

For this milestone, our tests are oriented around testing many of the required features of the requirements. This includes heavier testing of both the backend enemy classes as well as the tower functions. These were chosen because they are the areas that are less visually obvious or accessible as the project stands. We have 10 additional unit tests, with each member contributing at least 2. These tests were written in a semi-synchronous fashion with the code and help to ensure that these newly implemented functionality do not break in future development.

The tests and purposes are as follows

EnemyTests.java -- Tests relating the enemy agents

testAddEnemy - Tests to make sure that the path can store an enemy

testRemoveEnemy - Tests to make sure that the path can remove an enemy

testGetEnemy - Tests to make sure the path can return its stored enemy

swapOutEnemy - Tests to make sure the path can swap out an enemy

getGoblinImage - Tests to make sure GoblinEnemy has correct image location

removePathImage - Tests to make sure the path can remove goblin image

setEnemyHealth - Tests to make sure the enemy health is adjustable

TowerTest.java -- Tests relating the Tower functions

setCost - Tests to make sure tower cost can be set

setCostInvalid - Tests to make sure invalid costs can't be set

setDescription - Tests to make sure the description can be properly set