

Description: This cold-calling system is intended to be used by a class instructor to easily call on students while giving a lecture. The instructor will be able to interact with a window at the top of their screen, and has the option to start the program. Upon the start, 4 random student names will appear in this window. The instructor will be able to use the arrow keys to move between students, remove a student once they have been called on, and flag student's who they may want to reach out to. The keyboard input will save data which can be viewed later which keeps track of how many times a student was called upon and any notes on the student's answer.

Authors: Ellie Kobak, Kayln Koyanagi, Liza Richards, Kelly Schombert

Date created:

1/14/2022

Reason for Project: This project was created for the Anthony Hornof's CIS 422 class during Winter of 2022 at the University of Oregon. It's reason is to allow the professor to easily call upon all students an equivalent amount of times during class lectures.

How to Compile and Run Program:

1. Go to github directory and select the Cold Call zip file
2. Download this zip file "ColdCall.zip" to your computer
3. Open the terminal application using "Command + space bar" to open the search window and type in terminal
4. In the terminal window, change the directory to where ColdCall is stored. In most cases, this command would be "cd Downloads/ColdCall"
5. Once in this directory, run the software by typing "python3 DucksOnDeck.py"
6. The terminal will prompt for user input on the class roster file. A test roster has been provided in the ColdCall directory. To use your own, type the path from the home directory to where your desired class roster file is located.
7. Press "ctrl + d" to submit the file
8. The system is now ready for use, select the "New Session" button on the main menu window to begin cold-calling!

Additional Setup: For this program to run as intended, the user must import a class roster list in a ".txt" and tab-delimited format. The first line of this file must contain the following headers as well as be in this order:

<first\_name> <last\_name> <UID#> <Email>

Software Dependencies: Python 3

Directory Structure:

1. "Cold Call" contains the four working system components in the form of python files which are DucksOnDeck.py, buildQueue.py, fileReader.py, fileWriter.py.
2. "Documentation" includes all of the required documentation including the updated SRS, updated SDS, updated Project Plan, Installation Instructions, User Documentation, and Programmer Documentation in pdf format.
3. "Initial Documentation" includes the first versions of the SRS, SDS, and Project Plan that were submitted in the Initial Project Plan.
4. "test\_data" is a folder that contains all of the testing data to be used when running the system. Note that the only file that properly works with the system is the .txt file.