FlightScraper

G8T2
Choy Jing Wei
Neo Jia Ying
Nor Aisyah
Tng Zi Xiang
Wong Wei Ling
Wong Xin Wei

TABLE OF CONTENTS

01
Solution Scoping
Brief description

02
Scenario Diagrams
Scenario Overview
User Scenario

03
Technical Overview Diagram

04
SOA

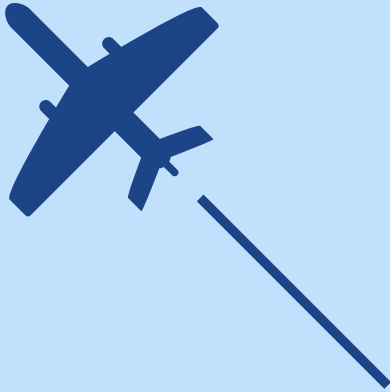
05
Beyond the labs

01

Solution Scoping

Brief Description

FlightScraper:

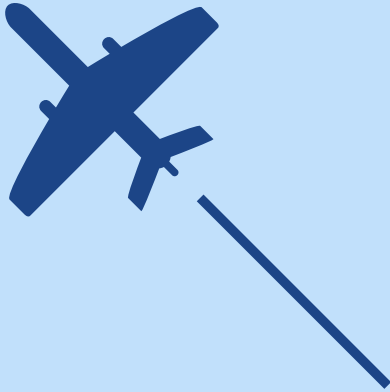


- Web-based solution
- Search for desired flights
- Place bookings for flights
- Make payment via PayPal
- Edit booking
- Check-in online
- Cancel booking
- Viewing of admin dashboard and admin configuration

02

Scenario Diagrams

Scenario 1



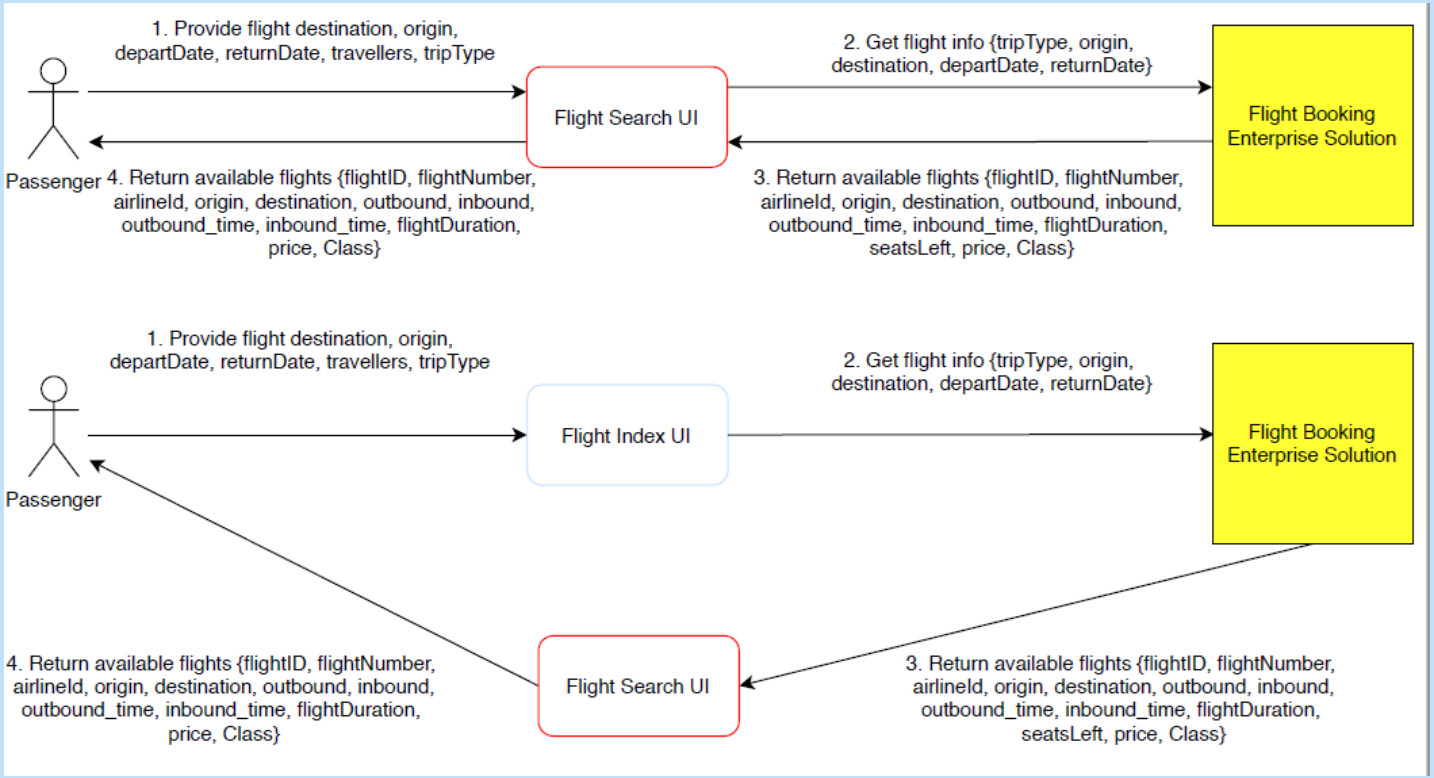
Passenger:

1) Passenger searches and selects a flight

- User enters the homepage of the website which comprises of Flight Search UI
- User enters his/her flight criteria such as origin, destination, flight dates and number of co-passengers
- A list of flights matching the criteria given will be displayed.

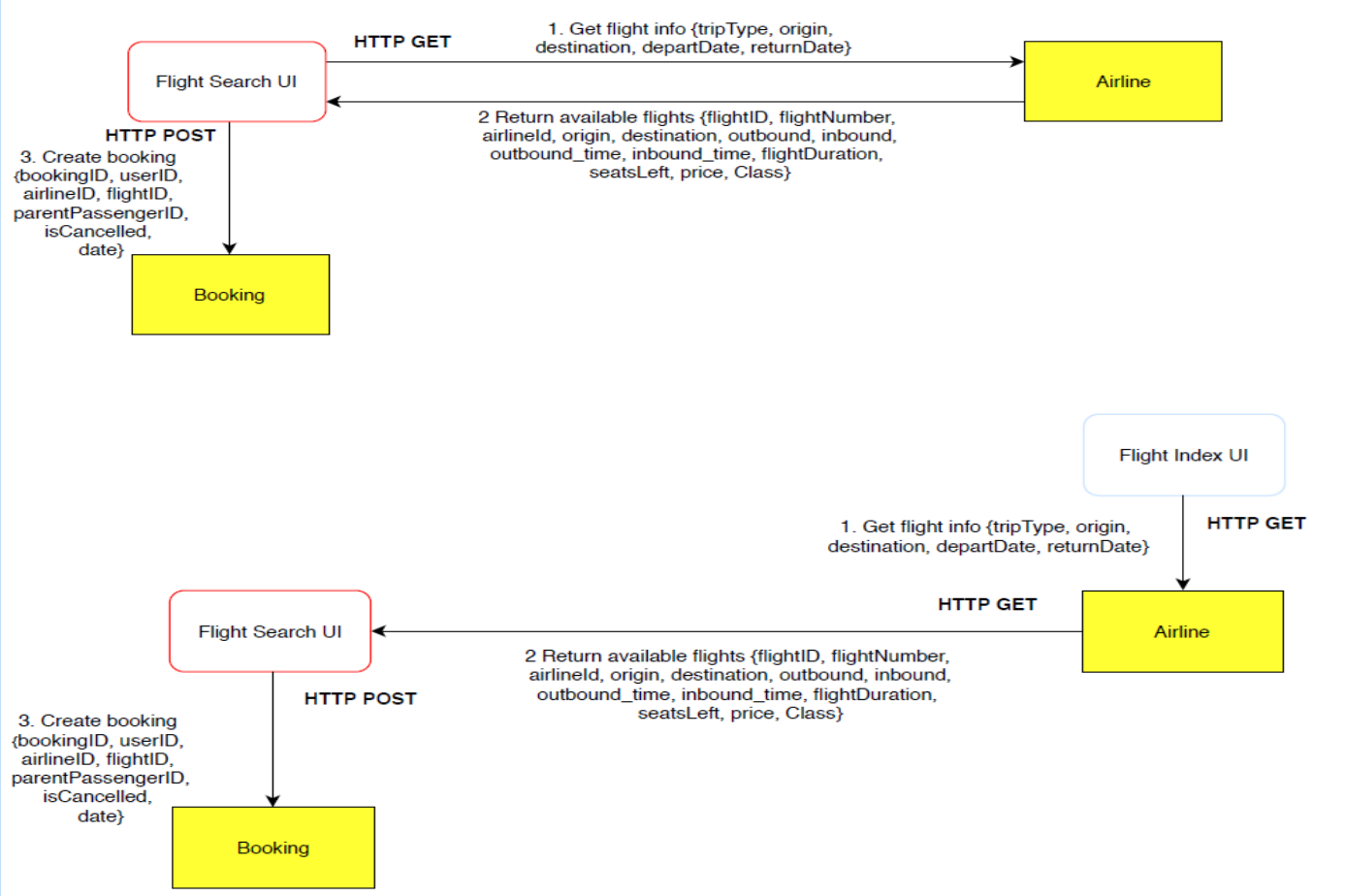
Scenario Overview Diagram 1

- Passenger searches for flight



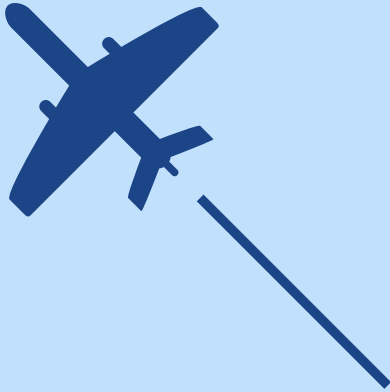
User Scenario Diagram 1

- Passenger searches for flight



Scenario 2

Passenger:

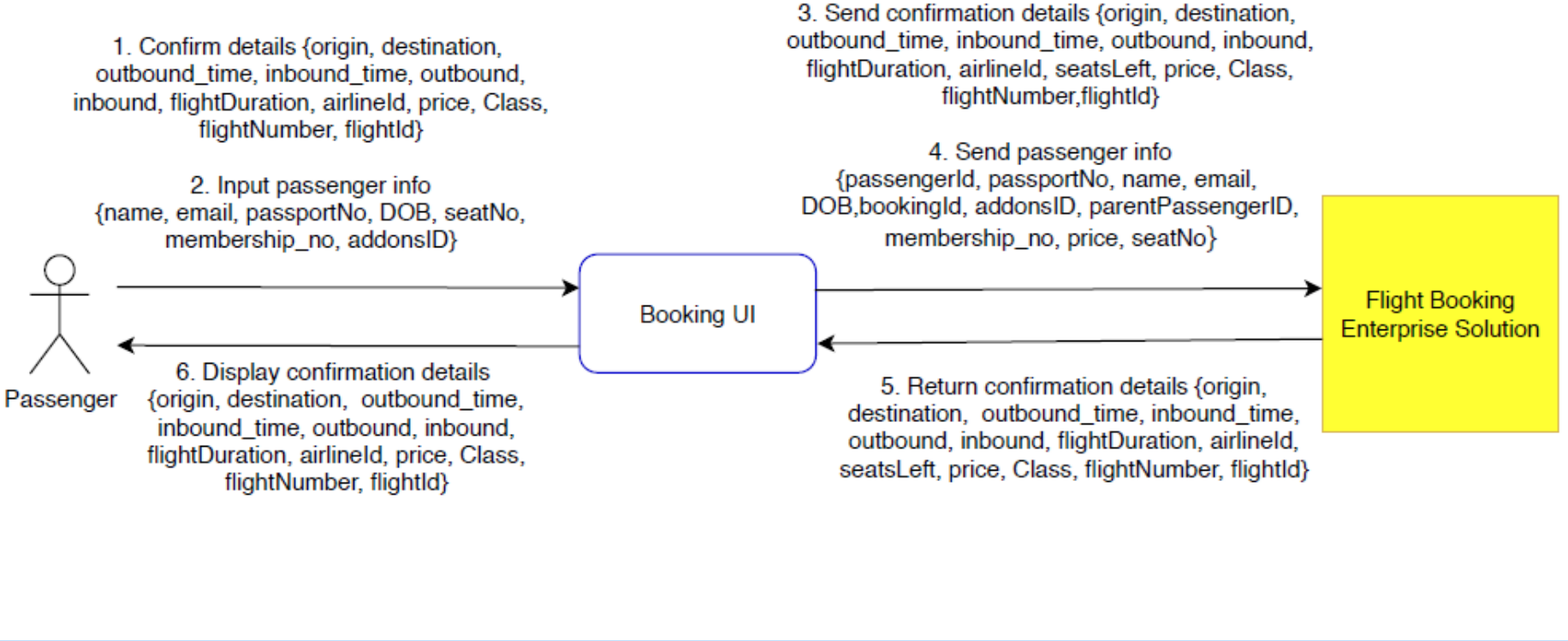


2) Passenger books for selected flight

- After user selects the desired flight, user will be redirected to the Booking UI.
- User is then required to fill in the details of his/her information as well as other co-passengers involved in the flight.

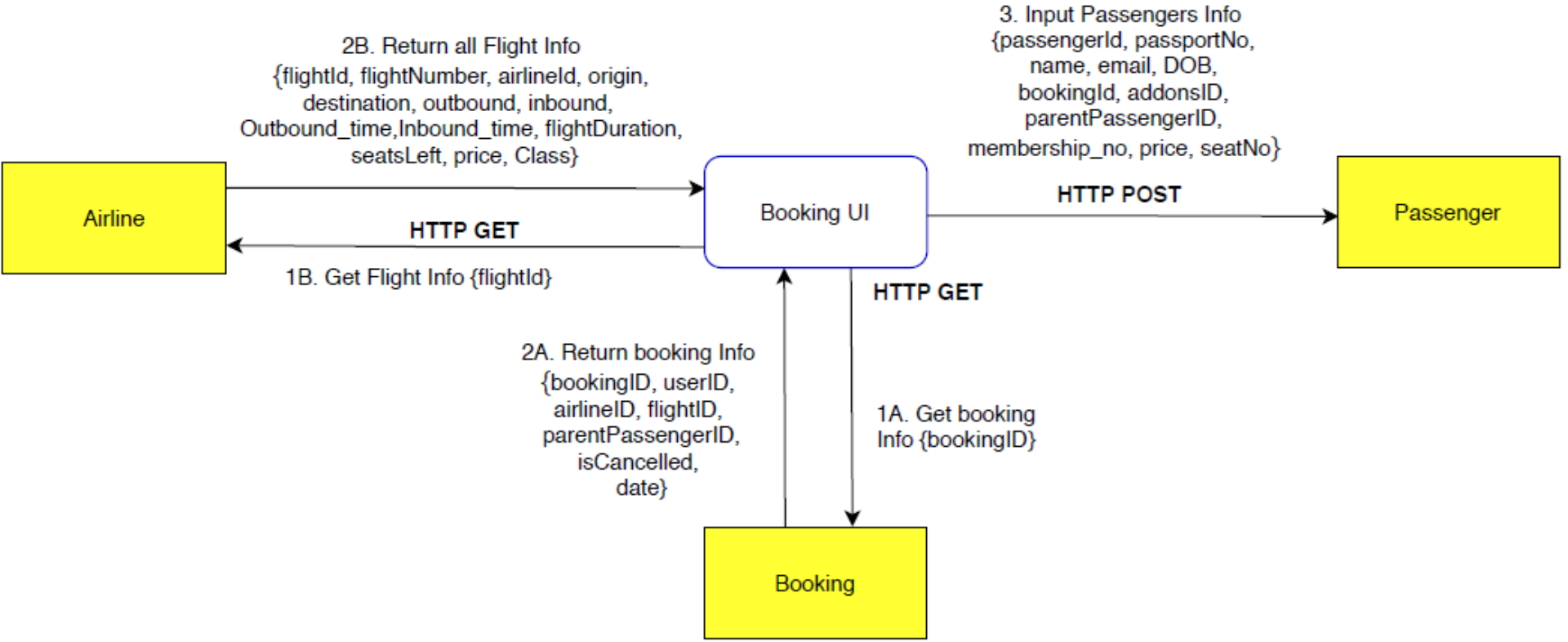
Scenario Overview Diagram 2

- Passenger books for selected flight



User Scenario Diagram 2

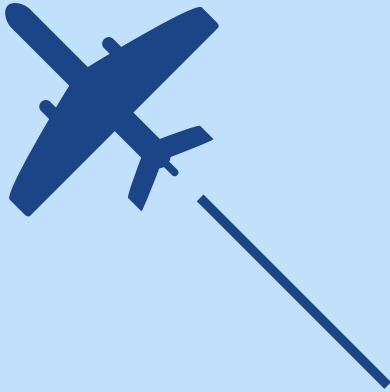
- Passenger books for selected flight



Scenario 3

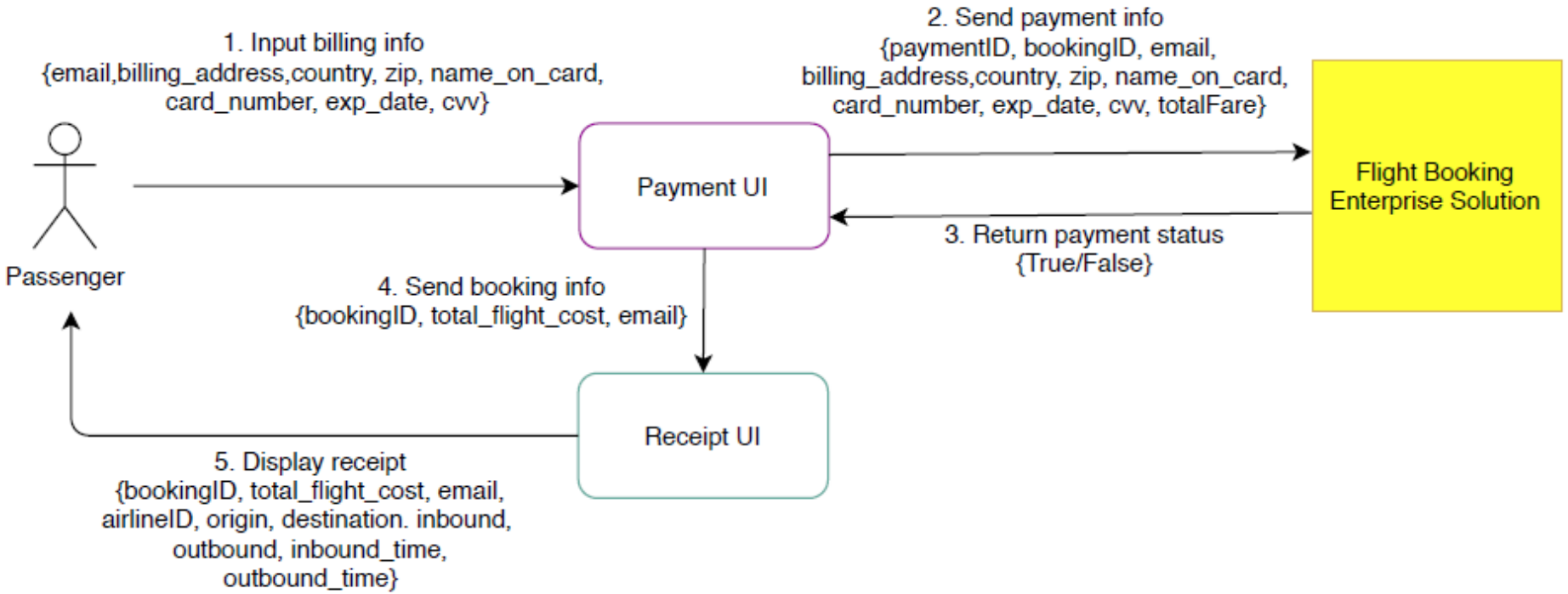
Passenger:

3) Passenger pays for booking



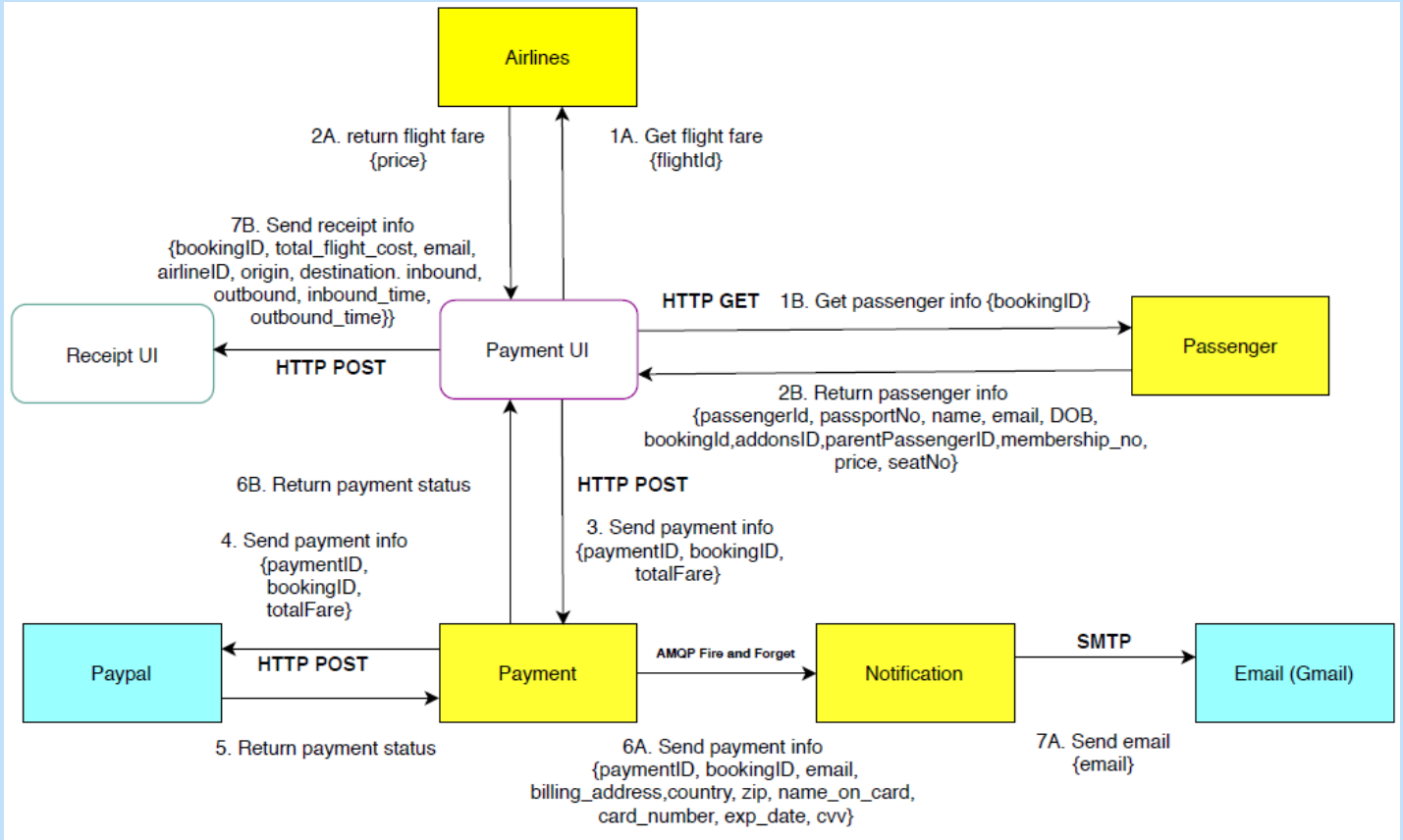
- After confirming the booking details user will proceed to make payment at payment UI, using **PayPal API.**
- Upon successful payment, user will be presented with an **online receipt** and **an email will be sent to the user**

Scenario Overview Diagram 3
- Passenger pays for booking



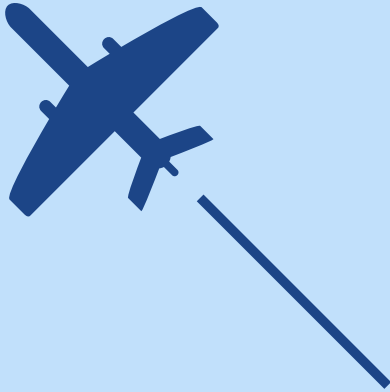
User Scenario Diagram 3

- Passenger pays for booking



Scenario 4

Passenger:

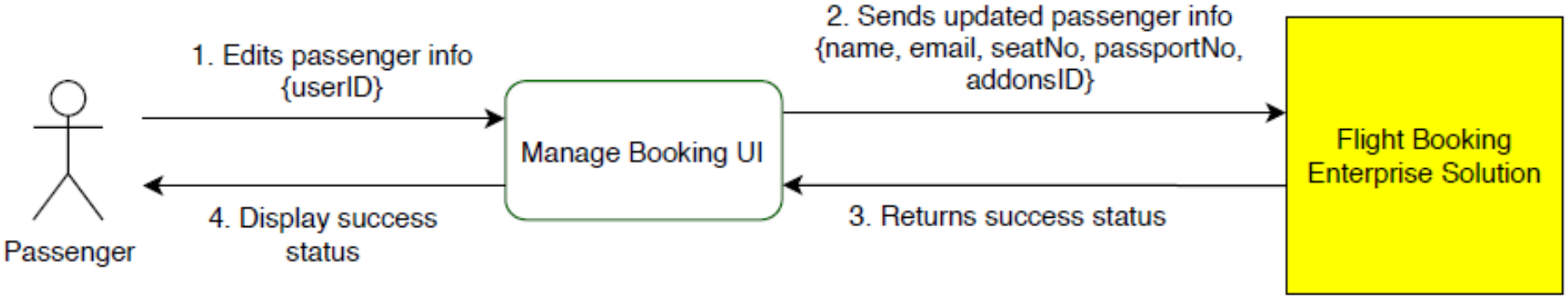


4) Passenger edits passenger info

- Passenger edits their info at the Manage Booking UI

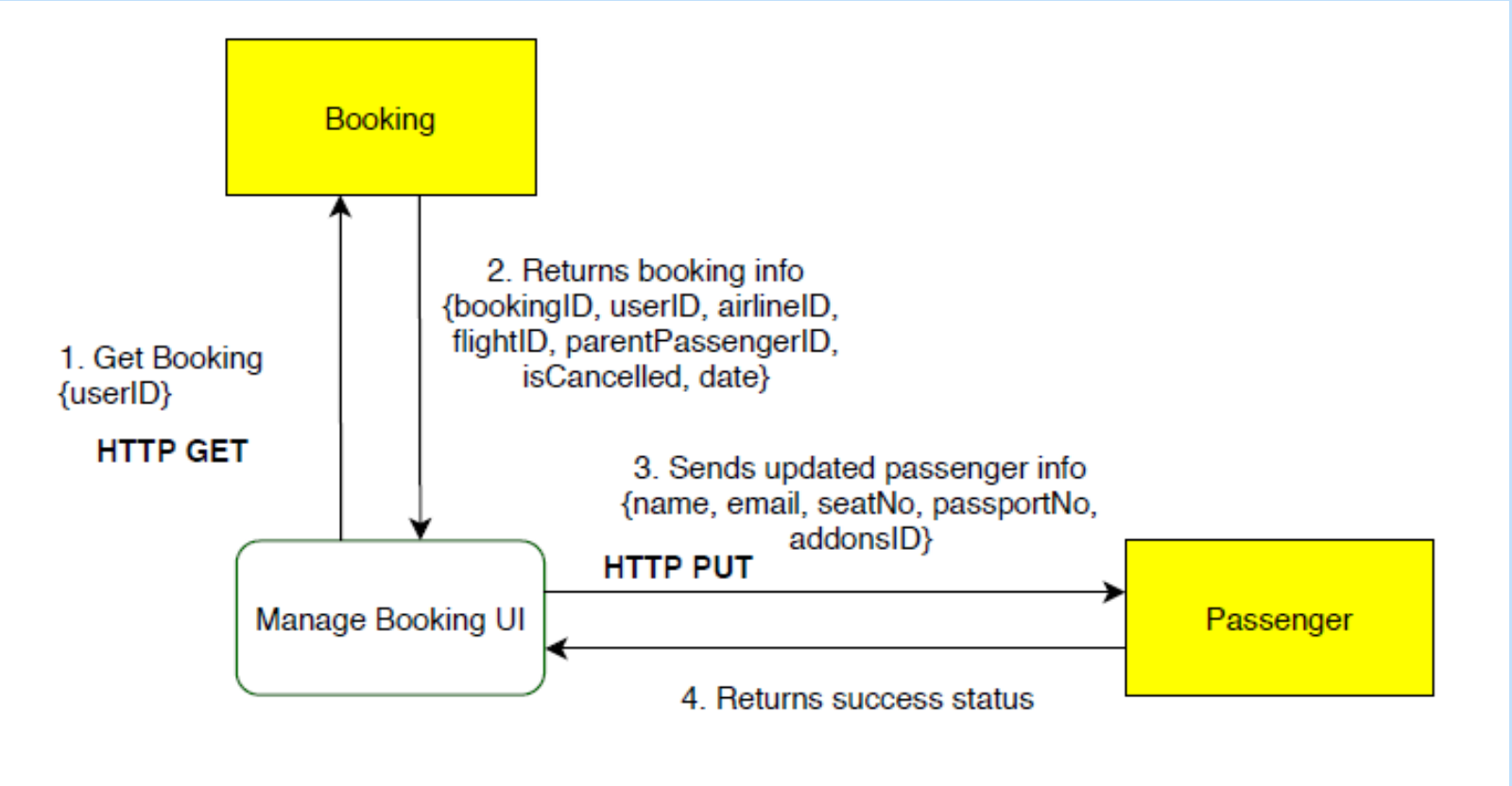
Scenario Overview Diagram 4

- Passenger edits passenger info



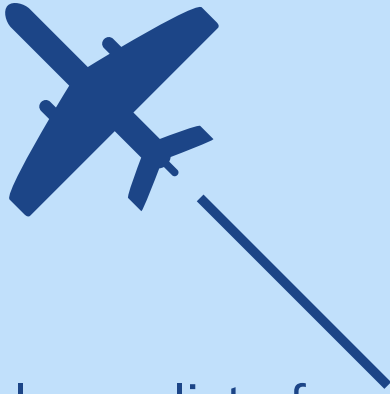
User Scenario Diagram 4

- Passenger edits passenger info



Scenario 5

Passenger:

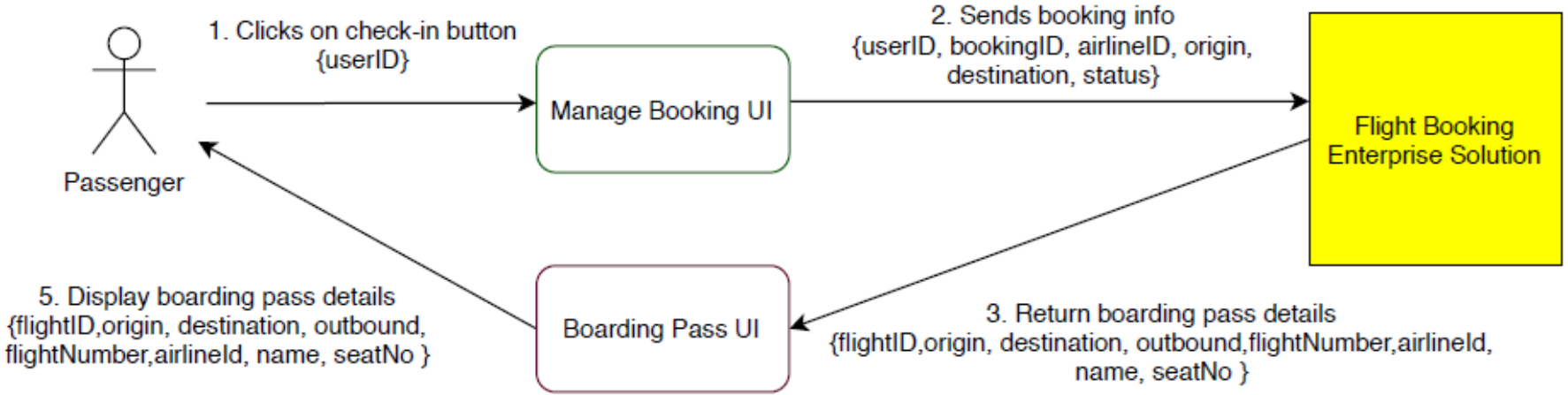


5) Passenger checks-in

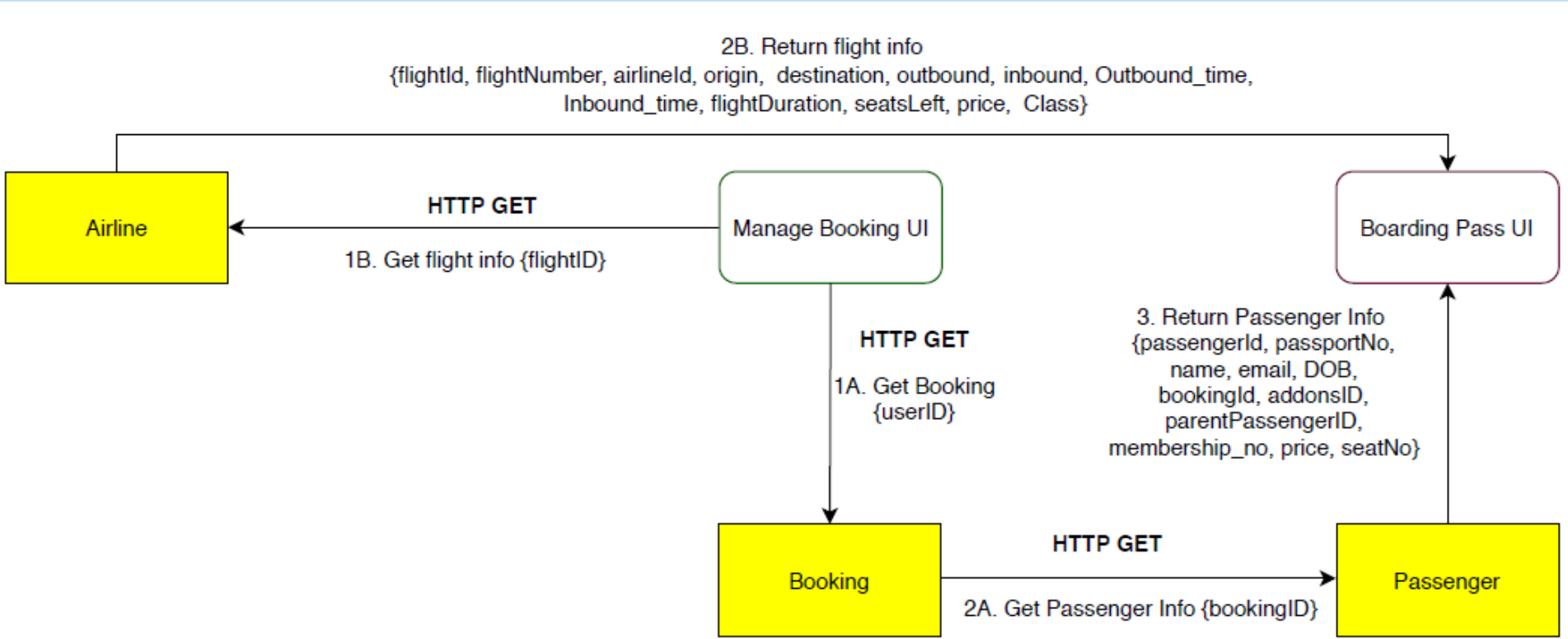
- User can navigate to the manage booking UI which displays a list of bookings made by the user
- **Each booking will have 3 options**: check-in, cancel and update details of the selected booking.
- If check-in is selected, status will then be updated and boarding tickets with passengers' details will be available for printing.

Scenario Overview Diagram 5

- Passenger checks in

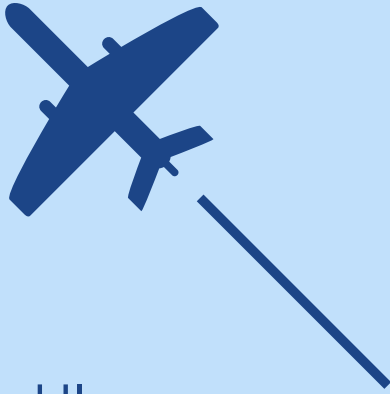


User Scenario Diagram 5
- Passenger checks in



Scenario 6

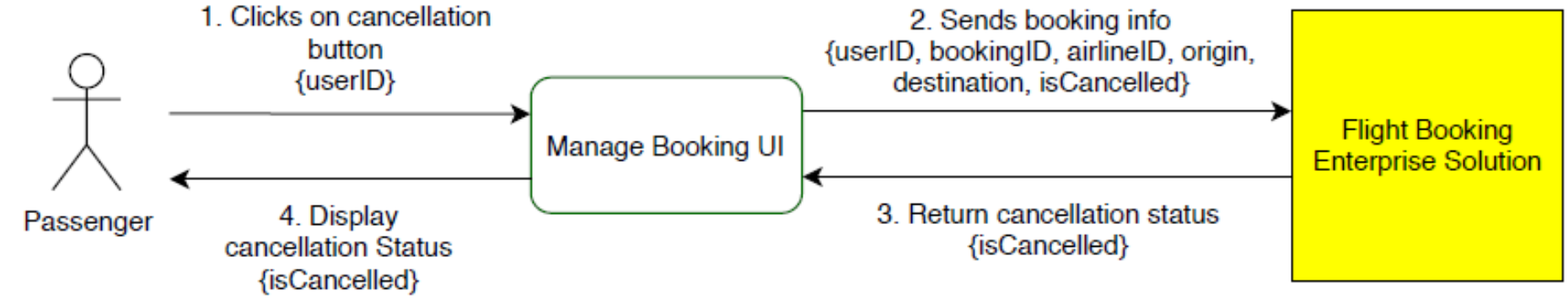
Passenger:



6) Passenger cancels a booking

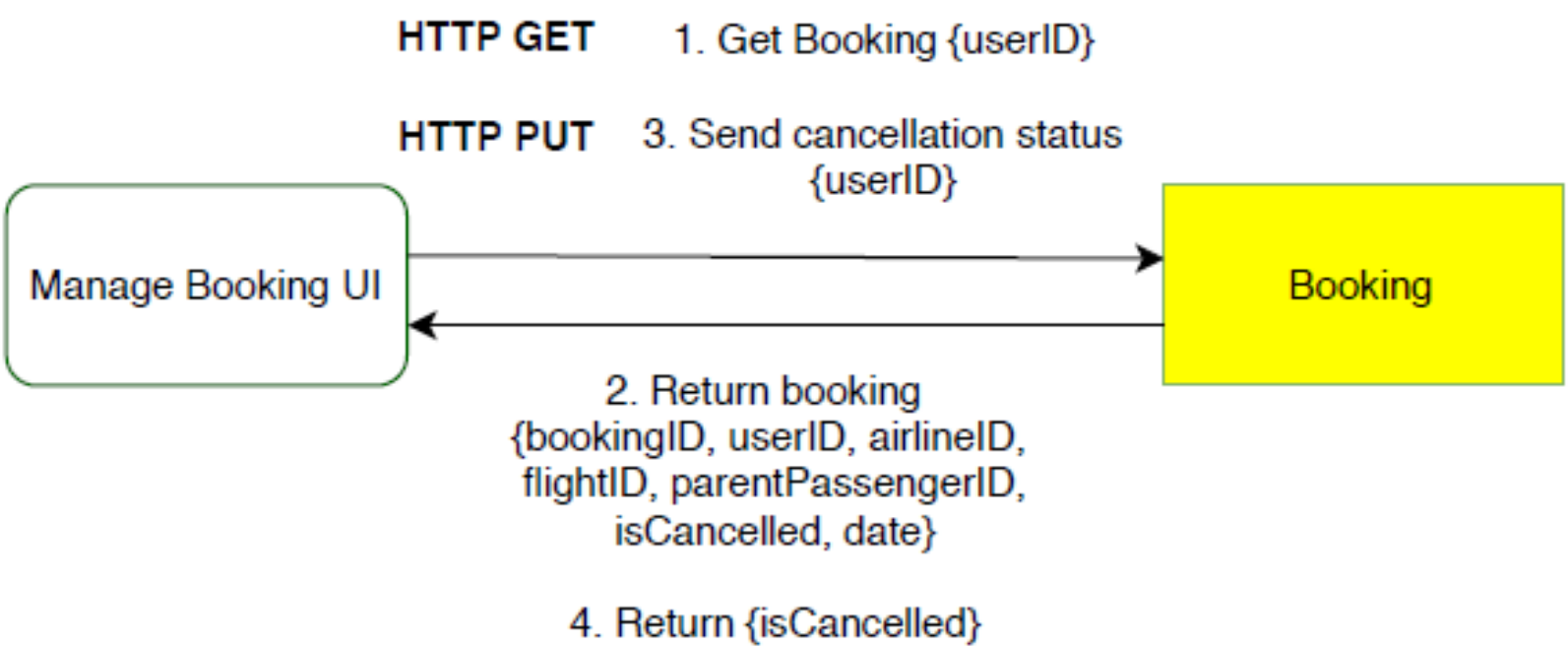
- Passenger cancels their booking at the Manage Booking UI

Scenario Overview Diagram 6
- Passenger cancels booking

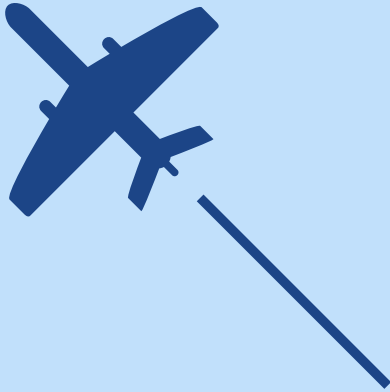


User Scenario Diagram 6

- Passenger cancels booking



Scenario 7



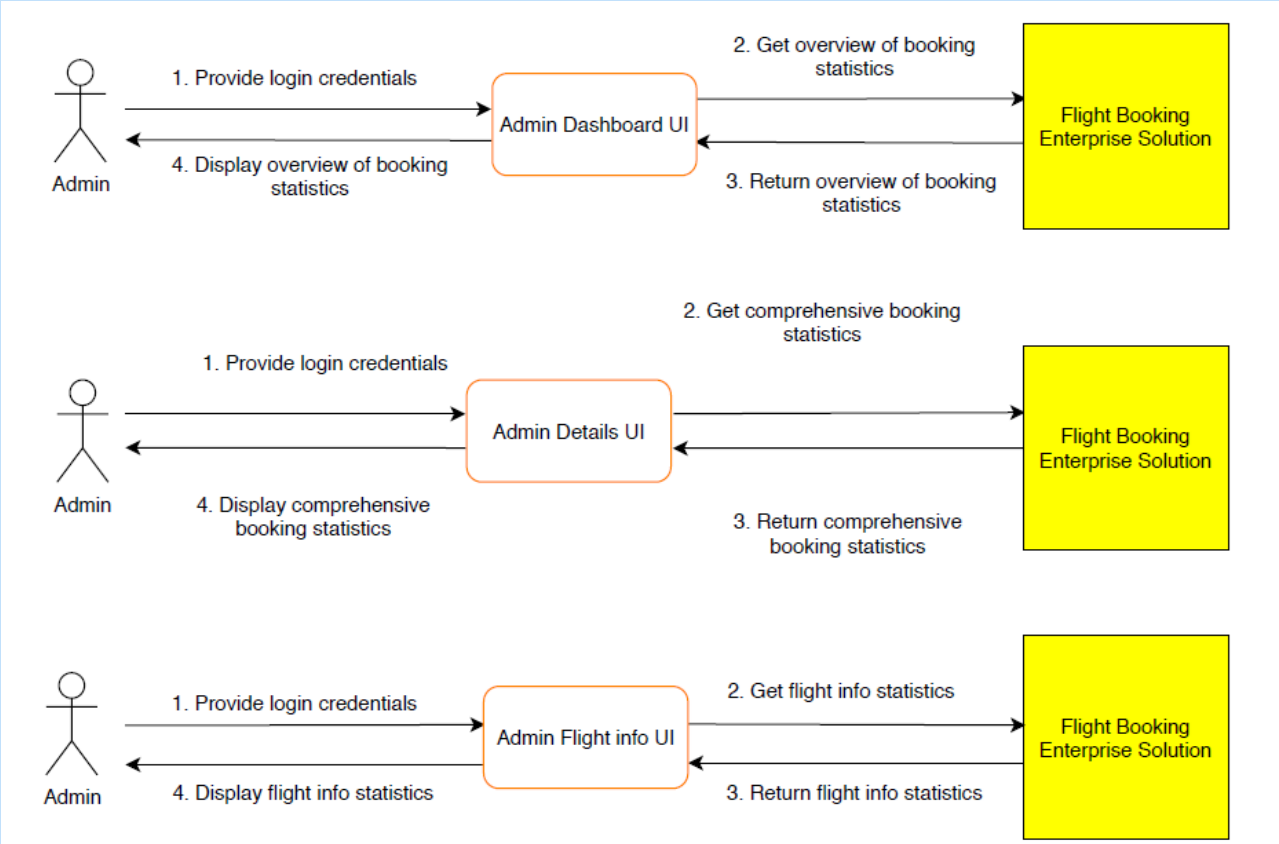
Airlines:

1) Admin views statistics

- Requires log in.
- Once admin logs in, admin is able to see an overview of booking details as well as specific booking details and flight information

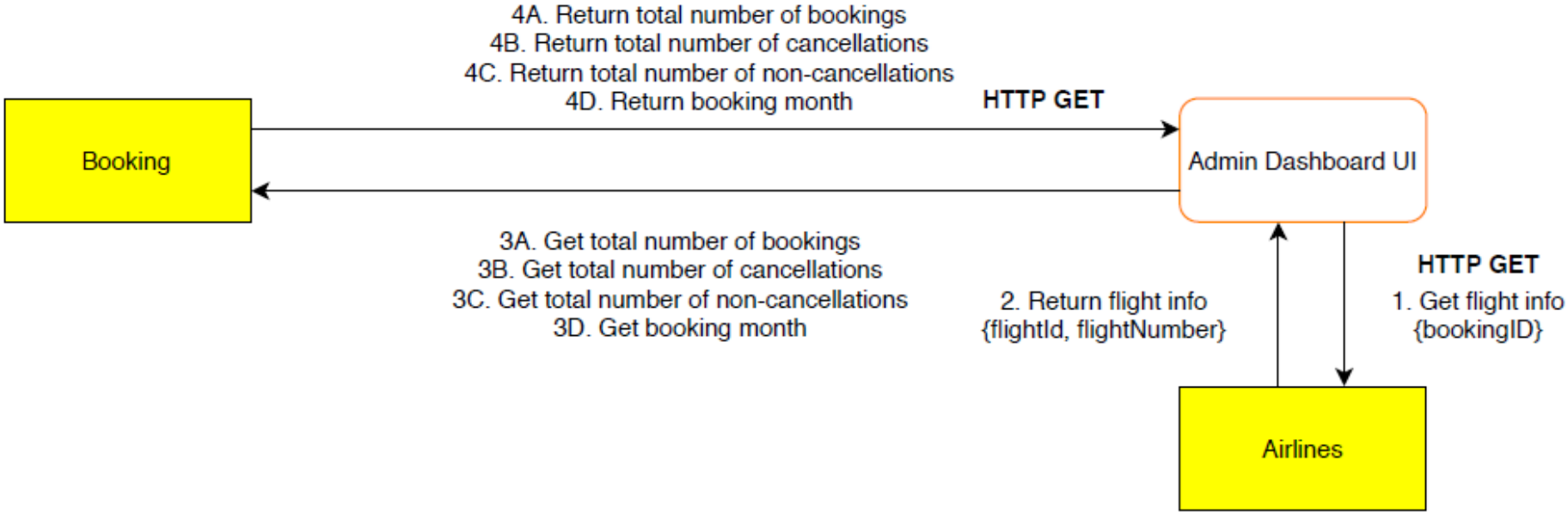
Scenario Overview Diagram 7

- Admin views statistics



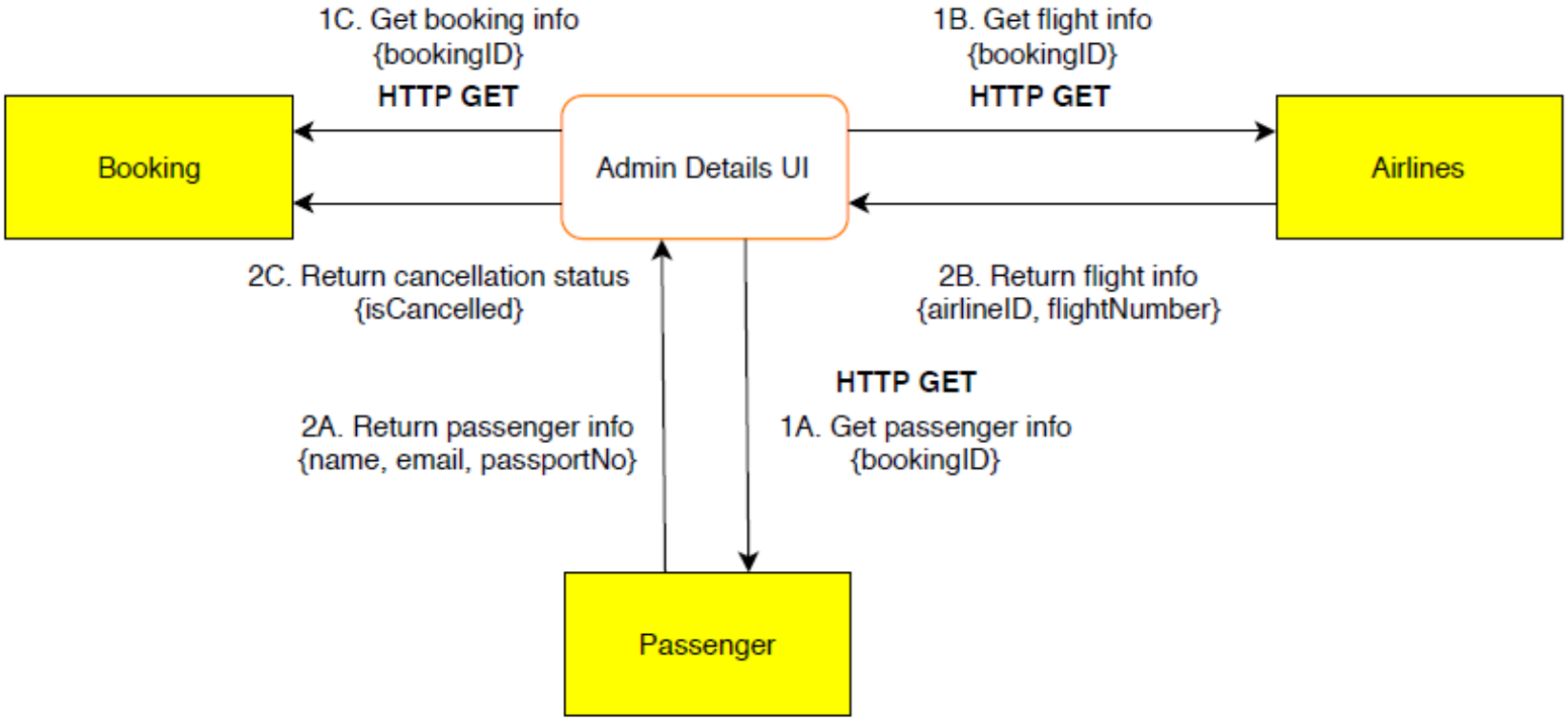
User Scenario Diagram 7

- Admin views statistics (overview)



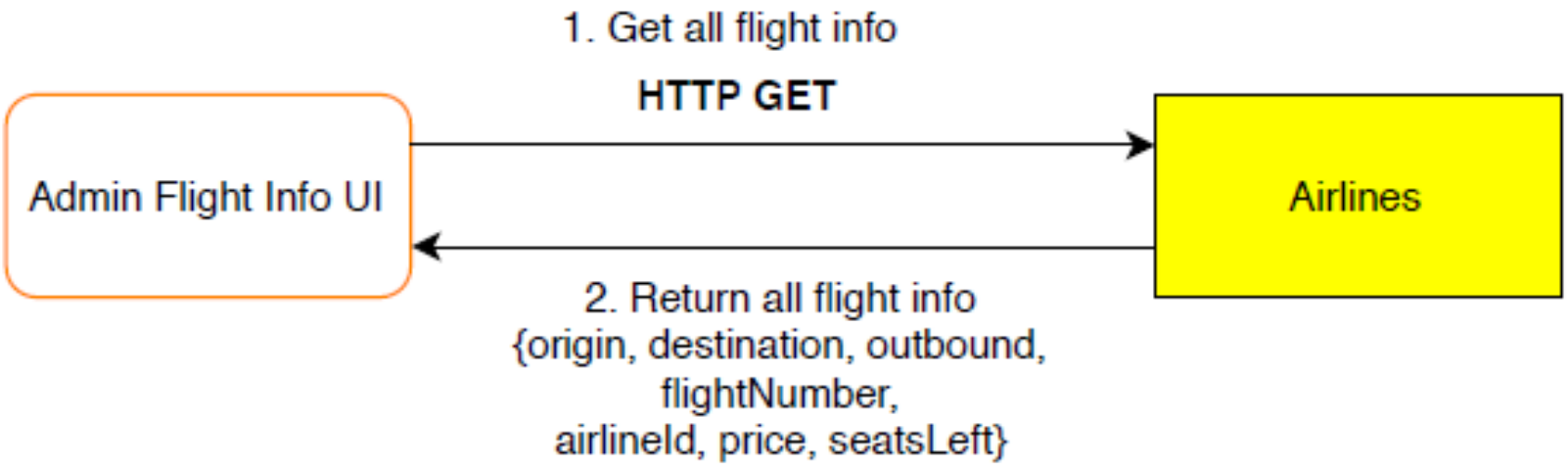
User Scenario Diagram 7

- Admin views statistics (comprehensive)



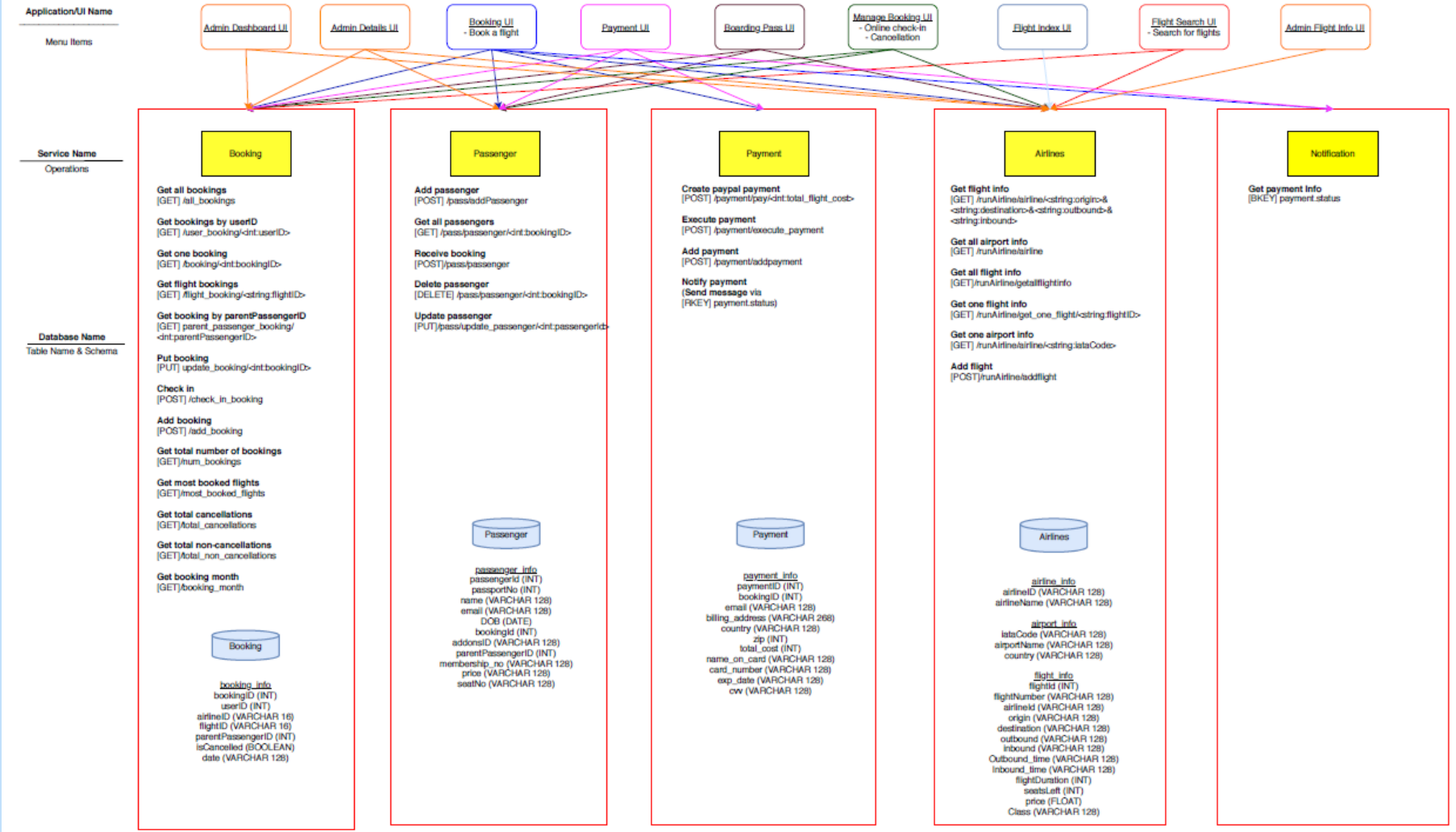
User Scenario Diagram 7

- Admin views statistics



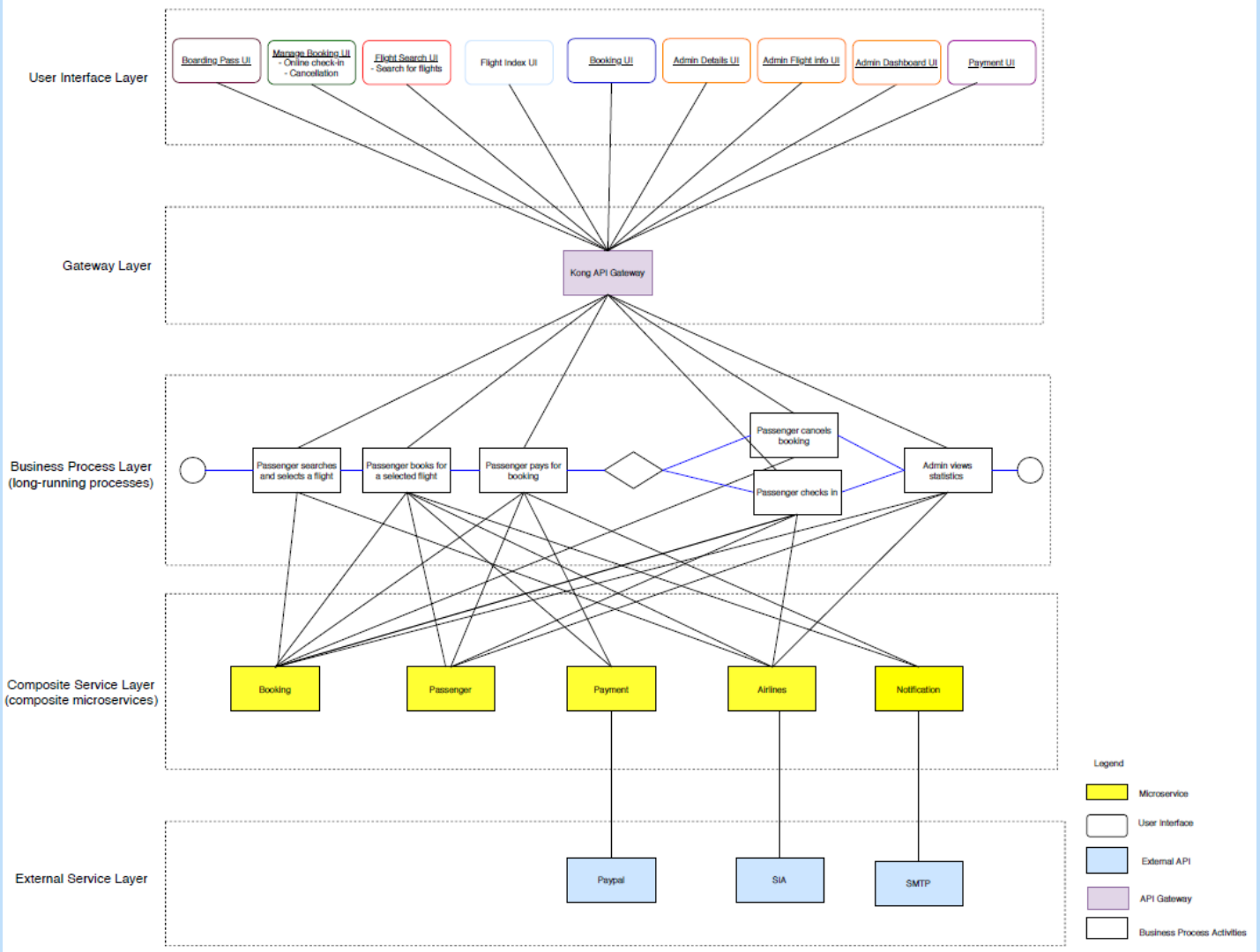
03

Technical Overview Diagram


Application/UI Name
Menu Items
Admin Dashboard UI
Admin Details UI
Booking UI
- Book a flight
Payment UI
Boarding Pass UI
Manage Booking UI
- Online check-in
- Cancellation
Flight Index UI
Flight Search UI
- Search for flights
Admin Flight Info UI
Service Name
Operations
Booking
Get all bookings
[GET] /all_bookings
Get bookings by userID
[GET] /user_booking/<int:userID>
Get one booking
[GET] /booking/<int:bookingID>
Get flight bookings
[GET] /flight_booking/<string:flightID>
Get booking by parentPassengerID
[GET] parent_passenger_booking/
<int:parentPassengerID>
Put booking
[PUT] update_booking/<int:bookingID>
Check in
[POST] /check_in_booking
Add booking
[POST] /add_booking
Get total number of bookings
[GET]/num_bookings
Get most booked flights
[GET]/most_booked_flights
Get total cancellations
[GET]/total_cancellations
Get total non-cancellations
[GET]/total_non_cancellations
Get booking month
[GET]/booking_month
Passenger
Add passenger
[POST] /pass/addPassenger
Get all passengers
[GET] /pass/passenger/<int:bookingID>
Receive booking
[POST]/pass/passenger
Delete passenger
[DELETE] /pass/passenger/<int:bookingID>
Update passenger
[PUT]/pass/update_passenger/<int:passengerId>
Payment
Create paypal payment
[POST] /payment/pay/<int:total_flight_cost>
Execute payment
[POST] /payment/execute_payment
Add payment
[POST] /payment/addpayment
Notify payment
(Send message via
[RKEY] payment.status)
Airlines
Get flight info
[GET] /runAirline/airline/<string:origin>&
<string:destination>&<string:outbound>&
<string:inbound>
Get all airport info
[GET] /runAirline/airline
Get all flight info
[GET]/runAirline/getallflightinfo
Get one flight info
[GET] /runAirline/get_one_flight/<string:flightID>
Get one airport info
[GET] /runAirline/airline/<string:iataCode>
Add flight
[POST]/runAirline/addflight
Notification
Get payment info
[BKEY] payment.status
Database Name
Table Name & Schema
Booking
booking_info
bookingID (INT)
userID (INT)
airlineID (VARCHAR 16)
flightID (VARCHAR 16)
parentPassengerID (INT)
isCancelled (BOOLEAN)
date (VARCHAR 128)
Passenger
passenger_info
passengerId (INT)
passportNo (INT)
name (VARCHAR 128)
email (VARCHAR 128)
DOB (DATE)
bookingId (INT)
addonsID (VARCHAR 128)
parentPassengerID (INT)
membership_no (VARCHAR 128)
price (VARCHAR 128)
seatNo (VARCHAR 128)
Payment
payment_info
paymentID (INT)
bookingID (INT)
email (VARCHAR 128)
billing_address (VARCHAR 268)
country (VARCHAR 128)
zip (INT)
total_cost (INT)
name_on_card (VARCHAR 128)
card_number (VARCHAR 128)
exp_date (VARCHAR 128)
cvv (VARCHAR 128)
Airlines
airline_info
airlineID (VARCHAR 128)
airlineName (VARCHAR 128)
airport_info
iataCode (VARCHAR 128)
airportName (VARCHAR 128)
country (VARCHAR 128)
flight_info
flightId (INT)
flightNumber (VARCHAR 128)
airlineId (VARCHAR 128)
origin (VARCHAR 128)
destination (VARCHAR 128)
outbound (VARCHAR 128)
inbound (VARCHAR 128)
Outbound_time (VARCHAR 128)
Inbound_time (VARCHAR 128)
flightDuration (INT)
seatsLeft (INT)
price (FLOAT)
Class (VARCHAR 128)

04

Service-oriented Architecture


User Interface Layer
Boarding Pass UI
Manage Booking UI
- Online check-in
- Cancellation
Flight Search UI
- Search for flights
Flight Index UI
Booking UI
Admin Details UI
Admin Flight Info UI
Admin Dashboard UI
Payment UI
Gateway Layer
Kong API Gateway
Business Process Layer
(long-running processes)
Passenger searches and selects a flight
Passenger books for a selected flight
Passenger pays for booking
Passenger cancels booking
Passenger checks in
Admin views statistics
Composite Service Layer
(composite microservices)
Booking
Passenger
Payment
Airlines
Notification
External Service Layer
Paypal
SIA
SMTP
Legend
Microservice
User Interface
External API
API Gateway
Business Process Activities

05

Beyond the Labs

Beyond the Labs

AnyChart

Demo

https://youtu.be/ysJCIZZ1zg1