Snarkbot Functional Description

Project Description:

“Snarkbot” is a chatbot I created for a GUI class. The high-level goal of the program is to provide a web application for kids to use to get through being bullied. The purpose of this project is to prevent bullying by giving victims a way to stand up to their oppressors. Snarkbot is catered towards children ages 5-22 as those are the age ranges spent in school.

What it does:

Snarkbot is a pocket friend that sticks up for a user when they need it. Snarkbot will feed the user comebacks when people are mean or rude to them. When the user inputs a mean phrase somebody said to them, Snarkbot will print out a witty response. The two sides to the conversation in the application are the user and their adversary. Snarkbot simply prints what the user should say in their speech bubble.

Key features of the Snarkbot are that it should look like any other application, so that bullies don’t see the application open on a victim’s phone and use it as a new topic for bullying. However, it should be simple enough to use that even a new user doesn’t have to spend precious response time fiddling with the application – if the user spends too much time looking at their phone in the face of a bully then using this app might backfire. Snarkbot should be able to parse a bully’s taunt and give the user a witty comeback to that specific taunt.

System Description:

Snarkbot is simply made of html and javascript. Rather than holding potential responses in a database, they’re stored in a 2d array matching keywords to preset witty responses. The javascript handles sending responses based on input and maintaining states, and the html formats the page.

Function Definitions:

Function	Initialize Profile
Parameters (explicit, implicit)	Age, Name (required), Gender, Snark Level
Description	A new user is created containing all information supplied by the user, most of which is not required. This information is stored in the system as the user’s profile. After entering this information and creating their profile, a user can modify or redo their profile from scratch.
Feedback	If a user chooses to cancel, no profile will be created and the responses generated by the system will be generic as in they won’t add personal details at all. If the user chooses to save their profile, responses generated by the system will be personalized using information given where appropriate depending on the response.
Error Conditions	• No name was given. Remain on the same page until a name is given. • Name is too long. The name will be cut off and saved.

Function	Input an insult & generate a response
Parameters (explicit, implicit)	Insult as a string. Implicit: User profile data and saved Antagonist data

Description	An insult is given to a user by an antagonist. The app is intended for one user at a time, but many different antagonists can be used. Whatever the “Active” antagonist is at the time will be used when this insult is entered, so the user doesn’t have to select the antagonist every time they use the application (to save time). After giving the system an insult, the user should get a suggested response back.
Feedback	If a user chooses to cancel, they will be taken back to the main page. If they submit a string, the string will be saved in the system under the list of things the antagonist has said already on the antagonist’s profile. The function should output a response for the user to make. Responses given will be personalized based on user data, antagonist data, and the original insult.
Error Conditions	• No insulting string was given. Remain on the same page until a string is given. • String is too long. The user will be asked to shorten it or take our irrelevant pieces.

Function	Initialize antagonist
Parameters (explicit, implicit)	Age, Name, Gender
Description	A new antagonist is created containing all information supplied by the user, all of which is not required. This is because there can be an antagonist that the user knows nothing about. This information is stored in the system as the antagonist’s profile. After entering this information and creating their profile, a user can modify or redo the antagonist’s profile from scratch. They can also delete an antagonist from the application which would also get rid of any records of what they have said.
Feedback	If a user chooses to cancel, no profile will be created and the responses generated by the system will be generic as in they won’t add personal details at all. If the user chooses to save their profile, responses generated by the system will be personalized using information given where appropriate depending on the response.
Error Conditions	• Name is too long. The name will be cut off and saved.

External documentation:

Some interesting documents related to this project are in the /olddocumentation subfolder of this zip file.

A quick demo for it that I made for the class can be found here:
https://www.youtube.com/watch?v=hR2H2SF1S7o

A presentation I made for the class with a funny little skit in it:
https://www.youtube.com/watch?v=cNlGGpfe7xQ

An additional README can be found at the github location of the code:
https://github.com/tessmichi/Snarkbot