Assignment

Stage One Submission

2805ICT/3815ICT/7805ICT

Group Number: 2

Student name: Derek Qiu Student ID: s5348593

Student name: Student ID:

Student name: Student ID:

# Table of Contents

Table of Contents 2

1.0 Project Planning and Documentation 3

1.1 Time Schedule 3

1.2 Total working hours 3

1.3 Effort and contribution table 3

1.4 Version Control System 4

2.0 Requirements Analysis 4

2.1 Functional requirements 4

2.2 Non-functional requirements 4

2.3 Use case diagram 4

2.4 Full use case description 4

2.5 Activity diagram 4

3.0 Video link 4

# 1.0 Project Planning and Documentation

## 1.1 Time Schedule

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Task | | Plan | | | | Actual | | |
| # | Task Name | Student | Planned Time | Cumulative Time | Finished Date | Time | Cumulative Time | Finished Data |
| 1 | Project plan & Review meetings | Derek,  Pouya,  Talia | 5 hours | 5 hours | 27/07/2023 | 5 hours | 5 hours | 27/07/2023 |
| 2 | Identify Functional Requirement | Derek,  Pouya,  Talia | 4 hours | 9 hours | 28/07/2023 | 4 hours | 9 hours | 28/07/2023 |
| 3 | Identify Non-Functional Requirement | Derek,  Pouya,  Talia | 3 hours | 12 hours | 29/07/2023 | 3 hours | 9 hours | 29/07/2023 |
| 4 | Create Use Case Diagram | Derek,  Talia | 4 hours | 16 hours | 30/07/2023 | 4 hours | 16 hours | 30/07/2023 |
| 5 | Full Use Case Description | Pouya | 5 hours | 21 hours | 1/08/2023 | 5 hours | 21 hours | on going |
| 6 | Requirement Use Case Traceability Matrix | Derek,  Pouya,  Talia | 3 hours | 24 hours | 2/08/2023 | 3 hours | 24 hours | 2/08/2023 |
| 7 | Design Class Diagram | Talia | 4 hours | 28 hours | 3/08/2023 | 4 hours | 28 hours | on going |
| 8 | Design Sequence Diagram | Derek | 3 hours | 31 hours | 9/08/2023 | 3 hours | 31 hours | 9/08/2023 |
| 9 | Design Activity Diagram | Pouya | 3 hours | 34 hours | 10/08/2023 | 3 hours | 34 hours | 10/08/2023 |
| 10 | Design Views (C&C, Implementation & Deployment Style) | Derek | 6 hours | 40 hours | 11/08/2023 | 6 hours | 40 hours | 11/08/2023 |
| 11 | Make game prototype | Derek,  Pouya,  Talia | 5 hours | 45 hours | 12/08/2023 | 5 hours | 45 hours | 12/08/2023 |
| 12 | Make video presentation | Derek | 4 hours | 49 hours | 13/08/2023 | 4 hours | 49 hours | 13/08/2023 |
| 13 | Coding: Start-up page | Derek | 5 hours | 54 hours | 14/08/2023 | 5 hours | 54 hours | 14/08/2023 |
| 14 | Coding: Configure page | Derek | 4 hours | 58 hours | 15/08/2023 | 4 hours | 58 hours | 15/08/2023 |
| 15 | Coding: Gameplay page | Derek | 6 hours | 64 hours | 16/08/2023 | 6 hours | 64 hours | on going |
| 16 | Coding: game field & block functionlities | Derek | 7 hours | 71 hours | 17/08/2023 | 7 hours | 71 hours | 17/08/2023 |
| 17 | Coding: Esc window | Derek | 3 hours | 74 hours | 18/08/2023 | 3 hours | 74 hours | 18/08/2023 |
| 18 | Coding: Score window | Derek | 3 hours | 77 hours | 19/08/2023 | 3 hours | 77 hours | 19/08/2023 |
| 19 | Coding: Other features | Derek | 6 hours | 83 hours | 20/08/2023 | 6 hours | 83 hours | on going |
| 20 | Github Version Management | Derek,  Pouya,  Talia | 6 hours | 85 hours | 21/08/2023 | 6 hours | 85 hours | on going |

## 1.2 Total working hours

|  |  |  |
| --- | --- | --- |
| **Student Name (#ID)** | **Plan (hours)** | **Actual (hours)** |
| **Talia Tran** | 36 hours | On Going |
| **Derek Qiu** | 62 hours | On Going |
| **Pouya Yazdani** | 35 hours | On Going |
|  |  |  |
| **Total working hours** | 133 hours | On Going |
| **Average working hours per person** | 44.3 hours |  |

## 1.3 Effort and contribution table

|  |  |  |  |
| --- | --- | --- | --- |
| **Student** | **Effort Level\***  (Rating from 0 – 5, the information is filled by the group) | **Contribution Level\***  (Rating from 0 – 5, the information is filled by the group) | **Justification**  If a student received level rating of 3 or less, your group need to give explanation for the low level rating |
| **Talia Tran** |  |  |  |
| **Derek Qiu** |  |  |  |
| **Pouya Yazdani** |  |  |  |
|  |  |  |  |
| **Total** |  |  |  |

* \*Level ratings, 5 = excellent, 4 = good, 3 = reasonable, 2 = poor, 1 = unacceptable, 0 = none

## 1.4 Version Control System

[Your group needs to use a version control system (VCS) to manage the source code development. Please use screenshot to demonstrate that a suitable VCS system has been applied in developing this project. ]

# 2.0 Requirements Analysis

The Requirements Analysis section of the project plan serves as a comprehensive exploration of the functionality and behavior expected from the Tetris game software. This section is vital for understanding the project's scope, as it defines both functional and non-functional requirements that the software must fulfill to meet user expectations and deliver a seamless gaming experience.

## 2.1 Functional requirements

The Functional Requirements table outlines the specific functionalities that the Tetris game software must exhibit. These requirements encompass various aspects, including user interactions, gameplay modes, buttons' functionality, scoring mechanisms, and game termination conditions. These requirements provide a comprehensive guide for the development team to ensure that the game meets the desired user experience and gameplay features.

|  |  |  |
| --- | --- | --- |
| **Identifier** | **Priority** | **Requirement** |
| F-REQ1 | 1 | The system should allow for the game to be executed on at least 2 different platforms |
| F-REQ2 | 1 | The system should display a start-up page when the game is started |
| F-REQ3 | 1 | The start-up page should display the title of ‘Tetris’, current year, the course code, and comprehensive list of all students in group |
| F-REQ4 | 1 | The start-up page should display an 'Exit' button |
| F-REQ5 | 2 | The Exit button displayed on the start-up page should be fully functional, that when pressed/clicked on it the user exits the program with result 0 (successful termination). |
| F-REQ6 | 1 | The start-up page should display a ‘Score’ button that when user clicked, displays the top 10 players along with the scores. |
| F-REQ7 | 2 | The 'Score' button displayed on the start-up page should be fully functional, when pressed it displays the top 10 players along with their respective scores |
| F-REQ8 | 1 | The start-up page should display a ‘Configure’ button that takes the user to a configuration page. |
| F-REQ9 | 2 | The configure button displayed on the start-up page should be fully functional, that when pressed/clicked on it takes the user to another page where they can choose the model between 'standard game' or 'extension'. They can also chose the size of the playing field and block dropping speeds. Player can also chose the player mode between 'Play as user' or 'Play as AI'. |
| F-REQ10 | 1 | When the user presses the 'Standard game' button, the standard game mode is activated which allows the user to play with 7 different block types, each comprising 4 squares. |
| F-REQ11 | 3 | When the user presses the 'Extension' button, the extended game mode is activated which allows the user to play with 7 original block types as in standard mode plus 2 additional block types, each made up of 3 squares |
| F-REQ12 | 1 | The 'Play as User' button displayed on the configuration page should be fully functional, when pressed it activates the normal player mode where the user can use the keyboard to control the dropping blocks |
| F-REQ13 | 3 | The 'Play as AI' button displayed on the configuration page should be fully functional, when pressed it activates the AI player mode where the game's AI takes control of the falling block's movements, dictating its direction, and rotational maneuvers as well as the drop speed |
| F-REQ14 | 1 | The start-up page should have a 'Play' button |
| F-REQ15 | 1 | The play button displayed on the start-up page should be fully functional, that when pressed/clicked on it, that transitions the player to the game interface. |
| F-REQ16 | 1 | The ‘Gameplay’ page should display the game field, a dropping block, group number, current score, number of lines eliminated, current level, game mode, play mode, a ‘next block’ prompt box and the accumulated blocks at the bottom of the field. |
| F-REQ17 | 1 | When the user presses and/or holds down the left arrow-key, the dropping block will move to the left direction, unless it touches another block or faced with a wall in which the dropping block keep falling. |
| F-REQ18 | 1 | When the user presses and/or holds down the right arrow-key, the dropping block will move to the right direction, unless it touches another block or faced with a wall in which the dropping block keep falling. |
| F-REQ19 | 1 | When the user presses and/or holds down the up arrow-key, the dropping block will rotate 90 degree clockwise, unless it touches another block. |
| F-REQ20 | 1 | When the user presses and/or holds down the down arrow-key, the dropping block will increase the falling speed, unless it touches another block. |
| F-REQ21 | 3 | When ‘P’ key is pressed, the ongoing game should be pause or resume. |
| F-REQ22 | 2 | When ‘Esc’ key is pressed, two button will be displayed. By selecting "Yes" will return the player to the start-up page, while "No" allows gameplay to continue. |
| F-REQ23 | 4 | When ‘M’ key is pressed, music and sound effects will be turn on or off. |
| F-REQ24 | 2 | The point/score should initialize at 0. A player or AI can accrue points only by eliminating lines. The more lines they eliminate with a single block drop, the more points they earn. |
| F-REQ25 | 2 | When the user/AI eliminates 1/2/3/4 lines, the total point will be increased by 100/300/600/1000 points, and this should be displayed to the user |
| F-REQ26 | 1 | A game ends when no room for a new block to enter play or the game been terminated. |
| F-REQ27 | 4 | When the play screen is being viewed, there should be suitable music playing |
| F-REQ28 | 5 | There should be appropriate sound effects played during game-play. |
| F-REQ29 | 3 | The system should have multiple level of falling speed depends on the level of game. |
| F-REQ30 | 1 | The system should record the score and keep the record updated in the back-end |
| F-REQ31 | 1 | As game ends, if player’s/AI’s score ranks in the top 10, a dialog box will be displayed, and a name will be required to entre, it will be recorded along with the score, and will be displayed on the high score page. |

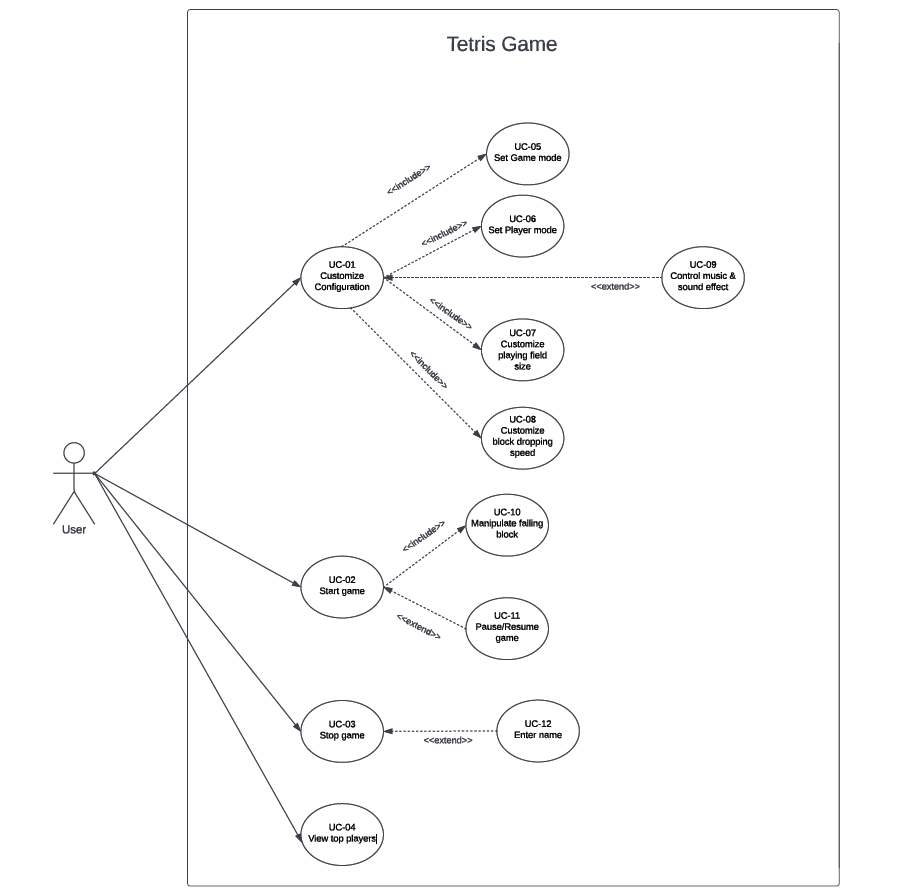
## 2.2 Non-functional requirements

The Non-Functional Requirements table captures the broader attributes that the Tetris game software should possess. These attributes focus on aspects beyond direct functionalities, including performance, usability, reliability, supportability, security, and more. The non-functional requirements emphasize the seamless and enjoyable gaming experience for users, from smooth responsiveness to user-friendly interfaces and consistent aesthetics, while also ensuring stability, data security, and efficient resource usage.

|  |  |  |
| --- | --- | --- |
| **Identifier** | **Priority** | **Requirement** |
| NF-REQ1 | 5 | Performance - The System should respond quickly to user input and render the falling xblocks smoothly, providing a seamless gaming experience without noticeable delays or lags. |
| NF-REQ2 | 5 | Usability - The system should have a “How to Play” page as documentation for the player to easily use the system, allowing players to understand the controls and gameplay mechanics. |
| NF-REQ3 | 5 | Usability - The user interface should be intuitive and easy to navigate, with clear labels and instructions |
| NF-REQ4 | 5 | Usability - The game's sound effects and music should be of high quality and suit the gameplay experience |
| NF-REQ5 | 5 | Reliability - The system should operate without crashes or significant errors, providing a stable and reliable platform for users to enjoy the gaming experience. |
| NF-REQ6 | 5 | Reliability - The scoring system should accurately calculate and display the players' scores without any discrepancies. |
| NF-REQ7 | 5 | Functionality - The system should be able to have responsive sizing of the game |
| NF-REQ8 | 5 | Usability – The system should be aesthetically pleasing, and consistent with the original 1980’s Tetris game. |
| NF-REQ9 | 5 | Supportability – The system should not require a large amount of computer memory in order to play locally. |
| NF-REQ10 | 5 | Security - The system should not collect or store any sensitive user data or personal information |

## 2.3 Use case diagram

The Use Case Diagram showcases the high-level functionalities that player can perform within the Tetris game system. This diagram provides an overview of the major use cases and their relationships, offering a clear understanding of how player and the system interact to achieve specific goals during gameplay.



## 2.4 Full use case description

This table presents a comprehensive description of the "Pause/Resume Game" use case, detailing how the game's pause and resume functionalities operate during gameplay.

|  |  |  |
| --- | --- | --- |
| **Fully developed use case** | | |
| **Use case name:** | Pause/Resume Game | |
| **Scenario/s:** | Game to be paused/resumed | |
| **Trigging Event:** | The player presses the 'P' key during gameplay to pause the game.  The player presses the 'P' key again to resume the game. | |
| **Brief Description:** | when pausing and resuming gameplay. The player has the ability to temporarily halt the game's progress and then continue playing from where they left off. | |
| **Actors:** | User | |
| **Related user case:** | UC-02 Start game, UC-10 Manipulate falling block | |
| **Stakeholders:** | Players, developers. | |
| **Precondition:** | The game is in progress.  The player is engaged in playing Tetris.  The 'P' key is functioning as the pause/resume control. | |
| **Postcondition:** | No ability to manipulator after game paused  Suspends all active elements on the screen after game paused  Game can be kept playing after game resume | |
| **The flow of activities:** | Player | System |
| 1. The player indicates the desire to pause the game by pressing the 'P' key.  2. The player observes the paused state and decides to resume the game by pressing the 'P' key again. | 1.1 The system detects the pause command and proceeds to suspend all active elements on the screen.  1.2 The system displays a paused state with a message indicating that the game is on hold.    2. The game recognizes the resume command and restores the game's progression, including block movement and animation. |
| **Expectation condition:** | - Physical keyboard malfunction.  - Game crash caused by computer configuration issues. | |

## 2.5 Activity diagram

The Activity Diagram illustrates the step-by-step flow of activities and actions within a specific use case, focusing on the sequence and logic of actions taken by the user and the system. This diagram offers the gameplay mechanics, control interactions, and outcomes in the Tetris game.

# 3.0 Video link

[please put the URL of your video, and make sure that the video can be viewed by the assessor]