CSCI 5409: ASSIGNMENT 4

Workflows and Workflow Management

Mr. Niravsinh Jadeja
Nirav.jadeja@dal.ca

Contents

1. WEB URL 2
a. REST Web service 2
b. Client 2
c. Web service Log 2
2. Environment/Resources used 4
a. REST Web Service (Heroku Cloud) 5
b. Client app (on Bluenose DAL Server) 6
c. Logic app Workflow 8
3. Code Reference 9
4. Testing 11
a. Wrong input in Insurance, Mortgage, Employee Details 11
b. Wrong User Authentication 14
5. Normal Operation 16
a. Create Operation 16
b. Authenticate Operation 19
6. References 21

1. WEB URL

a. REST Web service

Mortgage:

URL: https://a4-mortgage.herokuapp.com/mortgage/

Insurance:

URL: https://a4-insurance.herokuapp.com/insurance/

Employee:

URL: https://a4-employer.herokuapp.com/employees/

b. Client

Mortgage

URL: https://web.cs.dal.ca/~jadeja/csci5409/a4/createMortgage.php

Insurance:

URL: https://web.cs.dal.ca/~jadeja/csci5409/a4/createInsurance.php

Employee:

URL: https://web.cs.dal.ca/~jadeja/csci5409/a4/createEmployee.php

c. Web service Log

Download Heroku CLI tool to your machine via following link

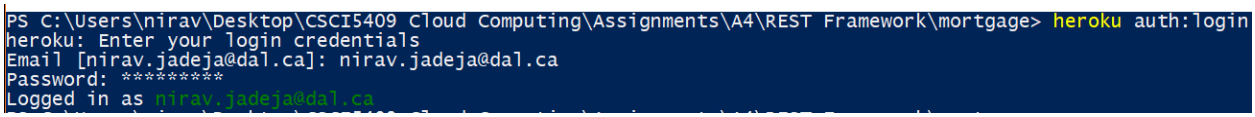
https://devcenter.heroku.com/articles/heroku-cli

After complete installation, open terminal or CMD in your machine in the project directory (Web Service).

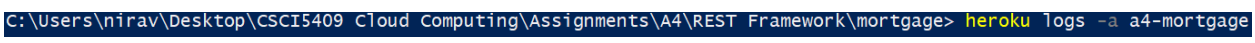
Login ID: nirav.jadeja@dal.ca

Password: Nirav@261

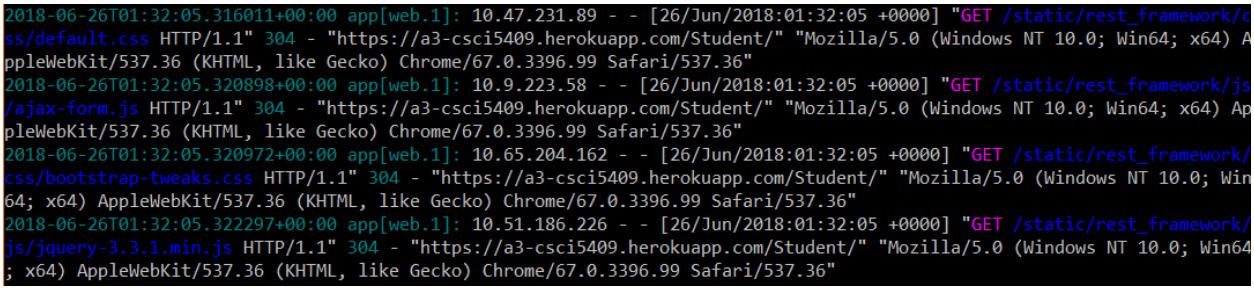
Now type following command "**heroku auth:login**"



Once you logged in, type following command "**heroku logs -a a3-csci5409**" for logs



Response (Logs):



The reason for providing this procedure for logs is that There wasn't other option to write logs, as Heroku wasn't writing any logs on bluenose server. Here, Logs shows that GET request with different time stamps.

2. Environment/Resources used

Hardware/Software Used: Acer Nitro5 with Windows 10, Atom (Editor), Chrome (Testing), FileZilla Desktop Client (For making connection with server), Bluenose Dalhousie Server (For Client Hosting), Heroku Cloud (For REST Web Service), Python-Django (For REST Web Service), PHP, HTML (For Client app), Heroku CLI (For web service testing and Logs), DB (SQLite3 – Django Inbuilt Engine).

Here, I have created 3 separate DB for each web service. For Mortgage web service, DB name is mortgage and schema are as follows.

Table Name: mortgage
Primary Key: mortgage_ID
mortagage_Name mortgage_Address mortgage_City mortgage_Province mortgage_Occupation mortgage_PhoneNumber mortgage_CompanyName mortgage_InsuranceName

Mortgage Web Service Table

For Insurance web service, DB name is insurance and schema are as follows.

Table Name: insurance
Primary Key: insurance ID
insurance_Name insurance_Value

Insurance Web Service Schema

For Employee web service, DB name is employee and schema are as follows.

Table Name: employee
Primary Key: employee_ID
employee _Name employee _Salary employee _Occupation

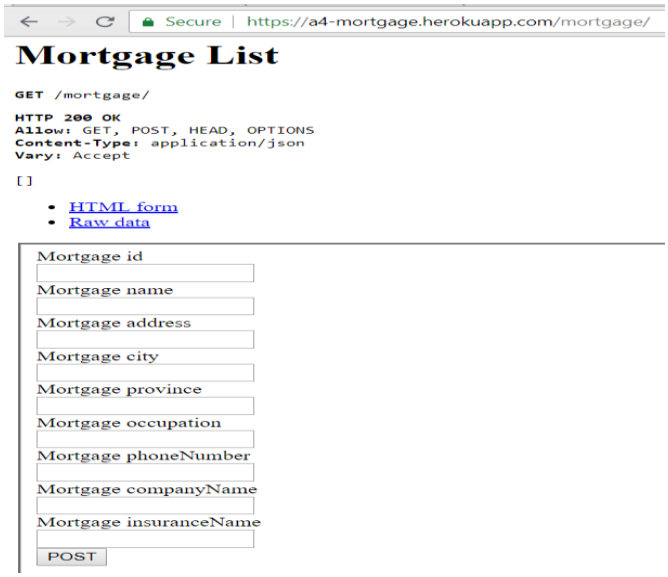
Employee Web Service schema

Further Mortgage ID will be of 3 digits and further validation have applied on client side. For insurance side, insurance ID will be of 5 digits and for employee side, employee id will be of 3 digits. The sample inputs are also provided along with input text.

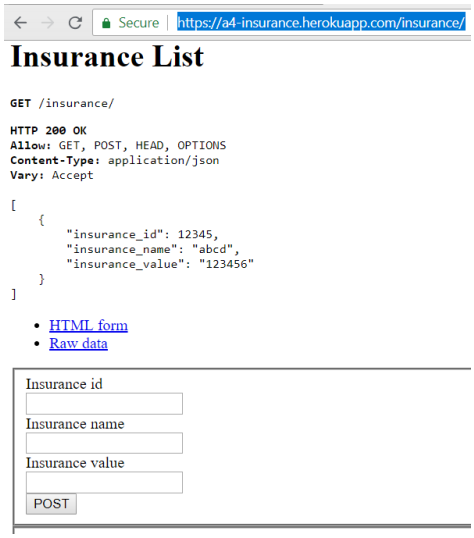
For Client-side application, I have built in PHP and HTML. For implementing REST requests, I have used CURL library with JSON encoded/decoded data. In addition, communication between client app and web service is taking place over HTTPS.

a. REST Web Service (Heroku Cloud)

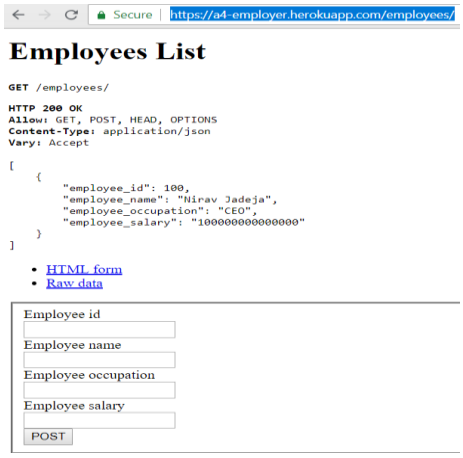
Mortgage



Insurance

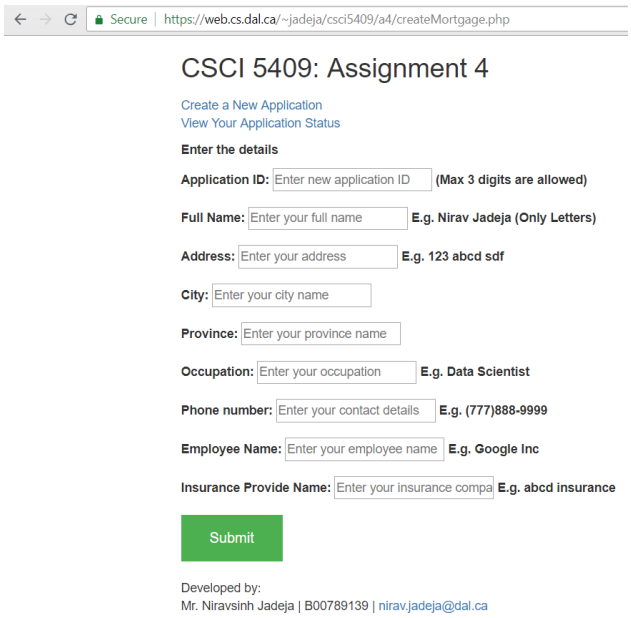


Employee



b. Client app (on Bluenose DAL Server)

Mortgage



Create New Application

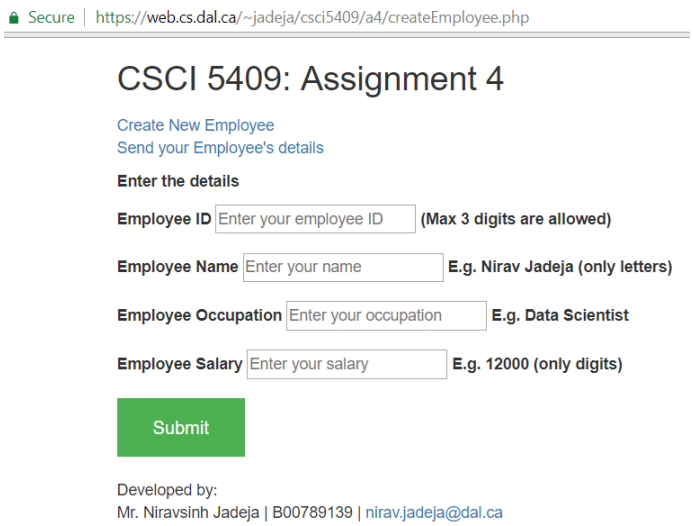
Authenticate user

Insurance

Create a new insurance policy

Authenticate insurance ID

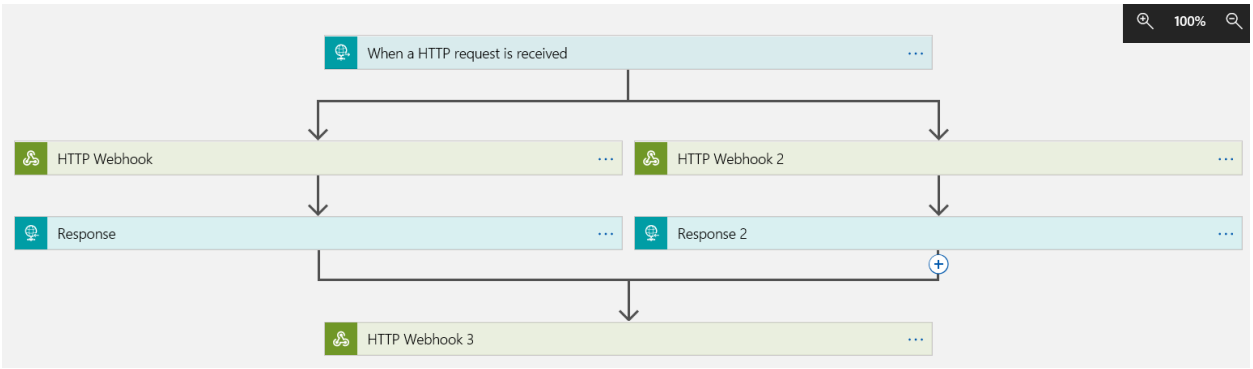
Employee



Create a new Employee ID

Authenticate an employee ID

c. Logic app Workflow



3. Code Reference

Reference:

```
.button {
    background-color: #4CAF50;
    border: none;
    color: white;
    padding: 15px 32px;
    text-align: center;
    text-decoration: none;
    display: inline-block;
    font-size: 16px;
}
```

My Code:

```
background-color: #4CAF50; /* Green */
border: none;
color: white;
padding: 15px 32px;
text-align: center;
text-decoration: none;
display: inline-block;
font-size: 16px;">
```

Reference:

```
$json = json_decode($json_data);

echo $json->profile->displayName;
```

My Code:

```
$obj = json_decode($result);
```

```
echo "<td>".$obj->student_id."</td>";
```

Reference:

```
curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS,$data_json);
```

My Code:

```
// 1 for post request
curl_setopt($ch, CURLOPT_POST, 1);

// encoded JSON data attached with curl post fields
curl_setopt($ch, CURLOPT_POSTFIELDS, $jsonDataEncoded);

//declaring application type as JSON.
curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));

//Execute the request
$result = curl_exec($ch);
```

Reference:

```
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_CUSTOMREQUEST, "DELETE");
curl_setopt($ch, CURLOPT_POSTFIELDS,$data_json);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$response  = curl_exec($ch);
curl_close($ch);
```

My Code:

```
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'DELETE');
curl_setopt($ch, CURLOPT_POSTFIELDS);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$response  = curl_exec($ch);
curl_close($ch);
```

Reference:

```
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);
curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS,$data_json);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$response  = curl_exec($ch);
curl_close($ch);
```

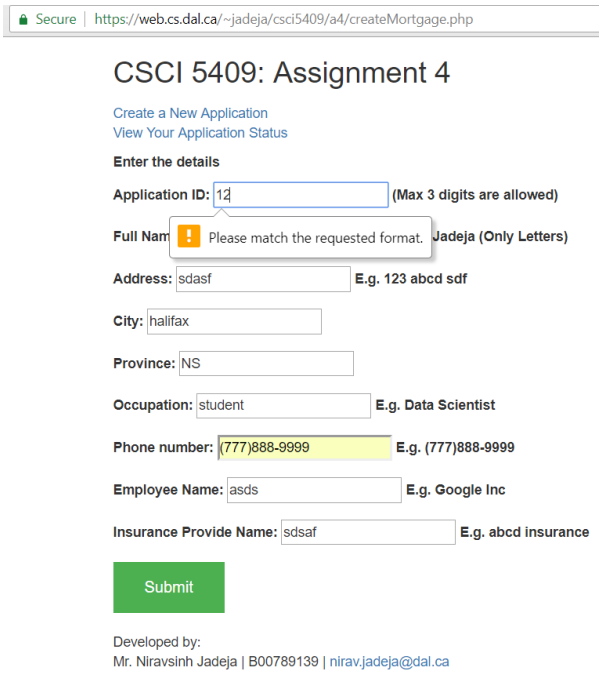
My Code:

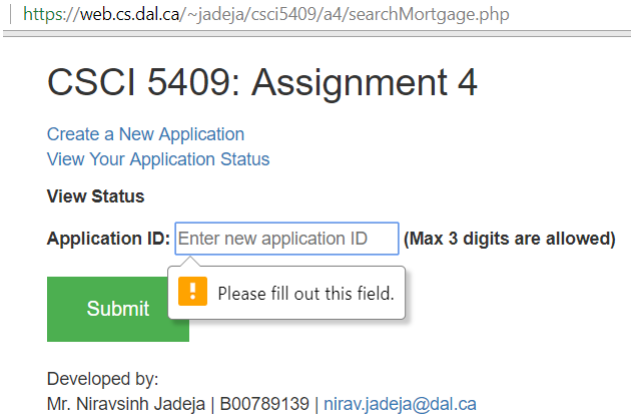
```
// curl intialization
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $url);

// content type JSON
curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));

// mode of request
curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'PATCH');

// attaching data with headers
curl_setopt($ch, CURLOPT_POSTFIELDS,$data_json);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);

// excute command
$response  = curl_exec($ch);
curl_close($ch);
```

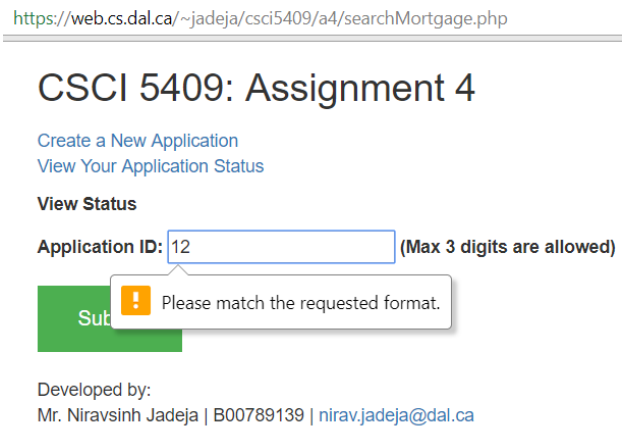
4. Testing

a. Wrong input in Insurance, Mortgage, Employee Details

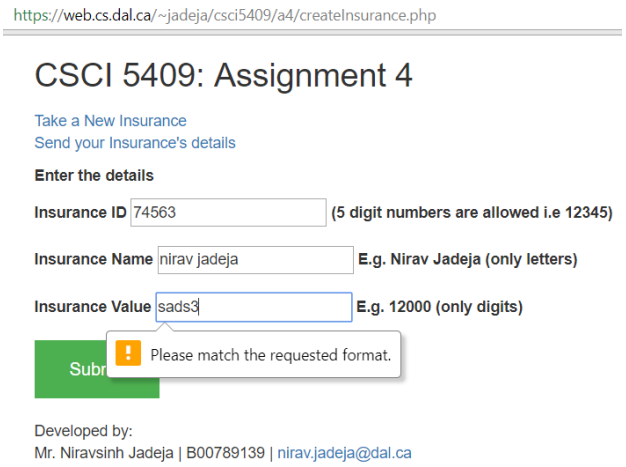
Mortgage:

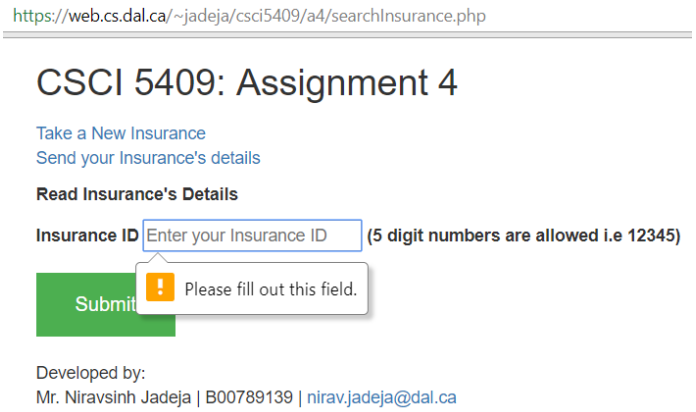


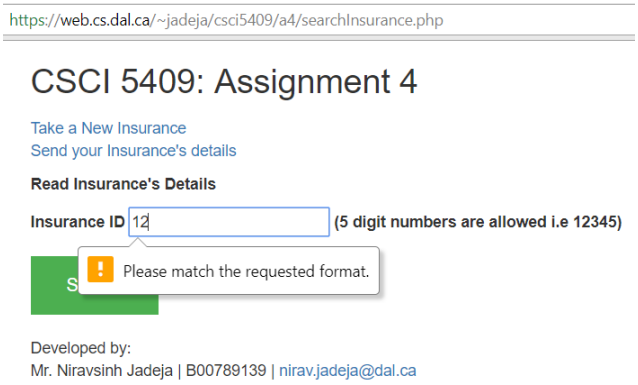



https://web.cs.dal.ca/~jadeja/csci5409/a4/searchMortgage.php
CSCI 5409: Assignment 4
Create a New Application
View Your Application Status
View Status
Application ID: 12 (Max 3 digits are allowed)
Please match the requested format.
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

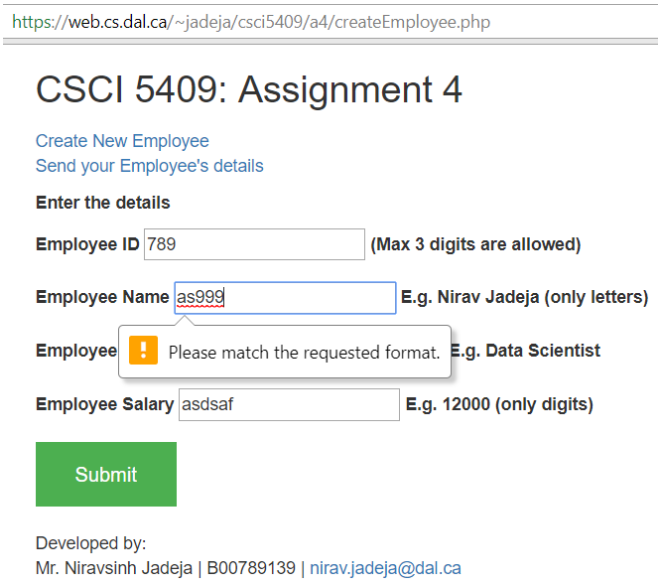
Insurance:


https://web.cs.dal.ca/~jadeja/csci5409/a4/createInsurance.php
CSCI 5409: Assignment 4
Take a New Insurance
Send your Insurance's details
Enter the details
Insurance ID 74563 (5 digit numbers are allowed i.e 12345)
Insurance Name nirav jadeja E.g. Nirav Jadeja (only letters)
Insurance Value sads3 E.g. 12000 (only digits)
Please match the requested format.
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca


https://web.cs.dal.ca/~jadeja/csci5409/a4/searchInsurance.php
CSCI 5409: Assignment 4
Take a New Insurance
Send your Insurance's details
Read Insurance's Details
Insurance ID Enter your Insurance ID (5 digit numbers are allowed i.e 12345)
Please fill out this field.
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca


https://web.cs.dal.ca/~jadeja/csci5409/a4/searchInsurance.php
CSCI 5409: Assignment 4
Take a New Insurance
Send your Insurance's details
Read Insurance's Details
Insurance ID 12
(5 digit numbers are allowed i.e 12345)
Please match the requested format.
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

Employee:


https://web.cs.dal.ca/~jadeja/csci5409/a4/createEmployee.php
CSCI 5409: Assignment 4
Create New Employee
Send your Employee's details
Enter the details
Employee ID 789
(Max 3 digits are allowed)
Employee Name as999
E.g. Nirav Jadeja (only letters)
Employee
Please match the requested format.
E.g. Data Scientist
Employee Salary asdsaf
E.g. 12000 (only digits)
Submit
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

https://web.cs.dal.ca/~jadeja/csci5409/a4/searchEmployee.php
CSCI 5409: Assignment 4
Create New Employee
Send your Employee's details
Read Employee Details
Employee ID Enter your employee ID
(Max 3 digits are allowed)
Please fill out this field.
Submit
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

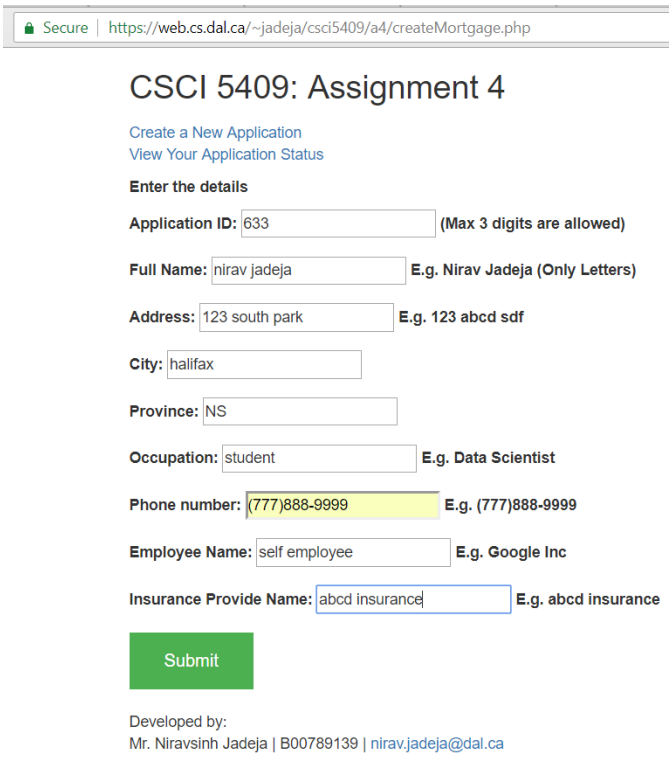
b. Wrong User Authentication

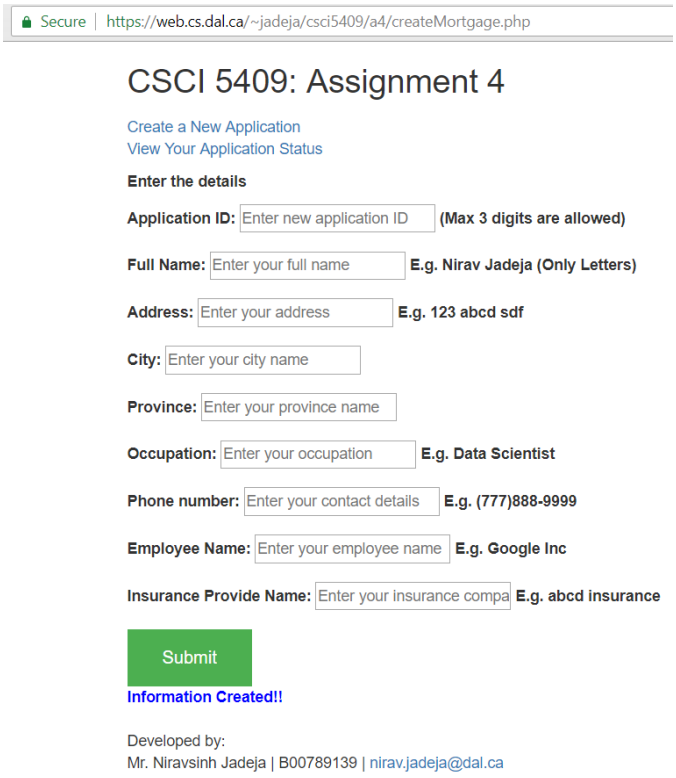
Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/searchMortgage.php
CSCI 5409: Assignment 4
Create a New Application
View Your Application Status
View Status
Application ID: Enter new application ID (Max 3 digits are allowed)
Submit
Record not found, please try again.
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

5. Normal Operation

a. Create Operation

Mortgage:




Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/createMortgage.php
CSCI 5409: Assignment 4
Create a New Application
View Your Application Status
Enter the details
Application ID: Enter new application ID (Max 3 digits are allowed)
Full Name: Enter your full name E.g. Nirav Jadeja (Only Letters)
Address: Enter your address E.g. 123 abcd sdf
City: Enter your city name
Province: Enter your province name
Occupation: Enter your occupation E.g. Data Scientist
Phone number: Enter your contact details E.g. (777)888-9999
Employee Name: Enter your employee name E.g. Google Inc
Insurance Provide Name: Enter your insurance compa E.g. abcd insurance
Submit
Information Created!!
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

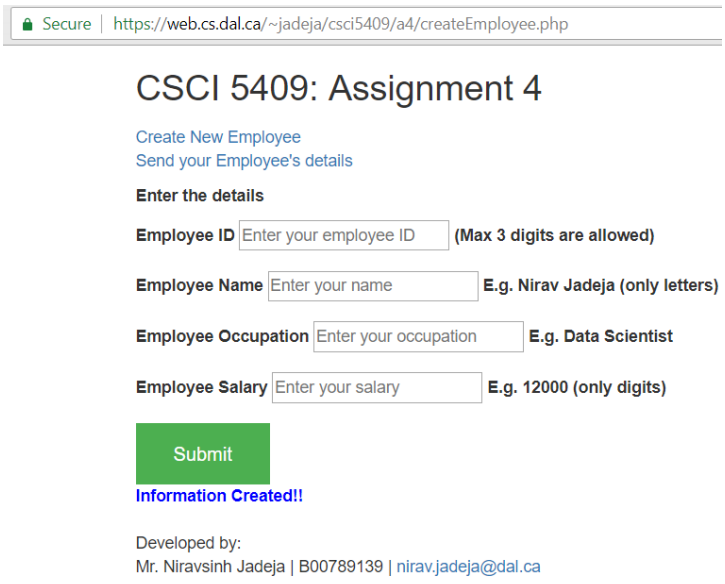
Insurance:

Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/createInsurance.php
CSCI 5409: Assignment 4
Take a New Insurance
Send your Insurance's details
Enter the details
Insurance ID 12312 (5 digit numbers are allowed i.e 12345)
Insurance Name nirav jadeja E.g. Nirav Jadeja (only letters)
Insurance Value 50000 E.g. 12000 (only digits)
Submit
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/createInsurance.php
CSCI 5409: Assignment 4
Take a New Insurance
Send your Insurance's details
Enter the details
Insurance ID Enter your Insurance ID (5 digit numbers are allowed i.e 12345)
Insurance Name Enter your name E.g. Nirav Jadeja (only letters)
Insurance Value Enter your occupation E.g. 12000 (only digits)
Submit
Information Created!!
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

Employee:

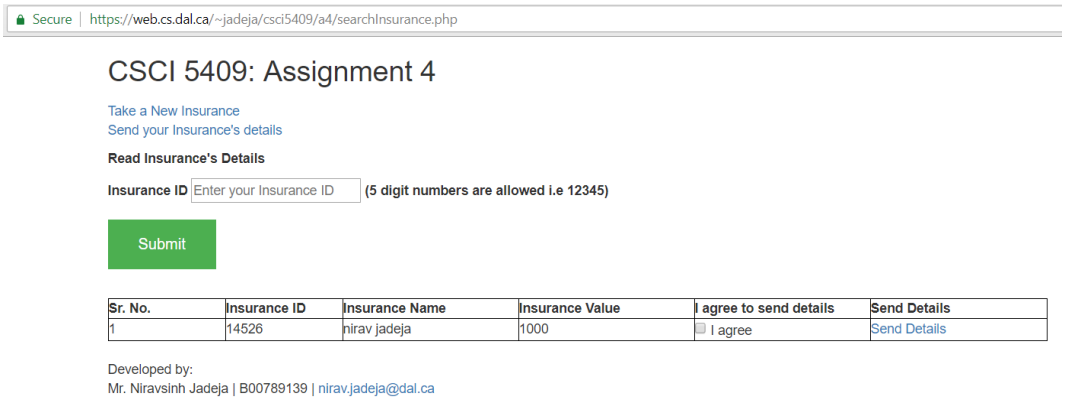
Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/createEmployee.php
CSCI 5409: Assignment 4
Create New Employee
Send your Employee's details
Enter the details
Employee ID 455 (Max 3 digits are allowed)
Employee Name nirav jadeja E.g. Nirav Jadeja (only letters)
Employee Occupation student E.g. Data Scientist
Employee Salary 12000 E.g. 12000 (only digits)
Submit
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca


Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/createEmployee.php
CSCI 5409: Assignment 4
Create New Employee
Send your Employee's details
Enter the details
Employee ID Enter your employee ID (Max 3 digits are allowed)
Employee Name Enter your name E.g. Nirav Jadeja (only letters)
Employee Occupation Enter your occupation E.g. Data Scientist
Employee Salary Enter your salary E.g. 12000 (only digits)
Submit
Information Created!!
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

b. Authenticate Operation

Mortgage:

Insurance:


Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/searchInsurance.php
CSCI 5409: Assignment 4
Take a New Insurance
Send your Insurance's details
Read Insurance's Details
Insurance ID Enter your Insurance ID (5 digit numbers are allowed i.e 12345)
Submit
Sr. No. | Insurance ID | Insurance Name | Insurance Value | I agree to send details | Send Details
1 | 14526 | nirav jadeja | 1000 | I agree | Send Details
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

Employee:

Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/searchEmployee.php
CSCI 5409: Assignment 4
Create New Employee
Send your Employee's details
Read Employee Details
Employee ID Enter your employee ID (Max 3 digits are allowed)
Submit
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

Secure | https://web.cs.dal.ca/~jadeja/csci5409/a4/searchEmployee.php
CSCI 5409: Assignment 4
Create New Employee
Send your Employee's details
Read Employee Details
Employee ID Enter your employee ID (Max 3 digits are allowed)
Submit
Sr. No. | Employee ID | Employee Name | Employee Occupation | Employee Salary | I agree to send details | Send Details
1 | 455 | nirav jadeja | student | 12000 | I agree | Send Details
Developed by:
Mr. Niravsinh Jadeja | B00789139 | nirav.jadeja@dal.ca

6. References

1. CSCI 5409: Advance Cloud Computing Assignment 3
2. https://www.youtube.com/watch?v=263xt_4mBNc
3. https://www.w3schools.com/css/css3_buttons.asp
4. https://stackoverflow.com/questions/17865544/read-json-data-response-using-php
5. https://www.w3schools.com/bootstrap/bootstrap_forms.asp
6. https://stackoverflow.com/questions/29373887/django-db-utils-integrityerror-unique-constraint-failed-rango-category-new-sl
7. https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-http-endpoint
8. https://docs.microsoft.com/en-us/azure/connectors/connectors-native-http