

Development of My Tool

- General order of development operations - what simple task to start with as the POC and to build from
- Key things which it should streamline (task priority)
- I've read about development tools inside of VR - VR design tools - would that be beneficial?
 - Object placement (Place its location and set its position from within?)
 - Animation paths drawn with controller motions
- Collection of properly documented scripts
 - A set of scripts to deal with picking up objects and interacting with them in that sense
 - There's some crossover and the need to rewrite code between projects. Why not just create all those scripts, properly document them and make them available as a drag and drop feature

Existing Toolkits

- What tools are you currently using
- How do you find using the existing tools
- What's the most difficult part of using the existing tools
- What's the most straightforward part of using the existing tools
- What would you like the existing systems to do differently
- What can't you do with the existing systems that you want to do
- What do you feel about the existing systems wastes the most of your time
- Where are you forced to compromise using the existing tools
- Examples of where you struggle with the existing toolkits
- Is the general set up a problem
 - How would it ideally work (pick a target headset and load settings?)
- Examples of tasks you'd like to be automated and setup with minimal configuration
- Features from existing toolkits you'd like to see carried over
- Features from existing toolkits you'd like to see streamlined
- How would you prefer to use the existing tools
- Most confusing part of using the existing tools
- You mentioned using NewtonVR partly because of the audio service it provides that VRTK does not. Can you give some examples of the types of audio tasks and uses that you would require from such a system.
- Typical hardware / platforms developed for
- Is it annoying to have the Vive constantly wired in / is that disruptive

Existing Workflow

- Do you target one platform at the start of development
 - Yes - then which do you want
 - No - what's the set you target typically
 - Maybe - project dependent
- How do you currently go from idea to product
- Crossover between projects

- Difference in workflow when creating a VR movie type of thing opposed to interactive room
 - Do you want both or one
 - Is there a difference in creating one over the other
- How would you like that process to go instead
- If you could change / streamline the existing process how would you
- Where is the most time wasted / what can be automated and streamlined
- Repetitive tasks which occur across every project
- Notion of establishing defaults - default gravity - and would that work / how easily should it be to change (recall tweaking the health variable in old PC games - that simple)
- Most frustrating part of current development process
- Most simple / straightforward part of current development process
- Is it annoying to have the Vive constantly wired in / is that disruptive
- What are the types of experiences you currently and would like to create
- What's the most confusing part of going from idea to product

Misc Dr. Williamson Points

- The easiest way to deliver this will be to launch it on the Unity asset store. When should I aim to have the base key functions of the system launched on there?
 - How long does it take to go through certification
 - How long does it take to release a patch on there with additional functionality
- What headsets to target?
 - Aim for Vive and Oculus or just Vive
 - Start with Vive and implement Oculus at a later date?
 - Implementing Oculus could be a suggested improvement in the write up that I'm actually working on in tandem to the write up. Maybe it gets implemented the day after the dissertation is handed in.