Product Requirements Document
ARails

Jenny Kang, Sonja Khan, Riley Porter, Nicholas Reiter
{jskang, sonjak3, rileymp2, nreiter}@cs.washington.edu
4 / 11 / 2016

Summary

The product is an augmented reality track building game that lets users lay down tracks around their physical space and simulate vehicle motion on those tracks. Building this app will contribute towards widening the scope of available applications on the HoloLens. Using AR allows us to use many of the benefits of physical and digital analogs to minimize the disadvantages of each, resulting in a more immersive and enjoyable experience.

Deliverables

A track simulation application built for the Hololens with the following features:
- Ability to place tracks
- Spatial mapping to place tracks within the room
- Simulate realistic motion along the tracks
- Spatial sound effects
- Texture and visual effects
- Player interaction with finished tracks
  - One of either racing or first person perspective

In addition to the core features, we have a list of stretch features that will be delivered should our schedule allow us to build them.
- Allow multiplayer collaboration
- More player interaction
  - If we implemented racing, we'll also do first person perspective, and vice versa
- Additional track types and obstacles
- Transferring tracks between locations

Besides the core application, we will kickstarter-style video promoting our product and a website with weekly blog posts detailing the progress of our development.

Critical Features

Very Critical:
Tracking building
   A user should be able to build connected tracks around their environment
Vehicle simulation

A user should be able to see a vehicle traverse their track in a physically realistic way

Critical:
Environment Aware

Spatial mapping to allow users to build along physical objects and support effects such as occlusion. Spatial audio to enhance the experience.
Save for later

A user should be able to save a track that is in progress and return to it at a later time.
Interaction

The user should have the option to play some sort of "game" with the built track, like racing or first person perspective

Less Critical:
Collaborative building

Two users should have the option to build on the same track.
Track rendering

A user should be able to change the look and feel of their tracks.

Performance Metrics

Functionality - Can users accomplish the following tasks (with no, minimal, or explicit instruction):
- Place a track segment
- Build a 'complete' track
- Skin/render the completed track
- View the track in motion

Performance - Interface and track should not be laggy on display. Interaction should be smooth and timely. Should not excessively drain the battery life.

User Interface - Should be unobtrusive, but be accessible when desired.

Smiles - Are people having fun? Would people use this? Would it motivate a HoloLens purchase? Do people lose track of time playing?

Milestones

Week 1

Milestone: Team assignments and project brainstorming
Responsibilities:

Everyone: Set up collaborative space on Google Drive, brainstorm potential projects.

Week 2

Milestone: Project Proposal Document and website
Responsibilities:

Jenny: proposal - challenges, budget, and feedback
Sonja: website
Riley: proposal - features and timeline
Nick: proposal - example scenarios

Week 3

Milestone: Product Requirements Document and Kickstarter style video

Responsibilities:

Everyone:

        Explore implementations, assets, resources

        Discuss design of multiple approaches

        Begin prototyping

        Video

        Requirements Doc (so meta)

Week 4

Milestone: Design and implement backend representation of track

Responsibilities:

Jenny: Track/rail asset investigation/dev

Sonja: Curves, and general bezier collection representation

Riley: Animation planning/framework investigation

Nick: Menu/UI prototyping

Week 5

Milestone: Ability to add straight tracks that are procedurally generated and move an object along it.

Responsibilities:

Nick: UI for adding tracks

Sonja: Procedural generation of mesh, one mesh per track

Riley: Object animation

Jenny: Texturing

Week 6

Milestone: Ability to create tracks in real-time that aren't straight

Responsibilities:

Nick: Solidify user interface for selecting and adding track pieces

Sonja: Work out mesh generation bugs. Render as one big mesh instead of blocks.

Riley: Refine animation, spacial sound

Jenny: UI for theming/texturing

Week 7

Milestone: Spacial mapping. Progress Report

Responsibilities:

Nick: Tunneling through surfaces

Sonja: UV coordinates and edge cases of mesh generation

Riley: Animation through surfaces

Jenny: Visual display on surfaces at intersections (tunnels, etc.)

Week 8

Milestone: Ability to adjust orientation of track to not always point up.

Responsibilities:

Nick: UI for adjustments
Sonja: Curve adjustment
Riley: Animation adjustment to follow orientation
Jenny: Visual cues adjusting to surfaces

Week 9

Milestone: Some sort of non-building interaction like FPP or racing. Investigate multiplayer.
Responsibilities:

Nick: First person perspective
Sonja: Investigation for racing
Riley: Multiplayer building
Jenny: Multiplayer building

Week 10

Milestone: Save and load track. Further multiplayer work.
Responsibilities:

Nick: UI for saving/loading, file format for visual bits
Sonja: Curve file format
Riley: Further multiplayer exploration
Jenny: Further multiplayer exploration

Materials and outside help needed

3D Assets: Various track textures and related assets, sounds for different themes (roller coaster, train, hot wheels, etc.)
Scanner: Potentially needed for scanning in basic track pieces for digital dissection
Software: Rollercoaster Tycoon, NoLimits, Minecraft for exploration of existing track paradigms
Battery packs: Allow testing/movement untethered, but still charging the device
Help: Various technical resources for various parts of the project

Budget

PC games ($100)

We plan to spend a small portion of our budget ($100) to buy popular PC games (NoLimits and Roller Coaster Tycoon World) that are already on the market in order to better understand this space. NoLimits is a game that models realistic roller coaster tracks, and allows the player to experience the ride in first person. Roller Coaster Tycoon World was one of the games that inspired our project idea, and it will be good to see what this game does well and how our project might build or differ.

Miscellaneous Items ($32)

We plan to spend $12 to purchase a domain name, and $20 to purchase pocket-size battery packs so that we can be free to move around a larger space while developing for long stretches of time.

Assets ($845)

We plan to spend the rest of our budget buying assets for the roller coaster tracks and roller coasters, and assets/sounds for special effects.
- Tracks and Rails ($45)
- MegaShapes ($100 x 4, requires one license per seat)
- Models & Textures ($300)
- Sounds ($100)

Risks and how risks will be addressed

Risk: Performance. Procedural generation of meshes can be computationally expensive, and optimization may be challenging.
How to Address: Use only very simple shapes. Constrain to prebuilt meshes (track creation will be heavily limited with this solution).

Risk: Energy consumption. Rendering complex scenes will drain battery life faster than simple scenes.
How to Address: We will have two separate modes, one for building a track that only uses very simple textureless meshes, that emphasizes segments to allow better editing, and a render mode that will contain more complex visual detail, and focus on animation and display.

Risk: Feature Creep. There are many features and extensions that we would love to implement, which puts the project at high risk for feature creep.
How to Address: Ruthlessly prioritize. We have compiled a list of features in order of importance, and have split up the work in a way that allows for independent development by individual team members to be as efficient as possible.

Risk: Running out of time. There is the potential to get stuck spending too much time on smaller parts, and not moving along in development, particularly given the limited timeframe of the course.
How to Address: We have created a schedule that will be revised weekly to ensure we will have a complete deliverable by week 10. We have split up the work in a way that allows for independent development by individual team members to be as efficient as possible. Weekly meetings will allow us to adjust the current docket of tasks, and make sure people are on track (:P), and not blocked.