

Fly2gether

Share your flight



Pritie Sougoumar
Hiba Souber



What is 'Fly2gether'?

The system we want to develop is a platform of flight sharing aiming to bring together non professional pilots as well as customers willing to go somewhere. The system will contain the flights available from a certain aerodrome, which should be specified by the user (unregistered) or the passenger in the search bar. In order to book a flight, the user must register and submit some vital informations, after that the request will be sent to the pilot, then he can accept or deny the request. The pilot must be registered as well, and must provide information on each flight. In case the request is accepted, a confirmation email will be sent to the passenger.

In addition to that, the system should display information on the pilots and the aircrafts involved, and should set a reminder to the passengers and the pilots participating in a flight a day before.



System features

- Display all available flights
- Display all the information concerning a selected flight
- Access the information regarding a pilot or an aircraft (experience, model...)
- Book a flight
- Send an email to the pilot with the number of booked seats
- Accept or deny a booking request
- Send a confirmation email to the passengers
- Set a reminder to the passengers/pilot the day before the flight

Who will use it?

Pilot

- planify flights
- receive an email for each booking with the number of seats booked
- choose the passengers
- view there public informations

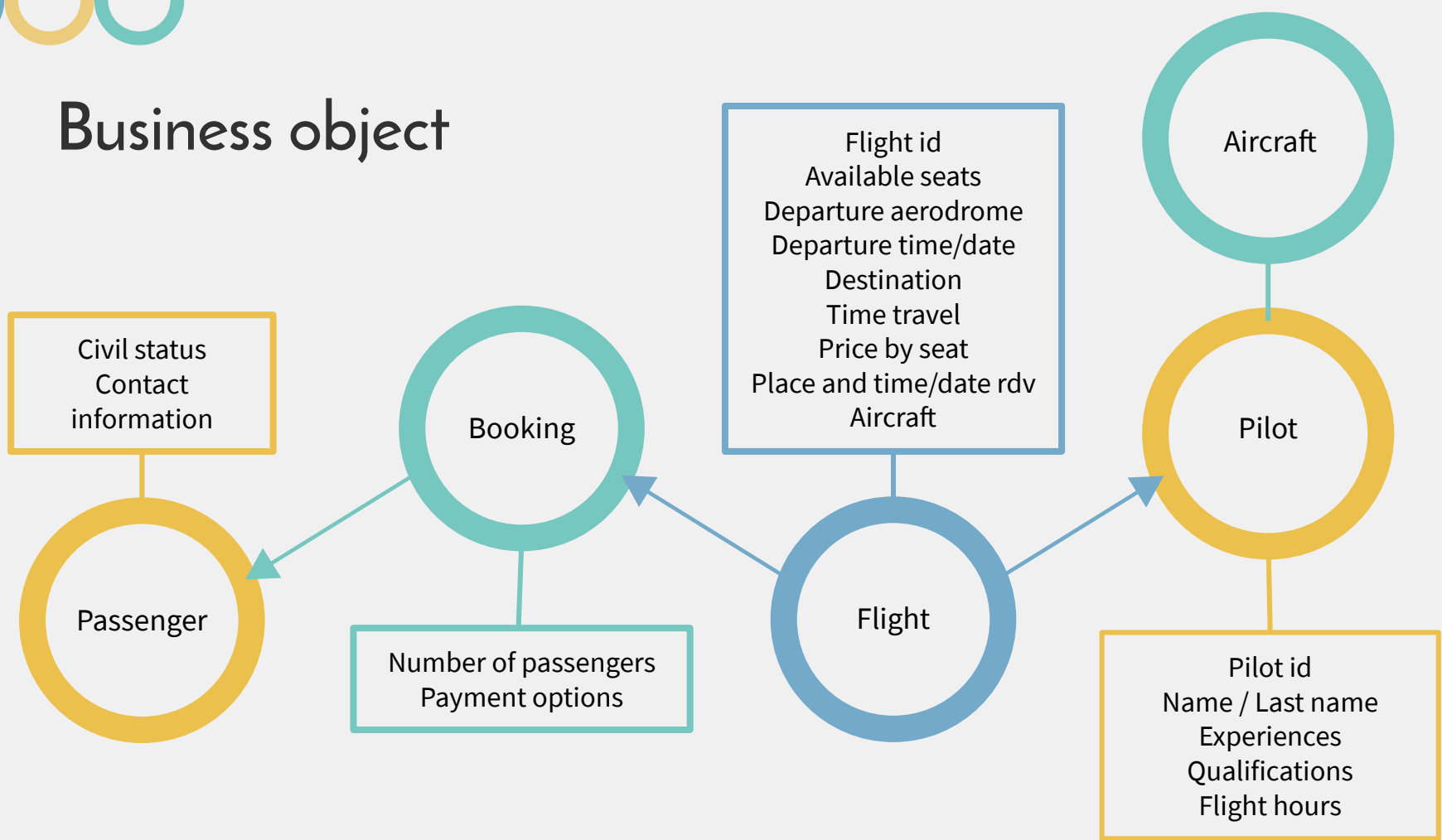
Passengers (registered)

- book a flight for several persons
- receive a confirmation email
- view the pilote and aircraft profile
- view available fights
- register

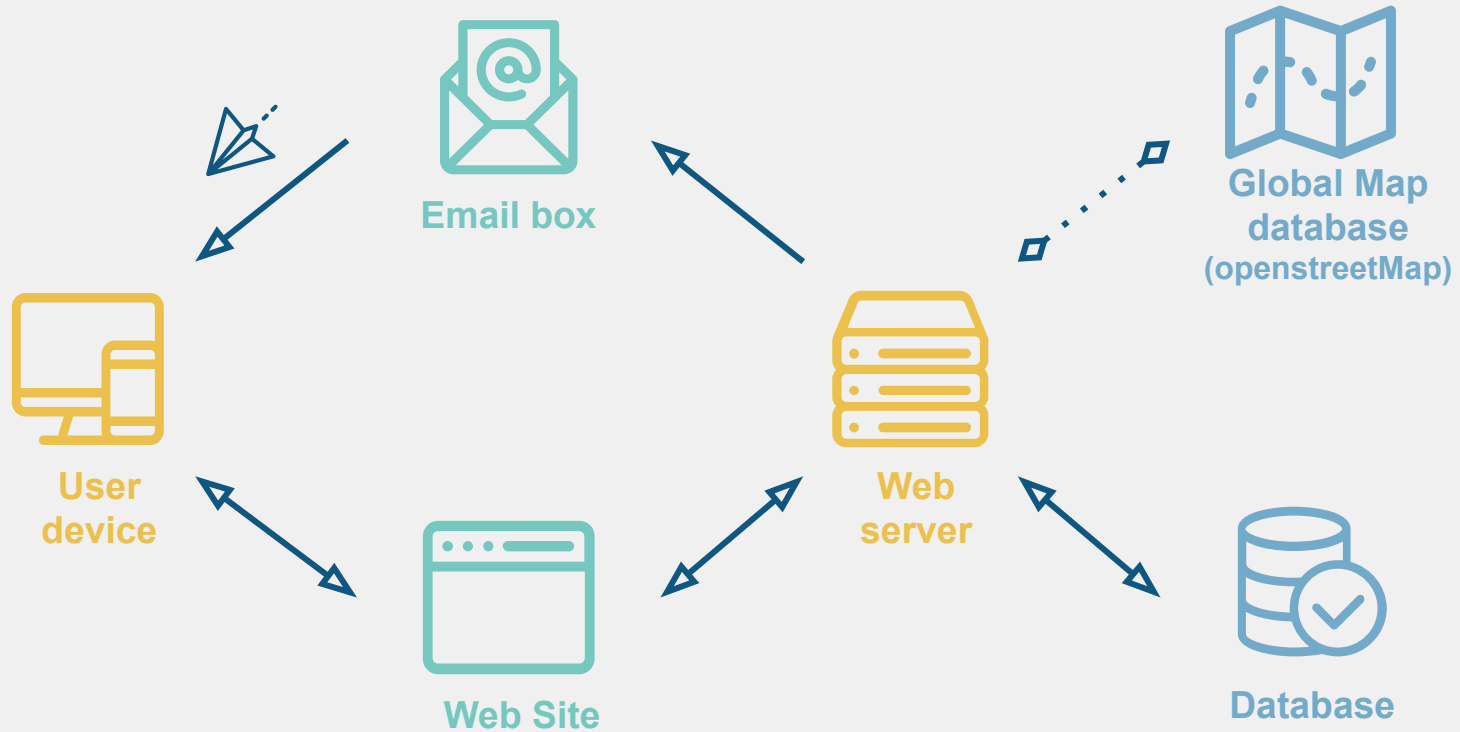
Visitors (unregistered)

The day before the flight they all receive a reminder

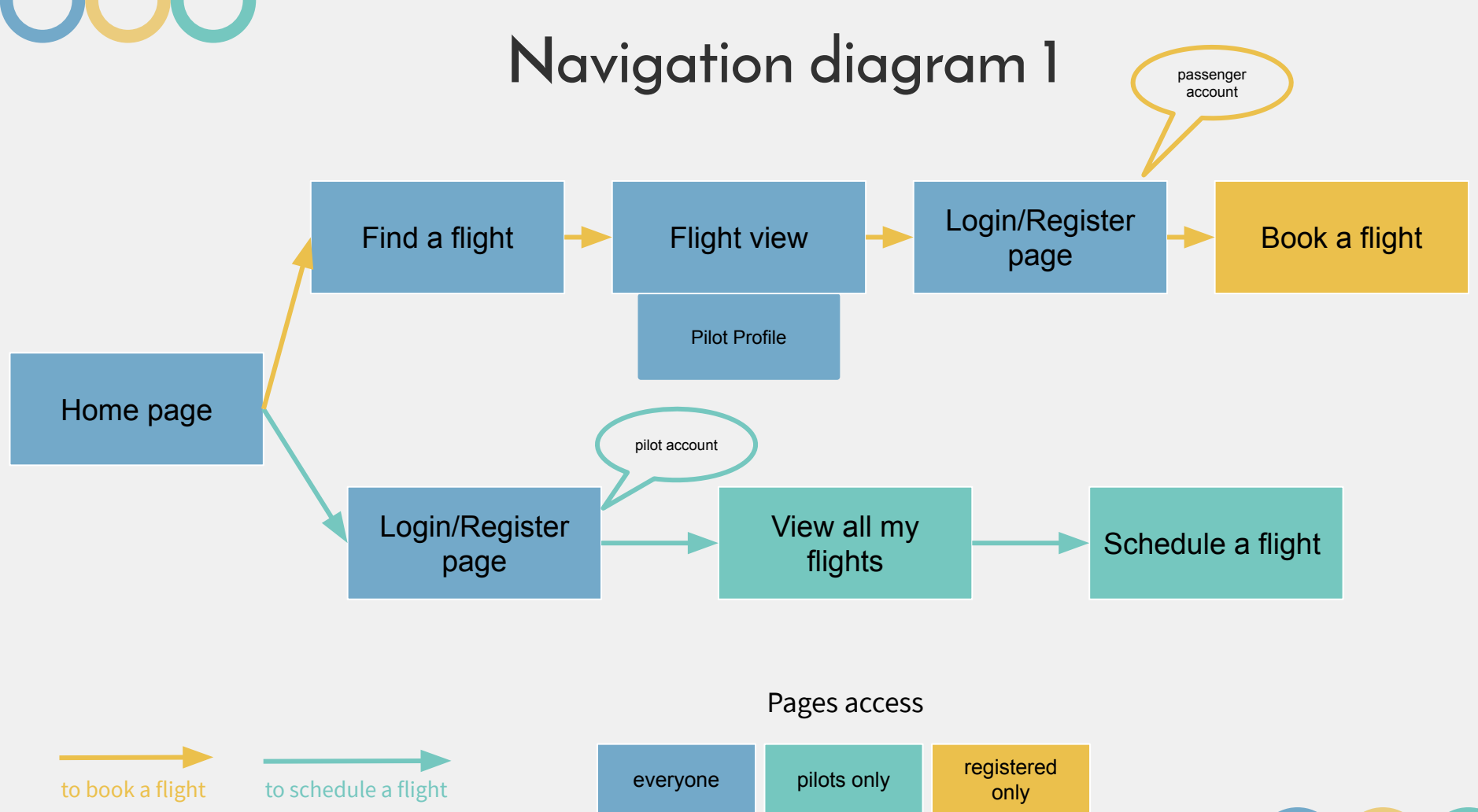
Business object



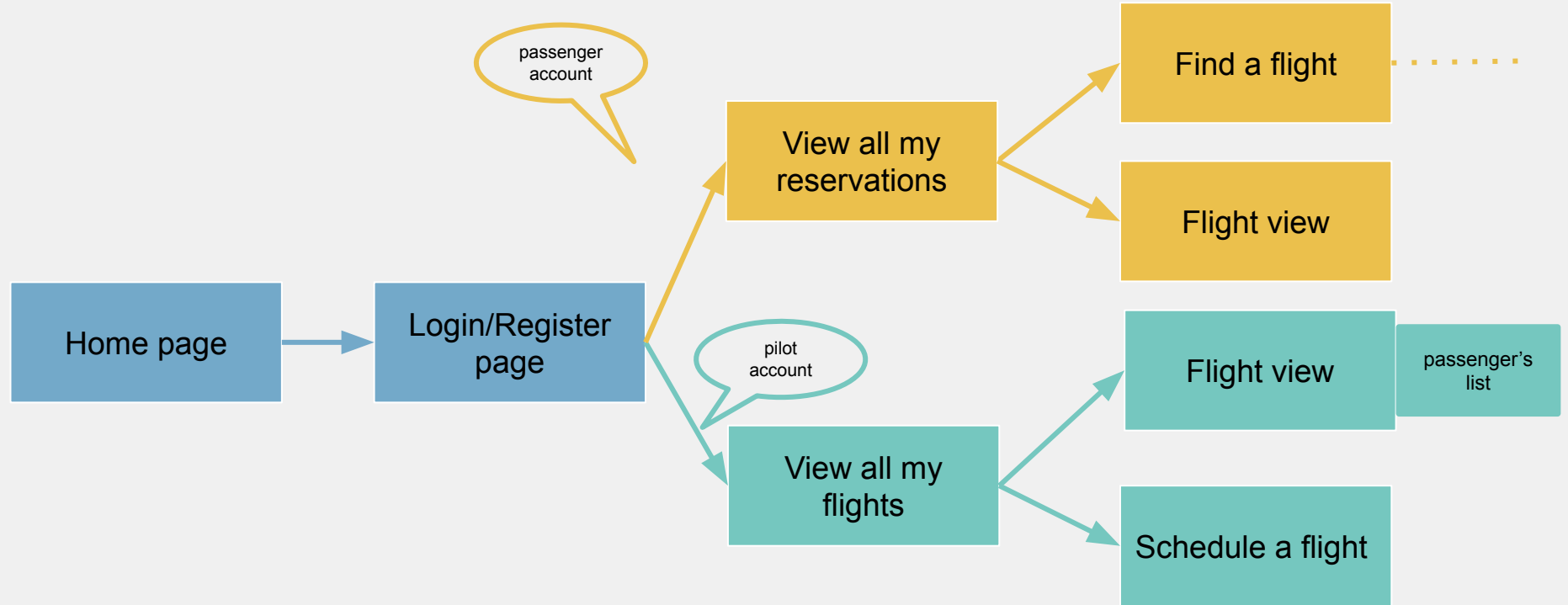
The System's structure



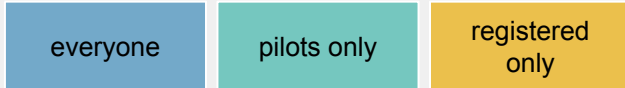
Navigation diagram 1



Navigation diagram 2



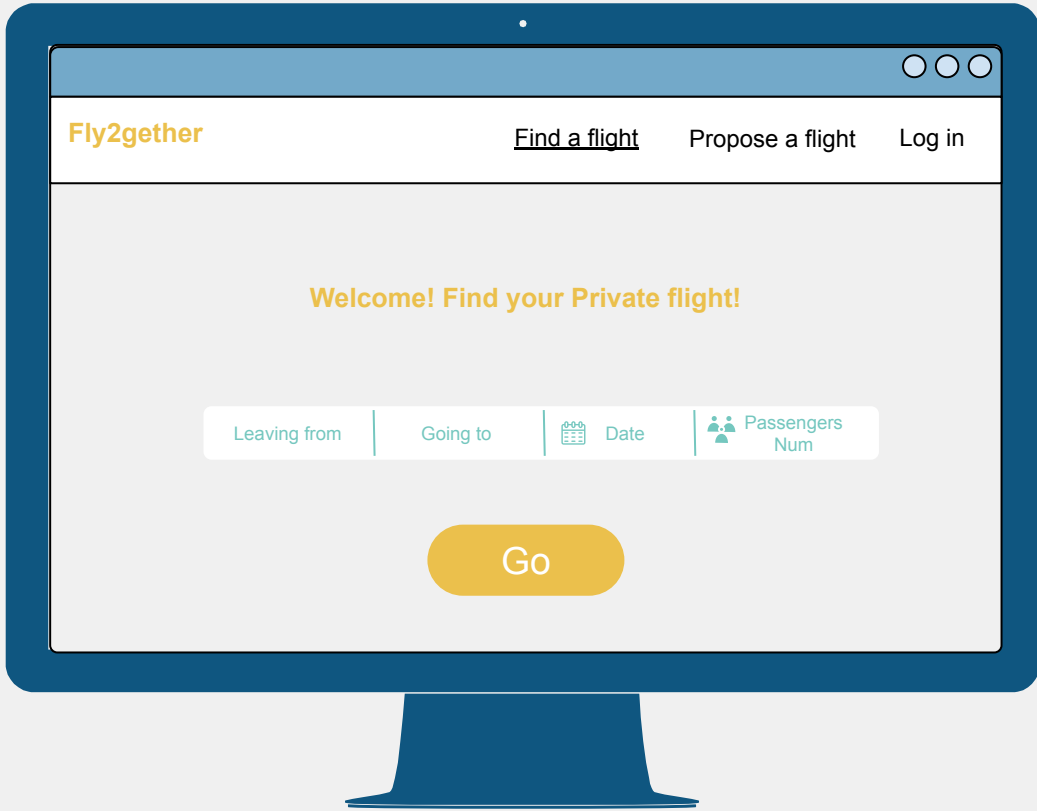
Pages access





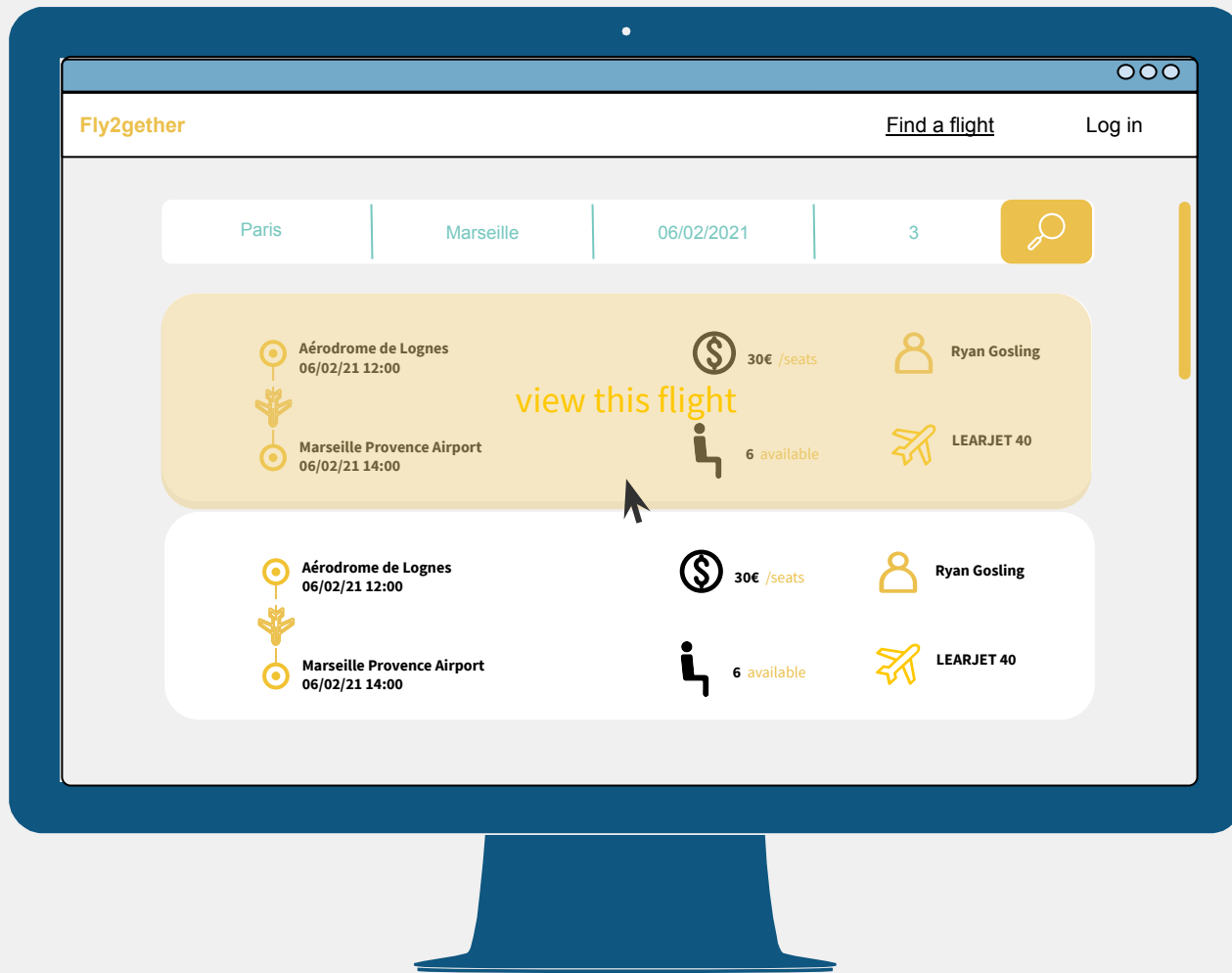
WEBSITE MOCKUP





Unlogged






Unlogged

Fly2gether


Find a flight

Log in


Flight view




Appointment place
Date
Time




Departure
Aérodrome
Date
Time




Arrival
Aérodrome
Date
Time




30€
/seats



6
available




LEARJET 40



Ryan Gosling

Choose this flight



Ryan Gosling

Hours of flying

800 hours

Qualifications

degree 1

YYYY

degree 2

YYYY

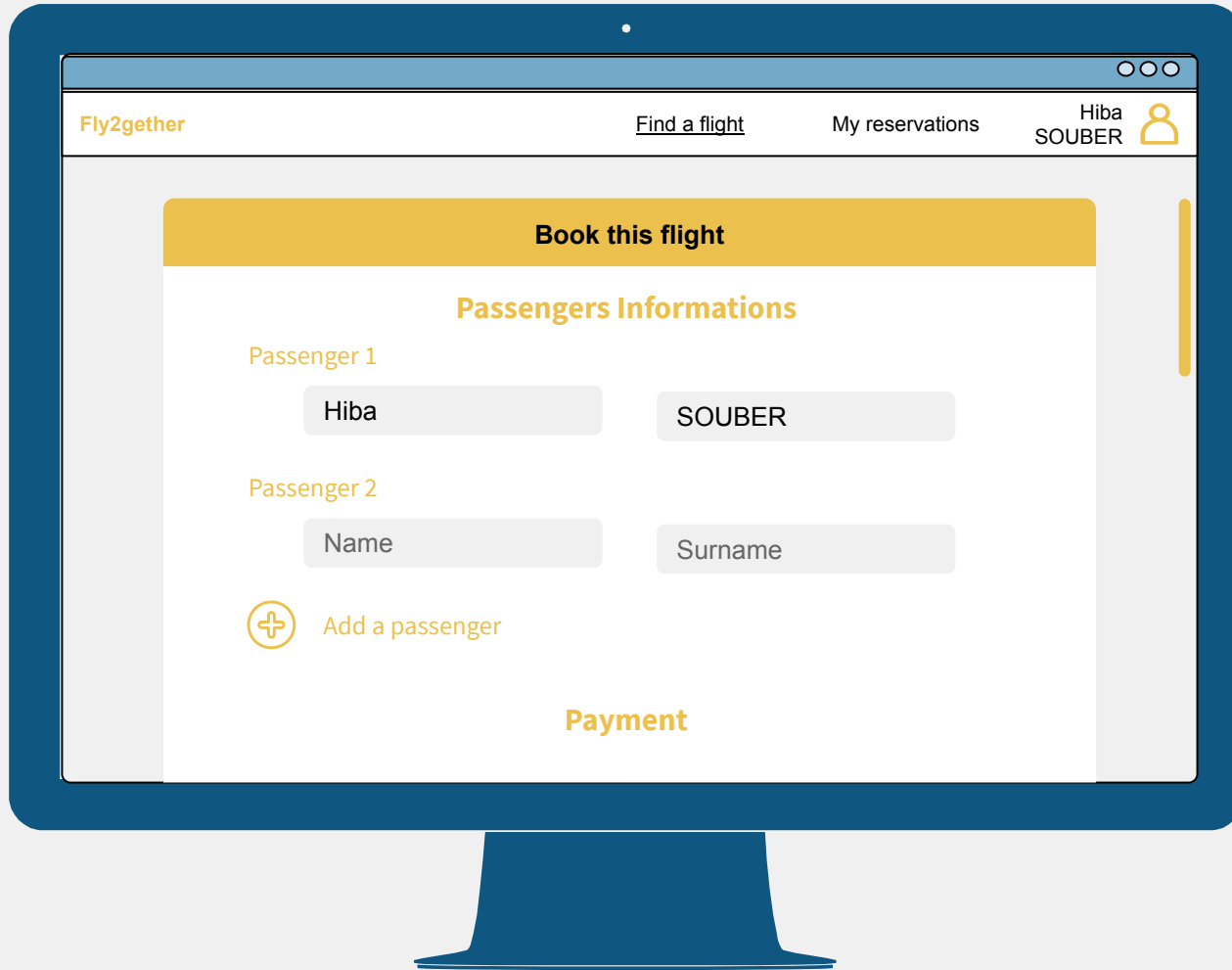
Professional experiences

experience 2

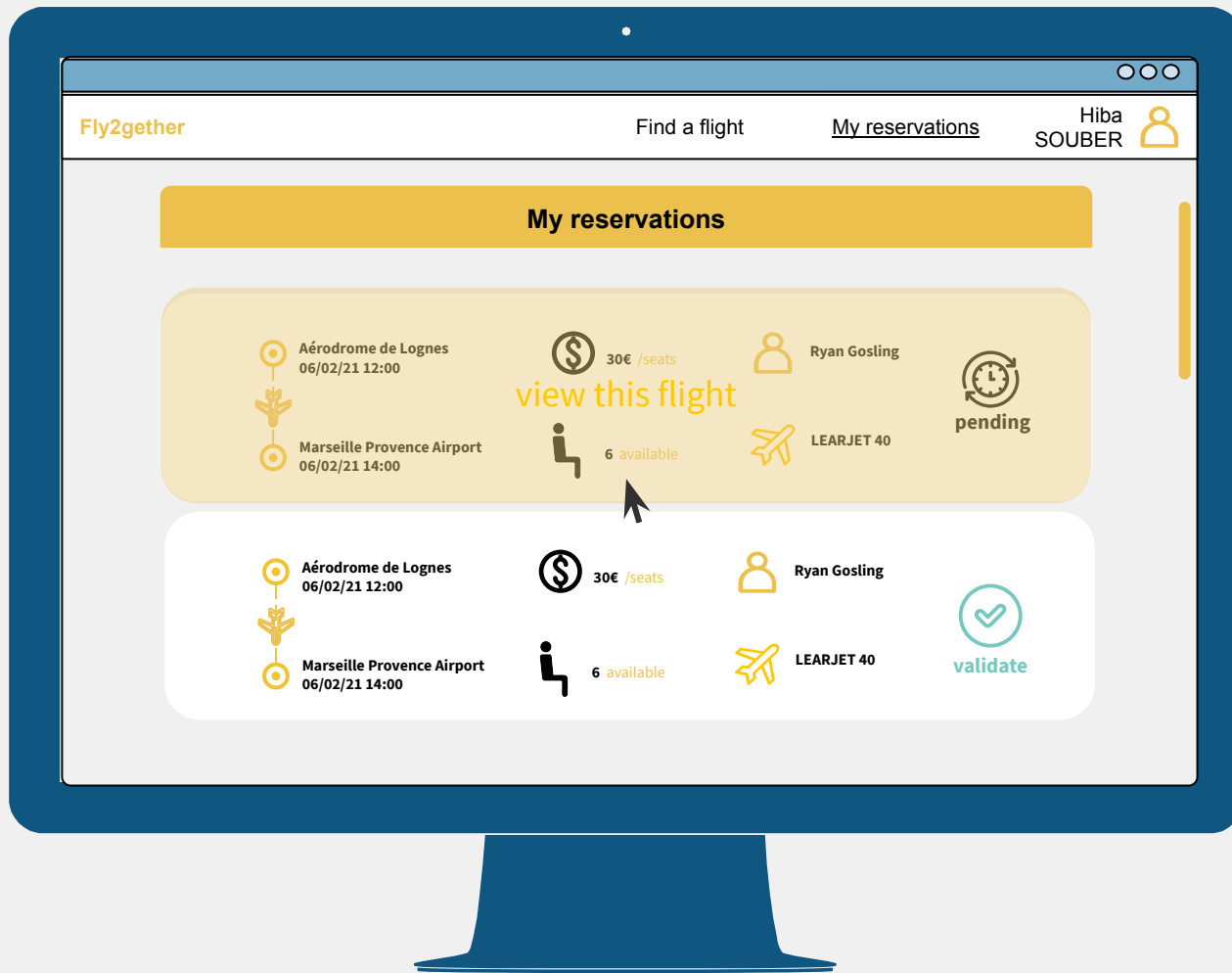
YYYY

Unlogged

12



Passenger
account



Passenger
account

Other Inputs:

- Available seats
- Price by seat
- Appointment place and time/date
- Aircraft

Fly2gether Find a flight Scheduled flights Plan a flight Hiba SOUBER

Plan a flight

Departure

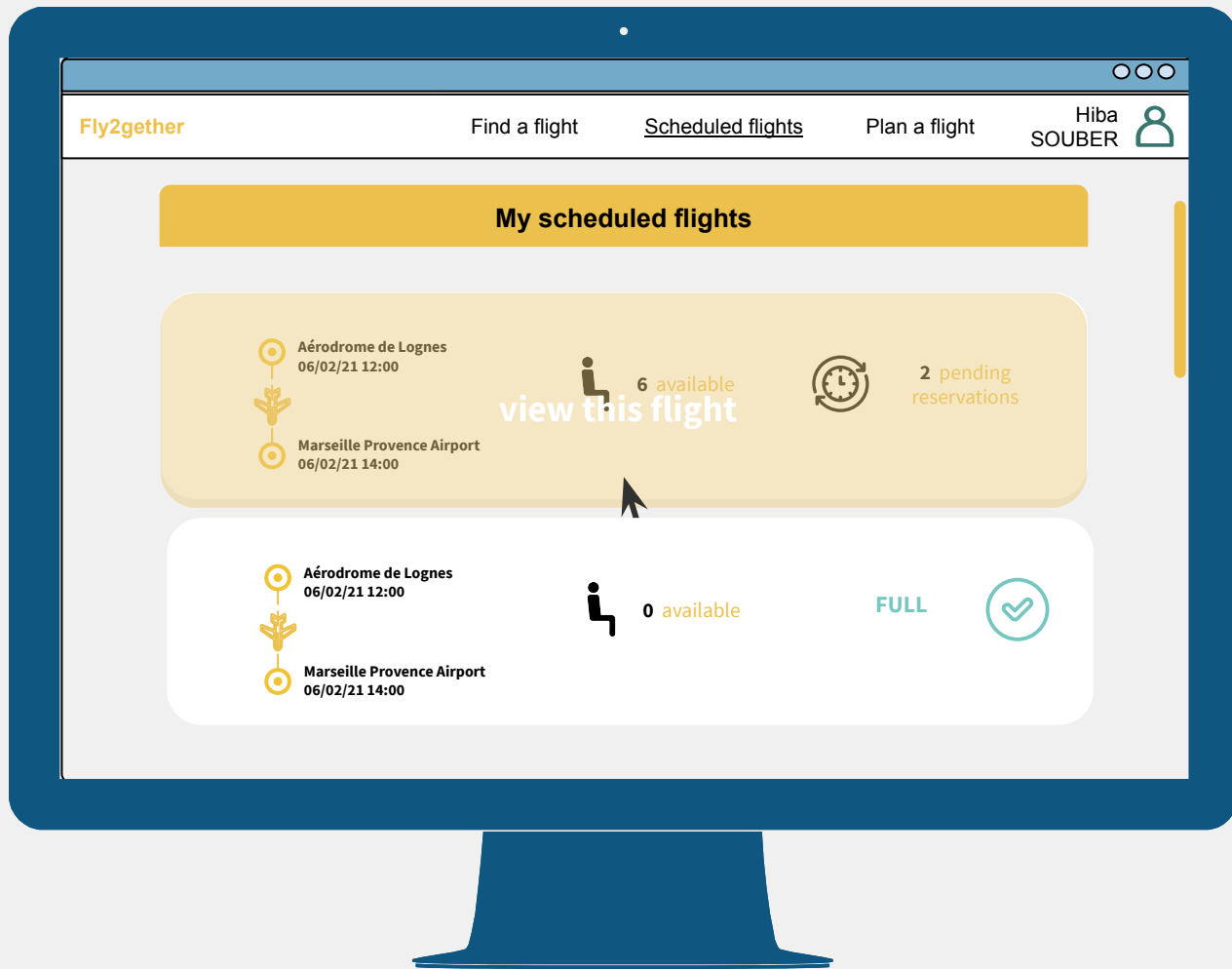
<input type="text" value="Aerodrome Name"/>	<input type="text" value="Address"/>
<input type="text" value="City"/>	<input type="text" value="Country"/>
<input type="text" value="dd/mm/yyyy"/>	<input type="text" value="hh:mm"/>

Arrival

same as departure ☐

<input type="text" value="Aerodrome Name"/>	<input type="text" value="Address"/>
<input type="text" value="City"/>	<input type="text" value="Country"/>

Pilot Account



Pilot Account

Fly2gether

Find a flightScheduled flightsPlan a flight

HibaSOUBER

Flight view

Appointment place
Date
Time

30€
/seats

Departure
Aérodrome
Date
Time

6
available

Arrival
Aérodrome
Date
Time

LEARJET 40

Passenger list

2 pending1 validate

Passenger list

2 pending1 validate

Pending

SOUGOUMAR

2 seats


réserveation name

x seats


Validate

Jacky Jack

Pilot Account



Function	Explanation	Priority	Workload
Search bar	Search for a flight	5	1
Passenger registration	Save information on the passenger (civil status, contact information)	4	3
Pilot registration	Save information on the pilot (qualifications)	4	3
Login	Save email and password of both the passengers and the pilot	5	2
Flight consultation	Show information on the flight : departure and arrival aerodrome, date and time	3	5
Pilot/Aircraft consultation	Show information on the pilot such as experience, qualifications, as well as the aircraft : model, hours of flight	3	5
Remove	Remove flights that are fully booked and that are unavailable	4	2



Planification	The pilot submits all the information regarding a flight (departure/time date and time, aerodrome)	5	2
Book	Book an available flight, with the number of desired seats	5	3
Accept	The pilot accepts or denies a booking request	5	4
Confirm	Sends a confirmation email to the passenger in case his request is accepted	3	5



Webservices

Type	URL	Behavior	Accessibility
GET	/find-flight?option1...	Returns the list of available flights that correspond to the options(date/time, departure, destination, remaining seats)	Public
GET	/flight-{id}	Return details of flight for the corresponding “id” (date/time, departure, destination, remaining seats, aircraft..)	Public
POST	/flight-{id}/book	Modify the flight booking list	Passenger only
PUT	/flight/schedule	Add a new flight	Pilot only
PUT	/register-{type}	Add a new user for the corresponding ‘type’ (pilot or passenger)	Public



POST	/user-{id}	Modify details of the user for the corresponding “id”	Pilot/ Registered passenger
DELETE	/user-{id}	Delete the user for the corresponding “id”	Pilot/ Registered passenger
DELETE	/flight-{id}	Delete a flight using its “id”	Pilot only
GET	/ user-{id}/flights	Return the list of the pilot’s scheduled flights	Pilot only
GET	/user-{id}/reservations	Return the list of the passenger’s booked flights	Passenger only
POST	/flight-{id}	Modify a flight by adding/changing details about it	Pilot only
GET	/user-{id}	Return the user profile for the corresponding “id”	Pilot/ Registered passenger



Persistence layers

In this project, we chose to create, in addition to a class, an interface for each object, we have :

flightDao : contains the definitions of methods that get infos on the flights, the lists of flights, filtered by min and max price, airport and time of departure or arrival, add or delete a flight

piloteDao : contains the definitions of methods that get infos on the pilots, the lists of pilots filtered by their experience, add or delete a pilot

passengerDao : contains the definitions of methods that get infos on the passengers, the lists of passengers of a certain flight, or passengers in a certain airport at a certain date, add or delete a passenger on a flight

reservationDao : contains the definitions of methods that get the lists of a passenger's reservations, or the reservations made with a certain payment method

aircraftDao : contains the definitions of methods that get infos on the aircrafts, the lists of aircrafts, add or remove an aircraft

as shown in our git : <https://github.com/pritie12/Fly2gether/tree/TP5>