CS 1530 – SPRINT 2 DELIVERABLE

13 Oct 2016

David Bickford (drb56@pitt.edu)

Craig Kodman (cmk126@pitt.edu)

Joe Meszar (jwm54@pitt.edu)

David Tsui (dat83@pitt.edu)

Decision Descriptions

During this sprint, our focus was on having a working walking skeleton. We started by planning which libraries and API's we would use for our project. Our communication was done during stand-ups in Slack, in-person meetings and text messages. We also used a Kanban board to let us know what was in progress, what was in need of review, and what needed or had been tested.

Some of the problems we encountered were:

1. We could not get the program to run on both Windows and Mac machines. It was determined that we had to have two separate paths, and write code to check for which machine was being used in order to choose the correct path.
2. We all worked in the same branch instead of using our own separate branches which is not how we had set out to do. Going forward, we will stress the use of separate branches.
3. At first we had allocated researching the API's and libraries to one person. We then realized that this was causing others to be blocked from making progress on their portion of the work as this was time consuming. We then all spent time researching. Between the issue with getting the project to run on a Windows machine and this lengthy research, we lost valuable time and didn't leave enough to properly test what we wanted to test.

4. Our velocity suffered also due to having to do tasks unrelated to our user stories such as making a gitignore file, doing research and being blocked waiting for some tasks to be completed.
5. We found that testing javaFX functions to be difficult to do. We are still debating if this should be automated or done via manual testing.
6. The stockfish engine can't be directly integrated into the source. Instead, a wrapper needed to be created. We grabbed an example of a wrapper from 'http://rahular.com/stockfish-port-for-java/'.

Our team gets along really well and enjoys working with each other. Any disagreements on how to proceed, such as the aforementioned issue about who is to do the research, is solved via discussions in our standups or in-person meetings. On one hand, we feel like we did a lot of work this sprint to set us up well for the next sprint, but we also feel that we got away from doing TDD. Going forward, we are going to focus more on writing the tests first and having good test coverage for what we have already written.

User Stories Completed

18. As a User, I want the save-game action to be a button, so that I can click the button and save the current game.

-We decided to do this user story because we needed to have a button for save-game and we needed to have a test to make sure it worked.

4 points

19. As a User I want the load-game action to be a button, so that I can click the button and load a saved game.

-We decided to do this user story because we needed to have a button for load-game in the program and testing a button being pressed is quite simple.

4 points

Velocity: 8

Github Link

— https://github.com/drb56/CS1530 —

Defects Found

1. **Problem:** Inputting the file path into Stockfish using Windows machines

   Solution: We had to make two different file paths, one for Windows and one for Macs and then we had to detect which operating system the user was currently on and use the appropriate file path to start the Stockfish engine.

   Reproduction steps:

   Run the program on a Windows machine without the WIN_PATH variable in the Stockfish.java file.

   Expected behavior:

   We expect that on both a Mac and Windows machine when you run the program LaboonChess.java a chessboard pops up and on command line you see the output "Engine has started..".

   Observed behavior:

   The chessboard pops up, and on the command line, you see the output "Oops, I did it again..", meaning that the engine did not connect, because it didn't have the correct file path to the Stockfish engine.

Fortunately/Unfortunately, we haven't found very many defects in our program because during this sprint we focused mainly on getting the GUI to work and display and integrating the Stockfish engine into our program. We collaborated and decided that these features were the highest priority and the most important to implement first; furthermore, without these features we would be unable to complete many of the user stories.