

M4 Testing Deliverable

These tests cover common use cases for the Enemy class, including situations of adding, moving, and removing enemies on the Path and grid. These cases were chosen for coverage since most other functionalities of the Enemies stem from these three basic processes in some combination. All new tests were found in EnemyTests.java.

testAddEnemy – Tests adding an enemy. Verified by attempting to put an enemy on the Path and checking that an enemy was added.

testRemoveEnemy – Tests removing an enemy. Verified by attempting to add and remove an enemy from the Path and checking that the enemy is no longer on the Path.

testGetEnemy – Tests retrieving the current enemy on the Path. Verified by adding an enemy to the Path and testing that the function outputs the added enemy.

swapOutEnemy – Tests changing which enemy is on the Path. Verified by adding an enemy to the Path, then another, and making sure that the function outputs the new enemy.

getGoblinImage – Tests that goblins are displayed using the correct image file. Verified by checking for equality with the correct filename.

removePathImage – Tests that a Path with a removed enemy is displayed correctly. Verified by checking that the current image of the Path with a removed enemy is the Rectangle image.

setEnemyHealth – Tests setting an enemy's health. Verified by attempting to set the health and checking that the current health matches this value.

placeEnemyGrid – Tests placing an enemy on the grid. Verified by attempting to place an enemy on the grid and checking that the placed position contains an enemy.

moveEnemy – Tests moving an enemy from one spot to another on the grid. Verified by placing the enemy at one position, then another, and checking that the new position contains the enemy.

removeEnemyGrid – Tests removing an enemy from the grid. Verified by adding and removing an enemy from the grid and checking that the enemy no longer exists at its old position.