DigiPiano

Overview

DigiPiano is a basic piano application that allows users to learn the basics of how to play the piano without reading sheet music in a follow-along style. It's focused towards pianists that are brand new to the piano, however it can be used by all skill levels. It is 100% free for everyone to use, does not require any kind of account, has several customizable features, and gives users access to a piano without being forced to purchase one.

Installation Tips

- → Navigate to the public repository kaylie-a/cs3300-group
- → Navigate to the light green button that says Code, and press the little greeen arrow
- → Click Download ZIP
- → Navigate to where the ZIP folder was downloaded
- → Select the folder, and extract all
- → Double click on the folder after extracting it, and should see all the github files
- → Open a python code editor, and open the extracted folder
- → Open the DigiPiano.py and main.py files as well as the computer terminal
- → Pip install all the python modules inside the computer's terminal that are currently not installed (the ones underlined in yellow)
- → Finally, also inside of thr command prompt, cd to the relative file path where the downloaded folder is, and update threnvironmental path variable using the command similar to this with thr specific file path $Env:Path = 'C:\Users\mrh3h\Documents\VirtualPiano\fluidsynth' + $Env:Path (Use the file path not this one)
- → From here, run the code either from the command prompt running python main.py, or by just running the executable (but the code will not be updated from just changing DigiPiano.py)

Troubleshooting

Did the downloaded zip folder get unzipped?

- → Go back to where the folder was downloaded, unzip the folder, and continue to follow the installation process

Getting an error regarding an import error saying (FluidSyncth library)?

- → This error is being caused by the environment path variable and needs to be updated correctly

→ Double check the command written in the terminal, and make sure it's written with the proper command syntax

Are some the python modules underlined in yellow within the code editor?

→ This means the modules werent pip installed

→ Pip install each of them inside of the command prompt, restart your python interpreter, and try it again

Architecture Overview

→ Project is broken down into 5 different folders, a .gitignore, the developer documentation, the main DigiPiano.py file, our LICENSE, and README.md

→ The DigiPiano python file is broken down into the top-level code (imports, color specifications, screen definitions, note label printing/key positioning arrays), and into 3 main classes: (the Button, the Callback, and the Piano classes)

→ The Button class is for creating buttons with specific coordinates, image, and scale ratio

→ The Callback class is just a simple container that holder a name and event type

→ The Piano class is used to build the piano UI (its broken down into several functions)

→ The key function draws the diffrence keys using the key positioning arrays defined in the top-level code,

→ The press_key and release_key functions update the UI when a key is pressed/released

→ The draw_keys functions draw the pianos separate keys

→ The play_piano function uses pygame methods as well as shows the piano interface as intended

→ There are a few other methods that are just used for drawing, or file testing

Key Components

→ Piano UI - Main visual of project, adapts/changes based on what the user does to it

→ Pygame – Python module project is based on (draws piano keys, draws icons, reacts accordingly when keys are pressed, highlights when keys are pressed, etc)

→ Mingus – Python module that deals with all music components specifically with the MIDI files (loads MIDI files into the project, plays the corresponding note with the corresponding key, checks that the MIDI files are actually MIDI files)

→ Keyboard – Python module that allows for custom phyical keyboard assignments/mapping

Test Cases

→ Iterate through each mode and menu

→ System visuals are compatible on different screen dimensions

→ Tests each piano key to run each note in sequence

→ Testing for crashing on file input for key presses
→ Verifying the file is MIDI type and run songs

Contribution Guidelines

All code is marked for reuse, but if ther're any questions regarding contribution guidelines, please contact kaguila@uccs.edu

Licensing

→ Using a MIT license
→ Permission is granted to use this software as wanted
→ More information about our license can be found on the GitHub or here cs3300-group/LICENSE at main · kaylie-a/cs3300-group

Additional Questions?

→ If there's have any additional questions, please email kaguila@uccs.edu and we will email respond back within 1-5 business days