Design Doc Milestone I

**Problem:** Body parts sprites coordination/Rotation

**Objectives:** Allow users to have the option to define movements for a fighter sprite without drawing more sprite images. In other words, split a sprite into multiple coordinated sprites each representing a body part/weapon. Each body part/weapon should be completely customizable from image, position relative to body(its owner fighter sprite), damage dealt when collided with another fighter, attack animation, etc.

**Alternative Designs**: From my conversation with UTA Tanner, there were roughly 3 different ways to approach this issue:  
 1) having pre-made sprites for each attack move, the traditional way. When attacking an opponent, just create a temporary sprite that detects collision on top of weapon/body parts, destroy the sprite once the attack is finished. This is a more simplistic approach to trying to figure out how limbs should rotate/interact with each other, but it sacrifices the potential to detect damage from different parts of a body, which could take damage in different ways. (i.e. attack to the head may have a higher damage multiplier.)

2) Split fighter into a tree of children nodes, each a separate sprite. This allows for more flexibility in designing a fighter character, but the downside is the rotation between limbs is a very difficult problem to resolve.

3) A combination of the two, but also has the worst of both worlds.

**Features** **Implemented this week**:

-LimbNode.java represent body parts sprites that make up a fighter sprite. Each node has references to their parent node, to all of its children nodes(each limb can have one or more children nodes), its own sprite image, and relative position to its parent node.

-BodyTree.java is the rough draft of fighter sprite, the tree contains 3 methods so far: 1) add node 2) recursively render a body part (LimbNode sprite) when render is called in the main game 3) recursively move a body part (LimbNode Sprite) when a user input is received

-These two classes accomplished:

1) Allows for splitting a sprite into multiple sprites, each part is completely customizable from image to relative position to body

2) Each limb and their children limb(s) are recursively rendered

3) All body parts are in sync/coordinated in motion

**Challenge this week**: I attempted to tackle the “rotation of body parts” issue, but a few unforeseen problems came up:

1. I attempted to rotate the image through using Java’s built-in Graphics2D rotate method. However the rotation is confined in the original size of the image, in other words, when I rotate a sprite, the part of a sprite that spills over the original frame gets cut off. In addition, as I animate the rotation, I often got outOfMemory error from Java, meaning it’s perhaps not the way to go about animation body part rotations.
2. I also tried to use GTGE’s built-in ImageUtil.rotate() method, for some reason the sprite did not rotate, just became really blurry.

**Features to be Implemented**:

-Rotation of limbs and their respective parent limbs

-Response to Collision from individual body parts. Each body part should know what damage it took when in collision with another fighter, pass it up to the fighter, and receive rotation instructions from the fighter.