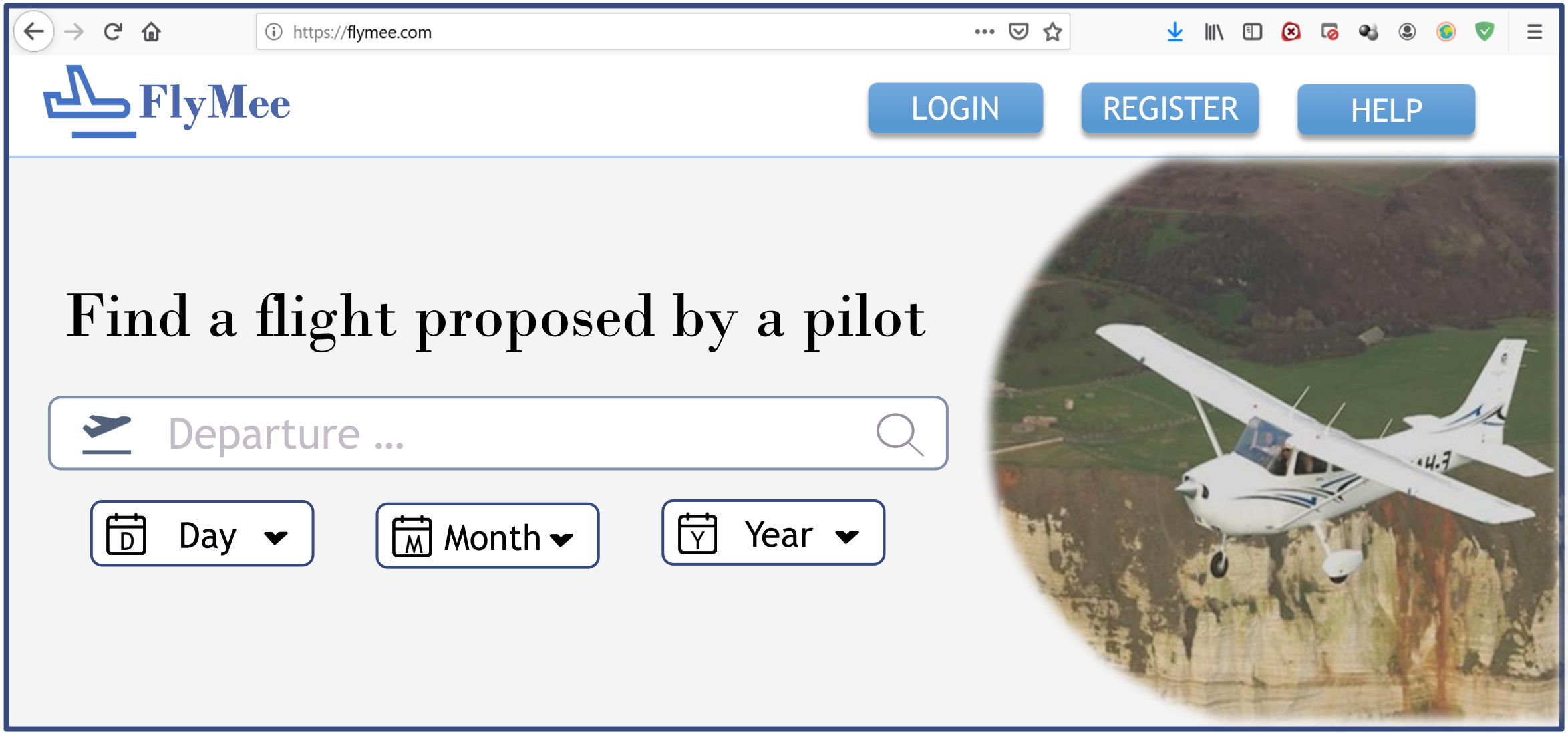
UserDao

```
- List<User> getUsers();
- User getUser(int userID);
- User postUser();
- User putUser();
- boolean deleteUser();
- boolean postFlight(String username);
```

Navigation Diagram



Mock-ups (1) – Homepage without register



Mock-ups (2) – Homepage connected

Only with a pilot account

https://flymee.com

FlyMee

ADD A FLIGHT

PROFILE

HELP

Edit Profile

My booked flights

Settings

Logout

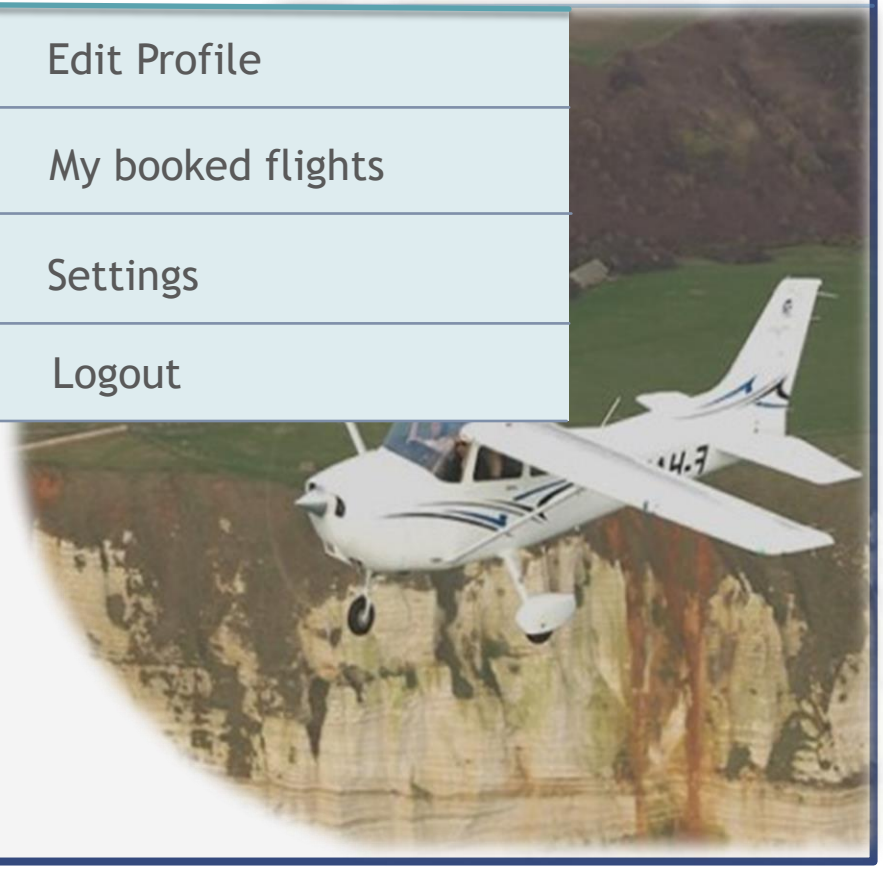
Find a flight proposed by a pilot

Departure ...

Day

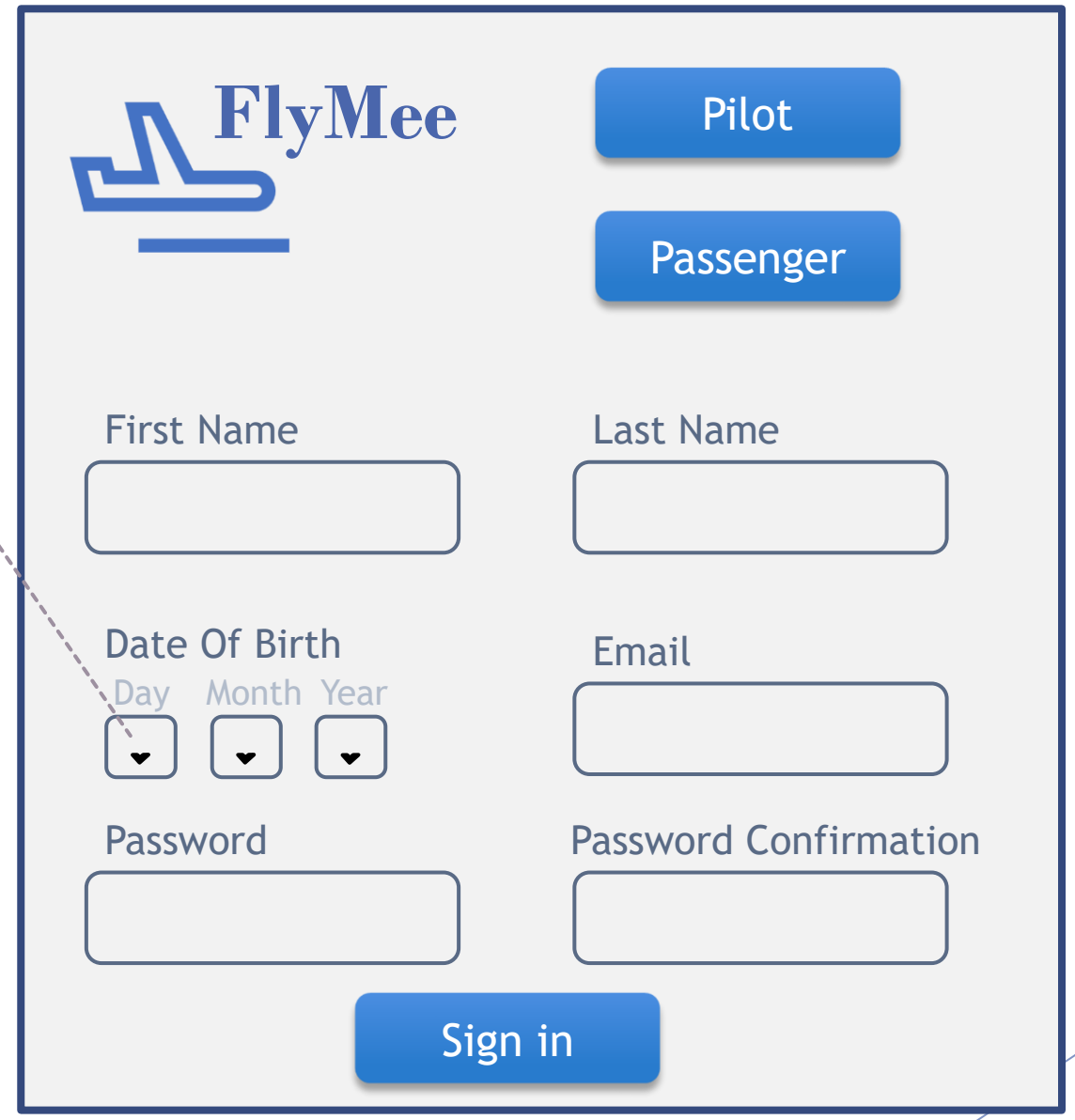
Month

Year



Mock-ups (3) – Registration form

We can type the day on the keyboard or scroll down the list of days



Mock-ups (4) – List of flights find

https://flymee.com

FlyMee

LOGIN REGISTER HELP

Cannes to Porquerolles

1h30 3

95$
Per passenger

View

Cannes to Menton

1h 4

87$
Per passenger

View

Mock-ups (5) – View flight

https://flymee.com

FlyMee

LOGIN REGISTER HELP

Cannes to Porquerolles

1h30 total flying time

3 seats

Your pilot: Mike

Description .....

My plane .....

Information .....

Route Details

1. .....
2. .....
3. .....
4. .....

Register to book this flight

95$

Per passenger

Backlog (few examples)

Example of Backlog		
Explanatory text	**Priority**	**Worload**
A Database with user accounts, flights, available aerodromes.	5	3
A pilot must be able to plan a flight	4	1
A visitor must be able to search for flights by entering the departure aerodrome and the date	4	2
A logged in user must be able to log out	3	5
All visitors must have access to a help page	2	4
A passenger must be able to cancel a flight	1	3

Find the complete table in the attached excel