MetaMed VR: Enhancing Medical Education through Immersive Virtual Reality

Project Team

Dawood Sarfraz	20P-0153
Waleed Akram	20P-0640
Shoaib Akhtar	20P-0147

Session 2020-2024

Supervised by

Dr. Muhammad Amin

Department of Computer Science

National University of Computer and Emerging Sciences
Peshawar, Pakistan

June, 2024

Student's Declaration

We declare that this project titled "*MetaMed VR: Enhancing Medical Education through Immersive Virtual Reality*", submitted as requirement for the award of degree of Bachelors in Bachelors in Computer Science, does not contain any material previously submitted for a degree in any university; and that to the best of our knowledge, it does not contain any materials previously published or written by another person except where due reference is made in the text.

We understand that the management of Department of Computer Science, National University of Computer and Emerging Sciences, has a zero tolerance policy towards plagiarism. Therefore, We, as authors of the above-mentioned thesis, solemnly declare that no portion of our thesis has been plagiarized and any material used in the thesis from other sources is properly referenced.

We further understand that if we are found guilty of any form of plagiarism in the thesis work even after graduation, the University reserves the right to revoke our BS degree.

Dawood Sarfraz	Signature: ________________
Waleed Akram	Signature: ________________
Shoaib Akhtar	Signature: ________________

____

Verified by Plagiarism Cell Officer
Dated:

Certificate of Approval

The Department of Computer Science, National University of Computer and Emerging Sciences, accepts this thesis titled *MetaMed VR: Enhancing Medical Education through Immersive Virtual Reality*, submitted by Dawood Sarfraz (20P-0153), Waleed Akram (20P-0640), and Shoaib Akhtar (20P-0147), in its current form, and it is satisfying the dissertation requirements for the award of Bachelors Degree in Bachelors in Computer Science.

Supervisor

Dr. Muhammad Amin Signature: ________________

Mr. Haroon Zafar

FYP Coordinator
National University of Computer and Emerging Sciences, Peshawar

Mr. Fazl-e-Basit

HoD of Department of Computer Science
National University of Computer and Emerging Sciences

Acknowledgements

This project would not have been possible without the support of many dedicated individuals. Our deepest gratitude goes to our advisor, Dr. Muhammad Amin, for his invaluable feedback and unwavering guidance throughout the numerous revisions. We are also profoundly thankful to our coordinator, Mr. Haroon Zafar, for his continuous support and encouragement.

We extend our sincere thanks to the entire Computer Science faculty for their mentorship and valuable time. Additionally, we are grateful to the National University of Computer and Emerging Sciences for providing the opportunity and financial support necessary to bring this project to fruition.

Dawood Sarfraz

Waleed Akram

Shoaib Akhtar

Abstract

The rapid evolution of technology continues to redefine traditional educational paradigms. This report delves into the pioneering exploration of leveraging Immersive Virtual Reality (VR) to transform medical education. Focused on the project **" Enhancing Medical Education through Immersive Virtual Reality (MetaMed) "** our study aims to revolutionize the pedagogical landscape of medical training.

In the initial phase of our project, we embarked on a journey to harness cutting-edge VR technology. The objective was to create immersive learning environments that simulate real-world medical scenarios, allowing students to engage in interactive, hands-on learning experiences. Through the integration of advanced VR techniques, our project aspires to bridge the gap between theoretical knowledge and practical application, fostering a deeper understanding of medical concepts.

Contents

List of Figures

Chapter 1

Introduction

- Metaversion is a virtual universe that integrates VR, AR, and the Internet, providing a shared, immersive, and often three-dimensional space for online interaction and presence.

- The Metaverse is a virtual world where users interact with digital avatars that reflect their identity and physical characteristics.

- Metaversion, incorporating virtual offices, classrooms, and training environments, offers new opportunities for learning, skill development, and remote collaboration in both entertainment and education.

1.1 History of the Metaverse

Metaversion, also known as collective virtual shared space or virtual reality space, has a rich history rooted in science fiction, technology, and online communities. Here's a brief overview of the meta version's history:

1.1.1 1970 - Sci-Fi Origins:

Neal Stephenson's 1992 novel "Snow Crash" popularized the concept of metaversion, a virtual reality space where users interact as avatars, serving as a conceptual basis for later developments.

1.1.2 Early virtual worlds:

In the late 1970s and 1980s, MUDs and MOOs were online environments that enabled multiple users to interact in text-based virtual worlds, paving the way for modern online communities.

1.1.3 1990 - Emergence of 3D virtual worlds:

In the 1990s, 3D virtual worlds like Second Life emerged, enabling users to create avatars, socialize, and build virtual environments, paving the way for immersive metaverse experiences.

1.1.4 2000 - Online Games and Social Media:

Online interactions and communities are transforming MMOs like World of Warcraft and social media platforms like Facebook and Twitter, fostering connected digital spaces.

1.1.5 2010 - VR and AR:

Virtual and augmented reality technologies, like Oculus Rift and HTC Vive, are revolutionizing metaversion by providing immersive virtual spaces through smartphones and AR glasses.

1.1.6 Current developments:

As of September 2021, companies are developing meta-related technologies like virtual reality, augmented reality, blockchain, and decentralized platforms, using NFTs to represent digital assets within the metaverse.

1.2 The Future of the Metaverse

Metaversion, a future of immersive virtual and augmented reality experiences, blockchain-based asset ownership, and economic opportunities, is expected to revolutionize various sectors, redefining social dynamics, raising concerns about privacy, security, regulation, and environmental sustainability.

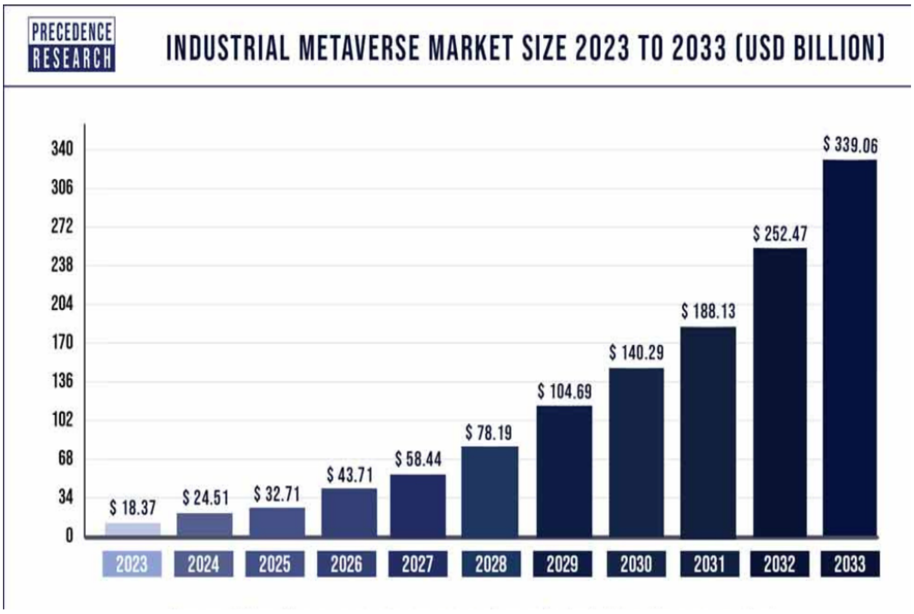

Figure 1.1: Future of Metaverse
[26]

1.3 How Metaverse Would Excel

The success of the metaverse will be influenced by several key factors:

Technology Advances:

The metaverse's success relies on advancements in technologies like VR, AR, AI, and high-speed internet connectivity, along with enhanced hardware and software for immersive virtual experiences.

User Adoption:

The metaverse's success relies on a critical mass of active users, with user-friendly interfaces and accessibility playing a crucial role in attracting a diverse user base.

Content and Creativity:

The Metaverse requires a diverse range of engaging content, experiences, and activities, including virtual worlds, games, social spaces, art, and educational content, for easy creation and sharing.

Interoperability and standards:

Open standards and protocols for the metaverse are crucial for creating a seamless, accessible environment by facilitating interoperability between virtual worlds and platforms.

Security and Privacy:

Users must rely on metaverse for their personal data and transactions, requiring robust security measures, data protection, and privacy controls to prevent abuse, fraud, and unauthorized access.

Economic Viability:

Metaversion necessitates a sustainable economy, attracting users and content creators through virtual real estate, digital currencies, and employment opportunities, utilizing blockchain technology and NFTs.

Regulation and governance:

Governments and regulators will need to create clear guidelines and legal frameworks for meta

Chapter 2

Literature overview

This literature review explores the integration of metaversion in healthcare, highlighting its potential for innovative solutions such as medical training, telemedicine, mental health therapy, and patient empowerment. It examines historical perspectives, technological foundations, practical applications, ethical considerations, and future prospects.

2.1 All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda [21][1]

Methodology:

The article explores the metaverse's evolution, highlighting core technologies like augmented reality, AI, and blockchain, user-centric factors like avatars, content creation, and virtual economies, and proposes a research agenda.

Limitations:

The article offers a broad overview of metaverse evolution, but lacks in-depth analysis, discusses challenges like ethical issues and inequality, and relies on a survey-based approach without empirical support.

Results:

Comprehensive framework for examining metaverse development. Concrete research agenda for metaverse development

2.2 Healthcare in metaverse: A survey on current metaverse applications in healthcare[16][7]

Methodology:

The article explores metaverse applications in healthcare, highlighting recent technological advancements, challenges, and potential improvements, emphasizing the need for sustainable, innovative solutions..

Limitations:

Existing healthcare systems limitations revealed during COVID-19 pandemic. Surge in healthcare innovation using virtual environments for alternative systems.

Results:

The metaverse for healthcare requires addressing connectivity, privacy, security, integration, interoperability, user experience, and technical issues with VR and AR technologies.

2.3 Medical Metaverse: Technology, Applications, Challenges and Future[28]

Methodology

The paper explores the use of metaverse technologies in healthcare, identifying challenges and proposing solutions for efficient diagnosis, education, and treatment in healthcare settings.

Limitations:

The paper provides a thorough overview of the Metaverse's potential in healthcare, highlighting its applications, technologies, and challenges, but lacks case studies and real-world implementations due to limited literature.

Results:

Review of technologies and applications of the metaverse. Exploration of potential and future direction in healthcare.

2.4 The Metaverse for Healthcare: A Survey of Potential Applications, Challenges, and Future Directions.[33][10]

Methodology:

The paper explores the use of the Metaverse in healthcare, highlighting applications, technologies, and projects, while identifying challenges and proposing future research directions, focusing on AI, VR, AR, IoMT, robotics, and quantum computing.

Limitations:

Adopting AI-enabled Metaverse in healthcare faces challenges like high-speed communication, massive computation, security concerns, data loss, heterogeneity of devices, and cost implications, impacting its comprehensiveness.

Results:

Medical education uses VR for learning body structures, improving future doctors' quality. However, AI-enabled Metaverse risks patient privacy and ethical issues, potentially leading to medical errors. Edge computing for real-time data retrieval requires additional devices, posing network scalability challenges.

2.5 Revolutionizing Medical Education with Metaverse[17][11]

Methodology:

Interdependence of characteristics in virtual teaching model. Overcoming obstacles through technological advancements in education.

Limitations:

The paper neglects to address technical challenges, adoption barriers, ethical considerations, and perspectives of educators, healthcare providers, and patients in integrating metaverse technology in medical education, including cost and infrastructure.

Results:

The paper focuses on methods for revolutionizing medical education using metaverse. It discusses the potential of metaverse in providing interactive and immersive experiences in healthcare.

2.6 The Metaverse in Medical Education and Clinical Practice[23]

Methodology:

Use of immersive virtual and augmented environments. Utilization of different models of stereoscopic vision glasses

Limitations:

No specific limitations mentioned in the abstract. Further details on limitations not provided in the text.

Results:

The paper presents immersive virtual and augmented environments for medical training. These environments use stereoscopic vision glasses to create a metaverse for enhanced medical training.

2.7 Virtual reality and the transformation of medical education[25]

Methodology:

The paper discusses the use of simulation in clinical training but highlights its resource-intensive nature. It advocates for the adoption of virtual reality (VR) in medical education, highlighting its effectiveness in healthcare and its potential for providing quality, geographically independent training.

Limitations:

The paper generalizes VR's effectiveness in medical education without addressing specific limitations or challenges, lacks in-depth analysis of barriers, and overlooks technological constraints crucial for real-world implementation, presenting a one-sided view and overlooking areas for further research.

Results:

Digital transformation has a great impact on medical education. Inclusion of AI and VR benefits medical students.

2.8 A Virtual Reality for the Digital Surgeon[32]

Methodology:

The paper reviews VR technology's potential in healthcare, surgical education, and support tools, highlighting its potential to optimize patient data, improve surgical outcomes, and revolutionize healthcare.

Limitations:

The paper's limited scope may limit its generalizability, overlooking challenges in implementing VR in healthcare. Additionally, reliance on a limited dataset or specific sources could affect the depth and reliability of the findings, potentially affecting the robustness of conclusions about VR's impact on surgical education and clinical practice.

Results:

VR has the potential to improve surgical education and skills. VR can optimize preoperative planning and intraoperative support in clinical practice.

2.9 Transforming medical education and training with VR using M.A.G.E.S.[6]

Methodology:

The paper introduces a new VR software system for healthcare training, specifically Psychomotor Virtual Reality Surgical Training, which enhances surgeon skills through gamification and advanced interactability, and supports multiple surgeons and assistants.

Limitations:

The paper primarily discusses orthopedic surgeries, neglecting generalizability, cost-effectiveness, VR surgical training solution validation, integrated educational curriculum effectiveness, long-term skill retention, and user experience feedback.

Results:

The paper focuses on methods for revolutionizing medical education using metaverse. It discusses the potential of metaverse in providing interactive and immersive experiences in healthcare.

2.10 Virtual Reality in Medicine[18][9]

Methodology

Multimodal interactions between user and virtual environment. Technical requirements and design principles of input devices, displays, and rendering techniques.

Limitations:

Physiological constraints. Technical requirements and design principles of multimodal input devices, displays, and rendering techniques.

Results:

Examples of virtual reality applications in surgical training, intra-operative augmentation, and rehabilitation. Provides technical requirements and design principles for virtual reality in medicine.

2.11 A Virtual Environment for Training and Assessment of Surgical Teams[24]

Methodology:

The paper explores the use of Collaborative Virtual Environments (CVEs) in surgical team training and assessment to enhance remote interactions and improve teamwork skills. The proposed CVE architecture supports team training and assessment in surgical simulations, reducing costs and requiring live subjects.

Limitations:

Cost reduction for training is a limitation. Use of guinea pigs and anatomical specimens is reduced.

Results:

Proposed architecture for training and assessing team skills during surgery. Use of statistical models to monitor and assess team performance.

2.12 Virtual reality technology and its application in modern medicine [18]

Methodology:

The paper discusses the use of virtual reality (VR) technology in medical applications such as surgery, telemedicine, and patient education, comparing it to traditional methods and highlighting its advantages and limitations.

Limitations:

Limited access to VR technology in medical settings. Challenges in integrating VR into existing medical practices.

Results:

VR applied in virtual human, assisted diagnosis, surgery simulation. VR used in virtual telemedicine for medical purposes.

2.13 Role of virtual reality for healthcare education [13]

Methodology:

The research paper explores the use of VR technology in healthcare education, highlighting its role in storing patient data as 3D points, enhancing clinical skills, and providing flexible training for practitioners, ultimately improving the quality and standards of medical education.

Limitations:

Few are using VR to evaluate medical students' success. VR technology not widely used for medical training evaluation.

Results:

VR improves learning and training of medical practitioners. VR enhances comprehension of anatomy and clinical outcomes.

2.14 Next-Gen Mulsemedia: Virtual Reality Haptic Simulator's Impact on Medical Practitioner for Higher Education Institutions[12][14][5]

Methodology:

The study used the core motivation hypothesis to boost motivation in the classroom. The study used the attention, relevance, confidence, and satisfaction (ARCS) model to analyze the impact of virtual reality on student motivation and content update implementation.

Limitations:

Lack of research on consequences of virtual reality Early stage of virtual reality technology research.

Results:

Virtual reality simulators improve student motivation and learning. VR has the potential to transform medical education.

2.15 Design and implementation of a 3D digestive teaching system based on virtual reality technology in modern medical education[8]

Methodology:

DX technologies: VR, AR, MR, XR, 3D images, holograms, AI. Utilization of HMDs, wearable sensors, 5G, and Wi-Fi.

Limitations:

Lack of systematic methodology, affecting evidence level Heterogeneity in XR techniques studies, limiting systematic reviews.

Results:

2.16 A Virtual Operating Room for Context-Relevant Training [4]

Methodology:

Virtual agents with unique personalities and knowledge structures defined. Immersive Virtual Operating Room (VOR) simulating surgical procedures described

Limitation:

Current medical simulators lack addressing errors in healthcare system. Existing simulators focus on procedural skills, not team dynamics.

Results:

Pilot session with surgical resident unfamiliar with VOR showed challenges. VOR allows team-based surgical training with virtual expert agents.

2.17 Virtual reality surgical training and assessment system[2][3]

Methodology:

Soft-tissue model based on volumetric mass-spring system Texture mapping for realistic organ representation and space perception.

Limitation:

Limits of realism in surgical simulation Challenges in obtaining reliable measures of surgical skills.

Results:

VR surgical system based on C-source code and OpenGL. System handles accurate interactions between soft-tissue and surgical instruments

Chapter 3

Project Vision

3.1 Problem Statement

The project aims to provide medical students with hands-on experience through a virtual medical training platform, leveraging the metaverse to provide realistic scenarios and risk-free practice opportunities, overcoming challenges like limited availability and high costs.

3.2 Goals

The project aims to create a metaverse-based medical training platform, offering a risk-free environment for medical students to practice skills, enhance learning, and reduce costs and ethical concerns.

3.3 Project Scope

The project aims to create a metaverse-based medical training environment, incorporating realistic simulations, haptic feedback devices, and hands-on practice for various medical specialties.

3.4 Limitations

The project faces constraints such as budget constraints, metaverse platform technology limitations, ethical standards, time constraints, and regulatory compliance.

3.5 Stakeholders

Key stakeholders in this project include medical students, physician educators, healthcare institutions, and potentially patients who may benefit from better trained medical professionals.

Chapter 4

Software Requirements Specification

This chapter will contain the functional and non-functional requirements of the project.

4.1 List of functions

List of key features for the "Improving Medical Education through Immersive Virtual Reality (MetaMed)" project:

Realistic medical simulations:

Create highly realistic medical scenarios and procedures that faithfully mimic real medical experiences, including various medical specialties.

Haptic feedback integration:

Integrate tactile feedback devices to provide tactile sensations, increasing the realism of medical training and allowing students to develop their tactile skills.

Immersive 3D Environments:

Create immersive, 3D virtual environments that accurately represent medical settings, including operating rooms, medical instruments, and anatomical structures.

Interactivity:

Enables hands-on interaction with virtual medical tools, instruments, and equipment, allowing students to perform medical tasks and procedures.

Remote Access:

Enable medical students to access a metaverse-based medical training platform from remote locations, facilitating flexible learning.

4.2 Test plan (test level, test techniques)

Test plan for our project will be as:

- Movement in VR
- Interaction with Objects
- Treatement to Patient

4.3 Use Case Diagram

Here is the Use Case Diagram of the Project.

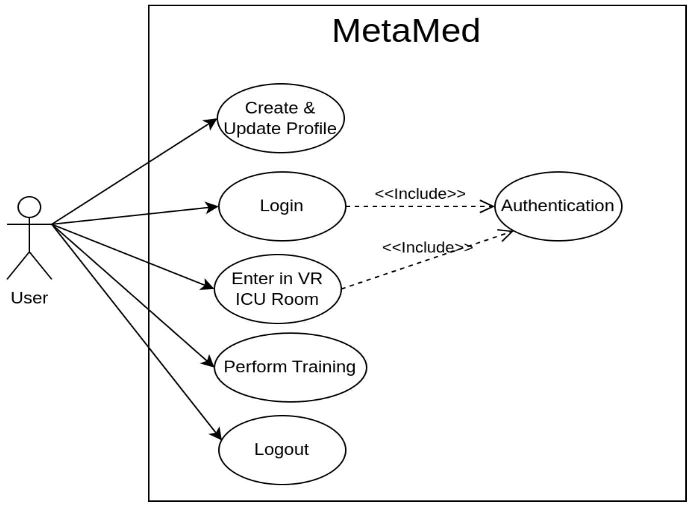

Figure 4.1: Use Case Diagram

4.4 Software Development Plan

Software Development Plan for our project will be as:

- Project Overview
- Requirements Gathering and Analysis
- System Design
- Development

- Testing
- Documentation
- Management

4.5 System Diagram

It is the System Diagram of the Project.

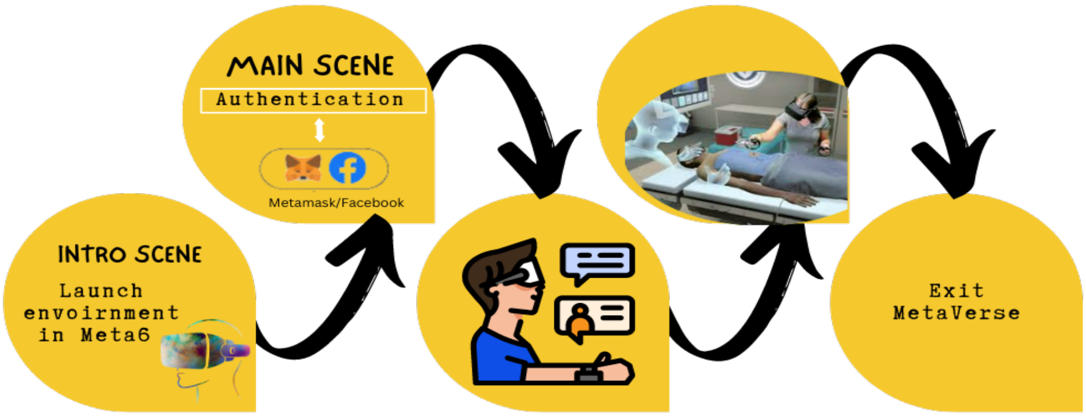

Figure 4.2: System Diagram

4.6 Activity Daigram

The activity diagram aims to illustrate the entire process of project working and demonstration.

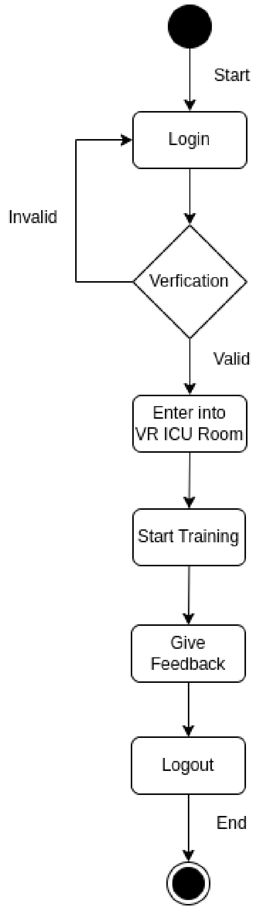

Figure 4.3: Activity Diagram

4.7 Tools and Technologies used

These are some tools and technologies used in this project.

4.7.1 C-Sharp:

C-Sharp is a programming language developed by Microsoft that runs on the .NET Framework. C-Sharp is used to develop web apps, desktop apps, mobile apps, games and much more.

Figure 4.4: C-Sharp[15]

4.7.2 Blender

Blender is a free, open-source 3D graphics software used for creating animated films, visual effects, art, models, motion graphics, interactive applications, and virtual reality.

Figure 4.5: blender[27]

4.7.3 Git

Git is a distributed version control system that tracks versions of files. It is used to control source code and collaboratively developing software.

4.7.4 GitHub

GitHub is a platform for developers to create, store, manage, and share their code, with Git software providing distributed version control, access control, task management, and continuous integration.

Figure 4.6: git[31]

Figure 4.7: GitHub[19]

4.7.5 Meta

Meta-verse is a sophisticated, interconnected virtual world that enhances user experiences by creating seamless, immersive digital environments.

Figure 4.8: Meta[30]

4.7.6 Unity

Unity was utilized to develop 3D and 2D games, interactive simulations, and has been widely adopted by various industries beyond video gaming.

Figure 4.9: Unity[29]

4.7.7 Meta Quest 2

Project MetaMed VR utilized Meta Quest 2 for its advanced features like hand tracking and gesture recognition, enhancing medical education through immersive virtual reality experiences.

Figure 4.10: Meta Quest 2[20]

4.7.8 Microsoft Visual C++

Microsoft Visual C++ is integrated with Unity 3D to optimize code, improve performance, and manage memory in complex applications, ensuring smooth and efficient gameplay.

Figure 4.11: Microsoft Visual C++[22]

Chapter 5

Iteration Plan

5.1 Planning Strategy

Planning Strategy for our project will be as:

5.1.1 FYP 1 Mid:

- Literature review related Metaverse, and Healthcare
- Designed Use-case, System Diagram

5.1.2 FYP 1 Final:

- Environment setup in Unity3D and Blender
- Designed basic environment of Hospital like Rooms and Cadavers
- Delivered Demo of our environment

5.1.3 FYP 2 Mid:

- User Movements
- Patient Information
- Medicines Information
- Basic Tools like syringes

5.1.4 FYP 2 Final:

- Debugging
- Oculus Integration
- Oculus Testing

Chapter 6

Iteration 1 (FYP-1 Mid)

- FYP 1 Mid:
  - Literature review related Metaverse, and Healthcare
  - Designed Use-case, System Diagram

6.1 Literature review related Metaverse, and Healthcare

This literature review explores the potential of metaversion in healthcare, a virtual, interconnected digital realm. It examines current research, technology, and developments in metaversion, focusing on historical perspectives, technological foundations, practical applications, ethical considerations, and future prospects. The review aims to illuminate the current state of knowledge and pave the way for further advancements in this rapidly evolving field.

6.2 Designed Use-case, System Diagram

- Use Case
- System Diagram
- Activity Diagram

6.3 Use Case Diagram

A use case diagram visualizes main system activities and interactions, identifying main processes in ovals. It is drawn from a scenario to explain system functioning.

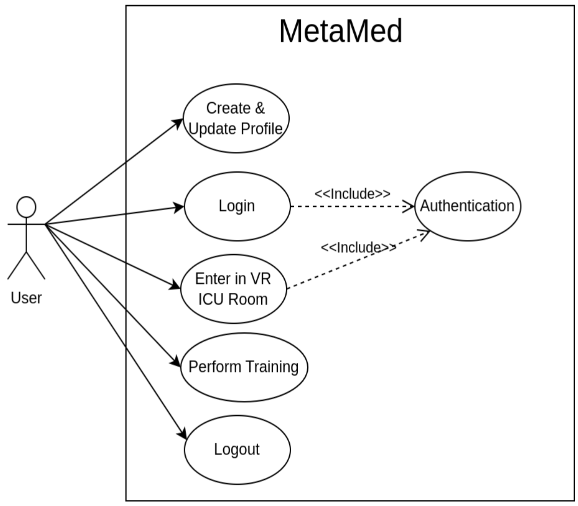

Figure 6.1: Use Case Diagram

6.4 System Diagram

System Diagrams are visual representations of dynamic forces and interactions within a process, serving as more than just a process flow chart.

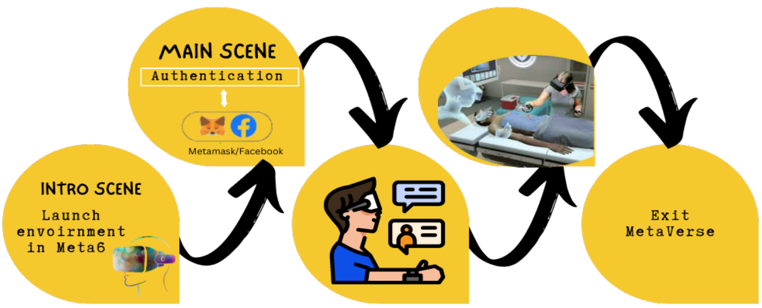

Figure 6.2: System Diagram

6.5 Activity Diagram

Activity diagrams are visual representations of a system's behavior, used to model various systems and processes, including business workflows, software systems, and organizational processes.

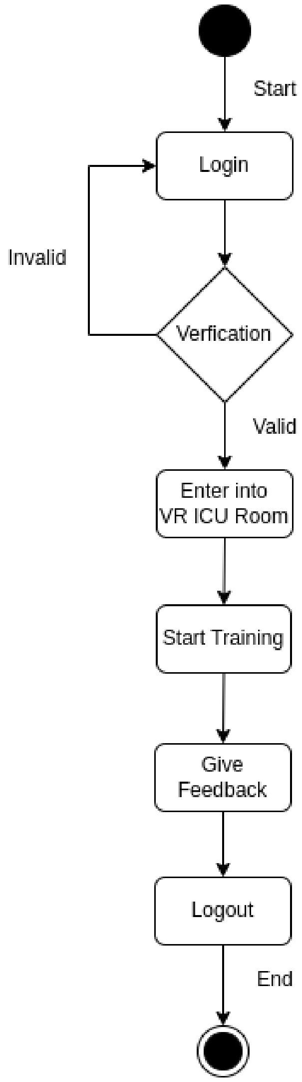

Figure 6.3: Activity Diagram

Chapter 7

Iteration Plan 2 (FYP 1 Final)

This chapter outlines the second iteration plan for our project, providing guidance on module development and addressing students' discussion on the second stage of implementation method.

7.1 FYP 1 Final:

- FYP 1 Final Presentation:
  - Environment setup in Unity3D and Blender
  - Designed basic environment of Hospital

7.2 Designed basic environment of Hospital:

- Reception Area
- Waiting Rooms
- Patient Rooms
- Training Facilities

7.2.1 Hospital Entrance:

The hospital entrance serves as the main entry point to the virtual hospital

Figure 7.1: Hospital Enterance

7.2.2 Reception Area:

The reception area is the primary entrance to the virtual hospital, where patients and visitors can check-in and provide necessary information.

Figure 7.2: Reception Area

7.2.3 Waiting Area:

Waiting area is designed to provide comfortable seating arrangements for patients and their companions while they wait for appointments or procedures.

Figure 7.3: Waiting Area

7.2.4 Patient Rooms:

Patient rooms are designed to offer a comfortable and healing atmosphere for patients during their hospital stay.

Figure 7.4: Patient Room

7.2.5 Patient Body:

Meta-verse Hospital has developed Patient Body to simulate patient interactions, offer realistic medical scenarios for training, and enhance healthcare professionals' virtual learning experiences.

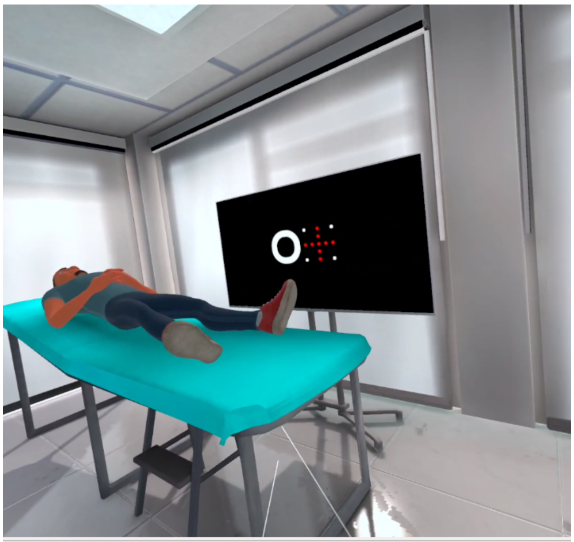

Figure 7.5: Patient Room

Chapter 8

Iteration 3 (FYP-2 Mid)

The first iteration is expected to be completed by the midterm of the FYP-2. This chapter will have some of the artifacts based on system design. The requirements analysis section is same for all the systems while the design may vary. There may have two types of designs the structural design or . First section is for the structural design.

8.1 FYP-2 Mid

Objectives for FYP-2 Mid

- Basic tools
- Medicines
- User movements
- Grabbing Tools

8.1.1 Tools and Medicines

Figure 8.1: Tools and Medicine box

8.1.2 Tools Grabbing

Figure 8.2: Grabbing Tools

8.1.2.1 Hands Washing

Figure 8.3: Hands Washing

8.1.2.2 Pour Alcohol

Figure 8.4: Pour Alcohol

8.1.2.3 Clean the injection site

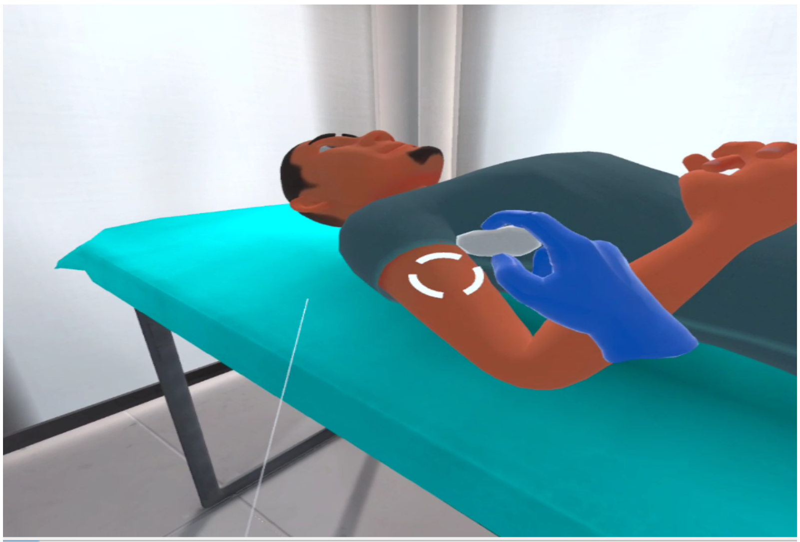

Figure 8.5: Clean the injection site

8.1.2.4 Dispose the Cotton

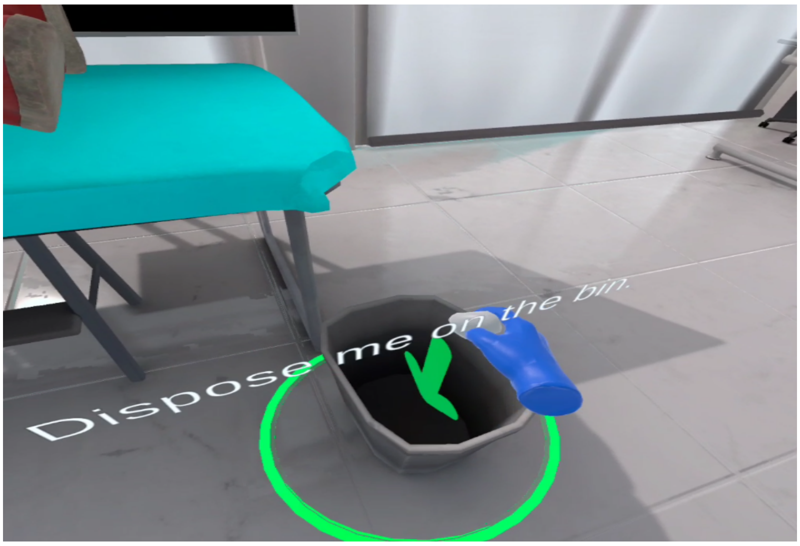

Figure 8.6: Dispose the Cotton

8.1.2.5 Picking the injection

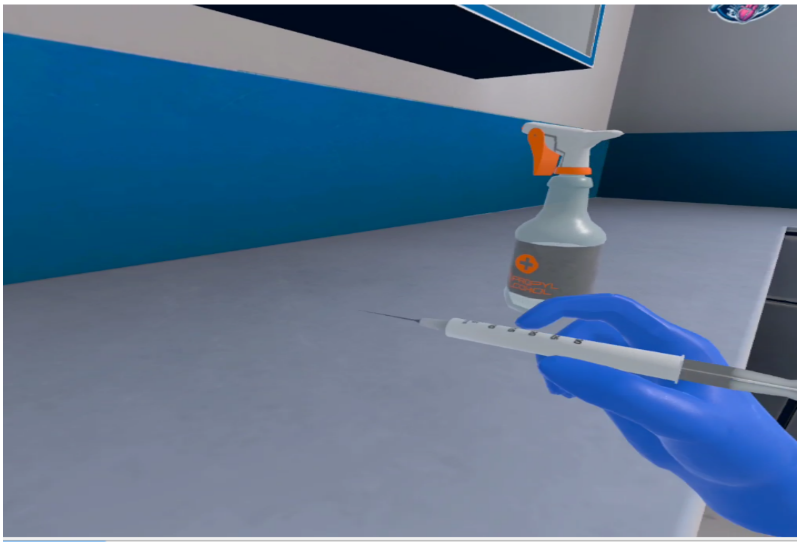

Figure 8.7: Picking the injection

8.1.2.6 Inject the medication

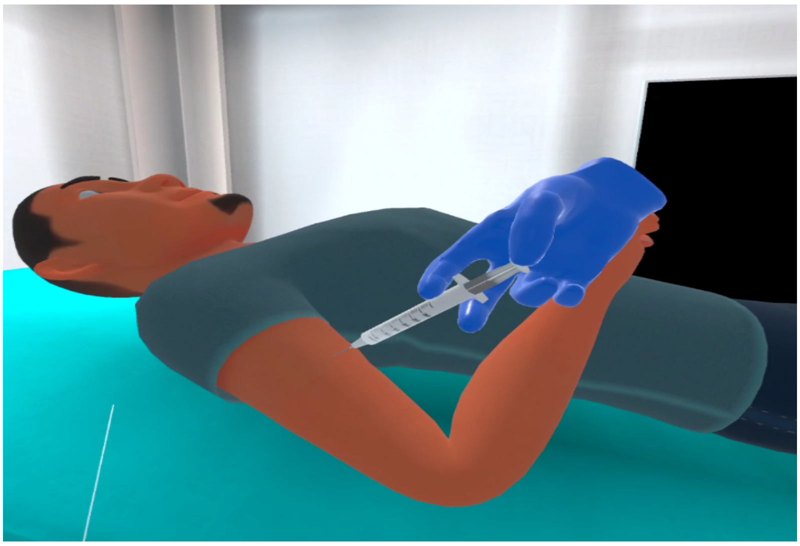

Figure 8.8: Inject the medication

8.1.2.7 Applying Cut on Body

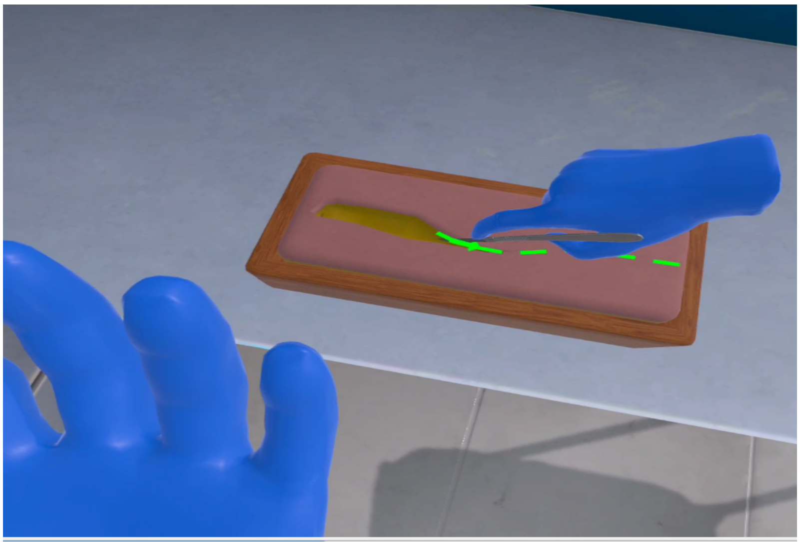

Figure 8.9: Applying Cut on Body

8.1.2.8 Objects Movement

Here are some code for actions performed.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class VRPlayerController : MonoBehaviour
{
    public float speed = 3.0f;
    public float rotationSpeed = 90.0f;
    public OVRHand leftHand;
    public OVRHand rightHand;
    public Transform leftHandAnchor;
    public Transform rightHandAnchor;
    public float grabRadius = 0.1f;
    public LayerMask grabbableLayer;

    private GameObject leftGrabbedObject = null;
    private GameObject rightGrabbedObject = null;

    void Update()
    {
        HandleMovement();
        HandleHandInteractions();
    }

    void HandleMovement()
    {
        // Handle forward and backward movement
        float moveDirection = OVRInput.Get(OVRInput.Axis2D.PrimaryThumbstick).y;
        transform.Translate(Vector3.forward * moveDirection * speed * Time.deltaTime);

        // Handle left and right rotation
        float rotateDirection = OVRInput.Get(OVRInput.Axis2D.PrimaryThumbstick).x;
        transform.Rotate(Vector3.up, rotateDirection * rotationSpeed * Time.deltaTime);
    }
```

Figure 8.10: User Movement Code 1

```
void HandleHandInteractions()
{
    // Handle left hand interactions
    if (leftHand.GetFingerIsPinching(OVRHand.HandFinger.Index))
    {
        Debug.Log("Left hand is pinching with index finger.");
        // Add code to interact with objects
    }

    if (OVRInput.GetDown(OVRInput.Button.PrimaryHandTrigger))
    {
        if (leftGrabbedObject == null)
        {
            TryGrabObject(leftHandAnchor, out leftGrabbedObject);
        }
    }
    else if (OVRInput.GetUp(OVRInput.Button.PrimaryHandTrigger))
    {
        if (leftGrabbedObject != null)
        {
            ReleaseObject(ref leftGrabbedObject);
        }
    }

    // Handle right hand interactions
    if (rightHand.GetFingerIsPinching(OVRHand.HandFinger.Index))
    {
        Debug.Log("Right hand is pinching with index finger.");
        // Add code to interact with objects
    }

    if (OVRInput.GetDown(OVRInput.Button.SecondaryHandTrigger))
    {
        if (rightGrabbedObject == null)
        {
            TryGrabObject(rightHandAnchor, out rightGrabbedObject);
        }
    }
```

Figure 8.11: User Movement Code 2

```
        else if (OVRInput.GetUp(OVRInput.Button.SecondaryHandTrigger))
        {
            if (rightGrabbedObject != null)
            {
                ReleaseObject(ref rightGrabbedObject);
            }
        }
    }

    void TryGrabObject(Transform handAnchor, out GameObject grabbedObject)
    {
        Collider[] colliders = Physics.OverlapSphere(handAnchor.position, grabRadius, grabbableLayer);
        if (colliders.Length > 0)
        {
            grabbedObject = colliders[0].gameObject;
            grabbedObject.transform.SetParent(handAnchor);
            grabbedObject.GetComponent<Rigidbody>().isKinematic = true;
            Debug.Log("Object grabbed: " + grabbedObject.name);
        }
        else
        {
            grabbedObject = null;
        }
    }

    void ReleaseObject(ref GameObject grabbedObject)
    {
        grabbedObject.transform.SetParent(null);
        grabbedObject.GetComponent<Rigidbody>().isKinematic = false;
        Debug.Log("Object released: " + grabbedObject.name);
        grabbedObject = null;
    }
}
```

Figure 8.12: User Movement Code 3

```
void HandleHandInteractions(){
    if (OVRInput.GetDown(OVRInput.Button.PrimaryHandTrigger))    {
        if (leftGrabbedObject == null)    {
            TryGrabObject(leftHandAnchor, out leftGrabbedObject);
        }
        else if (leftGrabbedObject.name == "AlcoholBottle" && !isHandWashing)    {
            StartCoroutine(HandWashingRoutine());
        }
    }
    else if (OVRInput.GetUp(OVRInput.Button.PrimaryHandTrigger))    {
        if (leftGrabbedObject != null && !isHandWashing)    {
            ReleaseObject(ref leftGrabbedObject);
        }
    }
    if (OVRInput.GetDown(OVRInput.Button.SecondaryHandTrigger))    {
        if (rightGrabbedObject == null)    {
            TryGrabObject(rightHandAnchor, out rightGrabbedObject);
        }
        else if (rightGrabbedObject.name == "AlcoholBottle" && !isHandWashing)    {
            StartCoroutine(HandWashingRoutine());
        }
    }
    else if (OVRInput.GetUp(OVRInput.Button.SecondaryHandTrigger))   {
        if (rightGrabbedObject != null && !isHandWashing)    {
            ReleaseObject(ref rightGrabbedObject);
        }
    }
}
void TryGrabObject(Transform handAnchor, out GameObject grabbedObject){
    Collider[] colliders = Physics.OverlapSphere(handAnchor.position, grabRadius, grabbableLayer);
    if (colliders.Length > 0)    {
        grabbedObject = colliders[0].gameObject;
        grabbedObject.transform.SetParent(handAnchor);
        grabbedObject.GetComponent<Rigidbody>().isKinematic = true;
        Debug.Log("Object grabbed: " + grabbedObject.name);
    }
    else    {
        grabbedObject = null;
    }
}
void ReleaseObject(ref GameObject grabbedObject){
    grabbedObject.transform.SetParent(null);
    grabbedObject.GetComponent<Rigidbody>().isKinematic = false;
    Debug.Log("Object released: " + grabbedObject.name);
    grabbedObject = null;
}
IEnumerator HandWashingRoutine(){
    isHandWashing = true;
    Debug.Log("Hand washing started");
    yield return new WaitForSeconds(5.0f);
    Debug.Log("Hand washing completed");
    isHandWashing = false;
    if (leftGrabbedObject != null && leftGrabbedObject.name == "AlcoholBottle")    {
        ReleaseObject(ref leftGrabbedObject);
    }
    if (rightGrabbedObject != null && rightGrabbedObject.name == "AlcoholBottle"){
        ReleaseObject(ref rightGrabbedObject);
    }
}
```

Figure 8.13: Pore Alcohol for Hands Washing

```
void HandleHandInteractions(){
    if (OVRInput.GetDown(OVRInput.Button.PrimaryHandTrigger))    {
        if (leftGrabbedObject == null)    {
            TryGrabObject(leftHandAnchor, out leftGrabbedObject);
        }
        else if (leftGrabbedObject.name == "CottonBud")    {
            TryCleanInjectionSite(leftHandAnchor);
        }
    }
    else if (OVRInput.GetUp(OVRInput.Button.PrimaryHandTrigger))    {
        if (leftGrabbedObject != null)    {
            ReleaseObject(ref leftGrabbedObject);
        }
    }
    if (OVRInput.GetDown(OVRInput.Button.SecondaryHandTrigger))    {
        if (rightGrabbedObject == null)    {
            TryGrabObject(rightHandAnchor, out rightGrabbedObject);
        }
        else if (rightGrabbedObject.name == "CottonBud")    {
            TryCleanInjectionSite(rightHandAnchor);
        }
    }
    else if (OVRInput.GetUp(OVRInput.Button.SecondaryHandTrigger))    {
        if (rightGrabbedObject != null)    {
            ReleaseObject(ref rightGrabbedObject);
        }
    }
}
void TryGrabObject(Transform handAnchor, out GameObject grabbedObject){
    Collider[] colliders = Physics.OverlapSphere(handAnchor.position, grabRadius, grabbableLayer);
    if (colliders.Length > 0)    {
        grabbedObject = colliders[0].gameObject;
        grabbedObject.transform.SetParent(handAnchor);
        grabbedObject.GetComponent<Rigidbody>().isKinematic = true;
        Debug.Log("Object grabbed: " + grabbedObject.name);
    }
    else    {
        grabbedObject = null;
    }
}
void ReleaseObject(ref GameObject grabbedObject){
    grabbedObject.transform.SetParent(null);
    grabbedObject.GetComponent<Rigidbody>().isKinematic = false;
    Debug.Log("Object released: " + grabbedObject.name);
    grabbedObject = null;
}
void TryCleanInjectionSite(Transform handAnchor){
    Collider[] colliders = Physics.OverlapSphere(handAnchor.position, grabRadius, bodyPartLayer);
    if (colliders.Length > 0)    {
        Debug.Log("Cleaning injection site on: " + colliders[0].gameObject.name);
        StartCoroutine(CleaningRoutine(colliders[0].gameObject));
    }
}
IEnumerator CleaningRoutine(GameObject bodyPart){
    Debug.Log("Cleaning started on: " + bodyPart.name);
    yield return new WaitForSeconds(3.0f);
    Debug.Log("Cleaning completed on: " + bodyPart.name);
}
```

Figure 8.14: Cleaning Injection Site Part

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class VRPlayerController : MonoBehaviour{
    public float grabRadius = 0.1f;
    public LayerMask grabbableLayer, bodyPartLayer;
    private GameObject grabbedObject = null;
    void Update(){
        HandleHandInteractions();
    }
    void HandleHandInteractions(){
        if (OVRInput.GetDown(OVRInput.Button.PrimaryHandTrigger)){
            if (grabbedObject == null)    {
                TryGrabObject(out grabbedObject);
            }
            else if (grabbedObject.name == "Syringe")    {
                TryInject();
            }
        }
        else if (OVRInput.GetUp(OVRInput.Button.PrimaryHandTrigger))    {
            if (grabbedObject != null)    {
                ReleaseObject(ref grabbedObject);    }
        }
    }
    void TryGrabObject(out GameObject grabbedObject)    {
        Collider[] colliders = Physics.OverlapSphere(transform.position, grabRadius, grabbableLayer);
        if (colliders.Length > 0)    {
            grabbedObject = colliders[0].gameObject;
            grabbedObject.transform.SetParent(transform);
            grabbedObject.GetComponent<Rigidbody>().isKinematic = true;
            Debug.Log("Object grabbed: " + grabbedObject.name);
        }
        else    {
            grabbedObject = null;       }
    }
    void ReleaseObject(ref GameObject grabbedObject)    {
        grabbedObject.transform.SetParent(null);
        grabbedObject.GetComponent<Rigidbody>().isKinematic = false;
        Debug.Log("Object released: " + grabbedObject.name);
        grabbedObject = null;
    }
    void TryInject()    {
        Collider[] colliders = Physics.OverlapSphere(transform.position, grabRadius, bodyPartLayer);
        if (colliders.Length > 0)    {
            Debug.Log("Injecting into: " + colliders[0].gameObject.name);
            StartCoroutine(InjectionRoutine(colliders[0].gameObject));        }
    }
    IEnumerator InjectionRoutine(GameObject bodyPart)    {
        Debug.Log("Injection started on: " + bodyPart.name);
        yield return new WaitForSeconds(3.0f);
        Debug.Log("Injection completed on: " + bodyPart.name);    }
}
```

Figure 8.15: Inject Syringe in Body

```
using System.Collections;
using UnityEngine;
public class VRPlayerController : MonoBehaviour{
    public Transform leftHandAnchor, rightHandAnchor;
    public float grabRadius = 0.1f;
    public LayerMask bodyPartLayer;
    private GameObject lastCutBodyPart = null;
    void Update()    {
        HandleHandInteractions();    }
    void HandleHandInteractions()    {
        if (OVRInput.GetDown(OVRInput.Button.PrimaryHandTrigger))    {
            if (TryApplyCut(leftHandAnchor))    {
                Debug.Log("Cut applied using left hand.");    }
            else if (TryApplyBandage(leftHandAnchor))    {
                Debug.Log("Bandage applied using left hand.");    }    }
        if (OVRInput.GetDown(OVRInput.Button.SecondaryHandTrigger))    {
            if (TryApplyCut(rightHandAnchor))    {
                Debug.Log("Cut applied using right hand.");    }
            else if (TryApplyBandage(rightHandAnchor))    {
                Debug.Log("Bandage applied using right hand.");    }
        }
    }
    bool TryApplyCut(Transform handAnchor)    {
        Collider[] colliders = Physics.OverlapSphere(handAnchor.position, grabRadius, bodyPartLayer);
        if (colliders.Length > 0)    {
            Debug.Log("Applying cut on: " + colliders[0].gameObject.name);
            lastCutBodyPart = colliders[0].gameObject;
            StartCoroutine(CutRoutine(colliders[0].gameObject));
            return true;
        }
        return false;
    }
    IEnumerator CutRoutine(GameObject bodyPart)    {
        Debug.Log("Cut started on: " + bodyPart.name);
        yield return new WaitForSeconds(3.0f);
        Debug.Log("Cut completed on: " + bodyPart.name);
    }
    bool TryApplyBandage(Transform handAnchor)    {
        if (lastCutBodyPart != null)    {
            Debug.Log("Applying bandage on: " + lastCutBodyPart.name);
            StartCoroutine(BandageRoutine(lastCutBodyPart));
            return true;
        }
        return false;
    }
    IEnumerator BandageRoutine(GameObject bodyPart)    {
        Debug.Log("Bandage started on: " + bodyPart.name);
        yield return new WaitForSeconds(3.0f);
        Debug.Log("Bandage completed on: " + bodyPart.name);
        lastCutBodyPart = null;
    }
}
```

Figure 8.16: Applying Cut and Bandage on Body

8.2 SVN or GitHub

Here is the GitHub link:

https://github.com/Daudsarfraz/MetaMed

Chapter 9

Iteration 4

Meta-Med VR project's fourth iteration enhances user experience by integrating Keyboard and Meta Quest 2 functionalities, improving immersion, realism, and educational value for healthcare professionals.

9.1 Introduction

Meta-Med VR is a revolutionary approach that aims to enhance medical education by providing immersive virtual reality experiences for training and skill development.

Integration Focus:

This document focuses on the integration of Keyboard and Meta Quest 2 to enhance object interaction and movement within the Meta Med VR environment, aiming to improve user engagement and learning outcomes.

9.2 Integration Process with Keyboard

We evaluated several input devices based on factors such as compatibility, affordability, and ease of use. After careful consideration, we chose Keyboard for its versatility and familiarity to users.

Compatibility Assessment:

The MetaMed VR software underwent rigorous compatibility tests to ensure seamless integration, verifying hardware specifications and assessing software support for keyboard inputs.

Software Configuration:

Custom software configurations were created to enable recognition and response to keyboard inputs in the MetaMed VR environment, mapping keyboard keys to specific actions or functions.

Testing and Calibration:

The integration underwent rigorous testing to ensure smooth object interaction and movement, with calibration procedures implemented to optimize performance and reduce input latency.

Enhancements and Features:

Users can manipulate virtual objects using keyboard inputs, performing actions like grabbing, rotating, and moving objects within the VR space.

Movement Controls:

The keyboard integration offers users intuitive movement controls, including locomotion and teleportation options, facilitating seamless navigation through virtual environments.

User Experience Improvements:

The integration improves immersion and realism in the MetaMed VR environment, enhancing engagement and effectiveness in training scenarios.

9.3 Integration Process with Meta Quest 2

Selection of Meta Quest 2:

We selected Meta Quest 2 due to its advanced features and user-friendly interface after evaluating various input devices for compatibility, affordability, and tracking capabilities.

Compatibility Assessment:

The integration of MetaMed VR software was ensured through rigorous compatibility tests, verifying hardware specifications and assessing software support for Meta Quest 2 inputs.

Software Configuration:

Custom software configurations were created to enable recognition and response to Meta Quest 2 inputs within the MetaMed VR environment, including integrating Meta Quest 2.

9.4 Testing and Calibration

Testing Procedures:

The integration underwent rigorous testing to ensure smooth object interaction and movement, with calibration procedures implemented to optimize tracking accuracy and minimize latency.

9.5 Enhancements and Features

Meta Quest 2 enables users to interact with virtual objects through gestures like grabbing, rotating, and moving within the VR space. Meta Quest 2's integration offers users intuitive movement controls, including locomotion and teleportation options, facilitating seamless navigation through virtual environments.

The integration improves immersion and realism in the MetaMed VR environment, enhancing engagement and effectiveness in training scenarios.

9.6 Challenges and Solutions

The integration process faced technical issues like input latency and mapping conflicts.

The challenges were resolved through software updates, configuration adjustments, and calibration procedures to guarantee a seamless user experience.

9.7 Future Directions

Future improvements include improving input mappings, enhancing keyboard shortcut support, improving gesture recognition algorithms, and introducing advanced hand tracking features.

9.8 Conclusion

The integration of Keyboard and Meta Quest 2 with *MetaMed* VR is a significant advancement in medical education through immersive virtual reality experiences. *MetaMed* VR enhances healthcare professional training and patient care outcomes by improving object interaction and movement controls.

Chapter 10

Implementation Details

Here are some implementation details of project. What we can do in project.

10.1 Introduction

This essay explores the use of Meta Quest 2 and the various Libraries to create a comprehensive virtual training environment for medical practitioners, detailing the process from user authentication to performing procedures in a virtual hospital setting, showcasing Virtual Reality's potential in medical education.

10.2 User Authentication and Environment Access

The training uses Meta Quest 2, a high-resolution virtual environment, with keyboard navigation. Authentication involves signing up through the meta-verse or Meta-Mask, ensuring security. Correct credentials grant access, while incorrect ones prompt a new sign-up, ensuring integrity and integrity.

10.3 Navigation to the Virtual Hospital Environment

Upon successful authentication, users are transported to a meticulously crafted virtual hospital environment, allowing navigation through various areas such as wards, operation theaters, laboratories, and administrative sections, enhancing the training experience.

10.4 Scenario Selection and Patient Interaction

Users visit the reception area to select an interactive training scenario from various patient cases, covering various medical conditions. They then practice diagnostic and treatment skills in a controlled, risk-free environment by observing detailed patient histories, symptoms, and physical examinations in the designated patient.

10.5 Laboratory Interaction and Tool Acquisition

Users choose a patient scenario and visit a virtual laboratory to gather tools and medications. The VR environment simulates a fully-equipped lab, allowing users to practice selecting and handling medical instruments. Some open-source frameworks enhance realism and training effectiveness.

10.6 Performing Medical Procedures

Users use VR to perform medical procedures, providing step-by-step guidance and real-time feedback. The immersive nature allows users to experience procedural nuances, such as injections, wound suturing, and diagnostic equipment usage.

10.7 Exit from the Virtual Environment

The user exits the virtual environment after completing the training scenario, allowing easy transition back to the real world, and the training sessions can be repeated for skill development and refinement.

Chapter 11

User Manual

This chapter will have the user manual. These UI/UX elements are crafted to be responsive and visually appealing, ensuring a seamless and enjoyable experience for users as they practice medical procedures and scenarios.

11.1 Choose Scenario

The user interface (UI) enhances the user experience (UX) with intuitive design elements. Users start by selecting a scenario corresponding to a patient's medical issue from a menu of various conditions or scenarios.

Figure 11.1: User Manual

11.2 1.1 Start Session

Once a scenario is chosen, the UI features a 1.1 toggle menu for keyboard interaction, enhancing user

Figure 11.2: Start Session

11.3 1.2 Choose Scenario

Here can choose scenario like operation of leg, arm, foot etc.

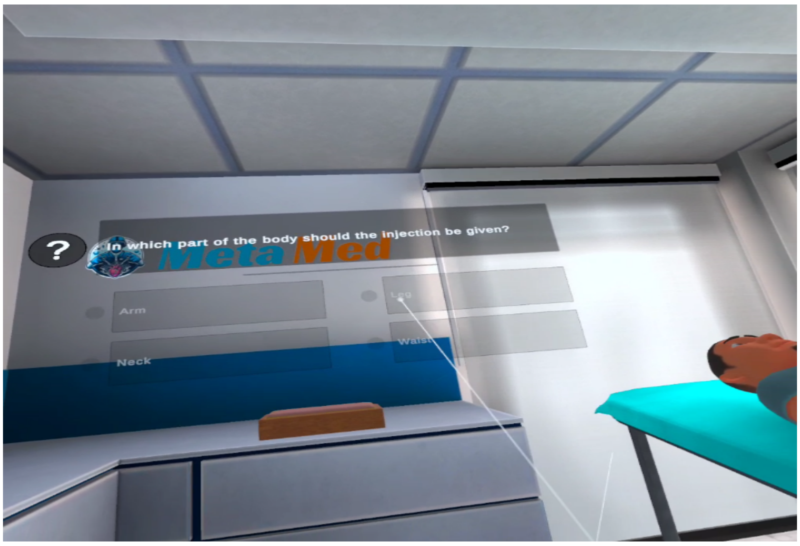

Figure 11.3: Choose Scenario

Chapter 12

Conclusions and Future Work

In this chapter we have defined which features can be implement in Future.

12.1 Conclusion:

Our *metaverse* project's basic capabilities, such as injection procedures, patient records, and medication records, have been effectively built throughout its first development. The foundation for a strong, engaging, and extremely useful virtual healthcare environment is laid by these fundamental instruments. But in order for this *metaverse* healthcare system to reach its full potential, a few things need to be improved and developed further.

12.2 Future Work:

Future research will focus on advanced clinical simulations, real-time feedback, and virtual diagnostic tools like blood tests, CT, and MRI scans. Emergency training scenarios will incorporate time-sensitive decision-making and multi-user cooperation. AI-driven virtual consultations, comprehensive patient histories, and secure communication will enhance patient interaction. Therapeutic environments will offer interactive modules for education, rehabilitation, and mental health therapy. Key priorities include interoperability with healthcare systems, IoT device integration, and blockchain for secure records. Educational resources will expand through interactive modules, virtual dissection labs, and online CME in the metaverse. User experience will be improved with a user-friendly interface, multilingual support, cultural adaptations, and accessibility features.

Bibliography

[1] Interacting with virtual reality, 1995.

[2] Virtual reality surgical training and assessment system, 2001.

[3] The history of simulation in medical education and possible future directions., 2006.

[4] A virtual operating room for context-relevant training, 2007.

[5] The history of medical simulation, 2008.

[6] *Transforming medical education and training with VR using M.A.G.E.S.* ACM, 2018.

[7] Development of metaverse for intelligent healthcare, 2022.

[8] Digital transformation will change medical education and rehabilitation in spine surgery, 2022.

[9] Utilizing the metaverse in anatomy and physiology, 2022.

[10] Current perspective of metaverse application in medical education, research and patient care, 2023.

[11] Medical education system based on metaverse platform: A game theoretic approach, 2023.

[12] Next-gen mulsemedia: Virtual reality haptic simulator's impact on medical practitioner for higher education institutions, 2023.

[13] *Role of virtual reality for healthcare education*. Elsevier eBooks, 2023.

[14] Virtual reality in medical institutions: Innovative tool to strengthen the process of delivery of medical education, 2023.

[15] Mads Torgersen Anders Hejlsberg. c sharp logo, 2000. https://tinyurl.com/2h297drd.

[16] Gaurang Bansal, Karthik Rajgopal, Vinay Chamola, Zehui Xiong, and Dusit Niyato. Healthcare in metaverse: A survey on current metaverse applications in healthcare. *IEEE Access*, 10:119914–119946, 2022.

[17] Kala Baskar. Revolutionizing medical education with metaverse. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 07 2022.

[18] Hrvoje Brajčić and Mirna Šitum. Virtual reality in medicine. 543:37–42, 2020.

[19] Tom Preston-Werner Chris Wanstrath, P. J. Hyett and Scott Chacon. Github logo, 8 February 2008. https://tinyurl.com/4v6fzea5.

[20] Reality Labs. Meta quest logo, October 13, 2020. https://tinyurl.com/2seud6w3.

[21] Lik-Hang Lee, Tristan Braud, Pengyuan Zhou, Lin Wang, Dianlei Xu, Zijun Lin, Abhishek Kumar, Carlos Bermejo, and Pan Hui. All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda. *arXiv preprint arXiv:2110.05352*, 2021.

[22] Microsoft. Visual studio icon, February 1993. https://tinyurl.com/mryj5drf.

[23] Juan Antonio Juanes Méndez. The metaverse in medical education and clinical practice. In *Handbook of Healthcare in the Metaverse*, pages 157–164. Springer, 2023.

[24] George Papagiannakis, Nick Lydatakis, Steve Kateros, Stelios Georgiou, and Paul Zikas. A virtual environment for training and assessment of surgical teams. In *Proceedings of the ACM Symposium on Virtual Reality*, pages 17–26, 2018.

[25] Jack Pottle. Virtual reality and the transformation of medical education. *Future Healthcare Journal*, 6(3):181–185, 2019.

[26] Precedence Research. Report code: 1781, 2024. Category: ICT, No. of Pages: 150+, Format: PDF/PPT/Excel, Published: February 2024, Historical Year: 2021-2022, Base Year: 2023, Estimated Years: 2024-2033.

[27] Ton Roosendaal. blender logo, January 2, 1994. https://tinyurl.com/yjvmu2k7.

[28] Liangjing Shao, Wei Tang, Ziqun Zhang, and Xinrong Chen. Medical metaverse: Technologies, applications, challenges and future. *Journal of Mechanics in Medicine and Biology*, 23(02):2350028, 2023.

[29] Unity Technologies. unity logo, June 8, 2005. https://tinyurl.com/2uamt6p8.

[30] Inc. (2004–2005) TheFacebook. Meta logo, January 4, 2004. https://tinyurl.com/5fkkp63x.

[31] Linus Torvalds. git logo, 7 April 2005. https://tinyurl.com/ybx33vk2.

[32] Diana Velazquez-Pimentel, Thomas Hurkxkens, and Jean Nehme. *A Virtual Reality for the Digital Surgeon*, pages 183–201. 2021.

[33] GOKUL YENDURI, RUTVIJ H JHAVERI, MAMOUN ALAZAB, REDDY MADDIKUNTA, and THIPPA REDDY GADEKALLU. Metaverse for healthcare: A survey on potential applications, challenges and future directions. *Journal of Medical Systems*, 2023.