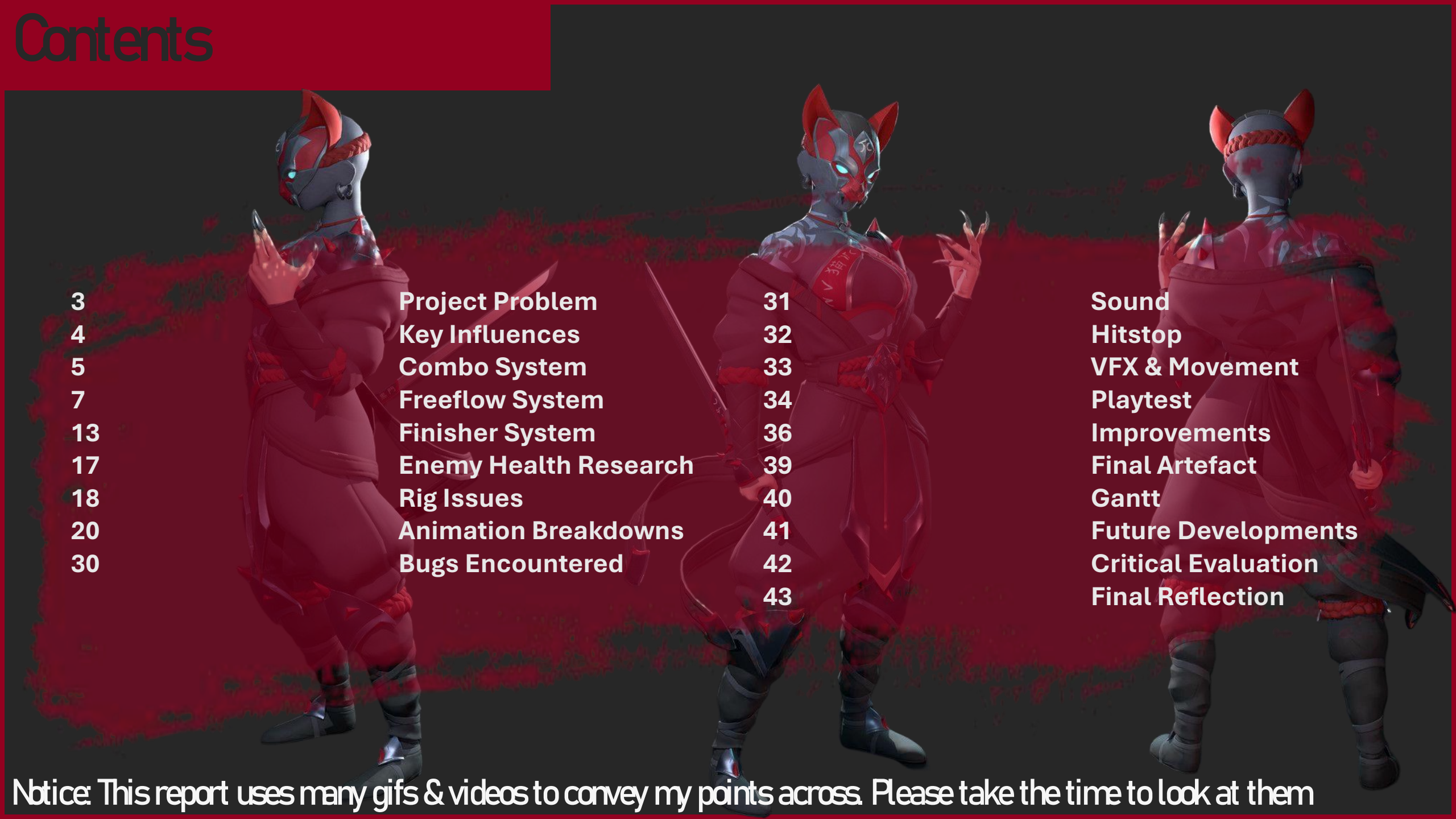
Responsive Third-Person Freeflow Combat System with Keyframe Combo and Finisher Animations

Presented by Jonny Renaut

Games Development Project

Contents



Notice: This report uses many gifs & videos to convey my points across. Please take the time to look at them

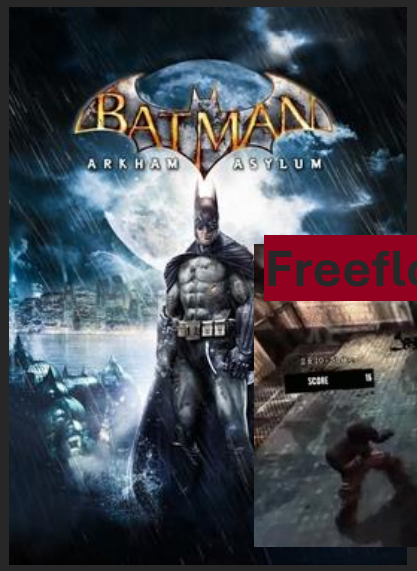
Project Problem

Creating a **responsive** UE5 **freeflow combat** system without breaking the flow of animation.

AIMS

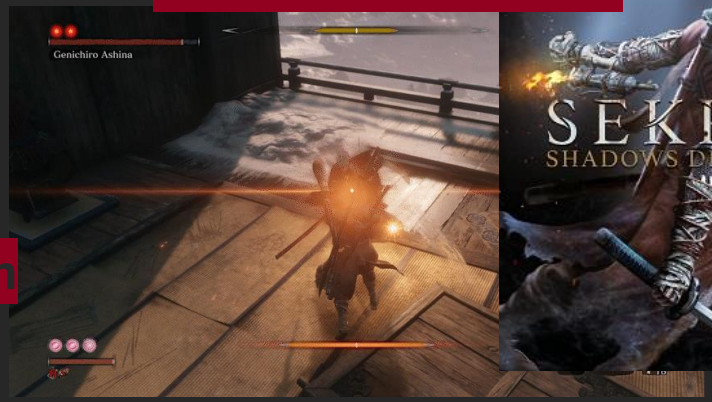
1. **Freeflow combat system:** dynamically switch between multiple enemies & chain attacks together
2. **Combos:** string together attacks that fits seamlessly into the overall combat flow
3. **Finisher/Takedowns:** execute a final finishing move when the enemy's health is low
4. **Smooth animation transitions:** all combos and finishers transition fluidly without breaking combat flow
5. **Responsive controls:** minimising input lag by making animations play immediately when the player presses a button

Key Influences



Freeflow combat system

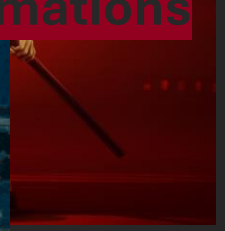
Fluid & fast combat



UE5 implementation:

Combat Fury link

Finisher animations



Combo animations

Combo Research

- Three individual attacks which are played after each other
- Small window to seamlessly transition into the next attack animation
- Not attacking will play a recovery animation returning to idle
- This video gives a good break down of all the parts of an attack by Sakurai from Smash Bros' dev team: link

Attack 1

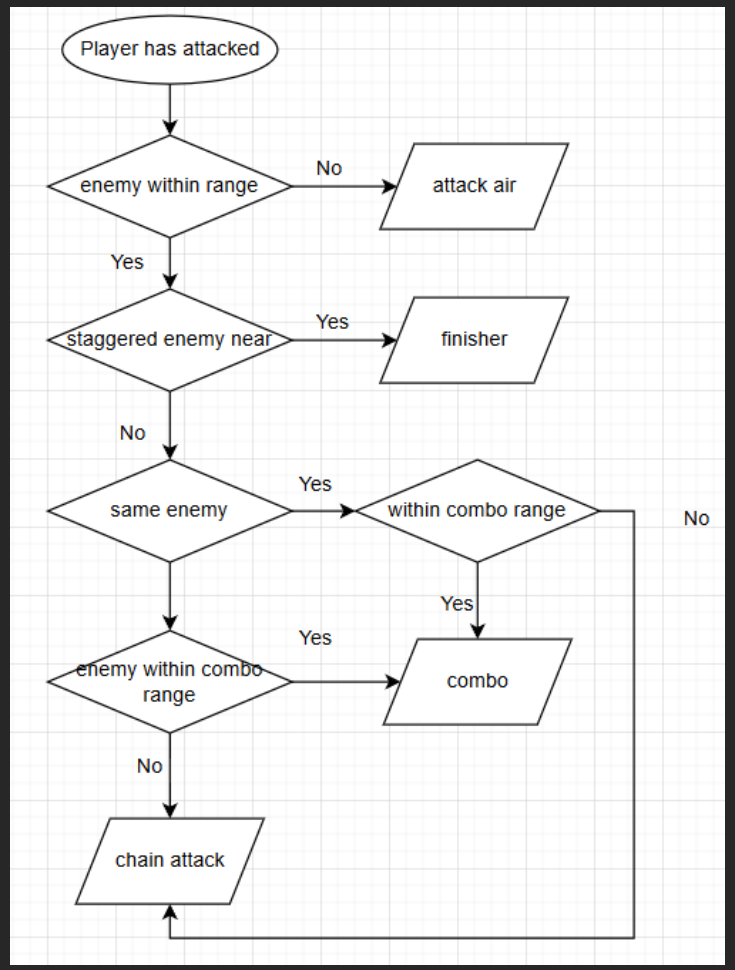
Attack 2

Attack 3

Combo

- Flowchart of adding +1 to float
- Anim notifies in the montages at the end of the animations so that player can't attack after returning to idle
- I used this tutorial link

This flowchart displays the logic that should be done every time the player attacks



Freeflow Research

- Referenced Batman Arkham games for freeflow mechanics
- Designed an enemy-targeting system based on player movement direction and camera orientation
- Used dot product function to determine the nearest enemy within attack range
- Highest dot product value (closest to 1) identifies the optimal attack target

- Theory video explaining different combat systems & why freeflow is one of the best implementations of it & what makes it unique: link

- Batman freeflow combat system recreated in Unity (different engine but breaks down the combat loop): link
- UE5 tutorial on freeflow aiming: link

Solved by ScottSpadea in post #2

you can use Dot product to compare the orientation of normalized vectors. a dot product between 2 vectors will give you a score between -1 and 1 representing the similarity between the vector's orientation.

so subtract your pawn's location from a potential target's location, to get a vector pointing from pawn to target. then normalize this vector. then dot product this with the forward vector of your pawn's rotation.

do this for all targets, and the target with the highest dot product will be the one closest to the front of your pawn, but you may want to factor distance into the calculation, so closer things have higher priority. you can do that by multiplying the distance of a target by a small fraction, then subtracting it from the dot product score.

$$a \cdot b = \sum_{i=1}^{n} a_i b_i$$

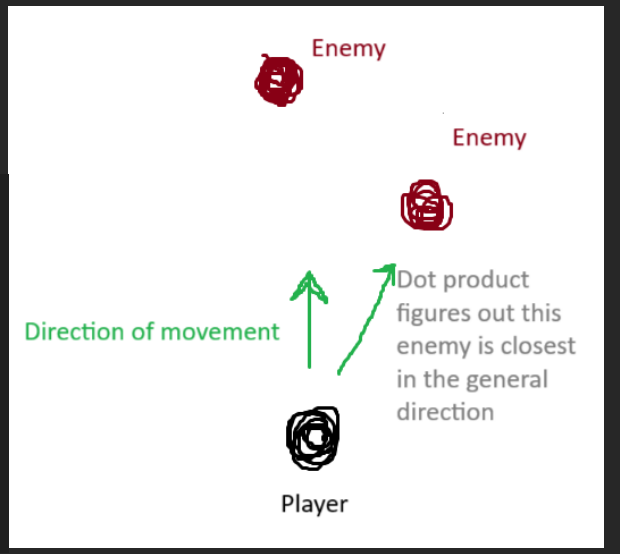
a = 1st vector

b = 2nd vector

n = dimension of the vector space

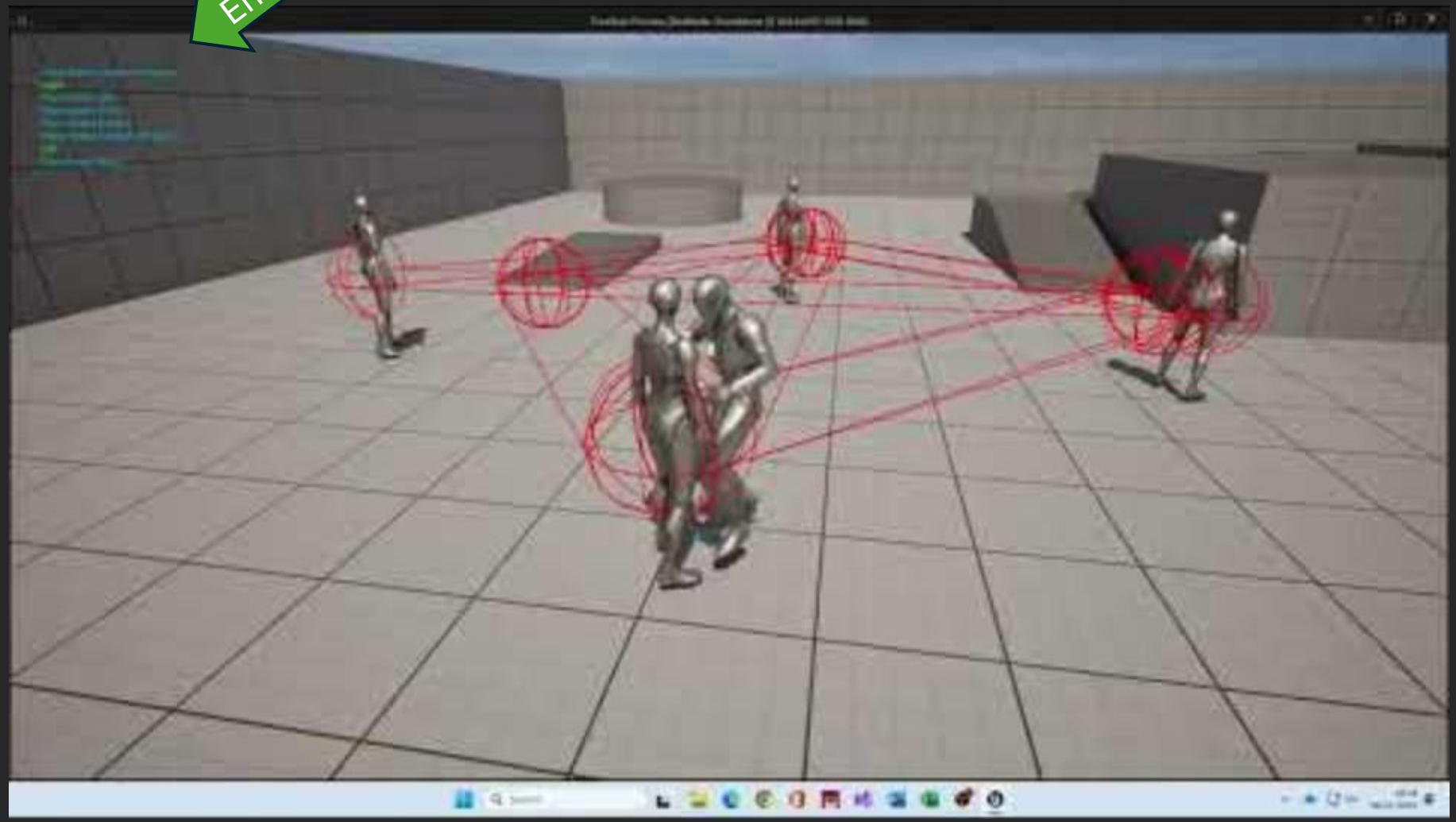
a_i = component of vector a

b_i = component of vector b



Freeflow Targeting

- Player teleports to random enemy in the vicinity



Freeflow Targeting 2

- Replaced the stock UE mannequins with characters, attack animations & enemy hit reactions that line up with which direction they have been hit from
- I used this talk by Mike Jungbluth about the anatomy of a hit reaction to help inform me on what the animations should look like: link
- I also made sure to include root motion on the hit reactions so that the animations would update their world location using this tutorial: link

Hit Back

Hit Front

Hit Right

Hit Left

Freeflow Targeting 3

- Uses motion warping for the starting section of the attack
- Smoother than teleporting

Freeflow Targeting 4

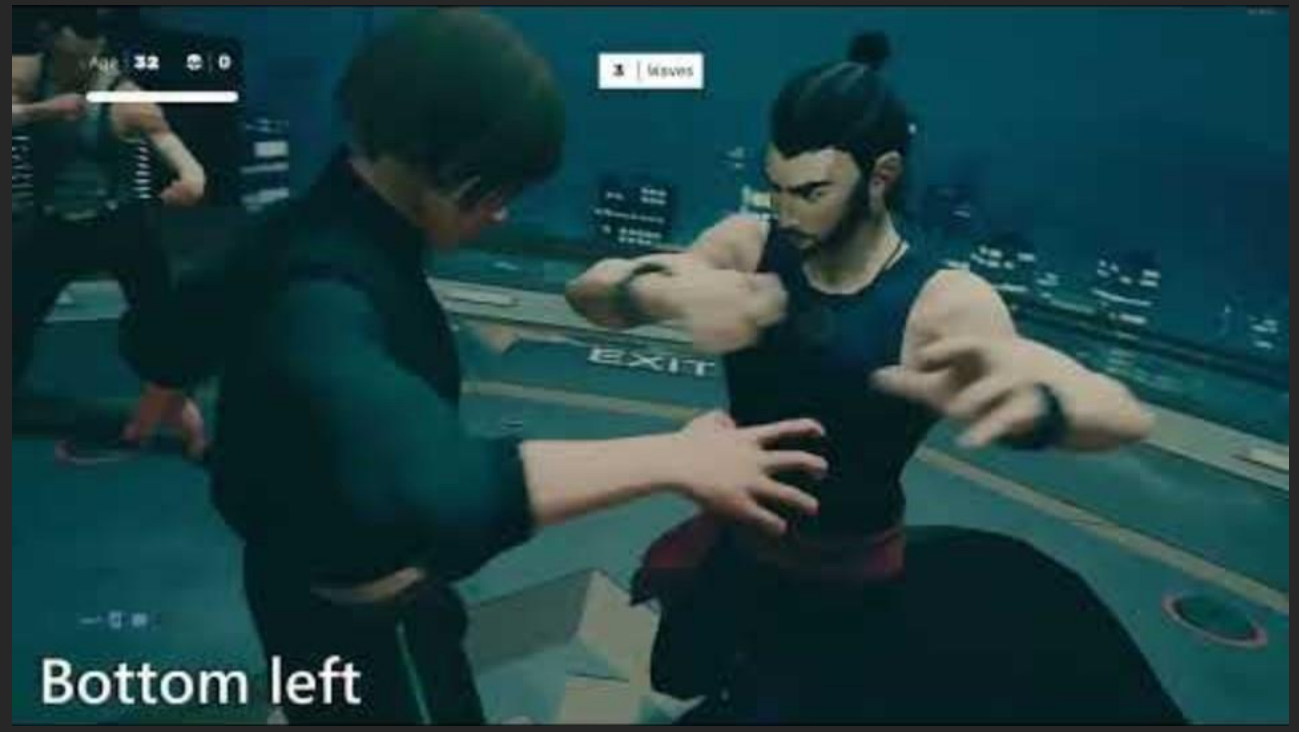
- I made it so that different attack animations will be played depending on the distance
- This means faster attacks will be played for shorter distances & slower kicks can be played for covering distances further away

Freeflow 5 Bugs

- The aiming of the freeflow targeting is random and only picks the closest enemies at random

Finisher Research

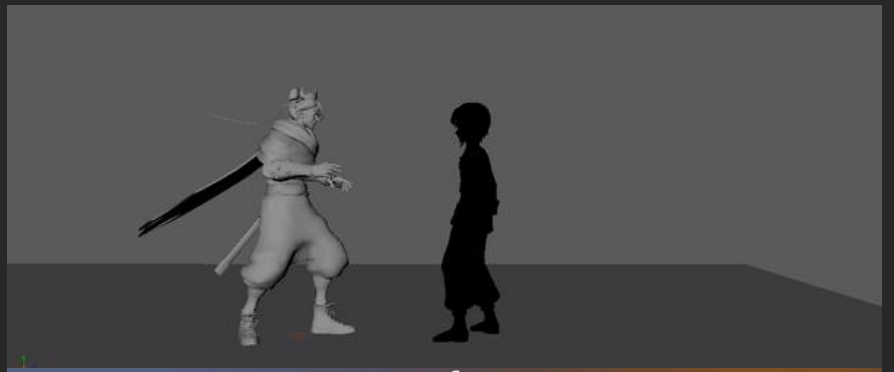
- I used inspiration for the finisher system from the Batman, Sifu & Sekiro games
- I used this tutorial to help me create the system where both the enemy & player play the animation at the same time & line up link
- I recorded & studied how Sifu handled its finishers from having the camera in different angles



Finisher 1

- I used this tutorial to help me create the system where both the enemy & player play the animation at the same time & line up [link]

Finisher anim



ref

Finisher 2 Camera

- Enemies warp in front of the player instead of the player previously warping in front of the enemies- This makes the player feel more in control
- I also experimented with animating the camera by blending the camera between level sequence & game camera using this tutorial: link

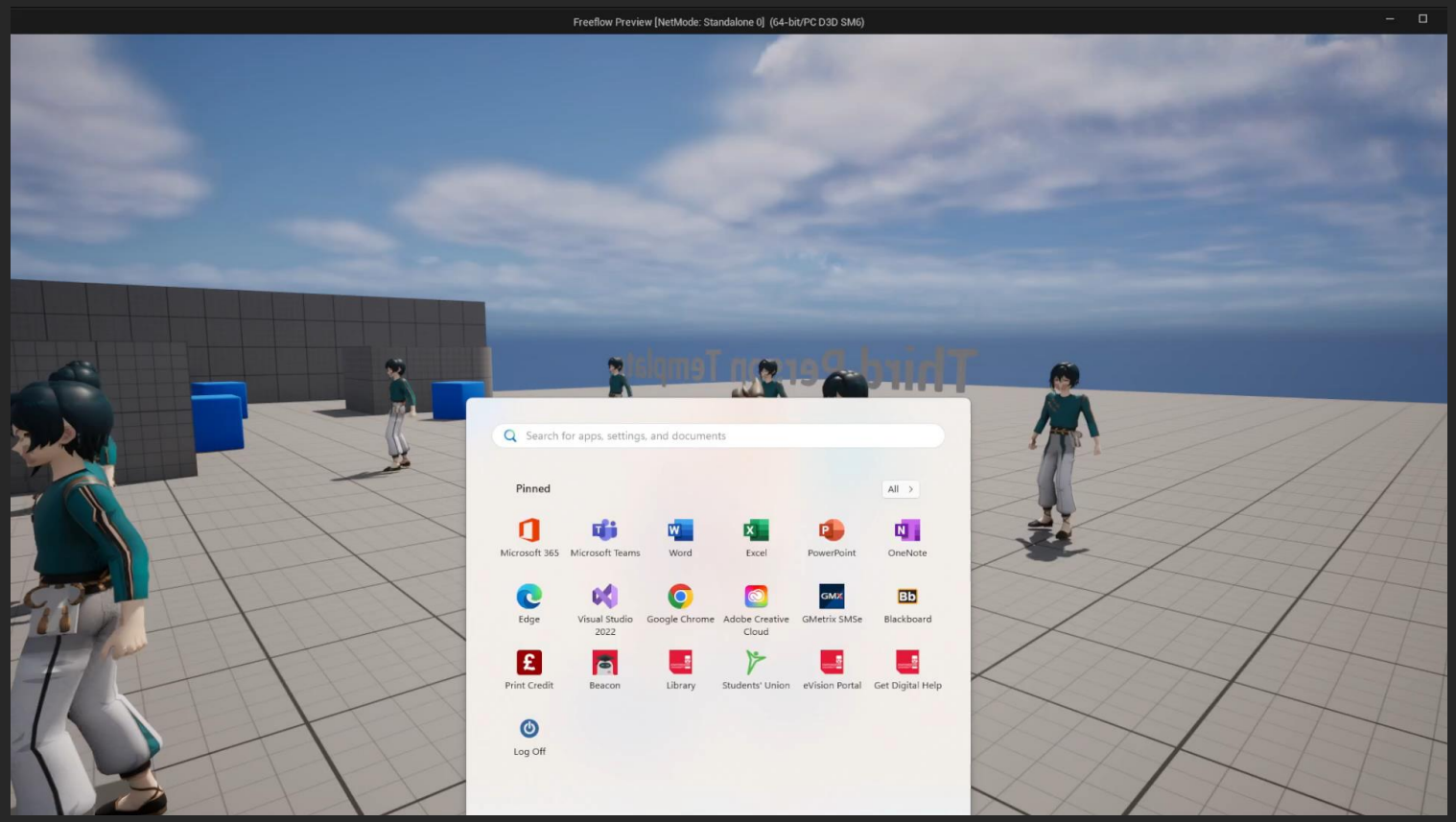
Sifu reference of enemy warping to player, not the other way round

Finisher 3 Bugs

- Currently the finisher cinematic camera is being added to the current camera position which causes issues like these
- Instead, the camera needs to blend to a separate one

Camera should be pulled down level when in birds eye view like in Sifu

Enemy Health Research

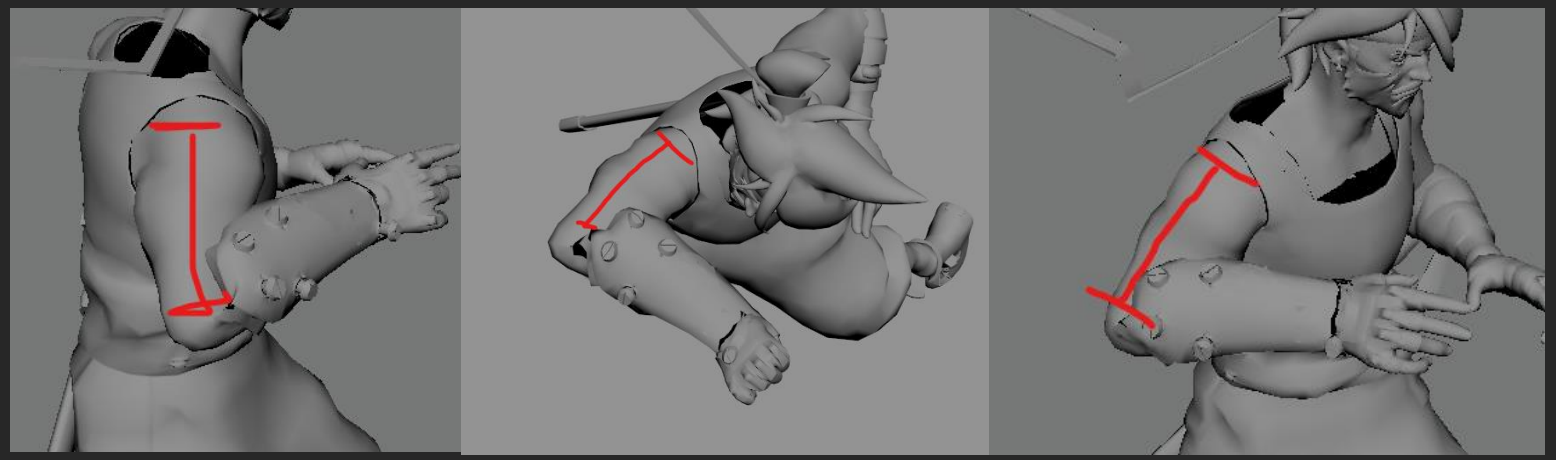


- There is no damage system currently implemented
- Healthbars need to be added to track enemy health and trigger finishing animations at low health
- Additional hit reaction animations need to be made to improve attack feedback reacting to the corresponding punches

Comparative analysis:

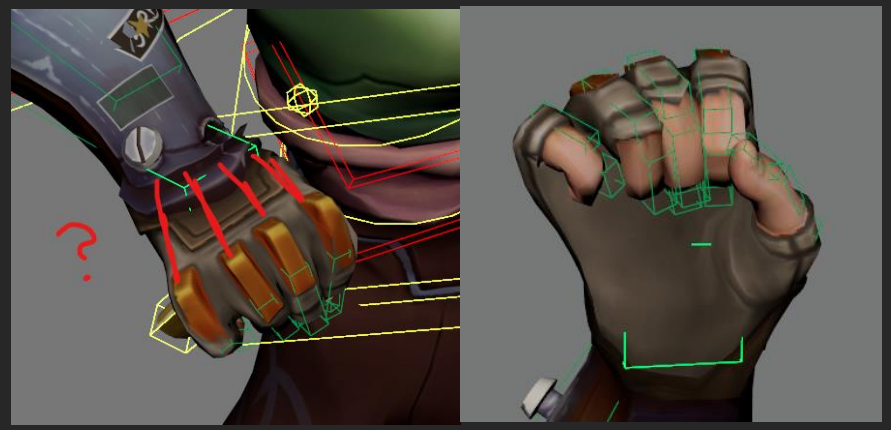
- Batman games use animation-based enemy health indicators (e.g., heavy breathing, VFX)
- Sifu uses visible healthbars for clarity
- Decision: Avoid complex animation variations due to production & scope constraints

Rig Issues



Player Model Issues

- Shoulder cape obstructs arm readability, affecting the silhouettes needed for combat
- Short forearms and lack of metacarpal controls limited fist & arm articulation

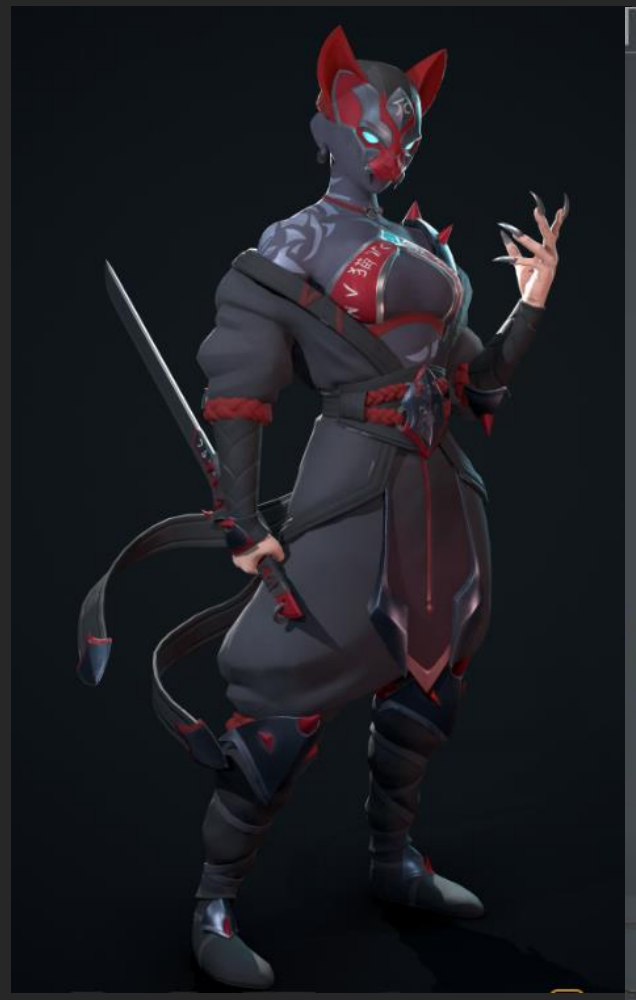


Enemy Model Issues

- Poor facial skinning caused vertex detachment
- Model did not match player character's art style
- Switched to a new model with a similar art style to the character to fix visual cohesion

Rig Update

- I decided to use the models as they have a similar art style



Kunoichi rig (PAID) Shinobi rig (FREE)

Combo Animations

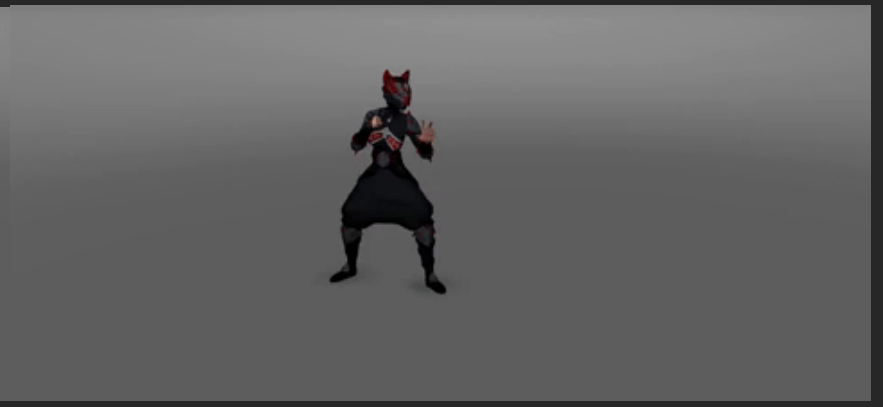
Iteration 1

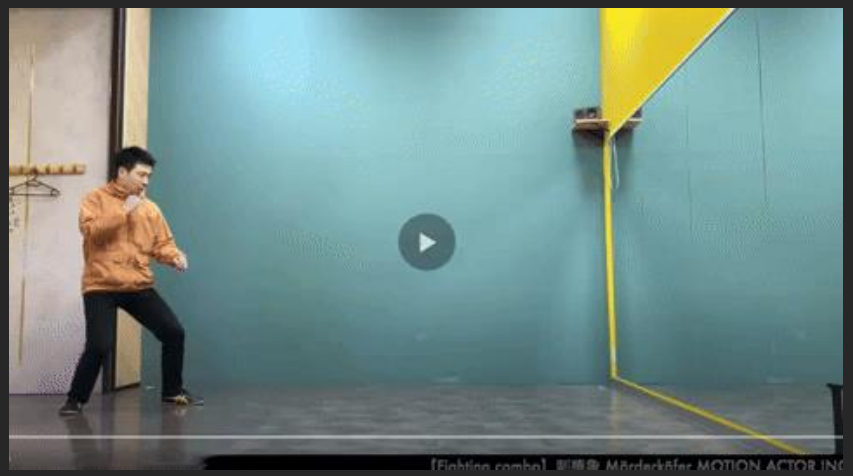
Ref

Iteration 2

Blockout

Spline





Ref

- I decided to scrap the first iteration attack combo because I didn't like the timing of it & believe it to be too slow

Combo Return to Idles

Combo 1

Combo 2

Combo 3

Combo 4

Combo 5

- I recorded my own reference for the return to idles because the reference video only has the return to idle for the last attack in the sequence
- This way I could physically act it out as well as watch myself back to see how it would look

Enemy Ht Reacts

Top left

Top right

+ combo

Mid left

Mid right

Chain Attack (Backhand)

Blockout

Spline

Polish + retime

Ref

Chain Attack (Butterfly)

Blockout

Spline

Polish + retime

Ref

Chain Attack (Snap Kick)

Blockout

Spline

Polish + retime

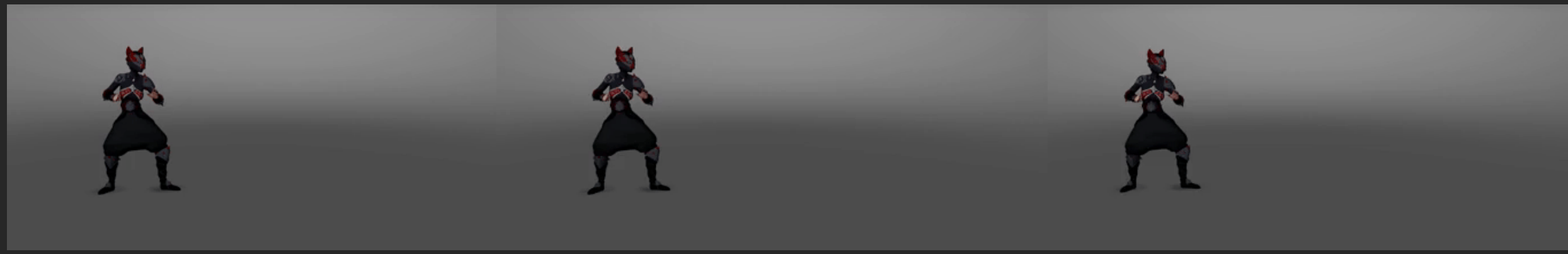
Ref

Chain Attack (Sobat Kick)

Blockout

Spline

Polish + retime



Ref

Finisher

Blockout

Spline

Polish + retime

Ref

Enemy Stagger

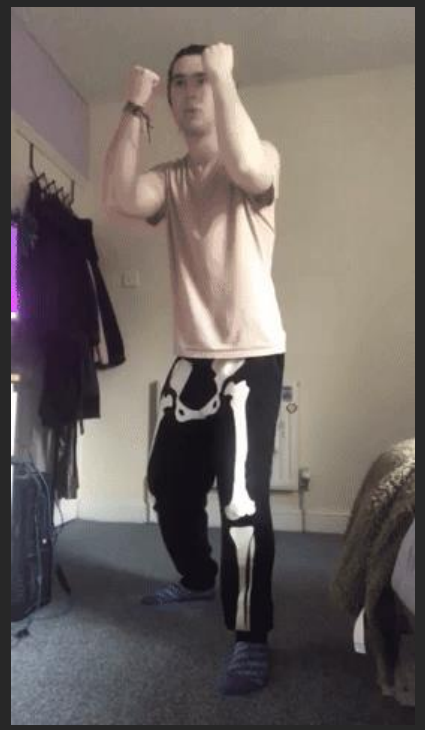
Blockout

Spline

Polish + retime

Loop

Ref



Engine Implementation

- I added all the animations to my project
- I used the SICKA Dynasty environment as it fit the art style of my characters: link

Bugs Encountered

Player Rig Scaling Bug

- **Issue**: Kiel Figgins' rigs used an incompatible small scale
- **Solution**: Manually scaled the character model & skeleton in 3DS Max and animations in Maya before importing into UE5
- **Impact**: Fixed animation blending errors where player shrank during transitions. Solved this by deleting unused states in the AnimBP state machine
- **Alternative Consideration**: Purchasing Kiel Figgins' **FBX exporter tool** (link) could have saved a week of development but would've increased my budget

Sound

- Collaborated with **Reuben Sambells** (**Sky Audio Engineer**) to create 20+ audio variations

- Programmed randomised selection logic for:
  - Rock footsteps
  - Weapon swooshes
  - Impact sounds

Game Feel Htstop

I made hitstop effect using this tutorial: link. This makes the combat feel impactful by emphasising the hits.
I did this in the blueprints rather than baking it into the animations. This is because the player should only have hitstop when they come in to contact with an enemy. If the player is punching the air then it will not play hitstop but if they hit an enemy, they will hitstop.

I also applied hitstop to the enemy on the hit reaction to exaggerate the impact

The hitstops are worked out depending on the damage of the attacks.

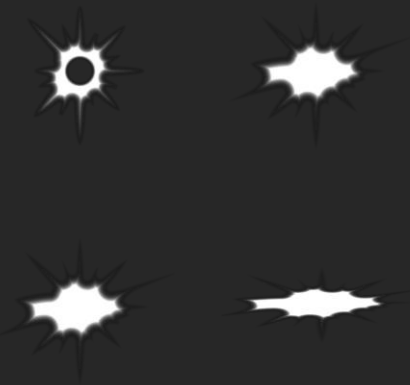
I used Smash Bros' Sakurai's blog on hitstop to help inform me. Blog: link From this I found that the stronger the attack, the longer the duration of the hitstop should be. I then re-arranged my damage system to be able to manually control which attack animation deals how much damage depending on animation notifies

Gameplay of VFX

VFX & Movement

VFX Development

- Created **custom hit effects** using **Substance Designer** and Niagara Systems
- Flipbook scrolling ensures responsive impact visualization
- Needs improvement: Sometimes clips into enemy models; potential fix requires adjusting Niagara render order



- I created a walk run blendspace & made a walk cycle
- I added a subtle additive lean to the run animation to give it more weight using this tutorial: link

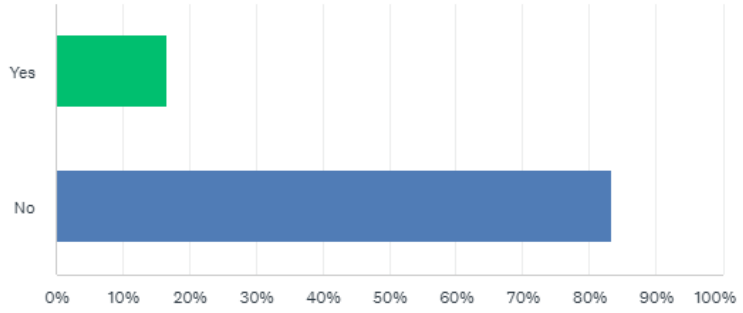
Playtest Q1 + Q2

I asked some quantitative & qualitative questions for my playtest survey. I used my connections to industry by getting former colleagues & friends to playtest my game. I found that having them stream themselves playing the game was better than any survey because I could see the nuances of how they interacted with it.

Question 1

Is there anything you'd improve about the animations? Select all that apply

Answered: 12 Skipped: 0



If yes, what would you improve?

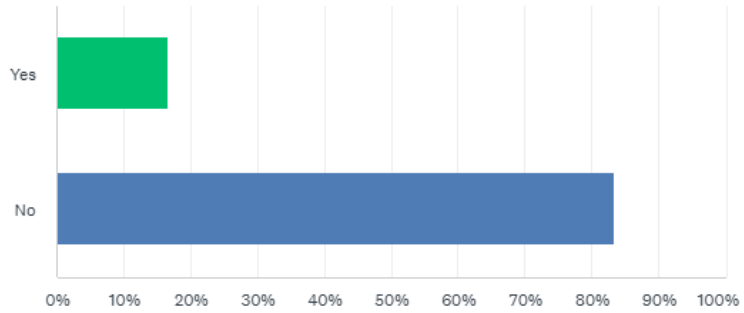
Answered: 2 Skipped: 10

☐	the sobat kick attack could be faster	2/19/2025 01:01 PM	View respondent's answers Add tags
☐	The walk animation could use some polish- looks very awkward	2/19/2025 07:20 PM	View respondent's answers Add tags

Question 2

Did you experience any issues with the animation blending?

Answered: 12 Skipped: 0



If yes, what was the issue?

Answered: 2 Skipped: 10

☐	you can do two finishers on the same enemy if you spam the attack button during the finisher animation	2/19/2025 01:01 PM	View respondent's answers Add tags
☐	The run to walk animation could use some smoother blending	2/19/2025 07:20 PM	View respondent's answers Add tags

Playtest Q3

What can I do to improve or add to the project?

Answered: 12 Skipped: 0

1. *“Improve combat immersion by having enemies attack the player. There's a tonne that could be added to this game but for what it is, it is great.”*
2. *“Add enemy AI”*
3. *“Make a cool post process flash effect when the enemy is killed”*
4. *“Add audio cue to the final hit of the combo”*
5. *“Add more audio like a bone crunch sound effect when the knee is broken on the execution animation”*
6. *“Would be cool if you added an axe kick finisher animation”*
7. *“enemy's fighting back. more finishers”*
8. *“Make it so that the enemies can move- will make gameplay more dynamic”*
9. *“More finisher/execution animations!”*
10. *“Maybe add aim offset to the player's idle”*
11. *“Add some more fun VFX to the character”*
12. *“Have some action music in the background”*

Playtest Feedback

I enhanced the player's VFX to improve game feel. Specifically, I added motion trails on the hands and feet during attacks to emphasise speed and create a motion blur effect. Also, I implemented dust VFX on the feet during movement to enhance realism and show that the player is interacting with the environment.

To improve animation fluidity, I increased the walk animation speed and adjusted the character’s arm positioning to rest naturally at the sides.

For impactful enemy deaths, I integrated a post-processing effect inspired by the "wasted" screen from Grand Theft Auto: link

To elevate game feel, I added a bone crunch sound effect for knee strikes and a higher-pitched impact sound for the final hit in the combo, creating a more satisfying audio response. I also found background music that aligns with the game's atmosphere from a royalty-free website: link

I expanded the combat finishers by adding an axe kick animation- finishers are picked at random which will increase unpredictability and dynamism in combat.

Lastly, I refined the idle aim offset system, leveraging a similar approach to the additive lean mechanic. I rotated the first frame of the idle animation left and right, then used a blend space to interpolate between them, dynamically adjusting the character’s head rotation based on player input. I also did this for the walk animation.

Playtest Feedback

Old walk

Blockout

Spline

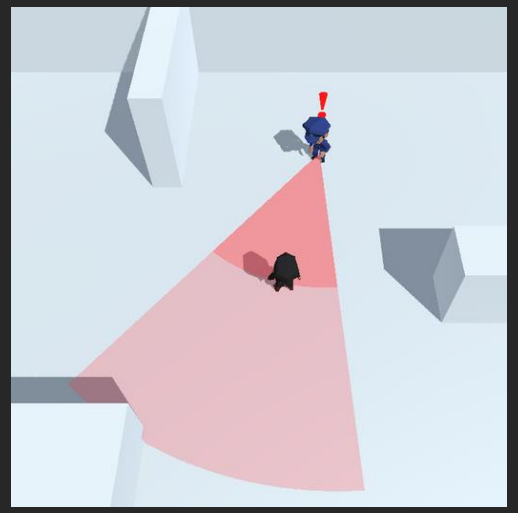
New walk

Enemy Movement

I also created enemy movement with the help of this tutorial: Youtube link

1st iteration had the enemies instantly detecting the player, constantly following the player & never losing agro. This caused issues as the enemies kept getting bunched up in a crowd together.

I resolved this in the **2nd iteration**. Enemies will only agro & chase the player if they are within range & the player is within the enemy's line of sight E.G.



I also made sure that the enemies circled round the player so that they were not all crowded together. I also added wait timers to the enemy logic so that they were not twinning & all moving at the same times. They also kept a certain distance away from the player. This is so that the player has breathing room to carry out attacks & switch between enemies.

Result of Playtest

Gantt

- I managed to follow my gantt chart closely & complete all of my deliverables

Responsive Third-Person Combat System

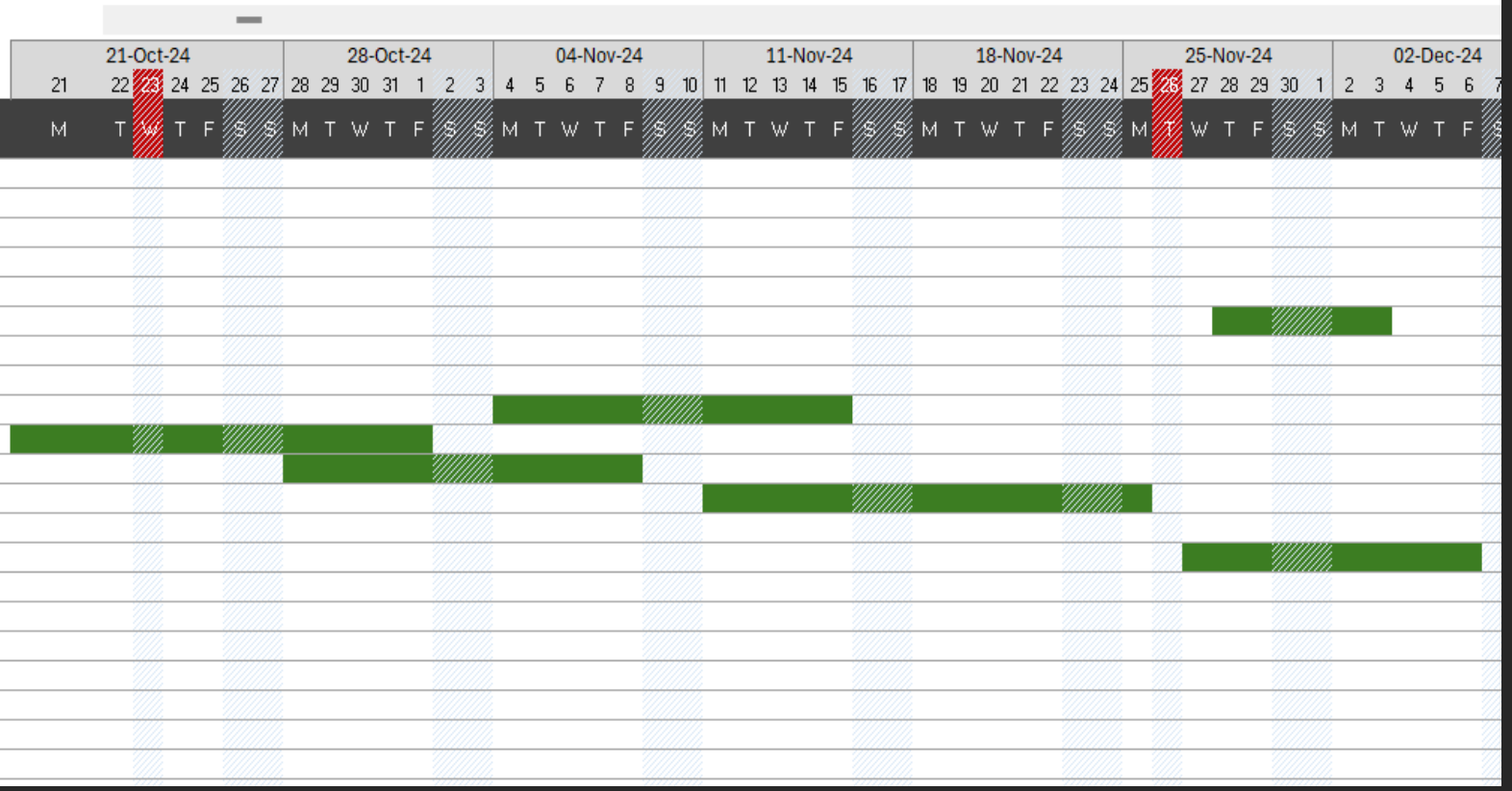
Course: BSc (Hons) Computer Games Design with Animation

Jonny Renaut

Project Start: Mon, 30-Sep-2024

Display Week: 4

TASK	ASSIGNED TO	PROGRESS	START	END	WORK DAYS
Planning				-	-
Project Proposal		100%	Mon 30/9/24	Fri 11/10/24	10
Ethics Form		100%	Mon 14/10/24	Fri 18/10/24	5
Find rigs		100%	Mon 7/10/24	Fri 11/10/24	5
Project Pitch		100%	Mon 30/9/24	Tue 8/10/24	7
Mid-Point Review Presentation		100%	Thu 28/11/24	Tue 3/12/24	4
Artefact Development Report		0%	Mon 17/2/25	Mon 10/3/25	15
Programming				-	-
Locomotion AnimBP		100%	Mon 4/11/24	Fri 15/11/24	15
Freeflow Combat System		100%	Mon 14/10/24	Fri 1/11/24	15
Combo System		100%	Mon 28/10/24	Fri 8/11/24	10
Finisher Attack Functionality		100%	Mon 11/11/24	Mon 25/11/24	11
Animating				-	-
Locomotion (Idle, run)		100%	Wed 27/11/24	Fri 6/12/24	8
Combo Attacks		100%	Mon 9/12/24	Wed 10/1/24	20
Finisher Attack		100%	Mon 6/1/25	Fri 7/2/25	25
Enemy Flinch		100%	Mon 16/12/24	Fri 20/12/24	5
Polishing				-	-
Sound Effects		100%	Mon 10/2/25	Fri 14/2/25	5
Cinematic Finisher Camera		100%	Mon 3/2/25	Fri 7/2/25	5
VFX		100%	Mon 17/2/25	Fri 21/2/25	5



Future Developments

Future Developments

(Beyond current project scope but planned for continued development)

- **Enemy attack AI** to take turns engaging the player
- **Heavy attacks** to diversify combat flow- potentially using weapons e.g. sword
- Additional **finisher animations** (e.g., neck snap)
- **Wave-based** enemy arenas
- **Combo counter** multiplier for dynamic scoring & incentive for players
- Player **parry & dodge** mechanics
- Randomised **enemy health** & **armour** for visual variation
- Finishers could **interact with the environment** e.g. slamming into a wall. Sifu reference: link

Critical Evaluation

- Successfully followed the **Gantt chart** timeline for structured progress
- **Prioritized engine** systems before finalizing characters, this allowed for a very **flexible development**
- Efficiently used **blockout animations** instead of polishing animations that later might not be used

Challenges Faced & Solutions

Major Issue: Player rig scaling problem

- **Lost a week of progress troubleshooting**
- **Solution**: Learned **scaling workflows** in **3DS Max** and **Maya**
- **Reflection**: While buying Kiel Figgins' FBX exporter tool (link) could have been a quick fix, manual troubleshooting allowed me to learn valuable **technical experience**

Strengths of Project

- **Game feel is highly satisfying**:
  - Hitstop, VFX, and sound effects enhance impact of combat
- **Smooth animation blending** achieved through **motion warping**

Areas for Improvement:

- **VFX** needs refinement; **hit shapes** could be improved & proper rendering to prevent **clipping into enemy models**
- **Control rigs** hand IKs could be used to position fists more accurately on enemies' faces. Potential tutorial: link
- Stopping music & sound during **hitstop** could further emphasise combat impact

Final Reflection

This project successfully meets the goal of creating a **responsive freeflow combat system** with smooth animations and precise controls. While technical challenges like rig scaling and enemy AI required troubleshooting, the iterative development approach ensured a polished gameplay experience. Further enhancements, such as expanded AI, attack variety, and advanced hit effects, will refine the project towards an industry-standard combat system.