

Manual Tests

These tests cover all the functional requirements found in our requirements document.

Req ID	Description	Manual Tests	Success
FR_GAME_START	The system shall allow the user to start the game from the menu screen	1. Execute the jar file or run the program. 2. Press the "New Game" button.	You are entered into a new game.
FR_GAME_QUIT	The system shall allow the user to quit the game from the menu screen	1. Execute the jar file or run the program. 2. Press the "Exit" button.	The program exits
FR_GAME_END	The system shall end the game when the user has the character sleep on the 7th day	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Go to the bed in the house. 5. Sleep 7 times (using E to interact with the bed).	The game will end.
FR_GAME_END_STATS	The system shall show the users the amount of activities they have completed on the end screen	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Do a number of activities. 5. Go to the bed in the house. 6. Sleep 7 times.	When the game ends it tells you what activities you have done.
FR_CHARACTER_SELECTION	The system should allow the user to select from a range of characters at the start of the game	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character.	The game starts with your selected character.
FR_CHARACTER_MOVEMENT	The system shall allow the user to move their character around the map	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Use the WASD keys to move.	The player moves around the map.

FR_CHARACTER_COLLISION	The system shall never allow the user to move the character through collidable objects	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Move to a wall using S.	Once the wall is met the player sprite should no longer continue moving downwards.
FR_CHARACTER_INTERACTION	The system shall allow the user to interact with the map	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Move to the house on the right. 5. Walk into the door. 6. The map should now change. 7. Move to the bed to the right and press E.	The day at the top should change.
FR_SCREEN	The system shall show the user different screens relating to their current stage in the game (start, settings, maps, end)	1. Execute the jar file or run the program. 2. Use the settings option to see the settings menu. 3. Go back to the main menu. 4. Press the "New Game" button. 5. Select a character. 6. Go into several buildings. 7. Go to the house. 8. Sleep 7 times.	All the expected screens should appear, including a start screen (Main menu), settings, several maps and an end screen.
FR_MAP	The system shall show user the map the character is currently on	1. Execute the jar file or run the program. 2. Use the settings option to see the settings menu. 3. Go back to the main menu. 4. Press the "New Game" button. 5. Select a character. 6. Go into several buildings. 7. Go to the house. 8. Sleep 7 times.	All the expected screens should appear, including a start screen (Main menu), settings, several maps and an end screen.
FR_MAP_BUILDINGS	The system shall include common Heslington East	1. Execute the jar file or run the program. 2. Press the "New Game"	There should be a variety of buildings

	campus buildings on the map for the user to enter and a map for each building	button. 3. Select a character. 4. Move around the map, entering buildings which allow the character to.	including a place to eat (representing the Piazza), accommodation and a place to study (Representing the Ron Cooke Hub).
FR_MAP_TRANSITION	The system shall allow the user to transition between buildings by walking to the doors	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Move the character right and enter the first building through the door. 5. Once inside the building, exit again via the doorway that the character just came through.	The character should move between the world and building maps when the character moves through the door.
FR_ACTIONS	The system shall allow the user to complete actions throughout the day	5. Execute the jar file or run the program. 6. Press the "New Game" button. 7. Select a character. 8. Move right to the basketball court. 9. Press E when standing on the court to complete an action. Observe the text at the top: the time should change. 10. Move to the football pitch to the left. 11. Press E to complete an action. Again, the time should change. 12. Move to the house to the left and enter it. 13. Go to any exclamation mark and press E, the time should change as long as the user is still allowed to do the action.	The character should be allowed to do a variety of actions throughout the day, assuming there is enough time to do so and the player's stats allow for it.
FR_ACTIONS_IN	The system should	See test above.	Every time an

DICATION	indicate when an action has or hasn't been completed		action is completed it should be clear by a change in the time and a possible change in statistics. If an action cannot be completed, these effects should not take place and text should indicate the action cannot be completed.
FR_ACTIONS_TY PES	The system shall contain different types of actions for the user to complete (study, eat, relax, sleep) (one of each)	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Move to and enter the house to the starting point's right. 5. For each activity you should press E to do them and observe the time changing at the top, indicating the activity has been done. 6. Go to the bookshelf and press E (relax). 7. Go into the kitchen, to one of the exclamation marks and press E (eat). 8. Go to the bed and press E (sleep). 9. Go to the desk and press E (study).	For each activity done the time at the top should change, indicating that the activity has been completed.
FR_ACTIONS_IN SUFFICIENT	The system shall never allow the user to complete an action if they don't have a sufficient amount of any resource (time, energy).	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Move to the football pitch to the right. 5. Press E several time to do the activity. After a number of times	After doing each activity, possibly a number of times, the player should have an indication that the activity is no longer

		(currently 2) the text at the bottom should indicate that you cannot do the activity due to low energy. Pressing E should have no effect. 6. Move to the bench to the bottom left. 7. Press E to daydream until the time is 00:00. The text at the bottom should indicate that you cannot do the activity due to a lack of time. Pressing E should have no effect.	possible and pressing E to interact should have effect.
FR_STATS	The system shall keep track of the games stats	1. Run through the game several times, doing different activities each time. 2. Observe the score at the end of the game each time.	The end score should change as expected, according to the actions of the player throughout the game.
FR_STATS_UPDATE	The system shall update the stats when an action has occurred	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Move to the football pitch to the right. 5. Press E to interact with the activity.	The player's stats at the bottom right should change when the activity has been done.
FR_STATS_RESET	The system shall reset some stats at the end of each day (energy, time)	1. Execute the jar file or run the program. 2. Press the "New Game" button. 3. Select a character. 4. Observe the player's stats when the game starts. 5. Enter the house to the character's right. 6. Do some activities (other than sleep in the bed) by moving to the exclamation marks and pressing E. 7. Go to the bed and press E.	Some of the player stats (currently study level and energy) should reset to the level it was at the start of the game.

FR_STATS_SHOW	The system shall show the stats for the day to the user so they can keep track of them	See test above (for FR_STATS_RESET)	Some of the stats should reset (a sleep causes a new day).
FR_LEADERBOARD	The system shall keep track of all users who have successfully completed the game and show stats of the top ten users	Execute the jar file or run the program. Play through the game to completion, ensuring various activities are undertaken. Complete the game. Check if your stats are recorded. Restart the game and play through it again with different activities. Complete the game again. Verify if your new stats are recorded and updated on the leaderboard. Repeat steps 5-7 with different gameplay strategies to ensure the leaderboard accurately reflects user performance.	The player name and score should show up on the leaderboard if you exit and reopen the game.
FR_ACTIVITY_COUNTER	The system shall keep track of the number of individual activities the user has completed	Execute the jar file or run the program. Play through the game, ensuring to complete different activities. Keep track of the number of times each activity is completed. Complete the game. Check if the total count of individual activities completed is displayed or stored somewhere in the game. Repeat steps 2-5 with various gameplay approaches to confirm accurate tracking of activity counts.	On the leaderboard screen, a list of achievements should show.
FR_ACTIVITY_STREAK	The system shall award the user a streak for each activity they complete multiple times throughout the duration of the	Execute the jar file or run the program. Play through the game, focusing on repeating the 'daydream' multiple times in a row. Ensure that each activity is	On the leaderboard screen, a list of achievements should show.

	game	completed consecutively without interruption. Complete the game. Check if the game recognizes and awards streaks for completing the same activity multiple times consecutively (DAYDREAMER). Restart the game and repeat steps 2-5 with different activities (gym,snack,teaching hours) to test streak functionality across various actions.	
FR_MUSIC	The system should adjust the volume of the background music based on user input	Execute the jar file or run the program. Access the settings menu on the home screen. Adjust the volume of the background music using the provided controls. Play the game, ensuring the background music plays at the adjusted volume level. Navigate through different screens and gameplay scenarios to confirm consistent volume adjustment. Experiment with various volume levels to verify the range of adjustment.	Music volume should adjust with the slider and stay consistent screen to screen, the checkbox will disable/enable the sound and maintain the volume.