

| Test_ID             | Input                      | Expected Output   | Actual Output                  | Description  | Related Requirements  | Category         | Author | Status |
|---------------------|----------------------------|---|--------------------------------|--|---|------------------|--------|--------|
| BulletDispenser_1.1 | N/A                        | Null  | Null                           | Testing that the BulletDispenser constructor is working as intended with standard input                          | UR_FIRETRUCK_REPAIR<br>UR_ET_UNIQUE_SPEC<br>UR_ET_IMPROVEMENT<br>UR_LOSS_CONDITION<br>UR_DIFFICULTY_LEVEL<br>UR_INTUITIVE<br>UR_FORTRESS_ATTACK | Functional, Unit | Ben    | Pass   |
| BulletDispenser_1.2 | A mocked Pattern class     | The time between firing patterns - 'PatternTime'            | 1f - the time given            | Testing that when passed a pattern and with one pattern in the List, the PatternTime is set to the correct value | UR_FIRETRUCK_REPAIR<br>UR_ET_UNIQUE_SPEC<br>UR_LOSS_CONDITION<br>UR_DIFFICULTY_LEVEL<br>UR_INTUITIVE<br>UR_FORTRESS_ATTACK                      | Functional, Unit | Ben    | Pass   |
| BulletDispenser_1.3 | A mocked Pattern class     | The same mocked Pattern Class as inputted - 'FiringPattern' | The mocked Pattern class       | Testing that the pattern passed is added and that the correct FiringPattern                                      | UR_FIRETRUCK_REPAIR<br>UR_ET_UNIQUE_SPEC<br>UR_LOSS_CONDITION<br>UR_DIFFICULTY_LEVEL<br>UR_INTUITIVE<br>UR_FORTRESS_ATTACK                      | Functional, Unit | Ben    | Pass   |
| BulletDispenser_1.4 | Two mocked Pattern classes | The 'PatternTime' from the first passed Pattern class       | 1f - the time given            | Testing that the if a second pattern was added the 'patternTime' do not change                                   | UR_FIRETRUCK_REPAIR<br>UR_ET_UNIQUE_SPEC<br>UR_LOSS_CONDITION<br>UR_DIFFICULTY_LEVEL<br>UR_INTUITIVE<br>UR_FORTRESS_ATTACK                      | Functional, Unit | Ben    | Pass   |
| BulletDispenser_1.5 | Two mocked Pattern classes | The first passed Pattern class - 'FiringPattern'            | The first passed Pattern class | Testing that the if a second pattern was added the 'firingPattern' do not change                                 | UR_FIRETRUCK_REPAIR<br>UR_ET_UNIQUE_SPEC<br>UR_LOSS_CONDITION<br>UR_DIFFICULTY_LEVEL<br>UR_INTUITIVE<br>UR_FORTRESS_ATTACK                      | Functional, Unit | Ben    | Pass   |
|                     |                            |   |                                |  |   |                  |        |        |

|            |  |  |  |   |  |                               |     |      |
|------------|--|--|--|---|--|-------------------------------|-----|------|
| Entity_1.1 | N/A  | Correct initialized variables                          | The correct initialized variables - True for isAlive - 100 for health - (1,2) for position | Testing that the Entity constructor is working as intended with standard input                          | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_FIRETRUCKS_REFILL<br>UR_FIRETRUCK_REPAIR<br>UR_ET_UNIQUE_SPEC<br>UR_ET_IMPROVEMENT<br>UR_FIRETRUCK_MIN_START<br>UR_ET_MIN_START<br>UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_MINIGAME UR_HIGHSORE<br>UR_PATROLS<br>UR_FORTRESS_ATTACK | Functional, Intergrated, Unit | Ben | Pass |
| Entity_1.2 | Int value into .applyDamage                                  | Entities health minus the input value                  | entityTest health minus the input value - 50   | Testing that applyDamage removes the amount of health with standard value - with not killing            | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_FIRETRUCKS_REFILL<br>UR_FIRETRUCK_REPAIR<br>UR_ET_IMPROVEMENT<br>UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_HIGHSORE<br>UR_FORTRESS_ATTACK   | Functional, Intergrated, Unit | Ben | Pass |
| Entity_1.3 | Int value into .applyDamage greater than Entities max health | Entities health minus the input value - negative value | entityTest health minus the input value - '-10'  | Testing that applyDamage removes the amount of health with standard value - does kill - negative health | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_FIRETRUCKS_REFILL<br>UR_FIRETRUCK_REPAIR<br>UR_ET_IMPROVEMENT<br>UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_HIGHSORE<br>UR_FORTRESS_ATTACK   | Functional, Intergrated, Unit | Ben | Pass |

|            |  |  |                          |  |   |                               |     |      |
|------------|--|--|--------------------------|--|---|-------------------------------|-----|------|
| Entity_1.4 | Int value into .applyDamage equal to the Entities max health | Zero                                     | Zero                     | Testing that applyDamage removes the amount of health with standard value - does kill - at zero health | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_FIRETRUCKS_REFILL<br>UR_FIRETRUCK_REPAIR<br>UR_ET_IMPROVEMENT<br>UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_HIGHSCORE<br>UR_FORTRESS_ATTACK | Functional, Intergrated, Unit | Ben | Pass |
| Entity_1.5 | Int value into .applyDamage that is negative                 | An IllegalArgumentException to be thrown | IllegalArgumentException | Testing that applyDamage should throw IllegalArgumentException when passed a negative number           | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_FIRETRUCKS_REFILL<br>UR_FIRETRUCK_REPAIR<br>UR_ET_IMPROVEMENT<br>UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_HIGHSCORE<br>UR_FORTRESS_ATTACK | Functional, Intergrated, Unit | Ben | Pass |
| Entity_1.6 | N/A (positive health)  | True when check if it isAlive()          | 1                        | Testing that isAlive works with a standard positive health   | UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_HIGHSCORE<br>UR_FORTRESS_ATTACK  | Functional, Intergrated, Unit | Ben | Pass |
| Entity_1.7 | Int value between 0 and max health but not equal             | True when check if it isAlive()          | 1                        | Testing that isAlive works after Entity takes damage but not kills                                     | UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_HIGHSCORE<br>UR_FORTRESS_ATTACK  | Functional, Intergrated, Unit | Ben | Pass |

|                 |                                   |                                  |   |  |  |                               |     |      |
|-----------------|-----------------------------------|----------------------------------|---|--|--|-------------------------------|-----|------|
| Entity_1.8      | Int value equal to max health     | False when check if it isAlive() | 0   | Testing that isAlive returns false when health is zero                               | UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_HIGHSORE<br>UR_FORTRESS_ATTACK                      | Functional, Intergrated, Unit | Ben | Pass |
| Entity_1.9      | Int value greater than max health | False when check if it isAlive() | 0   | Testing that isAlive returns false when health is negative                           | UR_WIN_CONDITION<br>UR_LOSS_CONDITION<br>UR_ET_DESTROYES_STATION<br>UR_HIGHSORE<br>UR_FORTRESS_ATTACK                      | Functional, Intergrated, Unit | Ben | Pass |
|                 |                                   |                                  |   |  |  |                               |     |      |
| StatBar_1.1     | N/A                               | Correct initialized variables    | The correct initialized variables - height of 3       | Testing that the constructor initializes StatBar correctly by checking values        | UR_FIRETRUCK_REPAIR<br>UR_ET_UNIQUE_SPEC<br>UR_ET_IMPROVEMENT UR_INTUITIVE   |                               | Ben | Pass |
|                 |                                   |                                  |   |  |  |                               |     |      |
| WaterStream_1.1 | N/A                               | Correct initialized variables    | The correct initialized variables - position of (0,0) | Testing that the constructor for WaterStream works correctly                         | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_FIRETRUCKS_REFILL<br>UR_ET_UNIQUE_SPEC UR_MINIGAME<br><br>UR_INTUITIVE                     |                               | Ben | Pass |
|                 |                                   |                                  |   |  |  |                               |     |      |
| GameObject_1.1  | N/A                               | Correct initialized variables    | The correct initialized variables                     | Testing that the GameObject Constructor is working as intended with a standard input | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_ET_UNIQUE_SPEC<br>UR_ET_MIN_START<br>UR_DIFFICULTY_LEVEL<br>UR_INTUITIVE<br><br>UR_PATROLS | Functional, Intergrated, Unit | Ben | Pass |

|                       |  |  |   |  |  |                               |     |      |
|-----------------------|--|--|---|--|--|-------------------------------|-----|------|
| <b>GameObject_1.2</b> | N/A  | The 'middle' between the position vector and the top-right corner (calculated by width and height)     | A vector between the position vector and the width and height - (25,25) | Testing that when the GameObject is at location (0,0) .getCentre() calculates the correct centre                                 | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_ET_UNIQUE_SPEC<br>UR_ET_MIN_START UR_PATROLS | Functional, Intergrated, Unit | Ben | Pass |
| <b>GameObject_1.3</b> | New position vector  | The 'middle' between the new position vector and the top-right corner (calculated by width and height) | A vector between the position vector and the width and height - (75,75) | Testing that wen GameObject is at any 'middle' location .getCentre() calculates the correct centre                               | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_ET_UNIQUE_SPEC<br>UR_ET_MIN_START UR_PATROLS | Functional, Intergrated, Unit | Ben | Pass |
| <b>GameObject_1.4</b> | N/A - Another instance of GameObject with (0,0) as dimensions  | The centre calculated when dimensions (width / height) are (0,0)                                       | A vector between the position vector and the width and height - (0,0)   | Testing that when the dimensions are (0, 0) .getCentre() calculates the correct centre   | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_ET_UNIQUE_SPEC<br>UR_ET_MIN_START UR_PATROLS | Functional, Intergrated, Unit | Ben | Pass |
| <b>GameObject_1.5</b> | New position of positive standard numbers - (500, 500)   | The new position of GameObject to be what is set   | The input - (500, 500)  | Testing that .setPosition() accepts standard values and changes accordingly  | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_ET_UNIQUE_SPEC<br>UR_ET_MIN_START UR_PATROLS | Functional, Intergrated, Unit | Ben | Pass |
| <b>GameObject_1.6</b> | New position of two negative numbers - (-10, -10)  | The new position of GameObject to be what is set   | The input - (-10, -10)  | Testing that .setPosition() accepts negative values and changes accordingly  | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_ET_UNIQUE_SPEC<br>UR_ET_MIN_START UR_PATROLS | Functional, Intergrated, Unit | Ben | Pass |
| <b>GameObject_1.7</b> | changePosition() with input of standard positive numbers - position (50, 50) and the change (100, 100) | The position before changePosition() plus the values passed  | The addition of position and change - (150, 150)                        | Testing that .changePosition()() changes correctly with standard values (.changePosition() changes current position by vector v) | UR_ET_MIN_START UR_PATROLS   | Functional, Intergrated, Unit | Ben | Pass |
| <b>GameObject_1.8</b> | ChangePosition() with input of negative numbers - position (50, 50) and the change (-25, -25)          | The position before changePosition() plus the values passed  | The addition of position and change - (25, 25)                          | Testing that .changePosition()() changes correctly with an all negative vector input   | UR_ET_MIN_START UR_PATROLS   | Functional, Intergrated, Unit | Ben | Pass |

|                    |   |   |   |   |   |                               |     |      |
|--------------------|---|---|---|---|---|-------------------------------|-----|------|
| GameObject_1.9     | changePosition() with input of one positive and one negative number - position (50, 50) and the change (25, -25)              | The position before changePosition() plus the values passed                                 | The addition of position and change - (75, 25)                                      | Testing that .changePosition()() changes correctly with one part of the vector being negative                     | UR_ET_MIN_START UR_PATROLS  | Functional, Intergrated, Unit | Ben | Pass |
|                    |   |   |   |   |   |                               |     |      |
| FireTruck_2.0_USER | User playing game - Attacks alien or fortress to use water - returns to FireStation - Completes Minigame - Checks waterLevels | The user will lose water and then refill it after completing minigame                       | Max water in the bar after using some   | Testing that user can complete a the most crucial game-life cycle to complete the game                            | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_FIRETRUCKS_REFILL<br>UR_FIRETRUCK_REPAIR<br>UR_FIRETRUCK_MIN_START<br>UR_ET_MIN_START<br>UR_ET_DESTROY_STATION<br>UR_MINIGAME,<br><br>UR_HIGHSORE<br>UR_FORTRESS_ATTACK | Functional, Intergrated, Unit | Ben | Pass |
| FireTruck_2.1_USER | User playing game - Does not attack - moves in FireStation - checks waterLevels   | No change in water levels and no activation of minigame                                     | Nothing happened  | Testing that there are no areas to automatically trigger minigame   | UR_FIRETRUCKS_UNIQUE_SPEC<br>UR_FIRETRUCKS_REFILL<br>UR_FIRETRUCK_REPAIR<br>UR_FIRETRUCK_MIN_START<br>UR_ET_MIN_START UR_MINIGAME   | Functional, Intergrated, Unit | Ben | Pass |
|                    |   |   |   |   |   |                               |     |      |
| Powerup_1.1_USER   | User starting a new game  | The user sees 7 powerups around the map, consisting of 5 types of powerup randomly assigned | 7 powerups spawn around the map with 5 different types of powerup randomly assigned | Testing to make sure that 7 powerups spawn around the map in different locations and with one of 5 random effects | FR_FIRE_TRUCK_POWERUP<br><br>UR_FIRE_TRUCK_POWERUP  | Functional, integrated, user  | Ben | Pass |
| Powerup_1.2_USER   | User staring a new game   | The user opens a saved game   | The powerups persist between game saves   | Testing to make sure that the powerups are saved when the game is saved and reopened                              | FR_FIRE_TRUCK_POWERUP<br><br>UR_FIRE_TRUCK_POWERUP<br><br>FR_GAME_SAVE  | Functional, integrated, user  | Ben | Pass |

|                            |   |  |   |  |  |                              |     |      |
|----------------------------|---|--|---|--|--|------------------------------|-----|------|
|                            |   |  |   |  | UR_GAME_SAVE,                                  |                              |     |      |
| <b>Powerup_1.3_USER</b>    | User starting a new game and obtaining a powerup  | The user obtains a powerup and after a random amount of time the powerup respawns                          | The powerup respawns after a random amount of time                              | Testing to make sure that the powerups respawn after a random amount of time                             | UR_FIRE_TRUCK_POWERUP<br>FR_FIRE_TRUCK_POWERUP | Functional, integrated, user | Ben | Pass |
| <b>Powerup_1.4_USER</b>    | User starting a new game and obtains a <i>speed</i> powerup                             | The user obtains a <i>speed</i> powerup which increases the fire trucks speed                              | The fire truck speeds up after obtaining the speed powerup                      | Testing to make sure that the speed powerup has the desired effect                                       | UR_FIRE_TRUCK_POWERUP<br>FR_FIRE_TRUCK_POWERUP | Functional, integrated, user | Ben | Pass |
| <b>Powerup_1.5_USER</b>    | User starting a new game and obtains a <i>damage</i> powerup                            | The user obtains a <i>damage</i> powerup which increases the fire trucks damage                            | The fire truck's damage is increased  | Testing to make sure that the damage of a fire engine is increased when moving over a damage powerup.    | UR_FIRE_TRUCK_POWERUP<br>FR_FIRE_TRUCK_POWERUP | Functional, integrated, user | Ben | Pass |
| <b>Powerup_1.6_USER</b>    | User starting a new game and obtains a <i>shield</i> powerup                            | The user obtains a <i>shield</i> powerup which adds a shield around the fire truck.                        | The fire truck gets a shield  | Testing to make sure that when the user obtains a shield powerup there is a shield around the fire truck | UR_FIRE_TRUCK_POWERUP<br>FR_FIRE_TRUCK_POWERUP | Functional, integrated, user | Ben | Pass |
| <b>Powerup_1.7_USER</b>    | User starting a new game and obtains a <i>refill</i> powerup                            | The user obtains a refill powerup and the amount of water in the fire truck is increased                   | The fire truck gets more water  | Testing to make sure that the fire trucks water supply is increased when running over the refill powerup | UR_FIRE_TRUCK_POWERUP<br>FR_FIRE_TRUCK_POWERUP | Functional, integrated, user | Ben | Pass |
| <b>Powerup_1.8_USER</b>    | User starting a new game and obtains a <i>repair</i> powerup                            | The user obtains a repair powerup and the amount of health remaining on the fire truck is put back to full | The fire truck gets repaired  | Testing to make sure that the fire truck gets repaired when using the repair powerup.                    | UR_FIRE_TRUCK_POWERUP<br>FR_FIRE_TRUCK_POWERUP | Functional, integrated, user | Ben | Pass |
|                            |   |  |   |  |  |                              |     |      |
| <b>Difficulty_1.1_USER</b> | User starting a new game in each difficulty   | Outputting the difficulty settings in the command line   | The difficulty that the game ends up on is the difficulty that the user selects | Testing to make sure that the game reflects the difficulty that the user selects                         | UR_GAME_DIFFICULTY<br>FR_GAME_DIFFICULTY       | Functional, integrated, user | Ben | Pass |
|                            |   |  |   |  |  |                              |     |      |
| <b>GameSave_1.1_USER</b>   | The user opens a new game and moves the fire truck, then saves and opens the game again | The fire truck is in the same location as before the save  | The fire truck is in the same location as before the save                       | Testing to make sure that fire truck locations are saved between saves                                   | UR_GAME_SAVE<br>FR_GAME_SAVE                   | Functional, integrated, user | Ben | Pass |

|                    |  |  |  |   |  |                              |     |      |
|--------------------|--|--|--|---|--|------------------------------|-----|------|
| GameSave_1.2_USER  | The user takes a powerup quickly saves and exits the game, then opens it again                     | When opened again the powerup is still active                            | When opened again the powerup is still active                            | Testing to make sure that the powerup state is saved between games      | UR_GAME_SAVE<br>FR_GAME_SAVE<br>UR_TRUCK_POWERUP<br>FR_TRUCK_POWERUP | Functional, integrated, user | Ben | Pass |
| GameSave_1.3_USER  | The user destroys a ET Fortress and saves the game, the opens it again                             | When the game is reopened the ET Fortress is still destroyed             | When the game is reopened the ET Fortress is still destroyed             | Testing to make sure that the fortress state is persisted through saves | UR_GAME_SAVE,<br>FR_GAME_SAVE  | Functional, integrated, user | Ben | Pass |
| GameSave_1.4_USER  | The user takes damage, uses water, and does damage to an ET Fortress, then saves and opens back up | When the game is opened again, the damage and water values are persisted | When the game is opened again, the damage and water values are persisted | Testing to make sure that entity attributes are persisted between saves | UR_GAME_SAVE<br>FR_GAME_SAVE   | Functional, Integrated, User | Ben | Pass |
| Key for colours:   |  |  |  |   |  |                              |     |      |
| Original Tests:    |  |  |  |   |  |                              |     |      |
| Altered/New tests: |  |  |  |   |  |                              |     |      |