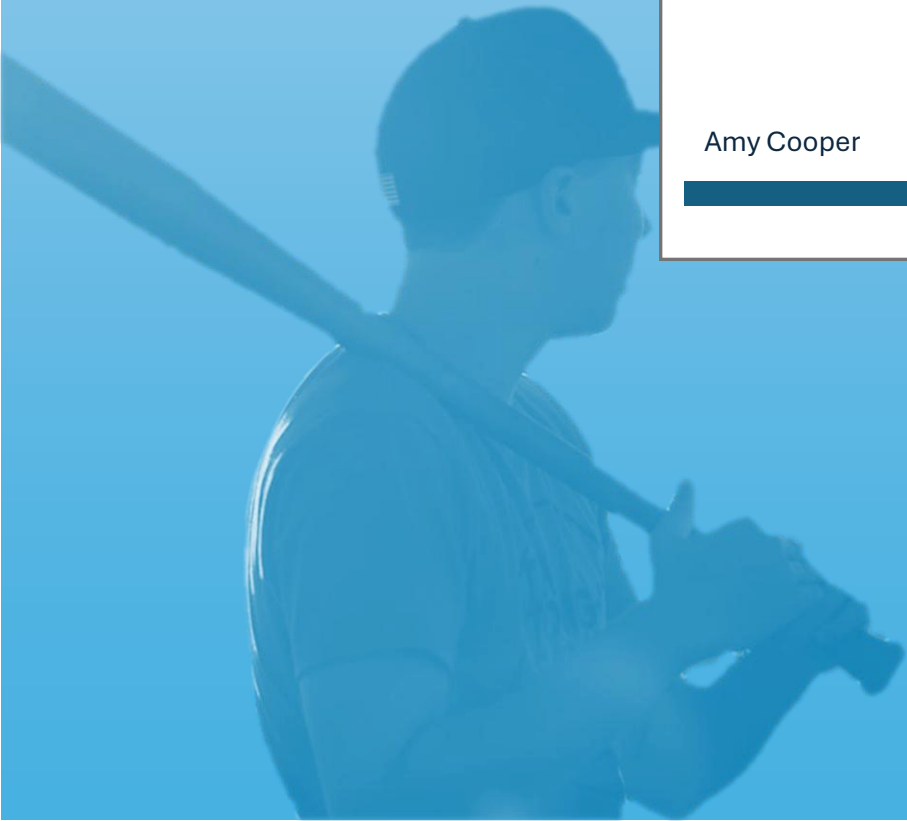




Baseball VR

Artefact Development Report

Amy Cooper



Contents

Introduction	2
Aims, Objectives, and Deliverables	2
Research Methodologies	3
Documentation of Production	3
Early Prototyping.....	3
Measuring a Swing.....	4
Scoring.....	5
Pacing	6
Game Feel	6
Evaluative Review.....	8
Bibliography	9

Introduction

Many fitness games both with and without VR fall into the same pitfall where the game can be played without making the motions intended to exercise the player. For example, Wii Sports can only detect how the player shakes the remote which can be done in very small flicks while sitting down.

VR provides full 3D tracking of the player's head and hands allowing for a full measurement of the player's movement. VR fitness games largely base around rhythm games with Beat Saber being the most popular, while many players use full swings while playing and score increases with wider angles of swings, when playing on higher difficulties the blocks come too fast for wide motions so the player eventually learns to play with small wrist flicks to keep up with the difficulty coming to the same problem of the exercise being diminished.

In this project I will solve this problem by creating a VR adaptation of baseball where the power of a swing is determined by the distance travelled throughout the swing. The end use of this application will be an easy to learn and play game that engages the player in meaningful exercise.

Aims, Objectives, and Deliverables

The aim of this project is to create a gamified virtual reality exercise program using baseball that will reward the player for engaging in meaningful exercise. Batting the ball should feel satisfying to the player especially when hitting a good shot.

I will achieve this by making a scoring system that rewards powerful swings with higher scores, and adjusting game feel through sound, visuals, and some experimental methods to tell the player when they have made a good shot and make those shots more satisfying to pull off.

This project will deliver 2 separate builds of the game. One made for windows computers that can be played using any Steam VR or OpenXR compatible VR headset. The other will be a build for android that can be played at a stable 90fps on the Meta Quest line of headsets with support for the Quest 2 and higher.

I will also deliver promotional materials including cover art, promotional banners, and an edited trailer showcasing the game. I will also produce a technical video outlining many of the design and technical decisions throughout the project along with the challenges that were faced.

Research Methodologies

When adapting baseball into a VR experience, I researched a variety of existing video game adaptations both for VR and not. These existing designs influenced the project in the hopes of creating a streamlined experience that users will be familiar with.

My primary research came in the form of user testing. Throughout development I have invited people to play the game and fill a form to comment on the various mechanics to gauge the exercise potential and the game feel using quantitative data and allowing participant to add extra details for notes on how to improve the project. Participants were at varied VR, sports, and fitness levels allowing me to have a view on how different demographics viewed and interacted with the game.

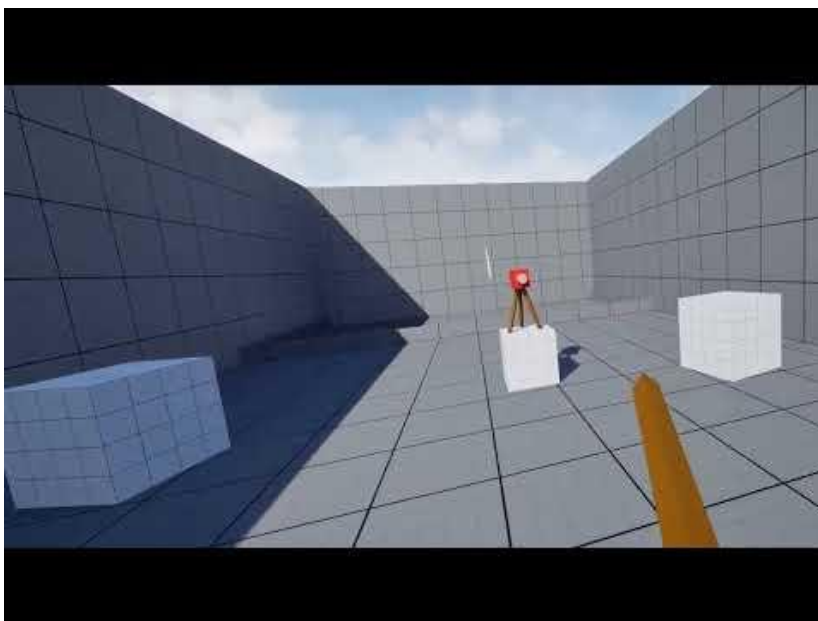
The project was developed using an agile methodology. New mechanics would be created and existing ones iterated in sprints. this allowed me to frequently test the game, make improvements, and quickly implement new systems.

Documentation of Production

Early Prototyping

The first thing I created for the project was a bat that could hit and launch a ball. To make the interaction stable and controllable I use a basic collision system that shoots a separate actor for the ball projectile. This would allow me to adjust the angle and speed of the hit with my own values opposed to using a full physics collision which would yield inconsistent results that can't be adjusted to the extent I needed.

I knew the feel of the batting would be extremely important to selling the impact of the hit, because of this I made sure to add effects early with a simple hit particle and a ball trail.



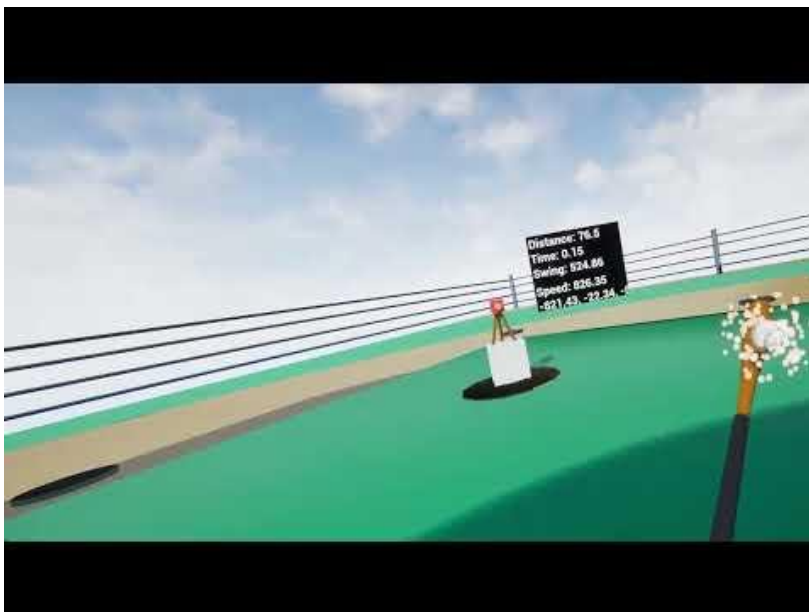
Measuring a Swing

Being the game's purpose to measure the player's swing, this was the mechanic that needed to work the best and would see a lot of iteration throughout the project. I had first planned on using the distance of the swing as that would show if there was a large arc better than measuring the speed, to do this I would use a spline to account for the curvature of the swing.

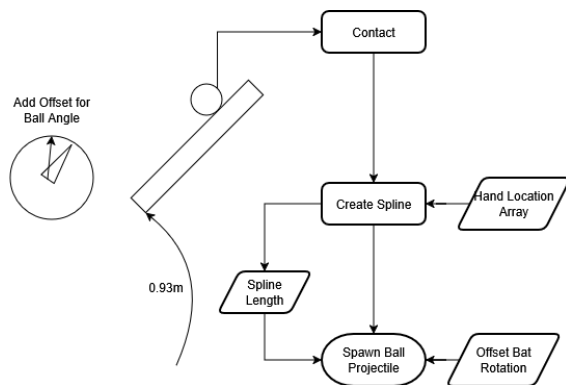
In the first iteration I would predict when the player started a swing and end the measurement when the ball was hit. My first attempt at swing detection was to test when the bat was moving towards X positive as then it is moving from behind the play towards the ball, when testing this in VR it was very messy and inconsistent. My first try to fix the issue was to start measuring the swing when the bat moves forward and reaches a speed threshold, from test swings I decided a good starting threshold should be 9.5 After testing the new system out in headset the swinging didn't feel too sensitive anymore but felt very inconsistent, it would sometimes recognise a full swing but most the time would record it late or early and show a much shorter swing.

The second major iteration I would instead record the swing based on a set time instead of predicting the start and stop of a swing. The swing detection was tweaked to only look for an overall speed threshold and when reached it would record the next 0.15 seconds of the player's swing. This was much more consistent however required the action frames to be enabled so the full 0.15 can be recorded if the swing was late.

In the final iteration I needed to remove the dependency on action frames and to do so I would make the spline once the ball was hit by replacing the swing detection with continuously tracking the bat's position in the past 0.15 seconds and creating the spline with those values once the ball was hit, this eliminated the remaining cases of the swing being miscalculated and allowed action frames to be disabled.



Next, I had to convert the calculated swing strength into ball velocity. At first, I had tested a few swings to find a power where I felt the distance should reach across the field, I then multiplied the power to reach the final value. The first problem with this solution was if the ball hit the bat with no speed it would drop on the floor causing issues of when to despawn the ball, this was fixed by adding a clamp so no swing would move the ball forward enough to be caught. The second problem was medium swings felt too varied where 2 shots that felt similar had very different results, to fix this I set the ball's velocity based on a curve instead of a simple multiplier, this allowed me to fine tune the balls trajectory.



Scoring

In my research to adapt the rules of baseball for VR, much of the core mechanics were influenced by the VR title Sports Scramble (Armature, 2019), a game involving tennis, bowling, and baseball where the equipment of the game such as the bat and ball changes throughout the game, in their implementation of scoring they had separate character actors that would run around the pitch in replacement of the batter themselves who would instead stay at the batting spot throughout the inning. This meant the player was never moved around the board and the camera perspective never changed for a more comfortable experience.

For calculating the score given, I looked towards Wii Sports (Nintendo, 2006) specifically from the documentation of how the game works in Wii Sports Wiki (Fandom, no date) where I learnt opposed to traditional baseball where the batter runs until the ball is caught, a timer is used to determine if the player score a single, double, or triple run. The exact timings of these scores were changed to suit my game based on internal testing. Jesse Schell in The Art of Game Design writes about the simplification of baseball in Wii Sports and how it was decided “since swinging the controller was the most unique part of this game, they would focus all their attention on getting that part of the baseball experience right”.

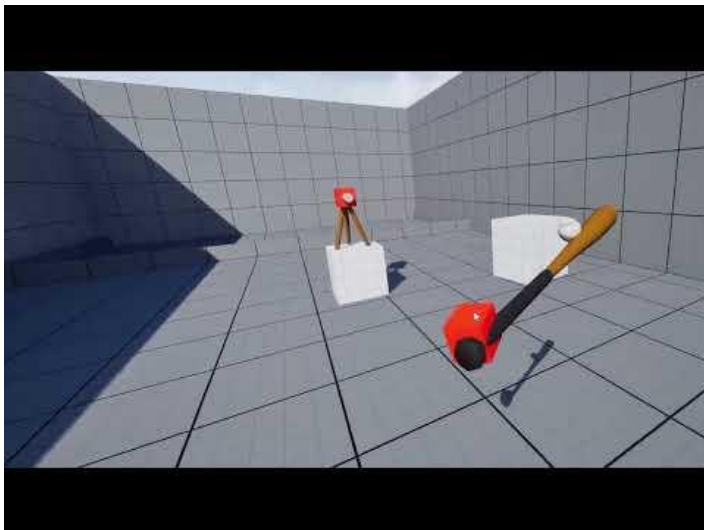
The implementation of scoring in my project started with fielder AI characters that can detect the ball within a set range and will chase the ball until it is caught by itself or another AI where it would then return to the location it was first placed. The AI was iterated throughout development first adding an extra collision to allow them to catch the ball before it hits the ground resulting in an out and creating a prediction system so an AI can decide if the ball will be at head height within a certain range it can move towards the predicted

of simple burst effects in the direction of where the ball is travelling. The travel particle is a ribbon effect fading from an orange hue to white to add contrast between the ball and the environment, this effect was highly praised in user testing. Other solutions such as in both Wii Sports (Nintendo, 2006) and Sports Scramble (Armature, 2019) have a much longer and thinner ribbon effect to show the ball's overall travel however for my solution I decided to prioritise highlighting the ball's location with the effect.

An early problem I ran into when internally testing the bat was the tracking would feel shaky while holding a long object. To solve this, I interpolate the location and rotation of the bat to the player's hand at a fast speed, this gave very little delay to tracking and filtered out any small movements and inaccuracies.

I also attempted more experimental methods of improving the game feel, this was built into the Action Frames 'A-Frames' system. In Stop for Big Moments! [Design Specifics]^[3] Sakurai mentions "In your game's big moments, stop! Let your players feel those moments of success-or defeat." It is a technique used in a variety of games, but I have found to be relatively untested in VR applications. The only example of A-frames in a VR game was in Until you Fall (Schell Games, 2020) where after a boss fight is defeated a few seconds of slow motion are used affecting the world and the player's hands to exaggerate the last swing.

To implement A-Frames within the project I utilised the bat interpolation by pausing the movement for a set time when the ball is hit before resuming after, this also involved adding a delay in the bat stopping the pitched ball and spawning the ball projectile. The setting was left completely optional once batting power calculations were iterated to support both scenarios. User testing was rather mixed on how they felt about A-Frames, some said it felt strange but really added to the impact of the hit while others said it felt like the game was frozen for a second and thought it more as an error than an intended feature.



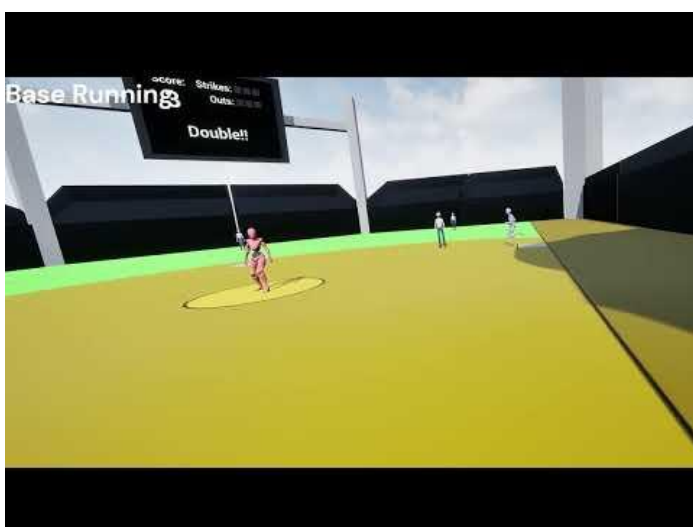
Evaluative Review

In this project I promised to deliver stable builds for a VR baseball game for both windows PCs and Quest standalone headsets on Android. The game has been successfully ported to both working stably at a consistent high frame rate. There may be room with the quest port to improve graphical fidelity before reaching any performance problems such as enabling anti-aliasing and improving lighting quality.

The quality of the game and its use as exercised can be measured by the feedback given in testing, the overall enjoyability of the game was measured in the first round of play tests to have an average rating of 4.25/5, in the second round of testing conducted near the end of the project the average rating rose to 4.4/5. These results show improvement of the game overtime and success on the part of making the game enjoyable.

User testing also showed the game was quite physically demanding with 55.6% of testers agreeing the game was physically demanding. The data also showed more people found the game demanding on later on in development showing an increased success rate in this goal.

There are some clear points of improvement that can be made to the project. Throughout testing many players felt the system that aims what direction the ball will go felt inaccurate and when stretched with unexpected data would break immersion such as holding the bat backwards. A potential solution for this problem would be to implement a new system using different values to calculate the angle, this could include the collision normal for a more realistic method, during the second round of playtesting a participant mentioned hitting the ball early or late could be another solution. Another problem faced throughout testing I believe could have impacted people's enjoyment of the game was space anxiety, wearing a VR headset acted as a blindfold which made some players hesitant to make the intended large swinging motions. I believe a way to resolve this issue would be to create a tutorial session teaching the player how to swing with clear visualisation of the player's play space.



Bibliography

Armature Studio (2019) *Sports Scramble*. <https://www.armature.com/games/sports-scramble/>.

Fandom (no date).

[https://wiisports.fandom.com/wiki/Baseball_\(sport\)#Types_of_Plays_\(List\)](https://wiisports.fandom.com/wiki/Baseball_(sport)#Types_of_Plays_(List)). (Accessed: February 10, 2026).

Hillmann, C. (2021) *UX for XR : user experience design and strategies for immersive technologies*. 1st ed. 2021. Place of publication not identified: Apress. Available at: <https://doi.org/10.1007/978-1-4842-7020-2>.

Masahiro Sakurai on Creating Games (2022) Stop for big moments! [Design Specifics]. <https://www.youtube.com/watch?v=OdVkeOzdCPw> (Accessed: February 10, 2026).

Nintendo (2006) *Wii Sports*. <https://www.nintendo.com/en-gb/Games/Wii/Wii-Sports-283971.html>. (Accessed: February 10, 2026).

Schell Games (2020) *Until You Fall*. <https://schellgames.com/portfolio/until-you-fall>. (Accessed: February 10, 2026).

Schell, J. (2020) *The art of game design : a book of lenses*. 3rd edition. Boca Raton, FL: CRC Press, Taylor & Francis Group.