BSC – HGP – Project Go

UI Design Document & Report

# Division of Work

Student Name1: Ryan Talbot Student Number1: 3001508

Student Name2: Conor Mc Cabe Student Number2: 3009601

Student Name3: Osebor Evbodaghe Student Number3: 3016732

Please complete the sections below with regard to the estimate of the division of work between the two partners

If the work was split in the range of 45% to 55% per partner, then that is fine and simply say “Work was evenly divided”. If this was not the case, then state with a summary sentence. This is the important statement of this file.

Division of work: work was evenly divided \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Code repository log (if applicable)

N/a

## Percentage of work completed by each partner on each class / task

Some areas require more work than others so this is only for reference. An average of these values will not be calculated.

|  |  |  |  |
| --- | --- | --- | --- |
| **Filename / Task** | **Ryan Talbot** | **Conor Mc Cabe** | **Osebor Evbodaghe** |
| GoBoard | 30% | 40% | 30% |
| Game Logic | 5% | 5% | 90% |
| Score Board | 25% | 50% | 50% |
| System design | 40% | 30% | 30% |
| How To | 50% | 25% | 25% |
| Learning rules of Go | 33% | 33% | 33% |

# UI Design

**Submission:** Edit this template and submit it as part of your submission.

**Length**: Should be 4 pages approx. Word count is flexible, but all decisions should be clarified.

To achieve good marks in this item ensure that this document is well structured and addresses each of the following headings and subheading. The explanation of each UI Design Choice should be clear, precise and show substantial consideration has been made, references are welcome. All decisions should be explained regardless of how basic they are. Do not cut and paste justification from the internet (plagiarism) or notes but include references and explanations in your own words where appropriate.

**Student Names:** Ryan Talbot, Conor Mc Cabe, Osebor Evbodaghe

* Include multiple screen shots of the application each focusing on a different component clearly labelled
* Clearly indicate what is working and what is not.
* Discuss each component under the following headings
* Location: e.g. The button was placed in the bottom right to as it is the default location to confirm and action
* Colour: The colour scheme was chosen to avoid the main form of colour blindness and produce high contrast for the visually impaired.
* Size:
* Style:
* Etc.

**N.B.** Clearly mention any additional features here either visual or functional so that appropriate marks are awarded

# Screen Shots of Working/Not Working Features

**N.B. Be sure to comment what is working and not working for each of the tasks. The boxes should be expanded to contain the content.**

All code should be testable where possible and error message should be displayed to show where code has failed.

|  |
| --- |
| **Task 1 (1 image with description + what is working/not working)** |
| Background pattern  Description automatically generated  Task 1) 7x7 Go game board presented  We wanted to use high contrasting colours for the tiles for accessibility. |

|  |
| --- |
| **Task 2 (6 images of working Menus/buttons/Labels including description + what is working/not working)** |
| Graphical user interface, application  Description automatically generated  Task 2a) A pop up before the game begins which shows a quick intro of how to play  The goal here was to have an quick run through of the rules without being too overwhelming, the user can simply close the box and start the game.  Graphical user interface, application  Description automatically generated  Task 2a) A menu option (CTRL + R) which shows the full rules of Go for reference at any time.  This is linked to the last image, should the user wish to revisit the rules they can do.  A picture containing chart  Description automatically generated  Task 2b) Show how many prisoners each player has taken.  We wanted a dedicated section to keep score, it doesn’t need too much space so we tucked the scoring between two buttons.  Task 2c) Show how much territory each player has taken – NOT WORKING  Chart  Description automatically generated with medium confidence  Task 2d) Show who’s turn it is  Simple confirmation of who’s turn it is, this switches each turn.  A picture containing text  Description automatically generated  Task 2e) Allow a player to pass  A button to allow the user to skip their turn. This button is spaced between both the start and reset button to minimize accidental presses.  Icon  Description automatically generated with low confidence  Timeline  Description automatically generated with low confidence  Task 2f) Allow the game to be reset  Users can reset the game if they wish, they will however need to confirm in-case the press was accidental. |

|  |
| --- |
| **Task 3 (2 images + what is working/not working)** |
| Icon  Description automatically generated  Task 3) Implement placement of stones using mouse clicks |

|  |
| --- |
| **Task 4 (2 images + what is working/not working)** |
| Icon  Description automatically generated with medium confidence  Task 4) Implement placement of stones in valid locations only – Suicide rule  For invalid moves we wanted a somewhat intrusive alert to really highlight an illegal move. |

|  |
| --- |
| **Task 5 (2 images + what is working/not working)** |
| Graphical user interface, application  Description automatically generated  Task 5) Implement placement of stones in valid locations only – KO rule |

|  |
| --- |
| **Task 6 (2 images + what is working/not working)** |
| Background pattern  Description automatically generated with low confidence  Task 6) Implement capture of single stones – NOT WORKING |

|  |
| --- |
| **Task 7 (2 images + what is working/not working)** |
| Icon  Description automatically generated  Task 7) Implement capture of multiple stones – NOT WORKING |

|  |
| --- |
| **Task 8 (2 images + what is working/not working)** |
| Text  Description automatically generated with medium confidence  Task 8) Winner detection via skip turn button – NOT WORKING |

|  |
| --- |
| **Task 9 (2 images + what is working/not working)** |
| A picture containing graphical user interface  Description automatically generated  Task 9) 2-minute timer for each move per player. Timer starts when game does and resets when a move is made for the next player. Game ends when timer runs out. |

|  |
| --- |
| **Task 10 (2 images + what is working/not working)** |
| A picture containing chart  Description automatically generated  Full Board)  For the board we wanted to choose tile colours that were both contrasting and still worked well with black and white stones.  To the right we have a panel which includes all current game information (i.e., time, score). In this panel we wanted to keep buttons spaced while keeping the same style for each. |

|  |
| --- |
| **Task 11 (2 images + what is working/not working)** |
| Task 11) The ability to undo/redo moves – NOT WORKING |

|  |
| --- |
| **Task 12 (2 images + what is working/not working)** |
| Task 12) Animation of pieces – NOT WORKING |