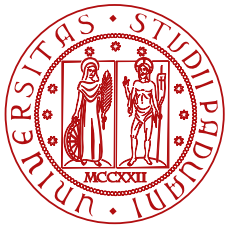


Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Master Thesis in Computer Engineering

Development of a virtual reality application for the training of cardiac surgeons

Master Candidate

Boscolo Cegion Nicola

Student ID 2074285

Supervisor

Prof. Savino Sandro

University of Padua

Academic Year
2023/2024

To my parents
and friends

Abstract

Contents

List of Figures

List of Tables

List of Algorithms

List of Code Snippets

List of Acronyms

VR Virtual Reality

PC Personal Computer

OS Operative system

HMD Head Mounted Display

UE Unreal Engine

HTTP Hyper Text Transfer Protocol

IP Internet Protocol

1 Introduction

1.1 Case introduction

At the university hospital of Padua, there is a very important cardiac surgery center where various heart interventions are performed on many people, the need to teach and visualize the case of operation is very important.
Thanks to MRI, doctors can see not only pictures of the heart, but also can even make 3D models. Each 3D model can show every detail of the patient heart, this can make medics able to analyze and show where and how to resolve the problem.

1.1.1 Virtual reality head mounted displays

The equipment: The university of Padua has some Meta quest 2[1] [fig:1.1] a Head Mounted Display (HMD) for Virtual Reality (VR).
The Meta Quest 2 is a standalone HMD, this means that they don't need other peripherals like an external console or Personal Computer (PC) for working.
For navigating, the Meta Quest 2 uses two wireless controller, but it also has the possibility to just use your own hands to navigate the interfaces. For the context awareness it uses 4 infrared cameras and sensors like multi axes gyroscopes and accelerometers.
They use a custom version of the Android Operative system (OS), this can give

[1] All rights to meta reserved

a certain degree of liberty in creating APPs for the device.

Figure 1.1: Meta quest 2

Use cases of VR: Principally the medics are using VR equipment for training and showing critical health conditions of different patients hearts. They are using an APP called Shapes XR, this APP has a web interface for upload 3D models and then show them on the HMD. The app has a multiplayer functionality so that multiple people can look at the 3D models in the environment, even if the developers recommend at max 8 people, they tested with 14 users connected and there weren't any problems.

1.1.2 APP PROBLEMS

How Shapes XR works: Shapes XR lets you create rooms, accessible via a code, were multiple people can create 3D models with basic tools like 3D brushes and standard shapes like cubes, pyramids and so on. It also let you upload a 3D model file on their own website, so that in the home you can download it and start to work on it. It let you also create your own avatar. This is the main feature that the surgeons are using for show the 3D Models

The Problems: Unfortunately Shapes XR is principally used for 3D modelling, so the app has a lot of features like changing the scale of the world or brushes for modelling the objects that aren't useful for the medics, and quite distracting

because for a lot of people is the first time using a HMD.
The user experience is extremely important in VR because is difficult to tutoring the user while is using the HMD. Then Shapes XR position the user in an empty 3D plane with little to none point of reference so If a user accidentally uses the teleport function, they may find themselves somewhere far away from the scene they are supposed to be watching.

Feedbacks from surgeons and nurses: The 12/12/2023 I have whiteness a lesson with the integration of the Quest 2 and Shapes XR. First it was pretty chaotic, a lot of people didn't even know how to use the controllers, and they had problems even for putting the code for entering the session. Unfortunately we didn't have time to make a nice lesson for teaching how to use the HMD, the tutorial that Meta it approximates takes 15min to complete, even more if the user want to try the mini-games. The main critical points were :

- Inadequate tutoring for teaching how to use the HMD
- Difficulty for accessing the multiplayer room
- Difficulty at moving in the room
- Some people avatar were blocking the visual of some people

Example of list

1.1.3 A subsection

Example of enumeration

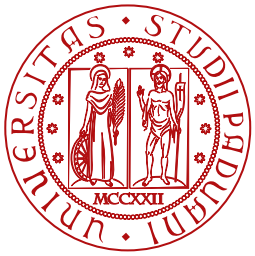
1. Item 1
2. Item 2

Example of quote

> *Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.*

2

Starting situation



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Figure 2.1: Example of image

3

Project management

4

Analysis of the requirements

Algorithm 1 An algorithm with caption

Require: $n \geq 0$
Ensure: $y = x^n$
  $y \leftarrow 1$
  $X \leftarrow x$
  $N \leftarrow n$
  while $N \neq 0$ **do**
    if N is even **then**
      $X \leftarrow X \times X$
      $N \leftarrow \frac{N}{2}$ {This is a comment}
    else if N is odd **then**
      $y \leftarrow y \times X$
      $N \leftarrow N - 1$
    end if
  end while

$$e^{j\pi} + 1 = 0 \tag{4.1}$$

5 The project

5.1 A section

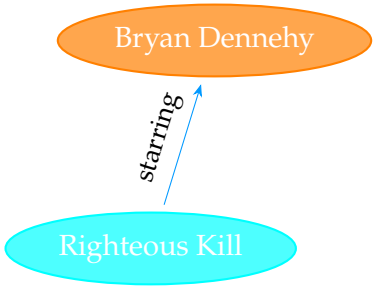


Figure 5.1: Image created with TikZ

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

```
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
```

```
    VT = np.zeros((n*m,1), int)  #dummy variable

    test = "String"

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

                if M is None:
                    M = np.copy(VT)
                else:
                    M = np.concatenate((M, VT), 1)

                VT = np.zeros((n*m,1), int)

    return M
```

Code 5.1: Code snippet example

6

Conclusions and Future Updates

A	**B**
C	D
E	F
G	H

Table 6.1: Table example

Acknowledgments