**Ah My 2 Legs: Pseudocode**

Slingshot Script  
**General info:**  
If player holds down mouse 1 and is within a certain range of player character. Then locate its position relative to the player character (rotation on the ‘Z’ axis).

Then calculate distance between mouse cursor and player, then draw a line from the center of the player to the cursor.

When player lets go of mouse 1, check if cursor is still within range of player.   
If it is, take the distance between mouse cursor and player, and add it to the following calculation.

measuredDistance \* velocityModifier(10 for example) == amplifiedDistance.

Then take ‘addedVelocity’ and apply force to player character in the direction of the cursor, as the mouse 1 button was let go.  
  
  
**Pseudocode:**

- Start  
- If GetMouse0Down is true, then  
 - If mousePosition is within playerRadius, then  
 - Drawline(white) from playerPosition to mousePosition, updating every frame  
 - Calculate distance between playerPosition and mousePosition into store into amplifiedDistance & update value every frame that GetMouse0Down is true.  
 - Set canSling to true  
  
 - else if mousePosition isn’t within playerRadius, then  
 - Drawline(Red) from playerPosition to mousePosition, updating every frame  
 - Set canSling to false  
- If GetMouse0Down was true, and is false and canSling is true  
  
 - If canSling is true, then  
 - Set slingshotFired to true  
 - Apply amplifiedDistance to player character, in the direction of mousePosition from playerPosition  
 - Set canSlign to false  
 - Set slingshotFired to false

Energy GUI Script  
**General info:**  
This meter located at the bottom-middle of the screen. This meter will be full at the start of the game. When the player launches themselves with the slingshot, the meter will have some percentage consumed.  
The meter will constantly refill itself over time at a slow rate.  
The meter also won’t be consumed if certain prerequisites aren’t met (available energy, canSling == true ect.)

**Pseudocode:**- start  
- on FixedUpdate  
- If ‘slingshotFired’ is true, then  
 - If there is enough meterPercentage equal to (amplifiedDistance \* 2)?, then  
 - Subtract (amplifiedDistance \* 2) from meterPercentage  
 - If there isn’t enough meterPercentage equal to (amplifiedDistance \* 2), then  
 - Terminate  
  
- if false, then  
 - If meterPercentage isn’t equal to 100, then  
 - Add 5 too meterPercentage every second  
 - if meterPercentage is equal to 100, then  
 - Terminate

Flowchart feedback