Week 3 Testing Plan:

Bug Fixing (Random Stopping): Up to 10-15 minutes of initial testing to see where the bugs are occurring within the game (more testing time will be needed as changes will be made). The input for “in-game” bugs would be the player/user controlling the character and having it interact with the environment to see where in the game the bug happens, be it in one area or on all parts of the level. The “output” would be the player being unable to move as the character becomes “stuck” on a part of the level and cannot move past it (getting caught in the bug). We’ll test the bug by changing the level so each of the pieces fit better as we suspect it is a placement issue, and will repeatedly adjust it after running through each change to see if it’s finally fixed.

Bug Fixing (Build Issue): The game at the moment can’t be “built” or “downloaded” as an actual game application due to some error in the coding. We’ll first test to see if we can build it or not, and if we can’t, we’ll look through the code and use online resources to check any errors we come across. The input would be us trying to build the game, and the output would be whether or not the game application is built. If the game is successfully built, we’ll see how it runs outside of the Unity Editor vs. inside of it.

Bug Fixing (Slime Interaction): We also need to test how the character interacts with enemies, like the slime. By having one correct enemy interaction done, it’ll lay the foundation for future enemy interaction. We’ll test for a good 10-15 minutes to see what happens as we move and collide with the slime. Our inputs would be the character colliding with the enemy and the output would be if it “interacts” with the character the way we want it to, whether it repeatedly damages the player or if it just pushes it back.

Animation Testing: Since the running animation and weapon animation often go hand-in-hand, we would test them together. We would allot about 5-10 minutes in total for testing, to see how well the animation flows with the gameplay and how the character interacts with other objects in the game. After seeing how well the animation flows, we’ll add more frames or change parts of the animation (or make a new one) to see if it fits better for the character. The input will be the character interacting with the level and the output will be how smooth the character’s animation is, whether it “flows naturally” or not with the game.