Software Testing Report

Group 1 Cohort 1

Tom Haslam, Christopher Oulton, Charlie Piper, Shirin Sitara, Dillon Anthony, Kevin Thomas, Ella Daramola

Introduction

To test our game, we chose to use 3 types of tests - unit tests, manual tests and end-to-end tests. We divided these approaches based on ease and appropriateness of the methods pertaining to each requirement.

To ensure a systematic and comprehensive approach to our testing, we began by creating a traceability matrix Traceability Matrix. This matrix allowed us to map each requirement to its corresponding tests, ensuring complete coverage and a clear understanding of the testing scope.

For each class tested, we aimed at checking 3 main parts - main task with appropriate inputs, boundary inputs and exceptions. This gave us a thorough understanding of the working and correctness of the class.

The classes that were directly linked to game logic and game components were unit tested. These include classes such as activity, movements and assets. This was the most suitable approach as these classes had multiple parts to be checked and a manual test on these would not be time or effort efficient. On analysing further, we noticed that most functional requirements were suited for this method of testing.

Other important aspects of the game such as the maps, collisions and accessibility came under manual testing. These aspects of the game are more subjective and thus can be more appropriately tested using human interaction. We had multiple runs of manual testing done by different individuals to get different perspectives and cover all the possibilities of errors that may arise. Affirming the previous observation, most user requirements and non-functional requirements fit well to be tested manually.

To ensure that all these puzzle pieces work well together, we conducted end-to-end tests, for each day and week, thus, the game.

Testing Report

1. Unit Testing:

- Activity Test: tests were devised to assess the mechanisms governing the player's energy, health, and happiness levels, ensuring they respond appropriately to in-game actions and events such as sleep, study, eat, and recreational activities. The test passed affecting health, energy, happiness and study level appropriately.

- Activity Point test: tests devised to ensure the appropriate change level of matics when an activity is done. The tests passed covering all types of activities, encompassing the whole scope.

- Player Movement Testing: tests were conducted to validate the movement of the Player class under various scenarios, including movement. The test passed covering

all scopes of movement including Up, down, left and right and accurately cover what the movement should do

- Study Level Testing: tests were employed to evaluate the progression and functionality of the study level system, verifying the accuracy of calculations and the impact on gameplay dynamics. The tests passed covering the basic but could be expanded on to cover a wider scope.

- Assets Management Testing: Automated tests were utilized to verify the loading, utilization, and disposal of game assets, ensuring seamless integration and optimal performance. The test passed encompassing the whole scope of assets used.

- Day length Test: Tests were automated to ensure time resets for eash day. This tests passed covering all scopes.

2. Manual Testing:

Manuel tests conducted - MANUAL TESTING

Manual testing was performed to assess aspects of the system that were not easily automated, such as:

- Screens: Manual testing was preferred for assessing the visual elements, screen transitions, and collision detection mechanisms within the game environment to ensure a seamless and immersive experience for players.

- Visuals: Manual testing was indispensable for evaluating the visual appeal and ease of the game. This test failed due to the difficulty of finding the activities of the game, to fix this issue activities should be made more obvious.

- Collisions: Manual testing was preferred for assessing the visual elements, screen transitions, and collision detection mechanisms within the game environment to ensure a seamless and immersive experience.

- Sound: Manual testing was indispensable for evaluating the auditory feedback of the game. Human evaluators provided subjective feedback on graphics quality, animation smoothness, and sound effects immersion.

- Transition: Manual testing was performed to assess the transitions between different screens or game states. This test passed and covered the whole scope of the game.

- Character/Sprite Handling: Manual testing was conducted to evaluate the handling and rendering of character sprites. The tests failed due to inconsistent character, after selection, throughout the game. Implementation should ensure consistent use of selected character to pass this test.

- Map boundaries: Tests were conducted to ensure that the player character cannot move outside the defined boundaries of the game map. All tests passed successfully covering all scopes boundaries including interior.

- Leaderboard & score test: Tests were performed to verify the correct calculation and display of player scores and the functionality of the leaderboard system. All tests passed.

End-to-End Tests:
Manual end-to-end tests were conducted to:

- Test the overall user experience and ensure that the game flow is intuitive and user-friendly. This test passed and overall met all requirements of the customer.

The comprehensive testing campaign conducted on the game application has yielded valuable insights and identified areas for improvement. While the majority of tests across various domains, including gameplay mechanics, asset management, user interface, and end-to-end functionality, have passed successfully, there are specific areas that require further attention and optimization.