Test controller

Controller is the central part of Model-view-controller, as such we should have confidence of their behaviors as what we intended for our program. Testing the controller can provide us with this confidence and help us to discover the errors before they reach production. And our tests will be more focusing on controller’s responsibilities because we want our controller to be minimal so it is important avoiding to have the controller have unnecessary responsibilities. Test the controller will be simply testing how the controller behaves based on valid or invalid inputs, test controller responses based on the result of the business operation it performs.

Test views

In the Model-View-Controller, the view encapsulates the presentation details of the user’s interaction with the program by taking in input and sending that information to the controller. We focus on testing two parts: testing teamSelectionPanel and GamePanel. We need to test teamSelectionPanel because it interacts with the robot librarian, without having it tested the game will not be operated properly. The functionality will be tested in the Controller interface; however, we still need to check the proper information displayed in the GUI.

To test our game Battle Bots we will have to approach each section in the MVC differently. Since we are using Test-Driven development we will be separately designing tests for our Model, View and Controller which we will implement before any of the game code. This means we will be able to minimize the amount of code required for the game to function correctly. Due to the nature of our design certain aspects of the game will not be testable until other classes that they rely on have been created. For certain classes we can avoid this by generating dummy classes, which are simplified classes used only for testing, which will be implemented if the required class has not yet been finished. In other cases, we will produce mock classes, which are classes where variables are set to permanent values, to allow easier testing of specific classes.