**INDEX**

**Approach and execution process**

**About BusBook (a brief or macro level requirement)**

**Modules (List of modules)**

**Technology and architecture**

APPROACH AND EXECUTION PROCESS

APPROACH

The whole Case Study project will be divided into many modules (35 – 45 modules). All functionality are divided in 3 groups with 3 different micro services and each group will work with one micro service and each participant of JEE Cloud -FLP will work on an individual module from each micro services. . Everyone will share a common database. The architecture and package structure (nomenclature) must be the same for everyone. Once the micro service and individual functionality is realized, it will be integrated in one system and upon successful integration, it will be deployed for project evaluation.

PROCESS (DAY WISE MILESTONES)

**Day 1 and 2:-**

* Project mentor will brief the associates about the different modules, and will assign each associate a unique module (as every associate will be working on a unique module)
* Associates will come up with data model (since everyone will be using only one database); the data model will be reviewed by mentor(s)
* So on day one - Module distribution and final DB model should be decided
* Product backlog and sprint backlog should be created by teams and share with Batch mentor by EOD.
* Create daily Scrum Chart team wise.
* The Associates will create a user-story (a low level requirement document, in which they will mention - the functionality they are going to work on and on how the end user will use it(all the constraints and exceptions will be there) This will be a one or two page document. It will send a user-story template for better understanding.
* We will provide the architecture; and the associates need to understand it, and run a dummy flow to ensure that they have understood the above said architecture and are able to use it
* So the day target will be - ensuring the understanding of the functionality and architecture

Day 3:-

* All associates will present a white board presentation to the mentor(s) and the whole batch. Here each associate will describe - what he/she is going to build and how. Which all tables (DB) are going to be affected and what dummy data will he/she be using etc. Basic idea of this exercise is to ensure that - everyone (developer, team mates and mentors) are on the same page in terms of requirement and approach. Here the mentor may prevent any future mistake that might come out of misunderstanding of requirement or out of wrong approach.
* UI (layout, color coding, look and feel etc) will be discussed on this day only
* Use Spring boot rest controller to distribute data.
* One of the possible findings of this discussion may be some changes in the DB. So final database creation will be last task of the day.
* Associate will send final draft of their individual user story to mentor(s) for review.
* By day 2, the DB model (database), page layout, HTML, architecture (SpringBootMVC), package structure is ready. Now the associates will create dummy data in database and will start real work on their respective modules and complete by Day 3 only.

**Day 4, 5, 6:-** Integration and testing and Deployments and UAT by mentor(s)

**Day 7:**- Sprint Review and Sprint retrospective.

**A BRIEF ON BUS Booking Application**

**ABOUT BUSBOOK**

This project presents a review on the software program "Online Bus Ticket Reservation System" as should be used in a bus transportation system, a facility which is used to reserve seats, cancellation of reservation and different types of route enquiries used on securing quick reservations. OBTRS is built for managing and computerizing the traditional database, ticket booking and tracking bus and travel made. It maintains all customer details, bus details, reservation details.

 The software achieved is capable of improving the customer hand and relationship management in ITC operations. It is recommended that despite the present functionality of the designed software, an additional functionality such as the use of E-mail to send tickets and notifications to the customer and an online payment using credit cards/debit cards should be implemented into the system. Furthermore, other operations carried by ITC such as the courier services should also be integrated in order to enhance the system.

**Macro level Operations/offerings:**

1. **User Management** :- Signup, Login or Logout , Create/updating profile, Check Travel History ,Password encryption, Change password, Update Password
2. **Booking Management** - Accept Booking details and list all bus details, Search Bus based on route and Date, Sort Buses based on Rating, Price feedback etc., Changed Search Criteria, Bus Ticket booking, ticket payment, cancellation of tickets, searching bus routes

Develop the application using Micro service architecture as follows.

* **User or Customer Management Micro service**
* **Booking Management Micro service.**
* **Admin Management Micro service**

Go through all the 24 functionality given in the follow table. Segregate it as per the above 3 micro service. Register each micro service with Eureka

**Enhancements for Future:-**

1. Chat Facility.

2. Customized Advertisements and event notifications.

4. Applications

5. SMS based notification

**MODULE LISTS**

|  |  |  |
| --- | --- | --- |
| **S no.** | **Functionality** | **Brief introduction of the functionality** |
| 1 | sign up | Signup or registration - here non existing user of BusBook will register or sign up for it; the email id will be user's login id. An activation mail will be send to the user (to authenticate if user's email id is valid) and on activation only user will be able to access his/her account |
| 2 | Login/Logout/session | User will login to the system, session will be established and can logout |
| 3 | User profile page | User's page. The page content will be different for logged in user and other user. After successful login - user will navigate to this page (we may term it as user's home page). The logged in user should have features to see his travel history. |
| 4 | Password encryption | The password of the user must be encrypted in database - and will be decrypted while login and change password and forget password feature |
| 5 | Change password | User may change his/her BusBook account password. The new password should not be same as old one, and should also fulfill all the password related constrains of signup time. |
| 6 | Forget password | an email contain user's password (decrypted) will be send to the user |
| 7 | User profile & Privacy settings | A user may set additional info about himself/herself like address, phone no. |
| 8 | Logged in user profile page content | On home page for logged in users, Bus booking feature should be available. Accept the booking details and accordingly display the list of all available bus details. |
| 9 | Search Bus | Customer should have the feature to search the bus for particular route and date details. After clicking on search button all the available bus details should displayed. |
| 10 | Sorting Bus Details | ON search page, sorting feature should be available so that customer can sort bus details using bus rating, price, early departure and late departure criteria. |
| 11 | Modify Search | Customer should be able to modify the search details like From City, To City and Travel Date. |
| 12 | add Route/Bus | Admin can add/update new buses and routes, change bus timings. Cancel particular bus and routes. |
| 13 | Generate reports | Admin should be able to see bookings for particular bus and route. |
| 14 | Update bus Details | Admin should be able to update the bus and route details like arrival time, .departure time, From and To stops. |
| 15 | Bus Booking | Once the customer selects the bus and seats, accept further details from customer and confirm the booking. |
| 16 | Generate Ticket | Once the booking is done, ticket should be displayed to the user .and customer should have option to send that ticket on mail as well as print the ticket. |
| 17 | Cancel Booking | Customer can cancel the booking and accordingly all data should be updated. |
| 18 | Refund Money | A Customer can cancel the booking up to 2 hours before the bus departs and then money will be refunded to the customer |
| 19 | Display fare | Once the customer selects the bus, display the fares after accepting no of passengers from user. Also display the fare distribution. |
| 20 | Seat availability | Customer should be able to see seat availability for selected bus and then do the booking. |
| 21 | Booking Details | Once the booking is done, customer should be able to see the booking details before final payment. If any correction is required he/she should have the feature to edit it. After booking available seats should be updated. |
| 22 | Give feedback | Customer should have facility to provide the feedback on his journey and suggestions on bus services. |
| 23 | Bus rating | Customer should be able to rate the bus. |
| 24 | home/profile page - integration | Integration of Admin Page, Customer Page. |

**TECHNOLOGY AND ARCHITECTURE**

**TECHNICAL SPECIFICATIONS**

**1. Software Requirement:**

Spring Source Framework 2.5.2

My SQL 5.0

JDK 1.8

• Jenking – Build Server

• SonarQube – Code Quality

• Git - Source Code Repository

• Postman

**3. System Architecture:**

1. **Angular - Front end UI**
2. **Spring Boot – Rest Controller – Back end functionality**

**Two API need to be design, one will get data from server (b) and another API will project the data using Angular (a).**

Angular Service

HTML page with Angular component

**Angular - Front end UI**

|  |  |
| --- | --- |
| Parameters | Marks |
| Individual Module Implementation | 40 |
| Integration and UI | 40 |
| Presentation | 20 |
| Total | 100 |

JSON Object Provided

Call Rest controller

Service

Dao - JPA

Rest Controller

**Spring Boot – Rest Controller – Back end functionality**