Juan Roman

Jet Set HoloLens

Technical Design Document



TDD

This page: Table of Contents and Team Member Listing

|  |  |  |
| --- | --- | --- |
| Table of Contents  [1 Game Overview](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel)  [2 Technical](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel) Summary  [3 Equipment](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel)  [4 Hardware](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel)  [5 Software](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel)  [6 Evaluation](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel)  [6.1 Game](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel) Engine  [6.2 Target](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel) Platform  [7 Sc](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel)heduling  [7.1 Development](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel) Plan  [7.2 Milestones](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel)  [7.3 Updates,](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel) maintenance and DLCs  [8](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel) Work Environment  [8.1 Remote](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel) Collaboration  [9 Levels](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel)  [9.1 Endless](https://n-xovwktmtjsnaxyc2mwes2xu7pohqedmdm6zjw5q-2lu-script.googleusercontent.com/userCodeAppPanel) Levels  10 Analysis   |  | | --- | |  | | Game Development Team Members  PRODUCER  Stephen Luu  Yoshi Nishikori  Josh Robey  Juan Roman  PRODUCTION MANAGER  Josh Robey  PRODUCTION COORDINATOR  Stephen Luu  Yoshi Nishikori  Josh Robey  Juan Roman  GAME DESIGNERS  Stephen Luu  Yoshi Nishikori  Josh Robey  Juan Roman  SYSTEMS/IT COORDINATOR  Stephen Luu  Yoshi Nishikori  Josh Robey  Juan Roman  PROGRAMMERS  Stephen Luu  Josh Robey  TECHNICAL ARTISTS  Stephen Luu  Yoshi Nishikori  Josh Robey  Juan Roman  AUDIO ENGINEERS  Stephen Luu  Yoshi Nishikori  Josh Robey  Juan Roman  UX TESTERS  Stephen Luu  Yoshi Nishikori  Josh Robey  Juan Roman |

# *Game Overview*

Title: Jet Set HoloLens

Platform: HoloLens

Genre: Genre of your game

Rating: (10+) ESRB

Target: Casual gamer (aging from 12 - 30)

Serious games

Release date:

Publisher:

Game Overview

Jet Set HoloLens is a 3d augmented reality spray painting game where the player is placed in a room (preferably with four walls) with certain places on the that need to be spray painted. The player must try and spray paint as much of these objects and accumulate as much as points possible before the time runs out. The player is armed with a spray paint can with others lying around. While possible to pick up other colors it while add more time wasted and may lead to some lost points.

*Technical Summary*

Jet Set HoloLens was developed in roughly two months by four people using Unity game engine alongside Windows Visual Studios. Different assets were used to create this game. The most important asset of all, the spray can was created by our own Josh on his laptop. The production cost to create this game was $0 although we did need some assistance to get our hands on a HoloLens device which was provide by a fellow student working with the university.

The game will be deployed on PC

Minimum requirements:

OS: Windows 10 SP2+

Graphics card: DX9 (shader model 2 .0) capabilities; generally

everything made since 2004 should work.

***Equipment***

*Hardware*

Members of the team utilized a collection of hardware from home. The game was mainly developed on team members Josh and Stephens hardware. Josh used an Asus laptop and Stephen used a Dell. Nothing was purchased in order to create this game. Assets that were used were either imported or created by our very own. In order to deploy and play the game a HoloLens is required. This was supplied by the school.

Product: Windows Surface 3, Asus, Dell, MacBook and HoloLens

Windows Surface: $800

Asus: $650

Dell: $650

MacBook: $850

HoloLens: $3000

Task: Asset Creation, Game Development, Texture Painting

Cost: $800

Total: $800

*Software*

All software used for development of Jet Set HoloLens will be able to produce high end visuals but requires a HoloLens device along with a Windows 10 operating system. Not all members were able to run these software tools due to limitations of their hardware.

***Evaluation***

*Game Engine*

The game engine used for the development of Jet Set HoloLensis Unity because we can create a 3D game with ease, we can make highly-optimized and beautiful, and we can deploy it on almost PC machine. We were limited to a PC only title because of the limitations that came with the HoloLens. Along with the PC only limitation, it is required to have a machine running Windows 10.

*Target Platform*

Jet Set HoloLens will be deployed on PC. The PC platform is a requirement from the HoloLens machine along with a Windows 10 operating system. However, the PC platform was the perfect target for this game as it helped us understand how develop a game for the HoloLens machine.

***Scheduling***

*Development Plan*

The development of this game was a little difficult compared to others. We began using class time to start the development process which were every Thursdays night. The issue then became that we needed more time with the HoloLens machine itself. Once we could schedule more time with the machine we would set a day and time and work on it from 2-4 hours.

*Mile Stones*

March: Team forms and settles on HoloLens project

April: Spray Paint idea becomes the main goal for the game. Able to push a scene into the HoloLens

May: First Beta

May 11th: Game Complete

***Work Environment***

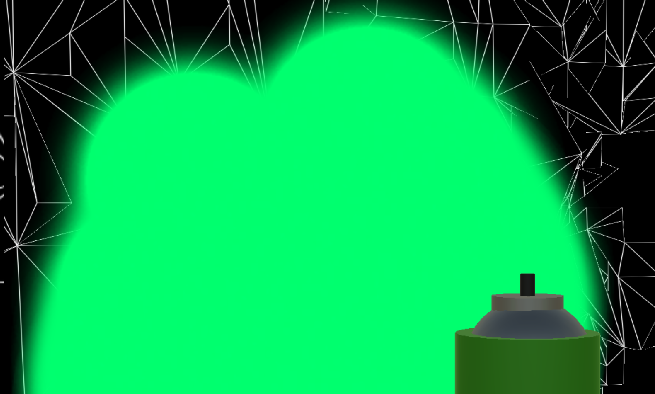
*Remote Collaboration*

Locations for the development of the game changed as the months went by. At the beginning of the semester we began brain storming in GMCS. Eventually class was moved to Adams Humanities. Besides class we had meetings in room 1113F

***File Formats & Naming Convention***

Assets: Spray Can, Posters and a very special Easter egg dedicated to Professor Price.

***Levels***



Jet Set HoloLens is one level. This level is based on the location of the player preferably a room with four walls. The reason for this is because the HoloLens will scan the room and create the level the player is set in.

Players: Player 1

Enemies: None

Props: Tables, Chairs (depending on location of player)

Environment: Room player is set in.

***Analysis***

Using new technology like the HoloLens came with new challenges. The first big challenge was getting our game to scan the room and make it part of the game. The script that controls this is the PlaySpaceManager. The great thing about this script is that not only does it scan the room but the objects in it like tables. Then there is the SpaceColletionManager which handles want is placed in the world and is place able inside the world. This script generates place able object in the world and sets them on planes that match their affinity. Then there is the InteractibleManager which keeps track of which GameObject is currently in focus for example like the Spray Can. And finally, there is the Placeable script which handles where the GameObject can be placed.

UPDATED FOR ISSUES ENCOUNTERED

as requested by Professor Price

-Require constant access to HoloLens as a group with a Windows 10 machine to push changes, making incremental changes impossible.

-Visual Studio solution building and sending to HoloLens was a massive hindrance on development time.

-Combining HoloLens user input like Tap with Unity systems like disabling particle systems failed to work for our group.

-"Room" scanning made gameplay and demonstration awkward. **Holograms 230 Microsoft tutorial was a great starting point** for terrain scanning and object generation.

-Insufficient scanning and terrain generation speed breaks any gameplay dependent on terrain interaction. Objects falling through unscanned areas, objects being unable to spawn because no matching type of terrain like "wall" or "floor".

-Particle system has to be created with care, as your computer may be able to handle complex movement of hundreds of particles, but the HoloLens will slow to a crawl and possibly even freeze.

-Scene transition using SceneManager failed to transition between scenes, works in Unity player.

-UI would not update on Hololens, works in Unity player.

-Wanted to add different colors to spray painting. Toggleable spray can instead of "always spraying". Multiple levels to allow extended gameplay and overall sense of game scope. Wanted spray can to follow hand, but could not get recognizer to follow, instead can is attached to main camera and follows Gaze input.