**Integrated Assignment**

**Developing Single Page Application using Angular**

**Instructions to use the project file provided:**

* You need to code the assessment in the folder containing partially coded project
* Read the problem statement, examples and the other details provided carefully and implement the solution
* Download the project **InfyAirlines-toTrainee** to your system
* **Backend** folder contains the **database JSON server**
* **InfyAirlinesUI** folder must contain the **client-side code**
* **DO NOT** alter the function name or the argument list of the function that is provided to you
* **DO NOT** alter the **meta data** provided in the components
* **DO NOT** add any new functions apart from the one given in the file where you write the solution
* Use the **bootstrap classes** wherever necessary
* Do not modify any **meta data** provided in the files

**Note:**

* By default, the JSON server runs on the port **3000**
* Execute **npm install** command in both the **Backend** and **InfyAirlinesUI** folders
* Execute **npm start** in both **InfyAirlinesUI** and **Backend** folder to run the frontend and backend servers.

**Problem Description:**

**InfyAirlines** wants to automate the process of booking airlines tickets to their customers. The following functionalities are required to be implemented.

* Display the list of airlines available for Booking
* Book the required Number of seats in the selected airlines
* Navigate to PageNotFound Component when a page is requested which does not exist

You need to implement these requirements one by one to complete the application for Infy Airlines

**Details of REST Backend Endpoints**

|  |  |  |
| --- | --- | --- |
| **URL** | **Method** | **Response** |
| <http://localhost:3000/airlines> | GET | An API call to this URL will return an array of all the available airlines |
| <http://localhost:3000/bookings> | POST | An API call to this URL with proper form data will help in booking the required number of Tickets |

**Implementation Details of InfyAirlinesUI folder**

|  |  |
| --- | --- |
| **File Name** | **Description** |
| app/app-routing.module.ts | **To be**  **Implemented** |
| app/app.component.html |
| app/infy-airlines.service.ts |
| app/airlines-list/airlines-list.component.ts |
| app/airlines-list/airlines-list.component.html |
| app/book-ticket/book-ticket.component.ts |
| app/book-ticket/book-ticket.component.html |
| app/app.component.ts | **Implemented** |
| app/app.module.ts |

**~~~~~~~~~~~To Be Implemented~~~~~~~~~~~**

1. Configure the routes with the below information:

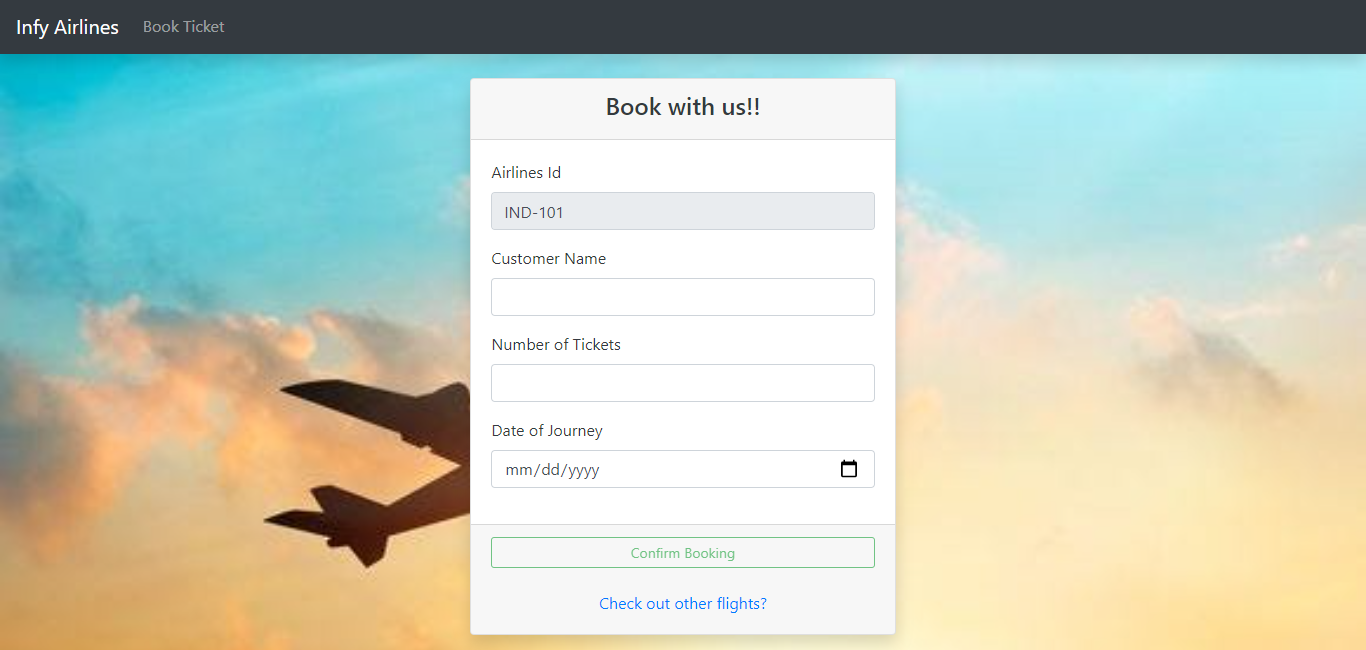
|  |  |
| --- | --- |
| **Route Path** | **Component** |
| *airlines* | AirlinesListComponent |
| *book-ticket/:airlinesId* | BookTicketComponent |

* In case if an **Invalid URL** is provided, the user must be redirected to the **airlines Page**

1. When http://localhost:4200 is entered in browser, **Airlines** page should be displayed as below:



1. On Click of **Book Ticket** Button for any of the airlines, it should navigate to the BookTicketComponent having the view as shown below:

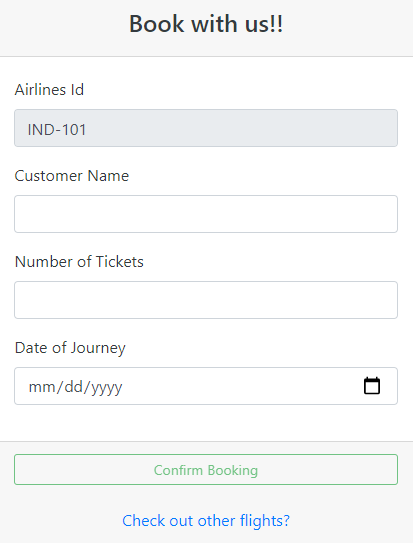


1. Add the validations for the given form controls appropriately

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Form Control Name** | **Validations to be Added** | **Default Value** |
| Airlines Id | airlinesId | Required Validation | Value of **airlinesId** property |
| Customer Name | customerName | Required Validation, should have only alphabets and space but cannot start or end with a space | “” |
| Number of Tickets | noOfTickets | Required Validation, should be atleast 1 and not more than 6 | “” |
| Date of Journey | dateOfJourney | Required Validation, validateDate custom validation | “” |

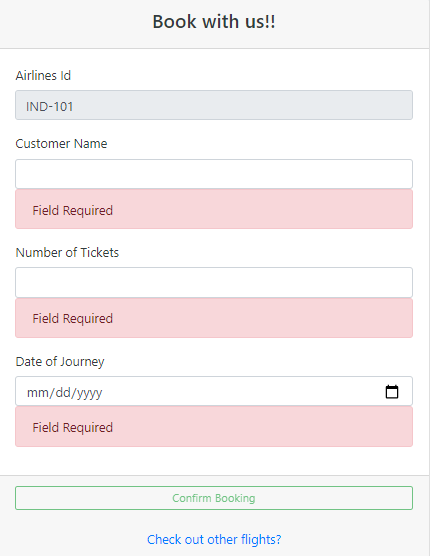
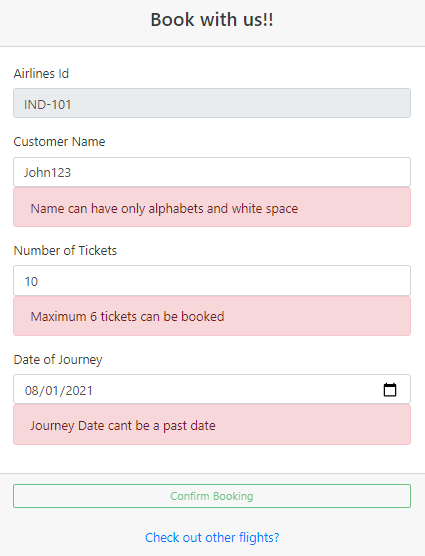
* **airlinesId** field should be permanently disabled
* **Book Ticket** button should remain disabled until the form becomes valid

1. When **Book** button is clicked from the Airlines Page, it should render the **Booking** Form as shown below:

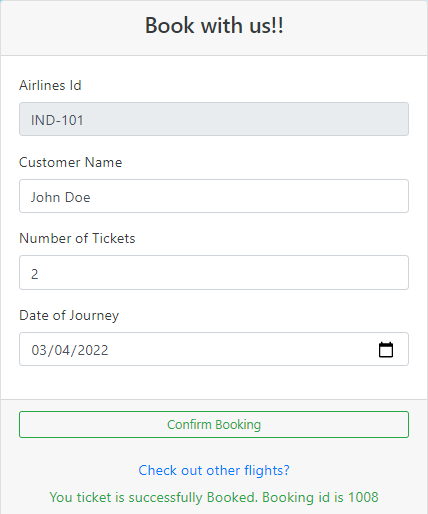


1. Validation Error messages should be rendered as below:

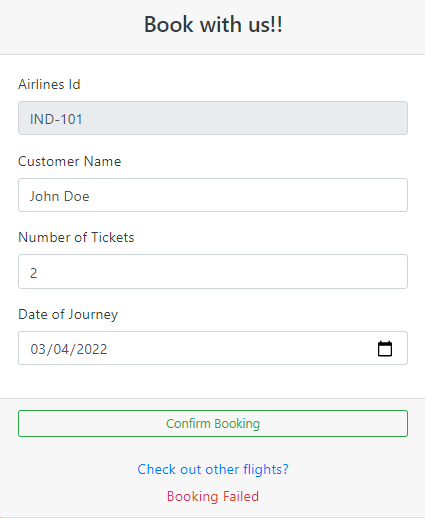
**Required Validations Other Validations**

1. On successful execution, the success message should be displayed as shown below:



1. In case if the booking fails due to server issues, it should render the error message as shown:



1. On Click of **Check out other Flights ?** button, it should navigate to the **airlines** page.