

# Software Testing Report

Cohort 1 Group 7

## **Group Members:**

Kyle Clifton (wnp512)

Bailey Evers (rkh552)

Matt Hollyhead (fpk506)

William Martin (zjc524)

Max Pither (rkm538)

Enkhtuguldur Sarantsetseg (wbj512)

Ronny Watt (dvn513)

## Testing Approach

When designing tests for our game, we devised a testing table that specifically links each of the project's requirements to a test designed to test that specific requirement, helping to ensure thorough testing of the project. We evaluated each requirement to determine a unique and tailored approach to validating the implementation of all parts of the game. As part of this evaluation, we considered whether manual or automated testing would be most appropriate. In cases where manual testing was chosen, we felt that it would be difficult or unfeasible to run an automated test, whereas a simple manual test could easily verify the completeness of the requirement. For example, some requirements specify specific graphical style choices, which can only be verified by a human confirming whether they meet the stated requirements. Such cases, and others with similar concerns, were assigned manual tests as a result.

## Tests

Requirements ID	Test Type	Test Description	Pass/Fail
UR_ESCAPE_MAZE FR_WAY_OUT	Manual	Play through the game until reaching the exit and escaping.	Pass
UR_POSITIVE_EVENTS FR_POSITIVE_EVENTS NFR_EVENT	Manual	Explore the map and ensure that there are at least 3 positive events that can be interacted with.	Pass
UR_NEGATIVE_EVENTS FR_NEGATIVE_EVENTS NFR_EVENT	Manual	Explore the map and ensure that there are at least 5 negative events that can be interacted with.	Pass
UR_HIDDEN_EVENTS FR_HIDDEN_EVENTS NFR_EVENT	Manual	Explore the map and ensure that there are at least 3 hidden events that can be interacted with.	Pass
UR_TIME_LIMIT FR_TIMER NFR_TIMER	Manual	Start the game and ensure that a 5 minute timer is displayed, counts down at the correct speed, and that the game ends upon reaching 0.	Pass
UR_PAUSE FR_PAUSE_MENU NFR_PAUSE	Manual	Start the game, then press the pause button to check if the pause screen appears.	Pass
UR_SCORE NFR_SCORE	Manual	Reach the end of the game and check whether a score is displayed to the user, and check whether the score has been correctly counted based on the statistics displayed.	Pass
FR_SCORE	Automated	Calculate the score based on predetermined interaction counts and remaining time.	Pass
UR_BOUNDARIES FR_BOUNDARIES	Manual	Explore the map in-game, checking every wall to ensure no gaps.	Pass
UR_DO_NOT_SAVE FR_DO_NOT_SAVE NFR_DO_NOT_SAVE	Manual	Start and end the game in multiple ways (close via pause menu, X button on window, and by reaching the exit), then start the game again to ensure the previous state of the game has not been saved.	Pass
UR_HIDDEN_EVENT_HINTS	Manual	Interact with the hidden events and check whether a hint message is displayed.	Pass
UR_DEAN	Manual	Explore the map and check whether a dean character is present.	Pass
UR_LEADERBOARD FR_LEADERBOARD	Manual	Complete the game 6 times or more, and ensure that the leaderboard menu displays the top 5 scores.	Pass
UR_ACHIEVEMENTS FR_ACHIEVEMENTS	Manual	Clear any existing achievement save data, then start and complete a game to check that the player is awarded achievements based on their gameplay.	Pass
FR_EVENT_TRACKER	Manual	Start the game and interact with some number of events, then exit the maze and ensure that the correct number of interactions for each event	Pass

		type is displayed.	
NFR_PRESERVE_GAMESTATE	Manual	Start the game, open the pause menu and wait a few seconds, then unpause and ensure that the game state has not changed (i.e. timer not decreased, entities not moved)	Pass
UR_ESCAPE_MAZE FR_MAZE	Manual	Start the game and check the there is a university-themed maze map for the player to explore	Pass
UR_ESCAPE_MAZE FR_END_GAME	Manual	Start and attempt to end the game via escaping, timer expiration or dean capture separately	Pass
UR_PROFESSOR FR_PROFESSOR	Manual	Explore the map and check whether a professor character is present.	Fail
UR_EVENT_AMOUNT S FR_EVENT_AMOUNT S	Manual	Explore the map and ensure that at least 5 negative, 3 positive, and 3 hidden events are present.	Pass
UR_THEME FR_THEME	Manual	Start the game and and check the map is visually styled to a university	Pass
UR_DIFFICULTY FR_DIFFICULTY	Manual	Check that the game supports multiple difficulty levels.	Fail
NFR_END_GAME	Manual	Complete the game and check that there is a transition to an end state	Pass
NFR_PERFORMANCE	Manual	Start the game and play through on an average device, ensuring that it does not crash	Pass
NFR_UI	Manual	Have a new player start and complete the game blindly to ensure that instructions are unambiguous	Pass
UR_AUDIO FR_AUDIO	Manual	Start the game and test that audio plays out of device speakers as expected	Pass

## Test Evaluation

Most of the tests passed as expected, and the game is almost fully functional for the end-user. There were two failures overall, being for the UR\_PROFESSOR and UR\_DIFFICULTY user requirements, however this is due to these features ultimately being removed from the project due to them being deemed as unnecessary for the functionality of the game and time constraints within the project, so there is no feasible way to pass them. If the professor and difficulty mechanic were to be added, they could provide a more engaging game experience and replay factor beyond what the game currently has, although the final product is already a well rounded and functional experience for the end-user, so this isn't necessary.

Excluding these two failures, all tests passed as planned providing the expected results.