Kyle Knotek

4/25/2023

CS 333

Final Project Design Document

For the final project, I'm choosing to use a Connect 4 game made in python. The game is quite simple, as it runs in the terminal and has no real GUI elements. I chose this because it uses the same testing methods that we have already established in class as it is coded in Python 3. I was originally considering using a web app made with html, CSS, and JavaScript but I thought that would take a considerable amount of time to learn and configure, as most testing with JavaScript relies on node.js configuration.

The program includes eight classes that can be unit tested because of their Five of these classes have arguments have passed into them and therefore can be integration tested. They can also be used as an edge case for invalid moves and invalid characters passed into the input. These tests should cover at least 75% coverage as the code is not too extensive for the program. For testing, I would implement the coverage.py package to determine testing coverage.

For centralizing, automating, and testing the code, I plan to use GitHub actions. I chose GitHub actions because we have already learned the base functionality for it from the class activity. I felt that using GitHub actions would be the most straightforward and intuitive way to implement an automated testing workflow. First, I would set up the unit tests and integrations test and upload them to the repository. GitHub would then set up a virtual testing environment using the latest version of ubuntu. The program would run the test script using python3 to determine if the test ran correctly. After tests have passed, the program would then deploy the program. The pipeline would follow the main steps:

For building and deploying, the actions can be put into GitHub actions to automate the building of python. This is done in a very similar fashion to the testing method. The program would be scripted to run a virtual ubuntu environment to run the python3 command in the terminal. This would then build and deploy the program and ensure its successful run. I chose this method again because we have used this in class and would be safer to use as opposed to learning a brand-new testing platform that might cause refactoring of the web app itself.