Griffin Chess Team Analysis

This chess project was a very interesting program to complete with a group of team members. Working together and communicating was a big part of this project. Our team communicated effectively throughout the entire project. Everyone provided their own set of skills to help successfully provide full documentation and working code to create a well-rounded application with high-level design. Although each process was very tedious and took a lot of time, it certainly helped lay foundation and provided plenty of detail on the functionality and characteristics of what the project would become. The need for requirements, specifications, and design documentation is reasonable because of the necessity for organization and a blueprint that the team can always reference. Applying object-oriented principles to this project was not an easy task. The implementation of the individual components and the interactions between each piece was a very challenging task, even while understanding the conceptual side of the MVC. Regardless, this provided for a very serious learning experience that benefited everyone on the team.

However, there were many challenges faced throughout the project. One of the many challenges was the documentation for the SRS. The example of the SRS outline that was provided to us through TRACS was not as helpful as it could have been. There were some sections of the SRS that were not particularly clear or necessary. In some cases, some sections were not needed and the team was left to question how to write each section. This issue wasted time for our team. Nevertheless, our team collaborated and came up with an adequate SRS that accurately describes the characteristics and functionality of what Griffin Chess provides. The same issue also occurred with the SDD. The example SDD given did not provide enough information to us for what was necessary to write. However, we did find similar SDD documents to help guide our team through the process. We concluded to add sections to the SDD such as the Use Cases and Test Plan. The code implementation of the project itself was not particularly hard. The biggest challenge of this part of the project was implementing the algorithms of the chess pieces and the special moves.

Overall, the project elaborated on the detailed design process that a company would potentially utilize to create professional applications and software. This is something absolutely necessary to gain experience and practice with so that everybody is prepared and understanding of the design pattern if it shows up in the future. Indeed, this made the process worth learning and was certainly an enjoyable experience.