

Name : Murakonda Shravani

Student ID : 16221927

Class ID : 34

Course:ASE

Lab No : 6

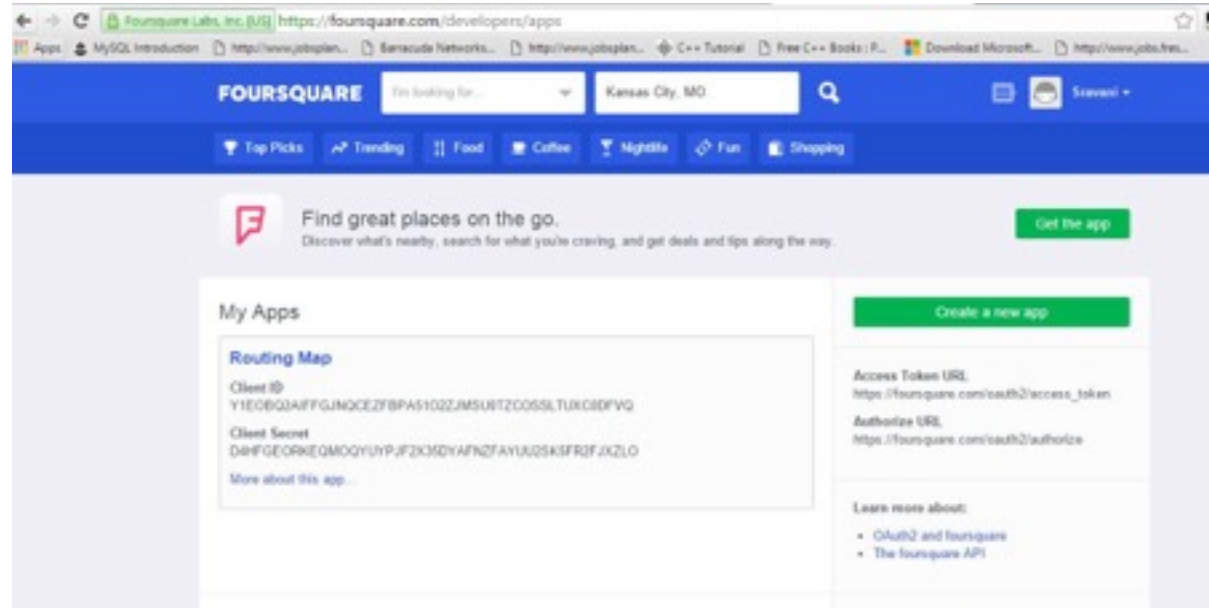
Date : 2nd Mar 2016

About Application

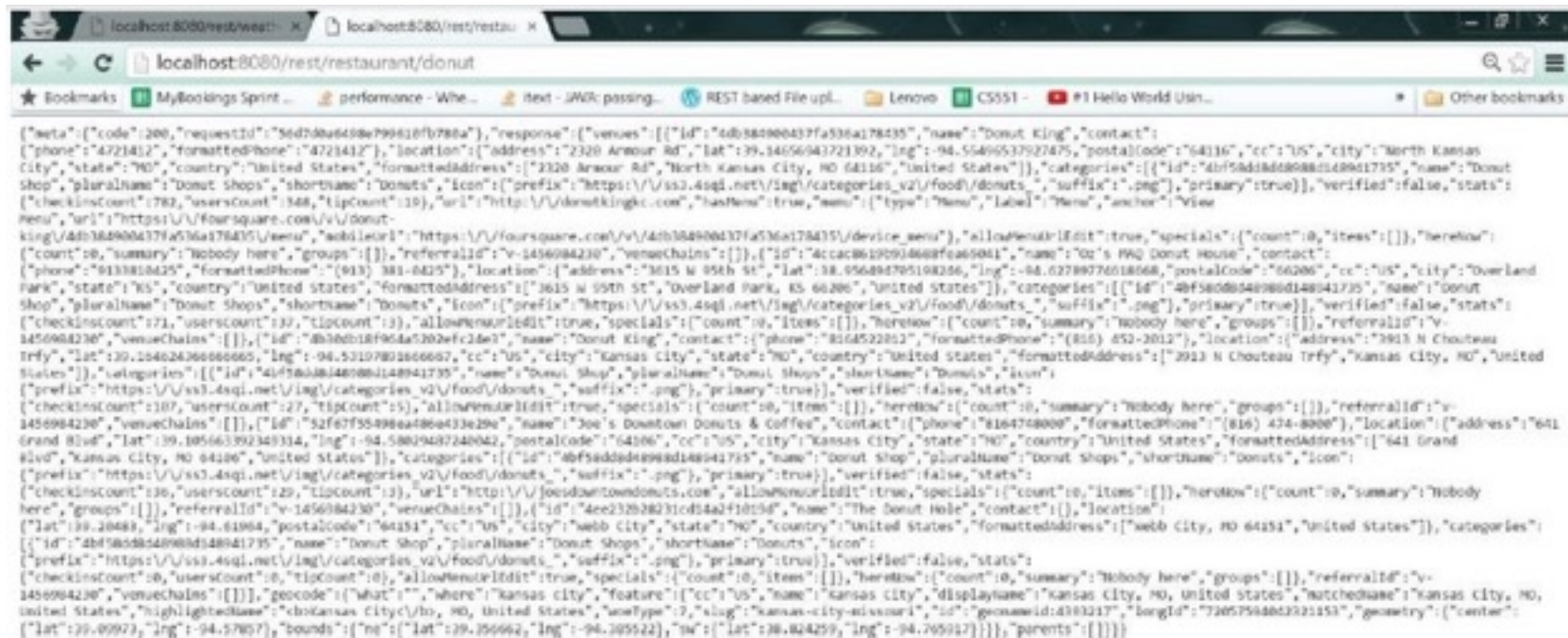
In this project we provide details about restaurants and weather in a given location using Foursquare API. Main aim is integrating these two services in a single servlet using REST(Representational State Transfer) services. And also for the developed services we have used Junit testing to know the working result of the application. For this we used the tools Eclipse and Apache Tomcat Server.

Application Steps :

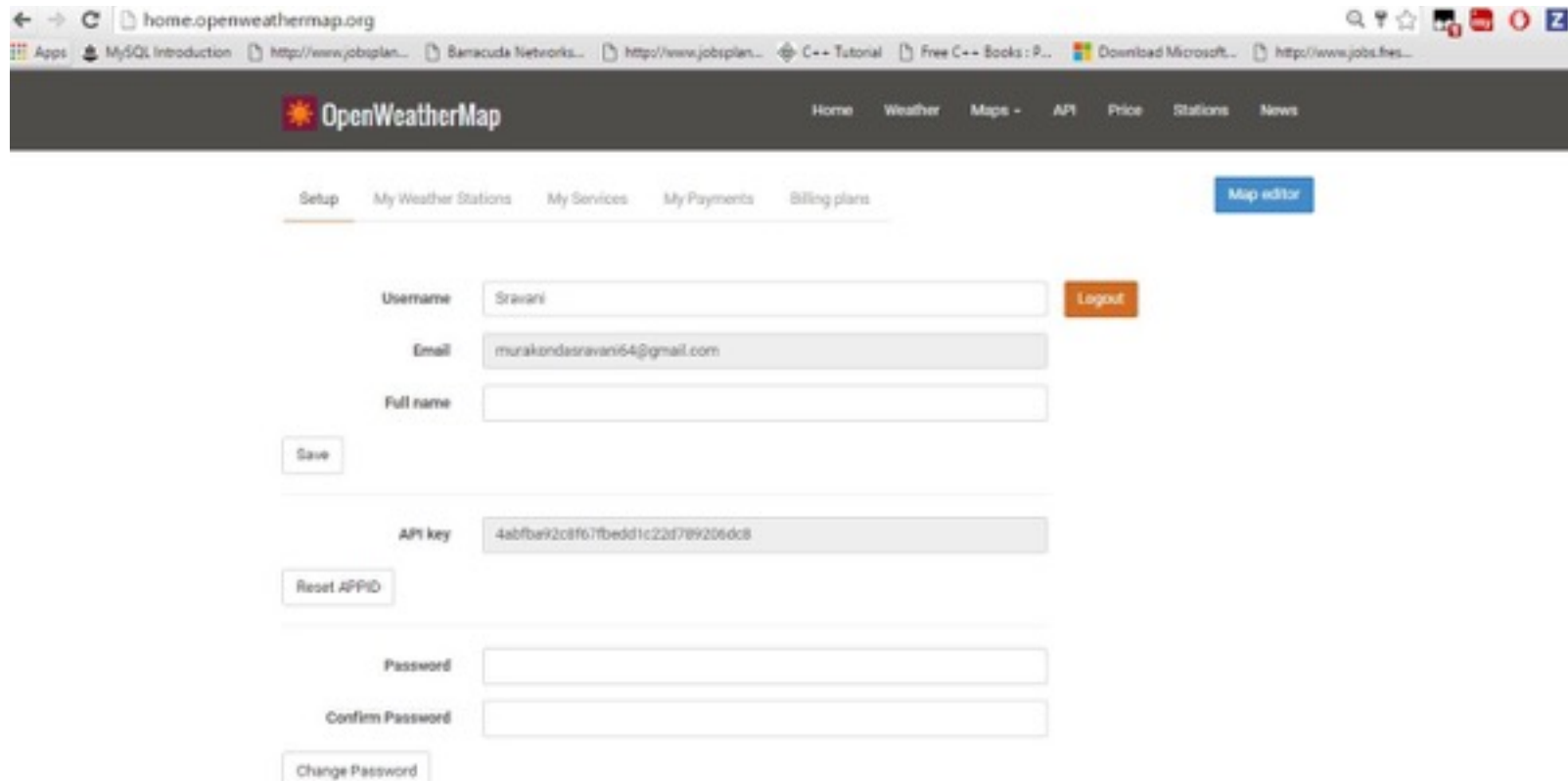
Step : 1 Foursquare ID creation : To use Foursquare API services we initially create a client ID and client secret by creating an account in the website to get the information for restaurants.



Step : 2 Foursquare Details about Restaurant: In this we get all the details related to restaurants from Foursquare API in the following format.



Step : 3 OpenWeather API ID creation : To use OpenWeather API services we initially create a API key by creating an account in the website to get the information for weather.



The screenshot shows the OpenWeatherMap website's account creation interface. The browser's address bar displays 'home.openweathermap.org'. The website's header includes the OpenWeatherMap logo and navigation links: Home, Weather, Maps, API, Price, Stations, and News. Below the header, a secondary navigation bar contains links for Setup, My Weather Stations, My Services, My Payments, and Billing plans, along with a 'Map editor' button. The main content area is a form for creating an account. It includes input fields for Username (containing 'Sravani'), Email (containing 'murakondesravani54@gmail.com'), and Full name. A 'Logout' button is positioned next to the Username field. Below these fields is a 'Save' button. Further down, there is an input field for the API key (containing '4abfba92c8f67fbedd1c22df789206dc8') and a 'Reset APIID' button. At the bottom of the form, there are input fields for Password and Confirm Password, followed by a 'Change Password' button.

home.openweathermap.org

Apps MySQL Introduction http://www.jobspan... Samacuda Networks... http://www.jobspan... C++ Tutorial Free C++ Books : P... Download Microsoft... http://www.jobspan...

OpenWeatherMap Home Weather Maps API Price Stations News

Setup My Weather Stations My Services My Payments Billing plans Map editor

Username Sravani Logout

Email murakondesravani54@gmail.com

Full name

Save

API key 4abfba92c8f67fbedd1c22df789206dc8

Reset APIID

Password

Confirm Password

Change Password

Step : 4 OpenWeather Details about weather: In this we get all the details related to restaurants from Foursquare API in the following format.



The screenshot shows a web browser window with a REST client interface. The address bar displays `localhost:8080/rest/weather/london`. The browser's bookmark bar includes entries like "MyBookings Sprint...", "performance - Whe...", "REST - JAVAc passing...", "REST based File upL...", "Lenovo", "CS551 -", and "#1 Hello World Usin...". The main content area displays a JSON response from the OpenWeather API for London:

```
{
  "coord": {
    "lon": -0.13,
    "lat": 51.51
  },
  "weather": [
    {
      "id": 500,
      "main": "Rain",
      "description": "light rain",
      "icon": "10n"
    }
  ],
  "base": "stations",
  "main": {
    "temp": 277.48,
    "pressure": 1009,
    "humidity": 75,
    "temp_min": 276.15,
    "temp_max": 278.55,
    "visibility": 10000,
    "wind": {
      "speed": 8.2,
      "deg": 89,
      "gust": 13.9
    },
    "rain": {
      "1h": 0.3
    },
    "clouds": {
      "all": 92
    },
    "dt": 1456984665,
    "sys": {
      "type": 1,
      "id": 5091,
      "message": 0.0431,
      "country": "GB",
      "sunrise": 1456987202,
      "sunset": 1457027126
    },
    "id": 2643743,
    "name": "London",
    "cod": 200
  }
}
```

Step : 5 Junit Test : After developing the application for each service done the Junit testing. For each section that is weatherTest() and restaurantTest().

```
10 public class ServiceTest {
11
12     @Test
13     public void weatherTest() throws Exception{
14         WeatherService weather = new WeatherService();
15         Response s = weather.getWeather("london");
16         System.out.println("The Weather Service is executed successfully-----"+Response.status(200).entity());
17     }
18
19     @Test
20     public void restaurantTest() throws Exception{
21         RestaurantService restaurantService = new RestaurantService();
22         Response resp = restaurantService.getRestaurants("donut");
23         System.out.println("The Restaurant Service is executed successfully-----"+Response.status(200).entity());
24     }
25 }
26
```

Markers Properties Servers Data Source Explorer Snippets Problems Console JUnit
terminated> ServiceTest [JUnit] C:\Program Files\Java\jdk1.7.0_02\jre\bin\javaw.exe (Mar 2, 2016, 11:48:51 PM)

The Restaurant Service is executed successfully-----org.apache.cxf.jaxrs.impl.ResponseImpl@76f90793
The Weather Service is executed successfully-----org.apache.cxf.jaxrs.impl.ResponseImpl@35282bf2