



UNIVERSITEIT VAN PRETORIA UNIVERSITY OF PRETORIA YUNIBESITHI YA PRETORIA

Denkleiers • Leading Minds • Dikgopolo tša Dihlalefi

DEPARTMENT OF COMPUTER SCIENCE

WHOOSH DIVISION

OcuViz - EpiUse Labs

Vukile Langa

u14035449

Wynand Hugo Meiring

u13230795

Nontokozo Hlastwayo

u14414555

Gerome Schutte

u12031519

May 27, 2016

Contents

0.1	Vision	2
0.2	Project Scope	3
0.3	Architectural Requirements	4
0.3.1	Access and integration requirements	4
0.3.2	Quality requirements	4
0.3.3	Architectural responsibilities	6
0.3.4	Architecture constraints	6
0.4	Architecture design?	6
0.5	Initial Design	7
0.5.1	Technologies	7

0.1 Vision

The vision of this project is to enable everyday people to use the power of virtual reality to grasp a sense of scale in manners that have usually been misunderstood. This would allow easy and feasible comparisons, compared to their real-world counterparts.

0.2 Project Scope

The main purpose of the project is to enable data visualisation with the aid of virtual reality. Casual users will be able to select rather complex, pre-set scenarios to be visualised using the Oculus Rift, and the software will also allow generation of new scenarios from input data and 3D models which may be imported from a supported cloud store.

0.3 Architectural Requirements

0.3.1 Access and integration requirements

As with the technology being used, access would need to be using an Oculus Rift.

The application would need to be run from a supported computer. This computer would need to be a major platform, such as, but not limited to

- Windows
- Mac
- Linux

0.3.2 Quality requirements

Code needs to follow high development standards to produce professional code. Source code needs to be easily understood and maintainable.

Flexibility

1. File type support:

The system needs to support a range of different file types for both 2D and 3D files and input data:

- **.csv:** For data input, where each row represents a single entity.
- **.obj:** For object models. Various model stores provide models in this format, and many 3D modelling applications are able to export models in this format.

Maintainability

- The system should be modular and allow easy updating and fixing in the future.

Scalability

All major platforms should be catered for in support or at least be relatively simple to port to. This includes:

- Windows
- Mac
- Linux

An added bonus would be:

- Android
- iOS
- Any other mobile platform (Windows mobile, etc.)

Performance requirements

To ensure the best experience and prevent any motion sickness meeting performance requirements is crucial.

- Consistent and high frame rate (75 or more fps or 13.33 ms max per frame)
- Low latency between input and display

Oculus Rift's requirements for CV1 (Consumer Version 1) has the following explicitly stated requirements:

- NVIDIA GTX 970 / AMD 290 equivalent or greater.
- Intel i5-4590 equivalent or greater.
- 8GB+ RAM.
- Compatible HDMI 1.3 video output.
- 2x USB 3.0 ports.
- Windows 7 SP1 or newer.

Working backwards, DK2 requires at least a GTX 770 equivalent card or better owing to the fact that it has 25% less pixels compared to the CV1. Also it only needs to run at 75 fps compared to 90 fps of the CV1.

Reliability

- The system should be able to handle user input without crashing.

Security

Currently security is not part of the requirements, however this could change at any stage.

Auditability

Testability

There should be proper unit tests which cover all the contracts of the project. All tests should be automated and allow for both mock objects and integration tests.

Usability

- The system must have a simple and easy to use interface
- The system should not require training before use
- Designing scenarios should be simple

Integrability

Deployability

0.3.3 Architectural responsibilities

0.3.4 Architecture constraints

0.4 Architecture design?

Is this needed?? Subsections include: architectural tactics, architectural components addressing architectural responsibilities, infrastructure, concepts and constraints for application components

0.5 Initial Design

0.5.1 Technologies

Software

1. **Game engine:**

Developing for VR, the obvious choice was to use a game engine as the underlying technology. Game engines provide a complete solution to many challenges like Virtual Reality support, object modelling, scene rendering, texturing and model file type support, and because of this the game engine itself will form the biggest part of the project's technology stack. The game engine chosen to this end is Unreal Engine 4, as it provides:

- Cross platform support for Windows, Linux and Mac OS.
- Built-in support for the Oculus Rift
- Support for all general 3D modelling functionality, such as object modelling, scene rendering, texturing, etc.

Alternatives considered:

- (a) Unity 5
- (b) CryEngine

Hardware

1. **Oculus Rift VR Headset:**

The Oculus Rift is listed as a requirement by the client, and provides integrated support for Unreal Engine 4

. **Alternatives considered:**

- Since the Oculus Rift was specified as a requirement by the client, no alternatives were considered.