Team Results Analysis

for

Project 1 – Chess Game

Version 1.0 approved

Prepared by Madison Kuyawa, William Widmer, Bunrith Seng, Robert (James) Castleberry, Scott Flolid

CS 3398

March 06, 201

\*\*\*Given instructions: “I would like a team driven analysis (what we call post-mortem) of the project. Two or three pages are fine. Please consider and present what challenges your team faced during this project. Also, describe any consensuses or disagreements your team had. For example, did you debate adding a feature? Did you have disagreements between testing and requirements? This is common, and it is good to be aware of them.”

Overall, our team worked quite well together. We believe we were able to follow the steps of the Waterfall method and grasp the concept of the Waterfall method. We were able to communicate regularly and easily. There were few disagreements, if any, among our team. We were able to overcome most challenges we faced, however the biggest challenge was the time constraints on this project that coincided with our other classes. In the end, the timing issue was one we could not overcome as we underestimated time allocation for each of the individual parts of the project and ultimately left us with little time to code.

As mentioned above, most of our team challenges were minor and the only challenge we could not overcome was meeting the deadline for the Chess Game. Our first minor challenge was finding a time to meet in-person. Being students with busy schedules and taking multiple classes led to this challenge. However, one teammate created a survey online to find the best time for each person and we were able to set up a weekly in-person meeting schedule. Another minor challenge was figuring out to split up the workload for each step of the Waterfall method and correctly understanding the requirements of each step. We were able to understand the requirements of each method and found it best to follow the steps in order of Waterfall as they were meant to be followed and to split up the workload pretty evenly. During our additional requirements everybody for the most part had really good input as to what sort of additional requirements our team could add. We discussed and argued a bit about which requirements would be best or even be big enough to be considered requirements. This lead to constructive arguments and us eventually all coming to agreement on our best possible options. The final and most difficult challenge for our team was the time constraint. This is due to the nature of Waterfall (not writing code until the end) and also each person’s duties to their other class workloads. We believe teams in the real world also have the challenge of time constraints and that is why the Waterfall method is not very likeable as compared to other methods like Agile.

Our team did have to go back to the SRS and change one feature later on. We did not disagree on the feature, but we changed the feature based on feedback from the professor. The feature we decided to change was originally: “An indicator arrow on the chessboard shall point to whichever player’s turn it is”. The feature was too simple, so we had to think of a more complicated feature. The new feature we came up with was “When a piece is captured the system shall place an icon of the piece under the corresponding player’s name”.

As for testing, our team did not have any disagreements. Our time constraint issue did not allow us to run any unit tests, and hence we did not find any major issues/bugs with our Chess Game.