Gameplay Programmer Test - Combat Presentation

Overview

The combat is inspired by the free flow combat system of games like Shadow of Mordor and the Arkham games. https://www.youtube.com/watch?v=9aq30UvqmRI

The goal for the player is not to memorize complex inputs and combo chains. The combat is more based on timing and activating the right defensive move at the right time. Making the player feel powerful in the way he handles a large group of enemies.

All the controls are touch based and shouldn't rely on virtual buttons/stick.

What we need to see in the prototype:

- Input system
  - Attack (tap right side of the screen)
  - Dodge (tap left side of the screen)
  - Target Selection (Swipe)
- 2 types of melee enemies
  - Weak enemy
  - Strong enemy
- Animation system
  - Character must use the right animation based on how far away he is from the enemy.
- UI
  - Health Bar
  - Combo Meter
- Camera
  - You can use the camera described below or use a more simple system like a top down camera.

Full detail below

Playable character Movements

- **Attack New Target:** Movement + First attack (swipe)
  - Allow the player to select/lock a target. The character will always close the gap with his opponent with this attack.
  - Does low damages
- **Basic attack** (tap Right side of the screen)
  - Allow the player to hit the opponent until he's down. The PC will close the remaining gap between him and his target.
  - Does High damages
  - The last attack of the combo (4^{th} attack) should always knockdown the target.
- **Dodge** (tap Left Side of the screen)

- Allow to escape an incoming attack
- This movement can interrupt any movement currently performed by the PC.

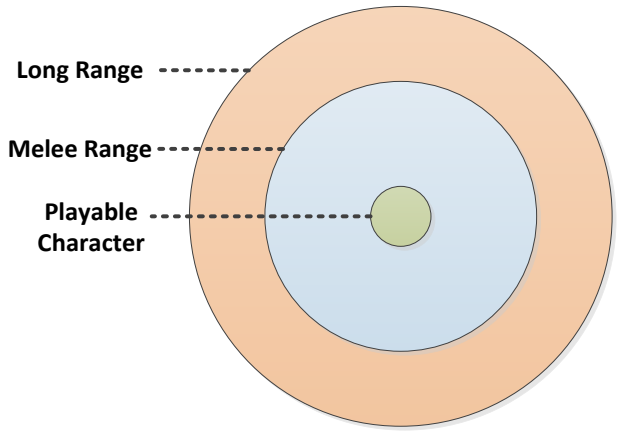
Parameters for designers

Main character's stats

1. **Health**: 150
2. **Attack Damages**: 15 (every attack does the same amount of damages)

Attack Range

Based on the distance, the playable character will automatically pick the most appropriate attack (melee or ranged).



In the prototype, we'd like 3 ranges:

- **Melee**
  - Max range settings: 2.12
  - Animations: Atk_Melee00a to Atk_Melee03a
- **Mid**
  - Max range settings: 3.51
  - Animations: Atk_Mid00a to Atk_Mid03a
- **Long**
  - Max range settings: 6
  - Animations: Atk_Range00a to Atk_Range03a

Combo meter

As the character performs hits and dodges, **his combo meter rises**.

The **combo meter will be reseted under certain conditions**:

- If the PC is hit
- If the PC doesn't attack within a certain time
- If the PC performs a dodge at the wrong time.

The combo meter **doesn't have any impact on the** actual **gameplay**.

Parameters for designers

1. **Combo reset duration**: amount of time before the streak and combo meter are reseted

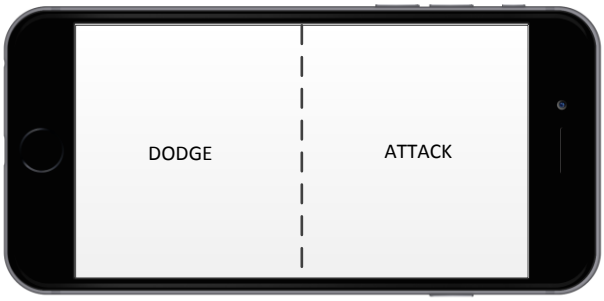
2. **Attack combo meter bonus:** amount by which the streak and combo meter are increased with a basic attack

Controls

Overview

Controls are entirely touch based.

- **Tap** (Left Side of the screen): Dodge
- **Tap** (Right Side of the screen): Attack on current Target. Due to hold and swipe inputs, the attack should be activated on release of the finger.
    - If the player as no target, the tap should automatically target the closest enemy (just like a swipe)



- **Swipe** (anywhere): Movement towards target + attack. Allow the player to select a new target.
    - This should behave like the first attack of the combo. The character moves close to its target and hits him.
    - If the player performs a tap afterward, the character should continue his combo.
    - If the player performs another swipe, the character should look for another target.

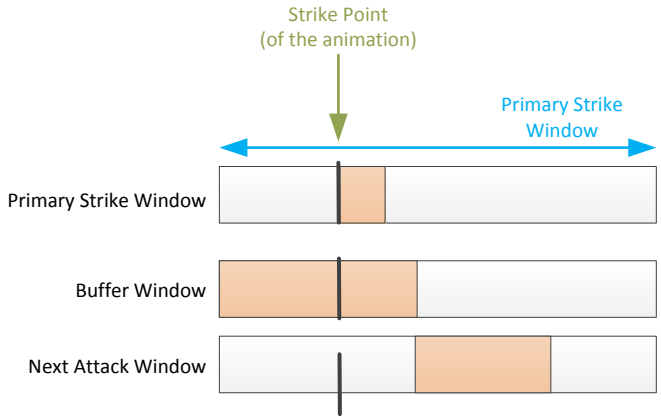
Camera: The player has no control over the camera.

Parameters for designers

- **Min swipe input length**: distance that must be travelled by the finger to be registered as a swipe

Input Activation Windows

Core setup of the combo activation windows are as follows.



Primary Strike Window

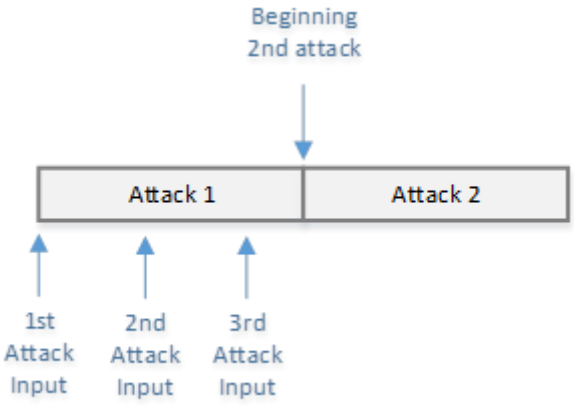
This is the moment a character hits the target he's facing.

Buffer Window

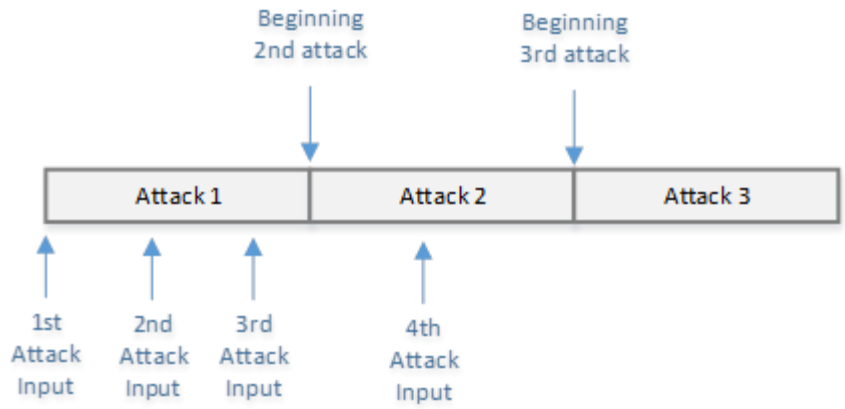
Any attack input within this should be stored and then performed when the next attack window opens. The buffer window opens when the animation starts, and ends when the next attack window opens. For the last attack of a combo, the buffer is moved later, before the new combo window.

Reminder of the Buffer window

Make sure that when I press an input while an attack/movement is currently active, only my last input is buffered. So for instance, if I press the attack button 3 times before the beginning of the 2nd attack, you should perform the second attack based on the last input of the player (swipe or tap) and ignore any previous input.

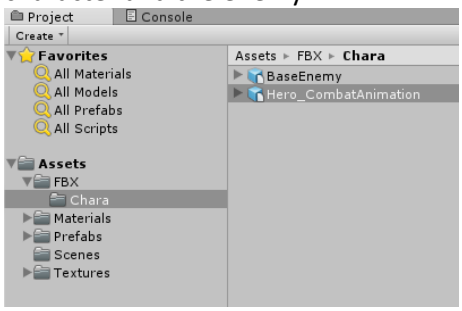


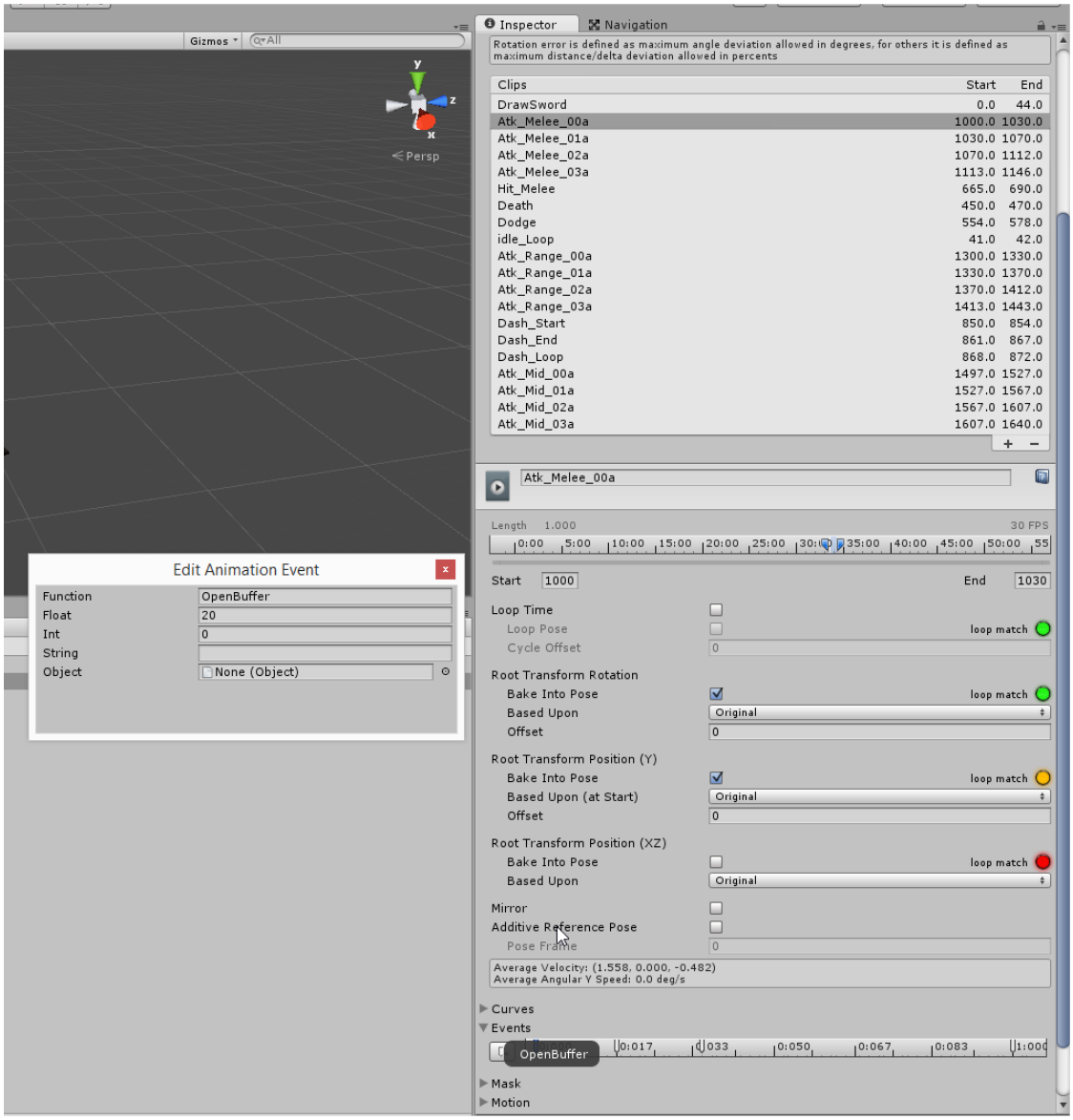
Only the last input should be recorded (in this scenario that would be the 3rd input).
The third attack should be played if there's an input during the 2nd attack (4th input occurs during second attack).



Next Attack Window
Any attack input performed within this window should perform the next attack in your combo. If the player presses an attack input after the closure of this window, the character should perform the first attack of the combo.

Side note (for the test)
All animations timing are already setup. Check the events of each attack animation for both the main character and the enemy.





Events for the Playable character's animations:

- Open Buffer: Input is registered until close buffer
- Close Buffer: Following attack should start from this point
- Strike Point: Moment the target is hit
- Knockdown Strike Point: Moment the target is knocked down on the floor (only on last attack)
- End Attack: Moment the combo is reset

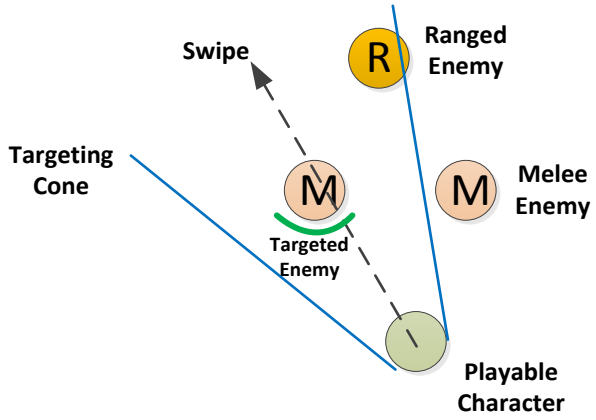
Events for the Enemies' animations:

- Open Parry: First moment the player can dodge the attack
- Close Parry : Last moment the player can dodge the attack
- Strike Point : Moment the target is hit
- End Attack : Moment the attack ends

Swipe Targeting

Overview

The player can use a swipe to select a target. The swipe will then allow the game to select the most interesting target based on several parameters.



- **Playable Character**
  - Character controlled by the player
- **Melee enemy**
  - Enemy that can only attack at close range.
- **Ranged Enemy**
  - Enemy that can only attack at long range.
- **Swipe**
  - Direction of the swipe performed by the player.
- **Targeting Cone**
  - Projected area within which the target should be selected.
- **Targeted Enemy**
  - Enemy selected as the new target of the swipe based on the targeting rules

Parameters for designers

1. **Targeting cone length**
2. **Targeting cone angle**

Targeting rules

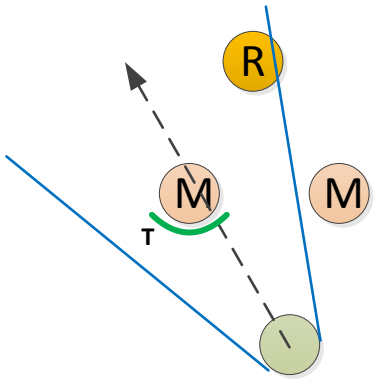
Within the targeting cone, **the biggest threat** should be targeted.

If there are **several characters of the same threat** within the cone, **the closest one to the swipe** should be selected.

If there are **several characters close to the swipe**, **the closest one to the playable character** should be selected.

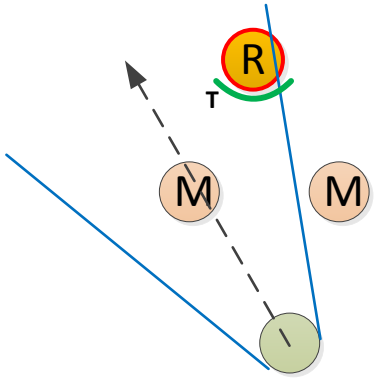
User cases

Case 1: No enemy is attacking



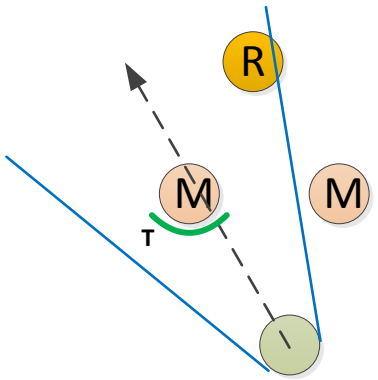
Both **ranged and melee characters are vulnerable** and both have the **same threat level**. The **closest one** to the swipe is targeted

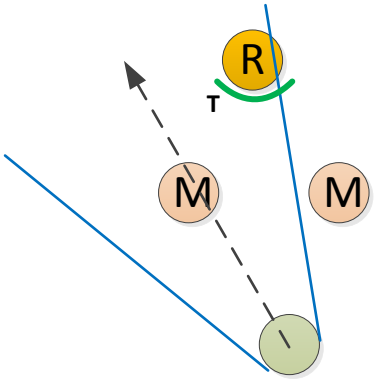
Case 2: An enemy is attacking



The **ranged character is about to attack**, his **threat level** is the **highest** within the targeting cone. As a result, he's the one targeted.

Case 3: A swipe is perform twice in the same direction





The player performs two swipes in the same direction. The same character can't be selected twice.

1. The **first swipe would select the melee enemy** since he has the same level threat level as the ranged enemy.
2. The **second swipe will find the next best target** that is not currently selected.

Enemies

Parameters for designers

- **Health**: amount of health
- **Attack Damages**: amount of damage the attack deal
- *Characters placement:* variables that define the position of an enemy
    - **Melee Range Min**: Min distance from which the player can attack with melee attacks
    - **Melee Range Max**: Max distance from which the player can attack with melee attacks
- **Melee attack distance max**: if a melee enemy is beyond this distance, he will move closer to start his actual attack.
- **Knockdown duration**: Amount of time the character will spend on the ground before standing up.

Enemy AI

Possible status

- **Idle**
    - The character is waiting for his turn to attack and is vulnerable to attacks.
- **Moving**
    - The character is replacing himself to an attacking position
- **Attacking**
    - The character is currently attacking.
- **Knocked down**
    - The character is lying on the ground, can't be attacked
    - After a certain amount of time, the character will stand up.
- **Dead**
    - The character is defeated.

Attack Token

Enemies attack one at a time. The next enemy to attack is selected based on the following aspects:

- **Is the enemy in 'sight' of the camera?**
    - Only characters within the 'attack zone' are allowed to start an attack.
- **What's the attack priority level of the character?**
    - A special enemy like a ranged enemy will have a higher priority than a basic enemy
- **Has the enemy already attack?**
    - If the enemy has just attacked, he can't attack again unless he's the only character able to attack.

A new enemy should attack 2 seconds after the current enemy has stopped attacking.

Parameters for designers

1. **Characters**
    - Attack priority level

Attack priority

The game will select an enemy to attack based on the 3 aspects:

- **The default attack priority**: Shielded enemy will have a higher priority than a basic enemy
- **The distance from the player**: If there are multiple enemies with the same priority level, the game should pick the enemy the closest to the player. Some characters such as the range enemy can override this distance requirement.
- **Has this enemy type already attacked**: If an enemy of the same class has already attack, they should wait a certain amount of rounds of attack until he can attack again.
- **Has the enemy already attacked**: If an enemy has already attack, he should wait a certain amount of rounds of attack until he can attack again.

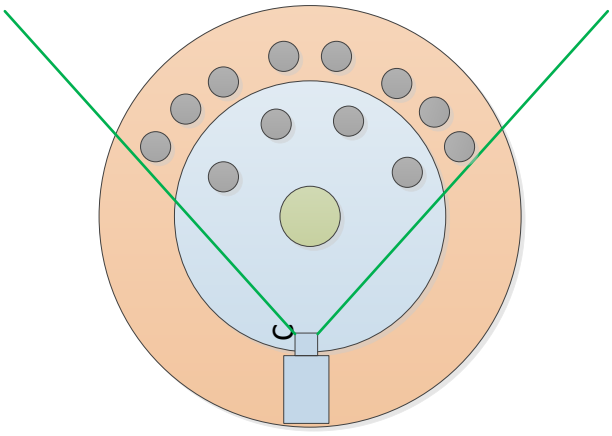
Parameters for designers

- **Default attack priority:** Priority of an enemy to attack
- **Round of attacks before character type reset**: amount of attacks performed by other characters, this type of character will have to wait until they can attack again.
- **Round of attacks before reset**: amount of attacks performed by other characters, the character will have to wait until he can attack again. Typically this value will be higher than the one linked to the character type.

Enemy placement

Enemies will position themselves based on the following point:

- **Placement points**



Multiple points are placed around the PC within the camera field of view. These are the various positions, the enemies can move towards.

As a reminder, only enemies within the camera field of view should be able to attack.

2 Enemy Types

Both enemies attack **at close range.**

Weak enemy

- **Health**: 60
- **Attack Damages**: 10
- **Threat level**: 1

The character should die with 8 attacks from the MC (2 full combo)

Strong enemy

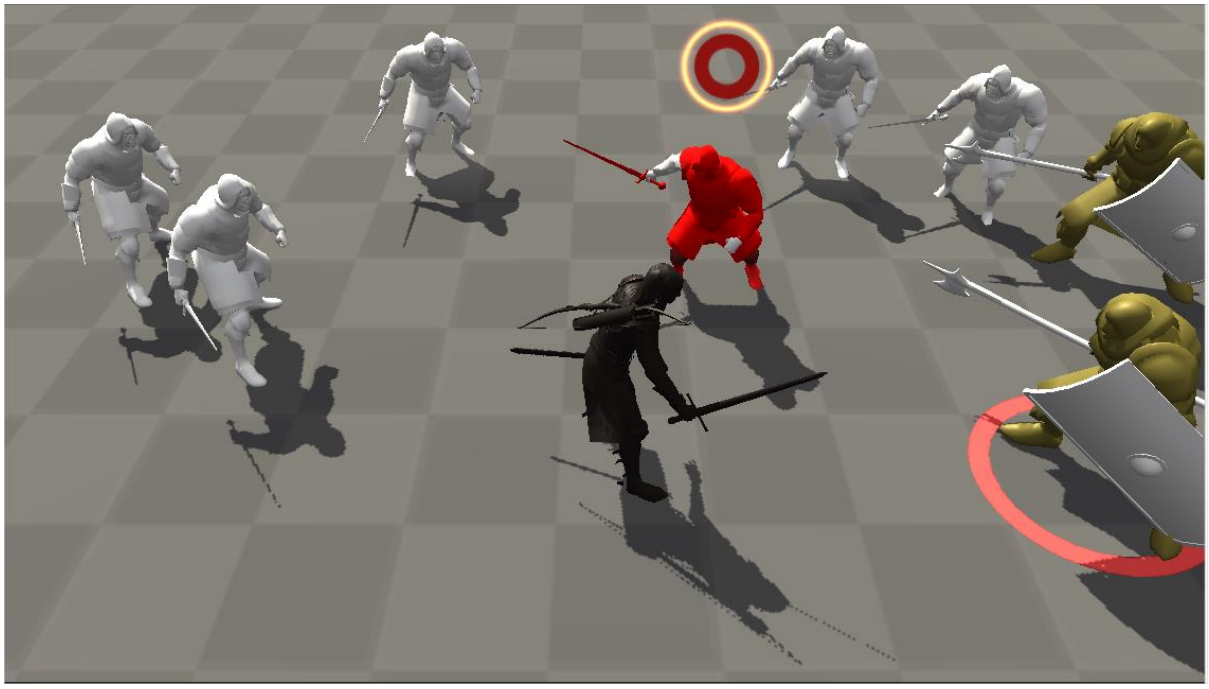
- **Health**: 90
- **Attack Damages**: 30
- **Threat level**: 2

The character should die with 12 attacks from the MC (3 full combo)

Camera

Overview

The player has **no direct control over the camera**.



The main character should always **keep the same position on screen**.

The camera should **reposition itself automatically based on the amount of enemies on screen and off screen**. If there are less than X enemies on screen and more enemies off screen, the camera should be repositioned.

When repositioning itself, the camera should turn around the MC.

If an enemy is already attacking, the camera shouldn't try to reposition itself.

Camera parameters

These are the parameters that should be taken into consideration

- **Amount of enemies on screen**
- **Amount of enemies off scene**

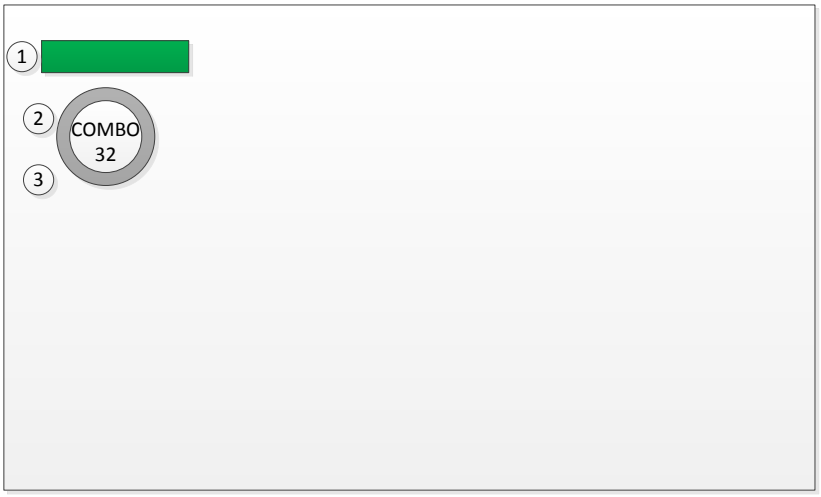
Parameters for designers

- **Reposition threshold:**
    - Ìf the the amount of enemies on screen is lower than the reposition threshold, the game should check if there more enemies off-screen than on-screen. If this is the case, the camera should reposition itself

UI & FX

Overview

The **UI** requires **most of these elements**.

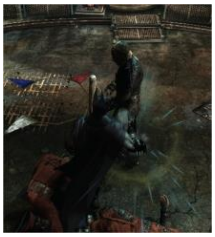


1. **Playable Character's Health**
2. **Combo meter**
3. **Combo meter timer countdown**: amount of time before the combo meter is reset. Should probably be a bar that is emptying itself.

List of FX / Feedbacks

- **Incoming Attack:** should be a FX on the incoming attacker.

- **Strike impact**



- **Defeated enemy:** appears when the enemy dies (should amplified by a specific sound effect)