Architecture Design

Context: Health Informatics

Group: HI1 a.k.a. Geen Naam (Group Number 4)

29-4-2016

Group Members:

Name:	NetID:	Studentnumber:
Jonas Dieterich	jdieterich	4317424
Maiko Goudriaan	maikogoudriaan	4302125
Ruben Meeuwissen	rmeeuwissen	4287150
Louis Sikkes	lsikkes	4298977
Thijmen Jaspers Focks	thijmendewilde	4158393

Overview:

Introduction

This document provides an approximate overview of the system that will be built during the context project in cooperation with CleVR. The application will be presented in a component based manner, including the division into sub-components if applicable.

Design goals:

The following design goals are important for our project (Design Goals, n.d.):

Availability
At each end of the week's sprint we build a product that is ready to be used by the user. In this way we always have a working product to show to the user. The product has a minimised amount of possible failures. If there is a failure this is easy to fix, so the program will never have a long downtime.

Manageability
The product is easily manageable. The interface for the user is as straightforward as possible. You can click on an item and the possible actions of that item will appear. For more advanced options there is a manual, which explains how to change certain user preferences. The code of the product is well commented so developers who didn't work on the product, can read through the documentation and will understand the code.

Performance
The virtual reality world runs at 90 fps and should be fluent. The map that we're building should give the current overview of what is happening in this world and should also be fluent. Since we're giving commands to the world, and showing what is happening in there our map should be fluent. Since the plan of CleVR is to use our map as a basis to build other extensions on, like a notepad, if the performance is not good now than extending it is only going to make it worse. All actions should be short and easily processed to maintain high performance.

Reliability
Reliability is one of the most essential design goals due to the kind of medical simulation being run. The patient should at all times during the simulation be present in the virtual world and exposed to the virtual environment. Unexpected application termination could have serious implications for the mental state of the patient, as he might start panicing or reach a state of mental unease. Therefore, reliability will be prioritized at all times to meet the constraints of this health application.

Scalability
The product will at all times be scalable in the sense of being able to add new characters and interactive objects at will. Furthermore, new actions can be implemented and easily added in a modular approach. Moreover, additional features such as patient monitoring (e.g. heartbeat sensor) should be easily implementable, which will be supported by thorough documentation and strict code styling. This will allow the customer to expand his product from the point of handing it over.

Securability
Due to the secluded environment that this software is used in, there will be no major steps taken to secure the application. Furthermore, there is no remote possibility of disturbing the simulation due to the constraints of the setup. Additionally, since no data is collected in the process, there is no need to be anticipate data theft.

Software Architecture Views

The following chapter will discuss the architectural aspects of the application, by laying out the subcomponents and their interaction between them. The mapping between software and hardware as well as data management and concurrency will be discussed.

Subsystem decomposition

- Virtual World: The VR world is the environment that the patient will find itself in when using the VR goggles. It should be able to initialize the VR world and show it to the patient through the goggles. When given commands, like move person A to position X, it should execute these properly and update the world accordingly. It should also be able to send its current state to the therapist UI and should be able to respond to commands given by the therapist UI.

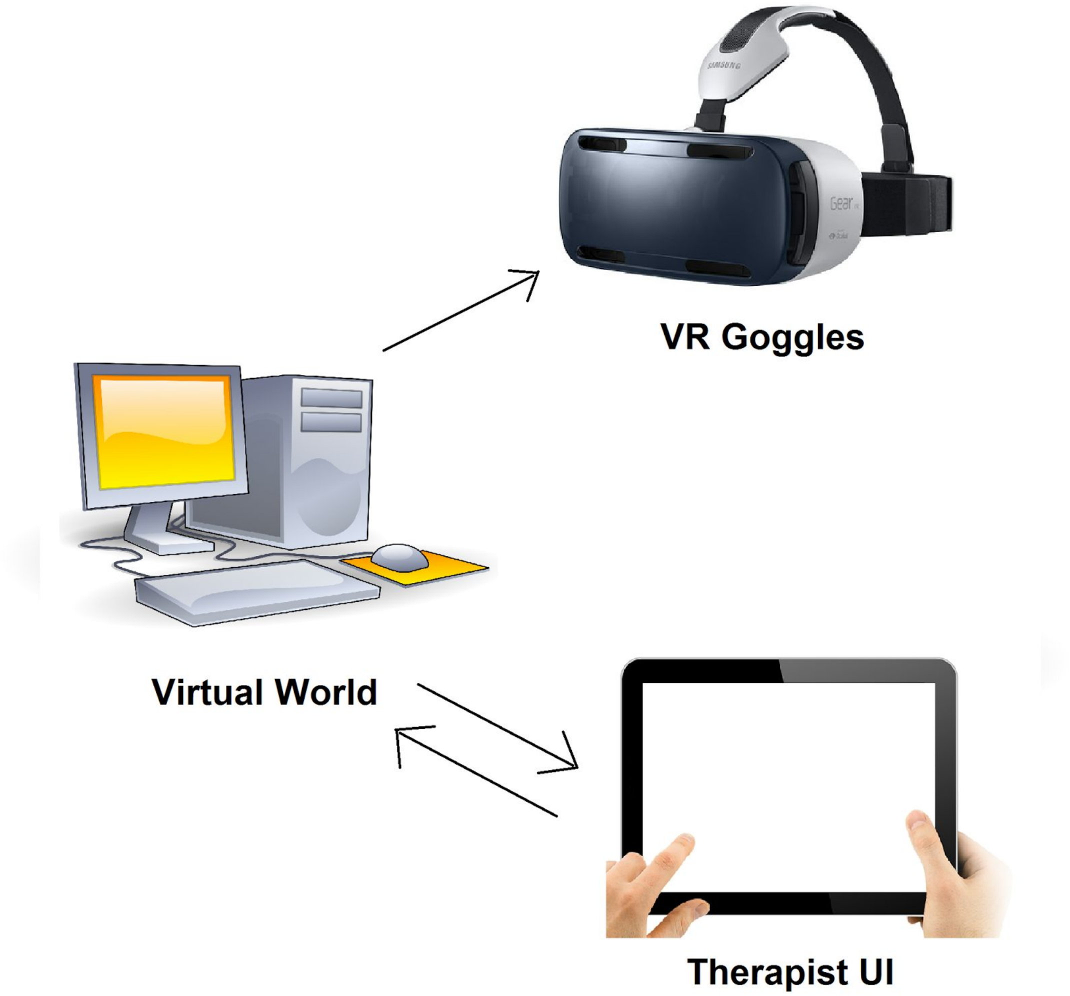
- VR Goggles: This is what the patient will use to see the virtual world. It should be attached to the virtual world and provide the patient with everything he or she needs to experience the situation.

- Therapist UI (2D map): The therapist UI is a 2D map and is what the therapist sees and uses to modify the current VR world to their needs. This UI will run on a separate tablet / computer than the VR world. The map should display the current state of the environment. The UI will receive the current state of the environment from the VR world and should make a clear view of it. The UI should also be able to give commands to the environment and display the changes that this commands makes.

Hardware/software mapping

- The communication between the VR-World based computer and the GUI based computer representing the 2D map will take place via the network. This will be implemented via a cable (Ethernet) connection.

- The communication between the VR goggles and the virtual world will take place through a USB cable.



Persistent data management:

We get data from the virtual world, this data is binded to the visual map. So emotions of the virtual characters in the virtual world are shown on the map. The data is also send in the other direction. So if the therapist clicks another emotion, this new data is send to the virtual world.

Concurrency:

There are two main processes running during the simulation, one on the VR-world computer (to which the VR goggles are connected to) and one on the UI computer. The graphical representation of the virtual world is constantly being sent to the VR goggles (via cable) to display the simulation to the patient. Furthermore, the current state of the VR-world is sent to the UI computer (via network), hence sharing the resources of the VR-world with the UI computer. Furthermore, there is a data exchange occurring in the other direction, namely the issuing of commands from the UI computer to the VR-world.

Glossary:

VR goggles - A virtual reality headset that allows its user to view and look around in the virtual world.
VR - Virtual reality
CleVR - Company based in Delft working on Virtual Reality Therapy using interactive virtual worlds that clients are placed in and can interact with.
UI - User interface
GUI - Graphical user interface

References:

- **Design Goals. (n.d.). Retrieved April 29, 2016, from https://msdn.microsoft.com/en-us/library/aa291862(v=vs.71).aspx**