**Project** – Booking-Website

**Document** – Backend API documentation

**Developers:**

1. Ahamad Shaik
2. Manish Alluri

**APIs:**

1. Search Airport API
2. Search Flights API
3. Price Details API
4. Payment API
5. Email API

**Search Airport API:**

**Description:**

* This API functionality is implemented to enhance the customer interface experience by recommending the Airport names that match the customer search criteria.
* When the customer enters an input into the source and destination text fields in the home page, this API is called with the input string attached as a request body to the API.
* The API performs necessary implementations and generate the data as shown in the example below in the form of json.
* This json is then processed at the front end and displayed to the customer as a drop down.

**Acceptance Criteria:**

* The API is called when the customer input consists of 3 characters.
* The 3 characters might be the Airport code, or the first 3 characters of the city or the Airport name.
* When the input text field is selected and no data is entered, then the API returns all the existing Airport details.
* The input of the request should be either blank or a 3-character string.
* The API response can be an array of one or more jsons.

**Development URL:** http://localhost:8080/SrchArptAPI

**Request Json:**

{

    "ArptSrchString":" " //null or 3 characters (ex: MCO)

}

**Response Json:**

[{

        "cityname": "”, //Airport located city

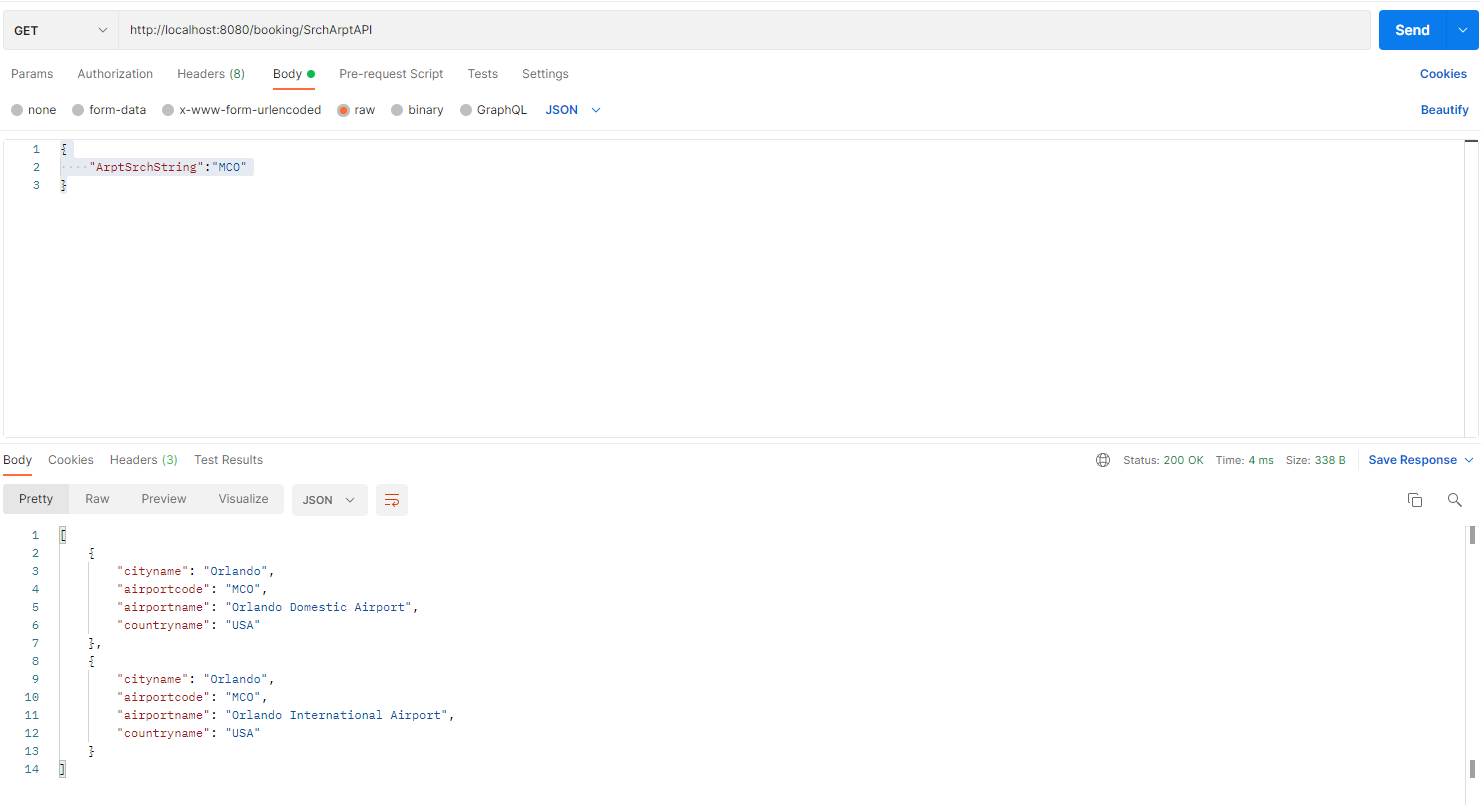
        "airportcode": "”, //3-character Airport code

        "airportname": "”, //Name of the airport

        "countryname": "” //USA

  },…]

**Test Sample:**



**Search Flights API:**

**Description:**

* This API functionality is implemented to get the available travel options that match the customer search criteria.
* When the customer enters an input into the necessary text fields in the home page and clicks the ‘Search’ button, this API is called with the input data attached as a request body to the API.
* This API is called when the customer filters the data by selecting the categories available and then clicks the ‘Apply Filters’ button.
* The API performs necessary implementations and generate the data as shown in the example below in the form of json.
* This json is then processed at the front end and displayed to the customer in GUI.

**Acceptance Criteria:**

* The API is called when all mandatory fields as shown below are filled.
* When API call is done by the search button, all the filter’s values are set to default.
* The response also includes an Airline ID that is mapped uniquely to every travel option available.
* The API response can be an array of one or more jsons.

**Development URL:** http://localhost:8080/booking/searchFlights

**Request Json:**

{

    "sourceName":" ", //Mandatory Field

    "destinationName":" ", //Mandatory Field

    "startDate":" ", //Mandatory Field

    "endDate":" ", //Mandatory Field

    "isRoundTrip":" ", //Mandatory Field (Boolean Value)

    "AirlineFilter":“ ",

    "ArrivalTimeFilter":"",

    "DepartureTimeFilter":"",

    "PriceRangeFilter":"",

    "JourneyTimeFilter":""

}

**Response Json:**

[

    {

        "flightnumber": " ",

        "departuretime": " ",

        "arrivaltime": " ",

        "sourceairport": " ",

        "destinationairport": " ",

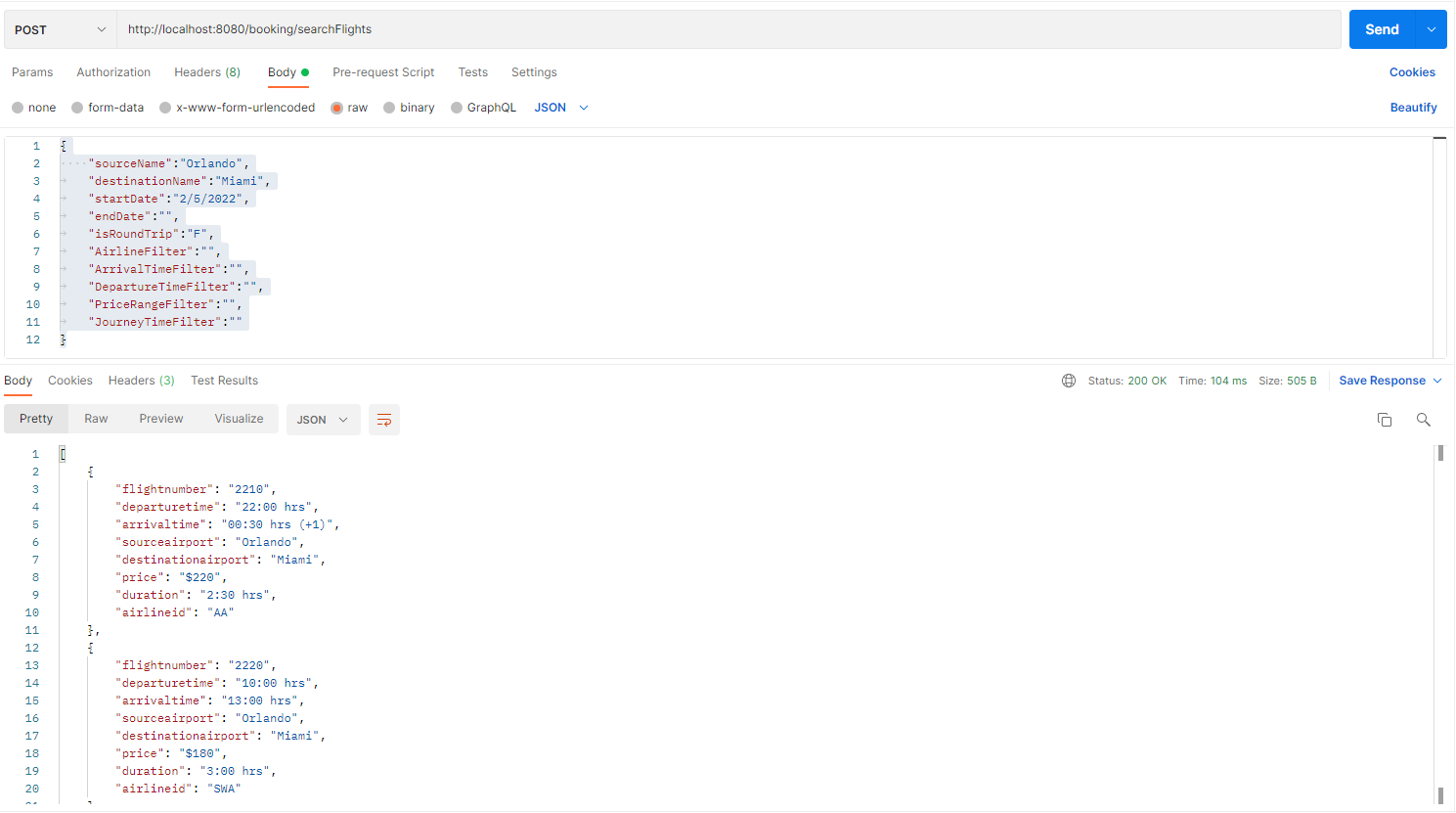
        "price": " ",

        "duration": " ",

        "airlineid": " "

    },…]

**Test Sample:**



**Price Details API:**

**Description:**

* This API functionality is implemented to get the available ticket pricing options for the selected flight plan by the customer.
* When the customer clicks on any one of the travel options provided, this API is called with the unique airline ID mapped to that travel option as input request to the API.
* The API performs necessary implementations and generate the data as shown in the example below in the form of json.
* This json is then processed at the front end and displayed to the customer in GUI.

**Acceptance Criteria:**

* The API is called when the mandatory field as shown below is filled.
* The response includes two prices standard and flexible in regards to the airline policy.
* The response is a single json.

**Development URL:** http://localhost:8080/booking/price

**Request Json:**

{

    "ID":" " //Mandatory Field

}

**Response Json:**

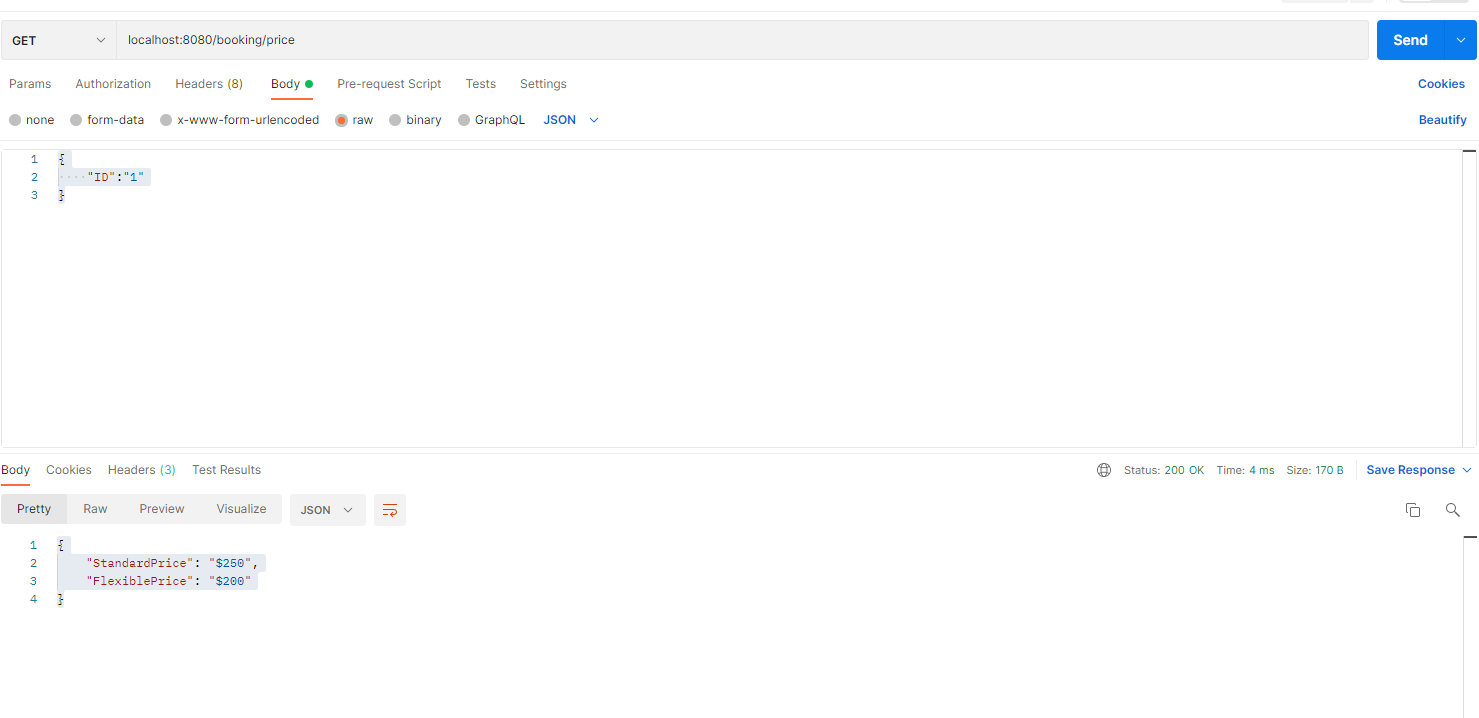
{

    "StandardPrice": " ",

    "FlexiblePrice": " "

}

**Test Sample:**

****

**Payment API:**

**Description:**

* This API functionality is implemented to confirm the booking to the selected flight plan by the customer by completing the payment using the payment information provided.
* When the customer clicks on the ‘checkout’ button, this API is with the input data attached as a request body to the API.
* The API performs necessary implementations and generate the data as shown in the example below in the form of json.
* This API makes an additional http request to the email API prompting it to send an 6-digit OTP to the email address provided by the customer.
* This json is then processed at the front end and booking status is updated to the customer in GUI.

**Acceptance Criteria:**

* The API is called when the mandatory field as shown below is filled.
* The response includes an Otpcode that is needed to authenticate the user as an authorized payer.
* After the response is returned a popup requires an OTP to be filled by the customer which is matched with the field ‘Otpcode’ of the response.
* The response is a single json.

**Development URL:** http:// localhost:8080/booking/payment

**Request Json:**

{

    "CustomerName":" ", //Mandatory

    "MobileNumber":" ", //Optional

    "EmailAdd":" ", //Mandatory (email Address)

    "JouneyInfo":" ", //Mandatory

    "PaymentInfo":" " //Mandatory

}

**Response Json:**

{

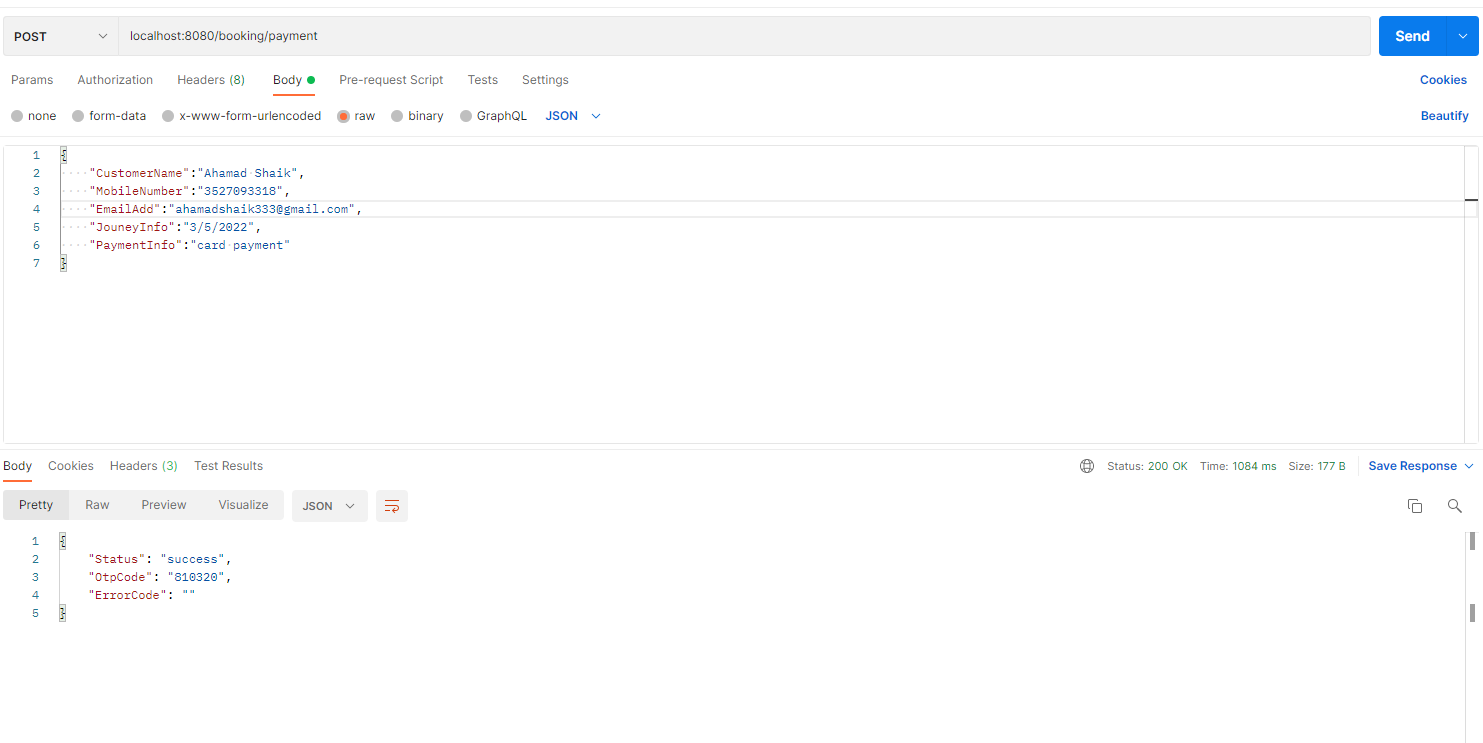
    "Status": " ", //payment status of the customer

    "OtpCode": " ", //6-digit code for authentication

    "ErrorCode": " “ //returns in case of any error

}

**Test Sample:**



**Email API:**

**Description:**

* This API functionality is implemented to send an 6-digit OTP to the customer email for authentication at the front end.
* This API also sends a booking confirmation email to the customer after the payment process is successful.
* The API performs necessary implementations and generate the data as shown in the example below in the form of json.
* This json is then processed at the front end and booking status is updated to the customer in GUI.

**Acceptance Criteria:**

* The API is called when the mandatory field as shown below is filled.
* The response includes an Otpcode that is needed to authenticate the user as an authorized payer.
* When no data is provided in the ‘msg’ field of the input, then it is an authentication email.
* If a msg exists, then it is a booking confirmation email.
* The response is a single json.

**Development URL:** http://localhost:8080/booking/email

**Request Json:**

{

    "Email": " ", //mandatory

    "msg":"" //Optional

}

**Response Json:**

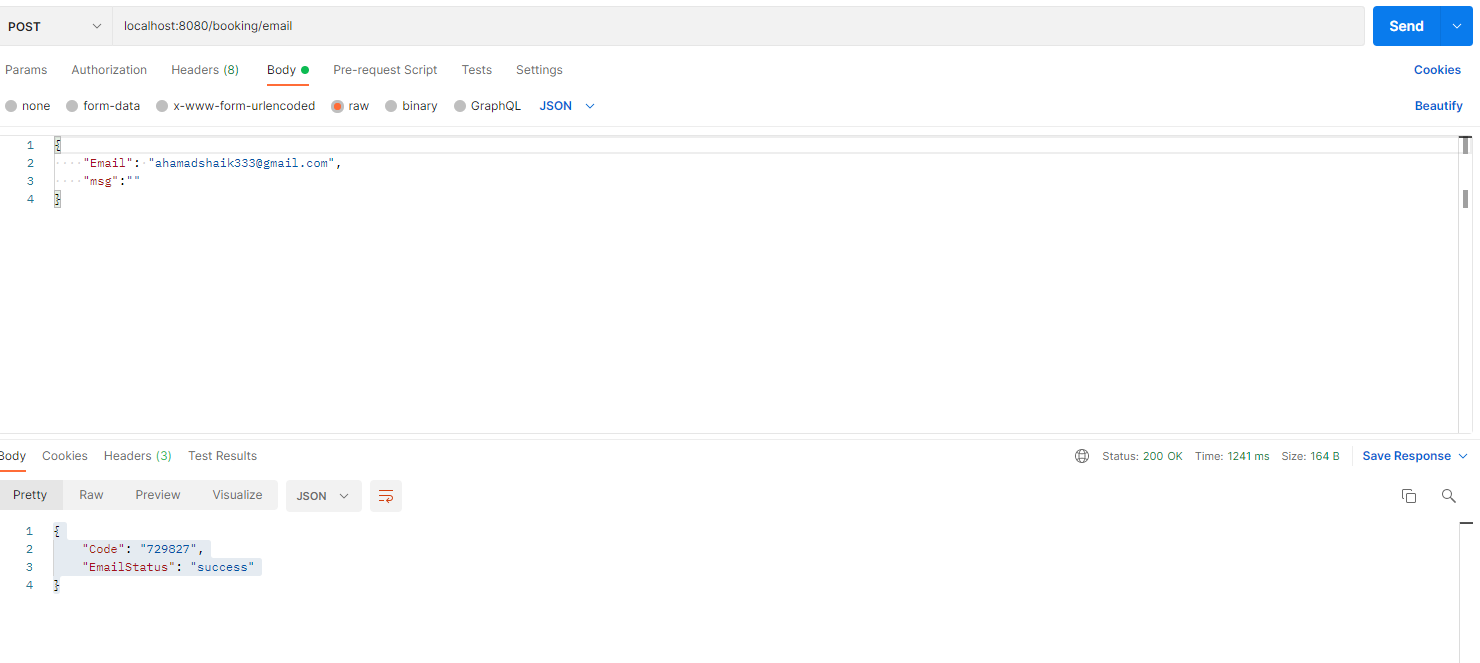
{

    "Code": " ",

    "EmailStatus": " "

}

**Test Sample:**

****