Table of Contents

NAH Group 2 Contributions Report

NAH group 2 consists of the following members:

GAVIN BAILEY
GRANT BELSEY
TYRONE LEE BRAMWELL
CHAK YAN LAM
THOMAS LETHEBY
MATHEW JAMES LLOYD
JAKE DARYL PLUMLEY
CHUN KIT SO

Like all previous group assignments we have had no contact from Grant Belsey. He had no contribution to any elements of this assignment nor did he attend any of the meetings that we arranged. The meeting dates, times and locations were all put up on the Facebook group, which we created for the group tasks, of which Grant Belsey is a member, so he would have received notifications of such arrangements.

As a team we believe that each member has contributed equally to this project, from the previous assignment where we designed this application, we had already assigned who was to implemented each part of it, so we already had a good idea of what was need to be done. However upon receiving this assignment we learnt that it was to be a partial implementation of the application, which meant we had to reassign the classes of the project to team members who's classes were not need for the application in its current form.

We all began to implement our classes and good progress was made by all, but due to misinterpretation and lack of communication, some classes were not implemented correctly. The classes did the jobs they were required to do and more, which was the problem. Some classes had methods within them that could have been put up the class hierarchy, so that they were made available to more classes without duplication. Due to this a major restructure was needed to make our application more efficient. The classes were split into a number of smaller more reusable classes for ease of reading and portability.

After the restructure the project became easier to handle and communication between all team members became more frequent, to prevent this problem from occurring again. With the classes being split there were more classes available for team members to work on, so two classes were assigned to each team member to complete. After this was done the production speed of the project greatly increased, leading to a smooth ending to the project.

As a whole the team worked well together later in the project, after the major restructure. We felt that all team members worked hard on this project, offering an equal contribution.

The team did well to overcome the problems we encountered during this project and no major disagreements occurred. If anything the mid project restructure helped the team to work better with one another, to get to the final goal.

Features

The application that the team has built includes the following features:

- The ability to select between the two games, Othello or Connect Four.
- The ability to create two human players.
- The users are able to play a game of connect four to completion, with a winner declared at the end.
- The users are able play a game of Othello to completion, with a winner being declared at the end.
- The users are able to start a new game at any point during a game.
- Information is provided to the player indicating:
    - What colour piece they are.
    - Which players turn it currently is
    - Score (Othello only)
- Users are able to select a new game when a game has completed.
- Users are able to pass their move to the next player when no moves are available to them (Othello only)

As a whole the partial implementation of this project is good, we believe that the program flow and design has been implemented very well. However we also feel that the area that is lacking in this project is the lack of animation to the game pieces. It would have been better to get the connectFour pieces to slide down the board and to get the Othello pieces to do an animated flip, but as it is said this is the partial implementation of the projects final form.

References

The group used many sources of information to help with this project; this was mainly used for the testing and GUI elements of the application. The main source of information was the Java documents website, many pages were used for so we could check if we were using an object correctly, or if there was another that would be more suitable.
http://docs.oracle.com/javase/tutorial/index.html
http://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html
http://docs.oracle.com/javase/tutorial/uiswing/layout/layoutlist.html
http://docs.oracle.com/javase/tutorial/uiswing/layout/gridbag.html
http://docs.oracle.com/javase/7/docs/api/java/awt/GridBagLayout.html
http://www.macs.hw.ac.uk/guidebook/?name=Layouts&page=7

This url was used to ensure we had the correct approach to JUnit testing
http://ptgmedia.pearsoncmg.com/images/013143697X/downloads/013143697X_book.pdf

Reference books used:

Used for information on packages:
Flanagan, D. (2002). Java in a nutshell: A desktop quick reference (4th ed.). Beijing ; Sebastopol, CA: O'Reilly. p81 - 82.

For information on events:
Boone, B., & Mark, D. (1996). Learn Java on the Macintosh. Reading, Mass: Addison-Wesley Developers Press. p235-238

Where to download our project from

Cloning our git repository onto your machine can download our project. The URL for our repository is: https://github.com/NAH2/cs235groupNAH2.git

Compiling our code

Some of our classes are put into packages so they will need to compile as a package or compile errors will occur. The piece package also contains images in order for it to draw the pieces onto the board. Packages are as follows:

boardGame package
BoardGame.java
Othello.java
ConnectFour.java
red.png
yellow.png
white.png
black.png

piece package
GamePiece.java
Othello.java
ConnectFourpiece.java

player package
Player.java
HumanPlayer.java

Other classes and resources are contained in our project but are not in packages, theses are:

GameController.java
GUI.java
Connect4GUI.java
OthelloGUI.java
SelectGame.java
Selection.java
empty.png
icon.png

Import all packages, classes and resources into a project using Netbeans or Eclipse to compile and run the program.

Individual Team Member Reports

Chun Kit So – Implementation Manager

In Assignment 4, I have implemented Othello class and BoardGame class. I also implemented Othello JUnit test and OthelloGUI JUnit test, make sure every method is working correctly. As an Implementation Manager, I ensured that everyone is clear on our A3 assignment design before start the implementation and every class is following the coding conventions. Moreover, I commented the Othello class for Doxygen and make sure it generated a document.

Jake Daryl Plumley – Customer Interface Manager

In the assignment I implemented two classes that were OthelloGUI and GUI. I also implemented JUnit testing for both of these classes. Whilst writing the code for these classes I ensured that I was following the coding conventions after discussing them in our group meeting. I also commented the code ready for Doxygen. I was elected customer interface manager so I communicated with the customer if the team had any questions that they wanted to raise. As customer interface manager I also ensured that all team members understood what is required by the customer to suit their requests. To ensure that the guide to development is sufficient I checked that our requirements documentation was to a high standard. I attended all team meetings and along with Thomas Letheby we recorded the group minutes for the majority of the meetings.

Tyrone Bramwell – Testing Manager

In the assignment I worked on the SelectGame and the ConnectFourPeice java files. I also did the testing on these files. I also created the demonstration video using U2Any Screen Recorder Free software from the apple app store this was because the software for mac recommend on Robert S. Laramee website left a watermark in the center of the video. I was also test manager for the assignment. The test plans where produced before the class where made so we know what data should be input into them and the output. The test plan where then modified to meet the class as the way we thought class where going to work changed. All classes have unit tests defined for them. Each team member understands JUnit. We need to learn how to implement tests for GUI objects on the JFrame. The team's integration strategy is to print out the start and ends of methods. As each class runs we can then see through print statements when a method calls another method from another class.

NAH2 Group 2 Contributions Report

Thomas Letheby – Doxygen Manager

In the assignment I helped the team by collaborating with Jake Plumley to create the minutes for the meetings we held, I created the Selection.java class, with guidance from Mathew Lloyd on how to set up the methods etc. I was also in charge of creating the GameController class. I also looked through all completed classes ensuring there were no errors in the code, and if so correcting them. I set up testing conditions in my classes and made sure I commented them correctly for use, ready for Doxygen. I looked over my classes and made sure that they met the coding conventions set by the customer. I was also the Doxygen manager so I helped to implement the commenting in most classes, and help others who didn't know quite how to comment the code correctly, I then generated the Doxygen documents ready for our customer.

Gavin Bailey – Design Manager

In this assignment I was elected to be the design manager where I ensured that the team were following our design. However I did feel that the design had to deviate a little as the classes were becoming too big and too complicated, so I encouraged the team to split classes so that they can be reused by others e.g due to lack of communication and misinterpretation amongst the team the Othello class began to implement the game board and gui. So I stressed that they needed to be three separate classes so that connect four could use them also. I also fell into the role of team leader along with Mathew Lloyd, where we worked together to assign tasks to each team member to complete the project. The classes I was assigned to implement were GamePiece and OthelloPiece, I also added the Doxygen comments and implemented the tests for these classes to ensure that the work as designed.

Chak Yan Lam – Implementation Manager

In this Assignment, I have implemented two classes, ConnectFour and Connect4GUI. I also did the testing for these two classes to make sure also the methods are working and follow the code convention. I have commented every single method in the class. My role in the team is being one of the implementation managers, I have reminded my teammates to follow the coding conventions while implementing the classes. I suggest ideas for making the code reusable.

Mathew Lloyd –Planning and Quality Manager

In the Assignment 4 I was to be the Planning and Quality Manager. As part of my assigned role I was responsible for planning the teams meetings, the way in which we reported these meetings and the planning of the Assignment in general. I kept up with these responsibilities that have helped to ensure that the team has stayed on track to complete the assignment on time and with the quality of which is to be expected. At first the classes and sub-systems did not meet the quality standards that were set until gradually I had them changed to fit with the quality required. I delegated the Doxygen management to Tom Letheby who completed the task to an acceptable standard. I also eventually fell into the role of team leader along with Gavin Bailey at which we worked together to assign the tasks to each team member to make the assignment process flow better. At first there was problems with trying to effectively get people to work on the project due to other coursework commitments but I

managed to work around this by only requesting small amounts of work to be completed until after the before mentioned deadline.

The team in the end did meet enough to keep communication going, at the start there were communication errors, of which have been documented above in this report. The minute's protocol was followed and the minutes of meeting were produced after each meeting. I in the end forced the principle of version control onto the team and so we setup a GitHub repository system of which in the end I think all team members were grateful for, especially when some people started editing classes at the same time as others, which meant the rollback feature became incredibly useful. Code Inspections were carried out at regular intervals and was useful in identifying that coding conventions weren't being followed and that there were problems with the project setup.

Other than general management duties I was involved with several classes, my person classes were Player and HumanPlayer, I also have provided Unit testing to these classes. I helped others with their classes and was always available to help upon request. I also implemented a lot of the testing and checked over other peoples code to ensure that they were following the 'Bob's Coding Conventions'.