**Project Report**

Dhwanil Chokshi

Zack Salah

Boodhi Alshaikh

Bader Alshaya

The application created is a large scale data processing software that has functionality of handling multiple different types of users on the system, catering to each of their specific needs. Furthermore, there was a heavy use of text files to save information and produce forms for ChocAn related purposes. As the project is being wrapped up by the developing teams, there were several elements in this project that went well and several elements that could have been improved to facilitate an easy path to the finish line.

One of the first things that went well was the very existence of this ChocAn project. The software models we learned during class were being put to practice when we were working on the different components of the project. The class was not intended to create an environment, where students come to class, take notes, and leave. The class was intended so the students could apply all the materials being taught in class and put theory to practice, and it very well accomplished this objective. Even though the focus of the ChocAn project was to adapt to the Waterfall Model procedures, We feel like we were able to learn about other types of development models (Agile, Spiral, etc.) and use them to compare the benefits and tradeoffs with Waterfall. The inclusion of this project further improved our understanding of the underlying benefits and drawbacks of Waterfall as we were able to immerse ourselves in the development work. Additionally, the minimal inclusion of the teacher and outside factors in this project left the burden of accomplishing each part on the team’s shoulders, just like how it would be in the industry. The important missions achieved by ChocAn project is mirroring some of the industry practices and learning about the importance of each step in the process of documentation.

Another element that went well with this project is working in team. We feel that this was a huge step in helping us learn about the procedures and routine of workplace after we have graduated. In the lower division classes, all of the programming work that was done was individual with little involvement of group work. The individual work is limiting because the only ideas that are present are our own. However, with this project, the group work meant that there would be an involvement of multiple ideas. These ideas by the group members would persist during each phase of the model because requirements specification, design analysis, test planning, and coding were all parts which involved team work. Though there are agreements and disagreements between team members with regards to some concepts, we feel that project was able to further our understanding and open our minds to accept multiple ideas on multiple different topics. This ability to share ideas and work closely helped us in creating a more successful product in the end. If a group member was stuck, another person was able to help them without having to stall the progress of the project since all group members are assigned different parts. Also, there was a greater sense of accountability. Since team members were responsible for finishing their parts to the best of their abilities, it led to motivating effort to put full effort because everyone was aware that they were responsible for their part.

The evenly distributed tasks of the ChocAn project helped us work with efficiency that would not be present if only one person were doing it. The teams had to take responsibility and split their parts among the group members. We were able to quickly reach a compromise on which member would accomplish which part. The distribution of work was done evenly, leaving little room for arguments and angry reactions. This helped us achieve greater efficiency in finishing the overall project at a faster pace than working alone. Sometimes, a group member was assigned a part which could be finished pretty quickly, in which case he would go and help another team member in completing their parts. The team effort was paramount in creating the software which we were assigned.

It is important to note that there were a lot of things that went well with the project. However with any major task, there are bound to be certain sets of pros and cons. Similarly there were certain elements of this project that did not go well for the team. One of the most prevalent element was task dependency. The project had a lot of interconnected parts, with major components that were dependent on one another. This made it hard for one person to work on their part if another person (who’s part they were dependent on) had not finished. We feel that this cost us in the overall efficiency of designing the project because there were a lot of task dependency stretching all the way from the initial period of requirement specification to actually coding the software. An example would be that the terminal part of the ChocAn was the “meat” of the software. However, the team members who were responsible for creating the functionality of the components that were included in terminal had to finish their part before the members responsible for terminal could start working. This had a great impact on the efficiency because not everyone was able to put their nose to the grindstone and start working as soon as their part was assigned. A suggestion for improving on this element is to really sit down and understand all the interconnected layers of the software before diving to the code. Working in teams will likely always mean that there will be dependencies, making it important to recognize those dependencies and working around them until you have access to all the parts you need.

Another aspect that did not go well was learning and adapting to the different aspect of version control; github. All members of the team were inexperienced in using all features of github to commit and push code. Therefore, there was a big learning curve in educating ourselves with git as we were coding so we could use github to our advantage in helping the team collaborate during the programming process. This was another element that put a slash through our group efficiency. As many part of this project were dependent on other parts, it was important for group members to know github to have a fast paced environment where everyone has access to the code they need. A suggestion on improving this element is to learn a platform for version control and code sharing like github, so there is no hindrance in the overall efficiency of code production from the team.

Lastly the main problem our team had was staying consistent with the design we had created. An important feature in Waterfall development is to finish all the documentations thoroughly before going on to the next step. Going back to a previous step is very costly because going back and changing the requirements specification would also mean changing the design. This domino effect ensures that each part of the phase is completed thoroughly before moving on to the next. However, we found it hard to stick to the script. There were many instances in which we found it difficult to adhere very strictly to the process that had been outlined in the design document because there was a more efficient or a better way to do it which had just been discovered. Also, during many instances we were going back to look at the project description to make sure that we were covering everything that was needed. An important suggestion to avoid the mistakes we made is to create a very strong and neatly typed requirements specification document that covers all aspects of what is needed. This way, there is no necessity to look back at the project description, when we could easily look back at the requirements specification and see what needs to be done.