Project Overview

• Distributed Power Handling Simulator
• Built in Unity with PS Technology’s Physics API
• Control Throttle and Braking of Train
• Keep Buff and Draft Forces in Check
• Complete Level to Unlock Tracks and Trains
• On-The-Go Training for Locomotive Engineers
• Cost-Effective
System Architecture

Core Technologies
- .NET
- C#
- PST
- Programming Framework
- Physics API

Development Tools
- Unity
- Development Environment, Game Engine
- Visual Studio
- Code Editor

Build Platform
- Windows

User
Main Menu

Mobile Train Handling Simulator

- Start Game
- Options
- Quit
Track Select

STRAIGHT 100

SELECT

PREVIOUS NEXT
Train Select

2x2x0
118 Cars
Game Scene
What’s left to do?

- Update UI
  - Implement Weight Graph
  - Implement Force Graph
  - Implement Force Gradient Key
  - Implement Pause Button and Screen
- Import All Tracks, Create Levels
- Saving and Loading Level Progress and Unlocks
- Dialogue System for Tutorial
Questions?
Updated UI Mockup
Train Generation
Train Generation

• Very Easy, Just Hit Play!
• Read CSV Files -> Produce Train Prefabs
• Set Real-Life Meter to In-Game Meter Ratio
• Can Modify Any Components As Needed
• Easily Scalable using Unity’s Transform Feature
Track Generation
Track Generation

• Very Easy, Just Hit Play!
• Parse XML Files -> Produce Track Prefabs
• Set Real-Life Meter to In-Game Meter Ratio
• Can Modify Any Components As Needed
• Easily Scalable using Unity’s Transform Feature
• Can Make Segments of Tracks
• Can Select How Often Points Are Sampled
Scaling Everything

• A Difficult Task

• Problem: How to Take a 1.5-Mile-Long Train Moving 22 mph on a 36-Mile Track and Create a Level Lasting Five Minutes?

• Answer:
  ▪ Train and Track Actual Size to Start
  ▪ Squish on X-Axis, Amplify on Y-Axis Appropriately to Fit Camera
  ▪ Simulate Game at 50x Speed