15-112, Summer 2, Term Project - Proposal

Brian Lim
Section B
blim2

General Info

Morse Coded is a project that seeks to offer its user a number of different ways to interact with morse code. Depending on how the program is used, the user will be able to exchange messages to and from the program in various ways: normal characters (ex. TEXT); marks and spaces (ex. - . .-.. -); flashing pulses (with audio tones, if possible); and pressing a button to form pulses. These forms of interaction will allow the user to play some morse code based games and to use a morse code translator.

The program will also allow the user to create an account and log in, or to just play as a guest. The program will keep track of various user data in separate files; for the purposes of this project, these files will just be stored locally. The user will be able to access their own performance history for the morse code games, even when the program is closed and reopened.

Features

Translator

This feature will contain two text fields: one that takes/displays normal characters, and one that takes/displays marks. Typing in one field will modify the other. There will also be a button that will modify both fields if used.

Speedometer

This feature will ask the user to transmit a sample message through button presses. When the user is finished, the program will attempt to determine the user's transmission speed by measuring the length of each pulse.

Coder

This feature will have two modes:

Encode, where the user needs to use a button to transmit a given character;
and Decode, where the user needs to identify a flashing pulse.

Points will be awarded to the user based on their accuracy. There will be options to adjust the transmission speed and to adjust the timing error that the program will allow for the user's button presses. There will also be a cheat mode, which will make Encode display the marks corresponding to the given character, and make Decode display a flashing character instead of a flashing dot.

Morse Search

Similar to conventional word searches, except using marks and spaces instead of normal characters (though the word bank itself will just use English words in normal characters). Because words can be very lengthy when represented in marks, the program will first verify that the word is below a certain length before accepting it into the word bank. It will attempt to place each word randomly; if it fails, it will undo its word placement and try again. If it fails enough times, it will assume that the word bank cannot be used and try a new word bank. Points will be awarded to the user based on the time they took to finish.

User Accounts

The user will be able to create accounts and log in. While they are logged in, the program will keep record of their performance. The user will be able to view their performance history.

Modules

All of the following modules are in python's default library.

hashlib

for encrypting user passwords

wave (or some other sound module)

for playing audio tones to represent morse code pulses