

## Masterarbeit

zur Erlangung des akademischen Grades  
Master of Science (M. Sc.)

Juan Felipe Diaz

---

Vorname Nachname

Biomedical Engineering, 2020, 5016761

---

Studiengang, Matrikel, Matrikelnummer

Theme:

### **Development of a Didactic VR Tool for Minimal Invasive Surgery**

Prof. Dr. Johannes Tümler

---

1st Examiner

Prof. Dr.med. Brigitte Kipfmüller

---

2nd Examiner

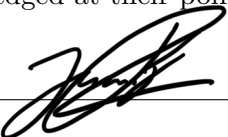
07.04.2023

---

Submission on

# Declaration of Authorship

I, Juan Felipe Diaz, declare that this thesis, "Development of a Didactic VR Tool for Minimal Invasive Surgery" is my original work except where otherwise indicated. I am aware of the university's regulations concerning plagiarism, including those regulations concerning disciplinary actions that may result from plagiarism. Any use of any author's works in any form is properly acknowledged at their point of use.

 \_\_\_\_\_ Signature

07.04.2023 \_\_\_\_\_ Date

# Acknowledgement

First of all, I would like to thank the University of Anhalt for its preparation and the commitment of its professors to provide their knowledge with such care. Especially to Prof. Dr. Tümler and Prof. Dr. med. Kipfmüller, who, both in their respective lectures and in the realization of this project, gave me invaluable advice and help, without which this paper could not have been completed. Especially because of the conditions under which this project was carried out. I would also like to thank all my friends, both the new ones I made during the duration of my master's degree and those who supported me from Costa Rica. Especially Carlos, Daniel, Gabriel, and Tales, you were and still are the engineers that I admire the most and whose support I have felt the closest. Los quiero maes!

Infinite thanks to Marie and my German family. It is with your patience (which I know you needed a lot of), support, and drive that not only this project but the master's degree in general was achieved. Thank you for everything. Finally, to my family, I can only say that this achievement is as much mine as it is theirs. Without you, none of what I have done could have been. Without your unconditional care, love and help I would not be all that I am today.

# Contents

Declaration of Authorship . . . . .	i
Acknowledgement . . . . .	ii
Illustration index . . . . .	vi
Table Index . . . . .	vii
<b>1 Introduction</b>	<b>1</b>
1.1 General Description . . . . .	1
1.2 Objectives . . . . .	2
1.2.1 General Objectives . . . . .	2
1.2.2 Specific Objectives . . . . .	2
1.3 Justification . . . . .	3
<b>2 Theoretical Background</b>	<b>4</b>
2.1 Virtual Reality . . . . .	4
2.1.1 History . . . . .	4
2.1.2 Current developments . . . . .	6
2.1.3 Operating principles . . . . .	6
2.2 3D modeling & Developing Software . . . . .	10
2.2.1 3D Modelling . . . . .	10
2.2.2 Development Platform . . . . .	11
2.2.3 3D Graphics Engine . . . . .	11
2.2.4 Unity . . . . .	12
2.2.5 Blender & Solidworks . . . . .	13
2.3 Minimal Invasive Surgery Principles . . . . .	14
2.3.1 General definition . . . . .	14
2.3.2 Tools . . . . .	14
2.4 The Gallbladder & Cholecystectomy . . . . .	19
2.4.1 General Anatomy . . . . .	19
2.4.2 Bile Duct & Gallbladder . . . . .	20
2.4.3 Pathology . . . . .	21
2.4.4 Surgical Procedure . . . . .	22
2.5 Pedagogy and education . . . . .	25
2.5.1 History . . . . .	25
2.5.2 Education today . . . . .	26
2.5.3 Evolution of education in the medical field . . . . .	26
2.6 Closing remarks . . . . .	27
<b>3 Related Work</b>	<b>28</b>
3.1 State of Art: VR in biomedical programs . . . . .	29
3.1.1 Using Virtual Reality in Biomedical Engineering Education (Singh, 2020)	29
3.1.2 Effective Design of Educational Virtual Reality Applications for Medicine Using Knowledge and Engineering Techniques (Górski, 2016) . . . . .	29
3.2 State of Art: VR for medicine & Minimal Invasive Surgery Simulation . . . . .	30

3.2.1	Virtual Reality Medical Training System for Anatomy Education (Ruthenbeck, 2015) . . . . .	30
3.2.2	Pre-graduation medical training including virtual reality during the COVID-19 pandemic: a report on students' perception (DePonti, 2020) . . . . .	30
3.2.3	Development of procedures for interactive simulation of minimally invasive surgical methods (Adler, 2014) . . . . .	31
3.3	Future Approach . . . . .	31
<b>4</b>	<b>Methodology</b> . . . . .	<b>32</b>
4.1	Study Methodology . . . . .	32
4.1.1	DOE: Design of Experiment . . . . .	34
4.1.2	Data collection instruments . . . . .	34
4.1.3	Questionnaire & Statistical analysis . . . . .	35
4.1.4	Statistical analysis tools . . . . .	35
4.2	Application Development . . . . .	35
4.2.1	Difficulties list . . . . .	35
4.2.2	Equipment . . . . .	36
4.2.3	3D modeling and editing software . . . . .	37
4.2.4	Virtual Assets & Objects . . . . .	39
4.2.5	Unity Development & Coding . . . . .	43
4.2.6	Coding for Physics Simulation & Items . . . . .	45
4.2.7	Teaching elements . . . . .	47
4.2.8	Working Method for Windows Simulation . . . . .	48
4.2.9	Morphological Box . . . . .	49
4.2.10	VR Integration . . . . .	52
4.2.11	Simulation Layout . . . . .	56
4.3	Closing Remarks . . . . .	63
<b>5</b>	<b>Results &amp; Analysis</b> . . . . .	<b>64</b>
5.1	User Test Results . . . . .	64
5.2	Case Study Results . . . . .	65
5.2.1	Questionnaire Results . . . . .	65
5.2.2	Relevant Variables . . . . .	66
5.3	Analysis & Discussion . . . . .	67
5.3.1	Descriptive statistic analysis . . . . .	67
5.3.2	2 Sample-t Analysis . . . . .	69
5.3.3	Relevant variables . . . . .	72
5.3.4	Outliers & Reviews . . . . .	74
5.3.5	Closing Remarks . . . . .	76
<b>6</b>	<b>Conclusions</b> . . . . .	<b>77</b>
6.1	Future Improvements & Recommendations . . . . .	78
<b>7</b>	<b>Appendix</b> . . . . .	<b>84</b>
Appendix A.	Complete pipeline for the creation of the simulation . . . . .	84
Appendix B.	Tools Used for the development of the project . . . . .	85
Appendix C.	Asset List . . . . .	86
Appendix D.	VR plugin Compatibility List . . . . .	87
Appendix E.	Presentation used for study . . . . .	88
Appendix F.	Code implemented . . . . .	98
Appendix G.	Case Study Raw Data . . . . .	106
Appendix H.	Questionnaire used for the case study . . . . .	107

# List of Figures

- 2.1 Milgram and Kishino’s Reality-Virtuality Continuum diagram [54]. . . . . 5
- 2.2 A) Sega VR-1, [6] B) Hololens 2 [9] . . . . . 5
- 2.3 HTC Vive VR headset and Lighthouse tracking system setup [10]. . . . . 6
- 2.4 Uncanny Valley curve [23]. . . . . 7
- 2.5 Example in the variation in the IPD among test subjects [15]. . . . . 8
- 2.6 Diagram of the functionality of both inside-out and outside-in tracking systems [19]. . . . . 8
- 2.7 Example of a trocar and it’s parts [14]. . . . . 15
- 2.8 Integrated Endoscopy latest Endoscope, with 4k wireless video transmission [13]. 16
- 2.9 Detail view of a grasping instrument developed with a new mechanism that allows 2 additional DoFs. [17]. . . . . 17
- 2.10 An example of a harmonic scalpel. [18]. . . . . 18
- 2.11 Divisions of the abdominal cavity [25]. . . . . 19
- 2.12 Blood artery connections to the gallbladder [31] & Anatomy of the gallbladder [27]. 20
- 2.13 Bile Duct connections [28]. . . . . 21
- 2.14 Trocar positions in the abdominal cavity in a patient with regular physiology [34]. 23
- 2.15 Gallbladder attached to the lower wall of the liver [38]. . . . . 23
- 2.16 Example of the correct position for performing the division of the cystic duct in a necrotic gallbladder [36]. . . . . 24
- 2.17 Implementation of VR for medical education purposes [52]. . . . . 26
  
- 4.1 Diagram of the study to be carried out . . . . . 34
- 4.2 Workflow chart for the 3D modelling Pipeline . . . . . 38
- 4.3 Front page for the Unity Asset Clinic-Operation Room by Mixall . . . . . 39
- 4.4 Endoscope Model by Hazam Eltaief . . . . . 40
- 4.5 Trocar model by "GulzarMrRobotics" . . . . . 40
- 4.6 Human body 3D model created by the artist Nico Robin. . . . . 41
- 4.7 Unity Asset Store banner for Yue Physics . . . . . 46
- 4.8 Working principle of BzSlicer . . . . . 47
- 4.9 Complete pipeline for the Windows Simulation . . . . . 49
- 4.10 VR Translation pipeline . . . . . 52
- 4.11 Scene Layout for the simulation . . . . . 56
- 4.12 Start screen & Main Menu view from the VR Headset . . . . . 56
- 4.13 Tutorial & Surgery Room view from the VR Headset . . . . . 57
- 4.14 Easy Mode & Hard Mode view from the VR Headset . . . . . 58
  
- 5.1 VR Group vs NonVR Group percentages comparison (Group Population) . . . . . 68
- 5.2 2-Sample t Test for the mean difference between the control groups . . . . . 70
- 5.3 2-Sample t Test for the standard deviation difference between the control groups 70
- 5.4 Histogram for the comparison between the VR and NonVR group data . . . . . 71
- 5.5 Crossed data for the VR control Group . . . . . 73
- 5.6 Crossed data for the nonVR control Group . . . . . 73
- 5.7 Indication of the 2 anomalies found in the results from the case study . . . . . 75

# List of Tables

- 2.1 The 10 most commonly used VR headsets according to VR Mark, application to test Head Mounted Displays (HMD)[16]. . . . . 7
- 2.2 Comparison between Unity & Unreal Engine [33]. . . . . 12
  
- 4.1 OS Comparison for VR Development . . . . . 36
- 4.2 Morphological Box for the 3D modelling and editing software . . . . . 42
- 4.3 Morphological Box for the Code & Programming section . . . . . 50
- 4.4 Morphological Box for the VR Integration section . . . . . 54
- 4.5 Morphological Box for the Start, Main and Tutorial Scenes . . . . . 58
- 4.6 Morphological Box for the Surgery Room, Easy and Hard Mode Scenes . . . . . 60
- 4.7 Morphological Box for the Virtual Library and Sandbox Scenes . . . . . 62
  
- 5.1 Study case questionnaire results in binary form . . . . . 65
- 5.2 Summary of the study case results . . . . . 65
- 5.3 Relevant variables per participant . . . . . 66
- 5.4 Summary of the relevant variables from the study participants . . . . . 67
- 5.5 Comparison of results between each group against the population, as well as each group against each other in percentages. . . . . 67
- 5.6 Cross reference data for the relevant variables . . . . . 72
- 5.7 Number of participants per field of study . . . . . 72
  
- 7.1 Tools used for the development of the entirety of the project in all its stages . . . 85
- 7.2 Assets used for the creation of the simulation . . . . . 86
- 7.3 Open XR & SteamVR plugins compatibility list with different HMD . . . . . 87
- 7.4 Case Study Raw Data as imported to Excel from Lime Survey . . . . . 106

# List of Equations

- 4.1 Level of confidence calculation . . . . . 33
- 4.2 Sample size calculation . . . . . 33
- 4.3 Margin of error calculation . . . . . 33

# Abbreviations

<b>AR</b>	Augmented Reality
<b>BME</b>	Biomedical Engineering
<b>HSA</b>	Hochschule Anhalt
<b>HMD</b>	Head Mounted Display
<b>IDE</b>	Integrated Development Environment
<b>IPD</b>	Interpupil Distance
<b>MIS</b>	Minimal Invasive Surgery
<b>SBT</b>	Simulation Based Training
<b>VR</b>	Virtual Reality
<b>XR</b>	Expanded Reality



## **Abstract**

Virtual reality (VR) devices currently offer multiple advantages that can be exploited in various fields, such as education. In the field of biomedical engineering (BME), which combines engineering and medical knowledge, students are often required to familiarize themselves with medical procedures. Currently, BME students can only witness these surgeries; they cannot participate in them or interact with them as they are not medical professionals. It is particularly in minimally invasive surgeries, where the learning experience requires a hands-on approach, that the understanding of the use of instruments and procedures in general is very limited.

This project proposes to use VR technology as an educational aid to allow non-health science students to directly perform practical sessions safely. Therefore, the main focus is on the creation of an educational VR simulation and providing a better insight into its usefulness as a didactic tool. This will be achieved with the use of 3D development software, thus providing a clearer perspective on the technique, equipment, and complications that can occur in a real surgery. In this simulation, the student can become familiar with procedures such as cholecystectomy, the necessary pre-operative tools, and the steps to follow to perform it correctly. The objective of this is to provide the student with a new perspective on minimally invasive surgery, a better understanding of the necessary equipment, and a new didactic tool for the teachers in charge.

For this project, anatomically correct 3D models combining specialized 3D libraries and real surgical data will be used. For the theoretical foundation, the didactic materials that are part of the curriculum of the master's degree in biomedical engineering will be used, along with consultations with professionals in the field, to get as close as possible to the reality of the procedure and the use of its tools. The development will be done using tools such as Unity and Blender, both for the modeling section and for the creation and implementation of the simulation, since both are industry standards and there are plenty of tools available for them. Finally, the VR implementation will be carried out using the equipment of the XR lab of the University of Anhalt, more specifically the HTC Vive, the Valve Index and their controls.

# Chapter 1

## Introduction

### 1.1 General Description

The field of education has remained in constant evolution throughout history, responding to the needs and developments of the time. The changes that have been made have always been gradual, adopting new methods and technologies after their effectiveness has been proven. Parallel to this, it is undeniable that during the last century there have been great advances in the development of new technologies. This development can be seen as exponential, from the invention of the first computer to the first laptop (which took 40 years), which was orders of magnitude more powerful, to the present day, where every 2 years on average the computational power of these devices doubles. This has enabled the creation of completely virtual spaces that resemble the real world, which was previously only conceivable in science fiction.

This is how the concept of Expanded Reality (or by its abbreviation, XR) was created and has since made enormous strides in the development of these virtual places. Even though these developments have occurred, they haven't been thought of for other purposes than entertainment until recently. This is how virtual reality technology (a branch of XR) has started to be incorporated in other fields, such as business, healthcare, and education.

The use of virtual reality (VR) technology in the learning process has been gaining attention in recent years, especially with the boom in the virtualization of education due to the global health crisis that began in 2019. The effectiveness of VR as a teaching tool has not been fully evaluated, due to the novelty of the field and the slow (but steady) implementation of the technology in educational centers around the world. Therefore further research is needed to determine its impact on student learning outcomes.

One of the few fields where this technology has been evaluated as a complement to education is medicine. In this field, several higher education institutions have adopted VR technology to provide students with the opportunity to learn more about the human body, reinforcing the concepts learned in class before attending or observing them in real life. Nonetheless, there are other fields related to health and medicine where this amount of research has not yet been done, such as biomedical engineering.

Biomedical engineering is the branch that combines knowledge of mechanical and electronic engineering with knowledge of the health area, such as anatomy, physiology, and pathology, in order to provide health professionals with new and better tools for the treatment of diseases and surgical procedures. However, there is a gap between the theoretical knowledge and the practice where this equipment is used. This gap is due to the limitations of biomedical engineers not being able to interact with the machines and equipment in real situations such as surgeries, which undermines the understanding of the difficulties and possible points of improvement for

the tools that currently exist. Although, there is little research on the implementation of this technology in the education of biomedical engineers; of the few studies that are available, the results are promising.

The motivation behind this project arises from the requirement for more research on the potential of VR technology as an educational tool. To gain a more comprehensive understanding of the effectiveness of VR in aiding the learning process of both undergraduate and graduate biomedical engineering students, a simulation of a surgical procedure from the University of Anhalt's curriculum is created. This simulation is then tested against the standard master class method currently utilized by higher education institutions.

This study aims to determine whether a VR simulation can effectively support the learning process of biomedical engineering students and whether it can improve their overall understanding of the surgical procedure in question. This project seeks to corroborate with statistical tools whether there is an appreciable difference between the standard educational system and the use of VR as a supplement, what factors affect the use of this technology, and to contribute to the argument whether VR technology is currently a viable tool as a complement to current teaching methods.

This document outlines the concepts required for comprehending both virtual reality technology and the surgical procedure that will be simulated, the steps followed to create both the simulation and the case study, and the reasoning behind the decisions made. Similar and relevant studies are also discussed, along with their current relevance and how the results from this document relate to them. In general, this research proposes the use of technology to expand the arsenal of tools available to educators. This in principle directly affects the field of biomedical engineering; however, the contributions mentioned here are easily adaptable for incorporation into any field of study.

## 1.2 Objectives

### 1.2.1 General Objectives

The main objective of this project is the creation of a virtual reality teaching tool and to demonstrate its functionality in the learning process for the biomedical engineering curriculum. In this simulation, the steps to follow for a successful cholecystectomy are demonstrated, making use of virtual reality development technologies such as 3D modeling software, the programming of these objects, and the user interactions that occur during the course of the procedure.

### 1.2.2 Specific Objectives

- Design an interactive simulation of a surgical suite with all the prerequisites for performing a cholecystectomy, since this is the surgery that presents the most amount of equipment relevant for Biomedical Engineering in the field of minimal invasive surgery.
- Create a 3D simulation of the surgical equipment, the accessible organs, and the elements, tools and machines involved in the procedure by making use of libraries of 3D models and original designs.
- Establish the steps to follow within the simulation to carry out the practical session correctly, thus creating an interactive guide for the user.
- Create the basis for future VR applications at the University of Anhalt, where the libraries and resources used are available for future developers.

### 1.3 Justification

Biomedical engineering largely combines medical and engineering knowledge into a single branch, with the objective of guiding professionals towards the development of new equipment that improves or facilitates the work of medical professionals. This means that professionals in this area need to familiarize themselves with knowledge in areas such as anatomy and physiology in order to understand the implications of the different medical equipment and techniques that are being developed in this area. Within the biomedical engineering curriculum, both undergraduate and graduate, there are constant references to model instruments and the surgeries in which they are applied. Many of these courses include sessions of visits to actual surgical procedures where these procedures and tools can be observed in action, as well as the testimony of the physicians who perform these surgeries. However, these experiences are merely visual, since students in these areas cannot perform these procedures unless they are professional physicians.

There are also laboratory sessions where the functions of more general equipment are taught (and where there is direct contact through simulation mannequins); however, these practical sessions usually do not include surgical elements but general equipment such as pulse oximeters or equipment to obtain an electrocardiogram. This poses an approach that essentially distances part of the theory from practice, since theory and testimonials fail to conceptualize all the details and difficulties that arise during the procedure. This is how a deep theoretical knowledge of the functionality of the equipment and techniques used by physicians is developed without fully exploiting the potential offered by practical knowledge. "Both theoretical knowledge and practical knowledge are equally vital; they are like two sides of the same coin. Understanding both extremes of the spectrum is essential" [2].

It is especially important for students who are part of these engineering programs to understand how these medical devices work in real life, the details they entail, and how they can be modified to advance the medical field. As mentioned in the American Journal of Surgery, students' learning is only stimulated beyond the limits of theory when they come into contact with the reality of the procedures [1]. This implies that for the development of knowledge, both parts of learning—theory and practice—must come together. This raises a fundamental question: how can these practical sessions be conducted in a way that is safe for students to learn without risking the physical integrity of any patient? One of the ways in which this can be done is through the use of virtual reality.

Currently, in the medical field, several virtual tools are already being used within educational programs as preparation sessions for real surgeries. These range from the complete use of virtual reality to the use of augmented reality, where real and virtual elements are combined. This is one of the reasons why the use of these technologies has only increased in recent years, particularly with the global pandemic of COVID-19, where virtual classes became the norm. A study conducted at the University of Insubria (Italy), where virtual reality tools were implemented as didactic methods for medical students, demonstrates the effectiveness of implementing VR for distance practice sessions. "This training modality was considered useful and met the expectations of most of the students in the last year of Medicine and Surgery" [3].

It is in this scenario that the need for this project was born, to reduce this gap in learning for the biomedical engineering curriculum, providing educators with a new didactic tool, students with the possibility to safely engage in surgical procedures, and future developers with a base that can be adapted to more educational needs. As mentioned in a study related to the inclusion of virtual reality in biomedical engineering curricula carried out by multiple universities of Philadelphia, USA, "preparing future biomedical engineers to have an ability to work in dynamic clinical environments warrants a more authentic approach to education than a traditional classroom and a one-time or short-term clinical immersion experience" [40].

# Chapter 2

## Theoretical Background

Biomedical engineering is a specialized branch of engineering that focuses on developing innovative technologies, equipment, and tools that can be used in surgery by medical professionals. In order to do this, biomedical engineers must have a deep understanding of anatomical concepts and procedures. However, it is important to note that the approach taken by biomedical engineers differs from that of healthcare professionals. Biomedical engineers focus on understanding procedures and the equipment used, determining which technologies are most relevant to surgeries, and identifying the reasons why certain pieces of equipment are preferred over others.

Despite the importance of basic healthcare knowledge in this field, there is a clear contrast between the knowledge required of a biomedical engineer and the extensive study and experience that physicians acquire during their formative years of learning about human anatomy, physiology, and pathology. **It is for this reason that although the source of knowledge is the same, the depth and focus of the medical knowledge is not, since the biomedical engineering focus is oriented to the tools and machines used and this will be reflected in subsequent sections.**

In order to provide a comprehensive base for the methodology, results, conclusions, and future recommendations for the development of software in this field, it is necessary to compile all the scientific concepts and information that support this work. Therefore, this theoretical background section will provide detailed explanations of these concepts by taking a biomedical engineering approach.

### 2.1 Virtual Reality

#### 2.1.1 History

The concept of "extended reality" was developed mainly towards the end of the 20th century, but its roots go back to the late 1800s. In 1838, the physicist Sir Charles Wheatstone developed the stereoscope, better known as a binocular viewer, which allowed the viewer to observe images with "3D effects" on them, making use of visual tricks and a deep knowledge of how our sight and brains work.

This would be the predominant technology until the 1950s, when a machine called the Sensorama was developed. This concept, created by filmmaker Morton Heilig in 1956, was the first VR machine created, with the addition that it was invented to expand the user's experience when watching a movie. Years later, in 1960, the same filmmaker would patent the first virtual reality (VR) headset, a scaled-down version of the Sensorama.

The 1960s is when many of the concepts and ideas that are used today by the VR development community were developed. Many of these ideas can be found in the scientific paper "Ultimate Display" by computer scientist Ivan Sutherland.

With the dawn of modern computing, the development of VR technologies has also begun to accelerate. Between the 1980s and 2000s, even more VR headsets were created, which progressively became more accessible to the general public until they became available to the video game industry. This sudden increment of attention from the research community prompted the creation of the diagram shown in Figure 2.1 by the researchers Paul Milgram and Fumio Kishino in 1994, the result of their paper "A Taxonomy of Mixed Reality Visual Displays," where all the terms included in XR, including VR, are first exposed.

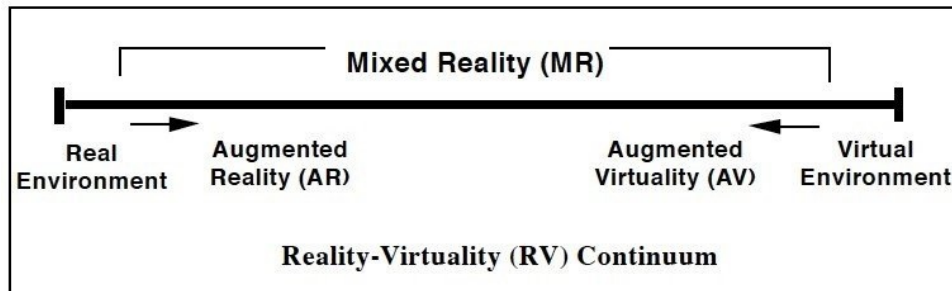


Figure 2.1: Milgram and Kishino's Reality-Virtuality Continuum diagram [54].

These VR headsets available at the time (such as the SEGA VR-1) offered the user the ability to observe and interact with 3D elements for the first time; however, the high cost of this technology ended up stopping its development. This is why the subsequent years between 2000 and 2010 are considered quiet years, since not much happened in the development of these technologies.

Parallel to this lack of development, the computer industry began a race to create ever more powerful microprocessors that are both more efficient in terms of energy consumption and heat dissipation. This is how the relevance of "Moore's law" arises, which states that "the number of transistors that can be packed into a given unit of space will double about every two years" [8]. This indicates that the necessary developments are finally being made to enable the creation of completely virtual environments.



Figure 2.2: A) Sega VR-1, [6] B) Hololens 2 [9] .

And so it is that in the decade of 2010, the most important developments of the last 20 years have been made in relation to both XR technology and the computational capacity necessary to create 3D environments.

Both VR and AR made leaps of quite considerable progress with the introduction of commercial equipment such as the HTC Vive and, in the case of AR, the Microsoft HoloLens. Right now, the focus of the technology is beginning to be on the ability to expand it to potential users and the refinement of the controls, which continues to be one of the biggest challenges so far.

### 2.1.2 Current developments

The general public has become familiar with the terms "virtual reality" and "augmented reality" due to the commercialization of both platforms. It is generally understood that the term "VR" is used to describe completely virtual environments, isolated from the real world, where the user must use special controls (and sensors) to interact with them. This is in contrast to augmented reality, which uses 3D elements and places them in real environments.

This information seems to indicate a clear advantage to the use of AR over VR; however, it leaves aside one of the most important aspects of any XR experience: the processing power. In general, AR devices are less powerful in terms of graphics capabilities than VR equipment, since one of the premises is the ability to be mobile and to show the real world with a virtual overlay. It is for this reason that AR applications can currently be used on devices such as smartphones and tablets, but those devices lack the raw horsepower to provide optimal VR experiences.

Thus, the usual setup for the use of VR equipment requires more space and specialized equipment for an immersive experience. In the image, it can be seen how the user must use the VR headset connected to the computer (recent developments allow the use of wireless transmission, but this adds considerable latency and reduces image fidelity), sensors, and controls.

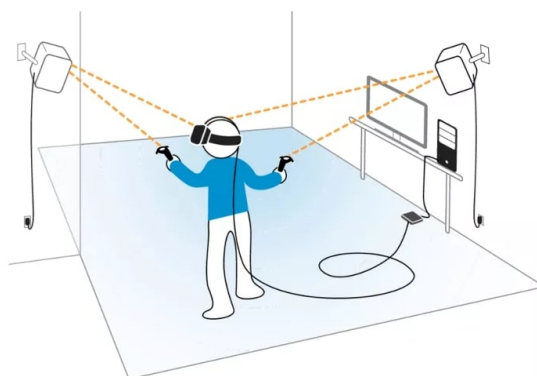


Figure 2.3: HTC Vive VR headset and Lighthouse tracking system setup [10].

It is important to note that, while current solutions provide the user with comfort and ease of use unrivaled by previous equipment, certain technical aspects must still be considered and prior knowledge is required before purchasing these products. The following subsection will characterize some of the aspects to be taken into consideration when evaluating the acquisition (or use) of such equipment.

### 2.1.3 Operating principles

In order to understand the working principles of VR technologies, it is necessary to first understand the general objectives of these technologies, which are to provide the user with an immersive experience. To achieve this, this technology relies on the human senses, through which people can experience their environment.

## Uncanny Valley

One of the most important factors to take into account when considering the immersive user experience is the limits of the simulations. Currently, there is a concept called the "uncanny valley," which represents an obstacle to any kind of simulation of reality. In a study carried out by the University of Waseda in Japan, where human interactions with robots are studied, the following definition has been obtained:

"The uncanny valley refers to the hypothetical changes in subjective robot eeriness with regard to their resemblance to people. At first, as this resemblance increases, the robots appear more familiar. Then, at a certain level of resemblance, the robots appear very eerie. However, when the resemblance becomes sufficiently high, their eeriness declines, and their likeability increases" [22].

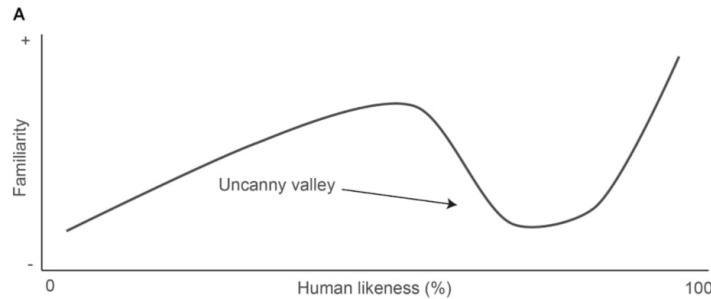


Figure 2.4: Uncanny Valley curve [23].

In terms applicable to VR, the closer a simulation gets to reality, the more the user will begin to notice small details, which causes all sense of realism of the simulation to be lost.

## Head Mounted Displays

The main sense that is stimulated is the sense of sight. The VR headset includes a pair of high-resolution displays (usually one per eye, although single-screen models are available), a wide viewing range, and a high refresh rate. Both of these specifications vary from model to model of a VR headset, however the general consensus is that the refresh rate should be a minimum of 75Hz, as lower than this can induce motion sickness, which causes the user to become disoriented, experience symptoms of nausea, and in some cases even have low blood pressure.

Table 2.1: The 10 most commonly used VR headsets according to VR Mark, application to test Head Mounted Displays (HMD)[16].

Manufacturer & Model	Resolution per eye	Refresh Rate (Hz)	Field of view (Degrees)
ASUS Windows Mixed Reality	1440 × 1440	90	95
Dell Visor	1440 × 1440	90	110
HTC VIVE	1080 × 1200	90	110
HTC VIVE Pro	1440 × 1600	90	110
Oculus Go	1280 × 1440	60	101
Oculus Rift S	1280 × 1440	80	110
PIMAX 8K	3840 × 2160	120	200
Samsung Odyssey WMR	1440 × 1600	90	110
Varjo XR-3	2880 × 2720	90	115
Vrvana Totem	1280 × 1440	75	120



It must be taken into consideration that each user's physiology is different, and these experiences must be adaptable. One of the factors that can be modified in most modern models is the interpupillary distance (IPD), as well as the focusing distance for a sharp image. In order to make these adjustments, HMDs use a set of lenses that are movable and can be adjusted to the user's needs.

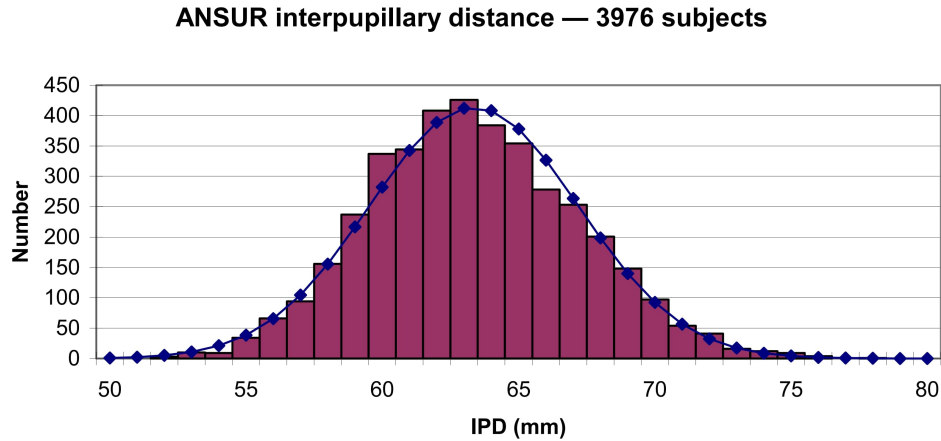


Figure 2.5: Example in the variation in the IPD among test subjects [15].

This only affects the sense of sight, but it would be no different from watching a television screen without the user tracking factor. For this, a combination of sensors is used in both the HMD and its controls. This allows the system to position the user within the virtual environment and make the necessary adjustments depending on position, movement, and interactions being done.

### Tracking systems

Depending on the type of system being used, these sensors can vary in accuracy, complexity, and size. There are two major tracking concepts: inside-out tracking and outside-in tracking. Inside-Out Tracking refers to the use of specific, predetermined positions where the user can move around in a specific way. Outside-In Tracking makes use of external sensors to pinpoint the user's position within a space. In commercial systems, it is common to find inside-out tracking, with outside tracking to a lesser extent. This is because commercial applications and games were created with the user's space limitations taken into account.

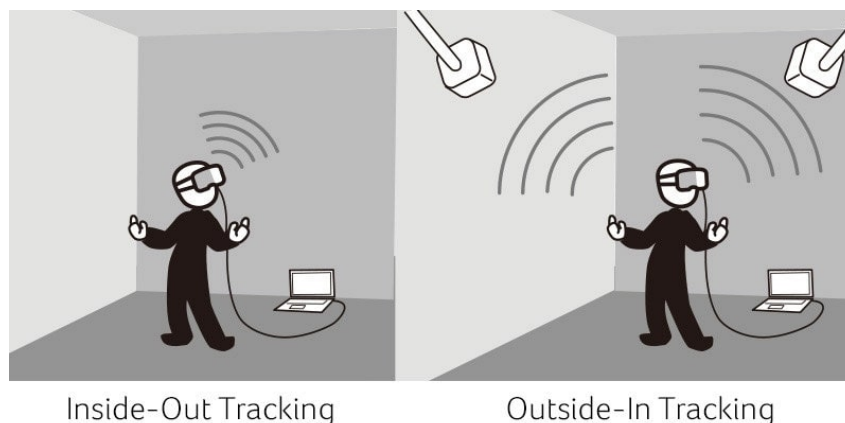


Figure 2.6: Diagram of the functionality of both inside-out and outside-in tracking systems [19].

For more complex applications such as scientific research and corporate use, highly accurate outside-in systems are primarily used in much larger spaces, often dedicated solely to these applications.

It is then necessary to note the different technologies used to perform the precise tracking of the user. In this case, the principle of operation is the use of markers at various points on the HMD as well as on the controls. These markers (usually photo reflectors) allow the sensor system (infrared or light in the non-visible light range) to pinpoint the position of the user, the position of the controls, and the movements made by the user.

There are specific applications that use space mapping systems that use lights outside the visible spectrum for humans. These lights, positioned at multiple points in the space where the simulation is to be performed, allow the system to determine the relative position of the entire user, not just the position of the controls and the HMD. However, these systems require spaces that are completely dedicated to these activities and have the fewest elements that can interfere with the tracking. Elements such as windows, reflective elements, or even equipment that generates electromagnetic waves can affect the signal and generate an incorrect reading, thus creating a faulty simulation. Similar to the tracking methods mentioned above, there are many other methods to track the user, such as:

- Mechanical tracking: is the use of physical parts with coordinates on three axes to allow software to determine their position. This has the advantage that the tracking of the pieces can be done using cables and sensors integrated within the pieces themselves, making them more accurate and therefore a much more immersive simulation.
- Eye tracking: Determines where the user's gaze is directed in the HMD. In most cases, this tracking method works by measuring the distance between the center of the pupil and the edge of the cornea. As the user's eyes vary the point at which they focus, the distance changes due to the variation of the angle. Infrared sensors that capture variations in light reflection and computational modeling are typically used to make these measurements.

This greatly helps the developers and the performance of the HMD, since due to their anatomy, the point of visual focus for people is also the point where the highest possible resolution can be found. This phenomenon is not limited to resolution but also to color and clarity, which cover only 2% of the field of view. This allows developers to render sections that are not in focus at a lower resolution, making more efficient use of the resources available on the computer and allowing the use of visual enhancements where the user is focusing.

- Body tracking: Using reflective elements attached to the user, it can track the user completely; however, this method requires a large number of sensors and the user to wear a special suit.
- Hand tracking: One of the more popular tracking methods at the moment, this allows not only to obtain information about where the user's hands are placed but also to develop the tracking of each finger individually. At present, such equipment already exists, although only for very specialized applications and with a price range far beyond the reach of the average user.

One of the parallel developments that is taking place within hand tracking is the inclusion of tactile feedback. Being able to interact with virtual elements and have the sensation of touch adapt to the simulation is one of the key points for the next generation of VR equipment. Some of these developments go hand in hand with developments in hand tracking technologies. For example, tactile feedback gloves, which allow the user to feel when their hands touch or grasp an object, are currently under development (in both software and hardware).

## 2.2 3D modeling & Developing Software

The development of these virtual reality technologies is intrinsically linked to the development of VR applications. It is therefore essential to understand how these scenarios are created, the complications involved, and the reasons why some software is more specialized than other.

In general, it can be said that the development of these applications does not differ from the development of 3D applications in modern video games, i.e., the developing process is the same; it only differs in the initial considerations (and approach) taken and the final implementation of the project, such as the complexity of the elements and controls to be used.

The development of any modern 3D application requires the incorporation of several platforms and tools, including a 3D modeling tool, a code development platform, and a 3D engine.

### 2.2.1 3D Modelling

3D modeling is the creation of 3D elements from drawings, sketches or equations. This is accomplished using dedicated programs that allow developers and engineers to create mathematical representations of drawings, extrude them, and create either completely solid objects or the so-called "shells" which are hollow elements with only an outer shell.

There are multiple types of software for 3D modeling, depending on the use being given to the models. Among the most relevant are:

- Polygonal modeling: This is the most common system used for video game modeling and digital animation. It uses triangles or 2D polygons to generate any shape. This represents a great advantage when considering rendering performance since modern processors and graphics cards are optimized for handling triangle-based geometric shapes. However, for scientific applications, this type of modeling is not recommended since it is quite imprecise and prone to rendering errors between tests.

However, as mentioned above, this is the preferred method for digital animation since the complexity of triangles in a model can be modified in these programs, thus adapting higher resolutions for more complex projects. Additionally, each of the vertices of the triangle can be considered a joint, so it is possible to animate the objects in a more natural way.

- National B-Spline & Nonuniform Rational B-Spline Modeling: These modeling systems use mathematical representations for the creation of 3D objects. Objects created from these representations are not composed of polygons or triangles but are composed of an infinity of points, which are the equivalent of vectors in images, so the perceived resolution is only dependent on how close the object is observed.

This generates 3D models of very high quality and accuracy; however, on most graphics cards, these are quite taxing elements, so the rendering of them in real time is extremely complex and limited only to scientific computing equipment.

With such reliable representations, it is with these types of models that it is possible to perform stress, thermal, and other simulations for manufacturing and scientific research purposes, so these are the programs of choice for the scientific community and engineering industries.

Considering these models, the applications that apply to them can be categorized depending on their use and the final construction of the objects.

- CAD Modeling: CAD stands for "computer-aided design" and is the natural evolution of the use of drawings in engineering. Using an equivalent system, it allows the creation of both 2D and 3D drawings prior to production.
- Solid Modeling: The use of these models is for the purpose of computational testing to see how they would behave in the real world. These tasks are of great importance for the scientific community since they allow the simulation of phenomena such as the prediction of the behavior of physical elements in collisions, the application of different forces to objects, and even the application of elements of nature such as pressure, temperature, and electricity.

This branch of the use of 3D elements is intrinsically linked to models generated from mathematical principles and equations. To perform these analyses, it is usually necessary to use really powerful computer equipment and even clusters, which are sets of powerful computers that are connected so that they work as one. In fact, the supercomputers that currently exist are used for this purpose, such as the simulation of complex mathematical equations in order to create simulations that allow the advancement of scientific knowledge.

- Wireframe Modeling: This is the derivation of the creation of elements from polygons. In this case, the so-called "shells" are created where only the outer shell of the object is created and no characteristics such as mass, stiffness, or the characteristics associated with the material, such as the behavior of different external forces, are attributed to them natively.

This being the case, these elements are much easier to render. This method of working with 3D elements is preferred by digital artists and animators since it allows a lot of flexibility of use, which the objects used in CAD and solid modeling do not allow.

## 2.2.2 Development Platform

Integrated development environments, or better known now as IDEs, can be defined as "platforms that provide tools that automate common tasks, such as refactoring, auto-complete, and correction of compilation errors" [39]. They provide a wide arrange of tools for developers, which allow them to increase development speed while minimizing errors.

For the development of 3D applications, especially those where the user has the ability to extensively interact with the 3D environment, it is also important to have the ability to see the changes made efficiently and to be able to interpret what the written code means in 3D objects. This is why, in many cases, IDE platforms have integration modules with game engines.

IDEs usually consist of a code editor, a compiler, and a debugger. These act as graphical text interfaces, and depending on the packages that are loaded into them, they can even demarcate the most common functions used in the programming language. This greatly helps developers, as it allows for fewer syntax errors and increases the speed at which code is created.

## 2.2.3 3D Graphics Engine

Currently, any type of 3D application, from games to didactic tools, requires the use of a graphics engine. These are the ones that bring together all the elements of 3D application development. This means that the above-mentioned compatibility modules will be connected depending on the game engine selected for the application's development. With 3D editing capabilities, programming, and a robust graphical interface, these are the most important pieces of software when planning an application, since they vary from what programming language to use to what types of 3D elements are supported.

Currently, there are many game engines with different uses; however, the two most used for their versatility and excellent results are Unity and Unreal Engine. Between both platforms, they have more than 60% of the current market for all 3D applications. Comparatively speaking, they are very similar; however, they present important differences in the desired final result, the platform of use, and the flexibility and compatibility with other programs.

Table 2.2: Comparison between Unity & Unreal Engine [33].

	Unreal Engine	Unity
Programming language	C++	C# and Java Script
Asset Store	~16000	~65000
Cross Platforms	Windows, Mac Os, Linux, iOS, Android, SteamOS, Xbox and Playstation	Windows, Mac Os, Linux, iOS, Android, SteamOS, Xbox and Playstation, VR (including Hololens), WebGL, Android TV and Tizen
Graphics	More realism	More artistic design
Market Share	17%	50%
Price	Free, but paid version includes all features	Free until the project revenues cross certain threshold

Based on the information presented in the table 2.2, it can be inferred that the most appropriate choice for the creation of 3D applications for virtual reality is Unity. It offers a wide range of platforms with which it is natively compatible and one of the largest developer communities, so the forums where solutions to problems are found are one of the largest available. This kind of software and other relevant elements will be discussed with more detail in the methodology section.

One of the most important elements is the use of game engine asset stores. It is here that the developer community offers for sale (or free access) 3D elements and pre-made codes that can be used in other projects, thus increasing even more the production speed of such applications. Nowadays, thanks to the 3D printing boom, there are many more freely available libraries as well as paid elements, with a great variety of 3D models. As part of Unity's inter-compatibility, it accepts a large number of formats, which can be incorporated into the program natively or using plugins created by Unity or by the developer community.

#### 2.2.4 Unity

Being the software of choice for creating VR applications, Unity has several features that allow a smooth workflow between coding, element creation, and rendering. The programming language used is C#, which is an object-oriented programming language, and the main features it uses are variables, functions, and classes. However, it only works as an interpreter for that language. Unity works in conjunction with Microsoft's Visual Studio program, which is an IDE with native compatibility with Unity.

Unity's graphical interface uses a hierarchy system in which parent and child objects inherit the same features as the parent objects while retaining individual features. It also uses a system of folders for the organization of textures, materials, scripts, and scenes. Each scene individually represents a space with its own rules; these can be interconnected, but they are handled individually.

Within this graphical interface, one of the most important elements is called the game window. In this window, the final product is rendered at a low level, with all program interactions and all objects present in the scene. Once this window is activated, the elements that were modified will remain unchanged once the simulation is complete.

With all the features mentioned above, one of the most important is Unity's ability to accept modules for different platforms and the ability to migrate the created simulations or export them to different platforms. This provides great flexibility for developers who need to export their projects to multiple platforms at the same time.

### 2.2.5 Blender & Solidworks

In the case of 3D modeling software, one of the most important considerations to take into account is the level of precision with which the user wants to work. Creating an endoscope for a surgery simulation is not the same as designing a video game character. In this case, the versatility and interconnectivity of the applications allow the user to rescue the best of each of them. This is how the combination of 3D software allows for both accurate models and high-quality graphics.

To obtain this combination, the previously mentioned nonuniform rational B-spline modeling and polygonal modeling methods can be combined using a common format. The STL format (which stands for "standard triangle language"), created in the 1980s for fast 3D modeling, is an open-source format that allows the user to recalculate the shapes of a 3D model into triangles and polygons, allowing faster processing and compatibility between different working sets.

However, one of the disadvantages of this process is that by moving to a format like STL, the nonuniform rational B-spline model loses the features that allow its use for physical and other scientific simulations. Even so, software such as Blender allows the user to reincorporate certain features by using scripts, which can be used in software such as Unity to simulate the physical characteristics of the object, but with the restrictions of the simulation itself.

This is how the work dynamics between the two software programs, Blender and Solidworks, become possible. Solidworks is a software oriented toward the creation of objects using nonuniform rational B-spline modeling, which allows it to create objects of great precision and is oriented to engineering manufacturing so they acquire the physical characteristics of the assigned materials.

Blender, in contrast, is a program that allows the creation of objects with polygonal models. It allows not only to create geometries based on polygons but also to edit them, simplifying the geometry (reducing the number of triangles), which is of vital importance for VR applications for the performance purposes mentioned above.

Each of these generates proprietary files that work together with the different versions of the respective programs, but they also have options to work together with other programs by being able to export to a wide range of file formats. This will be discussed in more detail in the methodology section, but the general consensus is the use of FBX, STL, and OBJ formats, which are the most universally accepted into these kinds of programs.

## 2.3 Minimal Invasive Surgery Principles

### 2.3.1 General definition

In order to understand the implications of this project in the area of biomedical engineering, it must first be analyzed in terms of the theoretical roots and foundations of anatomy and physiology. In the first instance, the procedure that will be the focus of this research is "laparoscopic cholecystectomy", which falls into the category of minimal invasive surgery procedures.

Minimal invasive surgery (MIS) are generally understood as "surgeries performed remotely with refined and long instruments, using trocars, under the use of a video system"[20].

Among the advantages it has over regular surgeries are:

- No direct manipulation.
- There is no significant external trauma.
- There is less chance of developing a scar hernia.
- Less recovery time.
- There is less internal mistreatment of the patient since the internal organs are not retained with hooks.
- Shorter hospital stays greatly reduce the risk of thrombosis, embolism, pneumonia, and lung collapse.

However, this also represents a new challenge for professionals in the area, since this type of surgery severely limits the range of action and flexibility that physicians usually have with open surgeries. It is for this reason that training, familiarization with the equipment, and a thorough knowledge of human anatomy are necessary to be able to successfully perform this type of surgery.

### 2.3.2 Tools

To be able to perform this type of procedure, it is necessary to take into consideration the instruments that make it possible. The first tools that make this surgery possible are called trocars, which are the instruments that allow access to the patient's body when there is no access from any of the natural cavities of the human body.

#### **Trocars & Gas insufflation**

These are sharp instruments that allow penetration to the point where the surgery is meant to be performed. They offer more safety advantages over other alternatives, such as the Veress needle. The Veress needles are very sharp instruments with a protective sheath that snaps into place once a space is reached without resistance, that is, after penetrating the skin. However, this mechanism can fail and cause some internal damage or injury to the patient, so it is not the safest option nor the preferred option for most surgeons when it comes to surgery in the abdominal cavity.

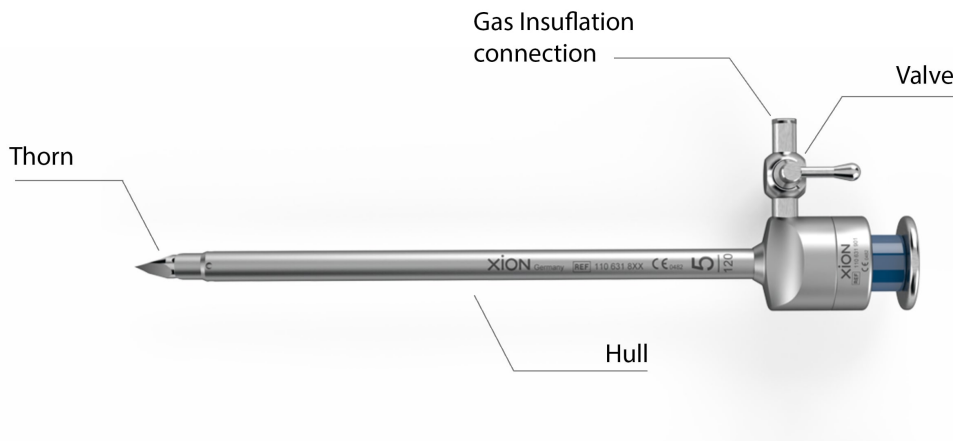


Figure 2.7: Example of a trocar and it's parts [14].

An alternative to the use of both the Veress needle and the trocar is the Hasson cannula. This is a variation of the access instrument, but with a blunt tip, so incisions must be made manually and then entered with the instrument. These have the advantage of being reusable, unlike most Veress needles or some trocars, as it is the case of the plastic ones.

Depending on the type of surgery being performed, gas is used to create space between the skin and the patient's organs. This allows for not only a greater range of motion for the instruments used but also better vision for the endoscope and thus the surgeon. Among the possible gas options that could be used, the most commonly used is  $CO_2$ .

Although this gas is toxic, it is the most commonly used because the human body is able to process the amount it absorbs during surgery through the breathing process (which is monitored and to some extent controlled during surgery by the anesthesiologist). The other gas options are listed below:

- Oxygen: It cannot be used for this type of surgery. Electrical equipment is used to cauterize when performing cuts during the surgery, and oxygen is flammable.
- Natural gas: As long as it contains oxygen, it is flammable.
- Inert gas: The cost of using this gas is very high.
- Nitrogen gas: This gas has the disadvantage that, when it passes into the bloodstream, it significantly increases the occurrence of embolisms.

In addition to this, certain care must be taken when insulating patients with gas for this type of surgery to take into consideration the internal pressure that the human body can withstand before starting to cause damage. In this case, the internal pressure should be kept between 8 and 12 mm Hg, since a higher pressure can start to cause damage to the internal organs as well as increase the amount of  $CO_2$  that is absorbed, causing respiratory difficulties.

This is counterproductive, for example, in cholecystectomy, since the increased internal pressure in the abdominal cavity also increases the pressure in the lungs, resulting in decreased respiratory capacity. There are also alternatives, such as the use of lifters; however, these can complicate the procedure and are not commonly used.



## Endoscope

The most important instrument in this type of surgery, without which it would not be possible, is the endoscope. These consist of long tubes (whose length varies by model and use) that at the end include a light and a video transmission system, essentially allowing surgeons to see inside the patient's body without the need to perform open surgery. An example of this tool can be seen in Image 2.8.



Figure 2.8: Integrated Endoscopy latest Endoscope, with 4k wireless video transmission [13].

There are several types of endoscopes, as well as considerations to take into account when selecting the right instrument.

- Flexibility.
- Lens System (there is no camera at the end).
- Chip on the tip.
- Stereoscopic view.
- Incorporate a magnifying glass
- Various degrees of inclination (0, 30, 70, and 120).
- Small diameter (5mm to 10mm).
- Different video resolutions (720p to 2160p).
- Can be sterilized.
- Can be preheated.

Among the points mentioned above, it is worth mentioning the preheating process: the tip of the endoscope should be preheated before use; otherwise, due to the difference in temperature between the environment and the surface of the camera, it will become cloudy and will hinder or completely block the vision inside the patient.

Modern endoscopy systems include a camera at the distal end instead of the previously used system of a set of lenses. This camera, however, does not have an image processing system or processor, as this would substantially increase the size, weight, and difficulty of use.

To solve this, the endoscopes are equipped with a separate unit, which is connected via a proprietary connection and takes care of the image processing and video transmission itself. This external unit also provides elements such as guide lines, the ability to record video and capture images for patient records, and even, in some cases, certain controls to modify the viewing range of the camera.

Currently, there are developments in both image clarity and the integration of low-latency wireless video transmission technologies. However, due to the apparent latency between the video feed and reality, these are not yet widely accepted.

The light source used for the endoscope also represents an important factor in the use of this tool. Most commonly included within the external unit itself, the light source must meet certain requirements as listed below:

- Must be a cold light source, i.e., with a light temperature between 5500 and 6000 K.
- It has to use fiberglass.
- It must have at least 150 watts of power.
- Use Xenon or LED technology.
- Have the ability to be used in flash mode for image capture.

### Grasp & Dissection

In addition to laparoscopic endoscopes, it is also necessary for the physician to be able to move, grasp, cut, and even cauterize the patient's body, which is a key component of any surgical procedure. There are multiple options for this type of tool, but the most common are:

**Forceps:** With a shaft length of at least 30 cm, these tools have one or two movable forceps, which give the surgeon the option to move or grasp inside the patient. Additionally, there are variations of these that allow suturing, either by incorporating special heads, stapling, or other forms of suturing.

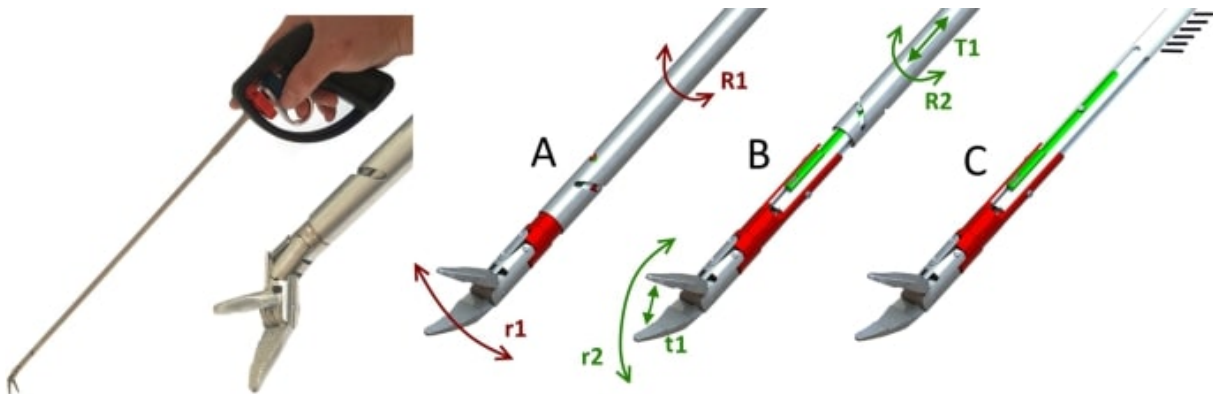


Figure 2.9: Detail view of a grasping instrument developed with a new mechanism that allows 2 additional DoFs. [17].

**Aspiration and rinsing tools:** Due to the nature of the procedures, it is a necessity to have equipment to rinse with water and to be able to aspirate the water used to clean the operating area or to remove blood or other liquids.

Among the elements that are expected from the aspiration and rinsing tools are:

- A grasp with two ventrils and a shaft.
- Automatic control of rinsing pressure and suction is necessary to cause the collapse of the abdominal walls and other areas of the body.
- Heater for the rinsing fluid, so as not to decrease the internal temperature of the patient.
- Suction and rising with their own separate channels.

**Scissors:** Same principle of operation and shape as the forceps, but incorporating a sharp head for cutting.

**Ultrasound cutter:** Also called the "Harmonic Scalpel", this tool is extremely versatile when it comes to this type of surgery. With it, the surgeon can both dissect the tissue and stop the bleeding. The way it works is by using a piezoelectric crystal, into which a current is introduced and which begins to vibrate. The vibrations can reach up to 55500 Hz, with an amplitude varying from 25 to 100 $\mu$ m at different levels.

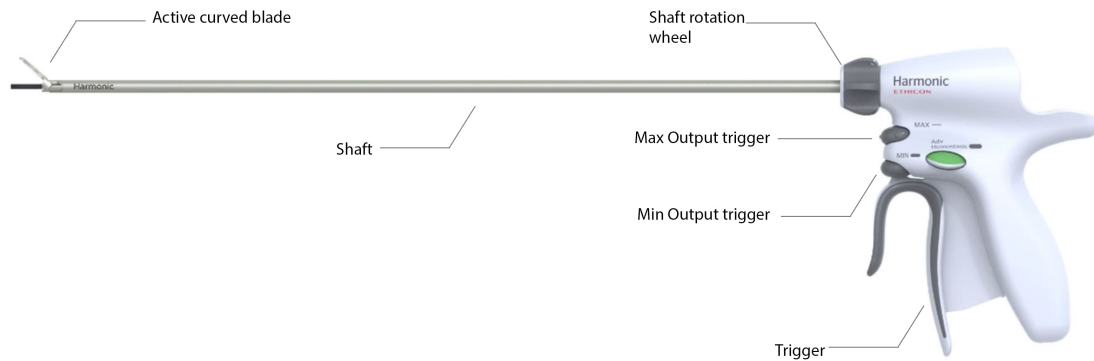


Figure 2.10: An example of a harmonic scalpel. [18].

This allows the following effects:

- **Cavitation:** the rapid evaporation of water caused by a sudden change in pressure.
- **Coaptation:** the fragmentation of proteins that results in the union of two tissues.
- **Coagulation:** The disintegration of the proteins causes the cessation of bleeding to accelerate.
- **Cutting:** Depending on the desired effect, either better clotting with a low cutting speed (low amplitude) or poor clotting with a high cutting speed (high amplitude) can be obtained.

This tool in itself represents a great advantage over other methods used for curdling, since most of them use an electric arc to provoke this effect, but this is contraindicated when using gaseous  $CO_2$ , since it is highly flammable. Additionally, other advantages of this method include:

- It does not use any electric current to perform the cauterization.
- It does not generate burns.
- Minimal lateral damage.
- Low distant tissue damage.
- No charring.
- There is no smoke to obstruct vision.

**Laser:** Making use of a high-power laser, this tool finds its use when the physician want to cut and cauterize at the same time.

**Argon Beamer:** This is a combination of a high-frequency cutter and argon gas, creating an electric arc. It allows for dry coagulation as well as greater area coverage. However, it must be used with great care, as it is not suitable for all types of tissue.

## 2.4 The Gallbladder & Cholecystectomy

Laparoscopic cholecystectomy was first performed by the German physician Carl Johann Langenbuch in 1882 in Berlin. However, it was not widely practiced until the 1980s, when the practice was refined and the tools were adequate. Today, laparoscopic cholecystectomy is the gold standard by which cholelithiasis is treated. In order to fully understand the implications of this surgery, a review of the human anatomy, the physiology of cholelithiasis, and the procedures to follow for a successful cholecystectomy will be done in this section.

### 2.4.1 General Anatomy

First and foremost, the location of this system must be determined. The abdominal cavity is the area where most of the digestive system, urinary system, and hepatic system are located. The precise location of this cavity is "from the osteocartilaginous thoracic cavity to the 4th intercostal space, thus forming the upper and largest part of the abdominopelvic cavity" [24].

This cavity is divided into nine zones for study:

- E: Epigastrium
- FD: Right flank
- LF: Left flank
- HD: Right hypochondrium
- HI: Left hypochondrium
- ID: Right Inguinal
- II: Left Inguinal
- P: Pubic
- U: Umbilical

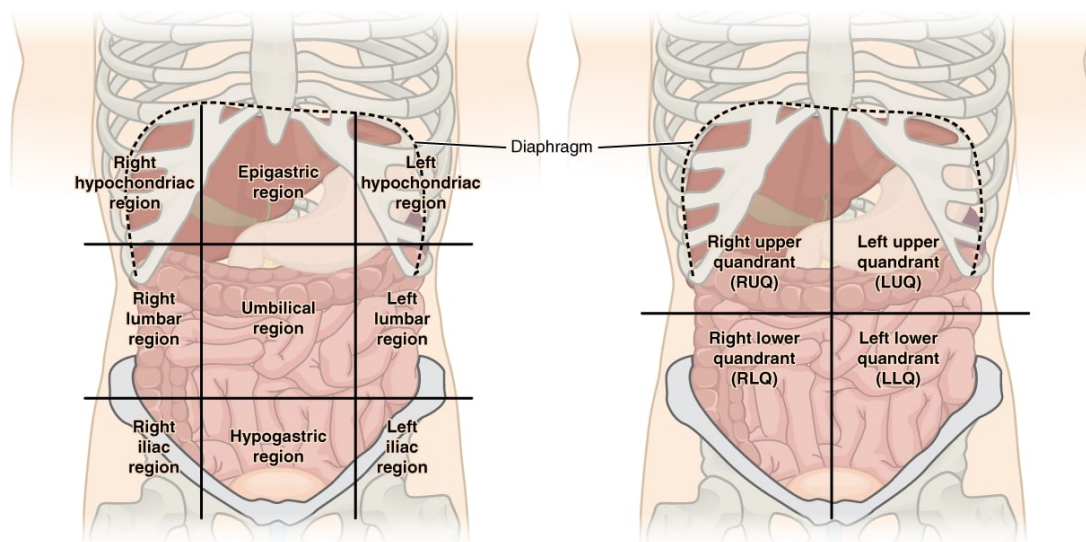


Figure 2.11: Divisions of the abdominal cavity [25].

It can also be divided into four quadrants:

- Right Upper Quadrant: right part of the liver, gallbladder, and right kidney; small portion of the stomach; part of the small bowel.
- Left Upper Quadrant: left part of the liver; most of the stomach; the pancreas; the left kidney; the spleen; and the small bowel.

- Right Lower Quadrant: cecum, appendix, right female reproductive organs, and right ureter.
- Left Lower Quadrant: small bowel, part of the large bowel, left part of the female reproductive organs, and left ureter.

The area of greatest interest for this type of surgery is the right upper quadrant, also called the right hypochondrium. This is where the gallbladder is located, and it is the organ of interest in this type of procedure.

## 2.4.2 Bile Duct & Gallbladder

The most important anatomic elements for the cholecystectomy procedure are undoubtedly those of the bile duct and subsequent gallbladder. The biliary tract is a series of ducts that connect the liver to the gallbladder and, consequently, to the small intestine. These ducts have a location and follow a course similar to that of the portal vein.

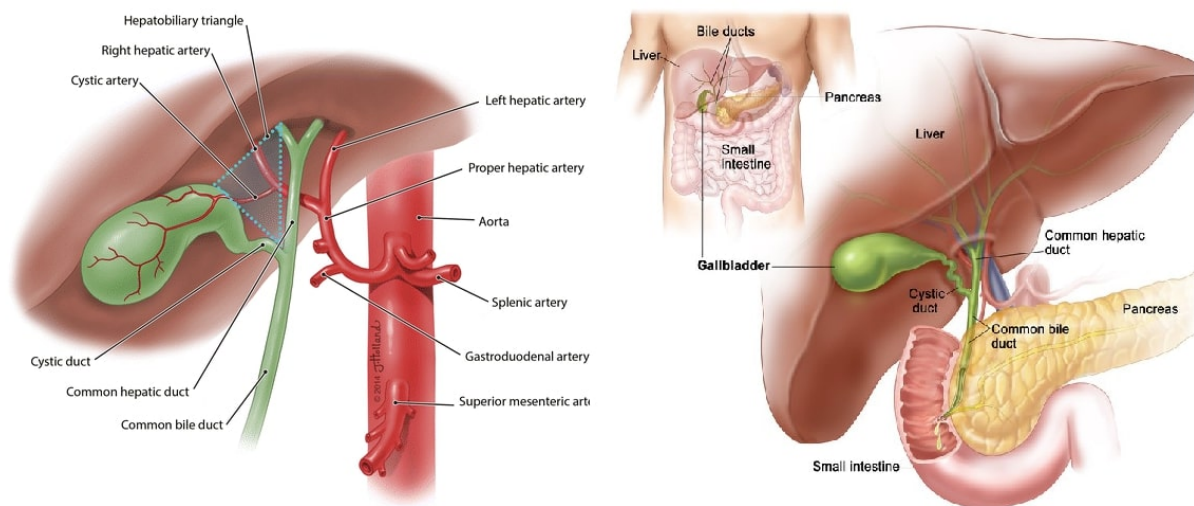


Figure 2.12: Blood artery connections to the gallbladder [31] & Anatomy of the gallbladder [27].

The definition of the portal vein, according to the U.S. National Cancer Institute, is "a blood vessel that carries blood from the intestines, spleen, pancreas, and gallbladder to the liver" [47]. The cystic artery is the main blood duct responsible for irrigating the gallbladder. The location of this artery can be said with almost total certainty that it is located in the cystohepatic triangle (since the anatomy may vary from patient to patient). Additionally, venous drainage is carried out by a series of small veins called cystic veins.

These are then joined to the portal vein, and this is how the blood drainage of the entire cystohepatic triangle is performed. Finally, there is the lymphatic drainage, which is carried out in the same way by the hepatic lymphatic nodules.

The bile duct, however, is not a single duct but the conformation of several of these, particularly along the route between the liver and gallbladder. This is formed by the posterior and anterior hepatic ducts. As shown in the image 2.13, each of these ducts has branches that connect to different parts of the liver.

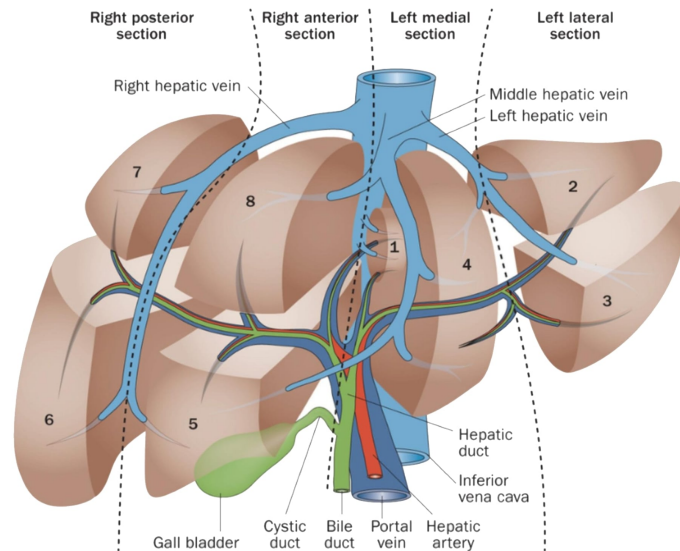


Figure 2.13: Bile Duct connections [28].

These ducts transport bile to the gallbladder. This organ can be defined as "pear-shaped with a length of approximately 7 to 10 cm and an average capacity of 30 to 50 ml"[24]. The main function of the gallbladder is the accumulation of bile produced by the liver. This liquid, of a dark green color, helps with the digestion process, so once it is accumulated in the gallbladder, it is then poured into the digestive tract.

The anatomy of this organ can be considered in two parts: the organ itself and its connections. The gallbladder itself can be divided into 4 sections: body, fundus, neck, and head. The connective tissue is the cystic duct that joins the gallbladder with the biliary tract. The previous sections can be seen more clearly in the image 2.12. It should be noted that bile's function extends beyond assisting the digestive system to excrete excess cholesterol but also other wastes from the hemoglobin metabolic process.

The gallbladder is located in the so-called "gallbladder fossa" which is located on the visceral side of the liver and anterior to the duodenum. It is in this last one where the common bile duct, or choledochus, enters, making use of a section called the sphincter of Oddi.

### 2.4.3 Pathology

The two main diseases that can affect the gallbladder are cholecystitis and cholelithiasis. There are also other types of affections, such as carcinomas and polyps, which can also be treated with laparoscopic surgery methods.

#### Cholecystitis

Cholecystitis is a form of inflammation of the gallbladder that can occur suddenly. While there are several causes for this, the most common is biliary tract blockage. One of the reasons that this blockage may occur is due to the appearance of gallstones, also called cholelithiasis. Currently, "In developed countries, about 10% of adults and 20% of individuals over 65 years of age have gallstones" [35].

The pathophysiology of cholelithiasis indicates the presence of bile grit, which is formed by calcium bilirubinate and cholesterol crystallization, among other factors. The presence of this grit in itself does not represent a problem, as long as the bile is regularly and completely

emptied from the gallbladder. In addition, there is evidence that factors such as diet and physical condition (how physically active a person is) can influence cholesterol and bilirubin levels, which would directly affect the change in bile. The size of these gallstones (unlike kidney stones) can be considerable, similar to the size of a golf ball.

These stones can block the cystic duct or the neck of the gallbladder, resulting in inflammation and acute pain for the patient. Management can be done using pharmacological treatments such as antibiotics and analgesics; however, these are only effective in cases where the stones are of a considerably small size and where there is insufficient reason to perform a surgical procedure. In addition, there may also be the development of cholangitis, a gallbladder infection usually linked to the appearance of gallstones.

### **Symptoms of cholelithiasis**

Although 80% of adults diagnosed with cholelithiasis have no symptoms at all, this is because the size of the stones allows them to pass through the cystic duct without any problems. However, when the size is larger or the stone simply gets stuck in the cystic duct and blocks it, this may generate biliary colic, which can lead to cholecystitis or cholangitis, which can be life-threatening.

"Biliary colic typically begins in the right upper quadrant of the abdomen, but can be located anywhere in the abdomen." It can be difficult to define, especially in diabetic patients and the elderly. The pain may radiate to the back or arm" [35].

These cramps do not usually remain constant but are cyclic, varying between 30 minutes and 1 hour, depending on the condition of the patient. This is characterized by intense pain radiating throughout the abdominal cavity, which can lead to frequent nausea and vomiting. One of the most important indicators when confirming whether there is infection or cholecystitis is the presence of fever in patients. This can also be an indicator of pancreatitis.

### **Diagnosis of cholelithiasis**

For the diagnosis of cholelithiasis, the most common method is abdominal ultrasound. This is not only able to show the presence of gallstones, but can also detect the presence of bile grit.

#### **2.4.4 Surgical Procedure**

For the cholecystectomy procedure, which is the removal of the gallbladder due to the aforementioned complications, there are well-defined steps that should be followed to successfully and safely perform this surgery [24]. After the application of anesthesia, the patient is placed in the supine position with the left arm tucked to facilitate intraoperative cholangiography. Intraoperative cholangiography is a procedure in which contrast dye is injected into the cystic duct in order to perform an x-ray reading.

##### **Step 1**

With the surgeon positioned on the left side of the patient and the assistant positioned on the right side, there's ample access to the abdominal cavity, and the gas insufflation can be started to generate the pneumoperitoneum. The positions for the trocars must be located; these are numbered from T1 to T4. The diameter of the instruments to be used depends on the patient's physiognomy; however, the average is the use of 5mm trocars in T2, T3, and T4. For the T1 location, a 10mm access is usually used.

For the T1 position, an incision of the 7 layers of the abdominal cavity should be performed; here, it is possible to use either the trocars, Veress needle, or Hasson cannula. In special cases where the patient has had previous preumbilical incisions, an incision should be made in an alternative area.

In the case of patients with overpressure, it is necessary to use another access point, which is located in the midline 15 cm below the xiphoid process and still allows the light source and camera to be positioned correctly. The positions of the other access points and their uses are shown below:

- T2: Right Lateral Subcostal Position: elevating the gallbladder with a grasper
- T3: Right Lateral Subcostal Mid-Clavicular Position/Surgeon's Working Trocar
- T4: Edge of the Falciform Ligament/Surgeon's Working Trocar

It is important to notice that the gas insufflation needs to be done in another location, called the Palmer's point, located "...in the area in the left upper quadrant, 3 cm below the costal margin, and in the midclavicular line" [37]. The suction of both gas and fluid can be done using one of the T-labeled trocars.

## Step 2

Usually the gallbladder is located behind the left lower section of the liver, so it is necessary to perform the approach through the T3. Using this access, the surgeon must manipulate the neck of the gallbladder, as long as it is not distended or inflamed, in which case "...is recommend decompressing it with a needle aspiration device to avoid perforation with spillage of bile and gallstones" [34]. In cases where the gallbladder is attached to the liver or there are other types of adhesion, it is necessary to use one of the above-mentioned instruments, such as the ultrasonic cutter or the mono-polar electric dissection tool, to separate it before proceeding with the removal of the gallbladder.

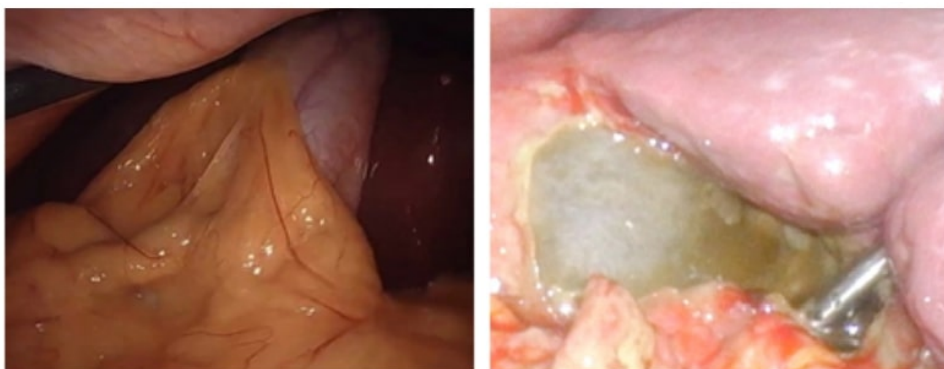


Figure 2.14: Trocar positions in the abdominal cavity in a patient with regular physiology [34].

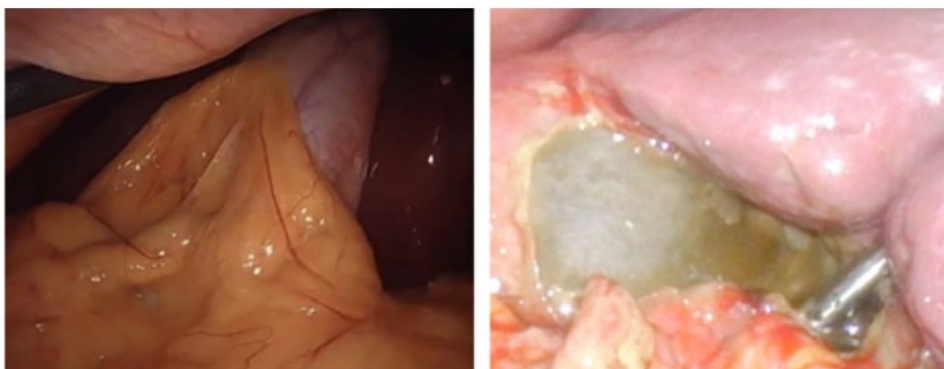


Figure 2.15: Gallbladder attached to the lower wall of the liver [38].



These energetic methods should be used in conjunction with rudimentary forceps, as overuse of energized instruments may cause more damage than necessary to the patient.

### Step 3

Once the characteristics of the gallbladder have been visually identified, it is necessary to make the video adjustments to have an optimal view at the time of cutting, maximizing patient safety. The main parameters by which to be guided are:

1. That the hepatocystic triangle (also called Calot's triangle), which must be free of all fat and fibrous tissue, is seen. In this case, it is necessary that the hepatic duct and the bile duct can be observed, as well as to keep them in position.
2. The lower third of the gallbladder should be sufficiently separated from the liver, revealing the gallbladder fossa.
3. Only the cystic duct and cystic artery should be observed as structures entering the gallbladder. It is critical to correctly identify each one, and if there is tissue connecting these two structures, separate them using energetic instruments and cauterize at the same time.

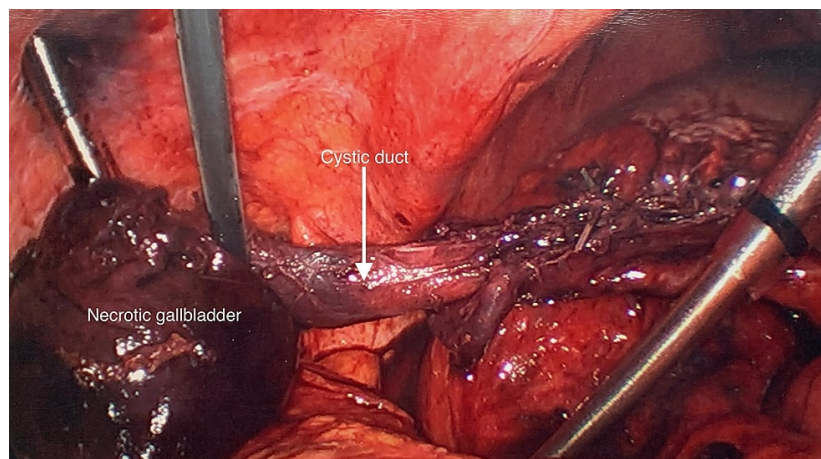


Figure 2.16: Example of the correct position for performing the division of the cystic duct in a necrotic gallbladder [36].

### Step 4

The next step of the procedure is to use clips on the cystic artery and proceed to make the cut and separation. For this, two clips (usually 8 mm) are used, one on the proximal side and the other on the distal side, taking the cutting point as the dividing line. These must be well positioned since, due to the internal pressure at the time of cutting, the clips can become displaced and come loose.

For the next step, it is first necessary to become familiar with the concept of cholangiography, which is the study of the bile ducts and gallbladder using x-rays and the injection of a contrast medium. This type of operation is considered a routine procedure that is used when it is considered that there are CBD stones, abnormal anatomy, a dilated bile duct, or a previously performed gastric bypass.

When cholangiography is not used, at least 3 clips should be placed: 2 on the distal side and one on the proximal side. Individual clips can be used on each side of the incision site as long

as the patient's specific physiology allows; however, for safety and patient protection, two clips are usually used on the distal side. Once the clips are secured in the correct position, the cut can be made with scissors.

### Step 5

Once the division of the cystic duct is done, the gallbladder must be separated from the walls of the liver. In this case, energetic methods are used to perform this separation since the use of simple instruments can generate more trauma than is necessary for the patient. Special care must be taken not to damage the liver, and suction and aspiration systems must be used to clear the liver and remove any excess blood or bile that may spill out.

In the case of acute cholecystitis, complications such as additional bleeding may occur, so it is necessary to use a tool such as the harmonic scalpel, which allows varying the speed of coagulation according to the needs and does not produce smoke. Once the gallbladder is completely separated from the biliary fossa, the gallbladder can be placed in a bag that is introduced through T1 (10 to 12 mm in diameter).

### Step 6

Once it is removed from the patient's body, the clips must be checked, the trocars and other instruments removed, and sutures applied. Before suturing, excess  $CO_2$  should be removed from inside the body. This needs to be done in coordination with the anesthesiologist.

## 2.5 Pedagogy and education

### 2.5.1 History

The educational process is the transfer of knowledge from an expert in the field to the student in order to not only replicate the knowledge already acquired but also generate new ideas that advance the field taught.

Current teaching models are a combination of several previously implemented models. The two most ancient models are the ancient Oriental system and the ancient Greek system. Before reaching the era of pedagogical development in Greece, the most ancient societies implemented a system of transmission of practical skills, survival knowledge, and traditions usually linked to religious dogmas. This is how the concept of formal education was developed in Greece, where schools were created where subjects such as mathematics, literature, and philosophy were taught. The Greek philosopher Plato believed that education should be a means to develop a person's character and moral values, as well as their intellectual capacities. Education was seen as a means to create a better society and to develop individuals who could contribute to it.

After a dark obscurantism period, where most of the educational institutions were controlled by religious authorities, the development of education was followed by the humanistic model taught in the Renaissance. This is relevant to current systems because this system seeks to generate knowledge in various disciplines in both theory and practice, generating both knowledge and experience.

Consequently, there was the development of the learning classroom and the rigorous scoring and testing schemes that were developed during the industrial revolution, where the main objective of the educational centers was to provide basic education as quickly as possible to generate more workers.

This leads to what is now known as the "traditional teaching scheme," where participants receive a master class supported by visual elements provided by an expert in the subject. These systems can be supported by the use of specialized laboratories or by occasional visits to workplaces where the concepts learned are put to the test.

### 2.5.2 Education today

In the field of education, it is necessary to emphasize how traditional education has been modified in recent years due to technological development, the reach of technology such as high-speed Internet to more and more populations, and the unprecedented conditions provided by the COVID-19 pandemic.

Before the global pandemic, this was the accepted educational system, in which a master class is presented with a teacher explaining the topic and, as a result, evidence of the knowledge imparted. According to Piaget's theory (1969), in order to manifest intelligence and learning progress, the student is the main person in charge of this process, with the teacher merely serving as a guide or facilitator. This is where the established pattern of traditional education starts to show some flaws.

Experts in pedagogy have begun to question this approach to education, wondering whether it is best to focus on the acquisition of knowledge in a world where students are also expected to have experience before they leave higher education. Thus, countries such as Finland have implemented novel systems where half of the knowledge is acquired through the implementation of the concepts learned. This has an impact not only on professionals trained in higher education institutions but also on the training of the youngest, both academically, psychologically, and socially.

### 2.5.3 Evolution of education in the medical field

Medical education in general was modified in 1910 with the Flexner Report. "This transformation occurred in the aftermath of the report, which embraced scientific knowledge and its advancement as the defining ethos of a modern physician. Such an orientation had its origins in the enchantment with German medical education that was spurred by the exposure of American educators and physicians at the turn of the century to the university medical schools of Europe" [42].



Figure 2.17: Implementation of VR for medical education purposes [52].

It is considered by experts in the field that the next great revolution related to the educational system is on the horizon and that the digitalization of education is what will bring about this pedagogical revolution. Taking into consideration the recent advances in digitalization and computational power, new educational systems have begun to be implemented where practical lessons have been gradually implemented using virtual and augmented reality technologies. However, this does not replace practical sessions with patient training but instead provides a new tool that allows students to acquire the necessary skills to perform procedures safely.

"Simulation bases the pedagogical conception of teaching on the student and not on the teacher and also develops the student's capacity for discovery, self-learning, and self-instruction or self-correction of their own learning objectives and does not have to submit to the traditional pedagogical method, obsolete and paternalistic, based on the master class" [41].

## 2.6 Closing remarks

In this chapter, it was sought to inform the reader about the basic concepts of XR technology, the different options that currently allow the development of applications for it, and the capabilities of these systems to be useful in different fields. In the same way, a brief introduction to the world of medicine is made, specifically in the area of minimally invasive surgery. The reader is directed here to focus on cholecystectomy and the underlying causes that lead to it.

There is still much to be developed in order to bridge the gap caused by the uncanny valley, but the decade from 2010 to 2020 has seen more development in the area of XR, owing to the parallel development of microprocessors and the growing interest in these technologies by the general public as well as by the scientific and pedagogical communities, which are beginning to consider it a very useful tool to demonstrate theoretical concepts when there is otherwise no safe alternative to allow students to experience them.

In the following sections, the concepts mentioned here for the development of the simulation, the subsequent analysis, and the conclusions themselves will be used as stepping stones for the work done.

# Chapter 3

## Related Work

The field of virtual reality for didactic purposes is something that has been worked on for the past decade, but nonetheless, it is still at the forefront of educational methods. Many specialized university programs have now adopted virtual reality as a serious didactic tool, particularly in light of the global health crisis of COVID-19, which forced many higher education institutions to the digitalization of teaching.

Due to these situations and the advancement of technology, both in the raw power of microprocessors and in the area of optics, there is a boom in research into the possibilities offered by these technologies. The three main focuses of this kind of research can be divided into three categories:

- Education.
- Video game development.
- Professional applications in fields such as medicine and engineering.

Taking into account the above-mentioned approaches, there are a large number of studies that do overlap with the concepts and developments of other areas for their own benefit. Thus, there are studies such as those carried out by Dr. Simon Adler for his doctoral thesis.

With the title "Entwicklung von Verfahren zur interaktiven Simulation minimal-invasiver Operationsmethoden" in German, Mr. Adler demonstrates a finite element study in order to provide an accurate simulation for minimally invasive surgeries. In this paper, it can be seen how finite element analysis and the equations derived from it can be used to simulate real slices and interactions, especially with blood vessels.

There are also studies that demonstrate the effectiveness of these technologies for information retention and student learning. In this case, it must be remembered that technology has advanced by leaps and bounds over what was possible a few years ago, so the more recent the study, the more relevant it is to the current project.

All of these related works will be discussed in this section, along with the conclusions that came along with them and their relevance to the development of the current project.

## 3.1 State of Art: VR in biomedical programs

### 3.1.1 Using Virtual Reality in Biomedical Engineering Education (Singh, 2020)

This study, conducted by joint researchers from different universities in Philadelphia, USA (2020), explores the implications of using VR for the teaching methods of biomedical engineering students. The most important premise of the study is the fact that biomedical engineering students, unlike their medical counterparts, possess very little actual clinical knowledge or experience. It's important to note that most biomedical engineering programs focus more on 2 areas: hospital-based or medical equipment design. In both cases, most programs offer the traditional teaching approach with the occasional visit to a procedure or clinic. This raises the question of what the results would be in terms of information retention and student feedback if other teaching methods, such as the implementation of VR technologies, were implemented.

However, the study does not use VR in the sense explored in the previous sections, but instead uses the word as it is known to the general public, namely, that any experience where a VR headset is involved is a VR experience. In this case, the study uses as a baseline the presentation of 2D videos and videos viewed through a VR headset with the ability to zoom in on the videos. In this case, procedures being taken with patients, such as the use of peristaltic pumps and dialysis, were presented. Even so, the results of the study demonstrate greater student engagement and retention of the information provided, with most students stating that this would be their preferred method of learning over the use of 2D video for online classes.

In its final remarks, the authors state that "this study confirms that the investment of time and money to create VR videos/experiences and acquire VR resources was worthwhile in creating an enhanced learning experience for BME students" [40].

### 3.1.2 Effective Design of Educational Virtual Reality Applications for Medicine Using Knowledge and Engineering Techniques (Górski, 2016)

In the case of this study conducted in Poland (2016), the development of a standardized method for the creation of VR tools for biomedical engineering is being proposed. The basis of the project is the multiple independent simulators that currently exist and how they do not fit with each other. In this case, the researchers propose the creation of VR application development programs for biomedical engineering under the methodology "knowledge-oriented medical virtual reality" (KOMVR), where a work scheme is proposed such that the creation of a new application by experienced developers can be reduced from 50 days to 14 days.

Within the same study, the arguments in favor of the creation of this type of tool are presented. One of the most important is the number of hours required by students to retain information consistently, which varies depending on the type of learning approach used. It also includes relevant information about the different needs to be met by VR simulations depending on how advanced the student or health professional is.

This project proposes the creation of a virtual atlas of human anatomy, which will be the basis for all projects developed under this methodology. This, together with a prefabricated set of options and codes, would be the basis for new simulations, depending on the type of need. However, this requires the training of at least one experienced developer in order to meet the goal of developing a new application in a period of only 14 days. Even so, the study manages to demonstrate the effectiveness of this methodology and corroborate that, with access to the right information and awareness the speed of development is greatly improved regarding the development of didactic applications in VR.

## 3.2 State of Art: VR for medicine & Minimal Invasive Surgery Simulation

### 3.2.1 Virtual Reality Medical Training System for Anatomy Education (Ruthenbeck, 2015)

This study presents the point of view of medical professionals on new VR technologies and how the development of these technologies can influence new professionals in the area of health. Currently, there are numerous medical programs worldwide where the incorporation of VR tools for teaching future professionals has begun. In this study conducted in 2014 in Glasgow, a new VR tool developed in Unity for teaching the anatomy of the human heart is incorporated into the medical system, providing a complete interface where sections and slices of the heart can be shown. Within it, there is didactic support material and even tests such as quizzes on the functions used and general knowledge of the anatomy of the human heart.

As soon as the study was carried out to corroborate its effectiveness, positive results were obtained from the test subjects (medical students at Glasgow Caledonian University).

One of the most important conclusions to be drawn from this study is the realization that although the technology at the time of the study was capable of aiding and enhancing the learning experience and understanding of the students, there was scope for further development and immersion of the technology.

### 3.2.2 Pre-graduation medical training including virtual reality during the COVID-19 pandemic: a report on students' perception (DePonti, 2020)

In this study conducted in Italy, 122 medical students were surveyed regarding the virtual reality elements provided by the University of Insubria, Italy.

Among the elements provided is the Body Interact™ software, which is a virtual reality platform that can be accessed from VR headsets and smartphones (using Google Cardboard or similar). Within the same library, students can observe a variety of tests and diagnostic procedures, as well as surgical procedures. However, it is worth noting the lack of interaction on the part of these systems, which is limited only to the selection of content, questionnaires, and watching "virtual reality" videos.

It should also be noted that the results of the study are mostly positive with respect to the acceptance and use of this technology as a teaching method by medical students, where 55% of the students preferred the experience to observing these procedures live.

This study was conducted at the beginning of the COVID-19 pandemic, when social interactions were severely limited by the impositions given by the respective ministries of health of each country, so it becomes all the more relevant for the study group since this was the only way to receive this information for these students.

In general, it can be highlighted from this study the growing interest of the students in this technology, as well as the implementation of the same even in more accessible equipment financially, and finally, the results at the end of the semester, where consistently favorable results were reported in the grades of the students.

"Finally, randomized prospective studies are necessary in the future to assess the adjunctive value of this training modality along with outcome measures assessing whether medical students really improve their medical knowledge and clinical reasoning skills using this modality of virtual patient reality as compared with traditional training" [3].

### 3.2.3 Development of procedures for interactive simulation of minimally invasive surgical methods (Adler, 2014)

In this study by Dr. Simon Adler as his dissertation for his Ph.D. in engineering, a whole method for the correct simulation of minimally invasive surgeries was created. At the time of the study (2014), there were no pre-established tools of this kind for finite element analysis, so this project was based on the creation of the entire finite element analysis system and its implementation in a simulation. Dr. Adler demonstrates how to generate simulations of minimally invasive surgical procedures using real objects, such as laparoscopic instruments, that were tracked and represented in the computer during the development of this project. In the simulations created, it can be observed the development of internal organs from CT scans, as well as the creation of the meshes of the objects in 3D using differential equations for finite element analysis; all this is done "in-house."

This allowed not only the simulation of solid elements but also realistic interactions and even the ability to perform cuts and sutures. A statistical study was also performed on the number of hours needed for health professionals to perform operations on real patients using rudimentary simulators and the advantages that the digitalization of these simulators would bring to the area. In his conclusions, Dr. Adler expresses the importance that this type of simulation will have in the future for the teaching of professionals in the field of medicine and how it is also necessary to make development in other areas for more realistic simulations, such as fluids, to take this type of simulation to the next level.

### 3.3 Future Approach

Based on the study of related literature presented above, the value of creating tools of this type for programs related to health and biomedical engineering can be noted. The studies presented above lay a solid foundation, taking into consideration the technological advances that have been made in the area of 3D computing and simulation in recent years. One of the most important factors that has supported the creation of this project is the commercialization of VR systems by large technology companies. The great competition in this space has made VR devices more accessible to the general public, making the creation of educational software for them relevant and interesting for the didactic community.

Considering the relevance that virtual education systems have had as a result of situations such as the COVID-19 pandemic and the globalization of education, the creation of these systems is of utmost importance for higher education institutions. This also raises relevant questions for the development of this document, which will be the basis for the development of the methodology and later the analysis of the results obtained.

1. With the use of current simulations, how effective can the use of this technology be in terms of information retention?
2. For minimally invasive surgical procedures and tools related to it, is VR alone the best approach for better learning?

The above questions, together with the information presented in the theoretical and this section generate the following research question:

**Is VR technology a statistically relevant tool for teaching medical procedures and all the relevant elements to engineering students?**

Subsequent sections will be oriented to find the answer to this question, using both the software developed for this purpose and statistical tools.



# Chapter 4

## Methodology

The methodology chapter aims to explain the actual implementation of the project and the decisions taken to carry it out. This chapter will be divided into two sections. In the first section, a detailed explanation of the study to be performed, the tools included in the study, and the rationale behind it is provided. The second part explains the development of the application itself, the development of the 3D models, the scripts (and their use), and how it was all implemented into the simulation. Special emphasis will be placed on the simulation part's 3D modeling as well as the programming and the procedures taken for the decision-making process.

### 4.1 Study Methodology

The methodology used for the project's research is, by nature, quantitative. **Given that the primary goal of this research is to study VR technology as a supplement to the standard educational system and not against it,** it is necessary to gather information regarding the methods used in similar studies to understand how to collect the most amount of information.

Given the research question stated at the end of Section 3, it is defined that the type of study is non-experimental, which is to say, the data is going to be provided by third parties performing the study prepared by the researcher. It is called non-experimental due to the fact that the developer is not performing the experiment itself.

One of the most important points for any type of study is the recognition of the variables that can alter the results of the study in question and that define the population in which the study will be carried out. In the case of the present study, the following variables are considered the most relevant:

- Field of study of the participant.
- Academic level.
- Experience using VR equipment.

Taking this into account, the study population is defined as students of careers (within engineering) related to the areas of health, biology, and computing (because of the virtual reality component and the prospect of developing future applications with this simulation as a basis). These include:

- Biomedical engineering.
- Computer/Programming Engineering.
- Biotechnology engineering.
- Molecular Biotechnology engineering.

Due to the generality of the concepts presented in the area of human anatomy and physiology, it is considered that both undergraduate and graduate students are fit to participate in the study. The level of experience in VR it is inferred that it doesn't represent a significant variable for the issue of information retention; however, it will be taken into consideration to corroborate the data's relevance. There are also other variables; however, the above-mentioned variables will be taken into account for the study as the main causes of variation among the participants.

Once the type of population to be used for this study has been defined, it is necessary to consider the type of metric to be used, the level of acceptable margin of error, the reliability of the study, and finally the number of participants.

For the type of metric, the binary metric is considered to be the one that can best represent the results of the study in question. Other types of metrics are not compatible with the type of study in question, such as cumulative or measurement. Due to the type of metric used and the type of study proposed, it is considered that, with a reliability of 95% and a margin of error of 20%, the number of participants to be used in the study is 25 for the entire population.

For the calculation of these values, it was taken into consideration the number of available students at Anhalt University who met the requirements at the time of the realization of this study. To further corroborate this calculations, the Raosoft tool was used, which contains within its statistical functions a sample size calculator. The reasoning behind the calculation can be seen in Formula 4.1 through 4.3.

$$X = Z \left( \frac{c}{100} \right)^2 \cdot r (100 - r) \quad (4.1)$$

$$n = \frac{Nx}{((N - 1)E^2 + x)} \quad (4.2)$$

$$E = \sqrt{\frac{(N - n)x}{n(N - 1)}} \quad (4.3)$$

Where:

- n = Sample size
- E = Margin of error.
- N = Population size.
- r = fraction of responses
- Z(c/100) = critical value for the confidence level c.

As mentioned above, a reliability level of 95% with a margin of error of 20% will be used for the population size. This is because one of the biggest problems with this type of study is finding participants who meet all the necessary prerequisites and who can participate in this type of study.

For this type of testing, it would be ideal to have as many participants as possible; however, the number of students interested in participating is very low, so the level of reliability of the study and the margin of error are affected.

This will have repercussions in the results section, where it will be discussed in more detail how this factor affects the values obtained and what conclusions can be drawn from it.

### 4.1.1 DOE: Design of Experiment

In order to gather information regarding the effectiveness of the didactic tool created, a study has to be carried out in which the knowledge of the test subjects is put to the test. The same basis should be used for all test subjects to avoid interference in the results, which is why it is indispensable that the students who participate in the study have no previous knowledge of the subject acquired through minimally invasive surgery courses in both the undergraduate and graduate programs.

For the study itself, it was determined that the best way to carry it out is to create 2 control groups, in which one of them will be exposed to traditional teaching methods (a presentation in a master class) and the second control group will have the mixed teaching method, i.e., the use of traditional teaching methods, plus the reinforcement of the VR simulation. In order to determine which students will receive the traditional teaching method and which will receive the mixed teaching method, a random number generator will be used to assign the participants to one of the control groups. Figure 4.1 shows the general scheme of the experiment.

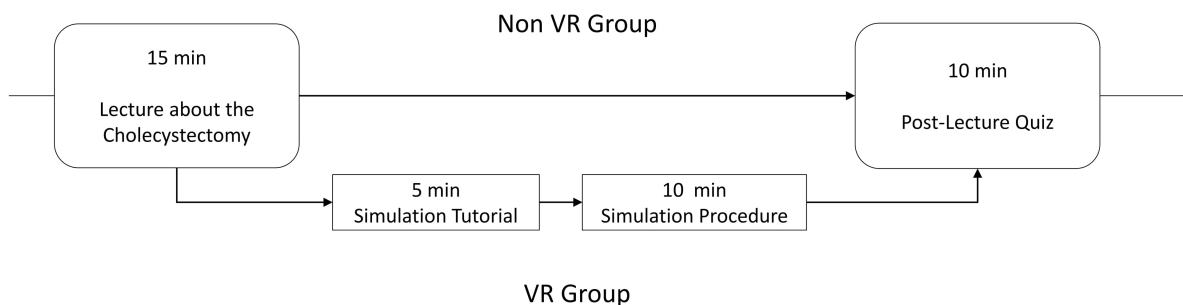


Figure 4.1: Diagram of the study to be carried out

Appendix E shows the presentation used for both study groups. It should be noted that not all the information necessary to answer the questionnaire is in the presentation, but part of it is transferred orally. This was decided to emulate more closely the real situations in a master class, where the presentation is only a support to the information provided by the teacher.

### 4.1.2 Data collection instruments

Information on the effectiveness of the didactic tool is obtained in the form of a questionnaire created to assess the participants ability to retain information. The effectiveness of the methods will be based on the number of correct answers obtained, the frequency with which they are given, and the feedback received from the study participants. For the data collection, it has been chosen to use the Lime Survey platform, which allows the creation of questionnaires with a wide range of complexity.

It also offers privacy benefits for the user, such as the option of using a local server and additional tracking prevention measures to ensure total anonymity of their data. Due to these reasons and the fact that Anhalt University has its own Lime Survey server, this option was chosen over the use of other, more mainstream platforms, such as Google Forms.

One of the most important points to mention in terms of privacy is that no personal information is collected beyond that discussed above (career, level of study, and whether the participant has previous experience with VR technology). From here on, in cases where a test performed by a specific participant is mentioned, the assigned number will be used.

### 4.1.3 Questionnaire & Statistical analysis

In order to assess if the use of the VR simulation translates into better information retention, it was necessary to create a tool to gather this data. The chosen method to do so was to create a questionnaire using the key concepts presented both in the presentation given to both control groups and in the simulation. The most relevant elements for biomedical engineering were used as the basis for the questions. Additionally, the types of questions used during the final exam of the Minimal Invasive Surgery master course were used as a reference. The questionnaire itself can be found in Appendix H.

### 4.1.4 Statistical analysis tools

For the interpretation of the data that will be obtained from the case study, it is necessary to make use of the right tools, both for data management and to perform statistical analysis with the data. One of the advantages of using a tool like Lime Survey is the ability to export all this data as an XLS file. These files are compatible with spreadsheet programs such as Excel from Microsoft's Office suite. This program is the industry standard for basic and advanced local data management (i.e., when data is not shared on a server with multiple people), so it is the software of choice for data processing. There are other alternatives such as LibreOffice, Google Sheets, and Apple Numbers; however, given the hardware chosen for the realization of this project (see more in Section 4.2.2), the most powerful and compatible option is Excel. With this program, it is possible to create diagrams for the graphical representation of the data; it allows the creation of pivot tables and even statistical analyses such as ANOVA. However, even using the Data Analysis plug-in, the options are limited in terms of the variety of statistical analysis, so the use of another tool that complements Excel is necessary.

For this, the Minitab software will be used. This is statistical software created by researchers at Pennsylvania State University in order to provide statistical inputs to a wide audience, especially within the manufacturing industry. Currently in the market, there are other options such as JMP, IBM SPSS Statistics, and Posit; however, these are oriented toward large corporations and are not easily accessible to the public, with trial versions with limited functionality. However, even in the trial version, all statistical analysis functions are present in Minitab. It also has an advantage over other statistical software because of its simplicity of use and the powerful "Statistical Helper" included, which guides the user on the best analyses that can be performed with the data provided by doing multiple test on the data before giving the user the different options available.

## 4.2 Application Development

### 4.2.1 Difficulties list

During the realization of the project, a series of situations were found that represented difficulties for its progress. Listed below (in chronological order) are the most relevant difficulties encountered.

- Problem #1: What design program should be used to work with? Is the software compatible with the other software used?
- Problem #2: There are no 3D models already available in the University resources for clinical spaces. It is then necessary to look for another source that can depict accurately a surgery room that fulfill the needs for this simulation.

- Problem #3: There are no 3D models for surgery tools already available in the University resources or physical tools available to model after them, therefore another source must be used.
- Problem #4: There are no 3D models of the organs required already available in the University resources and getting scans from patients proved to be inefficient, since there are a lot of privacy concerns and regulations, which is why another source must be used.
- Problem #5: How to modify user movement for the Windows simulation?
- Problem #6: How to realistically simulate the physics of organs?
- Problem #7: How to cut meshes already incorporated in Unity? Unity doesn't allow this natively.
- Problem #8: How is the user guided through the simulation?
- Problem #9: How can the VR controller commands be translated to Unity commands?
- Problem #10: How can unwanted interactions between scripts and objects be avoided?

During the development of this second section of the methodology, reference will be made to these situations in the sections where it is explained how they were overcome.

## 4.2.2 Equipment

### Operating System

The most common operating systems for 3D application development are MacOS, Windows, and Linux. However, the approach used by each of these systems differs greatly when it comes to working with virtual reality systems. In this case, the operating system of choice is Windows. This is because there are more drivers available for more computers than for Linux and MacOS. At the same time, the computers available with more powerful hardware configurations (dedicated graphics cards from either Nvidia or AMD, which have greater support from developers) are only available for Linux and Windows, due to the proprietary nature of computers using MacOS.

Table 4.1: OS Comparison for VR Development

	MacOS	Linux	Windows 10/11
Drivers Available	Many but proprietary	Open source, but not many options	Many open source and proprietary
3D Software Optimization	High optimization for 3D modelling	Limited options, hardware dependant optimization	Good optimization, moderate powerful equipment required
OS Application Development	Good development for the native OS	Good development for the native OS	Good development for the native OS
VR Application Development	Limited software compatibility	Mainly available through ports like SteamOS or OculusOS, gaming focused	High compatibility with most VR development software
Compatibility with VR Headsets	Moderate compatibility	Through ports as SteamOS and OculusOS the compatibility is good but only with the intended systems	High compatibility with most VR Headsets

However, there is an argument that for 3D design, Macs and the computers linked to this operating system are preferred by graphic designers, but this is only applicable for models that do not require a high level of precision, such as engineering models, whose software is supported almost exclusively for Windows. It is thus concluded that Windows is the optimal operating system for the development of this project.

### Computer Specifications

The specifications of the equipment used are:

#### Asus G14 2021 GA401QM Laptop

- Processor: AMD 5900HS
- Ram: 24GB Ram
- Graphics card: RTX 3060 6GB
- Operating System: Windows 11, build 22621.1194

#### Desktop-Custom Build

- Processor: Intel I7 8700K
- Ram: 64GB
- Graphics Card: RTX 2080 8GB
- Operating system: Windows 10, build 19042.572

### Virtual Reality System

The main VR system used for the development of the project is the HTC Vive Pro. This VR headset has the advantage of having one of the highest image fidelity of any VR headset available to the general consumer. There are devices such as the Primax 8KS DMAS that offer even better image fidelity, but the increase in resolution is unnecessary for this type of application when the price to pay is the system's performance.

One of the benefits of using the HTC Vive Pro is the use of the Lighthouse system, which allows for more accurate tracking of the controls and tracking of the user in a specific area, which is not available in all VR headsets. This is one of the headsets with the largest number of plug-ins created for the software choices described in the following sections. However, it is not the only VR headset taken into consideration for the development of this project. Additionally, plug-ins will be used that allow the use of other equipment and tools for developers to program for these headsets, as long as they are in the Steam VR or OpenXR compatibility lists. Both of these lists can be found in the appendix.

Another HMD system with which there is compatibility is the Valve Index. This is essentially the same HMD as the HTC Vive Pro, so it has the same level of compatibility as the aforesaid. This is an important point to highlight due to the availability of both HMDs during the realization of this project, where on more than one occasion these were used interchangeably.

#### 4.2.3 3D modeling and editing software

##### Solidworks 2020sP1:

This program was used for the creation of objects that required more precision, detail, and general models with more engineering orientations. The advantage of this program over others such as Revit or SketchUp is the wide variety of options for the creation of objects with a high level of accuracy. Additionally, because this is the educational standard for engineering careers, there is already familiarity with its use. The version used is the test version, so it is necessary to export the models in STL format and then convert them to be able to edit them.

**Blender 3.3.1:**

This program was used as the mediator between all 3D models. Due to its ability to import a wide range of 3D models, it was used as a bridge between the 3D models obtained from different online libraries, the own models, and the development program.

It was chosen over other software options because the use of it is available for free even though is an extremely robust tool, its high level of compatibility with multiple formats of 3D elements, and its powerful engine, capable of generating high levels of detail in the textures used. This program is also where the final editing and adjustment of textures are done in order to add realism to the objects. This software has a high level of compatibility with the chosen IDE and the other developing software, so it is the best option to integrate into the workflow. **This solves Problem #1 listed in section 4.2.1.**

**Unity 2020.3.37f1:**

This program was chosen to develop the simulation due to its high compatibility with different systems, including Windows (and the development of VR applications) and the HTC Vive Pro (and the Valve Index). To avoid compatibility issues between different versions, the version chosen for the laptop system is the same as that used by the computer assigned at the Hochschule Anhalt's XR Laboratory. Within the software itself, several so-called "Unity Assets" are used, which can be purchased through the Unity Asset Store. In the figure 4.2 the workflow used for the 3D pipeline is shown, covering the use of the three programs mentioned in this section.

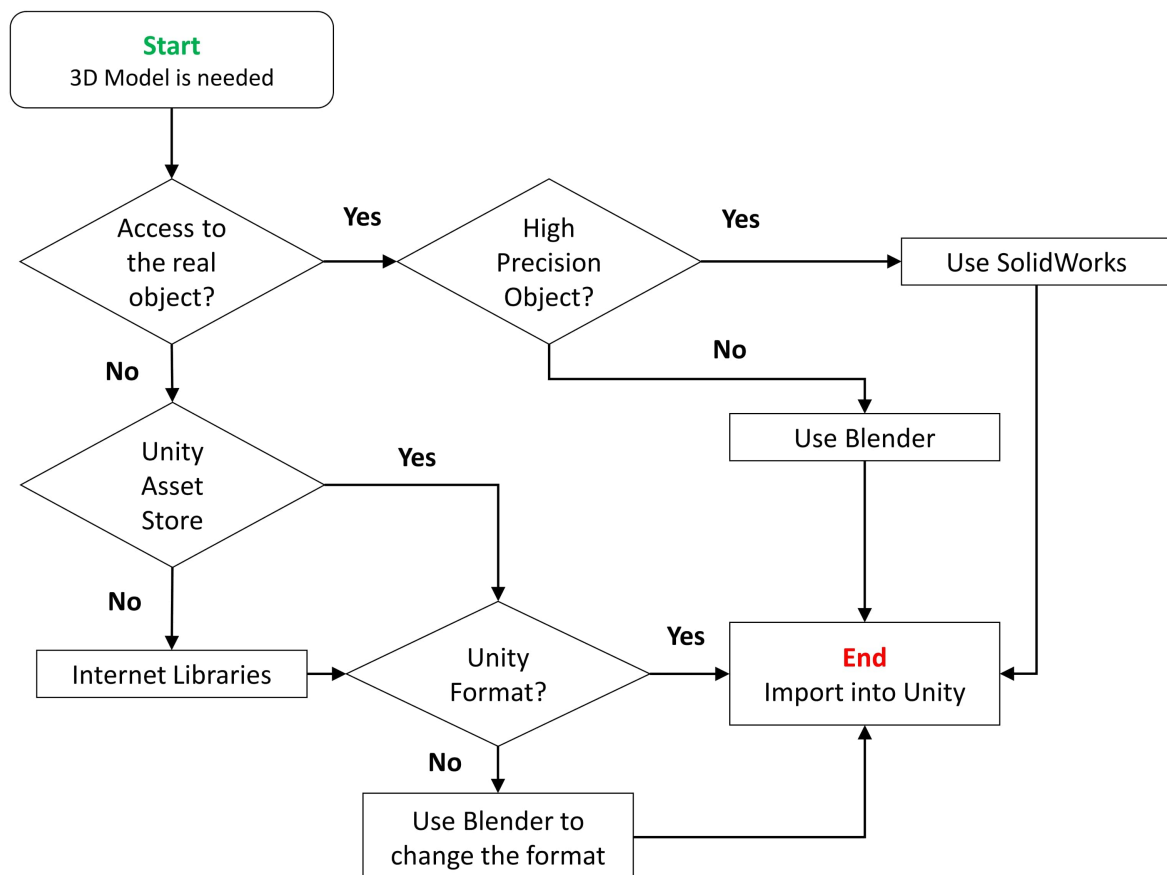


Figure 4.2: Workflow chart for the 3D modelling Pipeline

### 3D Model File Format

For the choice of the file format to work with, the most common types of files for 3D modeling were considered: STL, FBX, and OBJ. Each of these formats has advantages and disadvantages in terms of level of detail, final functionality, and other aesthetic considerations, such as the incorporation of textures.

In first place is STL. Currently, STL is not a format accepted by Unity, and there can be a lot of geometry errors when transferring from one program to another, so depending on the complexity of the element, it is necessary to make adjustments and do the procedure repeatedly. In the case of OBJ, it is a more widely accepted format by game engines such as Unity. This format, on the other hand, has the disadvantage of not bringing the textures designed within the model but having to be handled as separate files, which often results in incorrect implementation in the graphical engine of choice, whether an object is scaled or simply a matter of file organization.

This leads to the implementation of the FBX format. This is the preferred format, since the 3D models imported in this format contain all the texture maps, normal maps, and details baked into the 3D object, so no further modification is needed.

It should be clarified that FBX is an Autodesk proprietary format made for the creation of high-fidelity 3D objects. However, because of the partnership between these two companies, the fact that the format is Autodesk's has no bearing on its compatibility with Unity.

#### 4.2.4 Virtual Assets & Objects

##### Unity Asset Clinic, Operating Room by Mixall

This asset created by the Mixall group contains a series of 3D elements typical of surgery rooms for both major surgeries and outpatient procedures. It includes prefabricated sets of rooms with removable elements, lights, and textures incorporated into them. For this simulation, prefabricated set 1 was used, and the necessary modifications were made to accommodate the needs of the simulation.

This asset is by far the most complete in the Unity Asset Store, with hundreds of elements and a lot of features. This also solves the problem of modeling objects that are not easily accessible, such as surgical elements, many of which are single-use, so taking measurements of them would mean wasting a perfectly functional tool. **This solves problem #2 listed in section 4.2.1.**



Figure 4.3: Front page for the Unity Asset Clinic-Operation Room by Mixall



## Surgical Instruments

For the variety of surgical instruments used there's the following list:

- Scissors and Scalpels: Various Surgical Instruments from Sketchfabs by Fero Adezo. In this free package, it can be found: scissors, forceps, scalpels, and a variety of other basic surgical instruments. An edition was made to incorporate only the forceps and scalpel because the other parts of the package were considered irrelevant for the procedure.
- Endoscope: The 3D model used was acquired from the free GrabCad library by the creator of the model, Hazam Eltaief. For use in this simulation, the only modification that was made was with the built-in textures.
- Trocars: The models of the Trocars used were acquired from Sketchfab by the artist "GulzarMrRobotics". This model originally included an endoscope and a trocar; however, it was modified in such a way that only the trocar was used with the original textures.

The reason for only using the trocar model is the large number of polygons incorporated in the geometry and the performance hit that this would imply when used in combination with the other simulation elements.

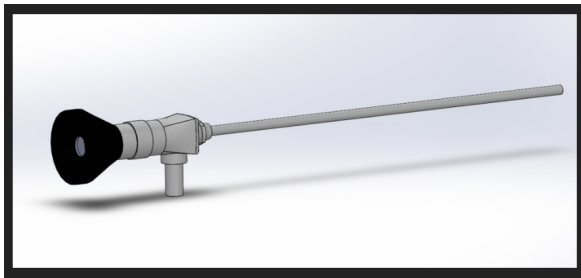


Figure 4.4: Endoscope Model by Hazam Eltaief

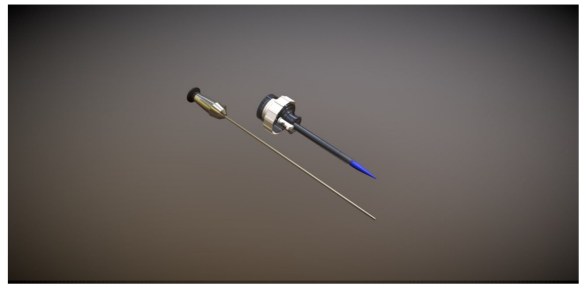


Figure 4.5: Trocar model by "GulzarMrRobotics"

- Grasper: The elements used for the creation of these surgical tools were taken from two different sources. In the first instance, the base model was obtained from CGTrader, which was acquired from the developers of EMAS-Studio. Modifications were made to the original model, including incorporating different heads in the tool to vary between scissors and graspers. For the model of the graspers, a free model was obtained from GrabCad, which is a platform for engineering students where they can find free models from other engineering students.

SolidWorks was used to make the modifications, and then Blender was used to join both models, so in the end, a model capable of using different heads for different functions was created. There are more elements used (which can be found in the appendix), but the most relevant for the development of the project are the ones mentioned above.

**All of the above mention assets and 3D models solve Problem #3, presented in section 4.2.1.**

## Body Models with Internal Organs

The complete model of the human body was acquired through CGTrader, which is a library where very detailed models can be purchased. The model used in the simulation is the one

developed by Nico Robin, which involves 3D models of all the organs of the body, including veins, arteries, and the lymphatic system.

Due to the complexity of the model, the number of meshes, and the high resolution of the textures, Blender was used to reduce the number of polygons of some of the less important elements in order to improve the performance of the simulation on both the computer and the HMD.

Similarly, modifications were made to the elements of interest such as the gallbladder, where unlike the rest of the models, an upscale was made to the resolution of both meshes and textures, which not only serves as an aid to the visual element but also—as it will be explained in the coding section—allows certain simulations to be performed in a better way.

The acquisition of this model solves one of the major problems with this type of simulation, which is obtaining anatomically correct models of the human body. One of the alternatives is the use of scans of real patients and the implementation of modeling through these images, or CT scans, in which case it is necessary to use a program such as 3D Slicer to create the models. However, this alternative requires access to patients whose scans contain the organs of interest, explicit permission from the patients, and provides a large number of inconsistencies in the 3D models, which is why a pre-fabricated model is chosen.

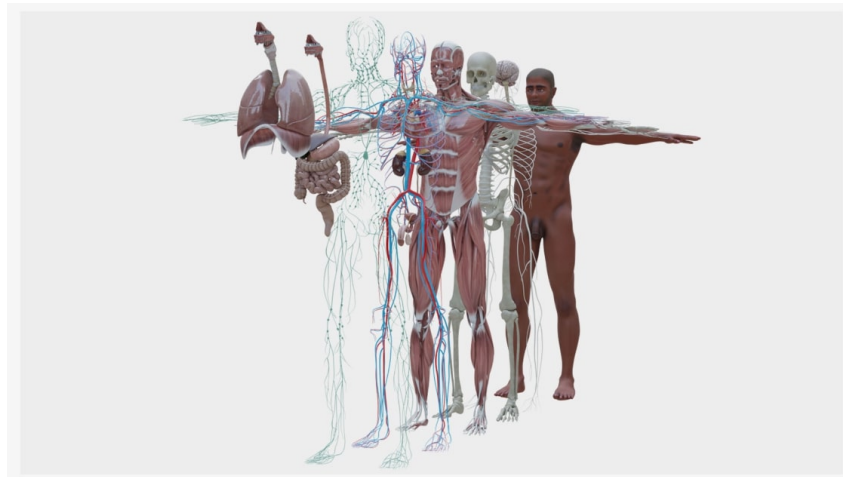


Figure 4.6: Human body 3D model created by the artist Nico Robin.

**This solves problem #4 as presented in section 4.2.1.**

#### **Player: Starter Assets - First Person Character Controller**

For the creation of the "user" object, the asset "Starter Assets - First Person Character Controller" from the asset store was used. First-person viewing means that the package provides a camera where everything is observed as if the user is in the scene and incorporates within Unity the new prefabricated input system, where the developer no longer has to create scripts and program the user's actions, but these come with a simple menu where it is only necessary to assign the desired inputs.

However, this is not the only system used. Unity also has the ability to use the "old input system" if desired by the user, which allows certain more linear movements used in some sections of the simulation.

**The use of this asset solves partially problem #5 list in section 4.2.1. Nonetheless, to fully solve this situation is necessary to refer to section 4.2.5.**

## Morphological Box

Morphological boxes are tools used to summarize the decision-making process explained in this section. These present the subtasks to be performed and the decisions to be made, based on the arguments presented in this section.

Table 4.2: Morphological Box for the 3D modelling and editing software

	Identifier	Problem	Solution #1	Solution #2	Solution #3
3D Modelling and Editing Software	1	3D models with high precision	SolidWorks	Revit	Blender
	2	3D models with normal precision	Blender	SketchUp	-
	3	Game Engine	Unity	Unreal Engine	-
	4	Anatomically correct organs	Create 3D models by using realife data in SolidWorks	Adquire them Unity Asset Store	Adquire them on the internet
	5	Acurate surgery instruments	Create 3D models by using real objects in SolidWorks	Adquire them Unity Asset Store	Adquire them on the internet
	6	Surgery Room	Create 3D model in Blender	Adquire them Unity Asset Store	Adquire them on the internet
	7	Setup	All in one scene	Multiple Separate Scenes	-
	8	Object Format	FBX	OBJ	STL
	9	Textures	Create own Textures	Adquired them on the internet	Use prefabs
	10	Fix scale of objects	Blender	Unity	Solidworks

- Solution #1: Solidworks was the software of choice for developing 3D models with a high level of accuracy when required, being a software used for this type of modeling in manufacturing. Being a standard software in many engineering programs worldwide, there is more familiarity with it. Additionally, it supports exporting 3D models in various formats natively.
- Solution #2: For models that require average or low levels of accuracy, Blender was chosen. This software allows not only the creation of objects in a fast way but also has the advantage of working in FBX format, with which it is possible to export objects with baked-in textures.
- Solution #3: The game engine of choice was Unity, due to its wide compatibility with VR devices, its robust and large community of developers, as well as the Asset Store, which contains thousands of items (both free and paid) available.
- Solution #4: For the creation or acquisition of the 3D organ elements, it was decided to acquire them from a 3D artist in an online marketplace. This was due to the limited options that exist in the Unity Asset Store and the lack of anatomically correct models to model the necessary organs in 3D.
- Solution #5: Similar to what was stated in solution #4, it was decided to acquire the elements that are not accessible through secondary markets on the internet.
- Solution #6: For the creation of the space where the simulation takes place, a combination of the asset store, clinic, and models created in Blender was used. Although there are options outside of the Asset Store in secondary markets, these are usually not optimized for use in VR.

- Solution #7: For the organization of the program, it was decided to use multiple scenes with different sections in each one of them, since the alternative of doing everything in a single scene can generate complications both in terms of performance and compatibility between all the different scripts.
- Solution #8: For the formats with which to handle the 3D objects, FBX and OBJ (wavefront) were chosen. OBJ has the advantage of being one of the most common and oldest formats, so there are many 3D elements in this format. In the case of FBX, it has the advantage of being a format that includes within the 3D model information the textures, normal maps, and height maps, so it is not necessary to import them separately into Unity.
- Solution #9: For the textures, it was decided to use both the original textures of some of the elements acquired either in the Unity Asset Store or through secondary online markets, as well as make the necessary modifications using programs such as Photoshop or its online free competitor, Photopea.
- Solution #10: For the management of the 3D elements, the decision was to use Blender. Due to the flexibility of the program, the number of formats it can export, and its easy availability, it was chosen as the best option.

#### 4.2.5 Unity Development & Coding

To develop the code used and ensure its correct implementation, Microsoft's Visual Studio 2023 software was used, together with the plugin for Unity development, which comes with all the basic functions and call-outs for the Unity Engine. It is because of these plugins that the integration with Unity is so seamless and therefore the preferred choice over other compilers or text editors.

The first thing that must be taken into account when creating any type of interaction in Unity is that the programming language is C#. This object-oriented programming language has a very direct syntax toward the functionalities that each object can host. In this way, it is possible to activate and deactivate prefabricated functions for an object, making use of the functionality.

One of the greatest advantages of this programming language is the closeness of its syntax to the English language. Many of the actions within the code share similarities with their colloquial language analogs, making it easy for a relatively new user to create simple functions. However, it should be noted that interaction between code from different sources, especially in packages purchased from the Unity Asset Store, can be limited and, in some cases, impossible because the developers of these functions focus only on compatibility with the Unity Engine, i.e., only the most basic functions.

In this section, it will be discussed the most important steps to be taken to create a simulation, as well as the developed, acquired, or edited code to obtain the desired results and the logic behind them.

#### User movement

In order to provide the user with a way to move within the created space, the necessary scripts must be created to allow an object to move. In this case, it should be noted that Unity currently has 2 types of input systems for Windows simulations, which are referred to as the "old input system" and the "new input system."

The old input system uses commands provided by scripts to demarcate a key of interest and determine if the key has been pressed and when it is no longer pressed. This can be done with the `Input.GetKeyDown()` and `Input.GetKeyUp()` commands. This has advantages when developing programs for multiple platforms because it allows to map actions even with unusual inputs, such as the VR kit controls.

Unity's new input system uses a prefabricated interface in which keys are assigned from a preset library for each action. This makes it easier to start programming; however, the developer must have the correct Unity packages in order to be able to develop for other devices that do not have a keyboard, and in the case where there are no packages for any product or platform, this method cannot be used.

In the case of this project, it was required to use both Windows and VR simulations, so Unity has the option to make use of both input systems at the same time. This is convenient for the development of the project, but special care must be taken in the guidelines marked in the code to avoid compilation problems by receiving more than one input or none at all.

**With the implementation of this code, it is now possible to say that problem #5 is resolved, as presented in section 4.2.1.**

## Camera

One of the most important concepts to understand in the development of user movement are the concepts of orientation, position transformation, and rotation (and the role that the camera plays). Every time a new project is created in Unity, it automatically creates two elements in the default scene: a light and a camera. The camera in Unity serves as an analog to a person's eyes.

Here the concept of `GameObject` needs to be introduced. The `GameObjects` are all those elements in Unity that can be assigned characteristics and are in the scene. It is important to clarify that the fact that the name refers to an object does not mean that it has to be a physical object or that it has to be seen in the scene. There are empty `GameObjects` whose only function is to receive input from scripts and implement their own.

Being a `GameObject` it has position, rotation, and scale, which are standard properties of any `GameObject`, even empty ones. These can be modified at the developer's convenience during program development; however, unless the necessary code or functions are assigned within Unity, they remain fixed during simulation. Based on the brief explanation above, code can be developed in which these parameters are modified to simulate the movement of a person. With code, it is possible to relate the user inputs to the parameterized position of the object.

To make such a correlation between the inputs and the position of the object, the transform option can be used, which makes an update of the linked parameter, in this case the position, but also the rotation can be used as long as it is transformed into degrees.

## Player

For the `Player` object, the developer can use Unity's integrated coordinate system, which has both horizontal and vertical references, allowing the developer to move the user between the x and z axes in Unity. Unity also recognizes the forward and backward parameters, which are used to move on the z-axis only, so the user must use it with these parameters in mind.

This should be merged with Unity's functionality to determine the time, initial condition, and what happens between each frame that is presented. When new code is created in C#, there will always be two standard sections: start and update. As its name suggests, Start dictates the

initial conditions of the script, and Update indicates what happens each time a new frame is rendered. It should be noted that objects and variables do not have to be defined in Start, since by this point the program expects to have defined all the parameters to be used when pressing Play within the simulation.

With these considerations in hand, it is possible to create a script where the initial position of the user is defined, and each time that the desired or assigned keys are pressed, a multiplication of how long they have been pressed is performed, and an update is made to the position of the user. At the same time, this can be related to the code to obtain the mouse position and indicate the direction of what's ahead, linking the camera object so that everything moves as a single unit. Care should be taken to place the camera and the user's game object in such a way that the user can observe the elements without the user being able to observe himself.

#### 4.2.6 Coding for Physics Simulation & Items

Once the user is able to move through a scene, it should consider the elements called colliders and their functionality within C# and Unity. These are boundaries that allow physical behaviors to be emulated, such as the perception of solids between objects. When a new object is imported into Unity, it lacks any physical properties; it is simply a shell with a desired shape or appearance. In order to add realistic characteristics to the mesh, there is the option to add borders of predetermined shapes such as cubes, pills, cylinders, and spheres, which of course can be modified to fit the shape of the object.

There is also the option to create a specific model for each mesh, using it as if it were a collider. This has advantages when a high level of detail is required, but in more general interactions it does not bring much benefit and, on the contrary, can make the simulations more difficult. The second option to provide realism to the objects is the incorporation of so-called rigid bodies. This makes use of Unity's physics engine to simulate gravitational forces as well as the speed of reaction of the bodies. Both options, colliders and rigid bodies, have their own set of functions accessible through scripts, which provide developers with greater latitude to determine what is required and the type of code being used.

The development of the simulation must take into account the different types of items to be used, their functions, and the scene in which they are to be implemented. In the case of the operating room, the surgical instruments are expected to be able to be taken and positioned correctly. This means that the correct position of the instruments, the rotation, and the characteristics once the correct position is reached must be predetermined. This allows the use of simpler colliders and scripts where the transform.position function is invoked.

However, using these functions only on objects is not enough to enable them to be picked up and moved freely. For this, a script has to be configured where user input is detected with the intention to pick up the object, move the object with the user, and the function to drop the object with additional input from the user. This can be done using the aforementioned new input system from Unity, which provides these prefabricated functions with the possibility of assigning different keys on different platforms to them.

This in turn raises a bigger question: How do the developers know if it is aiming at an object that should be able to be picked up or not? Here, the developer can use the raycast function. Raycast allows the developer to access the functionality of a laser pointer linked to the user, camera, or object. With this function, the developer can get the properties of an object without the need to invoke it directly by name in code, so it can be added to an identifier code (this is an empty code, whose function is only to provide a name that can be obtained with the GetComponent function) and the functions based on it can be assigned, mainly with "if" functions. The scripts related to this can be seen in more detail in the appendix section.

## YuePhysics

In the case of the interior of the human body, it is necessary to have more realism between the interactions of the objects, so it has been implemented through the use of a Unity asset called YuePhysics. This asset is a set of scripts and proprietary physics systems that allow the developer to simulate interactions such as stretching and retracting an object.

This works by employing a system of creating mini colliders, which for convenience can be called "particles." This system of particles follows the shape of the object to which the function is assigned, either reliably (which requires a lot of computing power, so it is not the preferred method) or by a simplified method, where it creates one of these particles at intervals defined by the developer. **This solves problem #6, listed in section 4.2.1.**

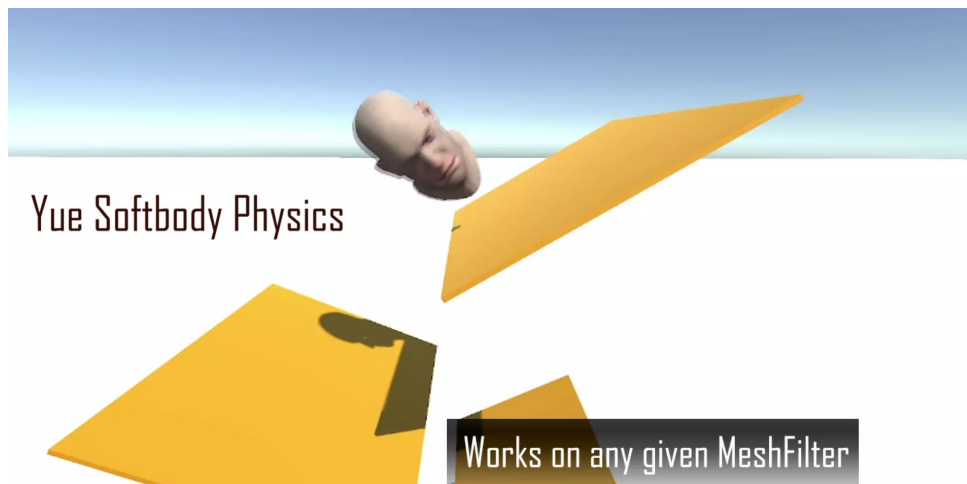


Figure 4.7: Unity Asset Store banner for Yue Physics

## BzSlicer

One of the most important functions within the implemented code is the possibility of being able to make cuts. This is especially complicated due to the nature of the meshes. To understand how this code works, it is necessary to understand the mathematics behind it. Each of the meshes imported into Unity has a certain number of vertices at the time of its creation, which are connected together to create the faces and edges of the mesh.

What is understood as making a cut to an object in Unity is to make a subdivision of a mesh into two segments or more. The mesh analysis of the Unity Asset Store's BzSlicer asset, which uses the mesh vertices and a Raycast to recalculate using finite element equations, creates two objects: the parent (the original mesh) and the child (the new mesh, which has the name of the original plus the added name "geo").

In principle, the developer is given the tools to create a raycast and a temporary plane, which will serve as a guide for the desired cuts to be made. Here the raycast works as a guide where temporary planes can be developed. The desired object is also assigned its own code, where an analysis of vertices and faces is done, assigning the sliceable function. Within this code are functions such as GetRenderers and MeshDissector, which make references to the own system created by the developers of BzSlicer. It is in these systems that the calculation and analysis of finite elements takes place, creating a surface analysis of the mesh into which it is incorporated and on the side of the cutting tool or slicer to capture the point of intersection between the plane created and the sliceable mesh.

From here on out, the guideline is to create a new object that retains the qualities of the parent object but has a new mesh structure that is only rendered from the normal of the cutting plane. The developer has the option to modify the scale and position of the plane to some extent, as this is dependent on how far the Raycast function reaches. Once both the mesh and the plane intersect, an input can be assigned by the user to execute the function, or, if desired, the function can always be active and whenever they intersect, to perform the cut.

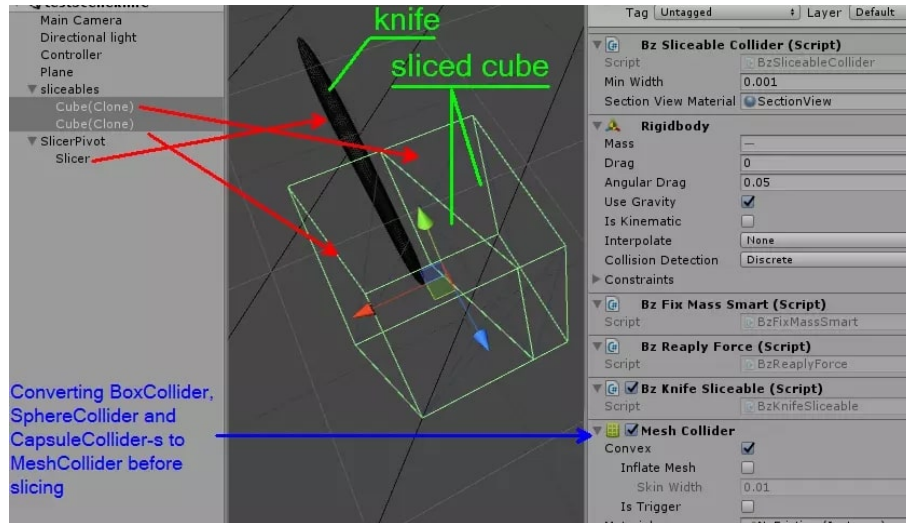


Figure 4.8: Working principle of BzSlicer

These codes show a great ability to adapt to complex mesh shapes; however, it must be taken into consideration that these are not based on the physical appearance of the object but on the mesh collider assigned to the objects, so curved objects may present problems when cutting in more than one section, so the developer must modify these shapes in 3D modeling programs before importing them into Unity for the scripts to work.

Because the raycast function is being used, it can be assigned to either the camera (it will run in the direction z in which the user is looking), the mouse position (it will run perpendicular to the "x" and "y" positions of the mouse on the screen), or an object within the scene. This last option is particularly important for the project because it allows for the use of objects capable of making cuts, such as scalpels and laparoscopic scissors, as well as the ability to disable such functionality when changing these tools using Scene Managers. **It's with the implementation of this asset that Problem #7 is resolved.**

#### 4.2.7 Teaching elements

One of the most important factors to take into account for this simulation is the transmission of knowledge to its users. That being said, it is not enough to make a 3D design and the interactions of the simulation; there must be added elements that guide the user and help with the understanding of the procedure, the details, and the steps to be performed.

#### Tutorial

Once all the programming and control assignment steps have been completed, it is necessary to give the user the opportunity to become familiar with the controls, so creating a tutorial with simple tasks is the best way to accomplish this.



The virtual reality experience, even though it is reaching the mainstream market, has the disadvantage that there are still a large number of potential users who have never experienced it before, so it is necessary to provide a comprehensive guide that demonstrates the controls, what actions can be performed, and the limitations of the experience.

For this to work, a new scene has to be created in Unity, making use of the aforementioned 3D elements and their respective assignment of controls. As part of the streamlined experience, a guide should be made with visual aids—large elements with which the user can become familiar.

### Guidance during the procedure

In the case of the cholecystectomy simulation, two sources of help were incorporated to make the educational experience more complete: visual help and auditory help.

In the case of visual aids, elements were incorporated to guide the user through the steps to be followed, as well as elements that allowed the user to learn about the most relevant theoretical concepts when performing the experiment. For this, Unity's ability to receive a large number of image formats can be used, which can be done in any editor such as Photoshop from the Adobe Suite or Gimp in the case of free license software. These aids have to be considered taking into account their placement within the simulation and the fact that they can only be represented as a plane, i.e., they cannot adopt more complex shapes without distorting the image and compromising usability.

There are also visual guides that have the function of orienting the user on the steps to follow within the simulation, such as the arrow elements, which indicate to the user which elements should be taken and in what order, following in a reliable way how the procedure is performed. One of the most useful elements to indicate the steps to be followed by the user, as well as to follow the process of the simulation itself, was the incorporation of a whiteboard with the detailed steps to be followed, check boxes to check that the tasks had been successfully completed, and reset buttons, with which the user could listen to the instructions again and/or repeat a specific step.

One of the elements mentioned above is the incorporation of an audio guide. For the creation of the audio guide, a script was written with detailed instructions on the steps to follow, hints on how to perform them, and all the descriptions of the controls indicated in the tutorial. Once completed, the script was divided into the individual parts to be incorporated into the program, and the audio file itself was created.

For this, a program called Replica Studios (free trial) was used, which used AI and pre-recorded voices from a database to obtain results well above average for this type of application. The developers of this program focus on the creation of voices and dialogues for video games, so it has native compatibility with Unity. Still, if the developer wants to access the files, they can be exported in MP3 or WAV format and edited in programs like Adobe Audition or Audacity. Once that's done, it's necessary to create C# scripts in Unity that can activate and deactivate the audio, depending on what's happening at that moment in the simulation. **Is the combination of both the visual and audio guides that solvents problem #8, listed in section 4.2.1.**

#### 4.2.8 Working Method for Windows Simulation

The steps to follow for the simulation can be seen in the following list.

1. Conduct research on the subject to be developed. In order for this to be achieved, a research methodology has to be developed where the technical background, related documents, and other relevant information sources are to be searched.

2. Create a list of 3D objects to be obtained, created, or modified. Consideration should be given to the limitations of the equipment the developer is working with (in terms of computational power) and the simulation approach (realism vs. practicality).
3. Use a program that allows the transformation of all object formats obtained or created into a single format, that accepts the transfer of as much relevant information as possible; FBX has been found to be the format that allows the greatest compatibility and reliable transfer of information between programs.
4. Based on the information obtained through research, generate educational content, either in the form of images, audio, or video. The necessary creative software must be used for this, while keeping the purpose of the content in mind at all times.
5. Make use of an IDE as a platform where all the previously developed elements are put together.
6. Develop a macro-schema of the functions, scenarios, and properties of the elements to be used, to optimize the development and performance of the application.
7. Develop the necessary code, making use of tools such as information forums and code available on platforms such as the Unity Asset Store. In this section, it has to be taken into account which platform is being developed, the type of input, and the target audience of the application.
8. Debugging. In this case, it is of special importance not only to use the code acquired through the Asset Store, but to understand its functionality and adapt it so that it can work with the rest of the developed code, since, as mentioned before, most of these assets were developed with basic Unity compatibility in mind.
9. Generate alpha and beta tests by using the "build" function to find deficiencies and compatibility problems.

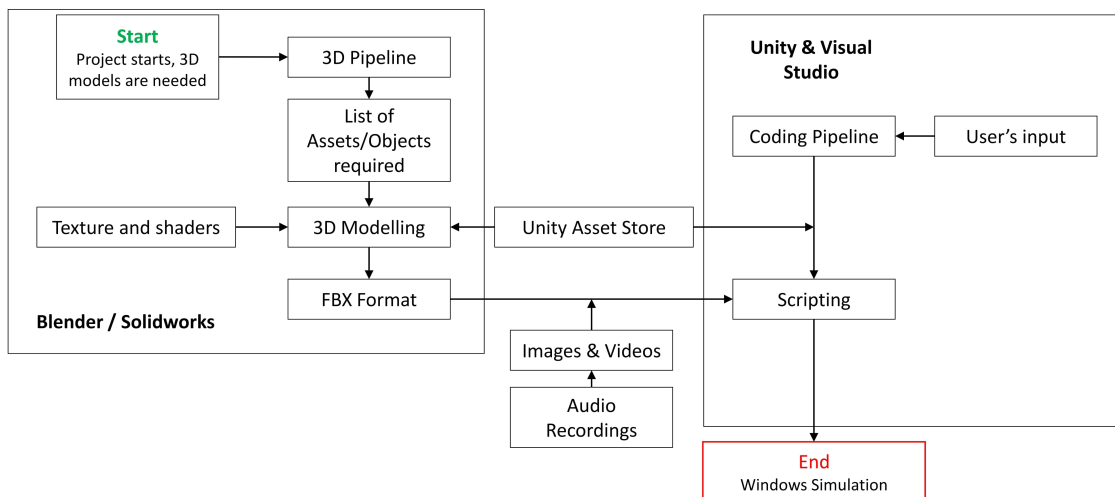


Figure 4.9: Complete pipeline for the Windows Simulation

#### 4.2.9 Morphological Box

The morphological box for the programming and code section is presented below. As mentioned in previous sections, this is a tool that aims to explain the decision making process performed in this section.

Table 4.3: Morphological Box for the Code &amp; Programming section

	Identifier	Situation	Solution #1	Solution #2	Solution #3
Code & Programming	1	User Input / Move player	Old Input System	New Input System	Both
	2	Add realism to the internal Organs	Create own code for interactive meshes	Adquire code Unity Asset Store	Code from Github/ Independent
	3	Ability to cut meshes	Create own code	Adquire code Unity Asset Store	Code from Github/ Independent
	4	Physics system	Create custom physics system	Use Unity's default	Get custom system from independent developers
	5	Lighting System	Normal Unity Maps	Create Custom Lightmaps	Use reflexion spheres and let Unity create one
	6	Internal Surgery Tools	Give the option to the user to choose	Give limited options	Don't give any options to change the tools
	7	User help	Create tutorial	Rely on the control prefabs	-
	8	User interaction with the enviroment	Only with intended objects	Wih all elements	-
	9	Manage Scenes	One code for all scenes	Independent managers per scene	-
	10	Educational elements	Appear according to what the user is doing	Always available	
	11	Right position for tools	Create own code	Adquire code Unity Asset Store	Code from Github/ Independent
	12	Tools interactable after being in the right position	Give the user the option to move it afterwards	Don't give the user the option	-
	13	Create interactable organs	Just the ones of interest	All the organs	None

- Solution #1: In order to determine the type of user input, the advantages of each of these systems were taken into consideration. The old input system has the advantage of being able to assign any type of key to the user's movements, which is advantageous for the development of applications for different platforms, while the new input system uses a prefabricated interface to assign the inputs within the options provided by Unity. This helps to make the incorporation of controls for certain actions easier, but does not provide flexibility when assigning buttons outside the spectrum offered by Unity. Therefore, it was decided to make use of both systems, depending on the scenario, at our convenience.
- Solution #2: To add realism to the organs and physical elements, it was decided to acquire the code through the Unity Asset Store. The task of providing the meshes with realistic elastic properties is a programming project in itself, outside the scope of the current document. Additionally, sources found on secondary markets and websites such as Git Hub have limited support and compatibility issues.
- Solution #3: In order to be able to dynamically slice meshes without resorting to default slice sections, it was opted to purchase the BZ Slicer from the Unity Asset Store. Similar to what happened with the realism for the internal organs, creating a method for finite element analysis and creating new meshes from them is out of the scope of this project.
- Solution #4: The physics system implemented is the standard one used by Unity, with modifications in a single scene where the option to create your own system is used. It is not necessary to make use of code from independent developers due to the simplicity of use and modification that Unity provides.
- Solution #5: One of the important points for the aesthetic part of the project is the creation of lighting maps and the use of reflection spheres to add realism. This was decided over the normal lighting maps created by Unity, given their simplicity and the fact that the development of custom lighting maps does not add any impact at the time of using the program, only in the initial rendering of the scene (which is done prior to the use of the program).
- Solution #6: Due to the nature of the simulation and the target audience for which it is designed, it was chosen not to give the user the option to switch to other tools in the

surgery section of the simulation. It was decided to use the most common and widely used tools for this type of surgery because these are the most relevant ones for the biomedical engineering branch. Furthermore, the focus of the simulation itself is not for biomedical engineering students to learn how to perform the surgery per se but rather to familiarize themselves with and better understand the equipment most commonly used by health professionals to perform this procedure.

- Solution #7: Even though VR technology has reached the mainstream market, most people have never used a VR device, so it is necessary to familiarize people with this type of technology before proceeding with the simulation. For this, it was chosen to create a tutorial before the procedure itself, where the most relevant interactions and the buttons associated with them are explained. Within the software itself, unity has certain prompts that can be implemented when the user approaches an object that has an associated action; however, it was considered that they are not enough for users who are less familiar with this technology.
- Solution #8: To simplify the experience and keep the procedure within the appropriate steps, it was opted to restrict the user to the elements relevant to the stage of the simulation that are encountered. This will allow the user to focus only on the steps with elements of interest from a biomedical engineering point of view. Additionally, it provides the advantage of avoiding unwanted interactions between elements and code not previously foreseen, thus decreasing the possibility of bugs.
- Solution #9: The use of scene managers is of vital importance to corroborate that the elements and the steps to follow within the simulation are kept within the script as expected in the different scenes. Taking this into consideration, the use of a single code that handles all the scenes, although possible, would be extremely inefficient and more prone to failure than the creation of independent managers for each of the scenes, which is why it is considered the best alternative.
- Solution #10: For the educational elements, a compromise between making information available at all times to the user and displaying only the most relevant information was chosen as the best option, depending on the stage of the process in which the user is. In this case, the user is presented with the relevant information for the step he or she is in, with the option of having an overview of all the steps to follow in the given scene. Additionally, a scene was created in which all the most important information of the surgical procedure is compiled and the user can review what he or she has learned.
- Solution #11: The code for the correct position of the objects, depending on their type, was created by registering the correct position of the objects and using the code `Vector3.Distance()`. The existing options in both the Unity Asset Store and GitHub included a considerable amount of unnecessary code for the simulation, so they were discarded as options.
- Solution #12: Once the objects have been placed in the correct positions, the user is not given the option to modify the position. This is to avoid bugs with the written codes since most of them use methods that are invoked in the `Update()` class section, which is refreshed in each frame of the simulation.
- Solution #13: With the same argument of providing the user with only the necessary interactions to complete the steps of the simulation, it was decided to only provide all the necessary interaction levels to the relevant bodies in the procedure.

### 4.2.10 VR Integration

For the integration of the VR section, it is essential to take into consideration the translation platforms that already exist within Unity and on the target device.

As stated before, the HTC Vive Pro and Valve Index are the VR headsets chosen for this simulation due to: screens with a higher resolution than average; greater accuracy of user tracking and controls thanks to the use of Lighthouse technology, which uses two external (or more) sensors that can track the user within a certain area; However, considering the limited options and the possible development of more didactic tools in the future, a wider integration of other VR systems will also be taken into consideration.

The first plugin to be used is SteamVR. This plugin allows Unity to connect to the Steam VR platform, which has a large library of applications and video games managed by Valve. This platform is very robust and has great compatibility with a wide range of VR headsets and controllers, making it the largest library of VR applications available to the general public.

This plugin comes with a number of built-in prefabs where basic user functions and control tracking are already included. It also includes numerous scripts to generate the most basic of interactions, such as grabbing and throwing an object.

Additionally, it includes the functionality to give the user movement, either in the space delimited for the simulation (in the Steam VR calibration section) or through the use of "teleportation" points.

However, the Steam VR translation layer is limited to the pre-made options provided by Valve developers, which is why it's necessary to use a second translation layer, which allows for a larger number of inputs and hardware. In this case, there is an option created by Unity and independent developers called OpenXR and OpenXR Interactions. With the integration of these 3 plug-ins (Steam VR, OpenXR, and OpenXR Interactions), the developer is able to make use of the integrated prefabs and provide compatibility to other VR systems without having to modify the codes or inputs already established.

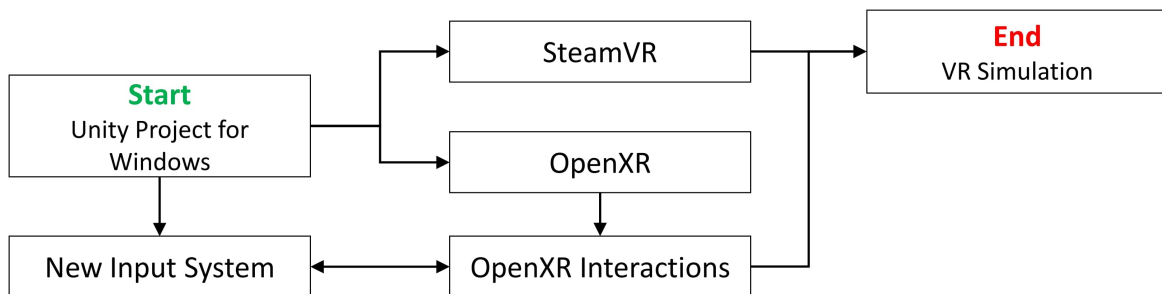


Figure 4.10: VR Translation pipeline

## User Test & Controls

### Input translation

The use of both OpenXR Interactions and Steam VR plugins provides the developer with the option to use two different input systems. First of all, OpenXR Interactions has native integration with Unity's input system, so its integration can be done by modifying the current input system file and binding the controls to the default actions in the "new input system." This has

the advantage that it can be used with any type of VR control, as long as the buttons used in the input manager exist.

In the case of Steam VR, it has native support and comes with its own input system. However, as of the 2.0 update in 2019, this system is no longer linked to commands accessible by the developer but instead uses an action callback system. This implies that the default actions (or the custom or preset profile being used) dictate what types of values can be received. This allows for better control of the actions that the user can perform; however, it makes the developer's work more difficult.

For the development of the application, both systems are used: creating the bindings in the input system for the OpenXR (for greater compatibility with other headsets) and calling the actions with the SteamVR (for better compatibility with the HTC Vive Pro and Valve Index).

When developing inputs for SteamVR, it is necessary to incorporate them into the code and not through a graphical interface, as is the case with OpenXR. This can present compatibility problems with the new input system, but by making use of functions like "or" (represented in C# as "||"), the desired result can be achieved. **These translation layers implemented in the code are what allows to solve problem #9.**

### Player movement & Interactions

Both plugins are capable of providing some freedom of movement with the headset. In the case of OpenXR, it allows the user to "teleport" to demarcated points in the scene using the controller's own data set. This is somewhat limiting, but it is considered the best system due to the different specifications and features that VR headsets can have.

In the case of SteamVR, it has integration with the Lighthouse tracking system, which can be used to do a room setup. With this option, a virtual space can be created in which the user can move, and this movement will be represented within the simulation. This also includes the teleportation system, with the addition that the developer can create entire teleportation areas where the user can be positioned in any way, not just at specific points.

One of the most important goals for a true implementation of VR technology is to provide the user with the ability to interact with objects. In this case, both systems work independently. The OpenXR plugin supports integration with the new input system of Unity, so the developer must manually create the bindings for each one of the VR systems using the graphical interface of the input manager.

SteamVR, with its prefabricated options, provides the developer with the necessary scripts to handle the most basic actions, such as grab, hold, and throw. This is the most straightforward method for creating interactions with SteamVR; however, it can generate compatibility or functionality problems with more complex options.

### User test

To corroborate the functionality of the simulation (specially in VR) and get feedback on the functions, controls, and bugs, it is necessary to test with users. For this, users with experience in both VR and application testing were used in order to test the different mechanics implemented in the project.

In this case, an alpha test is performed, which is defined as "is the initial phase of validating whether a new product will perform as expected" [46]. In the case of this application, the alpha testing was performed in the same XR laboratory of the University of Anhalt using the equipment described above.

## Morphological Boxes

The morphological box for the VR Integration is presented below. As mentioned in previous sections, this is a tool that aims to explain the decision making process performed in this section.

Table 4.4: Morphological Box for the VR Integration section

	Identifier	Situation	Solution #1	Solution #2	Solution #3
VR Implementation	1	VR System	HTC Vive Pro	Valve Index	Oculus Rift
	2	Unity VR Plug-in	SteamVR	OpenXR	-
	3	Interactions	Default interactions with that come with the plug-in	Custom interactions	-
	4	User movement	Teleporting	Walking	Both
	5	User aid	Create own visual/audio aid	User controller hints	Do not give hints
	6	Teleportation zones	Give the user complete freedom, so a teleportation area as big as the simulation	Teleport only to specific points demarked in the simulation	Teleport to only specific areas demarked in the simulation
	7	Menu interactions	Activated with pointers	Activated with hands	Both
	8	Interactables	All the objects all the time	Only objects relevant to the step that is supposed to be completed at a time	-
	9	User collisions	Control vibration	User warning	No warning
	10	VR Button implementation	Button pressable, text is not pressable	Button and text pressable	Just pressable text, no button

- **Solution #1:** As previously mentioned, due to the selection of available equipment and the versatility of the better display and tracking system, the HTC Vive Pro was chosen as the primary VR system. Additionally, it should be noted the use of the Valve Index, a HMD compatible not only with the same controls and lighthouse system but, for all practical purposes, the same VR headset as the HTC Vive Pro (even SteamVR recognizes it as an analog to the HTC system).
- **Solution #2:** The 2 most used plug-ins for VR application development in Unity are SteamVR and OpenXR. Both have their respective advantages: on the one hand, SteamVR has the support of Valve and its developers, so the tools embedded in the tool are extremely robust, although with little flexibility for changes. OpenXR is the opposite case. Developed by Khronos Group in 2017, it is an open-source standard to which all necessary modifications can be made by the developer. This provides compatibility with a large number of devices, which is why it is chosen to work with both plug-ins. Although the main focus is on a single VR device, the incorporation of OpenXR into the project allows it to be adapted for a much wider range of devices in the future.
- **Solution #3:** The layout of the controls in SteamVR comes as a default, based on the most common interactions and how they have been implemented in the most popular VR applications. Taking into consideration that not all the target audience of the application has had previous contact with VR technology, it is determined that the optimal way to implement the controls is to follow the convention of similar applications already established in the market.
- **Solution #4:** One of the biggest advantages of using the HTC Vive Pro as a reference HMD is the possibility of using the Lighthouse system. This system allows the tracking of the user's position in a given area. This allows a deeper immersion than the traditional teleportation system, in which the user moves in the VR scene only by pointing to the place where he or she wants to move. However, there are undeniable advantages to this teleportation system, so both systems were chosen, giving the user the ability to interact

with the simulation as they would in the real world, taking into consideration that even the teleportation method exists, which helps with the space limitations of the simulation.

- Solution #5: To provide help for inexperienced users, it was opted to incorporate explanatory diagrams, signs, and texts with the different functions that the controls offer within the simulation. Even though SteamVR offers so-called ControllerHints, these are only activated when the user spends 20 seconds or more without performing any action or when he or she is in front of an object with a specific set of assigned actions. This system was considered not ideal since it is not a permanent reference that users can consult at will.
- Solution #6: As mentioned above, teleportation is a very useful tool in VR simulations. However, this can induce errors in newer users, who may unintentionally teleport to unwanted zones by accident. For this reason, it was chosen to implement so-called teleportation zones, which are an asset of both SteamVR and OpenXR and designate areas to which users can teleport. These are not to be confused with teleportation points, which are just specific points of movement, while zones allow for more flexibility and less precision on the part of the user. Making use of these zones in the parts of the simulation that are of interest while still giving the user the opportunity to physically move in space (making use of the aforementioned lighthouse system) provides a good balance between guidance and restriction with freedom of movement for the user.
- Solution #7: The use of buttons in the software is of great importance, as it allows the user to move between scenes, reset the audio or the step being followed, and access information. One of the most traditional methods in applications is the option of using pointers. These are virtual extensions of the controls, which allow interacting with objects from a distance. However, to reduce the level of complexity and avoid confusion among new users, it was decided to use buttons that interact with virtual hands. This is because the trackpad (used for the teleport system) and the pointer work in a similar way, so in preliminary tests with new users, it was shown to increase the level of complexity and adaptation for such users. Additionally, due to the nature of the simulation, it was decided not to give the user the option to perform the procedures remotely, since this would defeat the purpose of the learning experience.
- Solution #8: In order to provide the user with a learning experience and avoid unpredictable or undesired interactions between objects that may cause bugs or take the user's attention away from the purpose of the simulation, it was decided to make use of specific steps. This means that the user cannot advance in the simulation (and therefore interact with other objects) until the current step is completed. **The implementation of these restrictions is what makes it possible to solve problem #10, as listed in section 4.2.1.**
- Solution #9: To provide realism to the interactions SteamVR has the option to use the haptic system integrated into the controls in the event of collisions. Making use of this option and the colliders integrated into the objects inside the simulation, a system was obtained where the user is warned through another sense that he is in contact with an object without having to interrupt the view of the object.
- Solution #10: One of the most salient points in preliminary testing with users unfamiliar with VR technology was the interaction with text and buttons. Most of these new users concluded that they instinctively wanted to interact with the text that referred to the button action rather than the button itself. It is for this reason that it was opted to integrate the interaction script and button actions into both the buttons and the text that accompanies them. For this, it was needed to implement the game's object plane, incorporate the text without a background in PNG format into it as a new material, and



use the Cutout option instead of using the Text option in a canvas, as it would be done in the traditional way. This results in the user being able to interact with both objects and get the same result.

#### 4.2.11 Simulation Layout

In order to provide the user with an educational and immersive simulation, it is necessary to have all the elements that are present during the actual procedure. The simulation itself takes place in multiple scenarios, such as the operating room, where all the pre-operative procedures will be performed, the internal surgery, where the actual surgery will take place, and more. In this section, the general layout of the simulation will be explained, as well as the reasoning behind the decisions made.

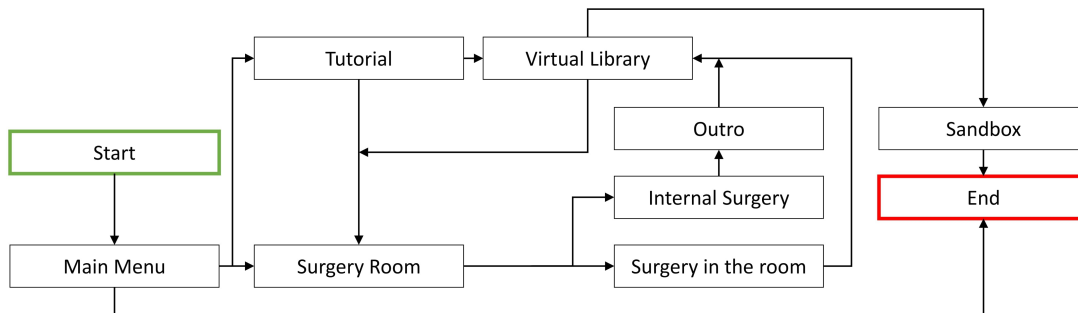


Figure 4.11: Scene Layout for the simulation

#### Simulation Scenes

The simulation begins with the start and main screen scenes. This division is due to the problems encountered in the early development of the VR application, where performance leaks were found when the player was started directly in the Main Screen scene. Later, it was found that starting the program from a scene with fewer objects largely neutralized the problem. It was not until the addition of Index Valve to the project that it was found that the root cause was due to the drivers implemented in the HTC Vive Pro. However, it was decided to keep this layout, as it gives the user the opportunity to interact in the first instance with a simple environment and allows the program to properly load complex scenarios.

There are three options for the main screen scene that the user can choose from: Start the simulation, skip the tutorial (this is for more advanced users with VR experience), and exit the program. In this scene, the user is not given the option to move around the surgery room in the back since its purpose is only as an aesthetic background at this point.



Figure 4.12: Start screen & Main Menu view from the VR Headset

The tutorial scene is intended to teach the user the basic interactions necessary for the simulation. To familiarize the user with the controls, a scenario was created where a cube and a sphere are placed on a table on the left side of the room and must be placed on the right side of the room in a specific place, demarcated by the shapes of the objects in the correct position. The shapes used are the simplest objects with colliders appropriate to their shape, so as to make the experience as easy as possible for the user. Here it was decided to make use of the functions of teleport and grabbing objects, which are necessary for the whole simulation. It is important to note at this point that the audio guide mentioned earlier in this document is implemented with the instructions for the tutorial and all the steps of the procedure, including the explanation of how the controls work.

The next scene is demarcated as the surgery room. In this scenario, all preparations for gallbladder removal are performed. Along with the audio guide, the user learns about the steps prior to performing the cholecystectomy as well as key concepts of the instruments used. In this section, it was decided to provide the user with a certain level of freedom of movement. Using the Teleport function and the prefabs associated with it, areas of interest for teleportation (near the objects with which the participant can interact) were determined.

The surgery tools are able to be placed in the positions demarcated by the guides given in the scene. In case of any misalignment in the correct position of the instruments, the tools will go back to their original position, giving the user the option to try again. This was designed to not discourage the learning experience of the users and to allow them to try more than once without the need to restart the scene. It should be noted that both the tutorial and the surgery room sections have a "reset scene" option located at the exit of the virtual room. This is because it is expected that less experienced users may have some difficulties and want to perform the procedure again. The guidance and steps indicated in this and subsequent sections are based on the theoretical section on performing a cholecystectomy.

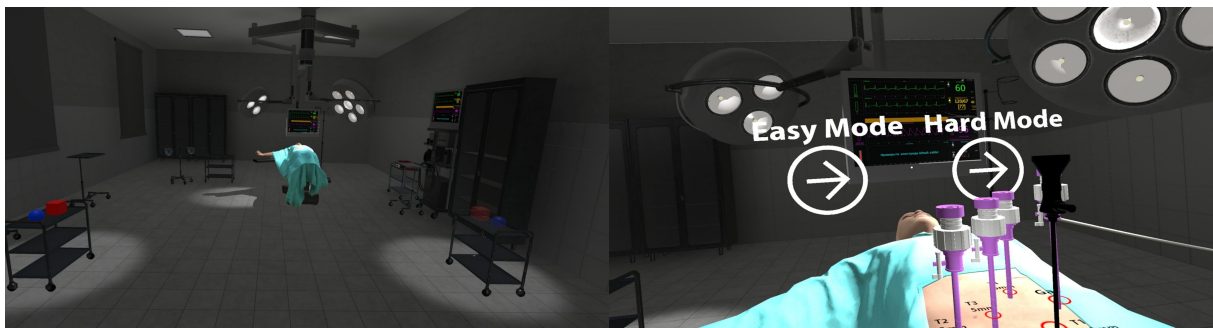


Figure 4.13: Tutorial & Surgery Room view from the VR Headset

After completing all of the necessary cholecystectomy preparations, the user is given the option of continuing with the easy or hard mode. It should be clarified that the simulation is intended for students outside the health area, but in careers that are linked to it. With the previous point clarified, the easy mode is offered, which is specially designed for people with little experience (or none) in VR and who wish to learn about the procedure without experiencing major complications. In Easy mode, the user is transported inside the human body on a 5:1 scale, i.e., the organs and instruments are considerably larger and therefore easier to handle. These instruments are linked to the user's controls and are positioned so that they can only move within a certain range, in which all the anatomy of interest is located.

Once the user completes the objective of gallbladder removal, the user is transported back to the operating room (another scene called "Outro"), where the gallbladder has already been removed and is just waiting for the user to discard it.

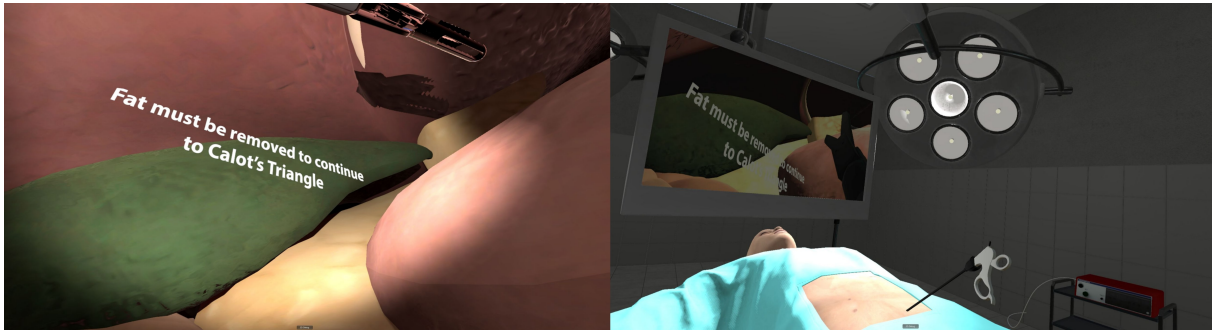


Figure 4.14: Easy Mode & Hard Mode view from the VR Headset

In the case of more experienced users, there is the scene in "hard mode," in which the user uses the tools that would be used in real surgery, having to place herself or himself in the correct position on the patient, taking into account anatomical knowledge, and locating the organs of interest with the video feedback received from the endoscope. There is a guide on the area in which the user with his/her instruments must access the device; however, no restrictions are offered, so the user must apply his/her knowledge to be able to perform the procedure. As in Easy mode, once the gallbladder is removed and properly disposed of, the user is offered the option to move to the virtual library.

In the virtual library, the user is presented with the six most relevant concepts of the procedure, from the perspective of a biomedical engineering student. It is for this reason that it was decided to divide the scene into two sections of knowledge: one dedicated to the tools used and the other to anatomy and the procedure itself. Here, the user has the option to use the buttons in the same way as during the whole simulation to access a small presentation of the most important theoretical concepts of the selected option. Additionally, there is the display of a 3D model of the selected tool or anatomical element. However, user interaction with these objects has been limited, as these elements can distract the user from the information provided in text form.

## Morphological Boxes

The morphological box for the Simulation Layout is presented below. As mentioned in previous sections, this is a tool that aims to explain the decision making process performed in this section.

Table 4.5: Morphological Box for the Start, Main and Tutorial Scenes

	Identifier	Situation	Solution #1	Solution #2	Solution #3
Start Scene	1	Player Initialization	One player per scene	One player for the simulation	Initialize and deactivate players when they get into a scene
Main Screen	2	Skip the tutorial	Give the user the option to skip it	Don't give the user the option	-
	3	Background	Black background as in the Start Scene	Custom Background	Same Surgery Room as the rest of the simulation
Tutorial	4	How to implement it?	Give the user a visual guide of the controls	Let the user experiment on her/his own	Create a simple task to be performed, using the same controls as in the simulation
	5	Which shapes ot use	Simple geometry, to make it easier to interact with them	The same instruments as the ones used in the procedure	-
	6	Guide	Visual guide, using arrows and schemes	Audio guide, explaining the controls and what to do next	Both options
	7	Mistakes	Give the user the freedom to make mistakes and reset the scene	Move the objects back to the original position, without the reset button	Don't let the user make any mistakes, by not letting the objects be moved to any other place

- Solution #1: In early versions of the simulation, a single game object player was used for the entire simulation; however, this led to a number of complications, including bringing the objects the user interacted with in one scene to another. These complications were due to the way SteamVR handles the scenes and the player in question. For this, SteamVR creates a scene within the scene called "don't destroy on load,", in which it puts the game object player and all the objects he interacts with, and as soon as a scene change happens, it simply transfers this temporary scene to the new scenario. Within the developer options, there is a code that is accessible and modifies this behavior, so a lot of the associated bugs were eliminated by simply modifying one player per scene.
- Solution #2: For more advanced users or recurrent users, a skip the tutorial option was created. This is in order to provide more flexibility and give participants options according to different levels of experience and comfort with the use of VR technologies.
- Solution #3: It was decided to use the same operating room in this and the other scenes as a background to familiarize the user with the scene where most of the simulation takes place. It should be noted that in some scenes (especially those with menus), the functionality of the objects that can be observed is limited, or the areas that the user can teleport to are limited.
- Solution #4: For the creation of the tutorial, it was decided to guide the user on the basic interactions to be used in the simulation itself, as well as to give clear visual and auditory guides on the tasks to be performed. One of the most important additions for all the sections with specific tasks is the creation of a whiteboard with all the tasks to be completed, the status of whether they have been completed, and the possibility of restarting the scene or just the selected step. With these elements, it is decided by the approach whether to guide the user through each step to complete both the tutorial and all the scenes successfully. However, there is in the educational area the concept of learning by doing, so within the same layout of the simulation, the user is given the opportunity to explore in depth all the elements available in the simulation in a specific scene called "the sandbox".
- Solution #5: Especially for participants unfamiliar with this type of technology, it is important that the first approach be as simple and concise as possible. This is why it was chosen not to introduce users to the surgical elements during the tutorial but to use simple geometries that are easy to interact with in order to make the familiarization process more effective.
- Solution #6: For both the tutorial scene and the rest of the simulation, it was decided to use both visual elements and an audio guide to orient the user on the tasks to be performed and how to perform them. It is important to note that the use of these elements individually has merit, but due to the different types of intelligence that exist and how they can vary from one participant to another, it is necessary to cover as many bases as possible.
- Solution #7: Users are given the option to make mistakes that are not self-correcting during the tutorial. This is in order not to omit part of the learning process, where making mistakes is not only an inevitable part of the process but also helps the user better understand how to perform tasks correctly in the future. However, this can lead to unwanted interactions between 3D objects, so the user is also given the option to reset the scene and return to the initial state.

Table 4.6: Morphological Box for the Surgery Room, Easy and Hard Mode Scenes

	Identifier	Situation	Solution #1	Solution #2	Solution #3
Surgery Room	1	Instrument Location	Instruments appear according to the step being performed	Instruments placed as in real life surgery	Instruments available all the time on the same side
	2	Gas Pressure Adjustment	Rotation	Buttons	Typing
	3	Instruments Position	Objects can be misplaced, but the user won't be able to advance to the next step of the procedure	Object can only be placed on the right spot, otherwise the position of the instrument resets.	-
	4	Completed Task	Audio signal that the task has been completed	Visual guide that the task has been completed	Both audio and visual indicators that the task has been completed
Easy Mode - Surgery	5	Instructions	Large and demarcated instructions of where the actions should be performed	Small and demarcated instructions of where actions should be performed	No instructions, relying only in the user anatomy knowledge
	6	Organ physics	Realistic organ physics	Only relevant organs contain the physics scripts	No physics scripts implemented
	7	Gallblader Cut	Just one cut, in the right area	Just one cut, but the user can do a mistake. Scene can be reseted	Multiple cuts. Scene can be reseted.
	8	Hand position	Free hand position	Restricted with certain freedom hand position to get closer to the righth position where actions should be performed	Complete hand position restriction
Hard Mode- Surgery	9	Instructions	Large and demarcated instructions of where the actions should be performed	Small and demarcated instructions of where actions should be performed	No instructions, relying only in the user anatomy knowledge
	10	Organ physics	Realistic organ physics	Only relevant organs contain the physics scripts	No physics scripts implemented
	11	Gallblader Cut	Just one cut, in the right area	Just one cut, but the user can do a mistake. Scene can be reseted	Multiple cuts. Scene can be reseted.
	12	Hand position	Free hand position	Restricted with certain freedom hand position to get closer to the righth position where actions should be performed	Complete hand position restriction

- Solution #1: To increase the realism of the simulation and provide the user with a deeper understanding of how procedures are performed on real patients, it was chosen to incorporate the surgical elements as they would be located in real life. This brings with it a complication that is not present in real life; in the simulation, the procedure is performed without the usual surgical assistants, who are in charge of some of the functions that the simulation participant has to perform by himself. It is for this reason that some of the equipment is located to the right and left of the patient, as it would be in real life, with the surgical assistant in charge of the secondary equipment and the surgeon in charge of the primary equipment. However, this is preferable, as it allows the user to become familiar with all the equipment involved in the procedure.
- Solution #2: Adjusting the insufflation pressure is one of the most important steps in the preparation for the cholecystectomy. This being the case, in real life, a  $CO_2$  tank is used, connected to the insufflator, which acts as a valve and regulates the flow and pressure at which the  $CO_2$  is delivered. This is done by opening the tank valve and using the buttons located on the front panel of the insufflator. In the case of the simulation, it was decided to use only one of the elements, which is the pressure control through the valve in the tank. For this purpose, the side buttons of the control and the trigger were used, since both resemble the natural position of the hand when interacting with a valve. This option was chosen over alternatives such as using buttons or typing the desired pressure since: it is closer to the real procedure; it provides attention to the user about the pressure regulation problem; if the pressure is too low, the simulation cannot continue; and if it is too high, the insufflator provides an alarm to call attention to the user about the potential risks of using that pressure.
- Solution #3: As mentioned earlier in the tutorial section, one of the biggest problems of letting the user position objects anywhere is the deviation of attention from the task at hand and the increase of potential bugs due to unplanned interactions. It is for this reason that the decision was made to only let the objects be placed in the appropriate positions

according to the procedure. Additionally, a unique script was used on each object to return it to its initial position in case of misplacement, letting the user know of his mistake and allowing the participant to repeat the task correctly.

- Solution #4: To alert the user that the current task has been completed, the aforementioned whiteboard and a sound effect were used. Using the audio, the user can know that the current step in the procedure has been completed without having to return his attention to the whiteboard, but always has the possibility to corroborate the same with access to the task list.
- Solution #5: The Easy Mode is oriented toward users with little experience using VR technology as well as users unfamiliar with the anatomy of the procedure. In this scene, the user is inside the abdominal cavity with the organs in a 5:1 ratio, so the size of the organs is considerably larger than in real life. This being the case, to provide more support to the user, all the guides are enlarged, and methods such as arrows are used to indicate the elements to interact with.
- Solution #6: Due to the closeness that the participant has with the organs in Easy Mode and the ease of rendering fewer elements like those found in the surgery room, the above-mentioned YuePhysics scripts are used in this scene. These allow to add more realism to the organs since it allows that the meshes of the objects acquire elastic proportions when positioning colliders in the vertices of the mesh of the object.
- Solution #7: In this mode, the user can perform a single cut, although mistakes can be made. This was developed in this way to draw the user's attention to the care that must be taken by surgeons, who usually only have one chance to perform the cut correctly. However, the opportunity is also given (as in the previous scenes) to restart the scene and try again. To further assist inexperienced users, the cut can only be performed in a delimited section (in the vicinity of the Carnot triangle); however, this does not mean that it can only be performed in the correct position.
- Solution #8: One of the major advantages of using the SteamVR plug-in is the access to some functions pre-programmed by Valve developers. In this case, it refers to the movement restriction implemented in the Easy Mode scene. It was chosen to make use of this function to help users not deviate too much from the correct position of the hands to perform the cholecystectomy procedure. It restricts the movement of the hands to only 35 degrees out of the optimal hand position, even when the user physically rotates, so that the user must be in the correct position to perform the procedure.
- Solution #9: Unlike what is implemented in Easy Mode, in Hard Mode the surgery is performed as it would be performed in real life, i.e., making use of the endoscope to observe the inside of the patient on a screen. In this case, there are still instructions; however, it was chosen not to draw the user's attention in the same way as in the Easy Mode to test the participant's anatomical knowledge.
- Solution #10: Due to the physical remoteness of the user and the organs, the fact that the user can only observe them through a screen, and the fact that the same scenario is used as in the surgery room, it was decided to implement the YuePhysics scripts only in the organs of interest and in a limited way, i.e., reducing the number of vertices and therefore the number of micro colliders implemented in them. This helps to reduce the load imposed on the GPU without compromising the user's perception of realism.
- Solution #11: Due to the level of realism implemented in the Hard Mode with respect to the real procedure, it was opted to give the user the possibility of making mistakes in the

same way they could be made in real life. As in other scenes, the whole scenario can be restarted if mistakes are made.

- **Solution #12:** To add realism, the user has complete freedom of placement of his hands and surgical instruments in this mode. However, due to the disparity that may exist in anatomical knowledge among users, guidelines were implemented for the approximate position that the participants should use in order to complete the surgery successfully.

Table 4.7: Morphological Box for the Virtual Library and Sandbox Scenes

	Identifier	Situation	Solution #1	Solution #2	Solution #3
Virtual Library	1	Button Interaction	The information is shown when only the button is pressed	Both the text and button can be pressed	-
	2	Information Available	Only the most relevant information is available	All the information tied to the procedure is available	Only the information relevant for the biomedical engineers is available
	3	3D Model Interaction	The user can interact with the 3D models used	The user only has limited interaction with the 3D model used	No interaction with the 3D model, as it is only for display
SandBox	4	Objects Available	All of the objects used in the simulation	Only the surgery elements	No objects present
	5	Object Interactions	All the standard interactions used in the simulations	No interactions tied to the objects	-

- **Solution #1:** As done in the tutorial section, both the text and the button located to the right of the text are now interactive. This is because in the preliminary tests at the end of the simulation, most of the users with less experience in the use of VR applications reported that it was confusing for them not to be able to interact with the text and only with the button. In order to facilitate interactions and avoid confusion, the same script used in the button is used in the text, enabling it as an interactable.
- **Solution #2:** Since the target audience of the application is biomedical engineering students, all relevant concepts are included in this section. In the same virtual library, it is divided into the tools section (left column) and the anatomical and physiological concepts section (right column). Within the highlighted information (mainly in the information associated with surgical instruments) are the most commonly used settings, as well as the special features of some instruments that make them the tools of choice for surgeons.
- **Solution #3:** To avoid diverting the user's attention from the information presented, it was decided not to make available user interaction with the 3D objects displayed. The latter are only present for illustrative purposes.
- **Solution #4:** The sandbox scene is designed to let the user interact freely with the VR simulation. The objective is for the participant to try all the options and freedoms offered by the technology and explore his environment in a way that the main simulation does not allow. This is why it was decided to place all interacting objects in this scene, where the user will be able to observe them in more detail and interact with them.
- **Solution #5:** Following the logic presented in Solution #4, the scripts implemented in the objects during the scenes are not used. Within them are integrated the functions of correct positioning and resetting the position in case the final location is not indicated, so they were removed in their entirety for this scene so that the user can experiment with the objects freely.

### 4.3 Closing Remarks

One of the most important points of this section is the approach of the study that will be the subject of study in the following sections, since this will ultimately be the tool with which it will be possible to measure how effective the simulation was as an educational tool.

Additionally, the pipeline used for the creation of the simulation is created. This can be observed in the image in the appendix 7, Appendix A. The process depicted in this diagram involves a wide range of different programs and skills, such as 3D modeling, graphic design, and code development. The developer has to be flexible to work with these different programs and skills in order to achieve positive results.

This methodology section aimed to explain the concepts used for the realization of this project at a macro level, so there are some minor details that are not represented here. Nevertheless, the appendices section provides more detailed descriptions of the most relevant codes and algorithms applied in this section. These descriptions help to better understand how the project was executed and can also serve as a guide for future projects related to or based on this project.



# Chapter 5

## Results & Analysis

This section will show the results obtained both from the user tests and study case carried out during the period from February 20th to March 17th, 2023. Also included in this section are the analyses derived from these results. Special emphasis will be placed on the implications of the results obtained from the participants in the study and their significance with respect to the usefulness of the tool and the variables that affect it.

### 5.1 User Test Results

As mentioned earlier in the methodology section, alpha tests were performed before starting the actual study to obtain feedback on the simulation and improve the experience prior to the case study. For this preliminary study, the participants were students and administrators from the XR Laboratory at the University of Anhalt. Some of them had extensive experience in developing and testing VR applications, while others are newer users of this type of technology, but always with developer knowledge.

In these first tests, the following bugs were found and solved before the case study: It is worth noting that the bugs pointed out are the most relevant.

- The user could pass through solid objects, such as walls.
- Controls with inaccurate responses.
- Unclear instructions or guidance for the user.
- Controls with different functionality than standard VR applications
- Objects lose properties if dropped in random places.
- The user is not able to move by any method other than teleportation.
- The user "drags" with himself the objects that interacted in the previous scene.
- The game object "Hands" disappears or loses precision if multiple actions are performed at the same time.

All these bugs were fixed in the code implemented in the objects. These can be seen in Appendix F. Once the correction of these bugs was confirmed, the case study could be performed.

## 5.2 Case Study Results

### 5.2.1 Questionnaire Results

The results obtained from the case study can be seen below. For a better understanding, a binary system is used, where the correct answer has a value of 1, and the incorrect answer has a value of 0. As mentioned before, no personal data was taken from the participants, so if reference is made to any of the specific results, it will be done through the numbers assigned to the participants. The raw data is included in the appendix section.

Table 5.1: Study case questionnaire results in binary form

Participant	Questions										
	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11
1	1	1	1	1	1	1	1	1	1	0	1
2	1	1	1	1	1	1	1	1	0	1	1
3	1	0	0	1	1	1	1	1	1	0	1
4	1	1	1	1	1	1	1	1	0	1	0
5	1	1	1	1	1	1	1	1	1	1	1
6	1	1	0	1	1	1	1	1	0	0	0
7	1	1	0	1	1	1	1	1	0	1	0
8	1	1	0	1	1	1	1	1	0	0	1
9	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	0	0	1
11	1	1	1	1	1	1	1	1	0	1	1
12	1	1	1	1	1	1	1	1	0	0	1
13	1	0	0	1	1	1	1	1	1	0	1
14	1	1	1	1	1	1	1	1	0	1	0
15	1	1	0	1	1	0	1	0	0	1	0
16	1	1	0	1	1	1	1	1	1	1	1
17	1	1	1	1	1	1	1	0	0	1	1
18	1	1	1	1	1	0	1	1	0	0	1
19	1	1	1	1	1	1	1	1	0	1	1
20	1	1	1	1	1	0	1	1	1	1	1
21	1	1	1	1	1	1	1	1	1	1	1
22	1	1	1	1	1	1	1	1	0	1	1
23	0	1	1	1	1	0	1	0	0	0	1
24	1	1	1	1	1	1	1	1	1	0	1
25	1	1	1	1	1	1	1	1	0	0	1

Table 5.2: Summary of the study case results

<b>VR Participants</b>	13			
<b>Non VR Participants</b>	12			
<b>Question</b>	<b>VR Correct Answers</b>	<b>VR Wrong Answers</b>	<b>Non VR Correct Answer</b>	<b>Non VR Wrong Answer</b>
#1	13	0	11	1
#2	11	2	12	0
#3	10	3	8	4
#4	13	0	12	0
#5	13	0	12	0
#6	13	0	8	4
#7	13	0	12	0
#8	12	1	10	2
#9	7	6	2	10
#10	6	7	8	4
#11	13	0	7	5

As a summary, table 5.2 collects the information presented in table 5.1. This collected data will be analyzed in depth in Section 5.3.1. Furthermore, some of the most standout comments from the participants of the study are included below, as are problems and bugs encountered during the simulation.

- Participant #3: "There was a bug, where one of the objects didn't stay in the right position after it was placed there."
- Participant #11: Despite not being part of the VR Group, the participant was a patient of the Cholecystectomy procedure.
- Participant #13: "There was a glitch where the T3 trocar wouldn't stay in place, even after placing it in the right place. Also when the trocar finally stuck in a position, it was slightly off from the position marked on the patient."
- Participant #21: "The teleportation system was difficult at first, because when pressing the trigger to grab something, I would press the trackpad at the same time."

### 5.2.2 Relevant Variables

Table 5.3 shows the variables mentioned in the methodology section 4.1. These will be analyzed in later sections, along with their relevance with respect to the results obtained from the study.

Table 5.3: Relevant variables per participant

Participant	VR Control Group	Previous VR Experience	Academic Level	Field of Study	Right Answers
1	Yes	Yes	Undergraduate	Biotechnology	10
2	Yes	No	Undergraduate	Computer Science	10
3	Yes	Yes	Undergraduate	Computer Science	8
4	No	No	Graduate	Biomedical Engineering	9
5	Yes	Yes	Graduate	Biomedical Engineering	11
6	No	Yes	Graduate	Biotechnology	7
7	No	No	Graduate	Molecular Biotechnology	8
8	Yes	No	Graduate	Molecular Biotechnology	8
9	Yes	Yes	Undergraduate	Biotechnology	11
10	Yes	No	Undergraduate	Biotechnology	9
11	No	Yes	Graduate	Biomedical Engineering	10
12	Yes	No	Graduate	Biomedical Engineering	9
13	Yes	Yes	Graduate	Biomedical Engineering	8
14	No	No	Undergraduate	Biomedical Engineering	9
15	No	No	Undergraduate	Biomedical Engineering	6
16	No	Yes	Undergraduate	Computer Science	10
17	Yes	No	Graduate	Biomedical Engineering	9
18	No	No	Graduate	Biomedical Engineering	8
19	Yes	No	Graduate	Biomedical Engineering	10
20	No	No	Graduate	Biomedical Engineering	10
21	Yes	Yes	Graduate	Biomedical Engineering	11
22	No	No	Graduate	Biomedical Engineering	10
23	No	No	Graduate	Biotechnology	6
24	Yes	No	Graduate	Biotechnology	10
25	No	No	Graduate	Biotechnology	9

As a summary, table 5.4 collects the information presented in table 5.5. The relevance of these data will be analyzed in depth in the following sections.

Table 5.4: Summary of the relevant variables from the study participants

	Condition	Number of participants	Average
Previous VR Experience	Yes	9	9.55
	No	16	8.75
Academic Level	Undergraduate	17	9
	Graduate	8	9.125
Field of study	Biomedical Engineering	13	9.23
	Molecular Biotechnology	2	8
	Biotechnology	7	8.85
	Computer Science	3	9.33

## 5.3 Analysis & Discussion

### 5.3.1 Descriptive statistic analysis

In this section, the results obtained from the case study are presented. This includes analyses derived from the results shown in the previous section, with a particular emphasis on the participants performance during the questionnaire and the significance of it with respect to the tool's usefulness and its performance as an educational aid.

To interpret the data collected during the study, it is necessary to make use of statistics, more specifically, the descriptive type of statistical analysis. Descriptive statistics is a branch of this field that involves the collection, presentation, and interpretation of data in order to describe patterns or characteristics of the data.

It is for this reason that in the first instance, the data is organized in such a way that the percentages of each control group are visible. Table 5.5 shows this conversion to percentages, where the results are compared against the total population and against the same control group.

Table 5.5: Comparison of results between each group against the population, as well as each group against each other in percentages.

Questions ↓	VR Control Group				NonVR Control Group			
	Right Answers (vs Population)	Right Answers (vs Group)	Wrong Answers (vs Population)	Wrong Answers (vs Group)	Right Answers (vs Population)	Right Answers (vs Group)	Wrong Answers (vs Population)	Wrong Answers (vs Group)
1	52%	100%	0%	0%	44%	85%	4%	8%
2	44%	85%	8%	15%	48%	92%	0%	0%
3	40%	77%	12%	23%	32%	62%	16%	33%
4	52%	100%	0%	0%	48%	92%	0%	0%
5	52%	100%	0%	0%	48%	92%	0%	0%
6	52%	100%	0%	0%	32%	62%	16%	33%
7	52%	100%	0%	0%	48%	92%	0%	0%
8	48%	92%	4%	8%	40%	77%	8%	17%
9	28%	54%	24%	46%	8%	15%	40%	83%
10	24%	46%	28%	54%	32%	62%	16%	33%
11	52%	100%	0%	0%	28%	54%	20%	42%
<b>Average</b>	<b>45%</b>	<b>87%</b>	<b>7%</b>	<b>13%</b>	<b>37%</b>	<b>71%</b>	<b>11%</b>	<b>23%</b>

From the data shown before, it must be established that a total of 25 participants took the test, 13 of whom were in the VR group and 12 in the nonVR group, this meaning that if all the participants got the questions right, the distribution of the percentages would be 52% and 48%, respectively (only when analyzing the entirety of the population).

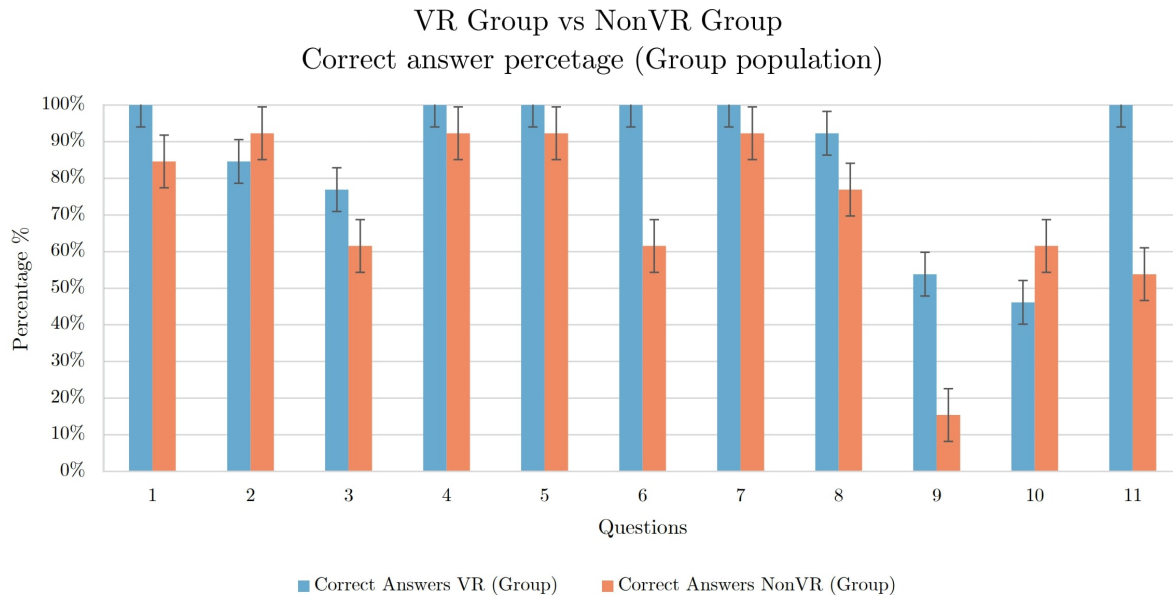


Figure 5.1: VR Group vs NonVR Group percentages comparison (Group Population)

From figure 5.1 it can be observed that there is an apparent disparity between the results, with the VR group showing higher percentages of correct answers than the nonVR group. This information seems to indicate that the results obtained by the VR group are better than those obtained by the nonVR group. However, in order to make this assertion, it is necessary to further analyze the data and the possible causes of these results. There are 2 main questions where this difference seems to be exacerbated: questions #6 and #9. Using the questionnaire created for this study (Appendix H), questions 6 and 9 are, respectively:

- Q6: What is the purpose of the trocar?
- A6: Generate access to the abdomen
- Q9: Where is the point where the cut should be made?
- A9: Calot's Triangle

To understand what happened in these two questions, it is necessary to observe the way in which the information was given in both cases. It can be inferred that in both cases the same phenomenon occurs since the information needed to answer both questions, analogous to how a regular master's class would be (both for undergraduates and graduates), is mentioned and pointed out but not presented graphically in the presentation, relying entirely on the attention of the participant. Thus, the functionality of the trocars (answer to question #6) is specified orally in slides 6 and 11, but not implicitly in the presentation.

For question #9, reference should be made to slide 8 of the presentation used for both groups. In this slide, the concept of Calot's triangle is introduced, pointing out in the presentation the points and mentioning the name because this is the point where the dissection of the gallbladder is performed.

In contrast, during the simulation in the gallbladder dissection section, the audio guide instructs the user to make the cut only in the Calot's triangle, and guides point out the exact position of the triangle, calling the user's attention at every opportunity and reminding the participant that the cut must be made in this area. This information points out that participants require graphical representations to retain information, which is consistent with current theories in pedagogy. This is also the case with the outliers that are questions #2 and #10 where the nonVR group outperformed the VR group. Here the relevant concept is also mentioned but not explicitly displayed this time during the simulation. More on outliers is discussed in Section 5.3.4.

While the non-VR group got 71% of the results correct, the VR group got 87%. These numbers are the average against the same population (i.e., between each group) so that the difference in population size is eliminated as a factor that can affect the average, as would be the calculation of the average across the total population. With a 16 % percentage difference between the populations, it is necessary to perform more tests to show whether this is sufficient to say that VR technology complements the master class scheme by testing if there's enough evidence to sustain this claim.

### 5.3.2 2 Sample-t Analysis

In order to find more evidence and better understand the results obtained from the case study and the descriptive statistic analysis, a 2-Sample t-test analysis is performed. This tool is a statistical method used to compare two independent groups. It is used to determine if there is a significant difference between the data of interest in the study.

To check if there is a difference between the means, there's the null hypothesis, which indicates that there is no significant difference; that is why the p-value is calculated. If this value is less than 0.05 (usually), the null hypothesis is rejected, and it is concluded that there is a significant difference between both means.

Minitab, which is the statistical software chosen for data analysis (see more in Section 4.1.4), was used to perform this analysis. Within the same program, the 2-Sample t study was used in both configurations, for the study of the mean and for the study of the standard deviation of the groups against each other. This software does a prerequisite test before attempting to do the analysis. One of the most important prerequisites is that the data obtained follows a normal distribution, which in the case of the data used it is, although the program did state that given the amount of participants, the normality of it could be improved by adding more data entries (for both the mean and standard deviation test). The results obtained from these tests are shown in figures 5.2, 5.3 and 5.4.

From the data and graphs provided in this section, it can be observed that there is evidence of a verifiable difference between the means of the two control groups. Within the data found in Figure 5.2, it can be observed that the p value is less than 0.1, which indicates that there is data to refute the null hypothesis and that the analysis of the data in terms of confidence intervals can begin. The p-value is a number calculated within the statistical tool that measures the strength of the evidence against the null hypothesis. The higher the p-value, the stronger the assertions can be made about the validity of the results. The following guidelines are followed:

- $p < 0.01$ : Really strong evidence against the null hypothesis
- $p < 0.05$ : Strong to moderate evidence against the null hypothesis
- $p < 0.1$ : Moderate to weak evidence against the null hypothesis (interpreted more as a trend).
- $p > 0.1$ : There is no evidence that the null hypothesis doesn't apply.

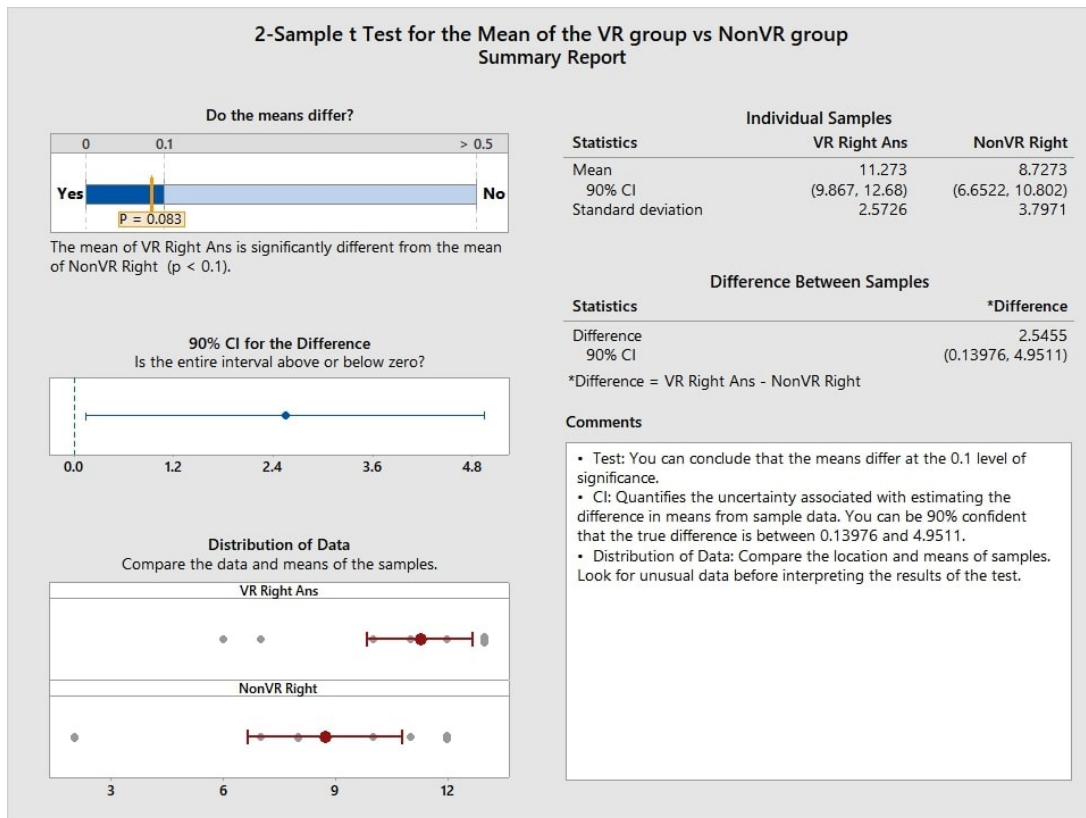


Figure 5.2: 2-Sample t Test for the mean difference between the control groups

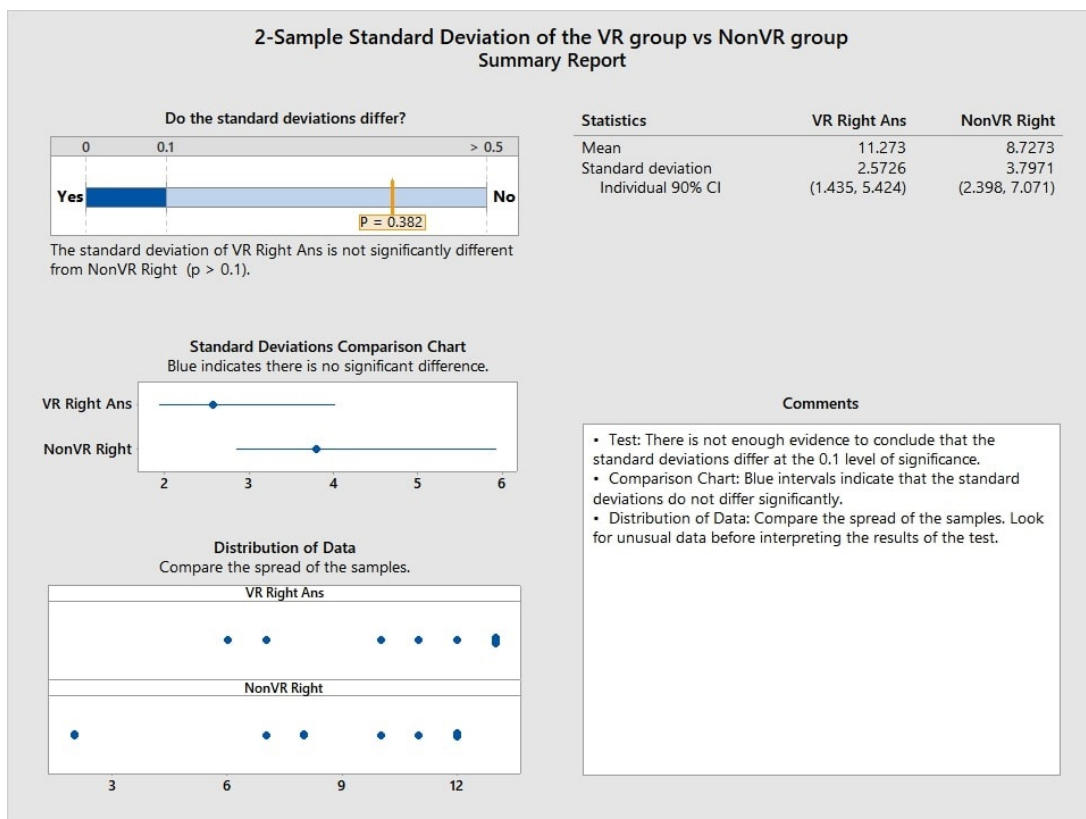


Figure 5.3: 2-Sample t Test for the standard deviation difference between the control groups

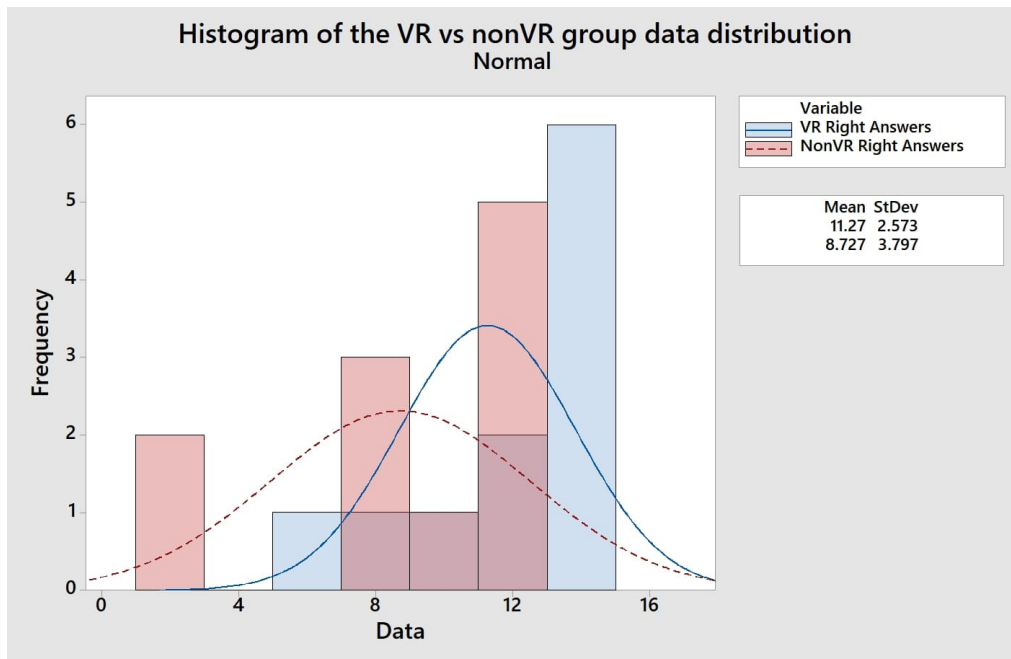


Figure 5.4: Histogram for the comparison between the VR and NonVR group data

The result obtained is a value of  $p=0.083$ , which falls within the 90% confidence interval for this test and within the category of moderate to weak evidence against the null hypothesis (interpreted more as a trend). This value has to be taken with the consideration that although normality was confirmed by the software, it did indicate that the accuracy of it was influenced by the small amount of data points. However, this value should be analyzed in conjunction with the totality of the study and the nature of the study, since (as stated before) the  $p$  value is dependant on the sample size.

In the case of the study, it can be considered a small sample size, so it can be interpreted that there are differences between the means of both groups because, even when the sample size is small, a  $p\text{-value} > 0.1$  is obtained. Nonetheless, the acquisition of more data entries (i.e., more participants) would have resulted in a better overall assessment of the results and their meaning regarding the tool's performance as an educational aid. This will be pointed in Section 6.1 along with the rest of the future recommendations for the development of more applications based on the simulation created.

In Figure 5.3, in the results of the 2-Sample  $t$  test for the standard deviation, it can be seen that, although in the previous image there was evidence that refuted the null hypothesis, in this case it cannot be inferred that the standard deviations differ significantly. In this test, the  $p$ -value is 0.38, which puts it well above 0.1, and thus there is not enough evidence to say that the results of the two groups differ, at least in terms of the standard deviations.

In analyzing these results, it is clear that there's evidence (although weak) that the VR group outperformed the other group in terms of correct responses. However, the data between the groups did not vary significantly, indicating that both groups had a similar level of variation between their results. As an analogy, this situation is similar to a dartboard where the darts in one group landed closer to the center than those in the other, but both groups had a similar level of accuracy overall.

**These results point to the effectiveness of this tool as an educational aid; however, it does not mean that VR technology can replace the current educational paradigm, but rather that it complements it and gives educators a tool with which to enhance student learning.**



### 5.3.3 Relevant variables

To get a better understanding of how the variables affect the results, data can also be obtained and analyzed from the information shown in the previous section. The data for the most relevant variables about the participants is processed and shown in Tables 5.6 and 5.7 (the raw data can be found in the previous section in Table 5.3).

Table 5.6: Cross reference data for the relevant variables

Field	Previous VR Experience	Academic Level	Number of participants	VR Control Group	VR Average	NonVR Control Group	Non VR Average
Biomedical Engineering	Yes	Graduate	4	3	10	1	10
		Undergraduate	0	0	0	0	0
	No	Graduate	7	3	10	4	9.25
		Undergraduate	2	0	0	2	7.5
Molecular Biotechnology	Yes	Graduate	0	0	0	0	0
		Undergraduate	0	0	0	0	0
	No	Graduate	2	1	8	1	8
		Undergraduate	0	0	0	0	0
Biotechnology	Yes	Graduate	1	0	0	1	7
		Undergraduate	2	2	10.5	0	0
	No	Graduate	4	2	9.5	2	7.5
		Undergraduate	0	0	0	0	0
Computer Science	Yes	Graduate	0	0	0	0	0
		Undergraduate	2	1	8	1	10
	No	Graduate	0	0	0	0	0
		Undergraduate	1	1	10	0	0

Table 5.7: Number of participants per field of study

	Undergraduate	Graduate	Total Participants
Biomedical Engineering	2	11	13
Molecular Biotechnology	0	2	2
Biotechnology	2	5	7
Computer Science	3	0	3

To better interpret the data from table 5.6, figures 5.5 and 5.6 are shown, where the subdivisions and the relationships between them can be observed graphically, making the division between the VR group and the nonVR group.

First, it can be seen that, for the VR group, the great majority of the participants belong to the biomedical engineering graduate program (52% of the participants). Since this is the most relevant field (the simulation was developed for the BME curricula) and the one in which the participants have the highest participation, it can be seen that, in general, the 2 control groups have the highest and most consistent average score among the participants.

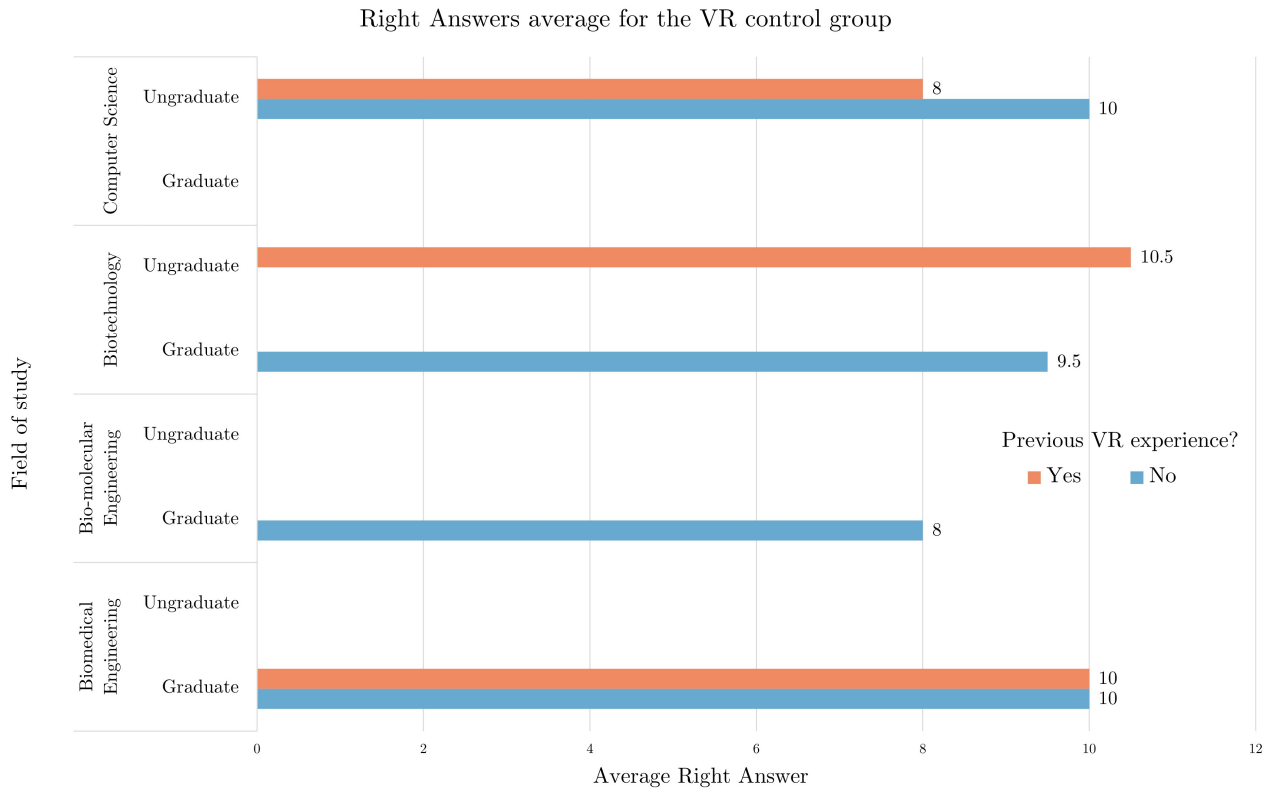


Figure 5.5: Crossed data for the VR control Group

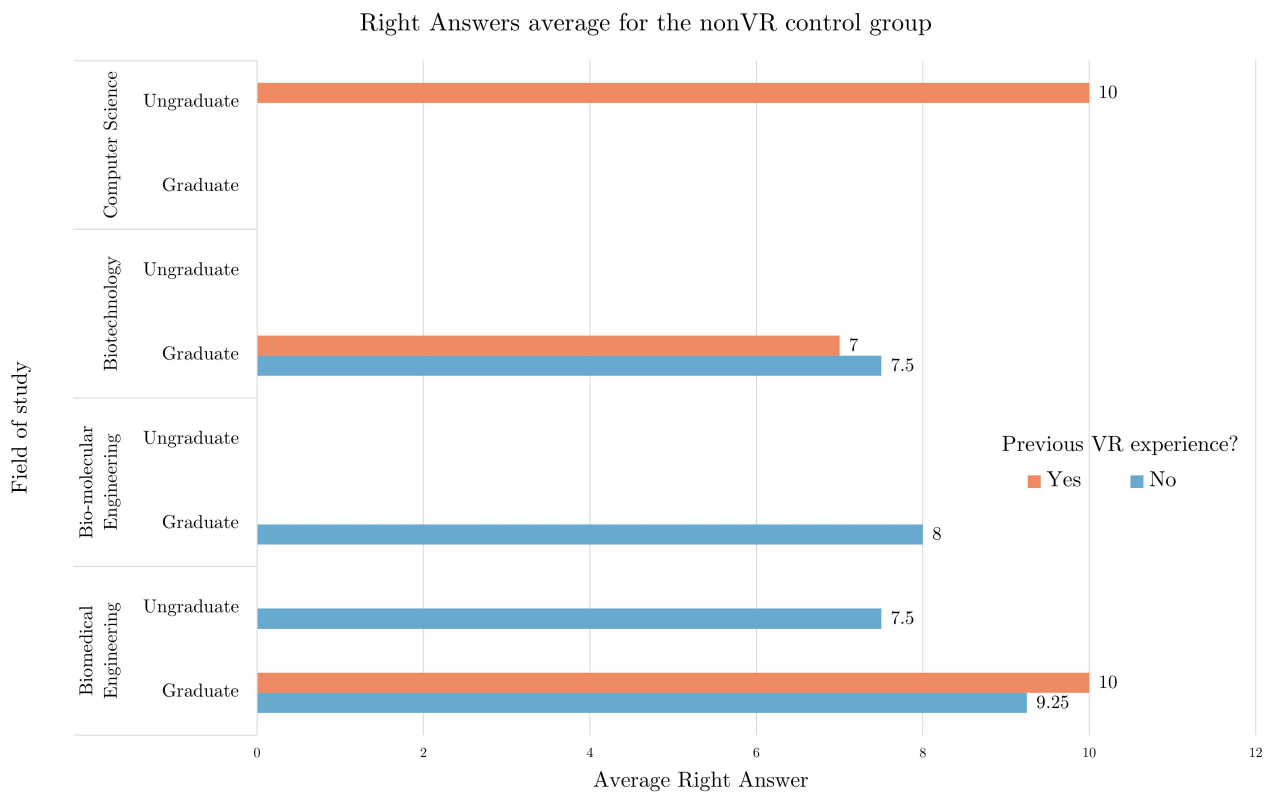


Figure 5.6: Crossed data for the nonVR control Group

The second largest group of participants are those from the biotechnology area, especially those who are graduates in this same career. The difference between the group of students who used the simulation and those who belonged to the other control group can be observed, with the average of those who did not use the simulation being 7.25 versus 10 for the biotechnology participants belonging to the VR group. This is corroborated by the data from the questionnaire. This implies a behavior similar to a trend among the groups, regardless of the area of study, except for the computer science participants.

As the second-smallest field of study, the data obtained from these participants has the greatest overall influence on the average results of this group. Two of the three Computer Science participants had extensive experience in the use of VR technology, so not only are the results obtained in the questionnaire above average, but they are also the participants who had the most comments after the simulation, and in general, those who performed the simulation in a quicker and more accurate manner.

Another point of view to analyze is the academic level of the students. Most of the participants are graduate students in one of the disciplines mentioned above (18 graduate students and 7 undergraduate students). With an average of 9 correct answers for undergraduate students and 9.125 for graduate students, it can be inferred that there is no significant difference between these two groups.

Finally, the relevance of the variable "previous experience with VR technology" is shown below. Of the total of 13 participants in the VR group, seven had no previous experience and six did. From the data in these results, it is obtained that:

- With previous experience in VR technology, the average right answer was 9.83.
- Without previous experience in VR technology, the average right answer is 9.28.

Based on the data analyzed, it can be concluded that most of the variables considered at the beginning of the methodology section have no significant impact on the results obtained, with a clear exception. The field of study of the participants was found to be a highly influential factor in the study. Specifically, those students who were studying biomedical engineering demonstrated a clear advantage over other groups.

As mentioned above, there is not enough evidence to support the idea that the level of experience with VR or the academic degree of the participants have a conclusive impact on the results of the study. The data gathered and analyzed from these two variables suggests that there is not enough evidence for these factors to be considered significant in determining the outcome of the case study.

In conclusion, it can be observed that the variables initially considered to be the most relevant may not have had a major effect on the results obtained, except for the field of study of the participants. This suggests that in future studies, more emphasis should be placed on understanding the impact of academic background on the use of VR technology.

### 5.3.4 Outliers & Reviews

This section will focus on the analysis of the anomalous data found during the study. Additionally, it includes an analysis of the bugs found by the participants and a summary of the most frequently made comments.

### Outliers & anomalous data

As mentioned above, the results from questions #2 and #10 can be considered anomalous given the rest of the results obtained from the case study.

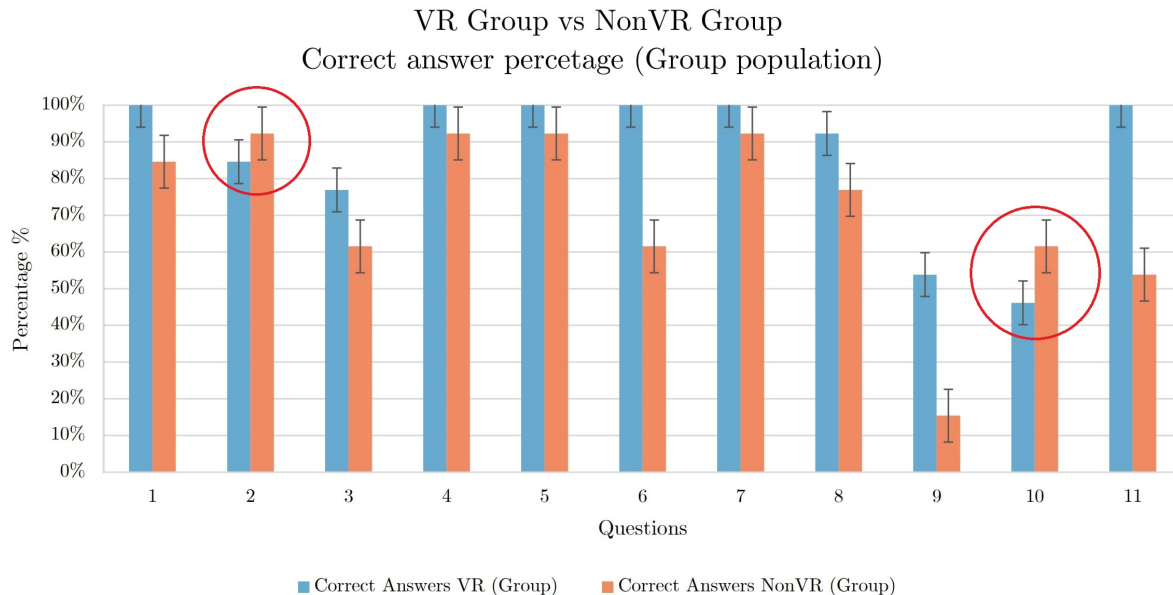


Figure 5.7: Indication of the 2 anomalies found in the results from the case study

Similar to what happened with other results from the questionnaire, the answers to both questions are mentioned orally during the presentation but not implicitly represented. In the case of question #2, reference should be made to slide 8 of the presentation, where the gallbladder diagram is mentioned and pointed out with all its parts. For the VR group, this concept is only pointed out when the dissection of the gallbladder is about to be done, so it is not reinforced in the same way as other concepts, given that more emphasis is placed on the correct position and location of the cut (Calot's triangle).

In this case, the lack of repetition of this concept during the simulation, added to the time since it was mentioned in the presentation, plus the simulation, may have played an important role in the retention of this information. This same argument can be used for question #10, where although the section on internal pressure management is an important part of the simulation (gas injection and aspirator), there is no mention here that respiration is another method used to expel excess  $CO_2$  into the patient. In both cases, it can be said that the lack of repetition of the information added to the time taken for the simulation did not allow the users to retain the information, and in general, they had a worse performance in these 2 questions, unlike their counterparts in the NonVR group.

One of the most unexpected outliers was the participation of a patient (participant #11) who underwent a cholecystectomy. While the rest of the participants in the NonVR group obtained an average of 67% of the correct answers (7/11 in the questionnaire), participant #11 had 82% of the correct results. In the subsequent review with the participant, he mentioned that before undergoing the procedure, he received information about the procedure and the tools from the doctors who performed the procedure. This would explain why he is the only participant in this group who got more than eight correct answers.

## Bug reports

When it comes to the bugs found by the users, as mentioned in the Results section, the most common was that objects wouldn't stay in the right position and would become unresponsive. It was found that the reason for these bugs is the permanence of the SteamVR "Interactable" script. This script allows the user to pick up objects and interact with them; however, it is specified in the object scripts that once the objects reach their correct position, the collider that allows the SteamVR scripts to work is disabled. However, especially with smaller objects such as cable connections, there is some delay before the disabling takes effect. If the user does not release the object even when it is already in the correct position, they are still able to modify the position of the object, nullifying the action of the script that removes the interaction ability.

This seems to be the case with what was reported by participants 3 and 13. It is possible to modify the action delay time by optimizing the code; however, in order to maintain the homogeneity of the simulation during the case study, these modifications were not made until after the end of the study.

## User reviews

While most of the users who performed the simulation reported that they found it useful, entertaining, and overall a good educational experience, some participants reported having trouble understanding the functions and limitations of the VR technology itself. Users who reported such difficulties had previously reported having no experience with VR technology, so it is understandable that there was some confusion with the functions and controls. This represents the most likely situation to happen in real-life scenarios due to the slow adoption of this technology in the mainstream market, and hence, by the general public.

One of the most difficult actions (or concepts to internalize) for inexperienced users was the concept of teleportation. Because of the way the controls are designed, accidental teleportation occurred when the user wanted to pick up an object because they unconsciously made the hand gesture needed to grab something by pressing the trackpad, whose function is teleportation.

This usually ended up with the user requiring assistance to relocate to the center of the laboratory, as sometimes the participant became disoriented. However, this seems to be just part of the learning curve and the users' adaptation process, as after the first few minutes of running the simulation, they were able to use the teleport function at will, even though most of the users with no VR experience preferred to move physically. Although this is a feature that both the HTC Vive Pro and Index Valve have, this is not a universal ability shared by all VR systems, so users should be encouraged to use the feature as much as possible, also considering the limited space in which the vast majority of VR systems are located.

### 5.3.5 Closing Remarks

From the combined test performed, both the descriptive statistics and the 2-sample t test show that there is evidence that implies that the results obtained by the VR control group are different from those obtained by the NonVR control group. These values hint that VR could enhance the learning process of students after receiving the necessary information in a master class, essentially strengthening the information retention by putting the theory into practice. Regarding the factors affecting these results, the number of participants and the field of study are the most significant variables to take into consideration, especially the number of participants, when it comes to the level of confidence and margin of error in the results obtained.

## Chapter 6

# Conclusions

From the research and analysis done, it can be concluded that, although there is evidence that VR as an educational technology provides improved information retention as a complement to the standard educational paradigm, more studies are needed to corroborate this with a larger number of participants, increasing the reliability of the results obtained. Responding to the research question at the end of Section 3.3, based on the results of the quantitative study and statistical analysis performed, it can be said that there is sufficient evidence that hints that VR is a technology that serves as an educational aid for engineering students related to the health area, provided that the information presented reinforces the concepts learned in the master lessons.

Additionally, although there are factors relevant to the study itself, the field of study of the participants is the most relevant. However, this does not give the full picture of what transpired during the study. The level of experience with VR and the speed of learning of each individual represent an important factor, not in terms of information retention but in the acceptance and engagement of the participants and how they feel about this type of technology.

This is consistent with the results obtained from the study "Using Virtual Reality in Biomedical Engineering Education" (see more in Section 3.1.1). Even though this study used the colloquial definition of the term "VR", it was shown that the simple fact of interacting beyond the master lessons engaged students significantly, and in subsequent interviews, they reported higher retention of information.

This suggests that, while useful as a tool, VR technology is currently ancillary, serving as a supplement to the current teaching system. This may change in the future with the incorporation of VR into the mainstream market and therefore into more higher education institutions, making this technology more accessible to the public. This incorporation into more aspects of life will also be linked to the development of networks, the speed of information transmission, and especially the reduction of latency, which currently represents one of the biggest obstacles to an immersive experience for the use of it online. There are currently initiatives such as Meta's Metaverse, which proposes an online platform for the use of virtual reality (whether for video games, education, or business), but it is still in its early stages of development.

In conclusion, it can be said that the study provides a better insight into the functionality of VR in the future of education and provides a basis for further development of applications aimed at the learning process of biomedical engineers. Although there are currently divided opinions about the usefulness and place of this technology in the current education scheme, research into it should not be neglected, given the potential benefits it may bring in the future. More research of this type can help take the next step in terms of education, which is not limited to biomedical engineering but potentially to all branches of knowledge.

## 6.1 Future Improvements & Recommendations

During the realization of this project, especially towards the final phases of it, there were points of improvement that can be implemented in the development of future VR didactic applications and in the studies to evaluate them.

**Programming from the beginning with VR platforms:** Although there is no initial problem with creating a simulation for a certain OS first and then translating it to VR, this can cause compatibility problems with some scripts, which leads to having to invest time in translating them or having to write new code in general. Working from the beginning with VR plug-ins simplifies this task and, in general, speeds up the developer's work.

**Compatibility issues with different software versions:** One of the biggest problems in the development of applications that require the use of multiple programs is the compatibility of the versions with each other. It is recommended that prior research be done on which programs work best with which versions of plug-ins because software such as Unity has multiple releases per year, so compatibility with plug-ins and assets varies greatly.

**Sample size for the study case:** A larger number of participants during the study case, i.e., a larger sample of data, would have given bigger confidence intervals and smaller margins of error both for the population and test results, so for future projects, it is recommended to have a population of at least 40 participants. One of the most important points to note is the influence of the data on the study population of 25 participants. In this case, the outliers have a greater influence on the results, so the possibility of them tipping the balance in favor of one study group or another is greater. Having a larger number of participants in the study mitigates the possibility of these situations arising. This should be taken into account considering the public's availability and the usually low participation of students in this type of project.

**Consideration of other factors that impact the results and VR experience:** Although it was found that the most relevant factor for the variation of results was the field of study of each of the participants, in future iterations of similar projects, the range of factors that can affect the results should be broadened. Age and affinity with computers (or technology in general) are factors that may be more relevant not only to quantitative results but also to overall user engagement with the developed software.

**Use of nonparametric statistics as an analysis tool:** nonparametric statistics are tools that allow rejecting or approving hypotheses without taking into account the normality of the data. Given that at the time of performing the 2-sample t test, results were obtained with a warning about the normality of the population, it is recommended that in the future, if the population size of a new study is equally small, consideration be given to the use of nonparametric tests such as the Mann-Whitney test, with which null hypotheses can be approved or rejected in the same way.

# Bibliography

- [1] Hazim Sadideen and Roger Kneebone. “Practical skills teaching in contemporary surgical education: how can educational theory be applied to promote effective learning?” In: *The American Journal of Surgery* 204.3 (2012), pp. 396–401. ISSN: 0002-9610. DOI: <https://doi.org/10.1016/j.amjsurg.2011.12.020>. URL: <https://www.sciencedirect.com/science/article/pii/S0002961012002656>.
- [2] *Lancers Army School*. Sept. 2022. URL: <https://lancersarmyschools.com/everything-you-need-to-know-about-theoretical-learning-vs-practical-learning/>.
- [3] Roberto De Ponti et al. “Pre-graduation medical training including virtual reality during COVID-19 pandemic: a report on students’ perception”. In: *BMC Medical Education* 20.1 (Sept. 2020). DOI: [10.1186/s12909-020-02245-8](https://doi.org/10.1186/s12909-020-02245-8). URL: <https://doi.org/10.1186/s12909-020-02245-8>.
- [4] Bernard Marr. *The fascinating history and evolution of extended reality (XR)*. July 2021. URL: <https://bernardmarr.com/the-fascinating-history-and-evolution-of-extended-reality-xr-covering-ar-vr-and-mr/>.
- [5] Joakim Vindenes. *The history of virtual reality - How did the technology develop?* Aug. 2020. URL: <https://www.matrise.no/2018/07/the-history-of-virtual-reality/>.
- [6] Chris Wiltz. *The story of Sega VR: Sega’s failed virtual reality headset*. June 2020. URL: <https://www.designnews.com/electronics-test/story-sega-vr-segas-failed-virtual-reality-headset>.
- [7] Elisabet Rodríguez. *Historia de los microprocesadores*. Dec. 2009. URL: <http://www.maestrosdelweb.com/historia-de-los-microprocesadores>.
- [8] Carla Tardi. *What is Moore’s law and is it still true?* Oct. 2022. URL: <https://www.investopedia.com/terms/m/mooreslaw.asp#:~:text=Moore's%20Law%20states%20that%20the,that%20this%20growth%20is%20exponential..>
- [9] Alexander Sandstrouml. *Hololens 2: The future of remote support*. Aug. 2022. URL: <https://www.xmreality.com/blog/hololens2>.
- [10] Darren Allan. *Thinking of getting an HTC Vive? check that your room is this size first*. Feb. 2016. URL: <https://www.techradar.com/news/wearables/thinking-of-getting-a-htc-vive-check-that-your-room-is-this-size-first-1314450>.



- 
- [11] Robert Bloomfield. *How do common virtual reality tracking systems work?* Oct. 2019. URL: <https://www.mechatech.co.uk/journal/how-do-common-virtual-reality-tracking-systems-work>.
- [12] Jorge Guerrero-Aguirre et al. “La cirugía de invasión mínima: antecedente histórico; presente y perspectivas futuras en el ISSSTE”. In: *Revista de Especialidades Médico-Quirúrgicas* 19.3 (2014), pp. 375–386.
- [13] HospiMedica International. *First-of-its-kind 4K endoscope for arthroscopic surgical procedures launched globally*. Nov. 2022. URL: <https://www.hospimedica.com/surgical-techniques/articles/294795218/first-of-its-kind-4k-endoscope-for-arthroscopic-surgical-procedures-launched-globally.html>.
- [14] Xion-Medical. *Trocar system for Laparoscopy*. 2020. URL: <https://www.xion-medical.com/en/products/instruments/trocar-system-for-laparoscopy>.
- [15] Neil A Dodgson. “Variation and extrema of human interpupillary distance”. In: *Stereoscopic displays and virtual reality systems XI*. Vol. 5291. SPIE. 2004, pp. 36–46.
- [16] Ul-Solutions. *VRmark - The Virtual Reality Benchmark*. Feb. 2022. URL: <https://benchmarks.ul.com/vrmark>.
- [17] Sem F Hardon et al. “A new modular mechanism that allows full detachability and cleaning of steerable laparoscopic instruments”. In: *Surgical endoscopy* 33.10 (2019), pp. 3484–3493.
- [18] Dilip Kumar Dutta and Indranil Dutta. “The harmonic scalpel”. In: *The Journal of Obstetrics and Gynecology of India* 66.3 (2016), pp. 209–210.
- [19] Ben Price. “Archaeology Ex Machina - Virtual Archaeology Accessibility: Assessing the feasibility and sustainability of minimal resource VR modelling and its applicability to small-scale archaeological research via the exploration of a Scottish Atlantic Iron Age roundhouse site on Orkney.” en. In: *Independent* (2020). DOI: [10.13140/RG.2.2.21060.96642](https://doi.org/10.13140/RG.2.2.21060.96642). URL: <http://rgdoi.net/10.13140/RG.2.2.21060.96642>.
- [20] Simon Adler. *Entwicklung von Verfahren zur interaktiven Simulation minimal-invasiver Operationsmethoden*. de. 1st ed. Magdeburger Schriften zur Visualisierung. Herzogenrath, Germany: Shaker, Mar. 2014.
- [21] Divya Seernani et al. *What is VR Eye Tracking? [and how does it work?]* Nov. 2022. URL: <https://imotions.com/blog/learning/best-practice/vr-eye-tracking/#>.
- [22] Kyoshiro Sasaki, Keiko Ihaya, and Yuki Yamada. “Avoidance of novelty contributes to the uncanny valley”. In: *Frontiers in Psychology* 8 (2017), p. 1792.
- [23] Ting Yan. *Uncanny Valley in VR/AR*. Sept. 2019. URL: <https://www.tingyan.me/uncanny-valley-research>.
- [24] Mauricio Parrales, Mario Medina, and Marco Zuniga. *Colecistectomía Laparoscópica, UN enfoque anatomoclínico*. July 2018. URL: <https://www.medigraphic.com/cgi-bin/new/resumen.cgi?IDARTICULO=80603>.

- [25] Richard L Drake and FIPAT. *Terminologia Anatomica: International Anatomical Terminology*. Stuttgart, Germany: Thieme Publishing Group, May 2011.
- [26] Antonio de Lacy. *Qué Es la vesícula biliar. funciones, Patologías Frecuentes y tratamiento*. 2021. URL: <https://www.iqlacy.com/que-operamos/cirugia-de-la-vesicula-biliar/vesicula-biliar/>.
- [27] Terese Winslow. *Gallbladder Anatomy*. Mar. 2010. URL: <https://visualsonline.cancer.gov/details.cfm?imageid=9078>.
- [28] Adam Yopp and Amit Singal. *Hepatocellular carcinoma*. Jan. 2019. URL: <https://www.cancertherapyadvisor.com/home/decision-support-in-medicine/hospital-medicine/hepatocellular-carcinoma/>.
- [29] Robin Germán Prieto Ortiz and Rolando Alonso Martínez Acero. “ANATOMÍA DE LA VÍA BILIAR”. In: *de Cirugía General* (), p. 209.
- [30] Susan Campbell Westerway. “Ultrasound of the gallbladder - is it necessary to fast”. In: *Educational Supplement* (Sept. 2003).
- [31] RG Andall et al. “The clinical anatomy of cystic artery variations: a review of over 9800 cases”. In: *Surgical and Radiologic Anatomy* 38.5 (2016), pp. 529–539.
- [32] Kivanç Muşlu et al. “Speculative analysis of integrated development environment recommendations”. In: *ACM SIGPLAN Notices* 47.10 (2012), pp. 669–682.
- [33] Eleftheria Christopoulou and Stelios Xinogalos. “Overview and comparative analysis of game engines for desktop and mobile devices”. In: *Institutional Repository of Academic Research* (2017).
- [34] Arnab Majumder, Maria S. Altieri, and L. Michael Brunt. “How do I do it: laparoscopic cholecystectomy”. In: *Annals of Laparoscopic and Endoscopic Surgery* 5.0 (2020). ISSN: 2518-6973. URL: <https://ales.amegroups.com/article/view/5766>.
- [35] Christina C. Lindenmeyer. *Síndrome poscolecistomía - trastornos hepáticos Y biliares*. Dec. 2022. URL: <https://www.msdmanuals.com/es-co/professional/trastornos-hep%C3%A1ticos-y-biliares/trastornos-de-la-ves%C3%ADcula-biliar-y-los-conductos-biliares/s%C3%ADndrome-poscolecistom%C3%ADa>.
- [36] Stephen Doxey et al. “Incidental Finding of Gallbladder Torsion During Laparoscopic Cholecystectomy”. In: *Cureus* (Sept. 2021). DOI: 10.7759/cureus.18058. URL: <https://doi.org/10.7759/cureus.18058>.
- [37] J.D. Kondrup and B. Sylvester. “Palmer's Point – Go for It! The Technique of LUQ and Direct View Entry”. In: *Journal of Minimally Invasive Gynecology* 19.6 (Nov. 2012), S177. DOI: 10.1016/j.jmig.2012.08.506. URL: <https://doi.org/10.1016/j.jmig.2012.08.506>.
- [38] Isaac Tranter-Entwistle et al. “Operative difficulty in laparoscopic cholecystectomy: considering the role of machine learning platforms in clinical practice”. In: *Artificial Intelligence*

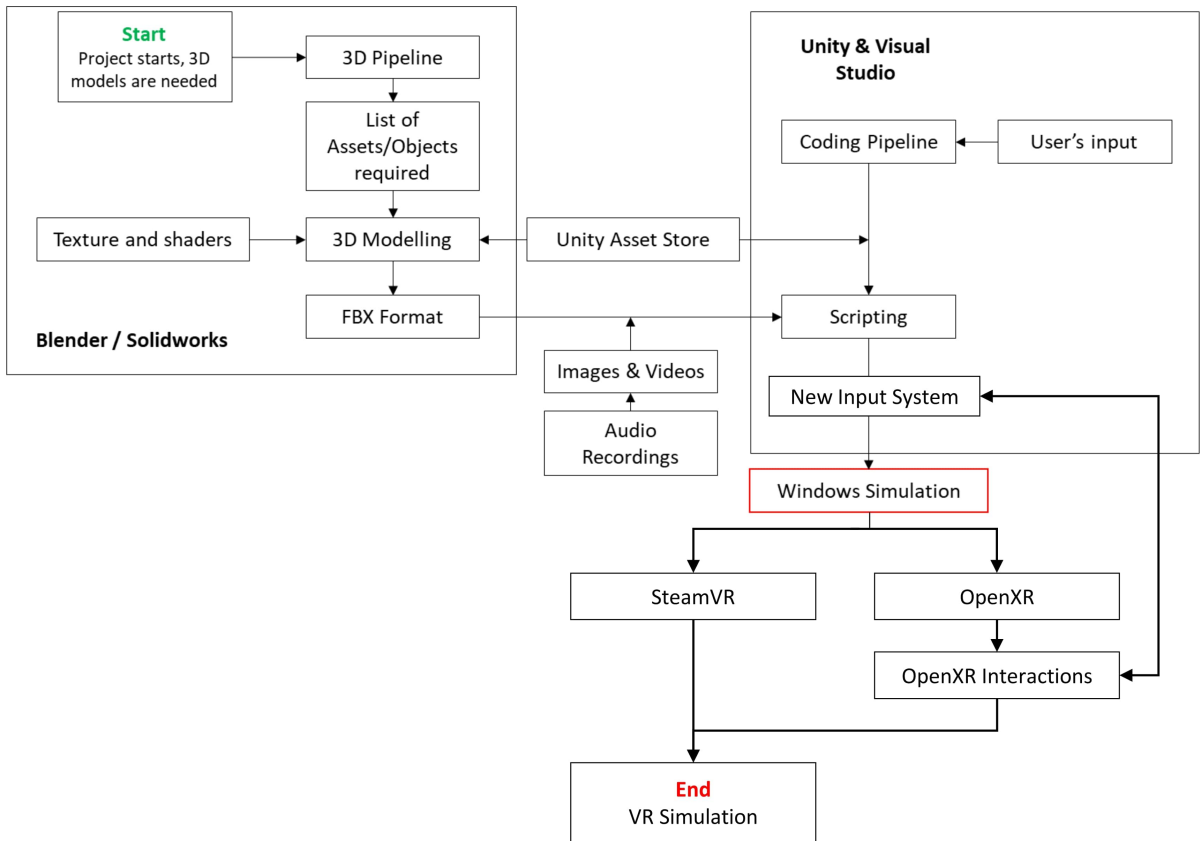
- Surgery* (2022). DOI: 10.20517/ais.2022.01. URL: <https://doi.org/10.20517/ais.2022.01>.
- [39] Joel Falconer. *What is an IDE? how does it enable faster development?* Dec. 2022. URL: <https://www.sitepoint.com/what-is-an-ide/>.
- [40] Anita Singh et al. "Using Virtual Reality in Biomedical Engineering Education". In: *Journal of Biomechanical Engineering* 142.11 (Sept. 2020). DOI: 10.1115/1.4048005. URL: <https://doi.org/10.1115/1.4048005>.
- [41] Hernando Matiz Camacho. "La simulación clínica: nueva herramienta para enseñar medicina". In: *Medicina* 34.3 (2012), pp. 242–246.
- [42] Thomas P Duffy. "The Flexner Report–100 years later". en. In: *Yale J. Biol. Med.* 84.3 (Sept. 2011), pp. 269–276.
- [43] *Virtual reality in medical training: Educate better, save more lives.* Mar. 2022. URL: <https://program-ace.com/blog/virtual-reality-in-medical-training/>.
- [44] Taylor Lyles. *'which VR headsets work with steam?': A full guide to virtual reality on the steam gaming app.* Nov. 2022. URL: <https://www.businessinsider.com/guides/tech/which-vr-headsets-work-with-steam>.
- [45] *OpenXR Compatibility.* 2022. URL: <https://mbucchia.github.io/OpenXR-Toolkit/compat.html>.
- [46] Margie Semilof. *What is alpha testing?* Oct. 2021. URL: <https://www.techtarget.com/searchsoftwarequality/definition/alpha-testing>.
- [47] National Cancer Institute. *Gallbladder.* 2022. URL: <https://www.cancer.gov/types/gallbladder>.
- [48] John French. *How to play audio in unity (with examples).* Jan. 2022. URL: <https://gamedevbeginner.com/how-to-play-audio-in-unity-with-examples/>.
- [49] Brian Beers. *P-value: What it is, how to calculate it, and why it matters.* Feb. 2023. URL: <https://www.investopedia.com/terms/p/p-value.asp>.
- [50] Rebecca Bevans. *Understanding P values: Definition and examples.* Nov. 2022. URL: <https://www.scribbr.com/statistics/p-value/>.
- [51] *Why the P-value is significant.* Sept. 2015. URL: <https://tidsskriftet.no/en/2015/09/why-p-value-significant..>
- [52] Noah Schwartz. *5 recent Virtual Reality Medical Innovations.* 2022. URL: <https://www.beckershospitalreview.com/innovation/5-recent-virtual-reality-medical-innovations.html>.
- [53] Mar García. *Platón y la educación del individuo - leer más en ihistorarte.* May 2020. URL: <https://ihistorarte.com/pequehistoria/platon-y-la-educacion-del-individuo/>.

- [54] Paul Milgram and Fumio Kishino. “A Taxonomy of Mixed Reality Visual Displays”. In: *IEICE Trans. Information Systems* vol. E77-D, no. 12 (Dec. 1994), pp. 1321–1329.
- [55] Infraction. *Visible cinematic music*. 2022. URL: <https://inaudio.org/track/visible-documentary/>.
- [56] MaxkoMusic. *Medical Examination*. 2021. URL: <https://soundcloud.com/maxkomusic>.
- [57] G S Ruthenbeck and K J Reynolds. “Virtual reality for medical training: the state-of-the-art”. In: *Journal of Simulation* 9.1 (Feb. 2015), pp. 16–26. DOI: [10.1057/jos.2014.14](https://doi.org/10.1057/jos.2014.14). URL: <https://doi.org/10.1057/jos.2014.14>.
- [58] Filip Górski et al. “Effective Design of Educational Virtual Reality Applications for Medicine using Knowledge-Engineering Techniques”. In: *EURASIA Journal of Mathematics, Science and Technology Education* 13.2 (Dec. 2016). DOI: [10.12973/eurasia.2017.00623a](https://doi.org/10.12973/eurasia.2017.00623a). URL: <https://doi.org/10.12973/eurasia.2017.00623a>.

# Chapter 7

## Appendix

### Appendix A. Complete pipeline for the creation of the simulation



## Appendix B. Tools Used for the development of the project

Table 7.1: Tools used for the development of the entirety of the project in all its stages

	Name	Version	Use	URL
3D Modelling	Blender	3.3.1	Creation of normal and low precision 3D models	<a href="https://www.blender.org/download/releases/3-3/">https://www.blender.org/download/releases/3-3/</a>
	SolidWorks	2020sP1	Creation of high precision 3D models	<a href="https://www.solidworks.com">https://www.solidworks.com</a>
Image and Audio Design	Adobe Photoshop 2023	24.2	Image and texture editing	<a href="https://www.adobe.com/products/photoshop.html">https://www.adobe.com/products/photoshop.html</a>
	Adobe Audition 2023	23.2	Audio editing	<a href="https://www.adobe.com/products/audition.html">https://www.adobe.com/products/audition.html</a>
	Replica Studios	0.2.0	AI Audio generation	<a href="https://replicastudios.com/">https://replicastudios.com/</a>
	Topaz Foto AI	12.1	Image and texture enhancing	<a href="https://www.topazlabs.com/topaz-photo-ai">https://www.topazlabs.com/topaz-photo-ai</a>
Development	Unity	2020.3.37f1	Development of the simulation	<a href="https://unity.com/releases/editor/whats-new/2020.3.37">https://unity.com/releases/editor/whats-new/2020.3.37</a>
	Visual Studio	2022	Development and deployment of scripts	<a href="https://visualstudio.microsoft.com/vs/">https://visualstudio.microsoft.com/vs/</a>
Writing	Overleaf	2023	Writing of the document	<a href="https://www.overleaf.com">https://www.overleaf.com</a>
	DeepL	2023	Translation	<a href="https://www.deepl.com/translator">https://www.deepl.com/translator</a>
	Quillbot	2023	Grammar check and paraphrasing	<a href="https://quillbot.com/grammar-check">https://quillbot.com/grammar-check</a>
	Citation Machine	2023	Bibtex creation tool	<a href="https://www.citationmachine.net/">https://www.citationmachine.net/</a>
	Notion	2023	Writing of the document and grammar checking	<a href="https://www.notion.so/">https://www.notion.so/</a>
	Google Scholar	2023	Paper and books research	<a href="https://scholar.google.com/">https://scholar.google.com/</a>

## Appendix C. Asset List

Table 7.2: Assets used for the creation of the simulation

Asset Name	Author/Creator	Origin	Description	URL
Anatomy liver and gallbladder Low-poly 3D model	Art-Studio3d	CGTrader	Model of the liver and gallblader with the right position of the blood vessels	<a href="https://www.cgtrader.com/3d-models/character/anatomy/anatomy-liver-and-gallbladder">https://www.cgtrader.com/3d-models/character/anatomy/anatomy-liver-and-gallbladder</a>
Clinic - Operating room	Mixall	Unity Asset Store	Props and scenerios of operating rooms and surgery equipment	<a href="https://assetstore.unity.com/packages/3d/props/interior/clinic-operating-room-92667">https://assetstore.unity.com/packages/3d/props/interior/clinic-operating-room-92667</a>
Dust bin 3D Model	sadik	GCTrader	3D model of a trashcan	<a href="https://www.cgtrader.com/free-3d-models/exterior/other/dustbin-533a0fd8-4fc7-41f9-b40c-99a346b63a9c">https://www.cgtrader.com/free-3d-models/exterior/other/dustbin-533a0fd8-4fc7-41f9-b40c-99a346b63a9c</a>
Gallbladder 3D model	Erik yavrumyan	CGTrader	Anatomically correct model of a gallblader and bile duct	<a href="https://www.cgtrader.com/3d-models/character/anatomy/gallbladder">https://www.cgtrader.com/3d-models/character/anatomy/gallbladder</a>
Human 3D model	Nicorobin3DModels	CGTrader	Complete model of the human anatomy	<a href="https://www.cgtrader.com/3d-models/science/medical/human-995f9266-65a7-40da-be89-c7036cee52cf">https://www.cgtrader.com/3d-models/science/medical/human-995f9266-65a7-40da-be89-c7036cee52cf</a>
Laparoscopic Instrument Scissors Low-poly 3D model	EMAS-Studio	CGTrader	Laparoscopic Intruments	<a href="https://www.cgtrader.com/3d-models/science/medical/laparoscopic-instrument-scissors">https://www.cgtrader.com/3d-models/science/medical/laparoscopic-instrument-scissors</a>
Laparoscopic Trocar	Gulzar Ahmed	Sketchfab	Model of a trocar and an endoscope	<a href="https://sketchfab.com/3d-models/laparoscopic-trocar-4553d8f1888743f3bef15b681794244a">https://sketchfab.com/3d-models/laparoscopic-trocar-4553d8f1888743f3bef15b681794244a</a>
Mesh Slicer	Stas Bz	Unity Asset Store	Scripts that allow to compute finite elements analysis and slice objects	<a href="https://assetstore.unity.com/packages/tools/modeling/mesh-slicer-59618">https://assetstore.unity.com/packages/tools/modeling/mesh-slicer-59618</a>
Obi Rope	Virtual Method	Unity Asset Store	Scripts that allow the simulation of ropes and the physics implemented into them	<a href="https://assetstore.unity.com/packages/tools/physics/obi-rope-55579">https://assetstore.unity.com/packages/tools/physics/obi-rope-55579</a>
Oxygen tank 3D Model	Elena marundik	GCTrader	Oxygen tank model used as CO2 tank in the simulation	<a href="https://www.cgtrader.com/sashamarundik">https://www.cgtrader.com/sashamarundik</a>
Player Starter Assets - First person Character Controller	Unity Technologies	Unity Asset Store	Camera and movement controls with the new Unity Input System	<a href="https://assetstore.unity.com/packages/essentials/starter-assets-first-person-character-controller-urp-196525">https://assetstore.unity.com/packages/essentials/starter-assets-first-person-character-controller-urp-196525</a>
Rigid endoscope	Hazam Eltaief	GrabCad	Model of a rigid endoscope	<a href="https://grabcad.com/library/rigid-endoscope-1">https://grabcad.com/library/rigid-endoscope-1</a>
Smart TV Free 3D model	Danthree	GCTrader	TV prop model	<a href="https://www.cgtrader.com/free-3d-models/interior/living-room/smart-tv-c075ac00-05c7-42ab-8665-aed7b0d575b5">https://www.cgtrader.com/free-3d-models/interior/living-room/smart-tv-c075ac00-05c7-42ab-8665-aed7b0d575b5</a>
Started Pack Grab&Drop	Sunny Valley Studio	Sunny Valley Studio Website	Unity project created to teach unity users how to develop the grab & drop system	<a href="https://drive.google.com/file/d/1YLzKjN7ABhOCWdHYMlRG59ZgJRemCuYF/view">https://drive.google.com/file/d/1YLzKjN7ABhOCWdHYMlRG59ZgJRemCuYF/view</a>
Surgical Instruments	Fero Adezo	Sketchfab	Models of multiple surgical instruments	<a href="https://sketchfab.com/3d-models/surgical-instrument-2971a44d334b45ff926792e92a6bcee7">https://sketchfab.com/3d-models/surgical-instrument-2971a44d334b45ff926792e92a6bcee7</a>
TextMeshPro	Unity Technologies	Unity Asset Store	Unity pack for placing text on surfaces inside Unity	<a href="https://docs.unity3d.com/Manual/com.unity.textmeshpro.html">https://docs.unity3d.com/Manual/com.unity.textmeshpro.html</a>
Wet Stuff	Placeholder Software	Unity Asset Store	Texture and script pack that allow to simulate wetness in a surface	<a href="https://assetstore.unity.com/packages/tools/particles-effects/wet-stuff-118969">https://assetstore.unity.com/packages/tools/particles-effects/wet-stuff-118969</a>
Yue Softbody Physics	Yueutility-Studios	Unity Asset Store	Scripts that allow to give objects retractable properties for added realism	<a href="https://assetstore.unity.com/packages/tools/physics/yue-softbody-physics-224754">https://assetstore.unity.com/packages/tools/physics/yue-softbody-physics-224754</a>

## Appendix D. VR plugin Compatibility List [44] [45].

Table 7.3: Open XR &amp; SteamVR plugins compatibility list with different HMD

<b>Steam VR Compatibility List</b>	
Virtual reality System	Full Compatibility
Valve Index	Yes
Oculus Rift	Yes
Oculus Rift S	Yes
HTC Vive	Yes
HTC Vive Pro	Yes
HTC Vive Cosmos	Yes
Razer OSVR	Limited
Pimax 4K, 5K, 5K Plus, 8K, and 8K Plus	Yes
Dell Visor	Limited
Samsung Odyssey and Odyssey+	Yes
Acer AH101	Limited
HP WMR	Yes
Lenovo Explorer	Limited
HP Reverb	Yes
Varjo VR-1 and VR-2	Limited
<b>Open XR Compatibility List</b>	
Virtual reality System	Full Compatibility
Windows Mixed Reality (HP Reverb, Samsung Odyssey...)	Yes
Oculus (Rift, Quest, Quest 2, Quest Pro...)	Yes
Oculus (via Virtual Desktop))	Yes
Varjo (Aero, VR-3...)	Yes
Pimax (5K, 8K...)	Limited
HTC Tier 1 (Vive original, Vive Pro)	Yes
HTC Tier 2 (Vive Cosmos, Vive Focus)	Yes
Valve Index	Yes
Pico (Neo 3, Neo 4)	Yes



## Appendix E. Presentation used for study

# Development of a Didactic VR Tool for Minimal Invasive Surgery

**Juan Felipe Diaz Quintero**  
5016761

**1st Reviewer**  
Prof. Dr. Johannes Tümler

**2nd Reviewer**  
Prof. Dr. Brigitte Kipfmüller



**Master Biomedical  
Engineering**

**Master thesis**

1

# Introduction

Developing of a Didactic VR Tool for Minimal Invasive Surgery

2

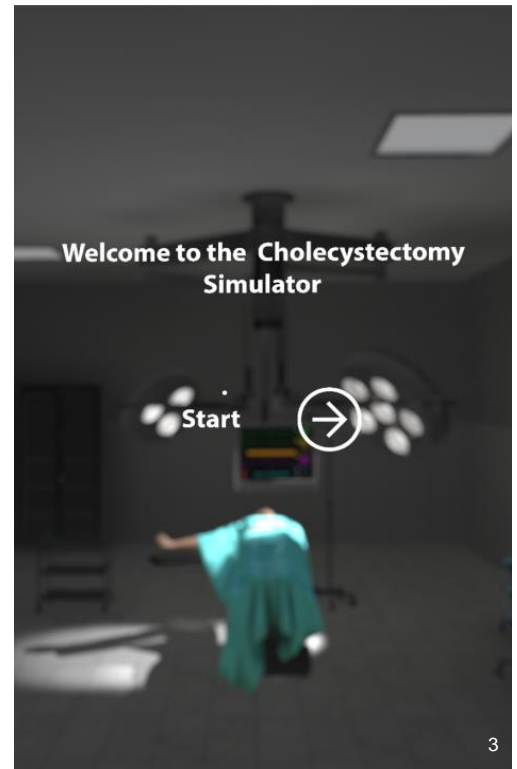
# Introduction

## General Topic

- Use of VR technology for non-health science students.
- Creation of a didactic VR tool adaptable to the class's needs.
- Simulation of the procedure for a cholecystectomy.



Developing of a Didactic VR Tool for Minimal Invasive Surgery



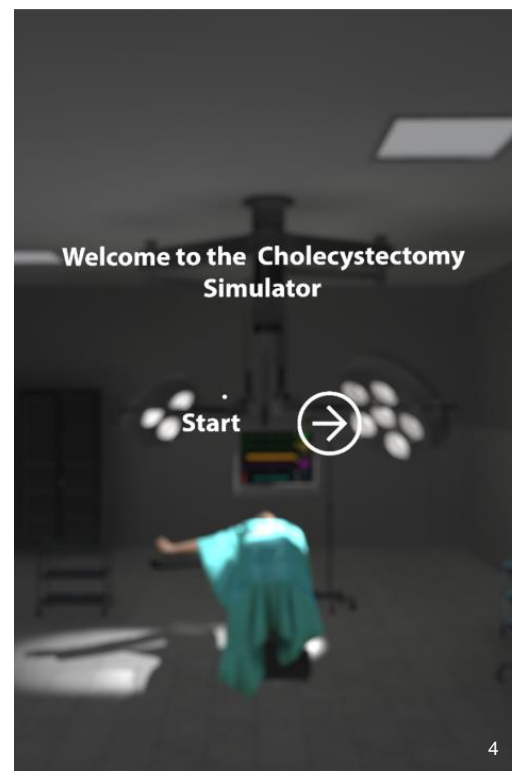
# Introduction

## Specific Objectives

- Creation of anatomical 3D elements and tools for cholecystectomy.
- Establish the correct steps for the procedure based on research.
- **Conduct a study** to demonstrate the effectiveness of this technology in teaching.



Developing of a Didactic VR Tool for Minimal Invasive Surgery



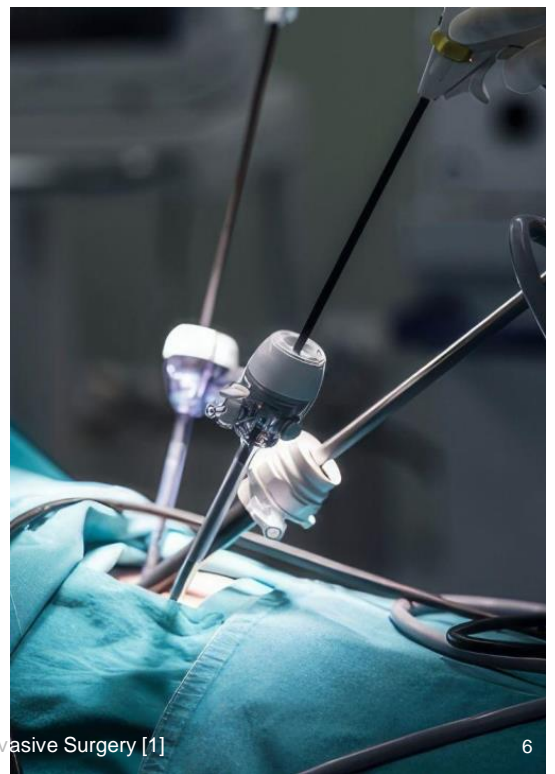
# Minimal Invasive Surgery

## What is MIS?

Is surgery performed from a distance, with long micro instruments applied through trocars and controlled by using a video system

Some advantages over open surgery are:

- Less trauma for the patient
- Shorter stay in the hospital
- Lower possibility of infection
- Organs are not retracted by hooks



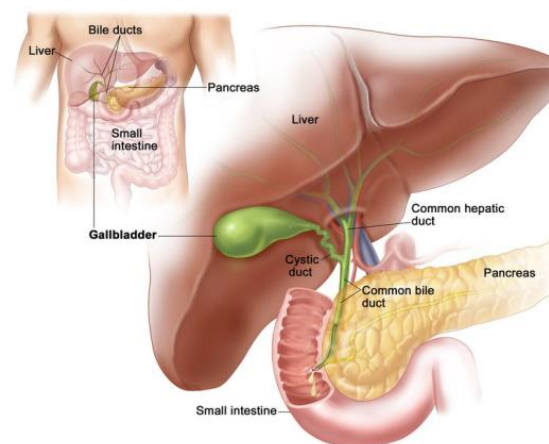
# Cholecystectomy

## What is the Cholecystectomy?

Minimal Invasive surgery to remove the gallbladder.

The gallbladder is the organ in charge of delivering the bile to the digestive tract.

Connected by the bile duct to liver and small intestine.



# Why removing the gallbladder?

## Cholecystitis

Inflammation of the gallbladder, this can be due to the presence of gallstones, which can be the size of golf balls.

This can cause severe nausea, swelling and severe pain for the patient.



# Tools for MIS

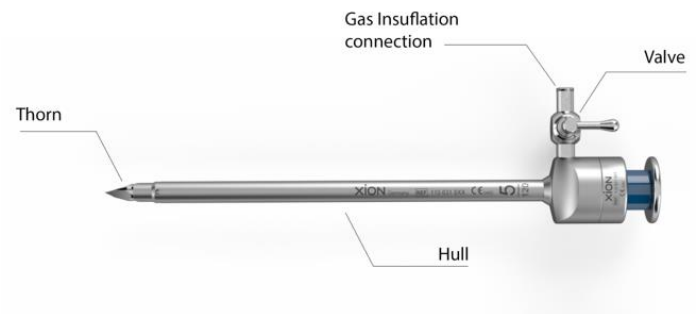
# What instruments are used in MIS?

## Trocar

Sharp instrument used for generating access into the abdominal cavity, for both instruments and gas.

Preferred instrument over the Veress Needle.

Multiple diameters, depending on the application. Includes a valve for the gas or aspiration connection



# What instruments are used in MIS?

## Gas insufflator

Gas is used to lift the abdominal wall, to be able to see with the endoscope.

The gas used is CO<sub>2</sub>, because is not flammable, not secondary effect and can regulated through the patient's breathing.

Pressure used is between 8-12 mmHg to not cause more trauma to the patient



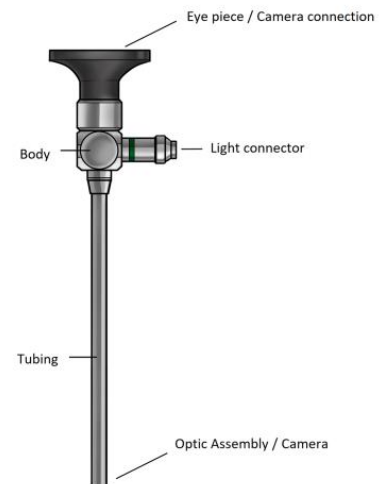
# What instruments are used in MIS?

## Endoscope

Long instruments with a camera and a light system that allows the video transmission to observe inside the patient's body.

Flexible or rigid (for cholecystectomy is always rigid).

Depending on the model can have multiple viewing angles (0 and 30 are the most common)



# What instruments are used in MIS?

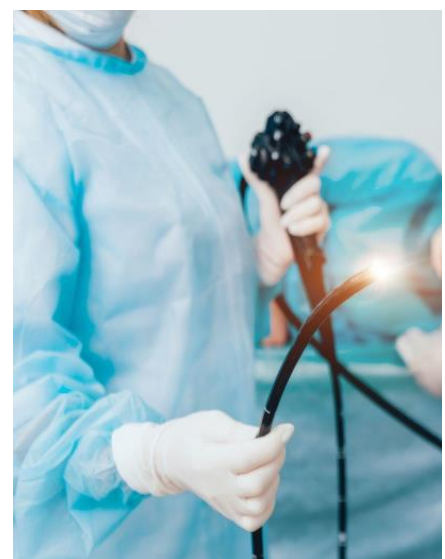
## Camera & light source

The endoscope is connected through a machine to a monitor

The light source needs to be extremely reliable (some lights can last more than 30.000 hours).

The light temperature used is 5500K (cold light) with at least 150W, and must be either LED or Xenon.

Must include a flash for documentation photography.



# What instruments are used in MIS?

## Graspers and scissors

Long instruments, designed with a tip at the end that allows to grab or to make blunt cuts inside the human body.

There are other instruments for such end, like:

- Laser cutters
- HF cutter
- Water Jet Cutter
- Harmonic scalpel



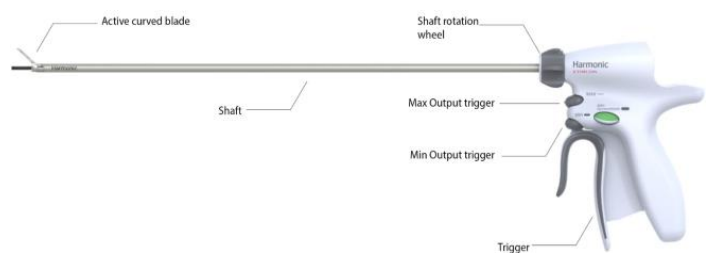
# What instruments are used in MIS?

## Harmonic Scalpel

This tool represents a great advantage over other methods, since most of them are riskier to use inside a patient's body

This tool is extremely versatile. The main 4 functions are:

- Cavitation
- Coagulation
- Coaptation
- Cutting





# What instruments are used in MIS?

## Harmonic Scalpel

The way it works is by using a piezoelectric crystal, into which a current is introduced and which begins to vibrate.

The vibrations can reach up to 55500 Hz, with an amplitude varying from 25 to 100 $\mu$ m at different levels.

Advantages:

- Safe to use with CO<sub>2</sub>
- No burning
- No smoke
- No current applied to the patient's body



# References

# References

- [1] Stansil, R. (2019, September 19). *How minimally invasive surgery has changed the healthcare industry*. Surgical Support & Operating Room Management Solutions. Retrieved March 5, 2023, from <https://surgical-solutions.com/blog/how-minimally-invasive-surgery-has-changed-the-healthcare-industry/>
- [2] RG Andall et al. "The clinical anatomy of cystic artery variations: a review of over 9800 cases". In: *Surgical and Radiologic Anatomy* 38.5 (2016), pp. 529–539.
- [3] Milanowski, A. (2018, July 23). *Gallstones: Watch and wait, or intervene?* Consult QD. Retrieved March 6, 2023, from <https://consultqd.clevelandclinic.org/gallstones-watch-and-wait-or-intervene/>
- [4] Xion-Medical. Trocar system for Laparoscopy. 2020. url: <https://www.xion-medical.com/en/products/instruments/trocar-system-for-laparoscopy>.
- [5] SA, O. (2022). *Co2 insufflator - all medical specialities - olympus medical systems*. CO2 Insufflator - All medical specialities - Olympus Medical Systems. Retrieved March 5, 2023, from <https://www.olympus-europa.com/medical/en/Products-and-Solutions/Products/CO2-Insufflator.html>
- [6] SA, S. C. (2020). *Rigid and semi rigid endoscope repairs laparoscope arthroscope*. ScopeCare. Retrieved March 5, 2023, from <https://www.scopecare.com/rigid-and-semi-rigid-endoscopes/>

Developing of a Didactic VR Tool for Minimal Invasive Surgery

# References

- [7] Lin, S. (2023, January 30). *Endoscopy: An overview of the procedure*. Medical Channel Asia. Retrieved March 5, 2023, from <https://medicalchannelasia.com/endoscopy-an-overview-of-the-procedure/>
- [8] STUDIO, E. M. A. S. (2020). *Laparoscopic Instrument Scissors: 3D model*. CGTrader. Retrieved March 5, 2023, from <https://www.cgtrader.com/3d-models/science/medical/laparoscopic-instrument-scissors>
- [9] Dilip Kumar Dutta and Indranil Dutta. "The harmonic scalpel". In: *The Journal of Obstetrics and Gynecology of India* 66.3 (2016), pp. 209–210.
- [10] Doyle, J. (2019). *Behzad Rahimi - Marketing manager*. J&J advanced energy device. . Retrieved March 5, 2023, from <https://ir.linkedin.com/in/behzad-rahimi-70a711107>

Developing of a Didactic VR Tool for Minimal Invasive Surgery

## Appendix F. Code implemented

This section of the appendix includes the most relevant scripts used in the development of the VR simulation. It should be noted that not all the code of these scripts is included, but they have been modified to show only the most relevant algorithms. All remaining code, as well as the unmodified code presented here, is available at the XR Laboratory of the Anhalt University.

First, the 7 code can be observed below. Here are the most common algorithms used in all audio scripts for the different scenes.

Code 7.1: Audio Handler for the Surgery Room (modified)

```

1  using BzKovSoft.ObjectSlicer.Samples;
2  using System.Collections;
3  using System.Collections.Generic;
4  using UnityEngine;
5  using UnityEngine.Events;
6  using Valve.VR;
7  using YuutilitySoftbody;
8
9  public class AudioHandlerSR : MonoBehaviour
10 {
11     //These methods calls into the code all the Adio Source and Audio clips
12     public AudioSource Beginning;
13     public AudioSource Gas;
14     public AudioClip GasConnect;
15
16
17     //These methods calls all the scene Objects whose state are linked to the Audio
18     //Guide
19     [SerializeField]
20     private GameObject BeginAObject;
21     [SerializeField]
22     private GameObject TGasAObject;
23     [SerializeField]
24     private GameObject AudioPos;
25     [SerializeField]
26     private GameObject GasItem;
27     [SerializeField]
28
29     //These bools are used to perform an action only once
30     bool Check1;
31     bool Check2;
32
33
34     //These Vectors are to determine if an GameObject is in the correct position
35     Vector3 RightP = new Vector3(-0.640999973f, 0.976999998f, -0.105499998f);
36
37     //Adds delay to let scene load correctly
38     public float delay;
39
40     void Start()
41     {
42         //Starts the first audio after a given delay, to let the scene load correctly
43         double beginDuration = (double)Beginning.clip.samples /
44             Beginning.clip.frequency;
45         double gasDuration = (double)Gas.clip.samples / Gas.clip.frequency;
46
47         double startTime = AudioSettings.dspTime + delay;
48
49         //Plays the second audio after the first one is finished

```

```

49     Beginning.PlayScheduled(startTime);
50     Gas.PlayScheduled(startTime + beginDuration);
51
52 }
53
54 void Update()
55 {
56     //Invoke the Sounds void, where all the rest of the audios are called back
57     Invoke("Sound", 0f);
58 }
59
60
61
62 void Sound()
63 {
64     //The algorithms used in this section largely rely on turning off the other
65     //audios and playing the right clip with the right actions.
66     if (!Check1 && GasItem.transform.position == RightP)
67     {
68         BeginAObject.SetActive(false);
69         TGasAObject.SetActive(false);
70         GameObject.Find("One shot audio").SetActive(false);
71         AudioSource.PlayClipAtPoint(GasConnect, AudioPos.transform.position,
72             1f);
73         Check1 = true;
74     }
75 }
76
77 //All the reset methods are linked to a button so the user can hear the audio
78 //once again. This deactivate the rest of the elements and activate only the
79 //desired one.
80 public void ResetGas()
81 {
82     if (GameObject.Find("One shot audio") != null)
83     {
84         GameObject.Find("One shot audio").SetActive(false);
85     }
86     BeginAObject.SetActive(false);
87     BeginAObject.SetActive(true);
88     double beginDuration = (double)Beginning.clip.samples /
89         Beginning.clip.frequency;
90     double gasDuration = (double)Gas.clip.samples / Gas.clip.frequency;
91     double startTime = AudioSettings.dspTime;
92     Beginning.PlayScheduled(startTime);
93     Gas.PlayScheduled(startTime + beginDuration);
94 }

```

For the handling of the different objects in the tutorial scene, the code 7 was used. Here it can be seen how the location of the objects of interest is obtained, the method used to place them in the correct positions, and the implementation of audio signals when the tasks to be performed are completed. It should be noted that, in the same way as in the previous code, a modified version of the code is added with only the most representative algorithms.

Code 7.2: Tutorial item management algorithm (modified)

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Tutorial : MonoBehaviour
6 {
7     //Allow to access the GameObjects and positions relevant for the script
8     [SerializeField]
9     private GameObject Test1;
10    [SerializeField]
11    private GameObject Test2;
12    [SerializeField]
13    private GameObject RightPos1;
14    [SerializeField]
15    private GameObject RightPos2;
16    [SerializeField]
17    private GameObject Continue;
18    [SerializeField]
19    private GameObject ContinueArrow;
20    [SerializeField]
21    private GameObject RedCheck;
22    [SerializeField]
23    private GameObject BlueCheck;
24    [SerializeField]
25    private GameObject AudioScene;
26    [SerializeField]
27    private GameObject Trigger;
28    public bool InRightPosition1;
29    public bool InRightPosition2;
30    private Vector3 Pos1;
31    private Vector3 RightP1;
32    private Vector3 Pos2;
33    private Vector3 RightP2;
34    private Vector3 InitialPosRed;
35    private Vector3 InitialPosBlue;
36    public AudioClip CompleteEffect;
37    public AudioSource mainAudio;
38
39    //Adds delay to let scene load correctly
40    public float delay;
41
42    //These bools are used to perform an action only once
43    bool Check1Sound;
44    bool Check2Sound;
45
46    void Start()
47    {
48        //Set the initial conditions for the tutorial to take place
49        Continue.SetActive(false);
50        ContinueArrow.SetActive(false);
51        RightP1 = RightPos1.transform.position;
52        RightP2 = RightPos2.transform.position;
53        InitialPosRed = new Vector3(1.63f, 0.575999975f, 2.22399998f);
54        InitialPosBlue = new Vector3(1.63900006f, 0.574999988f, 2.43700004f);
55        RedCheck.SetActive(false);
56        BlueCheck.SetActive(false);
57        Trigger.SetActive(true);
58    }
59
60    void Update()
61    {

```

```

62 //Keeps track of where the relevant GameObjects are
63 Pos1 = Test1.transform.position;
64 Pos2 = Test2.transform.position;
65
66 //If one of the objects approaches the final position, changes it to the
67 //right position
68 if (Vector3.Distance(Pos1, RightP1) < 0.1f)
69 {
70     Test1.transform.position = RightP1;
71     Test1.transform.rotation = Quaternion.Euler (0,0,0);
72     RedCheck.SetActive(true);
73     Test1.GetComponent<BoxCollider>().enabled = false;
74     Test1.GetComponent<Rigidbody>().isKinematic = true;
75     Test1.GetComponent<Rigidbody>().useGravity = false;
76 }
77
78 if (Vector3.Distance(Pos2, RightP2) < 0.1f)
79 {
80     Test2.transform.position = RightP2;
81     Test2.transform.rotation = Quaternion.Euler(0, 0, 0);
82     BlueCheck.SetActive(true);
83     Test2.GetComponent<SphereCollider>().enabled = false;
84     Test2.GetComponent<Rigidbody>().isKinematic = true;
85     Test2.GetComponent<Rigidbody>().useGravity = false;
86 }
87
88 //If both objects are in the right position and their attributes have been
89 //modified, the user can move to the next scene
90 if (Test1.GetComponent<Rigidbody>().useGravity != true &&
91     Test2.GetComponent<Rigidbody>().useGravity != true)
92 {
93     Trigger.SetActive(false);
94     Continue.SetActive(true);
95     ContinueArrow.SetActive(true);
96 }
97 //Invokes the method CompleteSound
98 Invoke("CompleteSound", 0f);
99
100 //This method checks if the objects are in the right place, and if so, let the
101 //user know by playing the "Completed Task" sound
102 void CompleteSound()
103 {
104     if (!Check1Sound && RedCheck.activeSelf)
105     {
106         AudioSource.PlayClipAtPoint(CompleteEffect, transform.position, 1f);
107         Check1Sound = true;
108     }
109     if (!Check2Sound && BlueCheck.activeSelf)
110     {
111         AudioSource.PlayClipAtPoint(CompleteEffect, transform.position, 1f);
112         Check2Sound = true;
113     }
114 }

```

Although it bears some similarities to the code shown above for object handling in the tutorial, the major difference between the object handling in the tutorial with respect to the operating room (and the scenes derived from it) is the implementation of this type of code on a per-object basis. This means that all objects handle their set of functions completely individually, and although the algorithms used are similar to each other, the set of objects and conditions that each script handles varies greatly from object to object.

Additionally, there is a mother script, called Manager, which, as its name indicates, manages the different states of the objects and allows communication between the scripts themselves. This allows to handle individually the parameters and prevent cascading failures, as would be the case if everything was implemented in a single script, as it was done in the tutorial scene.

Code 7.3: Surgery Room item management algorithm (modified)

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using Valve.VR;
5
6 public class RightPositionT2 : MonoBehaviour
7 {
8     //Creates the Vectors to store the right positions and final positions of the
9     //GameObjects
10    private Vector3 RightP;
11    private Vector3 RightPG;
12    private Vector3 PosInitialG;
13    private Vector3 PosFinalG;
14    public bool InRightPosition;
15
16    //Gets the inputs from the controllers through the SteamVR plugin
17    public SteamVR_Input_Sources LeftInputSource = SteamVR_Input_Sources.LeftHand;
18    public SteamVR_Input_Sources RightInputSource = SteamVR_Input_Sources.RightHand;
19
20    //These methods calls all the scene Objects whose state are linked to the Audio
21    //Guide
22    [SerializeField]
23    private GameObject CheckT2;
24    [SerializeField]
25    private GameObject Graspers;
26    [SerializeField]
27    private GameObject GraspersCheck;
28    [SerializeField]
29    private GameObject GraspersText;
30
31    void Start()
32    {
33        //Set the initial conditions for this item in particular and the objects
34        //linked to it
35        RightP = new Vector3(-0.549500048f, 0.94660002f, -0.106500044f);
36        RightPG = new Vector3(-0.570900023f, 0.794700027f, -0.209399998f);
37        PosInitialG = Graspers.transform.position;
38        GraspersText.SetActive(false);
39        Graspers.SetActive(false);
40        PosFinalG = new Vector3(-0.568599999f, 0.778199971f, -0.212300003f);
41    }
42
43    void Update()
44    {
45        //If the user presses the trigger of either the right or left controller
46        //then actions can take place
47        if (SteamVR_Actions._default.Squeeze.GetAxis(RightInputSource) < 1 &&
48            SteamVR_Actions._default.Squeeze.GetAxis(LeftInputSource) < 1)

```

```

44     {
45         //If the object approaches the final position, changes it to the right
46         //position, and activates and deactivates the linked objects
47         if (Vector3.Distance(transform.position, RightP) < 0.1f)
48         {
49             transform.position = RightP;
50             InRightPosition = true;
51             transform.rotation = Quaternion.Euler(0, -90, 90);
52             gameObject.GetComponent<BoxCollider>().enabled = false;
53             gameObject.GetComponent<Rigidbody>().isKinematic = true;
54             gameObject.GetComponent<Rigidbody>().useGravity = false;
55             CheckT2.SetActive(true);
56             GraspersText.SetActive(true);
57             Graspers.SetActive(true);
58         }
59         //If the user drops the object and it is not near the final position,
60         //goes back to the initial position
61         else
62         {
63             transform.position = new Vector3(0.75999999F, 0.513000011F,
64             -1.31299996F);
65             transform.rotation = Quaternion.Euler(0, 90, 0);
66         }
67         if (Vector3.Distance(Graspers.transform.position, PosFinalG) < 0.1f)
68         {
69             Graspers.transform.position = RightPG;
70             Graspers.transform.rotation = Quaternion.Euler(270, 11.67f, 0);
71             Graspers.GetComponent<Rigidbody>().useGravity = false;
72             Graspers.GetComponent<Rigidbody>().isKinematic = true;
73             Graspers.GetComponent<BoxCollider>().enabled = false;
74             GraspersCheck.SetActive(true);
75             GraspersText.SetActive(false);
76         }
77         else
78         {
79             Graspers.transform.position = new Vector3 (0.463f, 0.526000023f,
80             -1.42900002f);
81             Graspers.transform.rotation = Quaternion.Euler(0,0,-90f);
82         }
83     }
84 }

```



One of the most important codes is the implementation of BZ Slicer (see Section 4.2.6). One of the major problems in implementing this code is its activation. The base code receives inputs from the traditional systems of a computer: a mouse and a keyboard. It is for this reason that the 7 code was developed. In this code, the Valve.VR callback actions are used to obtain inputs from the VR controls. This is how the algorithm is created, stating that only when the user presses the trigger button in its entirety will the BZ Slicer script be activated. The BzSlicer creates new objects with the split meshes, so the algorithm uses the names that are created when a cut is made, shutting down the script as soon as the first cut is detected.

Code 7.4: Activation of the slicer script once (unmodified)

```

1  using BzKovSoft.ObjectSlicer.Samples;
2  using System.Collections;
3  using System.Collections.Generic;
4  using UnityEngine;
5  using Valve.VR;
6
7
8  public class Translation : MonoBehaviour
9  {
10     //Gets the inputs from the controllers through the SteamVR plugin
11     public SteamVR_Input_Sources LeftInputSource = SteamVR_Input_Sources.LeftHand;
12     public SteamVR_Input_Sources RightInputSource = SteamVR_Input_Sources.RightHand;
13
14     //Allow to access the GameObjects and positions relevant for the script
15     [SerializeField]
16     private GameObject Scissors;
17     [SerializeField]
18     private GameObject ScissorsMesh;
19     [SerializeField]
20     private Transform RayDirection;
21
22     private void Update()
23     {
24         //When the user removes the fat that is covering the gallbladder, this
25         //function comes into place
26         if (GameObject.Find("Fat 1") == null)
27         {
28             //When the user presses the trigger on the right hand it activates the
29             //slicer scripts
30             if (SteamVR_Actions._default.Squeeze.GetAxis(RightInputSource) >= 1)
31             {
32                 Scissors.GetComponent<SampleMouseSlicer>().enabled = true;
33
34                 //If any of the new meshes appears, deactivates it, indicating that
35                 //the cut has been made only once
36                 if (GameObject.Find("Digestive_Gall_Bladder_Geo_pos") != null ||
37                     GameObject.Find("Digestive_Gall_Bladder_Geo_neg_pos") ||
38                     GameObject.Find("Digestive_Gall_Bladder_Geo_pos_pos_neg") ||
39                     GameObject.Find("Digestive_Gall_Bladder_Geo_neg_neg_pos") ||
40                     GameObject.Find("Digestive_Gall_Bladder_Geo_pos_neg") ||
41                     GameObject.Find("Digestive_Gall_Bladder_Geo_neg_neg_neg_neg"))
42                 {
43                     Scissors.GetComponent<SampleMouseSlicer>().enabled = false;
44                 }
45             }
46         }
47     }
48 }

```

The algorithm presented in code 7 is the same that is implemented for all the different sources of information in the virtual library. In essence, the void functions are assigned to the buttons located in the scene in order to turn off the menu objects and display the information required by the user. In the same way, when the user has finished using the information, with the second method present in the code, he can return to the original state of the virtual library.

Code 7.5: Show information for the virtual library

```

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class GetInfoHarmonic : MonoBehaviour
6 {
7     //Allow to access the GameObjects and positions relevant for the script
8     [SerializeField]
9     private GameObject Light;
10    [SerializeField]
11    private GameObject HarmonicGroup;
12    [SerializeField]
13    private GameObject Tools;
14    [SerializeField]
15    private GameObject Procedure;
16
17
18    void Start()
19    {
20        //Set the initial conditions for the virtual library and this particular
21        //piece of information
22        Light.SetActive(false);
23        HarmonicGroup.SetActive(false);
24    }
25    void Update()
26    {
27        //Linked to the respective buttons, these actions activates the objects that
28        //contain the relevant information or deactivates them
29    public void showInfo()
30    {
31        Light.SetActive(true);
32        HarmonicGroup.SetActive(true);
33        Tools.SetActive(false);
34        Procedure.SetActive(false);
35    }
36    public void Exit()
37    {
38        Light.SetActive(false);
39        HarmonicGroup.SetActive(false);
40        Tools.SetActive(true);
41        Procedure.SetActive(true);
42    }
43
44 }

```

# Appendix G. Case Study Raw Data

Table 7.4: Case Study Raw Data as imported to Excel from Lime Survey

Response ID	Date submitted	Start language	Seed	VR Simulation	Question #1	Question #2	Question #3	Question #4	Question #5	Question #6
					What is the main objective of a Cholecystectomy?	What is the name of duct that connect the gallbladder and the liver?	What is the preferred kind of tool to dissect in MIS?	What is the gas used to insufflate the patient?	What is the ideal internal pressure when insufflating gas into a patient?	What is the purpose of the trocar?
1	06.03.23	en	735345123	Yes	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
2	07.03.23	en	1476373211	Yes	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
3	07.03.23	en	1531043946	Yes	Removal of the gallbladder	Gallbladder Duct	High Frequency Cutter	CO2	8-12 mmHg	Generate access to the abdomen
4	08.03.23	en	735550109	No	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
5	08.03.23	en	2050848300	Yes	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
6	09.03.23	en	1305916560	No	Removal of the gallbladder	Cystic Duct	High Frequency Cutter	CO2	8-12 mmHg	Generate access to the abdomen
7	09.03.23	en	272641044	No	Removal of the gallbladder	Cystic Duct	High Frequency Cutter	CO2	8-12 mmHg	Generate access to the abdomen
8	09.03.23	en	2038092737	Yes	Removal of the gallbladder	Bile Duct	High Frequency Cutter	CO2	8-12 mmHg	Generate access to the abdomen
9	09.03.23	en	309590831	Yes	Removal of the gallbladder	Cystic Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
10	09.03.23	en	1056374965	Yes	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
11	09.03.23	en	846533371	No	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
12	10.03.23	en	135655621	Yes	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
13	14.03.23	en	1933962584	Yes	Removal of the gallbladder	Gallbladder Duct	High Frequency Cutter	CO2	8-12 mmHg	Generate access to the abdomen
14	14.03.23	en	1270112608	No	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
15	14.03.23	en	1034389296	No	Removal of the gallbladder	Bile Duct	High Frequency Cutter	CO2	8-12 mmHg	Cut down the gallbladder
16	14.03.23	en	309666770	No	Removal of the gallbladder	Bile Duct	High Frequency Cutter	CO2	8-12 mmHg	Generate access to the abdomen
17	14.03.23	en	25430067	Yes	Removal of the gallbladder	Cystic Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
18	14.03.23	en	142508910	No	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate light inside the abdominal cavity to observe the point of operation
19	14.03.23	en	1123115691	Yes	Removal of the gallbladder	Cystic Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
20	14.03.23	en	67172775	No	Removal of the gallbladder	Cystic Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Cut down the gallbladder
21	17.03.23	en	938612607	Yes	Removal of the gallbladder	Cystic Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
22	17.03.23	en	1251961512	No	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
23	17.03.23	en	1462621288	No	Destroy gallstones	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Create a videofeed to be able to perform the surgery
24	17.03.23	en	429235867	Yes	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen
25	17.03.23	en	188910324	No	Removal of the gallbladder	Bile Duct	Ultrasound Scalpel	CO2	8-12 mmHg	Generate access to the abdomen

Response ID	Date submitted	Start language	Seed	VR Simulation	Question #7	Question #8	Question #9	Question #10	Question #11
					How does the harmonic scalpel work?	What is the purpose of an endoscope?	Where is the point where the cut should be made?	How does the surgeon/patient get rid of the excess gas?	For what purpose is it use hot water or alcohol before using the endoscope?
1	06.03.23	en	735345123	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Options 1&2	Match the body temperature
2	07.03.23	en	1476373211	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Respiration	Match the body temperature
3	07.03.23	en	1531043946	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Options 1&2	Match the body temperature
4	08.03.23	en	735550109	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Between the liver and the gallbladder	Respiration	Remove the dust from the lens
5	08.03.23	en	2050848300	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Respiration	Match the body temperature
6	09.03.23	en	1305916560	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Options 1&2	Sterilize the equipment before the surgery
7	09.03.23	en	272641044	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Respiration	Sterilize the equipment before the surgery
8	09.03.23	en	2038092737	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	None of the above	Match the body temperature
9	09.03.23	en	309590831	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Respiration	Match the body temperature
10	09.03.23	en	1056374965	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Options 1&2	Match the body temperature
11	09.03.23	en	846533371	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Respiration	Match the body temperature
12	10.03.23	en	135655621	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Options 1&2	Match the body temperature
13	14.03.23	en	1933962584	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Options 1&2	Match the body temperature
14	14.03.23	en	1270112608	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Respiration	Reduce the friction to insert it in the body
15	14.03.23	en	1034389296	No	Applying current to a piezoelectric crystal, and using the vibration from it	Generate access to the abdomen	Between the liver and the gallbladder	Respiration	Sterilize the equipment before the surgery
16	14.03.23	en	309666770	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Respiration	Match the body temperature
17	14.03.23	en	25430067	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Cut down the gallbladder	Carrots Triangle	Respiration	Match the body temperature
18	14.03.23	en	142508910	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Insufflation Machine	Match the body temperature
19	14.03.23	en	1123115691	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Respiration	Match the body temperature
20	14.03.23	en	67172775	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Respiration	Match the body temperature
21	17.03.23	en	938612607	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Respiration	Match the body temperature
22	17.03.23	en	1251961512	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Respiration	Match the body temperature
23	17.03.23	en	1462621288	No	Applying current to a piezoelectric crystal, and using the vibration from it	Generate access to the abdomen	Carrots Triangle	Insufflation Machine	Match the body temperature
24	17.03.23	en	429235867	Yes	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Calots Triangle	Options 1&2	Match the body temperature
25	17.03.23	en	188910324	No	Applying current to a piezoelectric crystal, and using the vibration from it	Create a videofeed to be able to perform the surgery	Carrots Triangle	Options 1&2	Match the body temperature

## Appendix H. Questionnaire used for the case study



What is the main objective of a Cholecystectomy?

Choose one of the following answers

- Destroy kidney stones
- Removal of the gallblader
- Destroy gallstones
- Removal of the appendix

What is the purpose of an endoscope?

Choose one of the following answers

- Generate access to the abdomen
- Grab the gallblader
- Create a videofeed to be able to perform the surgery
- Cut down the gallbladder

What is the name of duct that connect the gallbladder and the liver?

This is a question help text.

Choose one of the following answers

- Liver Duct
- Bile Duct
- Cystic Duct
- Gallblader Duct

Where is the point where the cut should be made?

Choose one of the following answers

- Calots Triangle
- Carnots Triangle
- Between the liver and the gallblader
- Between the bile duct and the cystic artery

What is the preferred kind of tool to dissect in MIS?

Choose one of the following answers

- Ultrasound Scalpel
- Laser
- High Frequency Cutter
- Blunt tools

How does the surgeon/patient get rid of the excess gas?

Choose one of the following answers

- Respiration
- Insuflation Machine
- Options 1&2
- None of the above

What is the gas used to insufflate the patient?

Choose one of the following answers

- Oxygen
- Inert Gas
- CO2
- Nitrogen Gas

For what purpose it is use hot water or alcohol before using the endoscope?

Choose one of the following answers

- Match the body temperature
- Sterilize the equipment before the surgery
- Remove the dust from the lens
- Reduce the friction to insert it in the body

What is the ideal internal pressure when insufflating gas into a patient?

Choose one of the following answers

- 13-17 mmHg
- 1-4 mmHg
- 5-7 mmHg
- 8-12 mmHg

How does the harmonic scalper work?

Choose one of the following answers

- Generating electrical pulse to cut down the tissue
- Applying current to a piezoelectric crystal, and using the vibration from it
- Heating up with electricity the end of it, to dissect inside the patient's body

What is the purpose of the trocar?

● Choose one of the following answers

- Create a videofeed to be able to perform the surgery
- Generate access to the abdomen
- Generate light inside the abdominal cavity to observe the point of operation
- Cut down the gallbladder