Test ID	Description	Related Requirement	Category	Author	Status (A2)	Status (A3)
test_1.1	Testing that the 'position' and 'topRight' variables are initialised correctly for the Entity Class	UR_instruct_engines	Functional, Unit	Bruno Davies	Pass	Pass
test_1.2	Testing that setPosition() of the Entity class will allow for the boundary input of 0	UR_instruct_engines FR_precision	Functional, Unit	Bruno Davies	Pass	Pass
test_1.3	Testing that passing negative numbers to the setPosition() function of the Entity class throws an IllegalArgumentException	FR_precision	Functional, Unit	Bruno Davies	Pass	Pass
test_2.1	Test to ensure default constructors for the Unit class work as intended via getters	UR_fresh_health FR_auto_repair	Functional, Unit	Bruno Davies	Pass	Pass
test_2.2.1	Test if the Unit class' isDead() function works when health is above 0	FR_enemies_die FR_engine_destroyed	Functional, Unit	Bruno Davies	Pass	Pass
test_2.2.2	Test if the Unit class' isDead() function works when health is zero	FR_enemies_die FR_engine_destroyed	Functional, Unit	Bruno Davies	Pass	Pass
test_2.3	Test if the Unit class' setCurrentHealth() function throws an IllegalArgumentException when health is negative	UR_fresh_health FR_auto_repair	Functional, Unit	Bruno Davies	Fail	Pass
test_2.4.1	Test if the Unit class' setCurrentHealth() function will cap the health input to the max from constructor	UR_fresh_health FR_auto_repair	Functional, Unit	Bruno Davies	Pass	Pass
test_2.4.2	Test if the Unit class' addHealth() function will add health (not hitting max health)	UR_station_refill FR_fortress_recover	Functional Unit	Bruno Davies	Pass	Pass
test_2.4.3	Test if setCurrentHealth() will cap the additional health to maxHealth	UR_fresh_health FR_auto_repair	Functional Unit	Bruno Davies	Pass	Pass
test_2.4.4	Test if addHealth() will take the absolute value	UR_station_refill FR_fortress_recover	Functional Unit	Bruno Davies	Pass	Pass

	of the parameter passed to it					
test_2.5.1	Test if the Unit class' takeDamage() works within a standard range (not 0 or negatives)	UR_instruct_engines FR_engine_fire	Functional Unit	Bruno Davies	Pass	Pass
test_2.5.2	Test if takeDamage() prevents the health from going below zero. .	UR_instruct_engines FR_engine_fire	Functional Unit	Bruno Davies	Pass	Pass
test_3.1.1	Test to ensure default constructor for the Projectile class works as intended via getters	UR_instruct_engines FR_engine_fire	Functional Unit	Bruno Davies	Pass	Pass
test_3.1.2	Test to ensure second constructor for the Projectile class works as intended via getters	UR_instruct_engines FR_engine_fire	Functional Unit	Bruno Davies	Pass	Pass
test_3.2	Test if the Projectile class' setLength() function correctly sets the length with standard values	UR_instruct_engines FR_precision FR_engine_fire	Functional Unit	Bruno Davies	Pass	Pass
test_3.3.1	Testing through parameterized testing that the projectile's hitUnit() class works with both false and true values.	UR_instruct_engines FR_engine_fire	Parameterized Functional Unit	Bruno Davies	Fail	Fail - returns the opposite value to what is expected
test_3.3.2	Checking result of 3.3.1 with a manual test, checking to see if projectiles hit intended target	UR_instruct_engines FR_engine_fire	Manual	Bruno Davies	Fail	Fail
test_3.4	Tests whether the 'random' projectile initialises within the correct range	UR_instruct_engines FR_engine_fire	Functional Unit	Peter Clark	N/A	Pass
test_4.1	Test to ensure default constructor for the Fortress class works as intended	UR_select_level UR_fresh_health FR_auto_repair	Functional Unit	Bruno Davies	Pass	Pass
test_4.2	Test to ensure the aliens are initialised in the correct positions for a given level	UR_select_level UR_fresh_health FR_auto_repair	Functional Unit	Peter Clark	N/A	Pass

test_5.1	Test to ensure default constructor for the Character class works as intended via getters	UR_instruct_engines	Functional Unit	Bruno Davies	Pass	Pass
test_5.2	Testing to make sure setTarget sets the character's target to the given unit	UR_instruct_engines	Functional Unit	Bruno Davies	Pass	Pass
test_6.1	Test to ensure default constructor works as intended for the FireTruck class via getters	UR_refill_warning	Functional Unit	Bruno Davies	Pass	Pass
test_6.2 .1	Testing that the FireTruck class' updateCurrentWater() function works with standard input	UR_instruct_engines UR_refill_warning	Functional Unit	Bruno Davies	Pass	Pass
test_6.2 .2	Testing that updateCurrentWater() allows the water value to remain at 0	UR_instruct_engines UR_refill_warning	Functional Unit	Bruno Davies	Pass	Pass
test_6.2 .3	Testing that UpdateCurrentWater() if in the negative water level sets it to zero	UR_instruct_engines UR_refill_warning	Functional Unit	Bruno Davies	Pass	Pass
test_6.2 .4	Test if the truck moves correctly when commanded	UR_instruct_engines	Functional, Unit	Peter Clark	N/A	Pass
test_6.2 .5	Test if the truck returns to its original position if it is moved equally in all directions	UR_instruct_engines	Functional, Unit	Peter Clark	N/A	Pass
test_6.2 .6	Test if the FireTruck's 'willCollide' function returns true if in range of a given entity	UR_instruct_engines UR_collisions FR_deny_collisions	Functional, Unit	Peter Clark	N/A	Pass
test_6.2 .7	Test if the FireTruck's 'willCollide' function returns false if not in range of a given entity	UR_instruct_engines UR_collisions FR_deny_collisions	Functional, Unit	Peter Clark	N/A	Pass
test_6.3 .1	Test if the truck changes direction to suit a given key being pressed	UR_instruct_engines	Functional, Unit	Peter Clark	N/A	Pass
test_6.3 .2	Test if the truck turns diagonally if two adjacent arrow keys are pressed at once.	UR_instruct_engines	Functional, Unit	Peter Clark	N/A	Pass

test_6.4.1	Test if the truck's getHealth() function returns 0 if its health is a negative value	UR_instruct_engines FR_engine_destroyed	Functional, Unit	Peter Clark	N/A	Pass
test_6.5.1	Test that the truck moves if there are no obstacles in its way and it is not at the edge of the screen	UR_instruct_engines	Functional, Unit	Peter Clark	N/A	Pass
test_6.5.2	Tests that the truck does not move if it is at or beyond the edge of the screen	UR_instruct_engines	Parameterised, Functional, Unit	Peter Clark	N/A	Pass
test_6.5.3	Tests that the truck does not move when there is an obstacle in its way.	UR_instruct_engines	Parameterised, Functional, Unit	Peter Clark	N/A	Pass
test_7.1	Test to ensure default constructor works as intended for the Alien class via getters (not including those in the Entity class)	UR_enemy	Functional Unit	Bruno Davies	Pass	Pass
test_7.2.1	Test if the Alien class' truckInRange() function will set a new target with an in-range **mocked** truck	UR_enemy UR_attack_notification FR_precision	Mocking Functional Unit	Bruno Davies	Pass	Pass
test_7.2.2	Test if truckInRange() will not change the target for a **mocked** truck not in range	UR_enemy UR_attack_notification FR_engine_destroyed FR_precision	Mocking Functional Unit	Bruno Davies	Pass	Pass
test_7.2.3	Test if truckInRange() will set target to null if current target has no health	UR_enemy UR_attack_notification FR_engine_destroyed FR_precision	Mocking Functional Unit	Bruno Davies	Pass	Pass
test_7.2.4	Tests whether the Alien changes its waypoint when the previous one is reached	UR_enemy FR_alien_move	Functional, Unit	Peter Clark	N/A	Pass
test_7.2.5	Tests whether the Alien returns to its initial waypoint once all waypoints have been reached	UR_enemy FR_alien_move	Functional, Unit	Peter Clark	N/A	Pass
test_7.2.6	Tests whether the alien moves towards its given waypoint	UR_enemy FR_alien_move	Functional, Unit	Peter Clark	N/A	Pass

test_7.2.7	Tests whether an Alien initialised close to the waypoint updates, and then moves towards the next waypoint	UR_enemy FR_alien_move	Functional, Unit	Peter Clark	N/A	Pass
test_7.2.8	Tests whether an Alien moves towards the Fire Station along the x-axis if at the correct y position	UR_enemy FR_alien_move FR_move_towards_station	Functional, Unit	Peter Clark	N/A	Pass
test_7.2.9	Tests whether an Alien moves towards the Fire Station along the y-axis if at the correct x position	UR_enemy FR_alien_move FR_move_towards_station	Functional, Unit	Peter Clark	N/A	Pass
test_8.1.1	Test if the pipe rotates by 90 degrees when the 'rotate' method is called	UR_minigame FR_minigame_play	Functional, Unit	Peter Clark	N/A	Pass
test_8.1.2	Test if the pipe rotation returns to 0 when it reaches 360	UR_minigame FR_minigame_play	Functional, Unit	Peter Clark	N/A	Pass
test_8.1.3	Test if the 'isCorrectRotation' function returns true when the rotation is correct	UR_minigame FR_minigame_play	Functional, Unit	Peter Clark	N/A	Pass
test_8.1.4	Test if the 'isCorrectRotation' function returns false when the rotation is incorrect	UR_minigame FR_minigame_complete	Functional, Unit	Peter Clark	N/A	Pass
test_8.1.5	Test if the pipe recognises when it is clicked in range	UR_minigame FR_minigame_complete	Functional, Unit	Peter Clark	N/A	Pass
test_8.1.6	Test that the pipe does not respond when it is clicked out of range	UR_minigame FR_minigame_play	Functional, Unit	Peter Clark	N/A	Pass
test_9.1.1	Test the start screen appears when opening the game for the first time	UR_start_screen UR_music FR_play_music	Manual	Whole Team	Pass	Pass
test_9.1.2	Test start screen comes back in correct position when coming back from Level select page	UR_start_screen UR_music FR_play_music	Manual	Whole Team	Pass	Pass
test_9.1.3	Test start screen comes back in correct position when coming	UR_start_screen UR_music FR_play_music	Manual	Whole Team	Pass`	Pass

	back from settings page					
test_9.1.4	Test start screen comes back in correct position when coming back from instructions page	UR_start_screen UR_music FR_play_music	Manual	Whole Team	Pass	Pass
test_9.1.5	Test start screen comes back in correct position when coming back from credits page	UR_start_screen UR_music FR_play_music	Manual	Whole Team	Pass	Pass
test_9.2.1	Test that the next levels become unlocked when finishing all levels.	UR_start_screen UR_select_level UR_fresh_health UR_music FR_auto_repair FR_new_level FR_play_music	Manual	Whole Team	Pass	Pass
test_9.2.2	Test that the next levels do not become unlocked when finishing level 1 and 2 but failing 3	UR_select_level UR_fresh_health UR_music FR_end_game FR_auto_repair FR_new_level FR_play_music	Manual	Whole Team	Pass	Pass
test_9.2.3	Test that level 3 does not become unlocked when finishing only level 1 and failing 2	UR_save_load_quit UR_select_level UR_fresh_health UR_music UR_collisions FR_auto_save FR_end_game FR_auto_repair FR_new_level FR_play_music FR_security	Manual	Whole Team	Pass	Pass
test_9.2.4	Test that the next levels become unlocked when completing some levels, closing the game, and then opening it again	UR_start_screen UR_select_level UR_fresh_health UR_music UR_collisions UR_instruct_engines FR_save_quit FR_auto_repair FR_new_level FR_play_music	Manual	Whole Team	Pass	Pass
test_9.2.5	Test that the next level does not become unlocked if a level is quit halfway through playing	UR_save_load_quit FR_auto_save FR_save_quit	Manual	Whole Team	N/A	Pass

test_10.1	Tests that the pause screen appears when the Escape button is pressed	UR_pause FR_pause_inlevel	Manual	Whole Team	N/A	Pass
test_10.2.1	Tests that the user completes a level once the health of the enemy base has been reduced to 0	FR_end_game	Manual	Whole Team	N/A	Pass
test_10.2.2	Tests that the user fails a level if all of their Fire Engines have been destroyed	FR_end_game	Manual	Whole Team	N/A	Pass
test_10.2.3	Tests that a 'win' screen appears when the user completes a level	UR_end_game_screen FR_end_game	Manual	Whole Team	N/A	Pass
test_10.2.4	Tests that a 'lose' screen appears when the user does not complete a level	UR_end_game_screen FR_end_game	Manual	Whole Team	N/A	Pass
test_10.3.1	Tests whether a 'warning' icon appears when the selected FireEngine's water level is below 20	UR_refill_warning	Manual	Whole Team	N/A	Pass
test_10.3.2	Tests whether a destroyed Fire Station prevents an engine from repairing or refilling	UR_station_refill FR_no_refill	Manual	Whole Team	N/A	Pass
test_10.4.1	Tests whether a warning icon appears when the fire engine is 15 seconds away from being destroyed	UR_attack_notification	Manual	Whole Team	N/A	Pass
test_10.4.2	Tests whether the user is unable to move through obstacles	UR_collisions FR_deny_collsion	Manual	Whole Team	N/A	Pass
test_10.4.3	Tests whether a timer appears on the screen as the game is being played	FR_display_timer	Manual	Whole Team	N/A	Pass
test_10.5.1	Tests that the minigame begins when the user is in range of the Fire Station	UR_minigame FR_open_minigame	Manual	Whole Team	N/A	Pass
test_10.5.2	Test whether the Engine only refills when the Minigame is complete.	UR_minigame FR_minigame_outcome	Manual	Whole Team	N/A	Fail - if the game is completed once, the

						Engine refills when the game is played again regardless of the outcome