

School of Computing, Creative Technology and Engineering

**Module: Fundamentals of Computer Programming**

**Academic Year: 2023/24**

**Level 4: Semester 1**

**Assignment Title: Project chabot-Documentation**

**Date Due: Jan 16, 2025**

**Tutor: Saurav Gautam**

**Student Name: Sange Doma Tamang**

**Student ID: 10260**

**GitHub Link:** [**https://github.com/Sangedoma/Programming-Portfolio/tree/main/project-chat**](https://github.com/Sangedoma/Programming-Portfolio/tree/main/project-chat)

**Project-chatbot Documentation**

**Introduction:**

The project follows the creation of a chatbot that answers questions for us regarding the university. It recognizes some words and answers accordingly.

**Functions Used:**

**Def get\_users\_name():**

Function is defined to input the user’s name. It further uses the users\_name func which will store the name entered by the user.

**Def get\_bots\_name(users\_name):**

This function is defined to randomly generate a name for the chatbot as per the user’s name. Her, random.shuffle(bots\_name) shuffles the bots name from the list. The length of the user\_name gets counted and if it is less than 5 then, returns the element with index 1 from the randomly shuffled list. And if it’s more than 5 then, returns with index 3 from the randomly shuffled list.

**Def exit\_phrases():**

Function defined which stores exit phrases.

**Def get\_responses(user\_input):**

Here, the users input are analysed and if it matches any of the responses in the bot\_responeses.py then, it responds accordingly. Otherwise, it gives an error message.

**Def chatbot():**

This is the main function where all the workflow for the chatbot gets controlled. The bot greets the user and responds to their response according the bot\_responses.py. If the recognized words match with the users input then it answers as per that. When the user input any of the exit\_phrases then, the bot gives a bye message and the loop ends as well.

**User Input Handling:**

The error handling in this project is quite simple. If the user input does not align with any of the responses then, it simply gives:

"I don't have information for that. Please ask me about something else."

In get\_responses(), the user input is converted to lowercase and split into words, making it case-insensitive and easier to compare with responses. The exit\_phrases also work well with the users input and immediately breaks the loop when any words present in that list is mentioned in the user input.

**Conclusion:**

While the chatbot welcomes the user with a cheery message. It tends to recognize the users\_response and replies accordingly. The name for the bot is also randomised as per the user name length. The chatbot then exits the loop when it recognizes any of the exit phrases.