Faculty of
Engineering Ain
Shams University Credit hours
program

CSE 426 - Software Maintenance and Evolution

Assignment (1)

Study

Submitted to: Prof. Ayman Bahaa-Eldin
Submitted by: Mariam Hassan Nassar
ID: 17P6075

Application Overview

Anubis-IDE is an open-source desktop text editor that helps provide a simple integrated development environment to write, edit, run and compile python code on different microcontrollers.

Scope

1) **In-Scope:** Functional testing was performed for the following modules:
    - Scanning and parsing different python input tokens
    - Compilation of input code sequence
    - Running compiled code on microcontroller
    - Saving input code to new file
    - Opening & editing existing python file
    - Connecting to port & flushing code
2) **Out of Scope:** Performance and Stress testing were not done for this program
3) **Items not Tested:** Verification of external packages (QT5 & PySerial)

Program Installation

To successfully run program, user must set up the environment then clone the project repo. If the environment is already set up, user can skip part (A).

A. Environment Set Up

1) Install any text editor, preferably VS Code
2) Install Python, recommended version 3.0 or above
3) Add Python extension to VS Code

B. Anubis-IDE Installation

1) Clone project repo from this link: https://github.com/a1h2med/Anubis-IDE
2) Open project director using any text editor, preferably VS Code
3) Open terminal and run the following commands to install project dependencies:

    $ pip install -r requirements.txt

    Or

    $ pip install pyserial
    $ pip install PyQt5

4) Run “Anubis.py”

Test Analysis & Issues Found

1) Failed to perform main task of compiling and flushing code to microcontroller. Steps to achieve this scenario:
    a) Attach microcontroller to device
    b) Choose plugged microcontroller from ports menu
    c) Click “Run”
    d) Error message displayed: “Sorry, there is no attached compiler.”

2) App crashes when any file type other than .txt is opened. Steps to achieve this scenario:
    a) Select file using tree view or using finder
    b) App freezes for 1 second then abruptly crashes

3) “Requirements.txt” is the file that contains dependencies and includes some unused packages (e.g. iso8601, PyYAML).
4) Save button is enabled and usable even when text editor is empty.
5) Tab name does not change to chosen file name. Even after saving the new code file, it remains “tab1”.
6) Addition of a second tab is not possible
7) The code is unreadable and unstructured. It contains unused variables (e.g. vbox at line 152 Anubis.py). This makes code difficult to maintain or evolve and contradicts the FOSS guidelines.

Improvement Recommendations

1) Use external package to compile code.
2) Validate file extensions and disable opening any invalid file types.
3) Remove unneeded pancakes from “Requirements.txt”.
4) Check if file is empty before saving it.
5) Set tab name to selected file name.
6) Unify variable names and make them meaningful and descriptive.
7) Split code classes into different files to improve scalability.

Conclusion

This software has potential but its execution begs for improvement. The code needs refactoring. Some requirements need to be met. Design documents must be provided. The main goal is to achieve readability, maintainability, reliability and scalability.