

Fire Degree

Game Degree Project Report
Robert Bünsow

A Venture in Technical Game Design, Sound Design and a bit of 3D
Artistry

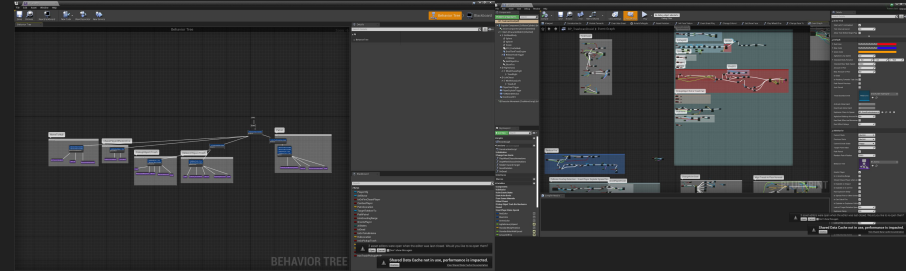
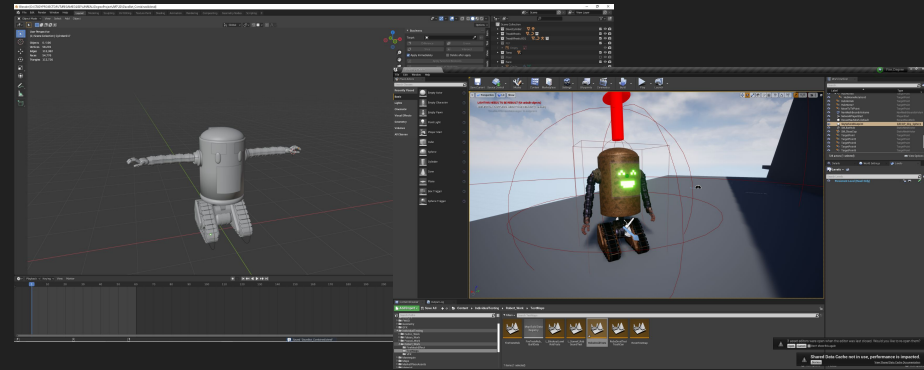
The Baby Droid Week 44 - 46

Focus on Modelling, AI and Systems Design.

3D modeled, textured, rigged, animated in Blender and implemented all the base mechanics of our principle **Baby Bot Droid** for the game through **Blueprints** and setting up the **Behaviour Trees** inside Unreal Engine. I created a Parent **BP_TrashCanDroid** which contained all the functionality, states, player detection, AI behaviour which we could then use as base class for all other children variants of droid. Plus we could work on the different variants at the same time thus avoiding that potential blocker.

Mechanics/States include:

- Patrol
- Pickup Trash
 - Open lid
 - Pickup
 - Dump at Location
- Different greet player states
 - Wave
 - Point
 - Agitation States Flail Hands, Wobble Body
 - Face Screen (Emojis)
 - Change Texture
 - Change Color
- On Fire state
 - Chase player
 - Go back to a (Hub) location when fire is extinguished
- Using Niagara FX for OnFire Static and Skeletal Mesh assets
- Interface functions that can be overridden (Very useful for child classes)
- Event dispatchers when entering states or performing certain actions for level events



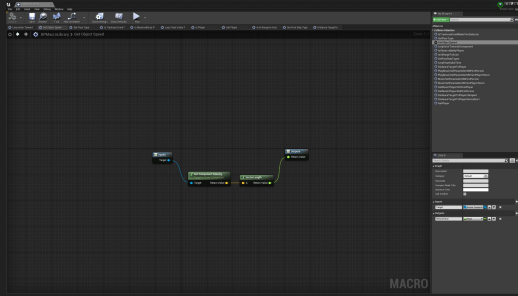
Challenges (Overcome)

- Importing setting up Blender Unreal workflow
 - Improved Modeling, Rigging, Animation skills
 - Learned a better Animation workflow in Blender, NLA strips
 - Improved creating Textures and UV Mapping skills
- Treadwheels align to surface normals
- Rotate torso towards target
- Trash pickup objects
 - Too complicated at first, should of stuck to faking the behaviour(sleight of hand) instead of simulating this full cycle(that we kind of got now).
 - Pickup Object
 - Open Lid
 - Drop Object in Can
 - Go to location
 - Release Object at location

T.* Game Designing Week 44 and onward - Macros, Event dispatchers, Highway Level

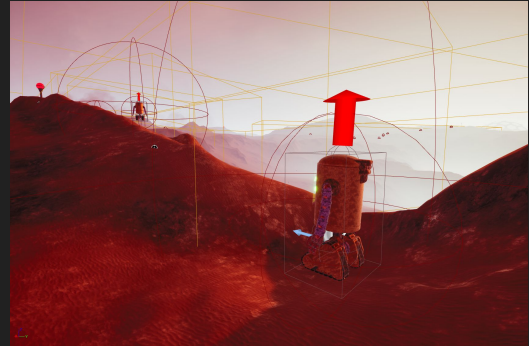
During the project I focused on setting up various helper functions through macros, learning more about having event dispatchers(observer listener principles) in Unreal, and certain systems managing “has player done all of this, do this” aspects of the game aside from bug fixing, and polishing. I also worked a lot on the Highway level of the game, building, placing level assets such as the droids, dust particle fx, roads etc. setting up all of the droids behaviour, navigation, etc.

* Technical, this project was very much hands on implementing systems for me. Apart from everything else hehehe...



Created macros for checking if an object is observable by player, getting speed of object, setting parameters in FMod events and more to help development.

Worked on the Martian Highway level of the game, where I set up the droids pathfinding and mechanics concerning greeting player driving by and dropping trash around the planetside.



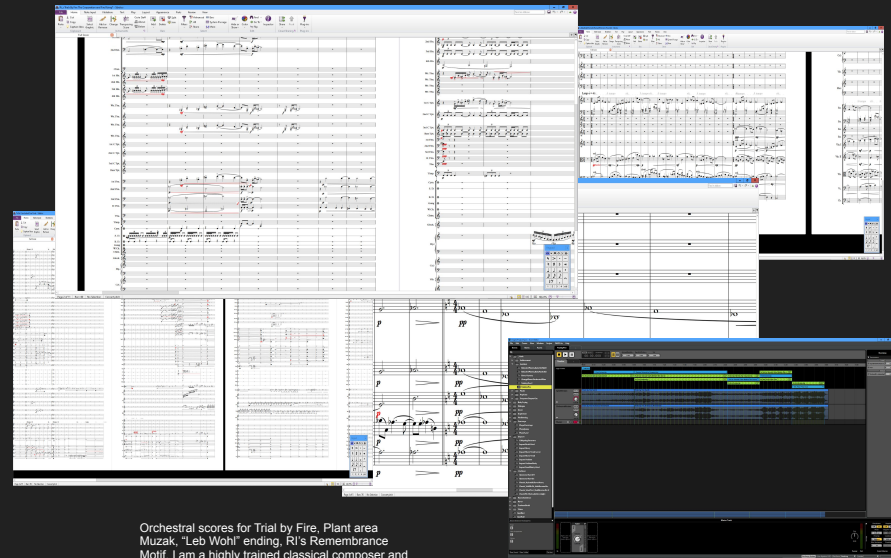
Did some level design and setting up martian dust particle systems to add to the feel and spectacular vistas of the games outdoor scenery. Game feel is important.

Audio Design

Week 48 and Onward -
Adaptive Audio in FMod, Composing Music and Making SFX

I implemented various sounds through samples and composite sound fx design(layers of samples creating a new sound in combination). Generated a generic pink and brown noise that could be EQ filtered and used for water, engine, background hum etc. We also wanted to test reverb so I learned how to use FMod's reverb inside unreal. Had various volumes placed throughout the level that triggered different convolution reverbs within the game to add to the feel of the environment.

My music was orchestrated, written, and exported using Notepformer sounds in Sibelius and then exported midi to a DAW where I added the synth instruments(U-HE freesynths zebralette etc.) and extra effects.



Orchestral scores for Trial by Fire, Plant area Muzak, "Leb Wohl" ending, RI's Remembrance Motif. I am a highly trained classical composer and write music fast.

FMod action block music areas, loops and parameters for transitioning between them

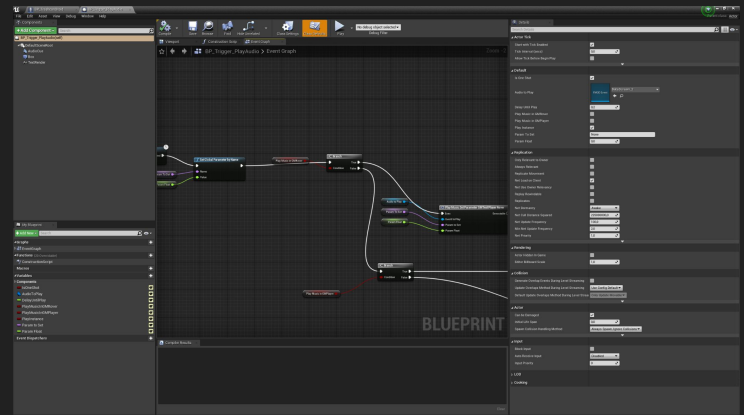
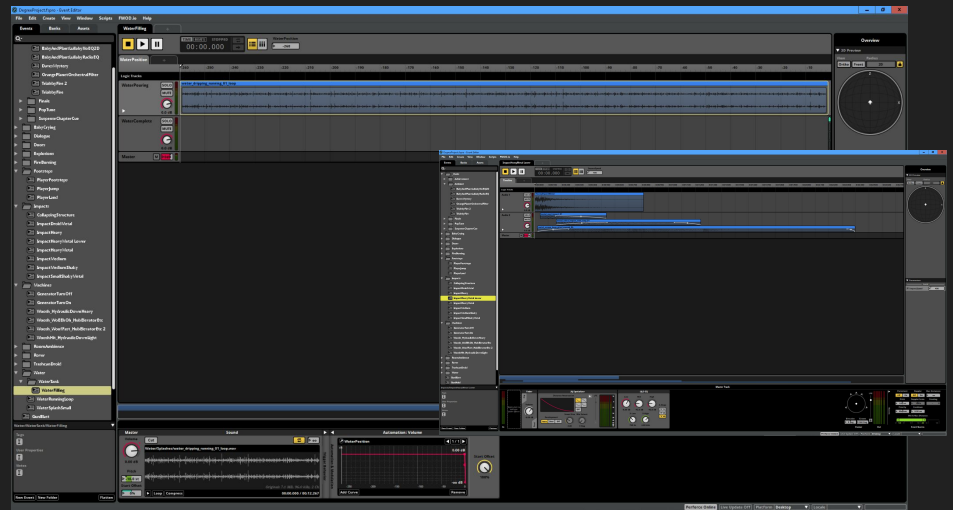
Challenges (Overcome)

- Fmod implementation in Unreal
- Adaptive Audio/Music
 - Parameter driven sounds
 - Speed of object when impact
 - Distance to object
 - Loops, transition points
- Reverb Zones
- Avoiding clipping distortion in audio samples (using fades etc)

Audio Design Continued

Various sounds implemented

- SFX
 - Player Sounds
 - Footstep(different areas)
 - Jump
 - Land
 - Water Gun
 - Droid Sounds
 - Baby(recorded my 7 month old baby) Distorted Voices
 - Vehicle Sounds
 - Impact
 - Engine
 - Speed controls EQ and Sounds of Wheels Squeeling
 - Fire Sounds
 - Amount of Fire fades between three sounds
 - Water Tanks
 - Fill water controls EQ HP Filter
 - Impacts
 - Speed of object
 - Metal
 - Soil
 - Explosions
 - Ambience
 - Various room noise
 - Generator sounds
- Music, Orchestral Score and Synthesizers
 - Scene X Intro
 - Transition to Chapter 1, Fire RI's Remembrance Theme
 - Chapter 1
 - Dave's Mystery/Corporation Theme
 - Gardening Muzak
 - Chapter 2
 - Planetside Music
 - Implemented Pop tune Violator by Brodet(not written by me!) with full rights
 - Chapter 3
 - Action Block Music, Corporation Theme
 - Finale, Leb Wohl, RI's Remembrance and Pop Tune Motifs/Themes
- Reverb Zones
 - Great Hall
 - Corridor
 - Outside
 - Small Tight Areas

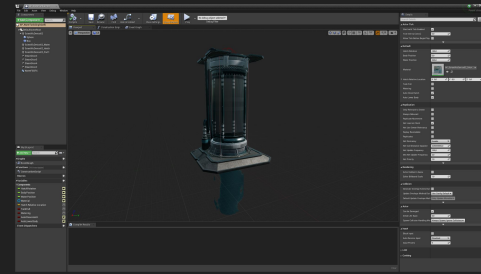


Created blueprints controlling, playing and setting parameters in the game

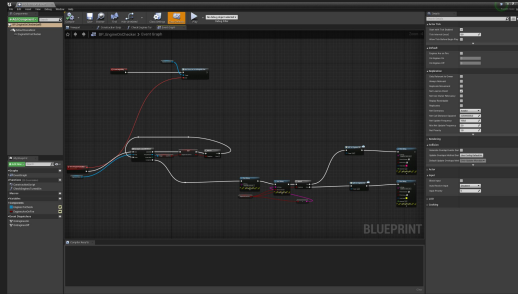
Final Development Stage

Late stage designing/implementing/testing

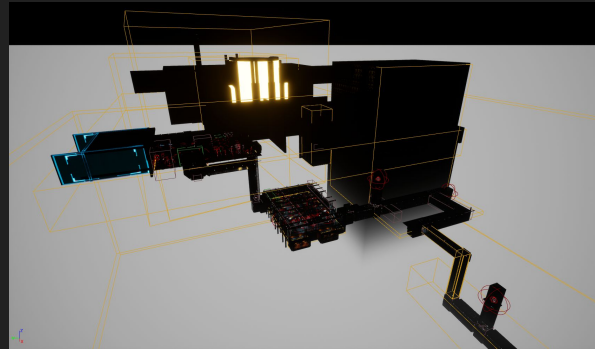
Focus on tying up all the parts of the game.



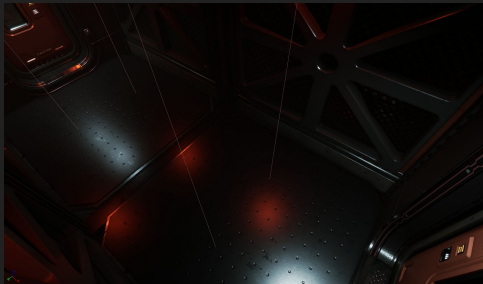
The water tank with the amount of water filled parameter controlling the audio.



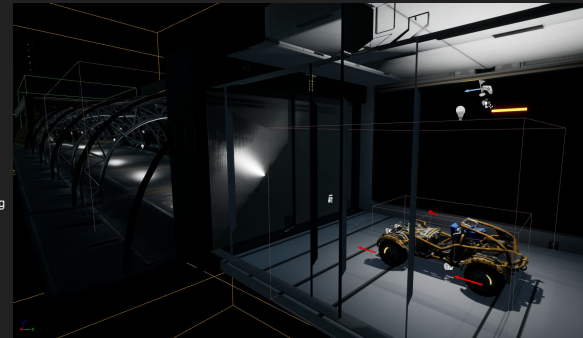
A simple system listening to if power generators are on fire or not and dispatching events for respective states. System can be called and used for any type of boolean checks.



The final action block sequences where I placed triggers for various music, sound areas, controlling parameters. I also implemented power generator checker system contained our class droid for action block functionality and in the end just bug fixing and testing.



Checking to see if a door is locked when passing through it needed an extra trigger or see where player exits so you can't get locked in by exiting trigger volume locking the door. Especially if, as seen here, is the final room before the end! XD



Transition triggers between levels. We fought with level streaming which proved more difficult than we imagined in the beginning.