APPVR - Summary of application requirements

UP739235

Przemyslaw Hendel

2016

**APPVR - summary of application requirements**

|  |
| --- |
| **Short description of the application** |
| The user after putting on the Oculus Rift and headphones, will be moved into a virtual barber shop where he will listen to the binaural recording that will have a virtual environment built around it.  The user will not be able to interact with the environment, and will have to sit in a virtual chair at a barber shop and listen to what will be happening. |
| **Application area and purpose** |
| The application area of this demo is entertainment, and this demos purpose is to test how the use of binaural sounds combined with a Head Mounted Display affect the user experience in the virtual world. |
| **Description of intended user group** |
| The intended user group for this application can be anyone over 13 years old as suggested by Brendan Iribe (co-founder of Oculus VR Inc.) because of the health issues that Oculus or any other VR technology might cause in children in its early days (Re/Code, 2015)  It would also be interesting to see people who have never went to a barber and never felt a hair clipper on their skin, particularly women. The Barber shop binaural recording itself feels very realistic and gives the person who is listening to it a sense of having their hair cut, and goose bumps.  But the application does not limit its users to women only.  Health and Safety should be considered when there is a user that would like to play the demo. The user must be supervised, in case of motion sickness occurring so the person who is supervising can turn the application off.  Oculus also has cables that the supervisor should watch out for so the user doesn’t tangle ad hurt himself by accident destroying the equipment at the same time. |
| **Input and output hardware** |
| Oculus Rift will be the hardware that will be used as an input/output device for this application. It was chosen becauseit has a high resolution (2160x1200) that allows a bigger field of view so the user can see more of the environment (which is crucial for this application) and because of its price that is $599 (which is still expensive) that has a lower cost than the HTC Vive HMD and technical specifications are almost identical.  Further in the development the demo will support the HTC Vive as well because Vizard also supports it. Vizards documentation is well written and has examples of how to integrate an application to allow the use of other Head Mounted Displays.  A cheaper option that the users might want to use because of the price of the two previous headsets could be Google Cardboard or Gear VR that Vizard also supports.    Also, the application will require a pair of stereo headphones so the user will be able to hear the binaural recording.  Mouse & Keyboard will be required because it will be a PC application that will require a Mouse to launch it, and keyboard to adjust sound volume. |
| **Software limitations or constraints** (does it support your chosen hardware? Do you need a plugin or sdk? Do you have the resources to make or acquire this? ) |
| The software that will be used for this application will be Vizard that supports Oculus and SteamVR HMDs as well as other displays such as gear VR and Google Cardboard (Worldviz, n.d.)  However, Vizard doesn’t natively support development for smart phones to make them act like a Head Mounted Display. However, this can be achieved by using a software/plugin called trinusVR that can be used to allow that to happen.  Future development of the application will include the application supporting the support of HTC Vive |
| **Description of user view of application** (what do they see, how do they interact, what feedback will they get) |
| The user wearing the Oculus HMD will be able to see the environment (inside of a barber shop) and see the characters that can be heard in the binaural recording. And because the binaural recording will not be 3D and it will play ‘on top’ of users’ head, users head moves must be limited.  The user will have to keep looking at one point so they can feel like they are in a real barber shop, if they fail to do that the experience might not be effective at all as the recording is not 3D.  There was an idea of locking the head movement in VR and not taking the sensor input from the HMD’s gyroscope making the user look at one point even if he moves his head in real life, but this can have negative effects on the user and make him feel uncomfortable and possibly make the user sick and destroy the experience.  To solve this, I had another idea that could possibly work, which is an implementation of a dot/crosshair and an object (3D model) in the environment or a circle that would be placed in the environment (GUI?) that the user will have to look at or keep the crosshair inside the circle.  If he doesn’t the environment would go gray, and the binaural recording would stop playing until the user is looking at the point that he is supposed to look at.  And because of Health and Safety, I decided not to lock the users head movements physically, so the user will not be hurt and the chances of the equipment being destroyed will be lowered.  An optional feature (further development) will be an implementation of a window in the environment where the user will be able to see people walking on a sidewalk outside the shop however this will be tested to see if it can go with the recording and not break the experience.  The recording that will be used will be a binaural barber shop recording that can be found [here](https://www.youtube.com/watch?v=IUDTlvagjJA). The recording is very immersive itself if the instructions at the beginning of the video are followed. |
| **Assets required (3D models, audio, animation files)** |
| **Sound**  -Virtual barber shop binaural recording  **3D Models**  -Guitar  -2 male human models (because they can be heard in the recording, Vizard models)  -mirror (must be tested to see if it breaks immersion)  - door  - phone  - some plants  - carpet?? (texture)  - paintings  - window  Optional 3d models  -Other human models that will be walking on a sidewalk behind the window  **Animations**  Animation files for the 2 male human models will be required so the user can see them moving around the environment and performing tasks such as opening doors in sync with the binaural recording.  Most of the models will be static and will just be placed in the environment, because the user will not be able to interact with them anyway.  The models I will use will be royalty free and I will use those websites to get hold of them.  <http://www.turbosquid.com/>  <tf3dm.com>  <http://archive3d.net/>  If I will not be able to find the model I want for this demo, I will use Autodesk 3Ds Max to create some of my models, and read the Vizards documentation to find out how to export the model so it will be supported by Vizard. |
| **Constraints** (e.g skills, equipment, safety, time) |
| Python 2.7 that is the programming language that Vizard uses, and will be used to create this application but both Python and Vizard are new to me so there a constraint is created.  There also is a risk that some of the features written in this document will not be implemented. However, I will try and learn Python as I start to develop the application, by reading documentation and researching solutions to problems that I might encounter online, and giving credit to other developers in my documentation as well as source files if I borrow some of their code.  Another constraint is that I don’t own Oculus and I will use Virtual Reality labs that are in Eldon Building, University of Portsmouth. But there might be a problem with booking the HMD if someone else books it before me, making me do less work & testing. But this could be solved by firstly creating this application for Google Cardboard that I own, and then making it support other HMD’s.    While developing the demo in the VR labs I will also have to have someone supervising me in case I get motion sickness or tangle myself because of the cables.    Time must be managed properly so I have to have a set number of hours I will try to spend weekly developing the application before the deadline as there is other coursework that I have to do for other units. |
| **Data recording** (do you need to record behavior / performance into a file? If so, describe the data and the format you want to record) |
| **-** |
| **Real-time input** (do you want to be able to change any parameters while the application is running? E.g. speed, number of objects in scene, audio settings………) |
| The application will have an option that will allow to adjust sound volume while the application is running using the keyboard.  HMD gyroscope input, to allow the user to move his head if he wishes so. |

**SMART Requirements**

Black – minimum requirements

Blue – Desired requirements

Red – Ideal

Specific: Features to implement list

* Binaural sound playing on application start
* Support for at least one HMD
* Found models to implement into the environment
* Human model animations
* Animations are in sync with the audio
* Support for at least two head mounted displays
* Crosshair
* GUI circle
* GUI where the user can choose when to start the experience
* Animations and sound stops playing while the crosshair leaves
* Keyboard can adjust sound volume
* Support for 2 HMD’s, Google Cardboard and Gear VR
* Window in the virtual environment
* People walking outside the window
* Tutorial

Measurable: Does it work?

* Does the HMD work?
* Can Audio be heard through headphones?
* Is the sound binaural?
* Can the user see the environment?
* Can the user see human models and their animations?
* Is the animation in sync with audio?
* Does the sound and animation pause when the crosshair leaves the GUI circle?
* Are the animations in sync with audio?
* Does SteamVR headset work with the application?
* Does Google Cardboard work with the application?
* Is there a tutorial?
* Window with people walking outside of it doesn’t break the experience?

Agreed: What to do?

* Get 3D assets that are required
* Play audio and animations at the start of the experience (if menu was not implemented)
* Create animations that sync with audio
* Implement support for Oculus Rift - IMPORTANT
* Implement pausing of the experience while the user is not looking at the point/object he should be

Realistic: Goals I believe I can achieve

* Support for Oculus and SteamVR
* Sound playing
* Animations
* 3D models in environment
* Keyboard adjusts volume
* Pausing/replaying depending on where the user is looking

Timebound:

* Get the HMD to work first (Book VR lab asap)
* Find out how to get sound into the application
* Animations
* Animations synch with audio
* 3D Models placed in the level
* Implement GUI ( for pausing the experience)
* Main meu
* Sound volume adjusting with keyboard
* Support for SteamVR headset

**Project Plan**

|  |  |  |
| --- | --- | --- |
| Implementation and Testing | Goal | How long will it take? |
| HMD shows the environment | Book VR lab and make Oculus display the environment | 1 day |
| Find assets on websites listed above | Find assets and store them on GitHub | 4 days |
| Create environment by adding models | First design on paper how the room might look like basing it on the binaural recordgin | 3 |
| Create animation | Read documentation, and search the internet | 6 |
| Make sure the animations sync with the audio | Keep adjusting the animations and animation paths | 6 |
| Implement pausing |  | 1 week |
| Implement support for SteamVR | Read documentation | 1 day |
| Implement a tutorial where the user is being told what to do and not do |  | 5 days |
| Implement adjusting sound with keyboard | Read documentation to find out hwo to get user input | 1 day |
| Google Cardboard support | Read documentation and find out how to set up TrinusVR | 4 days |
| Implement window |  | 1 day |
| Make people walk outside the window | Create new animation paths | 3 days |

**Methodology**

The methodology I will be using for this project is Waterfall, because the project is rather simple and does not have a long list of features that need to be implemented.

I will choose the requirement I want to work on and work on it until it is finished, then I will move to the next one. I will not be going back to fix stuff that is broken, because I will try to make it in a way that it can’t break after other features will be implemented.

Then I will test the implemented features and move to the next requirement.

When all requirements are completed the application will be released.

Choose requirement

Implement requirement

Test Requirement

And fix any errors until it works as intented

Release new Version

# References

Re/Code, R. M. (2015, May 28). 2015 Re/Code - brendan Iribe Clip. Retrieved from https://youtu.be/AN78F-Pnzpw

Worldviz. (n.d.). Retrieved from http://docs.worldviz.com/vizard/#Introduction\_to\_displays.htm%3FTocPath%3DReference%7CStereo%20%26%20Displays%7CStereo%20Basics%7C\_\_\_\_\_2