SOFTWARE ENGINEERING (CEN5035)

RideShare

Backend API documentation

Developers:

1. Kalyan GVS
2. Dheeraj Kumar M

API's Developed:

1) signup
2) login
3) user
4) delete
5) PostaRide
6) SearchaRide

1) **signup** :

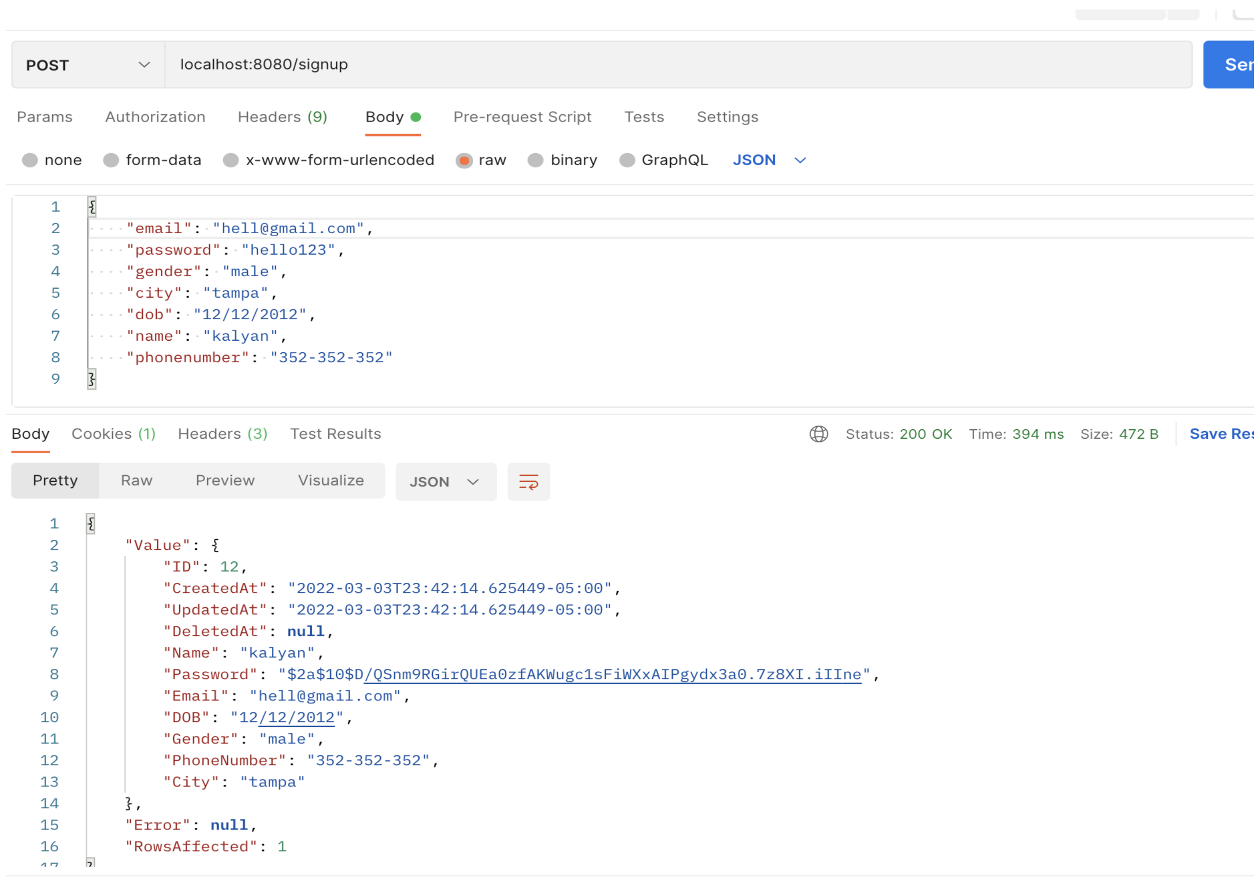
Description**:**

- Method: POST
- This API allows the new users to register on the application.
- Once the user is registered, they can login into the website whenever, using their email address and password.

Acceptance Criteria**:**

- For registering, the following fields are required to be entered by the user: *Name, Email, Password, Gender, City, DateofBirth, PhoneNumber.*
- If any/all of these fields weren't entered, the API throws an error message.
- Once all the inputs are properly entered by the user, all these details are updated to the *users* table in the database.
- If an user tries registering again using the same email, the API throws an error message: "*User already registered*"

Sample Requests and Responses:



localhost:8080/signup

```
{
    "email": "hell@gmail.com",
    "password": "hello123",
    "gender": "male",
    "city": "tampa",
    "dob": "12/12/2012",
    "name": "kalyan",
    "phonenumber": "352-352-352"
}
```

Status: 200 OK Time: 394 ms Size: 472 B

```
{
    "Value": {
        "ID": 12,
        "CreatedAt": "2022-03-03T23:42:14.625449-05:00",
        "UpdatedAt": "2022-03-03T23:42:14.625449-05:00",
        "DeletedAt": null,
        "Name": "kalyan",
        "Password": "$2a$10$D/QSnm9RGirQUEa0zfAKWugc1sFiWXxAIPgydx3a0.7z8XI.iIIne",
        "Email": "hell@gmail.com",
        "DOB": "12/12/2012",
        "Gender": "male",
        "PhoneNumber": "352-352-352",
        "City": "tampa"
    },
    "Error": null,
    "RowsAffected": 1
}
```

2) **login**:

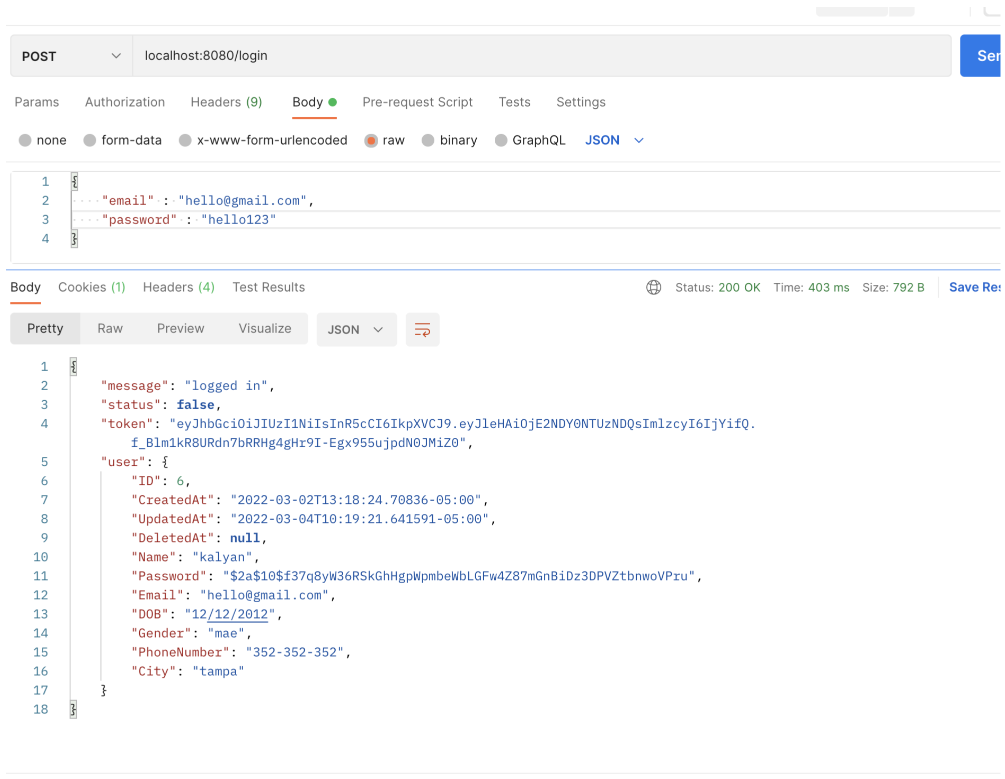
Description**:**

- Method: POST.
- This API allows the already existing users to login to the application.
- Valid credentials must be provided for a successful login.
- Logging in is mandatory for any user to use the functionalities of application like *postaride* and *searchrides*.

Acceptance Criteria:

- For registering, the following fields are required to be entered by the user: *Name, Email, Phone and Password.*
- If any/all of these fields weren't entered, the API throws an error message.
- Once all the inputs are properly entered by the user, the API throws a confirmation message: "*User added to the database*"
- If the same user tries registering again using the same credentials, the API throws the message: "*User already exists*"
- Similarly for login, the API should throw an error message, if the user entered either a wrong email or password.

Sample Requests and Responses:



localhost:8080/login

```
{
    "email" : "hello@gmail.com",
    "password" : "hello123"
}
```

Status: 200 OK Time: 403 ms Size: 792 B

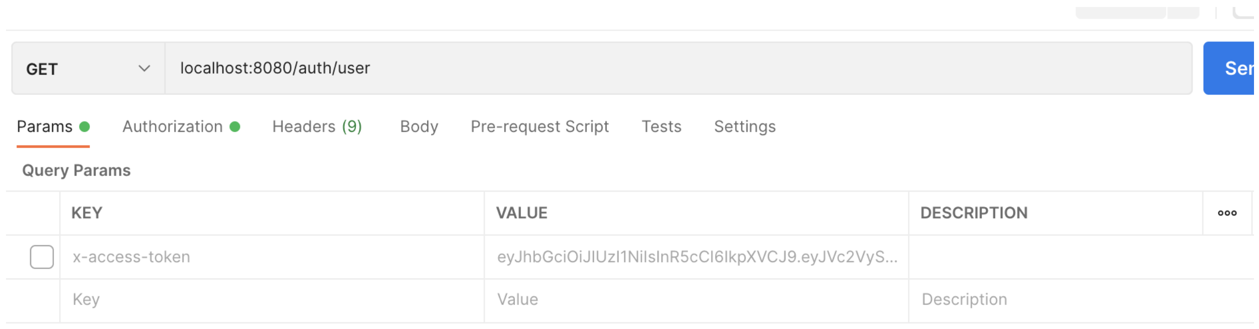
```
{
    "message": "logged in",
    "status": false,
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJleHAiOjE2NDY0NTUzNDQsImlzcyI6IjYifQ.
        f_Blm1kR8URdn7bRRHg4gHr9I-Egx955ujpdN0JMiZ0",
    "user": {
        "ID": 6,
        "CreatedAt": "2022-03-02T13:18:24.70836-05:00",
        "UpdatedAt": "2022-03-04T10:19:21.641591-05:00",
        "DeletedAt": null,
        "Name": "kalyan",
        "Password": "$2a$10$f37q8yW36RSkGhHgpWpmbeWbLGFw4Z87mGnBiDz3DPVZtbnwoVPru",
        "Email": "hello@gmail.com",
        "DOB": "12/12/2012",
        "Gender": "mae",
        "PhoneNumber": "352-352-352",
        "City": "tampa"
    }
}
```

3)user:

Description:

- Method: GET
- Once the user is logged in successfully an active JSONWebToken is generated. This API retrieves the user details such as *Name, Email, Password(encrypted format), Gender, City, DateofBirth, PhoneNumber* using the web token.

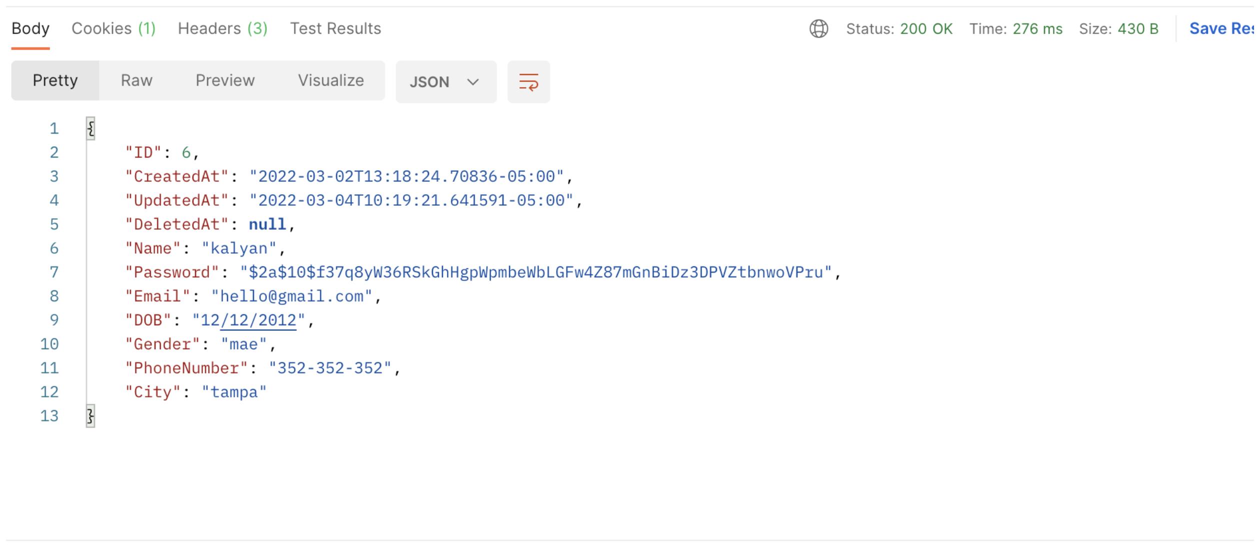
Sample Requests and Responses:



localhost:8080/auth/user

Query Params

	KEY	VALUE	DESCRIPTION
☐	x-access-token	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJVc2VyS...	
	Key	Value	Description



Status: 200 OK Time: 276 ms Size: 430 B

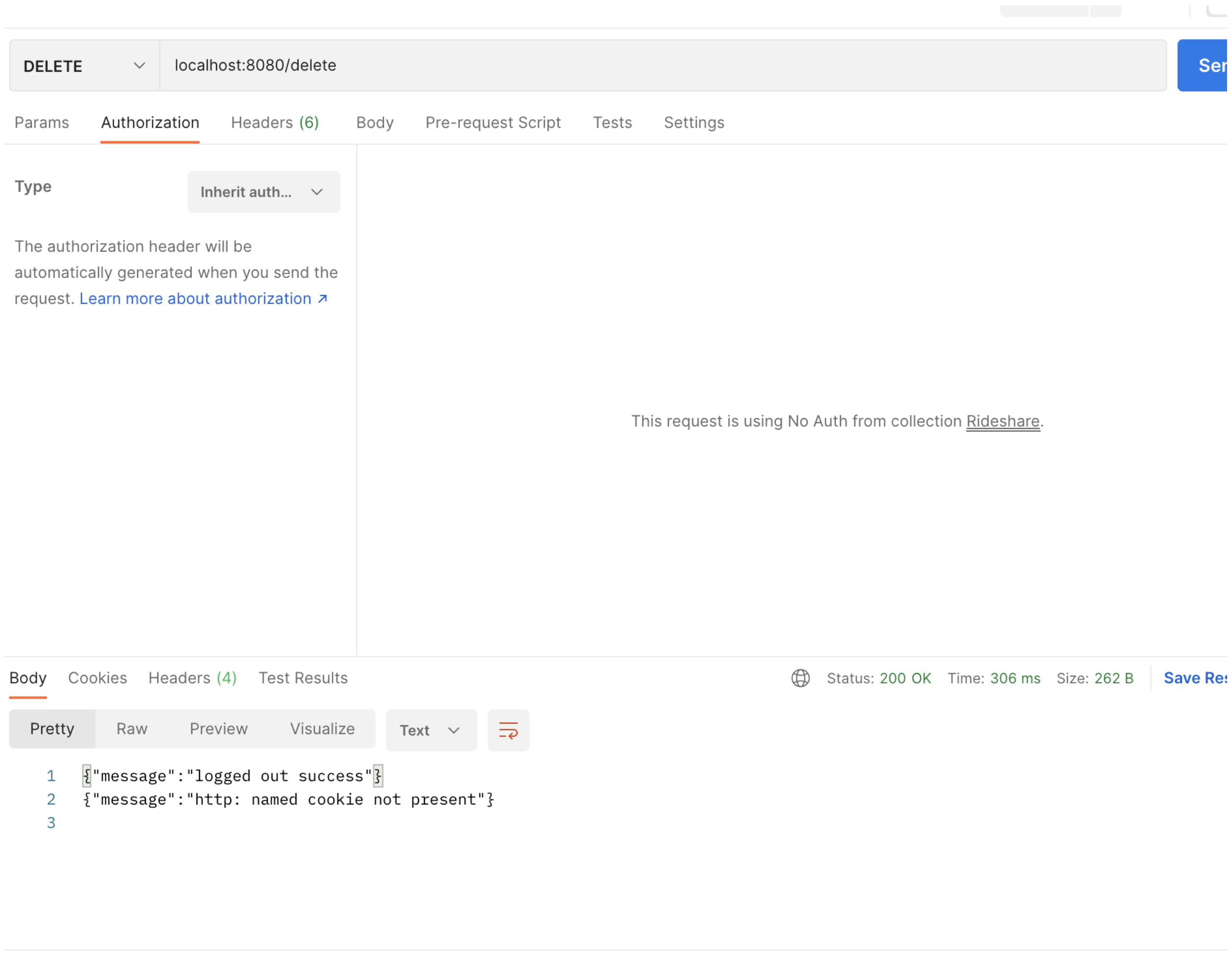
```
{
    "ID": 6,
    "CreatedAt": "2022-03-02T13:18:24.70836-05:00",
    "UpdatedAt": "2022-03-04T10:19:21.641591-05:00",
    "DeletedAt": null,
    "Name": "kalyan",
    "Password": "$2a$10$f37q8yW36RSkGhHgpWpmbeWbLGFw4Z87mGnBiDz3DPVZtbnwoVPru",
    "Email": "hello@gmail.com",
    "DOB": "12/12/2012",
    "Gender": "mae",
    "PhoneNumber": "352-352-352",
    "City": "tampa"
}
```

4)delete:

Description:

- Method: Delete
- This API is used to delete all the data of the existing user from the database records.
- This action is performed using JSONWebToken.

Sample Requests and Responses:



DELETE localhost:8080/delete

Type: Inherit auth...

The authorization header will be automatically generated when you send the request. Learn more about authorization ↗

This request is using No Auth from collection Rideshare.

Status: 200 OK Time: 306 ms Size: 262 B

Text

```
{"message":"logged out success"}
{"message":"http: named cookie not present"}
```

5)postaride:

Description:

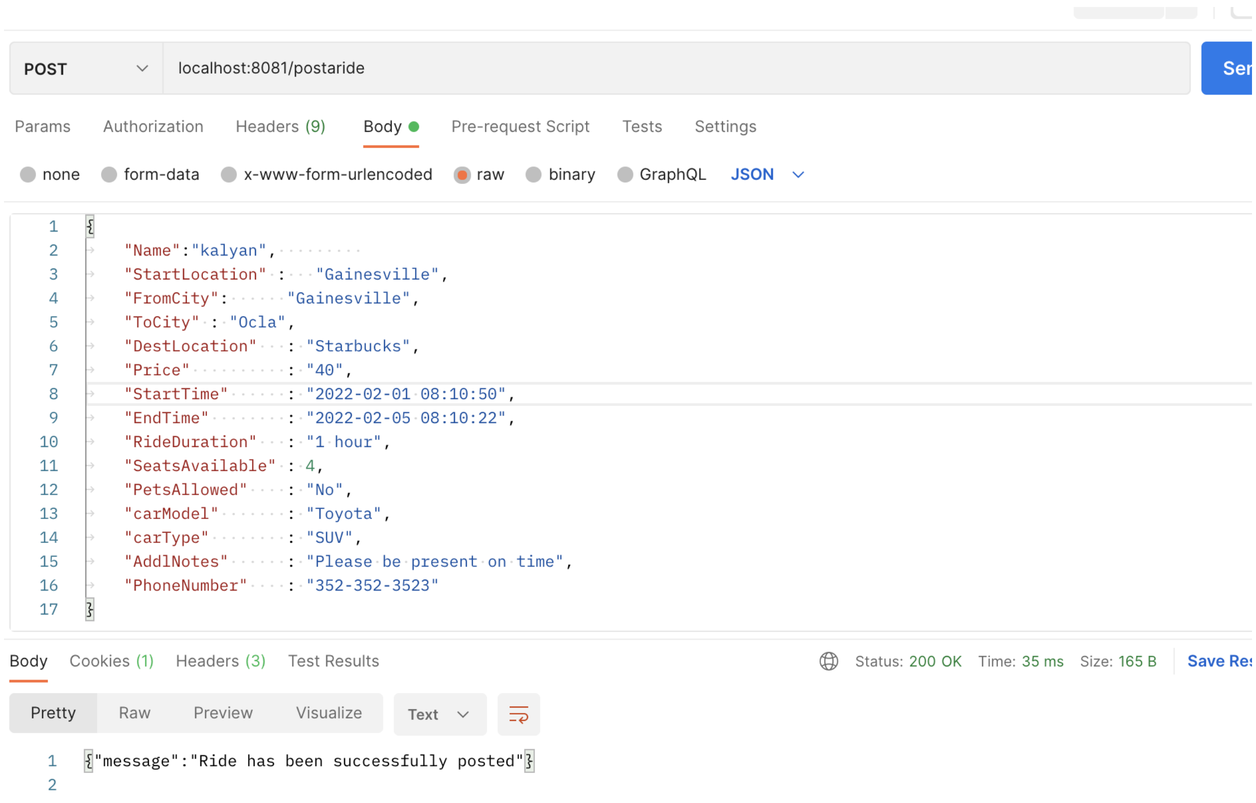
- Method: POST
- This API is used to "post a ride" details by the users.
- Users must be logged in to use this PostARide functionality.

Acceptance Criteria:

- For Posting a ride the following fields must be provided by the user: *name, startLocation, fromCity, toCity, destLocation, priceperSeat, duration, seatsAvailable, petsAllowed, carModel, carType, addlNotes, phoneNumber.*

- Once all the inputs are properly entered by the user, the API throws a confirmation message: "*Ride has been Successfully Posted.*"

Sample Requests and Responses:



localhost:8081/postaride

```
{
	"Name":"kalyan",        
	"StartLocation" :   "Gainesville",
	"FromCity":      "Gainesville",
	"ToCity" : "Ocla",
	"DestLocation"   : "Starbucks",
	"Price"          : "40",
	"StartTime"      : "2022-02-01 08:10:50",
	"EndTime"        : "2022-02-05 08:10:22",
	"RideDuration"   : "1 hour",
	"SeatsAvailable" : 4,
	"PetsAllowed"    : "No",
	"carModel"       : "Toyota",
	"carType"        : "SUV",
	"AddlNotes"      : "Please be present on time",
	"PhoneNumber"    : "352-352-3523"
}
```

Status: 200 OK Time: 35 ms Size: 165 B

```
{"message":"Ride has been successfully posted"}
```

6)**searchrides**:

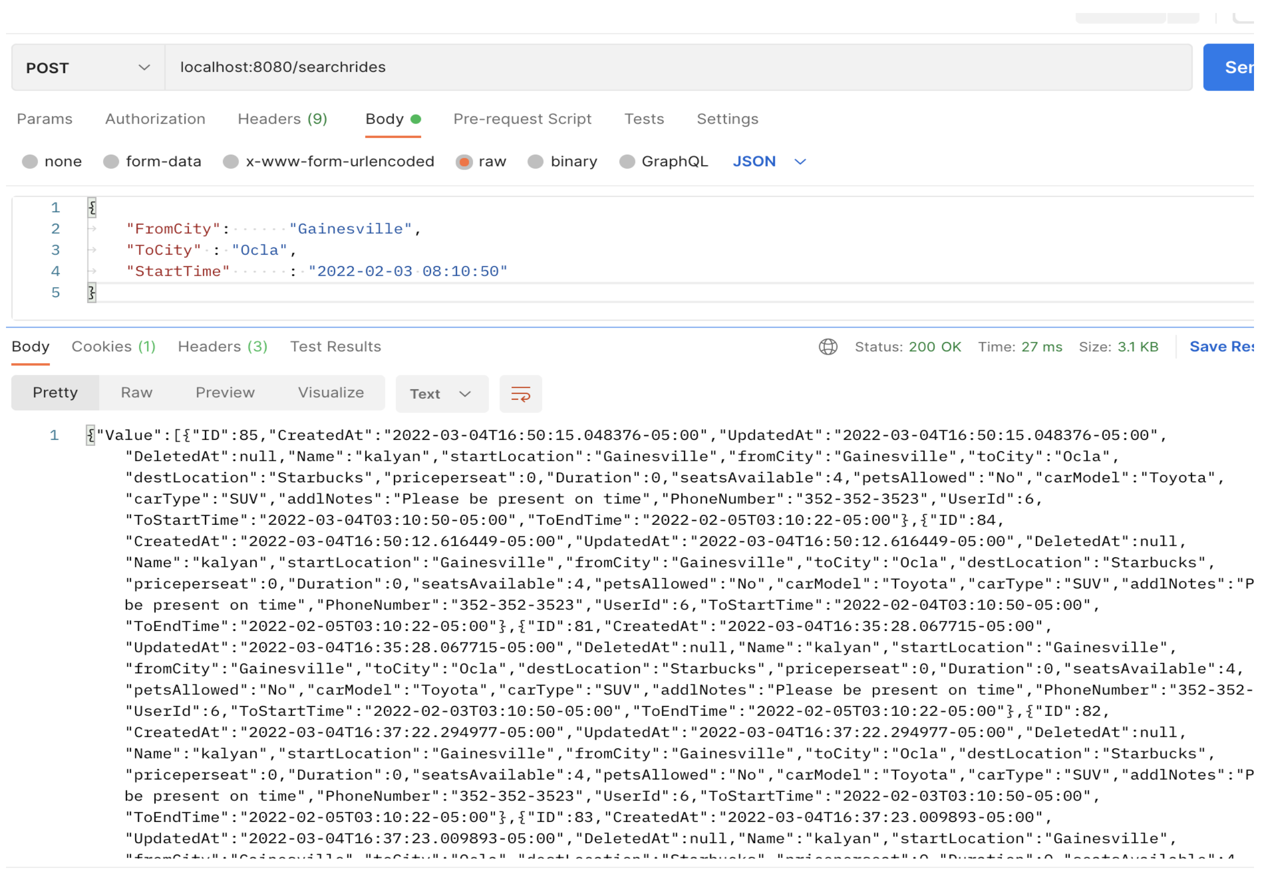
Description:

- Method: POST
- Using this API, users can get the available rides.
- To search for available rides *fromCity* , *toCity* and *startTime* must be provided.

Acceptance Criteria:

- The API should throw an error message whenever any of the above mentioned required parameters aren't specified.
- Whenever a request is sent the API should send a response with the list of available rides in the specified routes with details : *name, startLocation, fromCity, toCity, destLocation, priceperSeat, duration, seatsAvailable, petsAllowed, carModel, carType, addlNotes, phoneNumber.*

Sample Requests and Responses:



localhost:8080/searchrides

```
{
    "FromCity":      "Gainesville",
    "ToCity" : "Ocla",
    "StartTime"      : "2022-02-03 08:10:50"
}
```

Status: 200 OK   Time: 27 ms   Size: 3.1 KB

```
{"Value":[{"ID":85,"CreatedAt":"2022-03-04T16:50:15.048376-05:00","UpdatedAt":"2022-03-04T16:50:15.048376-05:00",
    "DeletedAt":null,"Name":"kalyan","startLocation":"Gainesville","fromCity":"Gainesville","toCity":"Ocla",
    "destLocation":"Starbucks","priceperseat":0,"Duration":0,"seatsAvailable":4,"petsAllowed":"No","carModel":"Toyota",
    "carType":"SUV","addlNotes":"Please be present on time","PhoneNumber":"352-352-3523","UserId":6,
    "ToStartTime":"2022-03-04T03:10:50-05:00","ToEndTime":"2022-02-05T03:10:22-05:00"},{"ID":84,
    "CreatedAt":"2022-03-04T16:50:12.616449-05:00","UpdatedAt":"2022-03-04T16:50:12.616449-05:00","DeletedAt":null,
    "Name":"kalyan","startLocation":"Gainesville","fromCity":"Gainesville","toCity":"Ocla","destLocation":"Starbucks",
    "priceperseat":0,"Duration":0,"seatsAvailable":4,"petsAllowed":"No","carModel":"Toyota","carType":"SUV","addlNotes":"P
    be present on time","PhoneNumber":"352-352-3523","UserId":6,"ToStartTime":"2022-02-04T03:10:50-05:00",
    "ToEndTime":"2022-02-05T03:10:22-05:00"},{"ID":81,"CreatedAt":"2022-03-04T16:35:28.067715-05:00",
    "UpdatedAt":"2022-03-04T16:35:28.067715-05:00","DeletedAt":null,"Name":"kalyan","startLocation":"Gainesville",
    "fromCity":"Gainesville","toCity":"Ocla","destLocation":"Starbucks","priceperseat":0,"Duration":0,"seatsAvailable":4,
    "petsAllowed":"No","carModel":"Toyota","carType":"SUV","addlNotes":"Please be present on time","PhoneNumber":"352-352-
    "UserId":6,"ToStartTime":"2022-02-03T03:10:50-05:00","ToEndTime":"2022-02-05T03:10:22-05:00"},{"ID":82,
    "CreatedAt":"2022-03-04T16:37:22.294977-05:00","UpdatedAt":"2022-03-04T16:37:22.294977-05:00","DeletedAt":null,
    "Name":"kalyan","startLocation":"Gainesville","fromCity":"Gainesville","toCity":"Ocla","destLocation":"Starbucks",
    "priceperseat":0,"Duration":0,"seatsAvailable":4,"petsAllowed":"No","carModel":"Toyota","carType":"SUV","addlNotes":"P
    be present on time","PhoneNumber":"352-352-3523","UserId":6,"ToStartTime":"2022-02-03T03:10:50-05:00",
    "ToEndTime":"2022-02-05T03:10:22-05:00"},{"ID":83,"CreatedAt":"2022-03-04T16:37:23.009893-05:00",
    "UpdatedAt":"2022-03-04T16:37:23.009893-05:00","DeletedAt":null,"Name":"kalyan","startLocation":"Gainesville",
    "fromCity":"Gainesville","toCity":"Ocla","destLocation":"Starbucks","priceperseat":0,"Duration":0,"seatsAvailable":4
```