

Assignment 3 Report

Features

Read me file in the repository describes what we have done for assignment 3. It includes a list of each feature programmed, how that feature improved our system and a snippet of the output that demonstrates our feature.

Data Flow Diagrams

Data flow diagrams level 0 and level 1 are also included in our repository.

Github Repository

<https://github.com/charles-bot-groupTwo/COSC310>

Network Graph

<https://github.com/charles-bot-groupTwo/COSC310/network>

Features Implemented for Assignment 3

- A GUI that has been updated so the user can see conversation history, and the users input.
- Named Entity Recognition
 - Breaks apart a sentence and goes through to recognize named entities.
- Sentiment Analysis
 - Analysis of words in a sentence to determine if the user input is positive or not.
- POS tagging
 - Converting sentences to lists of words to dictate if the word is a noun, verb, adjective, etc.
- Five or more responses that are used for off topic inputs from the user.
- Charles has a total for 5 topics (4 topics from assignment 2, and 1 new topic added for assignment 3). The topics consist of:
 - Watches
 - Phones
 - Laptops
 - Tablets
 - Desktop (New)

Expected Output

- 5 Different Responses -
<https://github.com/charles-bot-groupTwo/COSC310/blob/main/5differentResponses.png>
- Charles GUI -
<https://github.com/charles-bot-groupTwo/COSC310/blob/main/CharlesGUI.png>
- NER - <https://github.com/charles-bot-groupTwo/COSC310/blob/main/NERsnip.png>
- POS tagging -
<https://github.com/charles-bot-groupTwo/COSC310/blob/main/POSTaggersnip.png>
- Sentiment analysis -
<https://github.com/charles-bot-groupTwo/COSC310/blob/main/Sentimentsnip.png>

Automated Unit Testing

Jest framework was set up for automated unit testing for Charles.

The response.test.js file contains the code for all the automated unit tests of all the major functions.

17 test cases were written and all test cases passed successfully.

The list of the functions tested and their test cases are summarized as follows:

function greetings

1. when count is 0. (count is used to track the depth of the conversation).
2. when count is 1.
3. when count is 2.

function convoOne

1. if user chose watch as product type.
2. if user chose laptop as product type.
3. if user chose phone as product type.
4. if user chose tablet as product type.
5. if user chose desktop as product type.

function convoTwo

1. user chose watch and specified budget as 280 dollars.
2. user chose laptop and specified budget as 1440 dollars.
3. user chose phone and specified budget as 800 dollars.

4. user chose tablet and specified budget as 1200 dollars.
5. user chose desktop and specified budget as 1500 dollars.

function convoThree

1. user chose laptop, specified budget and size.
2. user chose phone, specified budget and size.
3. user chose tablet, specified budet and size.
4. user chose desktop, specified budget and size.

Snippet showing automated unit testing

```
Test Suites: 1 passed, 1 total
Tests:      17 passed, 17 total
Snapshots:  0 total
Time:       0.53 s
PS C:\Users\issah\Desktop\COSC310_MoreTests> npm run test

> cosc310@1.0.0 test
> jest

PASS ./responses.test.js
  ✓ if user choose watch on convoOne (1 ms)
  ✓ if user choose laptop on convoOne (1 ms)
  ✓ if user choose phone on convoOne (1 ms)
  ✓ if user choose tablet on convoOne
  ✓ if user choose desktop on convoOne
  ✓ greetings message test when count 0
  ✓ greetings message test when count 1
  ✓ greetings message test when count 2
  ✓ product type 1 with budget test for convo2
  ✓ product type 2 with budget test for convo2
  ✓ product type 3 with budget test for convo2 (1 ms)
  ✓ product type 4 with budget test for convo2
  ✓ product type 5 with budget test for convo2
  ✓ product type 2 test for convo3 (1 ms)
  ✓ product type 3 test for convo3
  ✓ product type 4 test for convo3
  ✓ product type 5 test for convo3

Test Suites: 1 passed, 1 total
Tests:      17 passed, 17 total
Snapshots:  0 total
Time:       0.319 s, estimated 1 s
Ran all test suites.
```

List of Limitations

1. After Charles provides the user with a recommendation, the user needs to refresh the page in order to have another conversation with him.

Output demonstrating Limitation

Hi my name is Charles 🙋
I will be assisting you by recommending you an Apple product perfect for you. Lets get started, what type of device are you looking for? (Watch, Laptop, Phone, Tablet, PC)?

phone

you chose phone, what is your budget?

6000

Got it! what size would you prefer (x-small, small, medium, large)?

large

Both the iPhone 14 and iPhone 14 Pro Max fit this screen size but the iPhone 14 Pro Max is \$300 more than the iPhone 14 Plus. A specification comparison can be made on the Apple website to list every one of their differences.

tablet

Not sure, could you try again please

[Enter Here](#)

2. For the budget conversation of Charles, Charles can only process numerical values. For example, if my budget was 500, Charles would not understand if the user writes it as “five hundred”.

Hi my name is Charles 🙋
I will be assisting you by recommending you an Apple product perfect for you. Lets get started, what type of device are you looking for? (Watch, Laptop, Phone, Tablet, PC)?

phone

you chose phone, what is your budget?

five hundred

[Enter Here](#)

Send

List of features that can be extracted from our code.

- Our conversation flow, questions, and recommendations can be used and implemented as an API for other users
- Our Sentiment analysis can be extracted and implemented into others code to help users further analyze their text so they can respond appropriately.
- The Named Entity Recognition feature can also be extracted to help users analyze parts of a given text to extract key words to further help the efficiency of the program.
- Users can also use our POS tagging that can further assist them in analyzing their text.
- Charles can be used by users as a fully functional API, which can be implemented for product recommendation purposes.

PROJECT MANAGEMENT TOOL

Jira was used as the project management tool. 8 tasks were assigned and completed successfully. AK is initials for Aman, J is initials for J, and IH is initials for Issa.

Projects / Charles

CHAR board

Search: [] | [IH] [AK] [J] [] | Epic ▾

TO DO	IN PROGRESS	DONE 8 ISSUES ✓
<p>+ Create issue</p>		<p>Improve GUI</p> <p>✓ KING-43 ✓ AK</p> <p>Additional responses for when user asks something outside of topic</p> <p>✓ KING-47 ✓ AK</p> <p>Extra dialogue and options for devices</p> <p>✓ KING-48 ✓ J</p> <p>Rewrite code for AUTOMATED unit testing</p> <p>✓ KING-53 ✓ J</p> <p>DFD Diagrams</p> <p>✓ KING-51 ✓ IH</p> <p>Automated unit testing</p> <p>✓ KING-45 ✓ IH</p> <p>NLP Toolkits</p> <p>✓ KING-46 ✓ AK</p> <p>Update readme file</p> <p>✓ KING-52 ✓ AK</p> <p>See all Done issues</p>

Meeting Minutes

October 21st, 2022

Meeting Minutes

Members Present: Aman Kaushik, Jai Patel, and Issa Hashim

Location: Discord

Discussion:

The assignment due date and rubric was discussed. Mainly, this meeting focused on assigning members with different tasks related to assignment 3.

- Issa Hashim – documentation, testing, and presentation.
- Aman Kaushik – programming features, testing, and presentation.
- Jai Patel – programming features, testing, and presentation.

Team members were in agreement that the task breakdown was fairly equal across the board.

Next meeting was decided to be scheduled on October 28th, 2022, at 1 pm PST.

October 28th, 2022

Meeting Minutes

Members present: Aman Kaushik, Jai Patel, and Issa Hashim

Location: Discord group call

Discussion:

This meeting centered around assessing progress on assignment. As team members were busy with

midterms and other projects for other courses, progress on the assignment was a bit behind schedule.

However, members agreed to pick up the pace in the following week for a successful submission by the

assignment deadline.

Setting up a Node.js environment was discussed and decided that it was necessary to implement toolkits. Team members agreed to read up on how to implement a Node.js environment for the project.

Discussion of selecting an API that would be best suited for NLP toolkits were discussed (Stanford, Google, wink etc.). A final API was not selected but a few choices were narrowed down.

Next meeting was decided to be scheduled on November 4th, 2022

November 4th, 2022

Meeting Minutes

Members present: Aman Kaushik, Jai Patel, and Issa Hashim

Location: Discord

Discussion:

This meeting centered around more or less the same topics that took place on the previous meeting of

Team members agreed to communicate via discord regularly this week to provide updates on the

assignment and to schedule a time for video recording once appropriate progress has been made.

Overall, a short discussion; however, team members spent time during the reading week working on their assigned tasks and also being available for any discord call sessions should the need arise.

November 7th to November 14th

Meeting Minutes

Members present: Aman Kaushik, Jai Patel, and Issa Hashim

Location: Discord

Discussion:

Multiple meetings over the reading week were held. There was a lot of trouble figuring out the implementation of the NLP toolkits. Node.js environment setup was proving to be more difficult than anticipated. Meetings centered around strategies/alternatives to progress forward.

Overall, there were a lot of discussions and collaboration during the week to make progress on the assignment.