**Flight Connection Builder Use Case**

Summary

Flight Connection Builder is an utility which will be used to find all possible flights arriving to a particular city which can be connected with all flights departuring from the same city. For example, if you want to travel from Mumbai (BOM) to New York (JFK) via Dubai (DXB), the utility will help us to find all possible flights arriving to Dubai from Mumbai which can connect with all flights departuring from Dubai to New York City.

Connection Builder Functional Description

Refer below the flight schedules between 2 cities. Let us assume that below schedules are daily schedules which means that you have 4 flights daily from BOM to DXB, 3 flights daily from DXB to JFK and 1 flight daily from DXB to IAH (Houston Airport).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Flight Schedules Data | | | | |
| Flight No | Departure Airport | Arrival Airport | Dep Time | Arr. Time |
| 501 | BOM | DXB | 4:30 | 6:00 |
| 2137 | BOM | DXB | 5:10 | 6:50 |
| 507 | BOM | DXB | 15:35 | 17:20 |
| 503 | BOM | DXB | 19:20 | 21:00 |
| 201 | DXB | JFK | 8:30 | 14:25 |
| 203 | DXB | JFK | 2:50 | 8:50 |
| 205 | DXB | JFK | 9:45 | 19:00 |
| 211 | DXB | IAH | 9:35 | 16:50 |

As a passenger, you want to travel from BOM to JFK via DXB. Use below logic to find all possible connections in DXB for BOMJFK route.

* Find all flights between BOM and DXB using above schedule data. You will get 501, 2137, 507, 503.
* Find all flights between DXB and JFK using above schedule data. You will get 201, 203 and 205.
* The connection time between 2 flights in DXB should be between 2 to 8 hours. The connection time can be called as a waiting time of a passenger in DXB airport. A passenger might need minimum 2 hours to deboard the BOMDXB flight and he should not wait more than 8 hours in DXB airport to board DXBJFK flight.

Using above logic, refer the possible connections for flight “501 BOMDXB” with all outgoing flight from DXB to JFK.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Flt No | Dep Airport | Arr Airport | Dep Time | Arr. Time |  | Flt No | Dep Airport | Arr Airport | Dep Time | Arr. Time |
| 501 | BOM | DXB | 4:30 | 6:00 | |  | | --- | |  | | 201 | DXB | JFK | 8:30 | 14:25 |
|  |  |  |  |  | f | 203 | DXB | JFK | 2:50 | 8:50 |
|  |  |  |  |  |  | 205 | DXB | JFK | 9:45 | 19:00 |

Below are the connection detail between BOM to JFK via DXB.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Onward Flight | Arr Time in DXB | Connecting Flight | Dep Time in DXB | Connection Time | Is a validation Connection | Reason |
| 501 | 6:00 | 201 | 8:30 | 2:30 | Yes | Connection time between 2 to 8 hours |
| ~~501~~ | ~~6:00~~ | ~~203~~ | ~~2:50~~ | ~~20:50~~ | **No** | It cannot connect on same day as 203 departs before 501 arrived to DXB. 203 can connect on next day. But the passenger has to wait 20hours 50 mins in DXB airport which is more than 8 hours |
| 501 | 6:00 | 205 | 9:45 | 3:45 | Yes | Connection time between 2 to 8 hours |
| ~~2137~~ | ~~6:50~~ | ~~201~~ | ~~8:30~~ | ~~1:40~~ | No | Connection time less than 2 hours |
| ~~2137~~ | ~~6:50~~ | ~~203~~ | ~~2:50~~ | ~~20:00~~ | No | It cannot connect on same day as 203 departs before 501 arrived to DXB. 203 can connect on next day. But the passenger has to wait 20hours in DXB airport which is more than 8 hours |
| 2137 | 6:50 | 205 | 9:45 | 2:55 | Yes | Connection time between 2 to 8 hours |
| ~~507~~ | ~~20:00~~ | ~~201~~ | ~~8:30~~ | ~~12:30~~ | No | It cannot connect on same day. For next day connection, the waiting time is 12:30 which is more than 8 hours |
| 507 | 20:00 | 203 | 2:50 | 6:50 | Yes | Connection time between 2 to 8 hours |
| ~~507~~ | ~~20:00~~ | ~~205~~ | ~~9:45~~ | ~~13:45~~ | No | Connection time is more than 8 hours |
| ~~503~~ | ~~19:20~~ | ~~201~~ | ~~8:30~~ | ~~11:50~~ | No | Connection time is more than 8 hours |
| 503 | 19:20 | 203 | 2:50 | 6:10 | Yes | Connection time between 2 to 8 hours |
| ~~503~~ | ~~19:20~~ | ~~205~~ | ~~9:45~~ | ~~13:05~~ | No | Connection time is more than 8 hours |

Assignment Detail

Create Spring Boot application using Microservice Architecture Pattern. The application should contain below backend services and a ReactJS based UI application.

1. Connection Builder Service

The service should accept departure and arrival airport code (3 letter) and return all possible connecting flight data. Refer “ConnBuilderInput.json”, “ConnBuilderOutput.json” for request/response payload of the service.

Technology:

Spring Boot Service,

Spring Data for data access,

H2DB as data store (prepare required data at the service start-up time)

Unit testing with any Mocking framework

1. Master Data Service

The service should expose end-points to “return list of all Airports” and services to perform CRUD (Create, Read, Update, Delete) operations.

Technology:

Spring Boot Service,

Spring Data for data access,

H2DB as data store (prepare required data at the service start-up time)

Unit testing with any Mocking framework

Below is Airport Master data

|  |  |  |  |
| --- | --- | --- | --- |
| Airport Code | Airport Name | City Name | Co-Ordinates |
| BOM | Chhatrapati Shivaji Maharaj International Airport | Mumbai | [19°05′19″N 72°52′05″E](https://geohack.toolforge.org/geohack.php?pagename=Chhatrapati_Shivaji_Maharaj_International_Airport&params=19_05_19_N_72_52_05_E_region:IN_type:airport) |
| DXB | Dubai Internation Airport | Dubai | [25°15′10″N 055°21′52″E](https://geohack.toolforge.org/geohack.php?pagename=Dubai_International_Airport&params=25_15_10_N_055_21_52_E_region:AE-DU_type:airport) |
| JFK | John F. Kennedy International Airport | New York | [40°38′23″N 073°46′44″W](https://geohack.toolforge.org/geohack.php?pagename=John_F._Kennedy_International_Airport&params=40_38_23_N_073_46_44_W_region:US-NY_type:airport) |
| IAH | George Bush Intercontinental Airport | Houston | [29°59′04″N 095°20′29″W](https://geohack.toolforge.org/geohack.php?pagename=George_Bush_Intercontinental_Airport&params=29_59_04_N_095_20_29_W_region:US-TX_type:airport) |

1. Spring Cloud API Gateway for rooting purpose

This service is going to act as a gateway for the services. It can be limited to do only routing to services based on url pattern.

1. Spring boot based ReactJS app which will serve UI assets to browser. The ReactJS application should have a page which will accept Departure and Arrival airport code and render Connection builder detail in a Map. It is a ReactJS based application where all ReactJS components are bundled as Jar and running as a spring boot application. Refer attached screen layout.

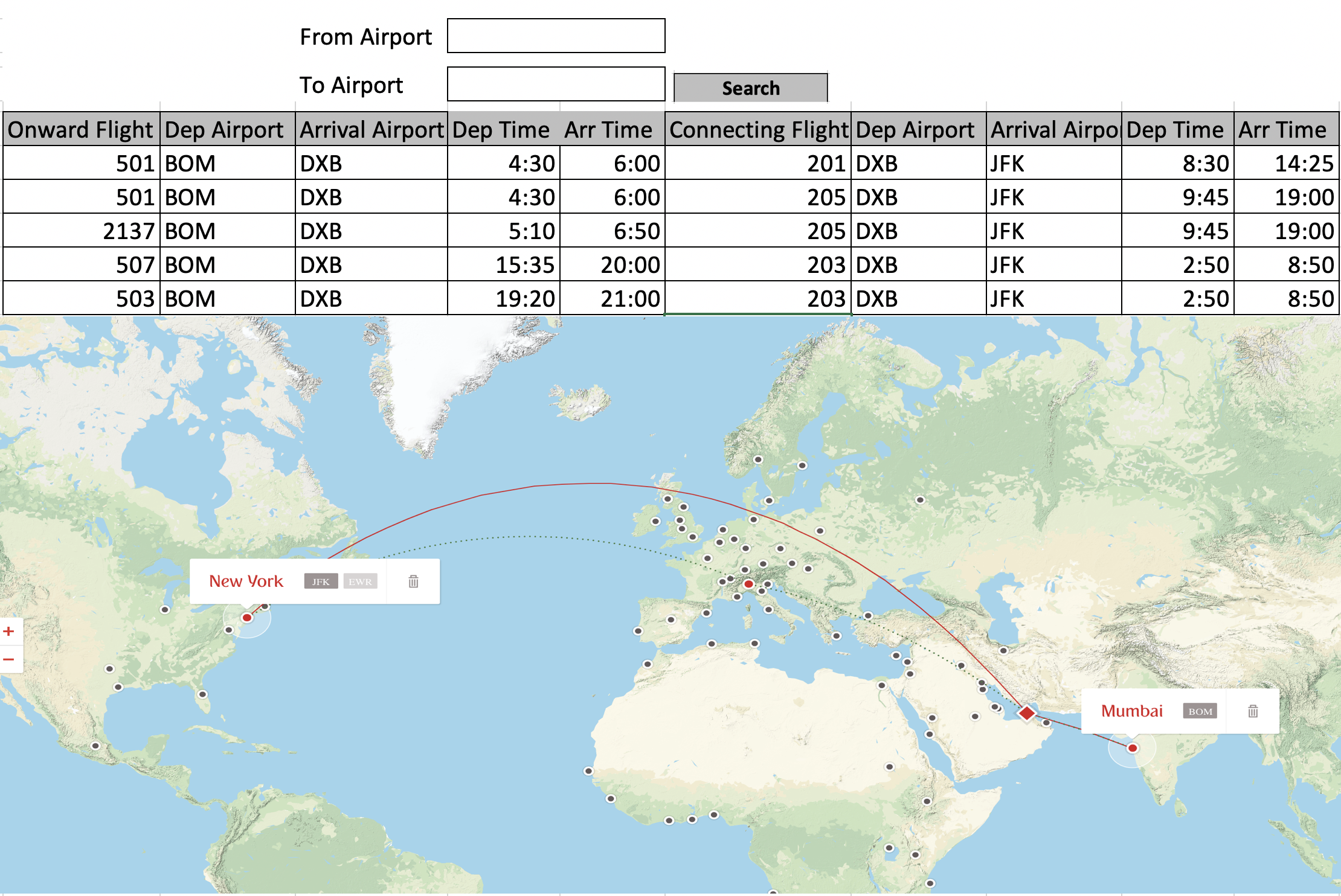
Technology:

Spring Boot Service,

ReactJS using type-script (or)javascript

MaterialUI for UI components

Any open-source map library



UI behaviour:

**From/To airport text field:** It should accept only characters. Maximum no of characters should be 3. It should be a auto-complete control. When user type a character, it has to show all airports start with entered character

**Connection Data Table:** On row selection, the map should show flight data in the map.

**Map Control:** It should show data based on selected row in connection table. Each city panel (Mumbai, New York city panel in above diagram. There is no city panel for DXB. But you should add panel for DXB city as well). On mouse hover of the city name, Below data should be shown in a tool-tip based on Departure (BOM), Connecting (DXB) and Arrival (JFK) airport.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| City Name: | BOM |  | City Name: | DXB |  | City Name: | JFK |
| Flight No | 501 |  | Flight No | 501 |  | Flight No | 201 |
| Flight Time | 4:30 (Dep Time) |  | Flight Time | 6:00 (Arr Time) |  | Flight Time | 14:25 (Arr Time) |
| Wait Time | - |  | Wait Time | 2h 30m |  | Wait Time | - |
|  |  |  | Flight No | 201 |  |  |  |
|  |  |  | Flight Time | 8:30 (Dep Time) |  |  |  |

The above screenshot is provided for reference purpose. You can follow your own approach. But it should plot city details in the map and it should be an interactive map.

Please do not limit yourself to what you can come up with, we love being surprised by your awesome ideas!

Project Guidelines/Instructions:

1. Please use your first & last name for the maven group id to identify your code uniquely
2. Every service should be a maven project and should be able to run as standalone spring boot service .
3. All the services should be grouped as "Maven Multi Module" project.
4. The Reactjs UI project should be bundled as Jar and should be able to run it as spring boot project.
5. Dependencies should be only retrieved from Maven Central Repository (https://repo1.maven.org/maven2/).
6. There shouldn’t be any build issues.
7. Project will be built using mvn install command.
8. Follow TDD approach for all services (Server and UI). Try to achive more than 80% coverage
9. Code coverage should be generated using jacoco.
10. Sonar analysis will be run from root of the maven multi module project. mvn -f pom.xml org.jacoco:jacoco-maven-plugin:prepare-agent clean install sonar:sonar