Detailed Sprint Breakdown And explanation of FSM

**Week 1**

* Formed the group with Nolan Hunt, Brian Icely, Christian Balagtas
* Created the High Concept
* Decided on a scrolling shooter style game
* Named Christian as Scrum Master

**Week 2**

* Decided some of the basic mechanics that would be in the game
* Confirmed that a scrolling shooter game was the game we would be doing and that we would make it in the Unity engine
* Decided on at least 3 levels and possibly a finished game for the end goal of the semester

**Week 3**

* Started on a UML document for the game to plan out mechanics

**Week 4**

* Did a short scrum to decide what mechanics should be kept and which ones we can remove
* Started a Github repository to store our progress

**Week 5**

* Finalized the UML and started some of the coding process
* Put the UML document on Github
* Coding process was underway and there were no problems currently

**Week 6**

* Gave roles out
* Nolan – Programmer
* Brian – Quality Check
* Christian – Art Lead

**Week 7**

* Did a short scrum to check our progress
* Changed our end goal when 3 levels was seeming less realistic

**Week 8**

* Reading Week
* Gave ourselves a short break for the week

**Week 9**

* Pushed some code on Github
* Decided we were slacking and needed to pick up the slack
* Also decided on finishing our one level as the end goal
* Code for player movement was created

**Week 10**

* Most of the terrain art was being worked on
* Brian made some scripts that were merged with Nolan’s script

**Week 11**

* Terrain art was being finalized
* Code for shooting mechanics was created

**Week 12**

* Code for AI was created

**Week 13**

* Spawn code was created
* Brian’s script was taken out because it didn’t compile with Nolan’s code

**Week 14**

* Created a Finite state machine as well as tests
* Pushed the terrain onto Github
* Needed to crunch time in order to get what we could done

**Week 15**

* Finalized whatever we had on the game
* Decided we didn’t have a realistic goal at the end thinking we’d have a finished product at the end of the semester but we worked hard and have a half decent demo first stage  
    
  FSM:  
    
  The state machine we created for our game was one that was built for our enemy. The state machine consisted of three states, Attack, Chase, and Idle. When the enemy spawns it is set to idle and then checks to see if the player is within it's chase distance. When it is in that distance the state switches to Chase and the enemy runs towards the player, once the enemy becomes visible on the screen it switches to it's Attack state. In this state it continues heading towards player and begins firing its weapon.   
    
  The state machine well simple is ideal for our game it fits all of the criteria for a simple basic enemy in a top down shooter. It can also be repurposed for more complex enemies adding different variations on the three states to make those enemies different and more interesting.