Group member Contributions Report

Managers:

We have kept the same positions of each manager as follows:

-- Customer Interface Manager - Callum Dicker

-- Design Manager - Daniel Kelleher

-- Implementation Manager - Kristoffer Page and Yiwei Li

-- Test Manager - Thomas Vass and Shannon Gahring

-- Planning and Quality Manager - Adam Yaziji

-- Doxygen Manager - Jones Lo

Member's Contributions:

Callum Dicker:

When we started this assignment, I was at first elected as project manager as well as keeping my current role as the customer interface manager. As the project manager, I have successfully headed many meetings and made sure that during each meeting every group member had work to do on either that day or by the next meeting. In each meeting, I have ensured that each group members explains what they are currently doing and what their progress is. As Customer interface manager, I have made sure that the specification was being followed at all times and I have always been with team members when questions needed to be asked either about the program or about the code whether this has been with each other or to Bob. During the actual implementation, I was first, put into a pair to sort out the saving, loading and pausing of the game. Daniel Kelleher eventually took over saving and loading fully, I have tried to assist in the saving and loading whenever I can or, asked for help. I fully implemented the pause function of the game as well as changing the menu so it was able to handle the additional functionality such as the AI, and loading. During the course of the project, I decided to implement the Connect Four animation to lessen the workload of another member. I also recorded and edited the screen capture demo of our application running; this shows all the additional features we implemented.

Daniel Kelleher:

At the Start of this assignment we delegated the roles of different tasks to each member and I took on the save and load function with Callum. I researched into the topic of the serialisation method stated by the group we acquired the application from and I eventually found that this would not work correctly in the application so we decided to go with the file format of ‘csv’. This worked very well with our system there were a few bumps in the road but these were overcome and the system of saving and loading now works to a high standard. As Callum was reassigned to the graphical side of connect four I took on save and load mostly, but I had help from Callum at times when some errors occurred. I edited the Game GUI so the user had a menu bar to select the different functions, also so the GUI could take the file names from the user as input. I created the saver class which took in all the necessary data to save a game to file. I used the csv file format to save all my data in rows and columns. I then crated the loader screen class, which allows the user to view all saved game files and select one to load. I finally created the loader class, which read the selected file, extracted all the data, and passed this in to create a game from the loaded data. I also research all of the doxygen set up so to produce a very professionally looking set of doxygen web pages with graphical images for the hierarchies, collaboration diagrams, and call and caller graphs. I also made a first draft of the group report and then edited it alongside Adam.

Kristoffer Page:

I was assigned the responsibilities of implementing the easy and hard computer player for Connect Four. I implemented 3 classes, abstract class ComputerPlayer, EasyConnect4ComputerPlayer and HardConnect4ComputerPlayer. I modified Connect4Game and Connect4Board to be to play with AI players. As implementation manager I had to ensure that the coding conventions were followed and the code was readable.

I've also helped with debugging and fixing the showing of the winning pieces in connect four and other general problems.

Yiwei Li:

For A5 I was assigned a task to show the winning pieces for Othello. However, because their code was implemented in a very different way compared to our previous implementation. During implementation, I came across some difficulty; however, this was overcome by loading the image. Therefore, I drew two images, a black piece with a yellow star and a white piece with star. After completing my task, I assisted Jones in his Connect Four winning piece. My team role is the implementation manager so I have gone through each class code so it does not violate Bob's concise code conventions.

Thomas Vass:

At the start of the assignment I was assigned the responsibility of implementing both the easy and hard Othello ai. This was completely to the deadline I promised (the Monday before the deadline). This was to leave plenty of time for commenting, testing and other basic admin requirements. During the assignment I helped on displaying the valid moves in Othello (I came up with the logic for this, as it was very similar to how my computer ai works). The Thursday before the deadline I was also approached by Shannon who was having trouble animating moves in othello. After suggesting to her various methods of how to go about this and none of them were working it was decided that I will take this over (with less than 36hours till deadline). Within this timeframe I was able to create several methods; one to find all the pieces that flipped, two for the different types of flips (white to black, and black to white), as well as some basic implementation of this. However due to the time constraint and the difficulty of the job at hand currently the program will only flip one of the human player's pieces but all of the computer player's pieces.

Shannon Gahring:

On A5 I was responsible for doing the animation in the Othello game. Since the groups code before us used Graphics2D instead of Swing, which is what we used in our previous assignment, it was much more difficult. Their names for methods and variables were also very unclear. An example was flip. I would assume that this would flip or change a piece where it actually just switched players. Finding out how they got their chain of pieces to flip was quite difficult to understand. In the process of doing the animation, I tried a multitude of ways to try and get it to flip. One of them was to change the size of the oval to a width of 0 and then bring it back to normal, just as the changed colour. This seemed to be the easiest way to go about it. Throughout this assignment I went through and did integration testing. This was just going through all the methods and making sure that they all worked properly and gave the correct and expected output, Tom and I both worked on this.

Adam Yaziji:

I was assigned the task to create feedback for the player if the current move they had selected was invalid for both Connect Four and Othello games. I implemented this by showing a dialogue box, using JOptionPane.showMessageDialog, every time they clicked on a position on the board that was not valid. I have helped Tom with showing all the valid moves available to the player. We achieved this by creating an array list with all the available valid places and changing the colour at that position to another colour (Yellow). I have also given help to Jones with showing the winning pieces on the board with Connect Four. I have checked over the requirement document to make sure the group was implementing the required functions for the assignment, just in case we were implementing something that was not needed. I have edited the contributions report and added in everything each group member would like to declare in it. As we have decided to keep the roles of each member the same as the previous assignment (A4), I am again the Planning and Quality Manager. This required me to take minutes of each meeting we had and make too sure we are submitting files in the right format. We had a total of seven minutes each of which consist of; who attended the meeting, the date of current meeting, start and finish time of current meeting, the date and start time of next meeting, topics discussed/done during the current meeting, progress made since previous meeting and what each member has been assigned to do by the next meeting.

Jones Lo:

I modified the class of Connect4Game and Connect4Board to implement the winning pieces being displayed for the player who won. In those, class I did make some method out and try to use if else statement to do the check wining piece out. Also I did print out some statement to check whether it compile or not. If it is not compiled correctly that's mean there is an error on the code that I am inputting. My group helped me when I was stuck during the coding process, we try to fix the error together and while doing that I may also improve my knowledge of coding and the skill can be improved as well. I was the doxygen manager; in this role I need to look at all the class of both games (Connect 4 and Othello) to check is there any comments error. When there is error I need to tell to my group mate and ask them to correct it. We are following the guide of the doxygen that Bob's hand out. Inside that handout is show very detail which statement comments is wrong or not.

Manager Report:

Customer Interface Manager - Callum:

1. During the implementation phase, we have ensured that we are implementing each request given by the customer.
2. We have followed the specification given to us in order to develop the additional features for the multiple changes, the specification given to us for development was very clear and allowed us, as a group to create the software in an effective manner. The Requirements documentation given to us by the other group had to be changed in certain area, this includes the save function which was noted in the requirements documentation to be XML, but due to the time limit and functionality, we instead changed this choice to save the files as a CSV file.
3. I have asked the group multiple times during the course of the implementation phase to ensure that they read and understood the requirements of the program. If they had any issue, then they could come to me, which then I would either try my best to help them, assign another more compatible group member or, if came a situation in which there was no clear solution then I would direct any questions they had to our customer which in this scenario was Bob.

Design Manager - Daniel:

1. Yes, all our members are adequate in the use of design methods; we will have a run through of the presentation for our software application several times to make sure that each group member understands the structure we will follow and what part they will present.
2. Each member's classes are of a high quality conforming to the design of the requirements document. I personally checked over each class to ensure this, this was also a part of my role in the group.
3. This application was handed to us and it took a long time to work out how it worked but we finally understood the classes and the system as a whole just enough to add all the other functionality to the game application,
4. Every class in this game system has its own comments and doxygen pages giving an in-detailed explanation of everything in it, this being methods or variables, and how these are used within the class itself and the application as a whole.
5. Each of our team members has the following responsibilities:
Callum – Game Chooser menu improvements.
-- Connect Four game graphics and animation.
-- Implementation of a timer in both games.
-- Ability to pause the game.
-- Collaborated with Daniel on the save and load function.
Daniel – Ability to save either game at any point to a file of csv format.
-- Ability to load any game from file and display it on the screen for the user to play.
-- Add the functionality of playing a new game.
-- Add the functionally of all the features to the game menus and the main menu screen.
Kris – Add the functionality of an easy computer player to play against the user in the connect four game.
– Add the functionality of a hard computer player to play against the user in the connect four game.
Tom – Add the functionality of an easy computer player to play against the user in the othello game.
– Add the functionality of a hard computer player to play against the user in the othello game.
-- Add the animation of flipping pieces in Othello game.
-- Add the graphical view of all available valid moves to the user in Othello game.
Shannon – Collaborated with Tom to add the animation of flipping pieces in Othello game.
Jones -- Display all winning piece in the Connect four game.
Yiwei – Display all winning piece in the Othello game.
Adam -- Collaborated with Tom to add the valid moves display.
-- Add the display invalid moves feedback in both Connect Four and Othello games

Each of our team members edited the following classes:
Callum – Edited Game Chooser, Game GUI, Connect Four Game.
Daniel – Added Save, Load Screen and Load classes.
– Edited Game Chooser, Game GUI.
Kris – Added Easy and Hard Computer player classes for Connect Four game.
Tom – Added Easy and Hard Computer player classes for Othello game.

-- Edited Othello Game, Othello Board.
Shannon – Othello Game.
Yiwei – Othello Game.
Jones – Connect Four Game, Connect Four Board.

Implementation Manager - Kristoffer and Yiwei:

1. All members have knowledge of Java from previous modules/experience. Every group member researches on the internet or o in the reference book.
2. The design documentation describes everything in clear detail. It states the responsibilities and collaborations of each class, and what method we should use for the program, so everyone should be very clear about what they doing.
3. All classes have been checked to ensure each class completely follows the coding conventions.
4. Classes are using the same code from the super class and inheriting method where it is possible to use

Test Manager - Shannon and Thomas:

1. Tests were done throughout each individual's implementation. Having them done throughout the process helps to avoid large errors and makes integration testing much easier.
2. Yes, all classes do have unit test defined. This helps each individual rule out errors. Not all methods have unit tests. There are certain ones that are very straight forward and logical and do not need to be tested.
3. There are some sub-system tests defined
4. The test plans are complete and fairly thorough. There can always be room for more thorough and in depth tests, but what we have, gives us a very good indication into what works and what does not within the application.
5. Yes, each team member does understand how to produce basic tests. If anyone was unsure on how to produce tests other members in the group gave them a hand.
6. Our team's integration test strategy was to go through and thoroughly test all of the methods in the classes and make sure that they all worked properly together. This made sure that once one class was combined with another that there were not any little errors that we had missed when doing unit testing.

Planning and Quality Manager - Adam:

1. Yes, I believe the team has met up often enough. We have a total of seven minutes. The team has met up three times before the week of the deadline and four times during the week of the deadline. Most meetings took place in the Faraday labs and usually consisted of 2 hours or over.
2. Yes, the team is following the minute's protocol. In the minutes we included; who attended the meeting, the date of current meeting, start and finish time of current meeting, the date and start time of next meeting, topics discussed/done during the current meeting, progress made since previous meeting and what each member has been assigned to do by the next meeting. We are using last assignments (A4) minutes as a template.
3. The team has used version control. We have decided to only use dropbox for this assignment instead of using GitHub as we were having problems with GitHub.
4. Yes, code inspections are being carried out. Kris and Yiewi are still both the implementation managers and are carrying out the code inspections thoroughly.
5. Yes, implementation is properly documented using Doxygen. Our Doxygen manager, Jones, will be using the same software as he used in the previous assignment (A4) to produce doxygen for this assignment.
6. Yes, all classes and sub-systems meets the team's quality standards. The team has high quality standards and I, being the Planning and Quality manager, have checked with each Implementation manager asking whether the coding conventions of each class meet the teams quality standards.
7. In the case of member A not contributing, we as a team, will politely ask them if they are having trouble with any aspects of the agreed upon tasks they have been dealt. If member A continues to fail producing completed work then we will confront them and ask them why they are not completing the agreed upon tasks. If for a third time, member a still fails to produce any work, we will file an anonymous contribution report to Bob. Contributions of each member will also be clearly stated in the contributions report.

Doxygen Manager - Jones:

1. Yes, the classes are implemented properly and documented using the doxygen software, which clearly explains what each class is, how it fits in the hierarchy and what comments it contains. This is shown very clearly in the HTML output.
2. We reused the downloaded software called "GraphViz" as this created the graph images and was a very useful tool.

Features:

Game system features implemented and functional:

1. Connect Four AI (Easy and Hard).

2. Othello AI (Easy and Hard).

3. Save a game.

4. Load a game.

5. Pause the game.

6. Partial flipping of Othello Pieces.

7. Show winning pieces for Connect Four.

8. Show winning pieces for Othello.

9. Updated the menu screen to include a load function and the AI selection.

10. Show invalid move message for Connect Four.

11. Show invalid move message for Othello.

12. Highlight all valid moves available to the current player in Othello.

13. Animation of Connect Four piece dropping down into bottom position.

14. Show how many pieces each player currently has on the board.

15. Implement a back button to Game Chooser screen from the input screen.

16. Automatically skip turn if there are no current valid moves for Othello.

17. Fixed high CPU usage due to paint component.

18. Implemented a timer for both Connect Four and Othello games.

Game system features non-implemented:

1. We have implemented all features that were required for the assignments A4 and A5.

Results of the project:

Expected:

1. The delegation of tasks made it a lot easier to get the project finished on time.

2. The implementation of the AIs runs and plays against the player very efficiently.

3. The redesign of the option and input screen was done to a high standard.

4. The outcome of the doxygen pages looked very professional and well laid-out with all the necessary diagrams, descriptions, code snippets and hyperlinks.

5. The demo video demonstrated all the functionality of the application and was of a high quality.

Unexpected:

1. Initially, we decided to create a save function using marshalling of the game objects to an XML file, serialisation, but we then found that this was far too complicated, time consuming and too difficult to get to work so we decided to create the save function using CSV file.

2. The readability of the given code was hard to understand and work with by all members of this team.

3. The difficulties arising from adding functionality and animation to the existing system was more difficult than expected. This was mainly the Othello filliping pieces were not properly implemented.

4. The design of the given code was very strange, as there seemed to be a system of reverse abstraction where they took two game classes and gave them both to one GUI to be displayed. This was difficult to work with.

5. The implementation of the AIs into the system took more work.

6. The merging of all the different copies of classes and code snippets took a lot of organising and time.

Minutes of Meeting:

This is the link to download the minutes of meeting from DropBox, our file share service, we used to share files between our team. Link: https://www.dropbox.com/sh/8yx0kj6js0383wp/Lezh9OenP0

Produce Application:

To produce this application from the source code the user must compiling the code in the following steps:

1. Take the 'source' folder and compile all the .java files in this folder.

2. Then run the Game Chooser file to run the game system application.

Java Programming reference Books:

These are the resources we used to reference coding help and tip.

[1].'JAVA for everyone', Cay Horstmann, Second edition, 31st January 2012, John Wiley & Sons.

-- Chapters used:(6.1)Arrays, (6.4) Using Arrays with Methods, (6.8) Array Lists, (7.1) Reading and Writing Text Files, (8)Objects and classes (9.2) Implementing Sub Classes, (9.3) Overriding Methods, (10)Graphical User Interface.

[2]. Holmes, Barry. Programming With Java. Sudbury, Mass.: Jones and Bartlett Publishers, 1998.

-- Chapters Used: All

[3]. Website used for researching information. Link: http://docs.oracle.com/javase/tutorial/