# Kill-a-Round

## Log 001: (11:30am - 15/02/2020)

Session Time: 5 hours

**Goals:**

* NPC “Runner” implemented with a Behaviour tree.
* Runner runs towards the player.
* Doors open.

**Details:**

* Added “AnimStarterPack” to the project to have premade animations.
* Implemented an AI controller “NPC\_AI” and a behaviour tree, at the moment the “runner” just finds a random location then navigates towards it.
* Implemented an Animation blueprint that switches from idle to moving depending on the speed.
* Copied and edited the “idle pistol” animation to lower the hands.
* Copied and edited the “sprint\_rifle” animation to lower the hands.
* Ai perception for sight added to the runner.
* Runner now runs towards the player once he is seen.
* Runners now play the attack animation, when the hitbox is overlapped
* Implemented a door which swings open when the player presses e.

## Log 002: (10:50am – 16/02/2020)

Session Time: 2 hours

**Goals:**

* Polish the Runner
* Open the runner up for merging

**Details:**

* Created the sword in blender.
* Imported it into Unreal Engine 4, with a 0.05 scale. (tried at 1, but was too big)
* Created slots on the mannequin for the weapons.
* Attached weapons to those slots.
* [Bug: 001]: the runners can jump off each other? Probably due to clumping together.

# Living Weapon

## Log 003: (8:30am – 20/02/2020)

Session Time: 2 hours

**Goals:**

* Get the player moving
* Player is made up of multiple parts

**Details:**

* Created core sprite
* Created track sprite
* Created a new player class and each base parent class for each type of part
* Created two sockets “North” and “South” on the core sprite for testing with attaching other paper actors onto the sprite.
* Player is now made up of the core and a track, at run time.
* [Bug: 001] Player now glitches through the floor because of gravity?

## Log 004: (5:54pm – 23/02/20)

Session Time: 2 hours

Goals:

* Fix Collision and motion.

Details:

* Majority of Bug 001 was due to Ue4 still accepting a 3rd dimension. The player sprite was just behind the platform and therefore never collided.
* Bug 001 Cleared, with “tank parts” not colliding with other parts.
* Bug 001 not cleared.

## Log 005: (2:00pm – 25/02/20)

Session Time: 4 hours

**Goals:**

* Fix Collision and motion.

**Details:**

* After spending four hours bashing my brain over google and unreal figuring out the issue with bug 001. I found a single reddit post that referenced my issue: <https://www.reddit.com/r/unrealengine/comments/5dsv4v/replacing_capsule_collision_with_mesh_collision/>
* To fix bug 001, I need to make a player class by scratch.

## Log 006: (8:00am – 26/02/20)

Session Time: 3 hours

**Goals:**

* Allow parts to be added on via a call in the parent class.
* Sprite can handle multiple additions

**Details:**

* Added William’s sprites to my branch to allow easier merging.
* Added the “West”, “East” and “Center” sockets to the core sprite.
* In the Parts parent class I added the “Add Existing Part” function that takes an existing part and adds it to a non-socketed, socket.
* In the parts parent class I added the “Add New Part” function that creates the part before calling the “Add Existing part”. Note: the new part is made at <0, 0, 0> and if “add existing fails” it destroys the new part.
* Started working on making a display (mainly for debugging purposes), for dynamically creating a tank at run time. Although the code could be used in future use for editing the players tank.

## Log 007: 1:30pm (26/02/20)

Session Time: 3 hours

**Goals:**

* Add a debug window to add and remove parts

**Details:**

* Implemented a breath-first search to get an array of all parts to be rendered.
* Created a child widget to be created at run time for each part.
* Issue: When trying to allow the player to add parts, I need to link up all parts appropriately. Currently the design is like a tree, where the root node is the core part. Each node has 4 branches {North, South, East, West}, however the clients expressed a that it would check if a part is still attached. I need to be able to make a cycle, removing the tree like structure.
* Minor Issue: infinite loop within the Breath-first search, fixed with restructuring the function. [Fixed]

## Log 008: 8:15pm (27/02/20)

Session Time: 3 hours

**Goals:**

* Add a debug window to add and remove parts

**Details:**

* Added a function “LinkToMap” that gives a part a position in the map then links all the parts around that position to make a net-like data structure.
* Allowed Display of mouse when debug is pressed.
* Game now adds parts, but display doesn’t refresh, for another session
* Small bug with display not allowing additions above or left.