Software Engineering II Final Report

Group Name: Five Guys

Group Members:

Al-Muqthadir Ajiboye #6148068

Nosetama Imoisili #6568588

Zach Yerrill #6589451

Michael Woody #6369201

Jordan Chilcott #6271357

Yanis Souiki #6284392

Kam Sadiq #6365548

Christian Perdigao #6223283

Features implemented from previous sprints

Completed:

- Build webpage interface
- Input field
- Styling interface and ability to close
- Support for other devices e.g. enabling it to run on mobile devices
- Database creation
- Chatbot response system
- Relevant answers based on context
- User feedback after input
- Display of chatlog
- NLU to match user input with context
- Web scraper for Data Collection to populate Database
- Improvement of scraper to gather athlete information
- Implementation of subcategories for chatbot to find answers easier
- Category and Subcategory Generation
- Improvements to how the NLP answers questions

- Migration of NLP models to backend for smoother run and quicker load
- Implementation of Linux server for
- Scraper improvements for athlete information
- Use of NoSQL structures for ease of access to information
- Testing chatbot response
- Proper collection of keywords
- Testing for chatbot dependability

    - Check if common questions can be answered properly

    - Check if chatbot correctly matches input with categories and subcategories

    - Check if system runs for more than 24 hrs.

- Improvements to Mobile UI
- Handling of unfamiliar input.
- Improved functionality on chatbot's ability to get information and display it.
- Name recognition
- Help function to display available commands
- Addition of open-source credit file

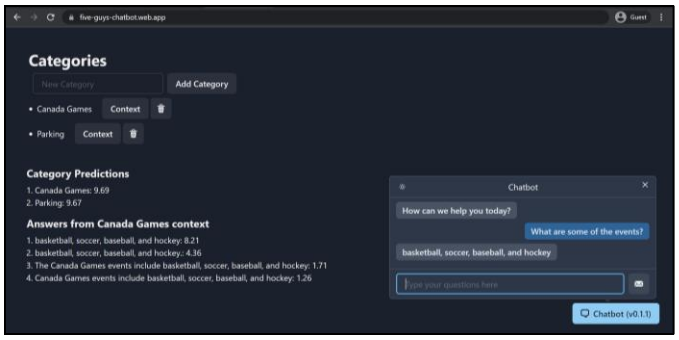
Features exempted

- Analytics Tracking
- Chat history
- Exporting chat log
- More greetings

Process description

From the start of developing this product we gathered all our user stories to gain a general idea of what the main goal is. The initial backlog consisted of basic functionalities for the chatbot's front-end and back-end such as the user interface and feedback from the chatbot and obtaining information through scraping used to fill the database. We decided to go with nothing less than 8 user stories and tasks for each of our backlogs to complete every task and have a feasible product by the end of the project's timeline. This also made it easier to divide the tasks and distribute them to each member of our group to optimize our work efficiency. From the initial backlog we were separated into back-end and front-end sub-groups, where each side communicated with each other and had 4 people tackling at least 4 tasks together with occasional communication between the sub-groups. As the sprints went on and tasks were being completed, we began to incorporate the connection between the two parts of our software there for abolishing the sub-groups so we could focus on making this connection work. There were some issues we came across which ultimately led us to abandon the relational database we had used with SQL and move forward with a NoSQL structure to storing our data since it was also more scalable for our software. During our sprints the tasks that were less important to the major functionality of the product were pushed back, prioritizing the more important ones and those that were challenging had more people working on them. Later on during the final stages of our product each person was assigned to a task with some people taking more than one. Following is a link to test the completed version of the chatbot software. fiveguys.chat

Below are some screenshots of the current working chatbot where it provides a greeting, and we have the categories and context it uses to obtain answers to user queries. We have now switched the model to categories and subcategories because it allows us to properly obtain answers and match the context of the query appropriately to the category in question. There is now a connection with the backend and the frontend. The Artificial Intelligent models used that were gotten from tensor flow which were giving us a loading issue at first since they were in the frontend causing the chatbot to be slow running at first and seemed to be unresponsive on certain browsers and iOS devices is now resolved since we have moved it from the frontend to the backend. This has also been improved more with the implementation and use of a Linux server called Digital Ocean. It serves the frontend by using a web server called Nginx and the backend by using NodeJS with pm2 as the process manager. The language processor works by querying the model using a given set of categories and subcategories. This list of categories and subcategories are stored in a global json file. To see more of the connection between the backend and the frontend and how the models work in the backend check out our server on GitHub (https://github.com/Muqtha/COSC-4P02-Project/tree/main/Server). This will be removed during the final stages before deployment. The server now uses the models to understand user input/queries passed in from the frontend, match the context of the query with the correct categories and subcategories, searches them for an appropriate response and then passes the response over to the frontend which then uses what it has obtained to display. Also seen from the pictures below a scroll bar to view previous chat with the bot has been implemented as well as feedback to let the user know that it has received their question and they are to wait for a response. There is also a close button at the top and the ability to minimize the window of the chatbot is done by pressing the chatbot again the same way it was opened. There were a few issues occurring with the chatbot related to formatting that were resolved, such as text displaying past the boundaries of the message window. If you would like to view the json that contains the current list of categories and subcategories, please click the following link categories.json which will take you to its location on the group's GitHub.



Categories

- Canada Games
- Parking

Category Predictions

1. Canada Games: 9.69
2. Parking: 9.67

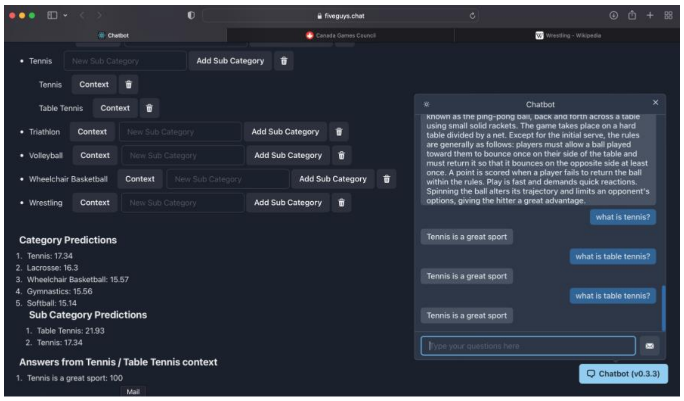
Answers from Canada Games context

1. basketball, soccer, baseball, and hockey: 8.21
2. basketball, soccer, baseball, and hockey.: 4.36
3. The Canada Games events include basketball, soccer, baseball, and hockey: 1.71
4. Canada Games events include basketball, soccer, baseball, and hockey: 1.26

How can we help you today?

What are some of the events?

basketball, soccer, baseball, and hockey



fiveguys.chat

Chatbot | Canada Games Council | Wrestling - Wikipedia

- Tennis — New Sub Category — Add Sub Category
  - Tennis — Context
  - Table Tennis — Context
- Triathlon — Context — New Sub Category — Add Sub Category
- Volleyball — Context — New Sub Category — Add Sub Category
- Wheelchair Basketball — Context — New Sub Category — Add Sub Category
- Wrestling — Context — New Sub Category — Add Sub Category

Category Predictions

1. Tennis: 17.34
2. Lacrosse: 16.3
3. Wheelchair Basketball: 15.57
4. Gymnastics: 15.56
5. Softball: 15.14

Sub Category Predictions

1. Table Tennis: 21.93
2. Tennis: 17.34

Answers from Tennis / Table Tennis context

1. Tennis is a great sport: 100

Chatbot

known as the ping-pong ball, back and forth across a table using small solid rackets. The game takes place on a hard table divided by a net. Except for the initial serve, the rules are generally as follows: players must allow a ball played toward them to bounce once on their side of the table and must return it so that it bounces on the opposite side at least once. A point is scored when a player fails to return the ball within the rules. Play is fast and demands quick reactions. Spinning the ball alters its trajectory and limits an opponent's options, giving the hitter a great advantage.

what is tennis?

Tennis is a great sport

what is table tennis?

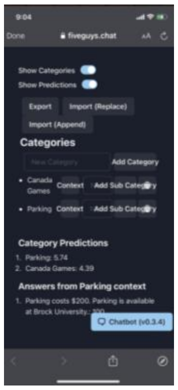
Tennis is a great sport

what is table tennis?

Tennis is a great sport

Type your questions here

Chatbot (v0.3.3)



9:04

Done — fiveguys.chat

Show Categories

Show Predictions

Export | Import (Replace) | Import (Append)

Categories

New Category — Add Category

- Canada Games — Context — Add Sub Category
- Parking — Context — Add Sub Category

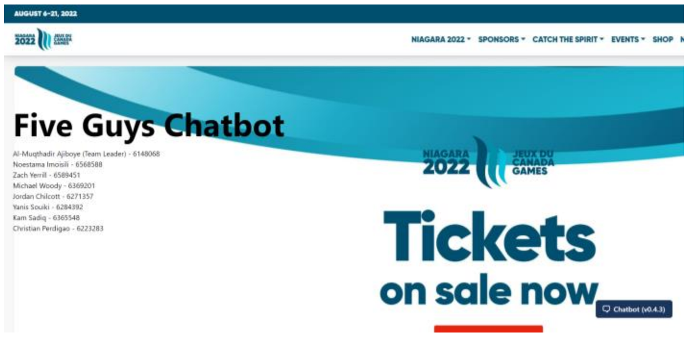
Category Predictions

1. Parking: 5.74
2. Canada Games: 4.39

Answers from Parking context

1. Parking costs $200. Parking is available at Brock University.: 100

Chatbot (v0.3.4)



AUGUST 6–21, 2022

NIAGARA 2022 | SPONSORS | CATCH THE SPIRIT | EVENTS | SHOP

Five Guys Chatbot

Al-Muqthadir Ajiboye (Team Leader) - 6148068
Noestama Imoisili - 6568588
Zach Yerrill - 6589451
Michael Woody - 6369201
Jordan Chilcott - 6271357
Yanis Souiki - 6284392
Kam Sadiq - 6365548
Christian Perdigao - 6223283

Chatbot (v0.4.3)

Challenges Encountered

- Communication - Resolved
- Version Control – Resolved
- Connection to DB – Resolved
- Containerizing – Resolved
- Quicker loading of models for NLP - Resolved
- Accessing Athlete Information - Unresolved
- Formatting Lists in Chatbot Responses - Unresolved

Contributions and achievements

Al-Muqthadir Ajiboye – Generation of User Stories, Implementation of Backend, Implementation of communication between Backend and Database, Communication between Backend and Frontend, Finding bugs or errors in the chatbot and the system, Backlogs, Organization of GitHub files, Progressive implementation on greeting messages and structuring. Added help function.

Nosetama Imoisili – Generation of User Stories, Web Scraper, Improvements on Web Scraper to obtain athlete information, Database, Implementation of communication between Backend and Database, Communication between Backend and Frontend, Implementation on translator for bilingual features, Finding bugs or errors in the chatbot and the system, Web Scraper improvements for Niagara Games Site. Scraped large portions of the database containing the names and information of athletes.

Zach Yerrill – Docker and containerization of chatbot and Web Scraper, Finding and correcting bugs or errors in the chatbot and the system, Backlogs, Organization of GitHub files, Generation and testing of common questions and their answers.

Michael Woody – User Interface, Design, Basic I/O, Implementation of NLP, Migration of AI models from Frontend to Backend, Implementation of Linux server, Improvements to Server and Frontend communication, Improvement of Mobile UI, Changed structure of chatbot from use of DB and SQL to NoSQL and Json files, Improvements made to NLP for name recognition. Progressive improvements to Mobile UI, Troubleshooting and fixing issues with chatbot and bugs in the system, Implementation of features for use of NoSQL. Added the Opensource credit file.

Jordan Chilcott – Structure for Pre-Generated Responses, Category and subcategory generation

Yanis Souiki – Backlogs, Changed structure of chatbot from use of DB and SQL to NoSQL and Json files, Finding bugs or errors in the chatbot and the system, Progressive integration of google analytics with chatbot. Implemented functionality for Categories and Subcategories

Kam Sadiq - Generation of User Stories, Generation and testing of common questions and their answers, Progressive implementation of obtaining chat history and logs.

Christian Perdigao – Schedule Organization, Backlogs, Implementation of communication between Backend and Database, Changed structure of chatbot from use of DB and SQL to NoSQL and Json files, Troubleshooting and fixing issues with chatbot, json files and bugs in the system, Head in category and subcategory generation and chatbot testing.

Link to Reports on GitHub: https://github.com/Muqtha/COSC-4P02-Project/tree/main/Reports