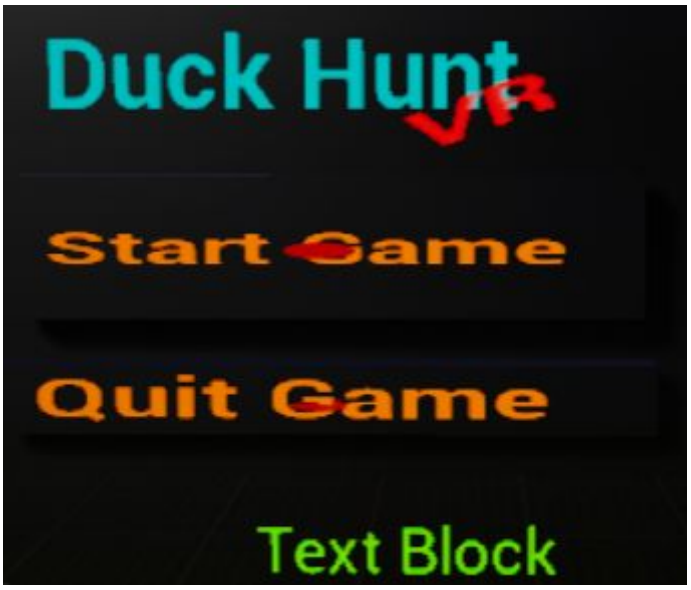
Duck Hunt VR

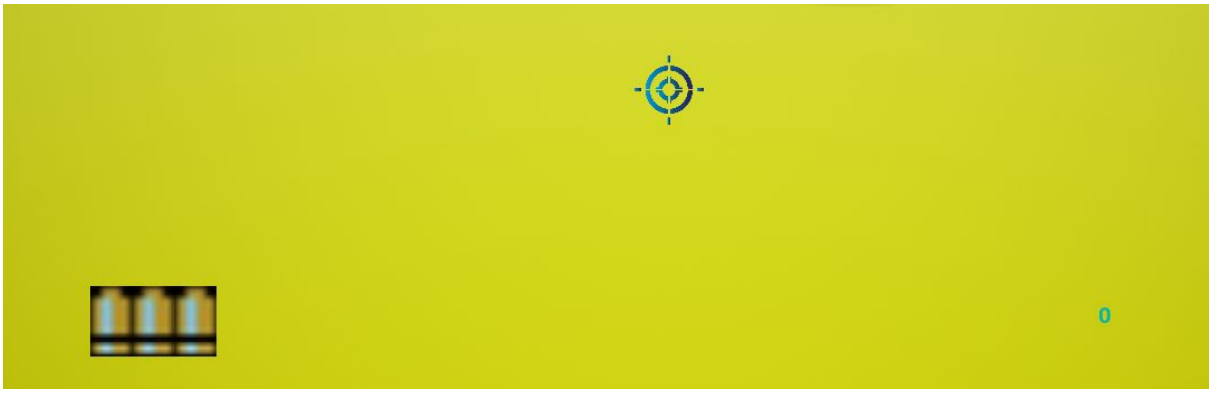
By George-Thomas Beazley

HUD Mock Ups

Main Menu HUD



Player Game Screen HUD



Player Controls

Windows Platform

Mouse X and Y control the Rotation movement of the camera of the player and the ray of where to shoot.
Mouse Left button mouse click controls the shooting.

Oculus

Head Rotation Controls the rotation of the camera.and controls the ray of where to shoot.
Controller trigger Controls the shooting.

Game Overview

Duck hunt vr is a Duck Hunt ‘84 inspired game. So that it reaches out a larger crowd of audiences, there is a version of this game that can be played on windows. The objective of the game is to takedown as many ducks as you can without missing. If you miss three times, you lose and the score that you end up with is compared with the high score. If you do not beat the high score your game is basically worthless. Best get to hunting!!

Software Environment

Programming Language	C++/Blueprints
IDE	Visual Studio
Engine	Unreal Engine 4
Source Control	GitHub

Actors (GameObjects)

Duck

The ducks will contribute to the score when shot. They travel to random locations every time the game is played. When a duck is blasted it respawns at its original position.

Game Player Pawn

The game player pawn can blast ducks to increase their score. The only way out of the game is if the game player pawn runs out of bullets. This actor only has three bullets. The game player pawn can also save their highest ducks blasted count.

Objective and Failure

Aim of the game

Beat the highest score by blasting ducks. The score is shown at the start menu on the bottom right-hand corner of the title board. At the end of the game the current score is compared to the high score. If it is higher than the highscore, the current score will replace the high score.

Failure

Missing three times will result in the player returning to the main menu. This is when the high score is compared to the current score. If the current score is lower than the highscore, the high score will not be replaced and that game was all for nothing.

Core Mechanics

Wanting to go over the old ‘84 mechanics, the way the ducks fly and player shoots were mainly focused on.

Duck Flying

The way the ducks flew in the game was sharp but flew really crazy. Instead of making it sharp, to look more like a natural 3d game, the ducks would a force in its direction. But to keep the craziness of the flight pattern, the ducks random path and differing speeds made the game have a little bit more of a challenge.

Player Shooting

The old game had a 2d type shooting where it wasn’t trying to make the bullet catch the bird. Instead, the player had to try and aim directly at the bird and shoot. To replicate the player casts a ray when they shoot. If that ray touches any duck. The duck respawns. This will be a new experience trying to use the vr headset to line the bird up and have the player hit its target.

Technologies and Considerations

Screen Size and Aspect Ratio Differences

Windows X - 1920 x 1080 (16:9)
Oculus GO - 1280 x 800 (16:10) [Fullscreen] 640 x 800 (4:5) [PerEye]

API/ Software Requirements

Github Desktop - Backing up source files and game to source control

Platform Specific Features and Constraints

Gyroscopics

In both the VR headset and VR controller, gyroscopics give a new control to the game when playing on the oculus go.

Mouse Controls

Controls the head movement to force players to put ducks to the center of the screen, making it easier to aim.

Deployment Methods

Windows X - All deployment methods
Oculus GO - APK

Environmental Considerations and Handling

Technology	**Consideration**	**Handle**
VR hand controller	Damages to surrounding objects	Use head controls only to reduce players swinging

Risks and Managements

Risk	**Cause(s)**	**Management**
Loss of data	- Not backing up frequently - Forgetting to	Have a routine where: - Before working ensure everything is working, or moving on to the next section/piece of work. This ranges

	back up previously updated files - Backing up the wrong files	from (but not limited to) Backing Up files in to github, Being able to pull files from github, Checking wifi signals ETC - Frequently back up files into github, (do not need to back up if it is not necessary) - Check the files that are being backed up is the correct files - Ensure the files that have been backed up are the correct files