

Documentation for CSC Task

Github Link: <https://github.com/SHOCKBEAST/CSCAssignment1V2>

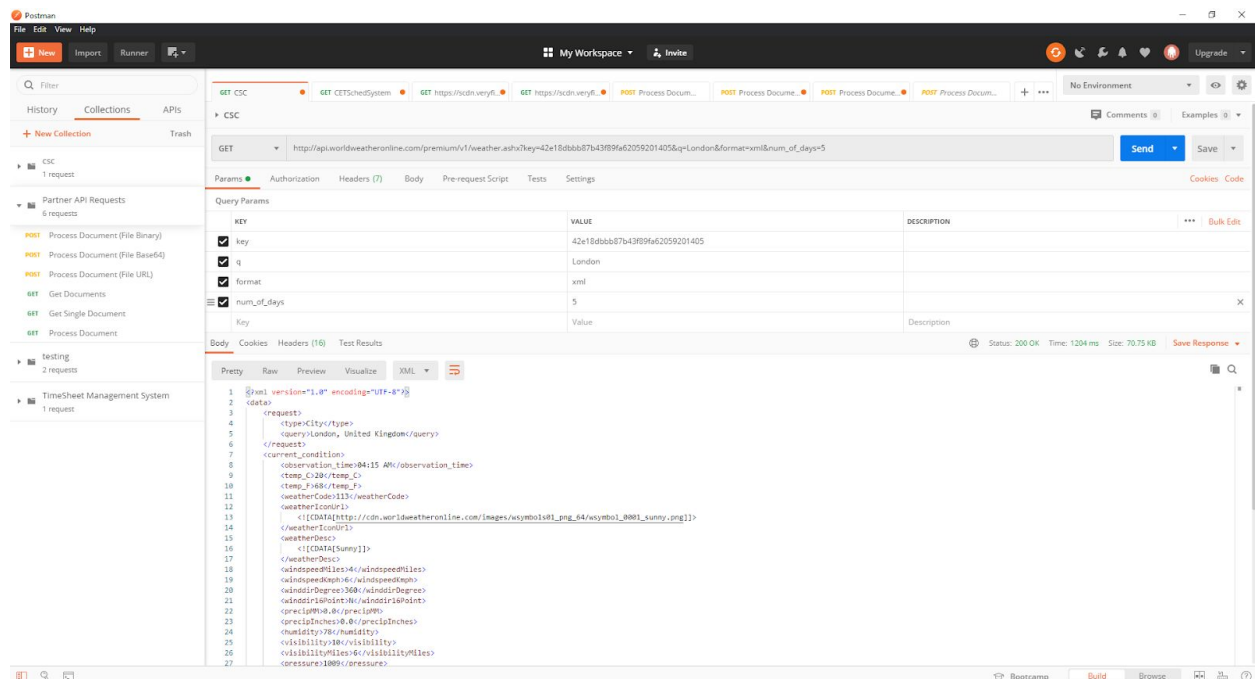
Done by : P1851830- Edwin Low :Task 3,4,5 Additional Feature: Dialog Flow Text-Speech :

Done by : P1234567- Amos Task 1,2,8

Done by : P1845192- Sui Sheng Task 6,7

Task 1

Test Query (London)



Actual Query (Malaysia, Kuala Lumpur)

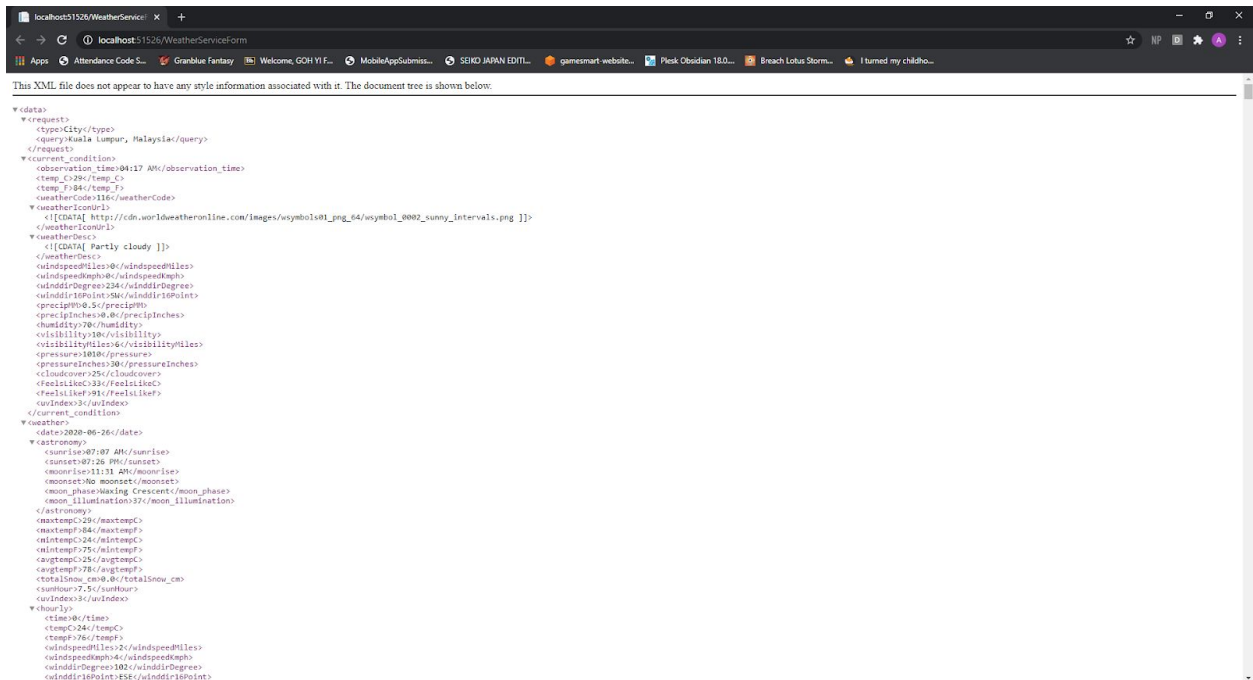


Diagram :



Web Service

User → JSON/HTTP → Internal System

Task 2

Web API document:

ProductV2Controller

GET /api/v1/Products/: var data:

"Id".",

```
"Name": "",
"Category": "",
"Price": ""
```

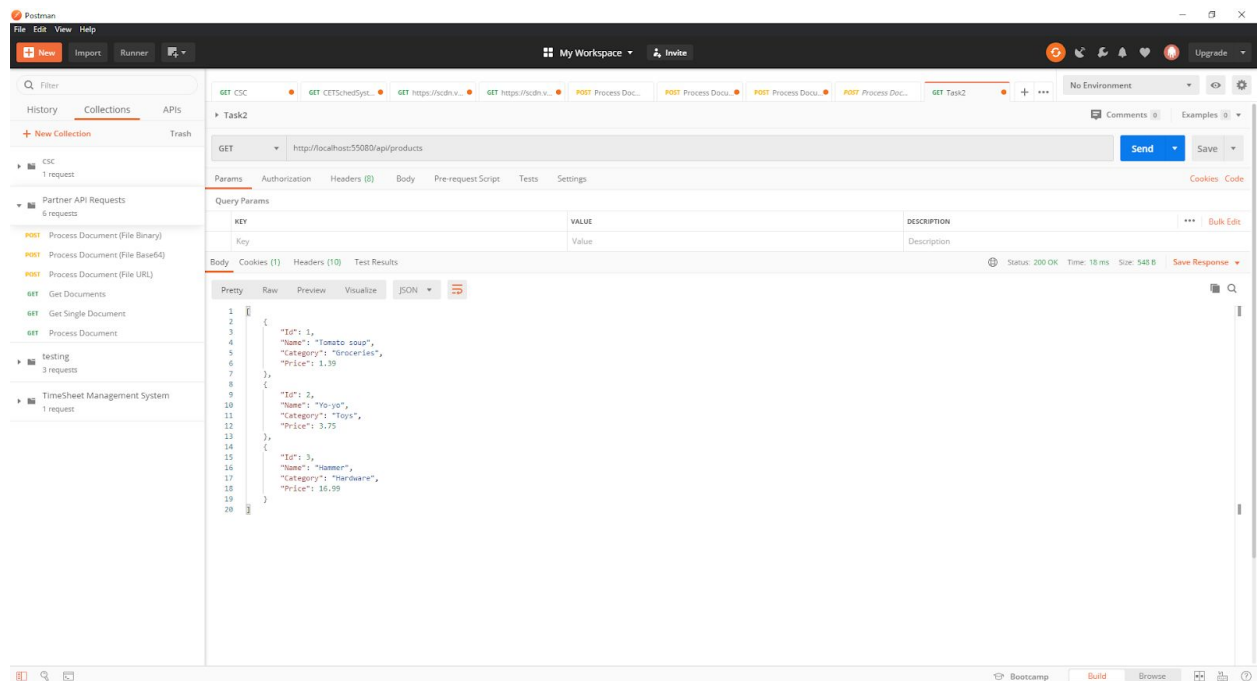
POST /api/V3/products/: var data:

```
"Item": "",
"Category": "",
"Price": ""
```

DELETE /api/v3/products/: var data:

```
"id": ""
```

Postman testing screenshot:



Postman interface showing a GET request to `http://localhost:55080/api/products/1`. The response status is 200 OK, Time: 16 ms, Size: 429 B. The response body is JSON:

```
{
  "id": 1,
  "name": "Tomato soup",
  "category": "Groceries",
  "price": 1.39
}
```

Postman interface showing a POST request to `http://localhost:55080/api/products`. The response status is 201 Created, Time: 8.25 s, Size: 473 B. The response body is JSON:

```
{
  "id": 4,
  "name": null,
  "category": "Groceries",
  "price": 0.0
}
```

Postman interface showing a DELETE request to `http://localhost:55080/api/products?id=4`. The request is configured with the following query parameters:

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> ID	4	
<input type="checkbox"/> Item	rice	
<input type="checkbox"/> Category	Groceries	
<input type="checkbox"/> Price	2M	
Key	Value	Description

The response status is 204 No Content, Time: 25 ms, Size: 305 B. The response body is empty.

Postman interface showing a GET request to `http://localhost:55080/api/products`. The request is configured with the following query parameters:

KEY	VALUE	DESCRIPTION
<input type="checkbox"/> ID	4	
<input type="checkbox"/> Item	rice	
<input type="checkbox"/> Category	Groceries	
<input type="checkbox"/> Price	2M	
Key	Value	Description

The response status is 200 OK, Time: 15 ms, Size: 431 B. The response body is a JSON array:

```
1 [
2   {
3     "id": 1,
4     "name": "tomato soup",
5     "category": "Groceries",
6     "price": 1.39
7   }
8 ]
```

Task 3

Document for Web Api:

Account/Controller :

/api/Account/Register': var data = {

Email: self.registerEmail(),

Password: self.registerPassword(),

ConfirmPassword: self.registerPassword2()

```
};
```

```
/Token: var loginData = {
```

```
    grant_type: 'password',
```

```
    username: self.loginEmail(),
```

```
    password: self.loginPassword()
```

```
};
```

```
/api/Account/Logout: done(function (data) {
```

```
    // Successfully logged out. Delete the token.
```

```
    self.user("");
```

```
    sessionStorage.removeItem(tokenKey);
```

Register Account Postman:

POST http://localhost:63838/api/Account/Register Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> Email	annie@hotmail.com	
<input checked="" type="checkbox"/> Password	@Abc123	
<input checked="" type="checkbox"/> ConfirmPassword	@Abc123	
Key	Value	Description

Body Cookies Headers (9) Test Results Status: 200 OK Time: 53 ms Size: 320 B

Pretty Raw Preview Visualize Text 1

Log In Account Postman:

POST http://localhost:63838/Token Send Save

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

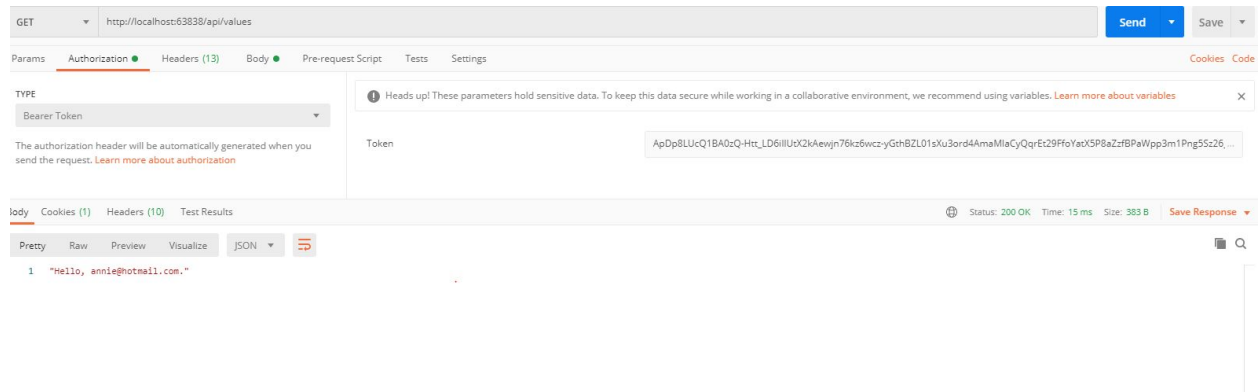
KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> username	annie@hotmail.com	
<input checked="" type="checkbox"/> Password	@Abc123	
<input checked="" type="checkbox"/> grant_type	password	
Key	Value	Description

Body Cookies (1) Headers (10) Test Results Status: 200 OK Time: 1647 ms Size: 1.51 KB Save Response

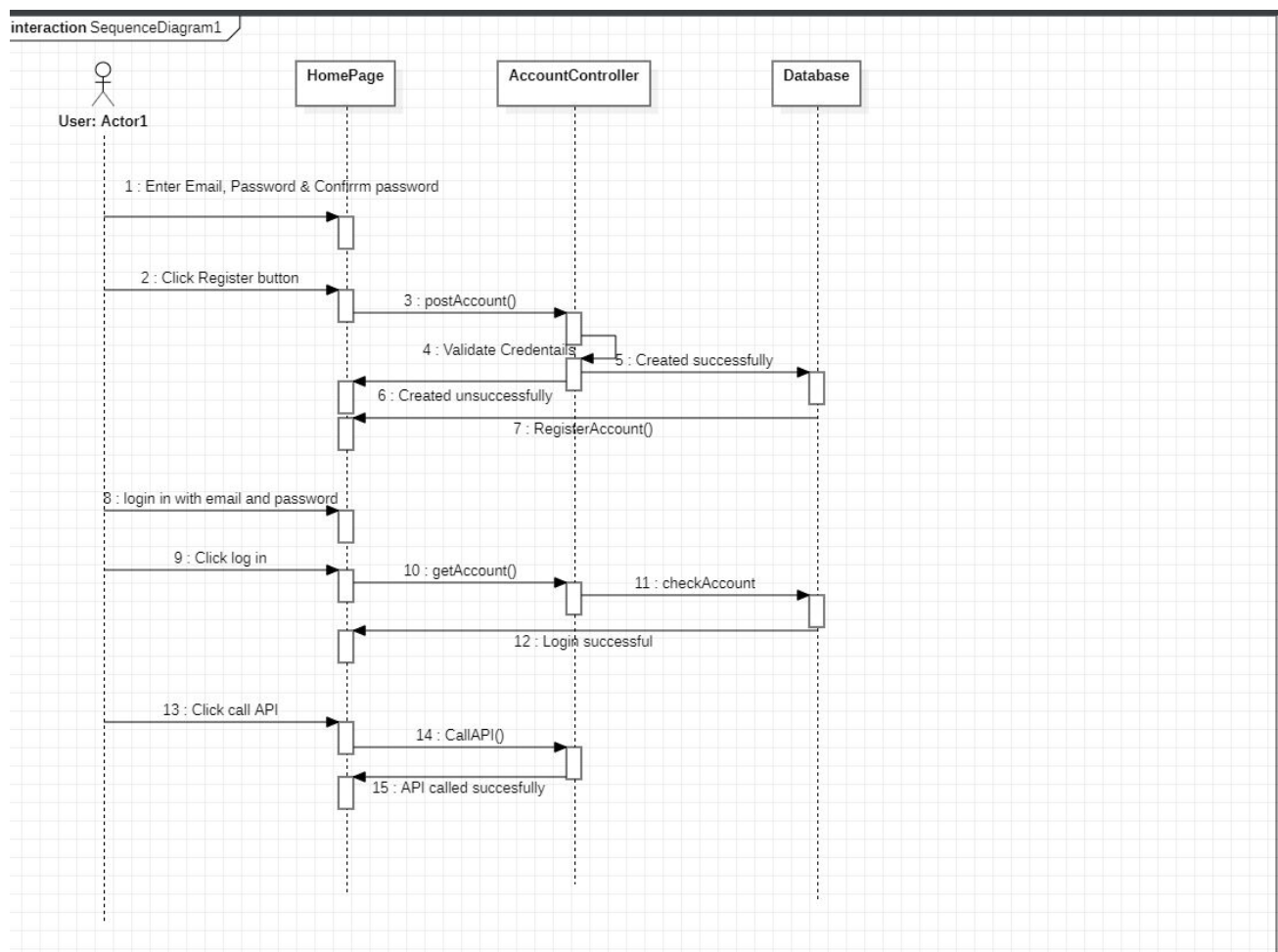
Pretty Raw Preview Visualize JSON

```
1 {
2   "access_token":
3     "G-P9ffffQvCgltz_3YEzQ6SgrforUcdpyUrxASISUCplVVtOhtdm8ff4-uv68soCcYlwQkIpcIdmwrEk396m76is8f9aZs2I4b86b3FHPgdc08US_w091nh2RgHV_RRDGuZP-12I3HQTLnfzKsnIYIk1bcwt2zL6@92VaxSLH1HQ9Tn2ze0Vec1eOm42wQFT13FQFulSP4ajdm1588GzLxnH5_P
4     r1Kx4Ldkg01ukmqPbEjVnCb6NH2xVsod1G1-FQL3OpRUG11ISpg31ONQ0YdXsBRFXC3hBhNRkgKU79cS6Uic_YE-cjoXaIvfrCvY1t3L8Nd6iounJgaUvvyL91PDWuDoJes1FOOudb0eYTXfpSNCmxCS3NF1pInfJzu7q1jY9pHV7HKmpt2HR-zqtNglGn342dQMqLE181L0wJC621fnCnTS1T
5     smpGwIou67X9NKKXduQVhloaAq--xqVU9_rxuTym8ht1ITFvaSE8GcmIQ7XU7rpahckN8",
6   "token_type": "bearer",
7   "expires_in": 1209599,
8   "username": "annie@hotmail.com",
9   ".issued": "Wed, 24 Jun 2020 02:43:54 GMT",
10  ".expires": "Wed, 08 Jul 2020 02:43:54 GMT"
```


GetAPI(Token) Postman:



Sequence Diagram:



Task 4

Document for Web Api:

TalentController:

Talent.cs :

```
public class Talent
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string ShortName { get; set; }
    public string Reknown { get; set; }
    public string Bio { get; set; }
}
```

Get : api/talents: return repository.GetAll();

api/talents/{id:int}:

```
Talent item = repository.Get(id);
if (item == null)
{
    throw new HttpResponseException(HttpStatusCode.NotFound);
}
return item;
```

Get Talent postman:

GET https://localhost:44352/api/talents/ Send Save

Params Authorization Headers (12) Body Pre-request Script Tests Settings Cookies Code

Query Params

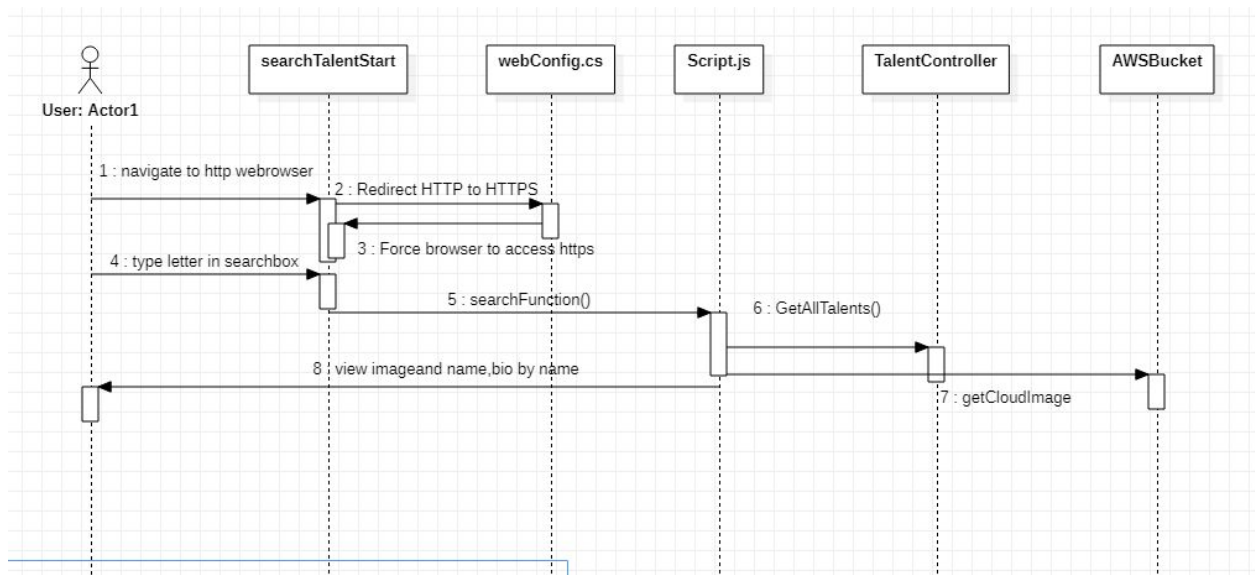
KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (10) Test Results Status: 200 OK Time: 754 ms Size: 2.46 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "Id": 0,
3   "Name": "Barot Bellingham",
4   "ShortName": "Barot Bellingham",
5   "Reknow": "Royal Academy of Painting and Sculpture",
6   "Bio": "Barot has just finished his final year at The Royal Academy of Painting and Sculpture, where he excelled in glass etching paintings and portraiture. Hailed as one of the most diverse artists of his generation, Barot is equally as skilled with watercolors as he is with oils, and is just as well-balanced in different subject areas. Barot's collection entitled \"The Un-Collection\" will adorn the walls of Gilbert Hall, depicting his range of skills and sensibilities - all of them, uniquely Barot, yet undeniably different"
7 },
8 {
9   "Id": 1,
10  "Name": "Jonathan G. Ferrar II",
11  "ShortName": "Jonathan Ferrar",
12  "Reknow": "Artist to Watch in 2012",
13  "Bio": "The Artist to Watch in 2012 by the London Review, Jonathan has already sold one of the highest priced-commissions paid to an art student, ever on record. The piece, entitled Gratitude Resort, a work in oil and mixed media, was sold for $750,000 and Jonathan donated all the proceeds to Art for Peace, an organization that provides college art scholarships for creative children in developing nations"
14 },
15 {
16   "Id": 2,
17   "Name": "Hillary Hewitt Goldwynn-Post",
18   "ShortName": "Hillary_Goldwynn",
19   "Reknow": "New York University",
20   "Bio": "Hillary is a sophomore art sculpture student at New York University, and has already won all the major international prizes for new sculptors, including the Divinity Circle, the International Sculptor's Medal, and the Academy of Paris Award. Hillary's CAC exhibit features 25 abstract watercolor paintings that contain only water images including waves, deep sea, and river."
21 },
22 {
23   "Id": 3,
24   "Name": "Hassum Harrod",
25   "ShortName": "Hassum_Harrod",
26   "Reknow": "Art College in New Dehli",
27   "Bio": "The Art College in New Dehli has sponsored Hassum on scholarship for his entire undergraduate career at the university, seeing great promise in his contemporary paintings of landscapes - that use equal parts muted and vibrant tones, and are almost a contradiction in art. Hassum will be speaking on \"The use and absence of color in modern art\" during Thursday's agenda."
28 },
29 }
30 }
```

Sequence Diagram:



Task 5

Generate client ID and Token :

The screenshot shows the Bitly Organization Settings page. The main content area displays 'Account Details' for a 'FREE ACCOUNT'. A modal window titled 'GENERIC ACCESS TOKEN' is open, showing a token and a 'COPY' button. The background page shows account details for a 'FREE ACCOUNT' and a sidebar with navigation options like 'Custom Domains', 'Users', 'Groups', 'Mobile Apps', 'Single Sign-On', and 'Webhooks'.

Organization Settings

SEARCH

CREATE

GENERIC ACCESS TOKEN

Many of Bitly's API methods require an OAuth access token for authentication. You can generate a generic access token by confirming your password below.

But first you must select your default API group. Links will be shortened under this group if you do not provide a group in your request.

Your current default is Free: o_4gr7tkts4. You can update below any time.

Free: o_4gr7tkts4

PASSWORD

Access Token: o_4gr7tkts4

Account Details

FREE ACCOUNT

ACCOUNT NAME

o_4gr7tkts4 Member since May 26.

MONTHLY USAGE

Total Links 6 / 1000

Branded Bitlinks 0 / 50

Custom Domains

Users

Groups

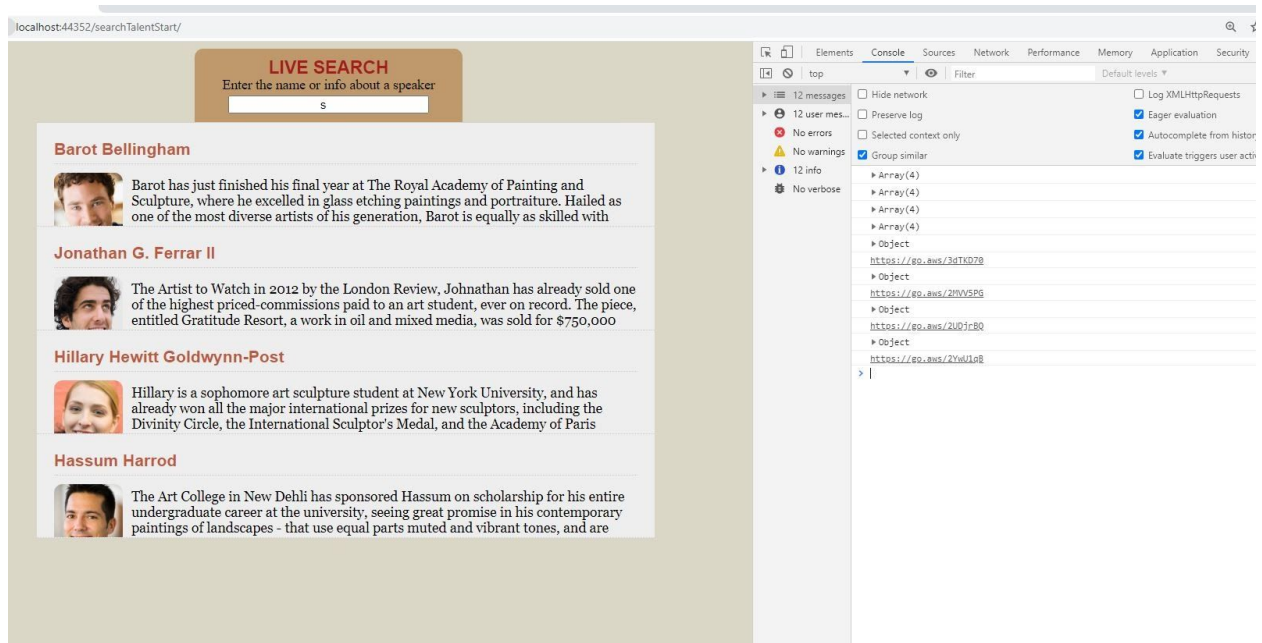
Mobile Apps

Single Sign-On

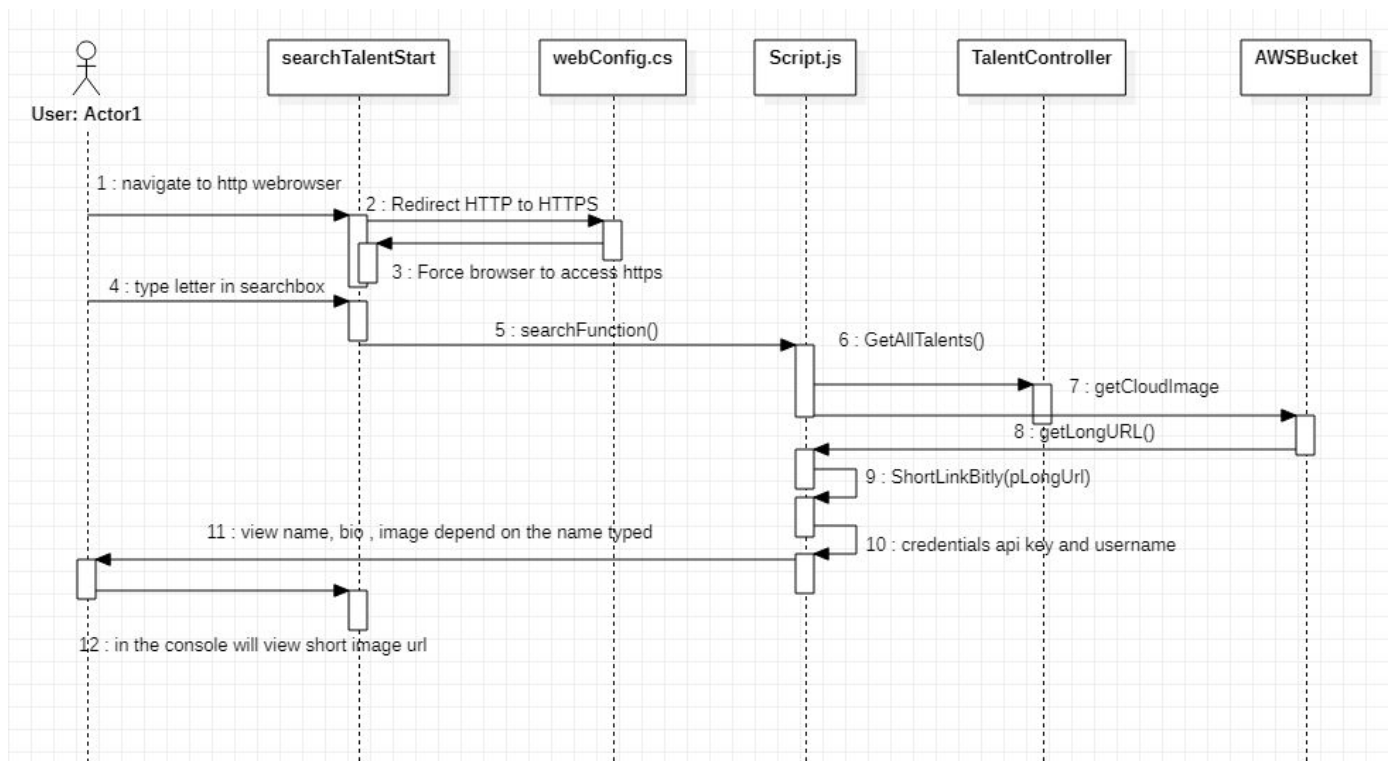
Webhooks

8:23 AM 26-Jun-20

Using short bitly:

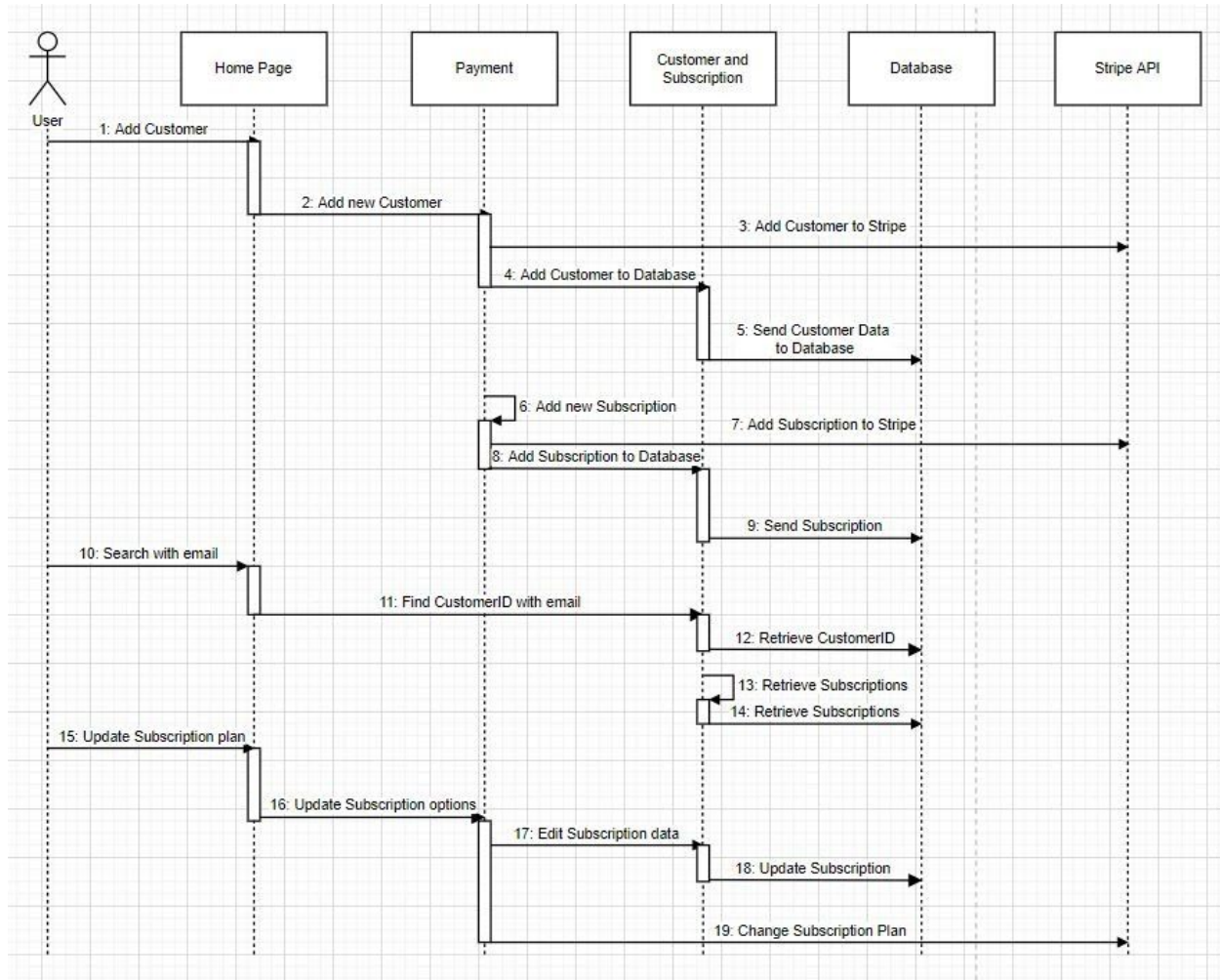


Sequence Diagram:



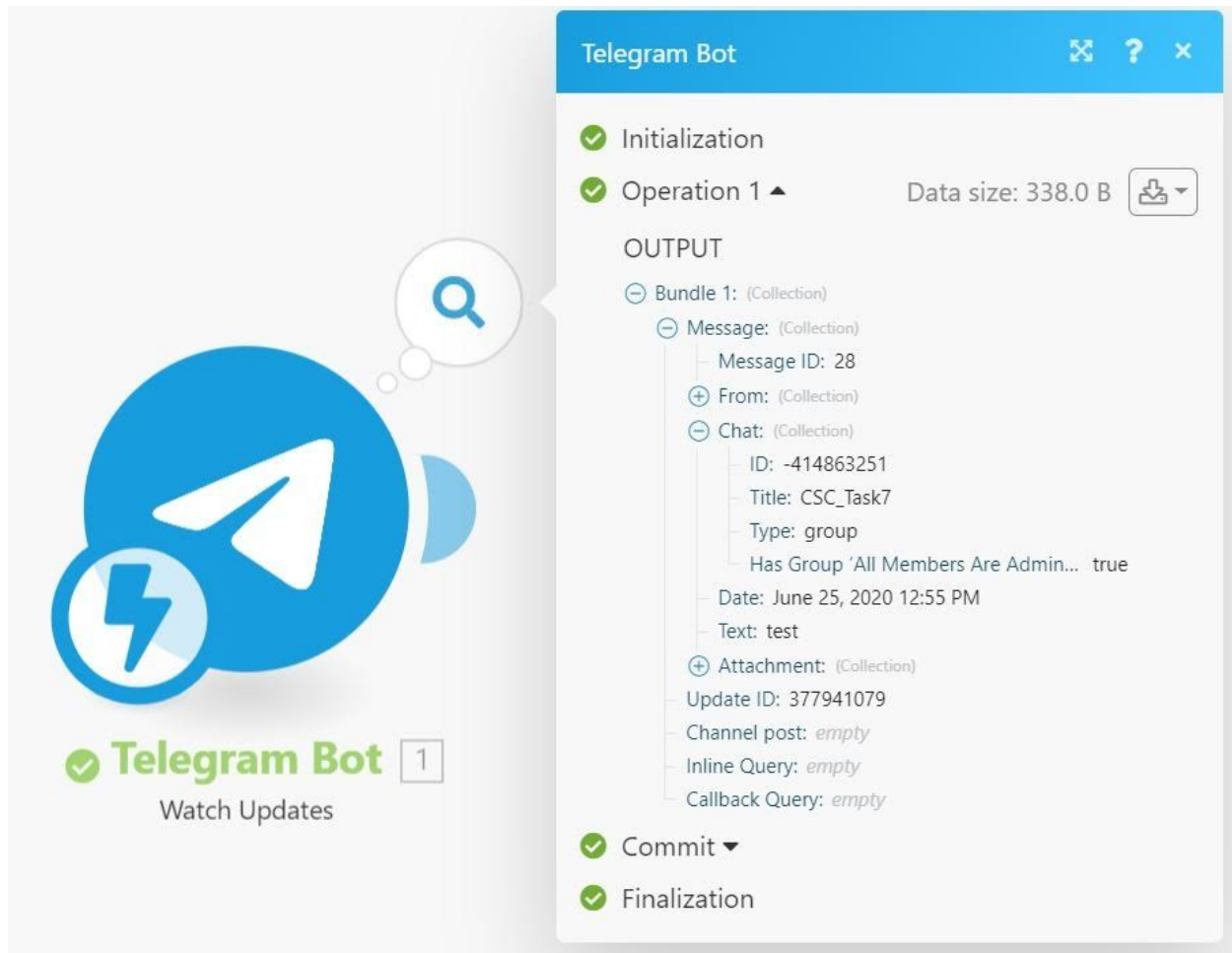
Task 6

Sequence Diagram:



Task 7

Integromat screenshots :



The screenshot displays the Integromat interface for a Telegram Bot. On the left, a large blue Telegram logo is shown with a lightning bolt icon and a magnifying glass icon. Below it, the text "Telegram Bot" is displayed with a green checkmark and a "Watch Updates" button. A small box with the number "1" is next to the bot name.

On the right, a panel titled "Telegram Bot" shows the execution status of a workflow. The status is "Operation 1" with a green checkmark and a data size of "338.0 B". The "OUTPUT" section shows a "Bundle 1: (Collection)" containing a "Message: (Collection)". The message details are as follows:

- Message ID: 28
- From: (Collection)
- Chat: (Collection)
 - ID: -414863251
 - Title: CSC_Task7
 - Type: group
 - Has Group 'All Members Are Admin...': true
- Date: June 25, 2020 12:55 PM
- Text: test
- Attachment: (Collection)
- Update ID: 377941079
- Channel post: empty
- Inline Query: empty
- Callback Query: empty

At the bottom of the panel, the "Commit" and "Finalization" steps are shown with green checkmarks.



Eventbrite 1

Watch Attendees



Telegram Bot 2

Send a Text Message or a Reply

CSC_task7

Ticket Bought

Event Id: 110598705706

Order Id: 1378354252

Amount Paid: £0.00

Ticket Class: General Admission

Attendee Information

Attendee Id: 1923408568

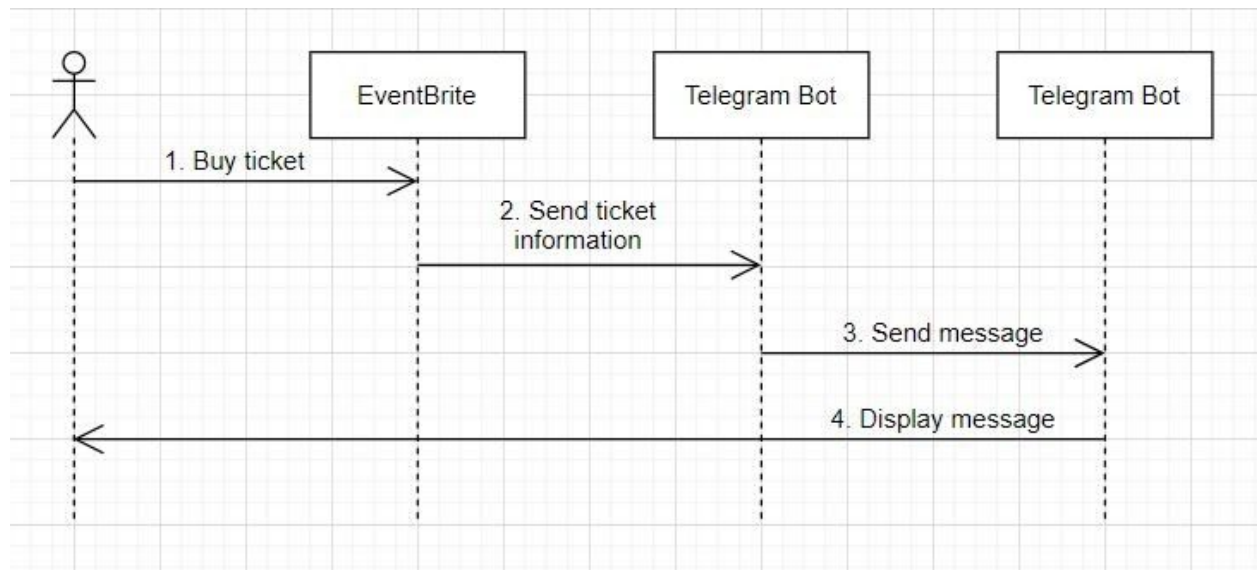
Attendee Email: tayss000129@gmail.com

Attendee Name: suisheng tay

12:58 PM

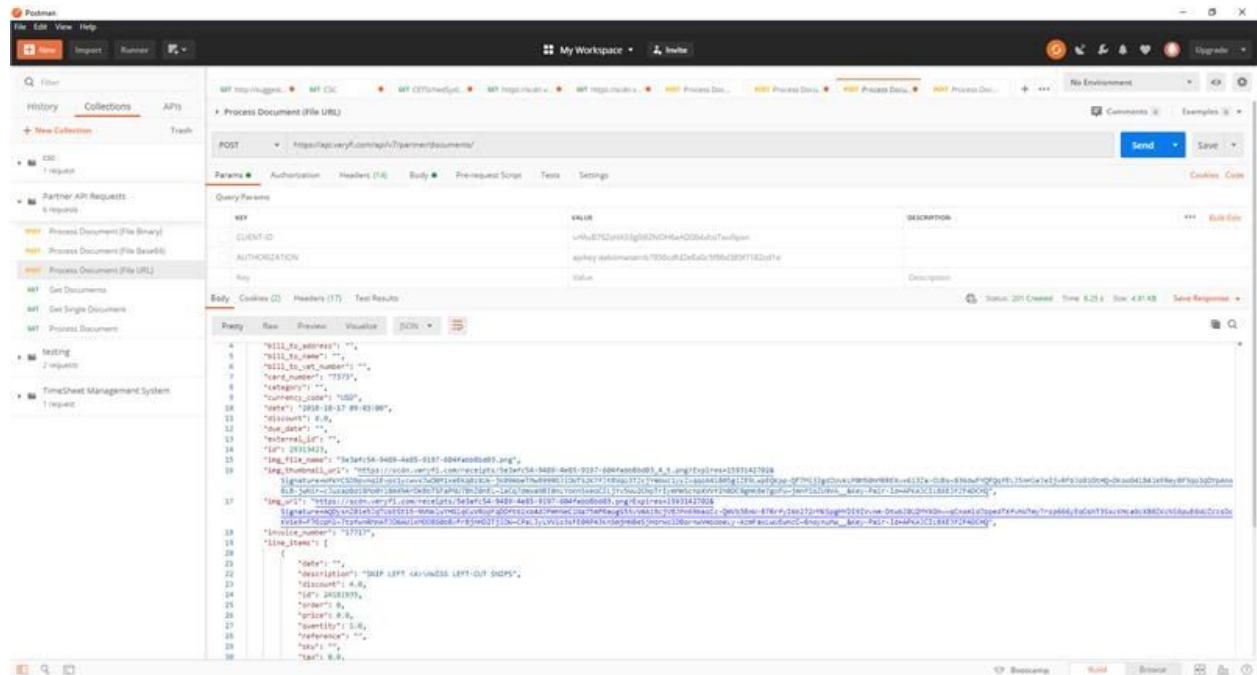
C

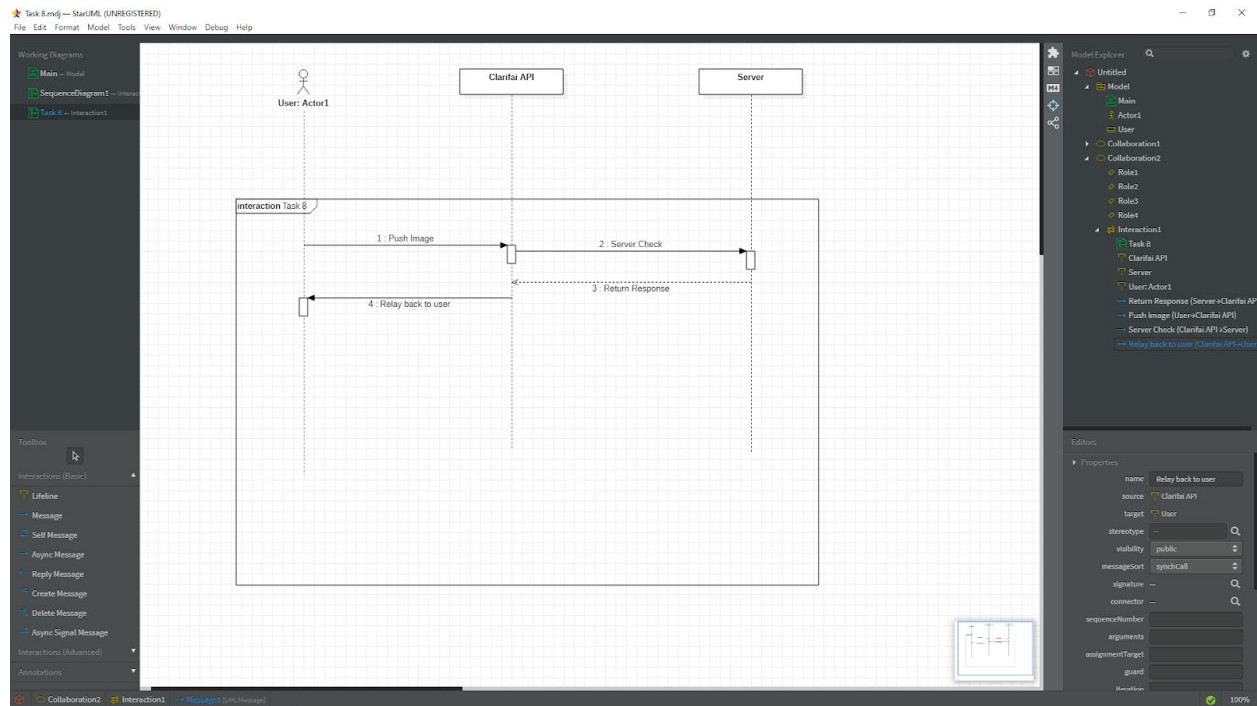
Sequence Diagram:



Task 8

Postman:



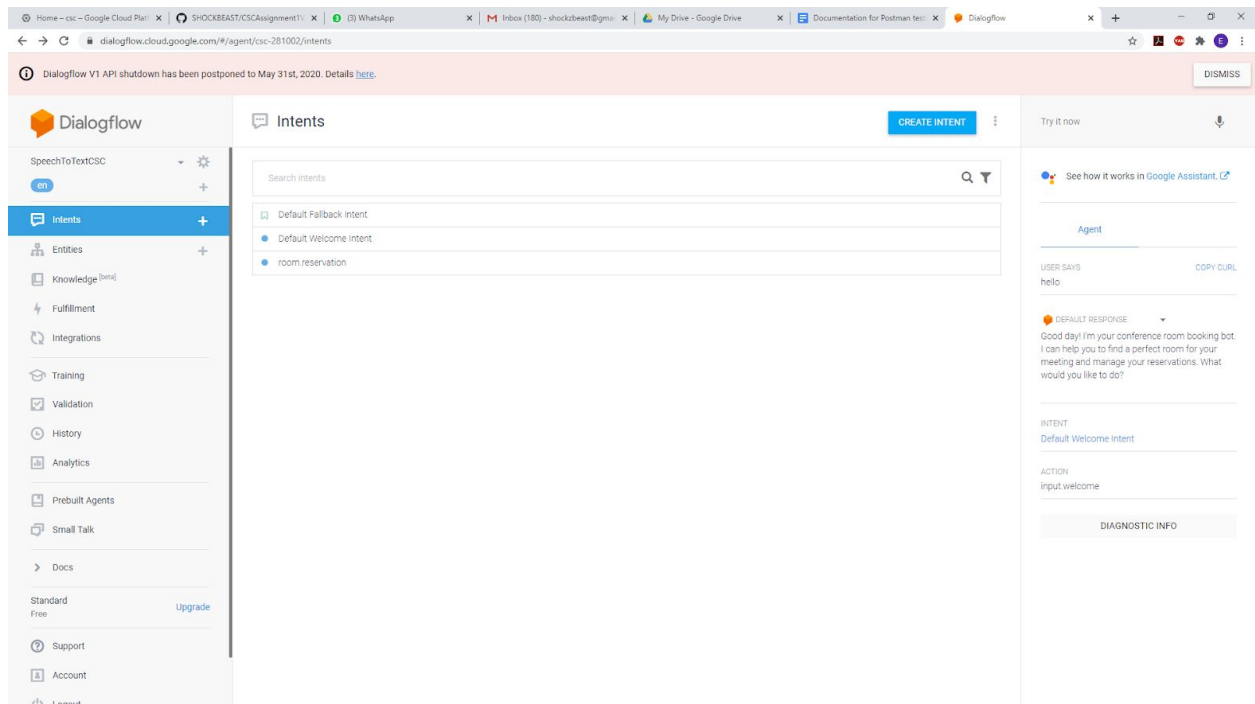


Extra Features:

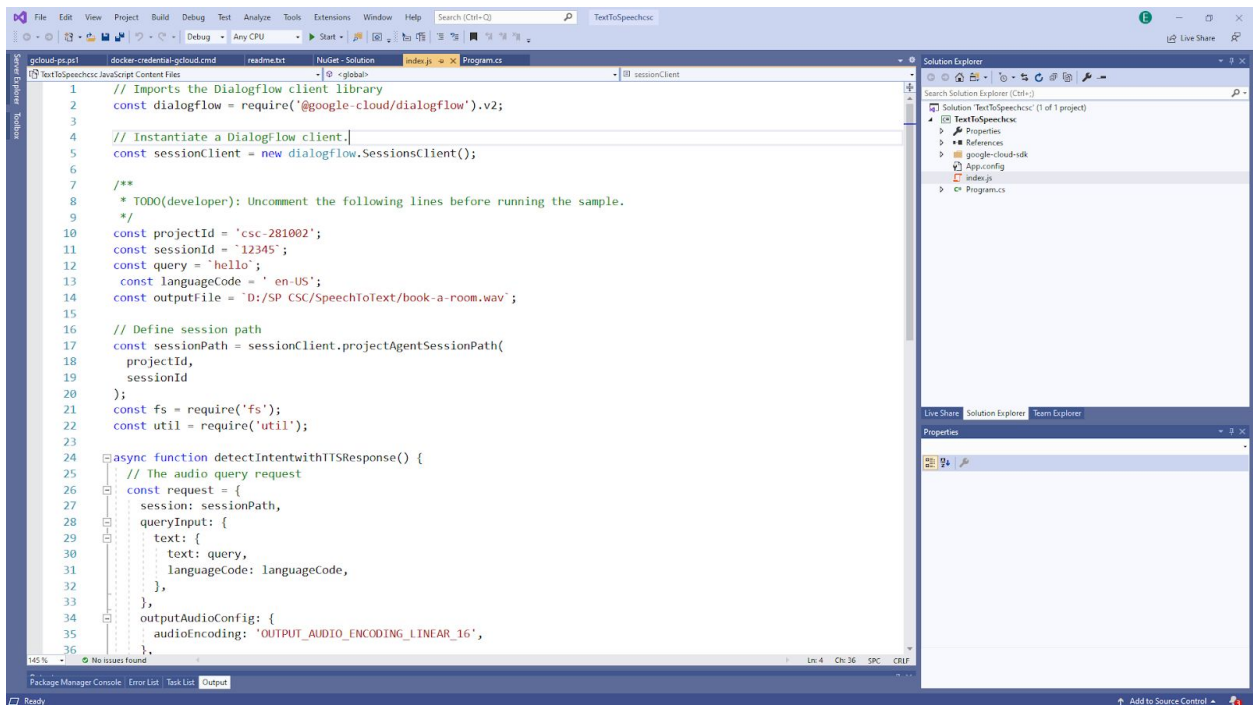
Text to Speech:

ScreenShots:

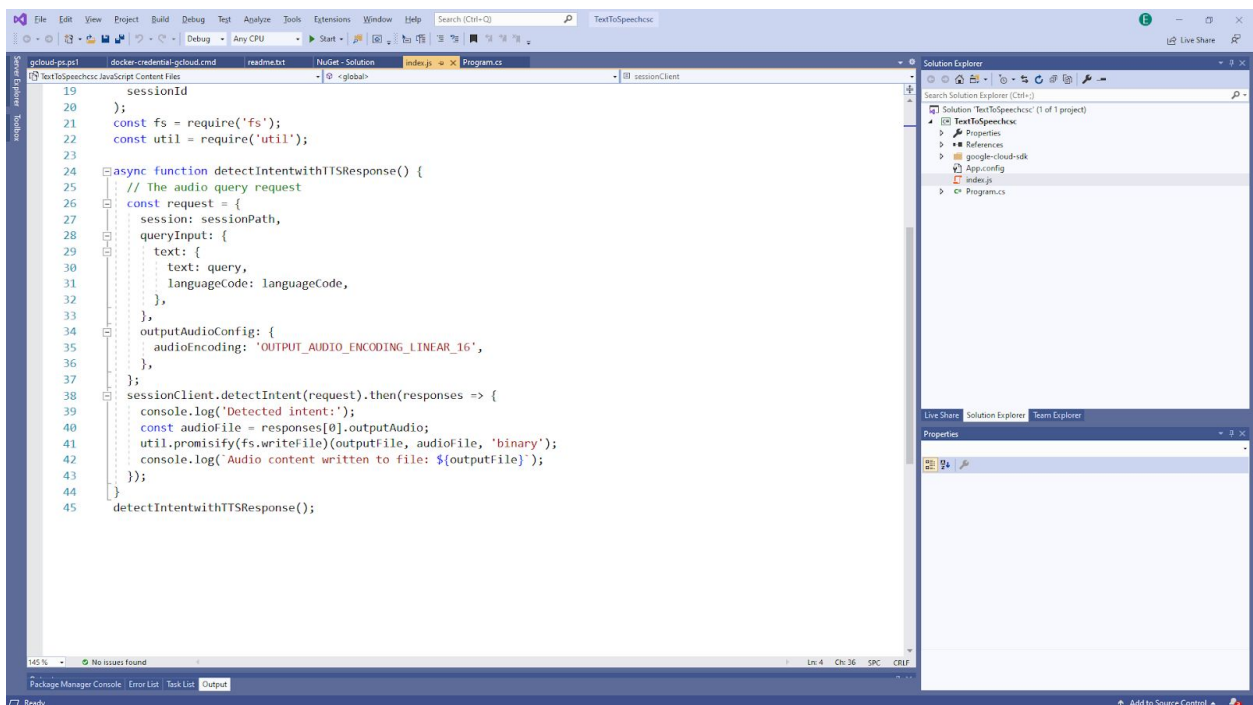
DialogFlow website:



Index.js (javascript file):



```
1 // Imports the Dialogflow client library
2 const dialogflow = require('@google-cloud/dialogflow').v2;
3
4 // Instantiate a Dialogflow client.
5 const sessionClient = new dialogflow.SessionsClient();
6
7 /**
8  * TODO(developer): Uncomment the following lines before running the sample.
9  */
10 const projectId = 'csc-281002';
11 const sessionId = '12345';
12 const query = 'hello';
13 const languageCode = 'en-US';
14 const outputFile = 'D:/SP CSC/SpeechToText/book-a-room.wav';
15
16 // Define session path
17 const sessionPath = sessionClient.projectAgentSessionPath(
18   projectId,
19   sessionId
20 );
21 const fs = require('fs');
22 const util = require('util');
23
24 async function detectIntentwithTTSResponse() {
25   // The audio query request
26   const request = {
27     session: sessionPath,
28     queryInput: {
29       text: {
30         text: query,
31         languageCode: languageCode,
32       },
33     },
34     outputAudioConfig: {
35       audioEncoding: 'OUTPUT_AUDIO_ENCODING_LINEAR_16',
36     },
37   };
38   sessionClient.detectIntent(request).then(responses => {
39     console.log('Detected intent:');
40     const audioFile = responses[0].outputAudio;
41     util.promisify(fs.writeFile)(outputFile, audioFile, 'binary');
42     console.log('Audio content written to file: ${outputFile}');
43   });
44 }
45 detectIntentwithTTSResponse();
```



```
19 sessionId
20 );
21 const fs = require('fs');
22 const util = require('util');
23
24 async function detectIntentwithTTSResponse() {
25   // The audio query request
26   const request = {
27     session: sessionPath,
28     queryInput: {
29       text: {
30         text: query,
31         languageCode: languageCode,
32       },
33     },
34     outputAudioConfig: {
35       audioEncoding: 'OUTPUT_AUDIO_ENCODING_LINEAR_16',
36     },
37   };
38   sessionClient.detectIntent(request).then(responses => {
39     console.log('Detected intent:');
40     const audioFile = responses[0].outputAudio;
41     util.promisify(fs.writeFile)(outputFile, audioFile, 'binary');
42     console.log('Audio content written to file: ${outputFile}');
43   });
44 }
45 detectIntentwithTTSResponse();
```

Command prompt Issue node index.js to run the code

Audio has been changed

The screenshot displays the Google Dialogflow console interface. On the left, a sidebar menu includes options like 'SpeechToTextCSC', 'Intents', 'Entities', 'Knowledge', 'Fulfillment', 'Integrations', 'Training', 'Validation', 'History', 'Analytics', 'Prebuilt Agents', 'Small Talk', 'Docs', 'Support', 'Account', and 'Logout'. The 'Intents' section is currently selected. The main area shows a list of intents: 'Default Fallback intent', 'Default Welcome intent', and 'room.reservation'. A 'CREATE INTENT' button is visible in the top right of this section. A terminal window titled 'Node.js command prompt' is overlaid on the console, showing the following text: 'Your environment has been set up for using Node.js 12.18.1 (x64) and npm.', 'C:\Users\User>cd C:\Users\User\source\repos\TextToSpeechCSC\TextToSpeechCSC', 'C:\Users\User\source\repos\TextToSpeechCSC\TextToSpeechCSC>node index.js', 'Detected intent:', 'Audio content written to file: D:\SP CSC\SpeechToText\book-a-room.wav', and 'C:\Users\User\source\repos\TextToSpeechCSC\TextToSpeechCSC>'. On the right side of the console, there is a 'Try it now' section with a microphone icon and a 'See how it works in Google Assistant' link. Below this, the 'Agent' section shows a 'USER SAYS' field with the text 'hello' and a 'COPY CURL' button. The 'DEFAULT RESPONSE' section shows a message: 'Good day! I'm your conference room booking bot. I can help you to find a perfect room for your meeting and manage your reservations. What would you like to do?'. The 'INTENT' section shows 'Default Welcome intent' and the 'ACTION' section shows 'input:welcome'. A 'DIAGNOSTIC INFO' button is located at the bottom of the right-hand panel.