

Dokumentation & Projekttagbuch

Innovation Lab 1
Jahr 2024-2025

Projekt: Anatomie- Lernwerkzeug Herz

Team: **16_HerzSimul**
Marcel Gössl
Felix Hadinger
Simon Marek
Karim Salem

1. Allgemeine Informationen

Projektname: Anatomie- Lernwerkzeug Herz

Supervisor: Lilly Tremel

Innovation Lab 1, Wintersemester 2024/25

Projektteam:

Marcel Gössl	if23b093@technikum-wien.at	Product Owner, Developer
Felix Hadinger	if23b079@technikum-wien.at	Scrum Master, Developer
Simon Marek	if23b111@technikum-wien.at	Developer
Karim Salem	if22b240@technikum-wien.at	Developer
Katarina Gmeiner	if23b092@technikum-wien.at	Studium abgebrochen

Management-Summary des Projektes

Ziel dieses Projektes ist die Entwicklung eines interaktiven Virtual Reality Lernwerkzeuges, das in anatomischen Lehrveranstaltungen eingesetzt wird. Dabei können medizinische Daten eines Herzes importiert und dreidimensional dargestellt werden (DICOM). Mithilfe von Handtracking soll sich das 3D Herzmodell intuitiv drehen, zoomen und interaktiv erforschen lassen. Darüber hinaus soll eine 2D Darstellung in 3 verschiedenen anatomischen Schnittebenen möglich sein, um verschiedene Ansichten des Herzens zu ermöglichen.

Rahmenbedingungen und Projektumfeld

Hardware Kompatibilität: Das Projekt erfordert VR- Headsets, um Handtracking für Zoom und Rotation des Herzmodells zu ermöglichen.

DICOM-Daten-Integration: Die Fähigkeit, medizinische Bilddaten (DICOM) einzulesen und in 3D-Modelle zu konvertieren, ist essenziell. Dies erfordert geeignete medizinische Bildverarbeitungsbibliotheken.

Benutzerfreundlichkeit und intuitive Steuerung: Die VR-Umgebung muss eine benutzerfreundliche Interaktion gewährleisten, insbesondere durch Handtracking für Zoom- und Rotationsfunktionen.

Unity-Engine: Als Entwicklungsplattform wird Unity genutzt, um das VR-Lernwerkzeug zu entwickeln. Unity bietet VR-Integration und 3D-Modellierung, was für dieses Projekt entscheidend ist.

Programmiersprachen und -tools: C# für die Entwicklung in Unity und möglicherweise Python oder C++ für die Vorverarbeitung von DICOM-Daten und die Modellierung der 3D-Strukturen.

Semester-Roadmap

Roadmap	InnoLab 1			
	Oktober	November	Dezember	Jänner 2025
Projekt Packages				
3D-Modell aus DICOM-Daten (~245h)				
Handtracking für Zoom und Rotation (~120h)				
Farbliche Darstellungen (~130h)				
2D-Darstellung in drei anatomischen Schnittebenen (~130h)				
Möglichkeit zur Einsicht ins Herz (~130h)				
Labeling (~125h)				
Interaktives Quiz (~250h)				
Aufwand: 3 ECTS = 75h/Person				
75 * 5 = 375h pro Sem				
375 * 3 = 1125h fürs Projekt				

[illegible]

Collaboration & Tooling

GitHub: <https://github.com/Foxatdoom/HerzSimul.git>

(aufgrund der Projektgröße von knapp 6 GB derzeit nur für Organisationsdokumente genutzt)

Trello:

<https://trello.com/invite/b/678d30dc502e325302570052/ATTIff90bafcab1d93725c410085bbbe1007D72E1D68/board-innolab>

(Projektmanagement)

OneDrive: https://1drv.ms/f/s!AsLH_B8P6VjunPcL0HyjaexvXzNoxg?e=6aagC6

(Unity Projekt, source code)

2. Projekt- Kurzbeschreibung

Projektziele- und Nutzen:

Das Projekt zielt darauf ab, ein interaktives Virtual Reality Lernwerkzeug für anatomische Lehrveranstaltungen zu entwickeln, das die Struktur und die Funktion des menschlichen Herzens anschaulich darstellt. Ziel ist es, eine immersive und lehrreiche VR- Umgebung zu kreieren, damit die Benutzer des Tools die Herzstruktur auf effektive Weise studieren können. Dadurch soll das Lernerlebnis der Benutzer durch einprägende Darstellungen verbessert werden. Um die Nutzer auf ihrem Lernpfad möglichst vielseitig zu unterstützen, werden folgende Features geplant:

Projektumfang:

1. Rendern von DICOM Bilder und Darstellung des Herzens

Erklärung:

Als Grundlage für die Darstellung des menschlichen Herzens dient der Import von sogenannten DICOM (Digital Imaging and Communications in Medicine) Bildern. Diese enthalten neben den eigentlichen Bilddaten (z.B. MRI- Scans) auch Volumes- Informationen und ermöglichen daher eine dreidimensionale, immersive Darstellung des Herzens, einschließlich Ventrikeln, Klappen, Blutgefäßen und Gewebetypen wie Fett- und Muskelgewebe. Dadurch soll ein detailliertes Studium der Herzstruktur erlaubt werden.

Anforderungen/ Ergebnisse:

- Das Herzmodell wird aus medizinischen DICOM- Daten erstellt und in einer 3D- Umgebung dargestellt.
- Die wesentlichen äußeren und strukturellen Merkmale des Herzens und die verschiedenen Gewebetypen werden präzise dargestellt.
- Das Modell ermöglicht eine interaktive Erkundung in einer virtuellen Umgebung.

2. 2D-Darstellung in drei anatomischen Schnittebenen

Erklärung:

Um unterschiedliche Ansichten des Herzens zu ermöglichen, werden die importierten MRI- Bilder zweidimensional in drei anatomischen Schnittebenen (transversal, frontal und sagittal) dargestellt.

Anforderungen/ Ergebnisse:

Folgende Schnittebenen müssen als 2D- Bilder dargestellt werden:

- Die Darstellung ist in der transversalen (axialen) Schnittebene verfügbar: in horizontalem Schnitt, der das Herz von oben nach unten zeigt. Diese Ebene liefert Querschnitte des Herzens, als würde es horizontal durchgeschnitten.
- Die Darstellung ist in frontaler (koronaler) Schnittebene verfügbar: Ein vertikaler Schnitt von vorne nach hinten, der das Herz aus der Vorderansicht zeigt. Diese Ebene teilt das Herz von der Vorder- zur Rückseite.
- Die Darstellung ist in sagittaler (lateral) Schnittebene verfügbar: in vertikalem Schnitt von der Seite, der das Herz von links nach rechts darstellt. Diese Ebene teilt das Herz in eine linke und rechte Hälfte.

3. Zoom und Rotation des Herzens

Erklärung:

Um ein optimales Lernerlebnis zu ermöglichen, soll sich das dreidimensionale Herz interaktiv erforschen lassen. Dazu wird Handtracking verwendet, um es zu drehen und um den Zoom zu verändern.

Anforderungen/ Ergebnisse:

- Das 3D- Modell ist mithilfe von Handtracking steuerbar, um es zu drehen und zu zoomen.

- Das Modell ist dadurch aus verschiedenen Blickwinkeln betrachtbar.
- Das Modell kann dadurch mit einer höheren Detaillierung der Strukturen betrachtet werden.

4. Farbliche Unterscheidung der anatomischen Strukturen

Erklärung:

Für die bessere Visualisierung des Modells werden die anatomischen Strukturen des Herzens mit unterschiedlichen Farben hervorgehoben. Dadurch sollen die verschiedenen Gewebe und Strukturen wie Fett- und Muskelgewebe von Benutzern identifiziert werden können. Dadurch soll das Erkennen und Lernen der verschiedenen Bereiche erleichtert werden, ohne den Fokus auf innere Details wie Vorhöfe und Kammern zu legen.

Anforderungen/ Ergebnisse:

- Äußere anatomische Strukturen und Gewebetypen (wie Muskelgewebe, Fett und Blutgefäße) müssen durch die Wahl passender Farben voneinander abgrenzbar sein.

5. Einsicht vom Innenleben des Herzens

Erklärung:

Benutzer sollen die Möglichkeit erhalten, das Innenleben des Herzens betrachten zu können. Dadurch sollen Vorhöfe, Hauptkammern, Herzklappen und die Zugänge zu den Kammern detailreich visualisiert werden können. Die Lernenden erhalten dadurch Einblicke in die verschiedenen Ebenen des Herzens.

Anforderungen/ Ergebnisse:

- Funktion, um das Innenleben darzustellen: entweder mithilfe von Transparenzstufen oder mit virtuellen Quer- und Längsschnitten.

6. Labels für die dargestellten Komponenten

Erklärung:

Um die lernenden während der Visualisierung mit Informationen zu unterstützen, sollen die dargestellten Komponenten (wie Gewebetypen, Fett, Blutgefäße, Hauptkammern usw.) durch Labels gekennzeichnet werden.

Anforderungen/ Ergebnisse:

- Alle dargestellten Komponenten werden in der VR- Umgebung beschriftet.

7. Anatomie Quiz

Erklärung:

Damit die Benutzer ihren Wissensstand interaktiv überprüfen können, soll ein Quiz zur Verfügung stehen, bei dem die verschiedenen Komponenten des Herzens benannt bzw. ausgewählt und zusätzliche Fragen darüber beantwortet werden müssen.

Anforderungen/ Ergebnisse:

- Es werden Fragen zu den verschiedenen Komponenten gestellt
- Bei richtigen Antworten (z.B. Auswahl der richtigen Komponente) werden zusätzliche Informationen eingeblendet
- Bei einer falsch ausgewählten Komponente wird die korrekte Komponente hervorgehoben

Optionale Features, die implementiert werden können (falls nach der Implementierung der darüber gelisteten Funktionen Zeit übrig bleibt):

Hilfreiche Animationen

Erklärung:

Benutzer sollen in der Lage sein, Animationen wie zum Beispiel Blutfluss oder Muskelbewegungen animiert darstellen zu können.

Ideen: Blutfluss, Muskelbewegung, Abgabe von CO₂ darstellen, Öffnen und Schließen von Herzklappen

Anforderungen/ Ergebnisse:

- Animation der jeweiligen Idee

Die High Level Goals 1. Rendern von DICOM Bilder und Darstellung des Herzens und 3. Zoom und Rotation des Herzens sind für InnovationLab 1 geplant.

Herausforderungen:

Einer der Größten Herausforderungen dieses Projektes stellt die fachliche Komplexität dar. Denn für die Planung und Durchführung ist neben umfangreichem Wissen in den Bereichen Unity und 3D- Modellierung auch grundlegendes Wissen über die menschliche Anatomie in Bezug auf das Herz notwendig. Daher ist die Auseinandersetzung aller Projektmitglieder mit diesen Themen für die erfolgreiche Durchführung essenziell. Vor allem Kommunikation und Wissensaustausch stehen daher im Vordergrund unserer Zusammenarbeit.

Implementierungsstrategie:

Um die beschriebenen Anforderungen zu erfüllen, soll vor der jeweiligen Implementierung zunächst eine Recherche über bereits entwickelte, vergleichbare Features durchgeführt werden. Darunter fallen vor allem der Datenimport, Handtracking und das Labeln der Komponenten. Im Vorfeld wird davon ausgegangen, dass dafür hilfreiche Bibliotheken verwendet bzw. angepasst werden können. Dadurch soll verhindert werden, dass Lösungen für bereits existierende Funktionen entwickelt werden. Der Fokus kann daher auf die Entwicklung von neuen Features gerichtet werden.

3. Spezifikation der Lösung

Systemumgebung

Die Lösung basiert auf Unity (Version 6 oder höher) und wurde mit einer Oculus Quest 2 64GB getestet. Es ist möglich, dass das Projekt auch für andere Meta VR Brillen funktioniert, das wurde allerdings nicht getestet. Das Importieren von DICOM Daten ist nur im Windows build möglich, weshalb ein Windows PC benötigt wird.

Die folgenden Requirements beziehen sich auf die High Level Goals „Rendern von DICOM Bilder“ und „Zoom und Rotation des Herzens“ wie im Kapitel „2. Projekt-Kurzbeschreibung“ beschrieben:

Übersicht Product Backlog

Titel	Beschreibung (User Story)	Akzeptanz-kriterium
Research existing DICOM import Libraries	Als Entwickler möchte ich eine Liste existierender DICOM-Import-Bibliotheken recherchieren, um eine passende Lösung für das Projekt zu identifizieren.	Mind. 3 Bibliotheken bewertet, Machbarkeit abgeschätzt
Test DICOM Libraries	Als Entwickler möchte ich die Bibliotheken testen, um eine Auswahl treffen zu können.	Mind. 3 Bibliotheken in Unity integriert, mit DICOM Datensatz in Unity getestet, Ergebnisse dokumentiert
Get To Know Unity and VR	Als Entwickler möchte ich grundlegende Unity-Konzepte und VR-spezifische Funktionen erlernen, um effektiv mit VR arbeiten zu können.	Installation von Unity durchgeführt und mit grundlegenden Konzepten bekannt gemacht
VR Umgebung aufsetzen + DICOM Library Verknüpfung	Als Entwickler möchte ich eine VR-Umgebung aufsetzen und sie mit der DICOM Library verknüpfen, um die Funktionalität in VR zu testen.	VR-Umgebung eingerichtet, Library integriert und getestet.
Organisation VR Brillen	Als Projektteam möchten wir VR-Brillen organisieren, um die Funktionalität in VR zu testen.	Mindestens eine VR-Brille organisiert, optimalerweise min. 2 Brillen
VR Umgebung aufsetzen: Hello World	Als Entwickler möchte ich eine grundlegende VR-Umgebung mit einer "Hello World"-Szene aufsetzen, um den Einstieg zu schaffen.	VR-Umgebung eingerichtet und einfache Szene getestet (z. B. Text "Hello World" sichtbar).
Basics Handtracking: Research	Als Entwickler möchte ich recherchieren, wie Handtracking-Funktionen in VR integriert werden, um zukünftige Interaktionen zu planen.	Mit VR Handtracking vertraut gemacht und dokumentiert
Basics Handtracking: Draft Implementierung	Als Entwickler möchte ich eine erste Implementierung des Handtracking erstellen, um Interaktionsmöglichkeiten zu testen.	Testprojekt in der VR-Umgebung erstellt, das grundlegende Handbewegungen erkennt und umsetzt.
DICOM Library mit VR Handtracking	Als Entwickler möchte ich die DICOM Library mit einem VR-Handtracking-Projekt	DICOM-Objekt in der VR-Umgebung eingebunden.

Projekt verknüpfen	verknüpfen, um medizinische Objekte in VR verfügbar zu machen.	
Handtracking Zoom + Rotation Basics Recherche	Als Entwickler möchte ich recherchieren, wie Zoom und Rotation mithilfe von Handtracking realisiert werden, um später medizinische Objekte manipulieren zu können	Informationen zu Zoom und Rotation recherchiert.
Zoom und Rotation mit DICOM Objekt implementieren	Als Entwickler möchte ich Funktionen implementieren, die erlauben, DICOM-Objekte in VR zu zoomen und zu rotieren, um medizinische Daten flexibel anzuzeigen.	DICOM-Objekt lässt sich in VR mit Handtracking vergrößern, verkleinern und drehen.
Marching Cubes Grundlagen recherchieren	Als Entwickler möchte ich grundlegende Informationen über das Marching Cubes-Verfahren recherchieren, um später 3D-Modelle aus DICOM-Daten zu generieren.	Dokumentation mit grundlegenden Informationen und Beispielen zu Marching Cubes.
Unity- und C#-Alternativen für 3D Rendering recherchieren	Als Entwickler möchte ich alternative Programmiersprachen und Frameworks recherchieren, die mit Unity für 3D-Rendering verwendet werden können, um zu prüfen, ob es bessere Optionen als C# gibt	mindestens 3 alternative Programmiersprachen und Frameworks für Unity untersucht
Kompatibilität von 3D Slicer Segmentierung und Export für Unity prüfen	Als Entwickler möchte ich überprüfen, ob und wie die Segmentierungs- und Exportfunktionen von 3D Slicer für Unity kompatibel sind, um sicherzustellen, dass diese Tools in unserem Projekt verwendet werden können.	3D Slicer überprüft: Tools für Herz Segmentierung getestet, Import- und Exportformat überprüft, bleibt Innenleben des Objektes erhalten? beantwortet
Unterschiede und Besonderheiten von DICOM-Daten recherchieren	Als Entwickler möchte ich Informationen über die speziellen Anforderungen und Herausforderungen bei der Nutzung von DICOM-Daten recherchieren, um die besten Praktiken zu identifizieren.	Recherche-Dokument mit spezifischen Informationen rund um DICOM Datenformat erstellt

4. Aufwandschätzung

InnoLab 1: Intuitive Schätzung der für das gesamte Semester

Es ist notwendig, den gesamten Aufwand für das Semester abzuschätzen, obwohl die exakten Aufgaben zu diesem Zeitpunkt noch nicht vollständig definiert sind. Dazu wird mehr Wissen rund um die spezifische Thematik im Laufe der Sprints gesammelt, um detailliertere Aussagen zu machen. Die Schätzung basiert daher auf einem groben Überblick sowie unseren Erfahrungen aus früheren Projekten. Die folgenden Schätzungen beziehen sich auf die High Level Goals „Rendern von DICOM Bilder“ und „Zoom und Rotation des Herzens“ wie im Kapitel „2. Projekt- Kurzbeschreibung“ beschrieben:

Aufgaben	Geschätzter Aufwand
Recherche und Analyse für Wissensaufbau	80h
DICOM Import	120h
Handtracking-Integration	90h
Dokumentation, Kommunikation, Meetings	30h
Gesamtaufwand	320h

Voraussichtlicher Gesamtaufwand für das gesamte Semester: 320 Stunden.

5. Auslieferung

Benötigte Hardware

Oculus Quest 2

Das VR Projekt unterstützt Handtracking und wurde mithilfe einer Meta Oculus Quest 2 64 GB entwickelt. Die Funktionen für Handtracking (Zoom und Rotation) basieren auf Meta XR All-In-One SDK und sind daher mit Meta VR Brillen kompatibel. Informationen über die Kompatibilität mit anderen VR Brillen können nicht gemacht werden, da diese nicht getestet wurden.

Verbindungskabel

Für die Verbindung zu einem PC wird ein USB- Kabel benötigt. Dieses muss hohe Übertragungsraten gewährleisten. Daher ist ein USB 3.X Kabel (USB A auf USB C oder USB C auf USB C) zu empfehlen. Beim USB- Anschluss am verwendeten PC sollte es sich auch um USB 3.X handeln.

PC

Grundlage für das Importieren von DICOM- Daten ist eine externe Bibliothek, welche auf einem Windows Betriebssystem funktioniert. Dieser sollte idealerweise über eine leistungsfähige Grafikeinheit verfügen, da diese essentiell für VR- Umgebungen ist.

Unter folgender Meta Seite werden Hardwareempfehlungen gelistet:

<https://www.meta.com/de-de/help/quest/articles/headsets-and-accessories/oculus-link/requirements-quest-link/?srsltid=AfmBOopINdQbTlywU3bldRiHsakk8ppa9mGrQPoiI-yZWIB177cBL0SR>

Bei Geräten mit geringerer Leistungsfähigkeit kann es zu einem eingeschränkten Erlebnis durch „ruckeln“ kommen. Sie sind aber dennoch kompatibel (werden also nicht Softwareseitig ausgeschlossen), wodurch eine Verbindung möglich ist.

Benötigte Software

Unity

Das Projekt wurde mithilfe von Unity entwickelt und benötigt daher eine aktuelle Version von Unity HUB. Damit können Projekte verwaltet werden, was die Installation des Projektes vereinfacht.

Meta Quest Link

Um eine Verbindung zwischen Oculus Quest und PC herzustellen, wird Meta Quest Link benötigt. Diese Anwendung ist kostenlos installierbar.

Installation und Aufsetzen des Projektes

Folgende Anleitung bezieht sich auf den aktuellen Stand des Projektes nach dem ersten Semester. Es ist möglich, dass sich die Schritte bis zur Endabgabe verändern (z.B. VR Brille im Standalone- Modus oder ähnliches).

1. Installation von Meta Quest Link:

https://www.meta.com/de-de/help/quest/pcvr/?srsltid=AfmBOorDiueByTQHVeN25C9e1FyL8eKCRae4_vKUHkaUIA5auq_814y

Die Applikation kann kostenlos heruntergeladen werden. Nach der Registrierung bzw. dem Login kann eine Verbindung zur Oculus Quest 2 hergestellt werden.

2. Einstellungen Oculus Quest 2:

Folgende Funktionen müssen in den Einstellungen der VR- Brille aktiviert werden. Diese befinden sich im Einstellungs- Untermenü „Bewegungstracking“:



Damit wird sichergestellt, dass die Brille die Hände (ohne Controller) erkennt und Handtracking ermöglicht.

3. Verbindung von VR Brille und PC

Die Applikation „Meta Quest Link“ muss auf dem PC gestartet werden. Danach kann die Brille per USB- Kabel (Anforderungen an das Kabel siehe darüber) verbunden werden.

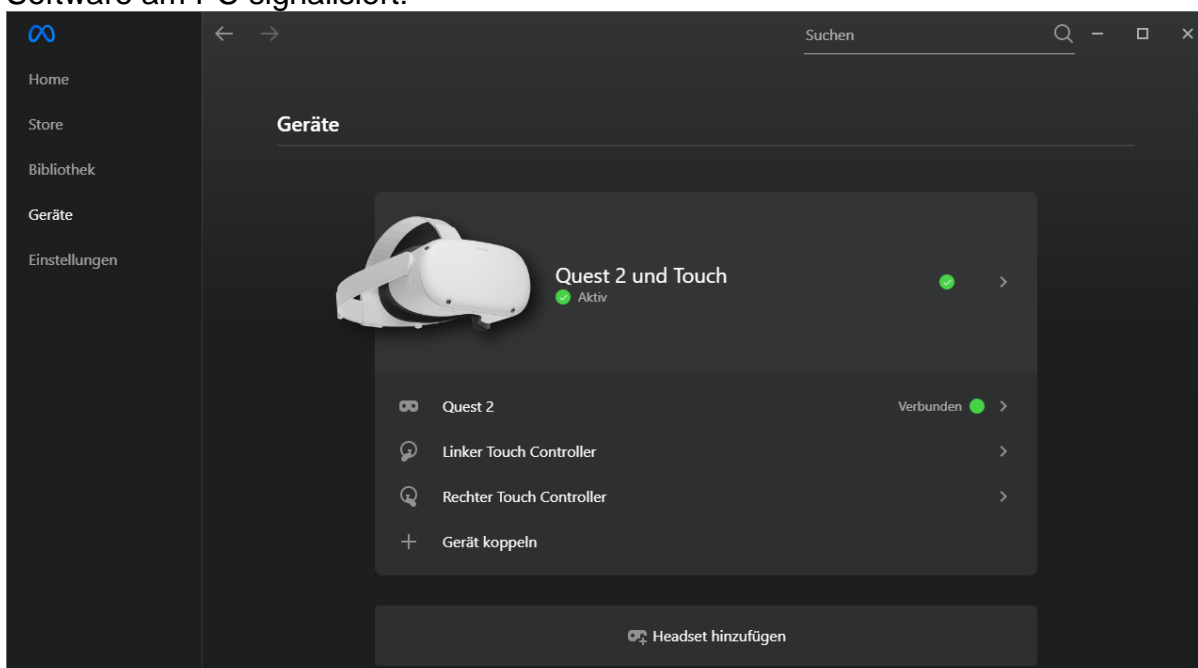
Auf Oculus Quest 2:

- Öffne das Hauptmenü
- Wähle das Uhrensymbol aus, um zu den Schnelleinstellungen zu gelangen
- Wähle Link aus und dann den Computer, mit dem dein Headset verbunden ist (hier erscheint üblicherweise nur der eine verbundene PC)
- Wähle Starten aus

Bei Bedarf können detailliertere Informationen auf der Website von Meta hilfreich sein:

https://www.meta.com/de-de/help/quest/articles/headsets-and-accessories/oculus-link/connect-with-air-link/?srsltid=AfmBOoo0RpjVVFpJOoNQeNZFI5cuf8cKbUo88ur1jhXib_1zK20XT1yS

Eine erfolgreiche Verbindung wird durch grüne Kreise in der Meta Quest Link Software am PC signalisiert:



4. Installation von Unity HUB:

Die Anwendung kann kostenlos auf folgender Seite installiert werden. Dazu wird ein Account benötigt, der ebenso kostenlos ist und per Emailadresse erstellt werden kann:

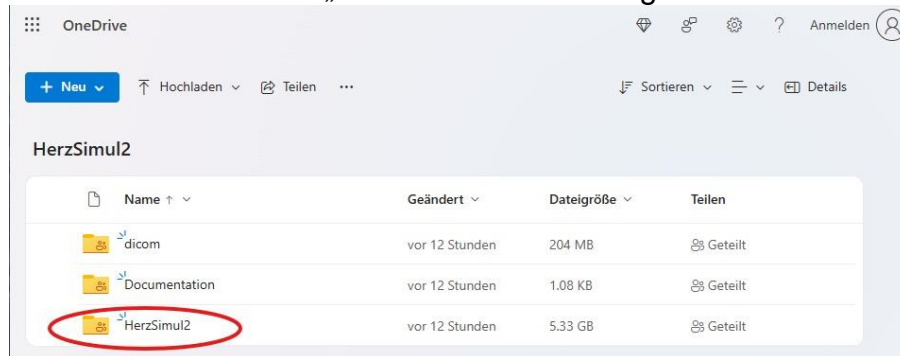
<https://unity.com/de/unity-hub>

5. Herunterladen des Projektes:

Da das Projekt knapp 6 GB aufweist, war es nicht möglich, auf GitHub (Maximum 2GB pro Repository) oder Moodle (Maximum 500 MB) geladen zu werden. Das Projekt kann von OneDrive heruntergeladen werden:

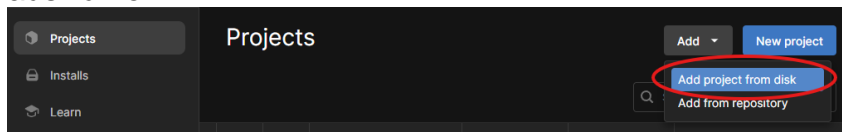
https://1drv.ms/f/s!AsLH_B8P6VjunPcL0HyjaexvXzNoxg?e=6aagC6

Dazu wird der Ordner „HerzSimul2“ heruntergeladen:



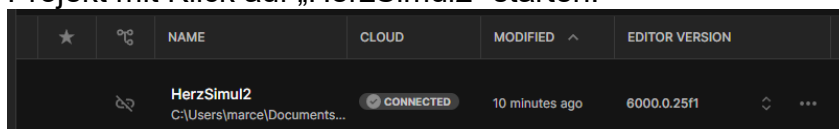
6. Projekt in Unity HUB integrieren:

- a. Klick auf „Add project from disk“ und danach heruntergeladenen Ordner auswählen:

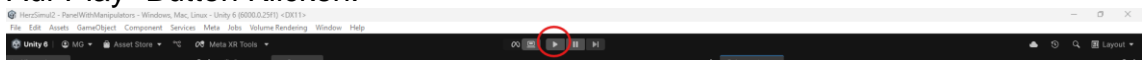


- b. Den heruntergeladenen Ordner „HerzSimul2“ auswählen.

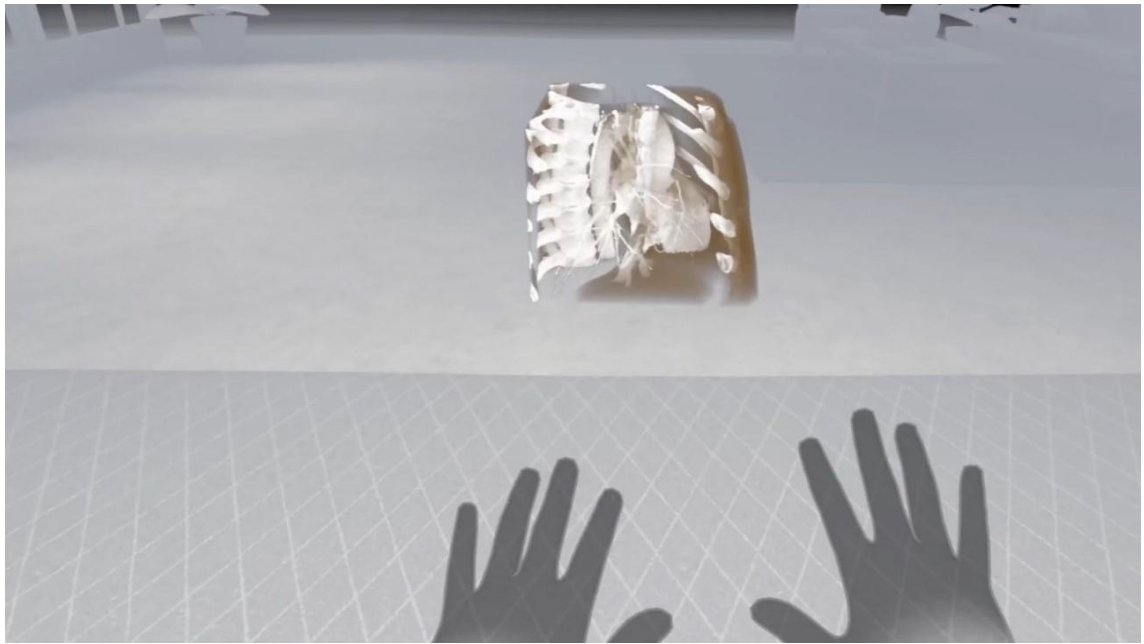
- c. Projekt mit Klick auf „HerzSimul2“ starten:



- d. Auf Play- Button Klicken:



- e. Nun öffnet sich die VR Umgebung auf der Brille (und eine Vorschau in Unity):



Viel Spaß beim Erkunden der Umgebung!

Copyright-Info für externe Bibliotheken:

Meta XR All-In-One: <https://developers.meta.com/horizon/licenses/oculusdk/>

Folgende Voraussetzungen müssen erfüllt werden:

„redistribution of sample source code or other materials must include the following copyright notice: “Copyright © Meta Platform Technologies, LLC and its affiliates. All rights reserved;”“

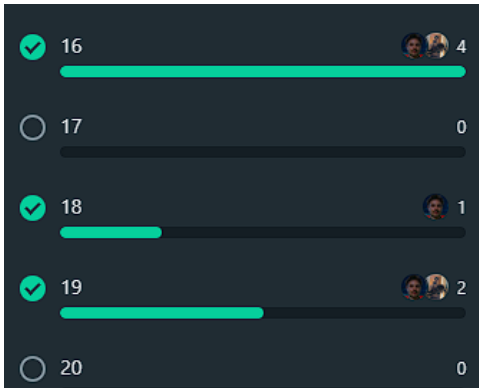
UnityVolumeRendering: [UnityVolumeRendering/LICENSE at master · mlavik1/UnityVolumeRendering · GitHub](#)

„Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software“

6. Unser Projekttagebuch

16.09.2024 KickOff und Team up:

Wir haben eine Gruppe von 5 Personen gebildet. Nach der Vorstellung der verschiedenen Projekte haben wir uns in einem remote Meeting getroffen, um über die Projektauswahl zu diskutieren und abschließend abzustimmen. Dabei war es uns wichtig, dass sich alle Teammitglieder mit dem gewählten Projekt identifizieren können.



Wir haben eine erste Wahl und als Backup eine zweite Wahl definiert. Unser Ziel war es, dass alle Gruppenmitglieder direkt um 20 Uhr so schnell wie möglich abstimmen, um eines unserer bevorzugten Projekte zu erhalten. Wir haben uns für 16_LungSim entschieden.

01.10.2024 Nur ein LungSim Projekt

Wir haben den Betreuer, Herrn Urbauer, per Email kontaktiert und vorgeschlagen, dass wir uns in einem Meeting treffen. Danach haben wir die Info erhalten, dass das Projekt LungSim fälschlicherweise zweimal vergeben wurde. Ein Meeting für die Vorstellung von alternativen Projekten wurde für 02.10. vereinbart.

02.10.2024 Alternatives Projekt

Im Meeting wurde ein alternatives Projekt für den Medizinbereich vorgestellt, das mehrere bestehende Applikationen rund um eMedikation vereinen soll. Beide Gruppen bevorzugten allerdings ein Projekt, das sich mit dem Bereich Bioinformatik befasst. Daher ist die Idee entstanden, ein HerzSim Projekt zu kreieren.

In einem Meeting zwischen unserem Team und der anderen Gruppe, die LungSim wählte, haben wir uns über die Projektwahl beraten und sind zu dem Entschluss gekommen, dass unsere Gruppe an HerzSim arbeiten wird. Diese Entscheidung haben wir dann den verantwortlichen Betreuern mitgeteilt.

08.10.2024 Erste Besprechung zu HerzSim mit Betreuerin

In einem remote Meeting hat uns Frau Lilly Treml die Anforderungen zum Projekt vorgestellt. Diese setzten sich aus 5 Must- Have- Requirements zusammen und aus 2 weiteren Requirements, die wir uns selbst überlegen konnten. Die 7 Requirements sind im Dokument unter Kapitel "2. Projekt- Kurzbeschreibung" einzusehen.

Zusammengefasst geht es darum, ein DICOM Image eines Herzens zu importieren und dieses als Lernwerkzeug in VR darzustellen. Es soll mit Handtracking visualisierbar sein. Darüber hinaus sollen verschiedene Komponenten farblich dargestellt werden, unterschiedliche Schnittebenen zweidimensional darstellbar sein und Komponenten des Herzens mit Labels ausgestattet werden.

Frau Treml hat uns darauf hingewiesen, dass der Import von DICOM Images ein sehr hoher Programmieraufwand ist, wodurch wir uns dazu entscheiden haben, bestehenden Code zu verwenden und eine externe Library zu verwenden.

Wichtige Ressourcen:

Unter folgender Website (Wayback Machine) können DICOM Images kostenlos heruntergeladen: (Pixmeo, 2016)

Informationen zum DICOM-Datenformat Standard: (Dicomstandard, abgerufen 2024)

11.10.2024 Erste Abgabe

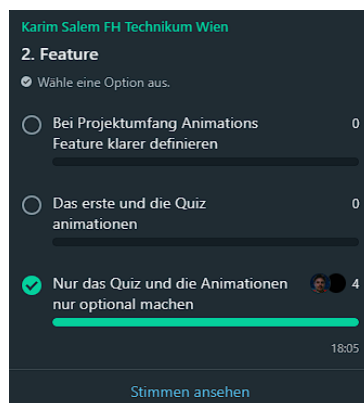
Wir haben uns dazu entschieden, dass Felix die Rolle des Scrum Masters übernimmt und Katarina als Product Owner agiert.

Die ersten zwei Kapitel des Projektstagebuchs wurden ausgefüllt. Dazu haben sich Felix und Marcel in einem remote Meeting getroffen, um das Projekt zu planen und anschließend einen Entwurf des Projektstagebuchs erstellt. Es wurden 2 neue Requirements festgelegt: Labels der unterschiedlichen Herzkomponenten und Strömungsfluss des Bluts simulieren.

Die restlichen Teammitglieder haben anschließend Feedback zum Entwurf gegeben und die überarbeitete Version wurde an Frau Tremml übermittelt.

14.10.2024 Feedback zur ersten Abgabe

Frau Tremml hat uns darauf hingewiesen, dass wir die Anforderungen an das Simulieren des Strömungsflusses nicht klar definiert haben und dass die Information in diesem Aspekt unserer Roadmap nicht mit den definierten Anforderungen übereinstimmen. Wir haben uns daraufhin innerhalb des Teams beratschlagt und hatten die Idee, ein alternatives Feature anstatt des Strömungsflusses zu implementieren: ein interaktives Quiz für das Lernwerkzeug.



15.10.2024 Feinschliff erste Abgabe

Das Projektstagebuch wurde angepasst, Korrektur gelesen und abgegeben. Die Requirements wurden mit Zeitabschätzungen versehen und dienen als Übersicht für die Sprints.

14.10. – 04.11 Sprint 1

Req ID	short description [opt]	Estimate [h]	Real Efort [h]	I
1	Research existing DICOM import Libraries	15	14	
2	Test existing DICOM Libraries	20	30	
3	Get To Know Unity	15	18	

Unser Plan war es, dass sich jedes Teammitglied mit Unity beschäftigt, um die Grundlagen des Programms zu verstehen. Wir haben uns darauf geeinigt, die aktuelle Version von Unity (6) zu verwenden. Die Festlegung einer gemeinsamen Version soll vor allem im weiteren Verlauf des Projektes verhindern, dass versionsabhängige Probleme auftreten.

Interessantes Video als Unity Intro: (Unity Tutorial Deutsch (Komplettkurs), 2022)

Darüber hinaus sollten bestehende Lösungen rund um den Import von DICOM Daten recherchiert und getestet werden.

Die ursprünglich angedachte Library hat sich nicht als bevorzugte Lösung herausgestellt, da es nach der Verknüpfung mit Unity mehrere Kompilationsfehler gab.

04.11 Sprint 1 Teambesprechung

Die Erkenntnisse aller Teammitglieder wurden diskutiert und zusammengefasst. Wir haben auch

über den weiteren Projektverlauf gesprochen, die Erfahrungen aller Teammitglieder und wie wir weitere Requirements umsetzen könnten.

Folgende Bibliotheken wurden recherchiert bzw. getestet:

Name	Link	Kosten	Getestet?	Anmerkungen
Animated Realistic Heart 3D	https://assetstore.unity.com/packages/3d/characters/animated-realistic-heart-3d-101942?srsltid=AfmBOopYpKNGTf8Bp6LUM4uqyHuVILKTKCwb90YEV0KUV0V0HGL	55	nein	Package, das ein pumpendes Herz zeigt, kein DICOM import
Motion open Heart	https://assetstore.unity.com/packages/3d/characters/motion-open-heart-102159	55	nein	Package, das ein aufgeschnittenes Herz zeigt (Innenleben des Herzens)
Unity Medical Visualization	Onlinekurs: Unity Medical Visualization: 01 Preparation and Design LinkedIn Learning, früher Lynda.com	Linkedin Premium (gratis Probemonat, danach 15€)	nein	Tutorial, wie man 3D Herz erstellen und darstellen kann in Unity
Volume and Surface Rendering of 3D Medical Datasets in Unity	https://miccai-sb.github.io/materials/Submission3_Volume%20and%20Surface%20Rendering%20of%203D%20Medical%20Datasets%20in%20Unity.pdf	0	ja	Funktioniert nicht mit DICOM Datenformat, nur mit .jpg, .bmp oder .png (DICOM muss vorher also umgewandelt werden in externer SW)
KTVR Working with DICOM data in Unity	KTVR Working with DICOM data in Unity – KNOW THYSELF AS A VIRTUAL REALITY	0	ja	Bis jetzt funktioniert etwas mit oculus nicht (auf der Website steht, dass man Unity 2019.3.4f1 benötigt!) aber trotzdem spannend weil VR Handtracking integriert. Habe es getestet mit Unity 6000.0.25f1 und mit Oculus Integration (Deprecated) Package aber es erscheinen 15 errors related to oculus
UnityVolumeRendering	https://github.com/mia-vik1/UnityVolumeRendering	0	ja	Karim hat Library mit kleiner Änderung zum Laufen gebracht (DICOM Import funktioniert)
DICOMVolumeRendering	https://github.com/hecomi/UnityDICOMVolumeRendering	0	ja	Ursprünglich angedachte Bibliothek: gibt build errors bei Kompilierung
Master Projekt mit DICOM Import und VR	GitHub - tobias2912/VR-DICOM-viewer: VR project for interacting with DICOM data from CT scanned fractures	0	ja	Open Source! Könnte sehr spannend für uns sein, gibt eine ganze Publikation dazu -> gibt sehr viele Errors (sowohl im original Editor als auch mit Unity 6

05.11. Sprint Review 1

Katarina Gmeiner hat uns mitgeteilt, dass sie nicht weiter an der FH Technikum studieren und daher das Team verlassen wird. Wir bedanken uns für ihre Hilfe! ☺ Dadurch wird Marcel Gössl die Aufgaben als Product Owner übernehmen. Durch die kleinere Teamgröße werden eventuell im 4. Semester Anpassungen bei den Requirements gemacht. Bis zum 4. Semester werden wir Erfahrung mit Unity und VR sammeln und daher besser abschätzen können, inwiefern der Projektumfang für vier Personen realistisch ist.

06.11. - 18.11 Sprint 2

Für die Planung des zweiten Sprints haben wir uns während eines Online- Meetings beratschlagt und folgende Items definiert:

short description [opt]	Estimate [h]
VR Umgebung aufsetzen + mit DICOM Library verknüpfen	60
Organisation VR Brillen	2
Recherche rund um VR in Unity	7
Test Libraries mit VR Support	5

Es war uns wichtig, dass wir mindestens 2 gleiche VR- Brillen für das Semester ausleihen konnten. Das war über den Bereich Computer Science nicht möglich. Daher haben wir uns auf den Vorschlag von Frau Tremel an vasil.vodenicharov@technikum-wien.at gewendet und wir konnten uns 2 Oculus Quest 2 Brillen ausleihen. Für nächstes Semester haben wir 2 VR- Brillen über den Studiengang Informatik reserviert, da uns Herr Vodenicharov darum gebeten hat, die Ausleihe zunächst über die Informatik Abteilung zu versuchen.

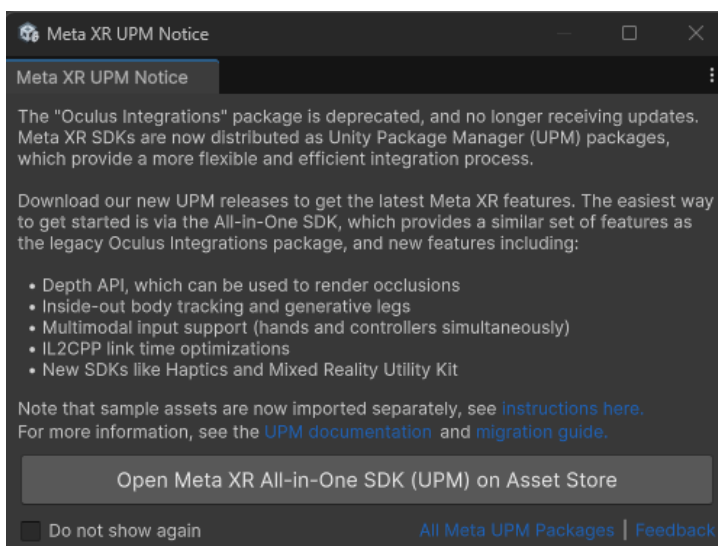
Nach einer Recherche rund um VR in Unity und dem Testen von vorhandenen Libraries war es unser Plan, eine VR-Umgebung zu erstellen und diese mit der DICOM Library zu verknüpfen. Bei letzterem Ziel hatten wir noch keine konkrete Vorstellung, wie wir dieses verwirklichen können und daher nur eine grobe Zeiteinschätzung mit 60h gemacht.

Aufgrund der intensiven Prüfungszeit Mitte November war es uns nicht möglich, die geplanten 60h für dieses Item aufzuwenden. Wir haben uns daher dazu entschieden dieses Ziel in Sprint 3 umzusetzen. Durch unsere Recherche rund um das Thema war es uns möglich, das Ziel in mehrere, kleinere Ziele zu unterteilen. Unser Learning: Deadlines (Prüfungen, Abgaben und andere Deadlines) bei der Sprintplanung genauer einbeziehen, damit die Zeit für die geplanten Items auch tatsächlich aufgewendet werden kann.

18.11 Sprint 2 Teambesprechung

Vor dem Sprint Review mit Frau Tremel haben wir unsere Erkenntnisse geteilt und zusammengefasst. Diese dienten auch als Grundlage für den kommenden Sprint 3:

Learnings zu VR: Obwohl es im Internet sehr viele Informationen und Tutorials rund um das Unity Package „Oculus Integration (Deprecated)“ gibt, wird dieses nicht mehr unterstützt. Das aktuellere Package ist „Meta XR All-in-One SDK“, das umfangreicher in den integrierten Funktionen ist aber dadurch auch erheblich mehr Speicherplatz benötigt (mehrere GB). Die Oculus Quest 2 Brillen können aber problemlos mit diesen Datenmengen umgehen.



19.11. Sprint Review 2

Heute haben wir die Ergebnisse präsentiert. Beim Sprint Review wurden die Anforderungen detaillierter formuliert, und es gab Tipps zur besseren Strukturierung. Im Sprint Planning wurden die Anforderungen durchgegangen und Probleme besprochen, außerdem gab es organisatorische Hilfestellung. Das Project Diary wurde auf die neue Situation angepasst. Abschließend haben wir uns darauf geeinigt, dass wir mit der andern InnoLab Gruppe „LungSim“ in Kontakt treten werden.

20.11 – 02.12 Sprint 3

Da wir im letzten Sprint durch intensive Prüfungsvorbereitungen weniger Zeit als geplant aufwenden konnten und das Ziel „VR- Umgebung aufsetzen + mit DICOM Library verknüpfen“ aus Sprint 2 nicht vollständig implementieren konnten, haben wir es in kleinere Ziele unterteilt: „VR- Umgebung aufsetzen: Hello world“, „DICOM Library mit VR verknüpfen“. Folgende Ziele haben wir für diesen Sprint geplant, die sich vor allem auf Handtracking beziehen:

Req ID	short description [opt]	Estimate [h]	I
1	VR Umgebung aufsetzen: Hello world	15	
2	DICOM Library mit VR verknüpfen	10	
3	Basics Handtracking: Reasearch	15	
4	Basics Handtracking: draft Implementierung	15	
5	GitHub Problem wegen pushen lösen	2	
6	Verbindungsproblem mit Meta Link lösen	2	

Zu Beginn des Sprints haben wir festgestellt, dass die Verbindung zwischen VR-Brille und Unity Probleme verursacht. Während sie bei Felix funktioniert hat, konnte Marcel das VR -Projekt nicht auf die Brille übertragen. Es war nur möglich, einen build eines Projektes mithilfe von „Meta Quest Developer Hub“ auf die Brille zu laden und ohne direkte Verbindung zum PC zu arbeiten. Deshalb war die Vermutung, dass es ein Problem mit der Verbindung über Meta Quest Link gibt. Daher haben wir das Ziel „Verbindungsprobleme mit Meta Link lösen“ hinzugefügt. Für die Lösung des Problems mussten mehrere unterschiedliche Ursachen identifiziert und ausgeschlossen werden. Durch eine Internetrecherche konnten wir folgende mögliche Ursachen definieren:

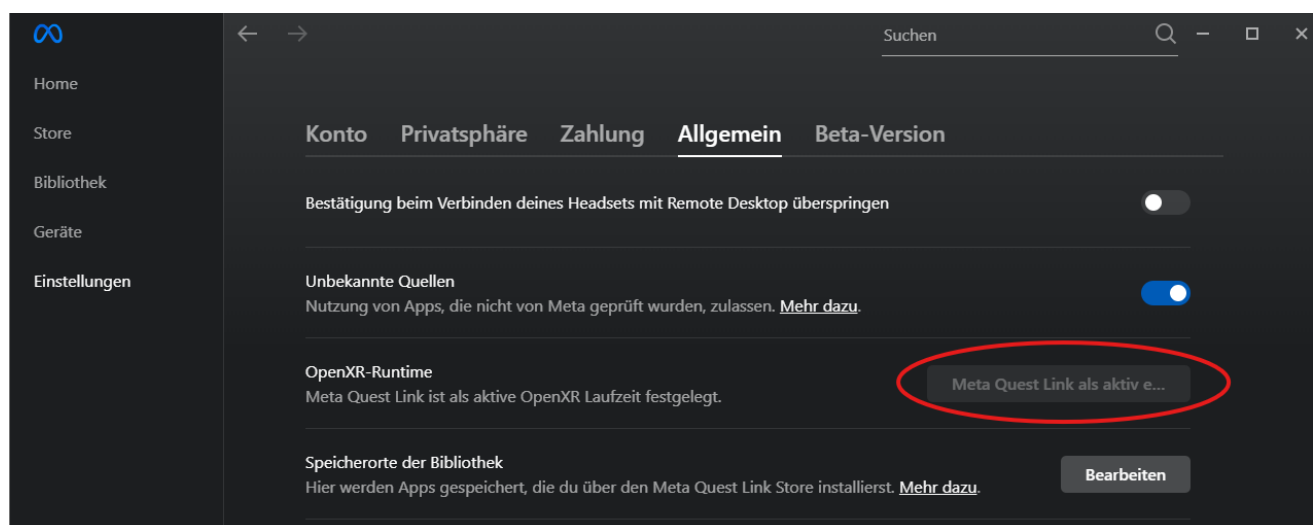
- Falsche Einstellungen
- Zu „schwache“ Hardware für VR (Ryzen 3 4300U mit 8 GB DDR4 RAM)
- Verbindungskabel (mitgeliefertes Ladekabel und andere Handyladekabel haben nicht funktioniert)

Folgende Voraussetzungen sollen daher für die Verbindung zwischen Oculus Quest 2 und PC via Meta Quest Link sichergestellt werden und haben in unserem Fall das Problem gelöst:

Einstellungen:

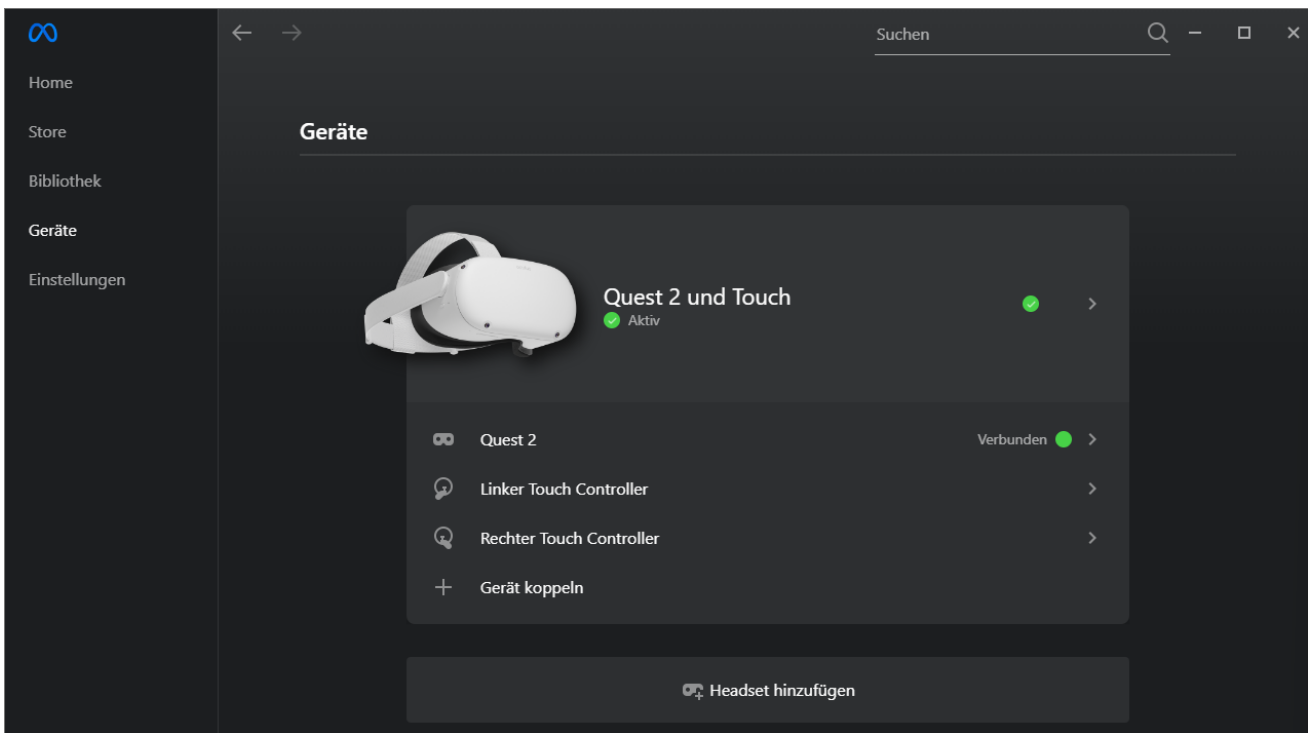
Meta Quest Link: OpenXR- Runtime:

Button muss gedrückt werden, sodass dieser „ausgraut“ und die Schrift „Meta Quest Link als aktiv e...“ anzeigt:



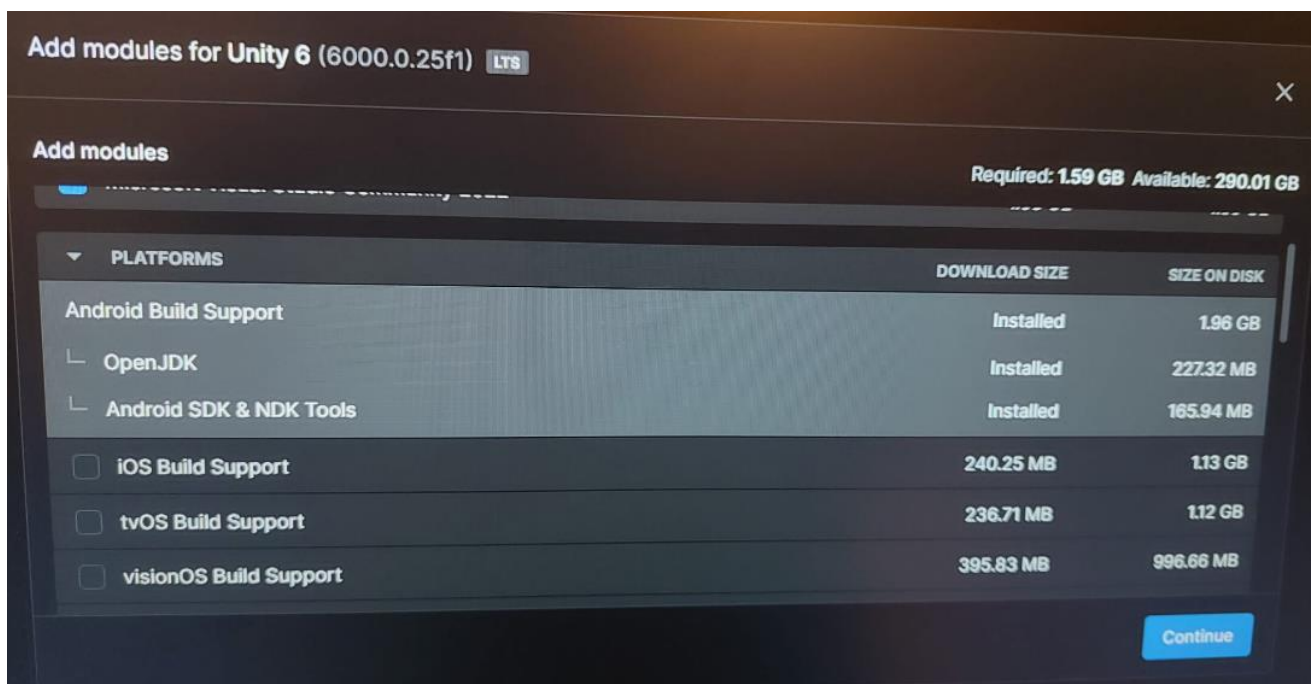
Meta Quest Link: Gerät aktiv

Die VR- Brille wird bei einer aktiven Verbindung unter „Geräte“ mit einem grünen Haken und dem Wort „aktiv“ angezeigt:



Unity Einstellungen:

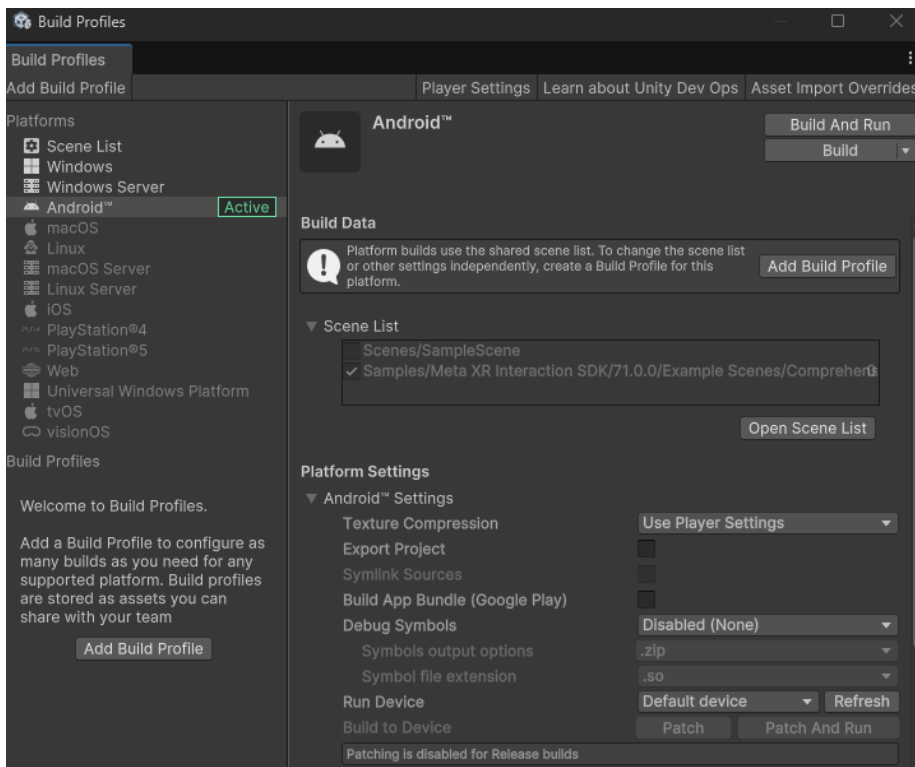
Android Build Support Installieren bei erstmaligem Öffnen des Projektes:



Android Build:

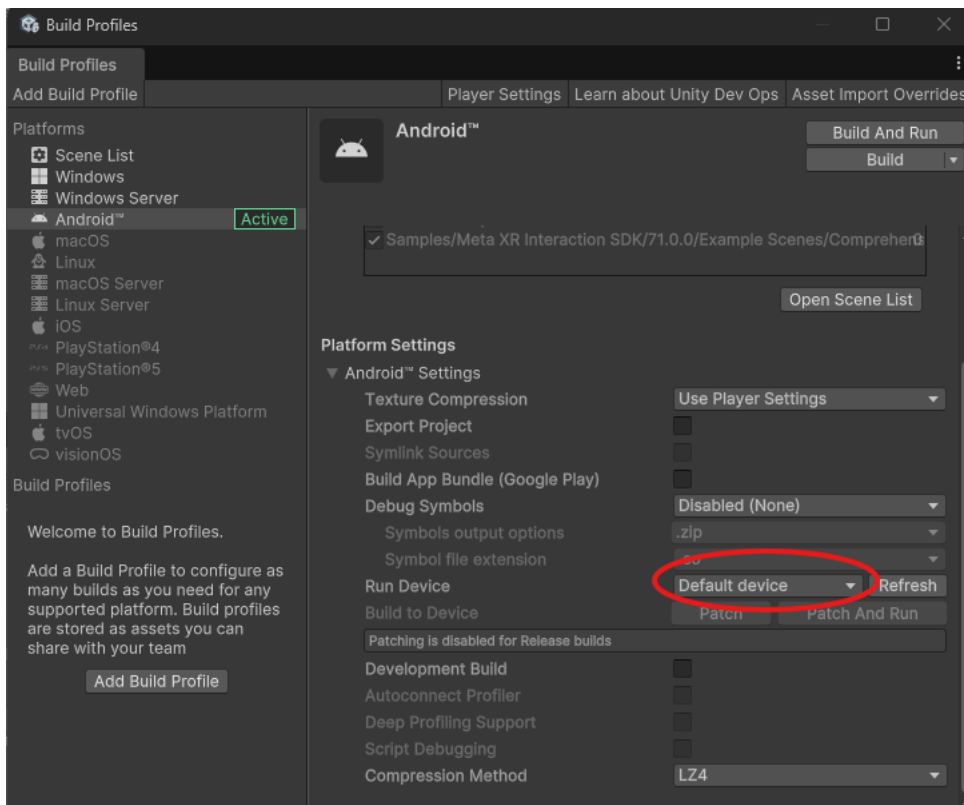
Klicke auf "File -> Build Profiles". Hier sollte ein grünes Rechteck mit „Active“ neben Android stehen.

Falls das nicht der Fall ist, klicke auf „Android -> Switch Platform“:



Default Device:

Im selben Einstellungsfenster kann das default device ausgewählt werden:

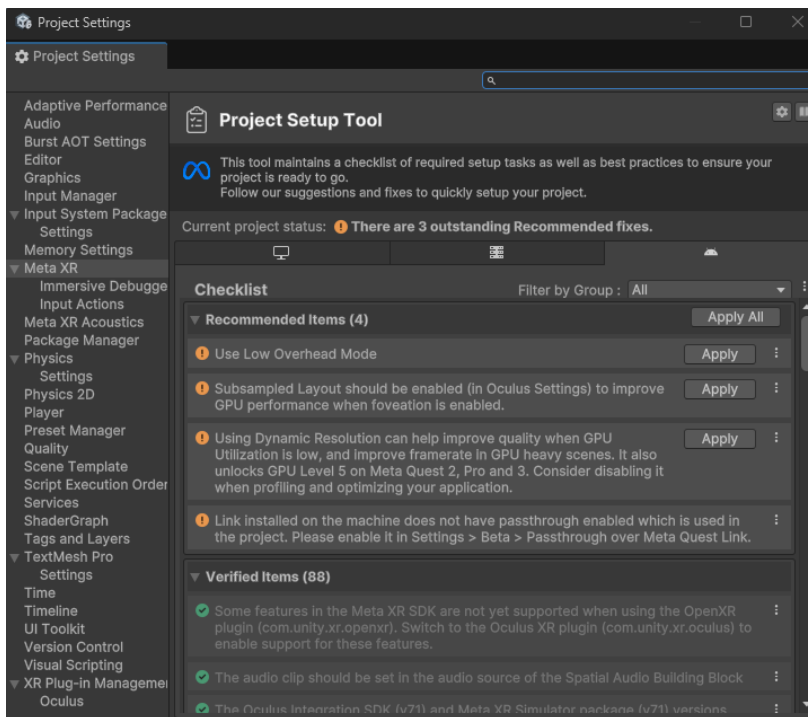


Bei einer aktiven Verbindung zur VR- Brille kann diese als default ausgewählt werden.

Meta XR Project Setup Tool:

Klicke in Unity auf "Edit -> Project Settings -> Meta XR"

Hier werden eventuell Errors angezeigt. Diese können meistens mit dem Klicken auf „Apply“ behoben werden.



PC Hardware Voraussetzungen:

Die Voraussetzungen werden auf folgender Seite gelistet:

(Meta, 2024)

Für die Link Verbindung ist vor allem die GPU essenziell. Falls der verwendete PC die Anforderungen nicht erfüllt, kann aber trotzdem eine Verbindung hergestellt werden. Das VR-Erlebnis ist dadurch eventuell „ruckelig“.

Kabel:

Meta empfiehlt ihr offizielles Meta Quest Link Kabel. Dieses kostet jedoch rund 100€. Wie oben erwähnt funktioniert das mitgelieferte Oculus Quest 2 Kabel nicht für die Verbindung mit Computer. Der Grund dafür ist, dass es sich dabei um ein USB C 2.0 Kabel handelt, das keine hohen Übertragungsraten liefert. Für die Verbindung können laut unserem derzeitigen Wissenstand folgende Kabel verwendet werden:

USB A (3.X) auf USB C (3.X) oder USB C (3.X) auf USB C (3.X). Dabei muss der verwendete USB-Anschluss, an den das Kabel angeschlossen wird, auch mind. ein USB A/C 3.0 (oder höher) Anschluss sein. Das Kabel sollte eine ausreichende Geschwindigkeit bereitstellen (ca. 5 Gbp/s oder mehr). Es empfiehlt sich ebenso, ein Kabel mit einer Mindestlänge von 2 Meter zu verwenden, da die Bewegungsfreiheit ansonsten eingeschränkt ist.

Folgendes Kabel hat bei den Verbindungsproblemen Abhilfe geschafft:

https://www.mediamarkt.at/de/product/_isy-datenkabel-usb-c-3-0-iuc-3200-1653863.html?utm_term=3957&utm_source=google&utm_medium=cpc&utm_campaign=rt_shipping_na_nsp_na_SFBO-Pmax-F-Base&gad_source=1&gclid=CjwKCAiAmfq6BhAsEiwAX1jsZ504_KCQ56Uc2Qx8UfRsiP_o130zBcP0kVvfj5MRKaKfplQHPjCdehoCWdgQAvD_BwE&gclsrc=aw.ds

GitHub Problem:

Wir haben festgestellt, dass es bei GitHub ein 2GB Limit pro File gibt. Daher ist es nicht möglich, unser Projekt vollständig hochzuladen. Wir haben daher vereinbart, dass wir größere Dateien per USB- Stick oder OneDrive austauschen und nur die Dateien, die häufig bearbeitet werden, auf GitHub laden.

Es war möglich eine VR Umgebung aufzusetzen und darin importierte DICOM's darzustellen. Handtracking konnte mithilfe vom Package „Meta XR All-in-One SDK“ verwirklicht werden. Dabei hat vor allem dieses YouTube Tutorial geholfen, in dem die wichtigsten Einstellungen und

Package- Integrationen in Unity erklärt werden: (Tutorials, 2024)

Wichtige Einstellungen auf der VR Brille sind folgende bezüglich Handtracking:



Diese 3 Optionen müssen aktiviert werden.

„Oculus Integration (Deprecated)“ haben wir nicht erfolgreich implementieren können: Handtracking war nur mithilfe von den Controllern möglich. Die Hände wurden nicht angezeigt und es war auch nicht möglich, mit Objekten zu interagieren.

03.12. Sprint Review 3

Wir haben besprochen, dass beim Semesterziel von InnoLab 1 („1. Rendern von DICOM Bilder und Darstellung des Herzens“) auch die Herz- Segmentierung aus DICOM Daten erfolgen sollte. Dazu gibt es Algorithmen, wie man die Grauwerte von DICOM Daten so berechnen kann, dass das Herz von den restlichen Teilen (Gewebe, andere Organe, Knochen usw.) heraus segmentiert wird. Herz ist vom Gewebe unterschiedlich als andere Dinge, die rundherum liegen. Man kann es theoretisch automatisch rausrechnen. Infos:

(Wikipedia, 2024)

(Wikipedia, 2024)

(Boyer, 2023)

Das kann mit externer Software (z.B. Python) gemacht werden oder direkt in Unity. Es wäre auch eine Möglichkeit, dass man mit Handtracking in Unity das Herz rausschneiden kann.

04.12- 16.12. Sprint 4:

In diesem Sprint lag der Fokus darauf, das Unity Projekt mit funktionierendem Handtracking auf die Rechner aller Teammitglieder zu übertragen und damit die Funktionalität zu reproduzieren. Zudem war es unser Ziel, ein DICOM Bild zu importieren und das importierte Objekt in die Handtracking- Umgebung zu integrieren. Darüber hinaus wollten wir Informationen rund um Zoom & Rotation sammeln, um diese spezifische Handtracking- Funktionen später implementieren zu können. Übersicht:

short description [opt]	Estimate [h]	Real Efort [h]
Funktionierendes Handtracking in VR auf allen Rechnern (Akzeptanzkriterium: VR Umgebung mit Handtracking funktioniert auf dem Rechner jeden Teammitglieds)	15	9
DICOM Library mit VR Handtracking Projekt verknüpfen als Basis für spätere DICOM Handtrackings (Akzeptanzkriterium: DICOM Objekt ist in VR Umgebung implementie...	20	18
Handtracking Zoom + Rotation basics Recherche (Akzeptanzkriterium: Informationen zu Zoom und Rotation wurden gesammelt)	20	12
Zoom und Rotation mit DICOM Objekt implementieren (Akzeptanzkriterium: DICOM Objekt kann gezoomt und gedreht werden)	0	14

Das Projekt konnte auf die bisher noch nicht getesteten Rechner übertragen werden. Grundlage dafür war die detaillierte Beschreibung der Einstellungen in Sprint 3. Es war auch möglich, das importierte DICOM Objekt mit Handtracking- Funktionen auszustatten. Dabei musste jedoch ein Kompromiss gemacht werden: Die Bibliothek zum Importieren von DICOM Daten funktioniert nur auf einem Windows build während VR Projekte mit Oculus Quest im Standalone- Modus (ohne aktiver Verbindung zum PC mit Unity Projekt) einen Android build benötigen (Oculus Quest 2 basiert auf Android und kann daher nur mit Android builds umgehen). Es wurde versucht, das Projekt im Windows build mittels Meta Quest Developer Hub (mit diesem Programm können Programme von Entwicklern auf die Brille geladen werden) auf die Brille zu

laden, was einen Fehler verursachte. Auch im Internet wird davon berichtet, dass dies nicht möglich sei. Im Hinblick auf das Endprodukt könnte das eine Einschränkung sein, da die User eventuell nicht immer Zugang zu einem PC bei der Verwendung der Brille haben. Andererseits ist die Hardware einer VR Brille durch ihre kompakte Bauform eingeschränkt und verringert die Auflösung einiger Objekte, was das Lernerlebnis beim Betrachten detaillierter anatomischer Strukturen auch einschränkt. Daher wäre die Notwendigkeit einer aufrechten Computerverbindung aus mehreren Gründen ein Vorteil. Wir haben uns daher dazu entschieden, vorerst mit dem Windows build weiterzuarbeiten.

Zoom und Rotationsfunktionen:

Bei der Recherche rund um Zoom- und Rotationsfunktionen mittels Handtracking ist uns aufgefallen, dass es nur begrenzte Informationen zu diesen spezifischen Anforderungen gibt (die meisten Objekte in VR können per Hand aufgehoben werden). Wir haben uns daher dafür entschieden, einen praktischeren Ansatz zu verfolgen und die Funktionalitäten direkt zu implementieren versucht. Dazu haben wir eine von Meta erstellte Sample-Scene modifiziert. Darin gab es ein ähnliches Objekt, das Funktionen für Vergrößerung und Rotation unterstützte. Darauf basierend haben wir diese Einstellungen an unsere Bedürfnisse angepasst und konnten nach einer trial and error Phase die Funktionalität auf das importierte DICOM Objekt anwenden.

16.12. Sprint Review 4:

Ein zentrales Thema der Besprechung waren die Einschränkungen des DICOM-Importers, speziell im Zusammenhang mit Windows-Builds. Diese Problematik erfordert eine Dokumentation der limitierten Funktionalität und die Festlegung einer fundierten Begründung, warum trotz der Einschränkungen am aktuellen Ansatz festgehalten wird. Alternativen wie Bibliotheken in Python oder MATLAB wurden diskutiert, darunter Viewer in C#, welche zusätzliche Flexibilität bieten könnten. Ergänzend wurde hervorgehoben, dass im Projekttagbuch detailliert auf diese Herausforderungen eingegangen und die durchgeführten Machbarkeitsanalysen zu verschiedenen Bibliotheken beschrieben werden sollen.

Für die Demo war zentral, dass Zoom- und Rotationsgesten mit dem Hand-Tracking für DICOM-Objekte funktional umgesetzt wurden. Während diese Aspekte erfolgreich demonstriert wurden, zeigte sich noch Optimierungspotenzial hinsichtlich der Bildqualität der DICOM-Objekte, insbesondere bei vergrößerter Darstellung. Als Lösungsansatz könnten Filtermethoden untersucht werden, um Unschärfe zu minimieren.

Darüber hinaus wurde beschlossen, sich intensiver mit dem DICOM-Format zu beschäftigen, insbesondere mit den enthaltenen Metadaten und den Algorithmen zum 3D-Rendering. Verschiedene Programme und Bibliotheken wie MATLAB (Marching Cubes und - Triangles, Surf-Funktion) wurden im Hinblick auf zukünftige Implementierungen evaluiert. Das Verständnis dieser Grundlagen soll künftig auch zur Erstellung und Verarbeitung anatomischer Modelle genutzt werden, wie etwa detaillierter Herzmodelle, die Klappen und Gefäße berücksichtigen.

Als Ziel für zukünftige Sprints wurde festgelegt, eine Bibliothek zu suchen, die mit den Projektanforderungen kompatibel ist, oder ein alternatives Verfahren zu etablieren, das sowohl das Laden als auch das Rendern von DICOM-Daten ermöglicht. Dabei sollen auch Möglichkeiten zur Manipulation von Objekten wie Skeletten, Voxel- oder Dreiecksmodellen geprüft werden. Der derzeit verwendete DICOM- Importer hat zwar einige Vorteile (wie automatischer Import), jedoch ist unklar inwiefern die Rohdaten des DICOM Bildes bearbeitet werden können. Das ist vor Allem in Hinblick auf die Ziele der kommenden zwei Semester wichtig (Herz- Segmentierung, farbliche Darstellung der Komponenten, Labeling).

18.12. – 13.01. Sprint 5:

Im 5. Sprint haben wir nach Absprache in einem gemeinsamen Meeting folgende Ziele festgelegt:

ID | short description [opt]

- 1 Machbarkeitsanalyse DICOM Bibliothek (Akzeptanzkriterium: Wissen, ob alle Requirements umgesetzt werden können)
- 2 DICOM viewer in beliebige Programmiersprache studieren und Datenformat verstehen (Akzeptanzkriterium: Dokument mit Infos über DICOM)
- 3 Grundlagen 3D rendering MarchingCubes recherchieren (Akzeptanzkriterium: Dokument mit Infos über Grundlagen)
- 4 Kompatibilität zwischen Unity + Quest + andere Programmiersprachen als C# recherchieren (Akzeptanzkriterium: Liste an Programmiersprachen)
- 5 3D Slicer Segmentierung + Export für Unity testen (Akzeptanzkriterium: Vor- und Nachteile der Anwendung kennenlernen)

Studie zu DICOM Viewern und Datenformaten:

Um das DICOM Datenformat besser zu verstehen, haben wir uns mit DICOM- Viewer in Java und Python beschäftigt. Darüber hinaus wurden zusätzliche Informationen aus Artikel gesammelt und alle Erkenntnisse in einem Dokument zusammengefasst. Auch der Java Source Code kann mit unterschiedlichen Daten getestet werden (Ergebnisse auf GitHub im Ordner DICOM_Infos).

Durch das neu gewonnene Verständnis über das Datenformat soll eine bessere Einschätzung bezüglich der Machbarkeitsanalyse der DICOM Bibliothek gemacht werden können.

Machbarkeitsanalyse DICOM Bibliothek:

Die Bibliothek bietet zwar reichliche Informationen über die Verwendung und deren Funktionen. Allerdings gibt es keine Hinweise darauf, wie sie die DICOM Rohdaten verarbeitet. Durch einen praktischen Test einer Funktionalität (Schnittebenen des 3D Objektes) konnten wir zeigen, dass die Bibliothek kein hohles Objekt generiert (das Innenleben des Herzens bleibt also erhalten). Es ist allerdings nicht klar, inwiefern es möglich ist, das importierten Objekt im Hinblick auf die Ziele der nächsten beiden Semester zu manipulieren (Herz segmentieren, einfärben und labeln einzelner Segmente). Ein Workaround wäre, dass die DICOM Daten vorverarbeitet werden, im DICOM Datenformat abgespeichert werden und die bearbeiteten Daten erst dann in Unity importiert werden. Daher haben wir uns dazu entschieden, eine solche Software zu testen. Die am häufigsten verwendete ist 3D-Slicer.

Recherche zu 3D Rendering und Marching Cubes (auch kurzer Überblick zu Blender):

Hier haben wir zu den Grundlagen und der Funktionalität von Marching Cubes recherchiert. Ebenso haben wir noch verschiedene Implementierungsmöglichkeiten dazu angesehen. Dazu haben wir auch ein Dokument mit den gefundenen Informationen erstellt. Das erworbene Wissen kann in Zukunft für das weitere Projekt verwendet werden.

Außerdem haben wir rund um die Funktionsweise und Implementierungsmöglichkeiten von Marching Triangles recherchiert, sind aber zu dem Entschluss gekommen, dass wir eine Implementierung mit Marching Cubes verwenden wollen, da diese Variante so wirkt, als wäre sie einfacher zu implementieren und für Marching Cubes generell mehr Material zur Verfügung steht.

Kompatibilität zwischen Unity, Meta Quest und alternativen Programmiersprachen:

Hier haben wir geprüft welche anderen Programmiersprachen in Kombination mit Unity und der Meta Quest möglich sind, um ggf. weitere Möglichkeiten zu eröffnen. Dabei sind wir zu folgenden Erkenntnissen gekommen. Bei der Entwicklung für Unity in Verbindung mit der Meta Quest ist C# die Hauptsprache, aber andere Optionen wie C++ für leistungsoptimierte Native Plugins, JavaScript (veraltet) und Python (über Drittanbieter-Plugins) sind ebenfalls möglich, um zusätzliche Anpassungen oder spezialisierte Funktionalitäten zu ermöglichen. Shadertypen wie HLSL oder GLSL erweitern die Grafikprogrammierung innerhalb von Unity.

Segmentierung und Export von 3D Slicer:

Falls die DICOM- Import Bibliothek in Unity für die Requirements der kommenden Semester nicht verwendet werden kann, wäre eine externe Applikation für Segmentierungszwecke eine denkbare Alternative. Im Internet wird häufig vom 3D Slicer berichtet. Daher haben wir getestet, ob wir einen DICOM Datensatz mittels 3D Slicer importieren, bearbeiten und die bearbeiteten Daten in die Unity Umgebung integrieren können. Dadurch haben wir folgende Erkenntnisse gewonnen:

- Export als DICOM: Dazu muss man einen Workaround in der Applikation machen, indem man einzelne Segmentierungen exportiert. Es wirkt, als wäre das Hauptaugenmerk der

Anwendung nicht der DICOM Export. Beim Unity Import mit DICOM Importer wird das Segment dargestellt. Es wirkt allerdings als ob die Farbinformationen der Pixel verloren gegangen wären. Auch das Innenleben des Herzens kann nicht mehr dargestellt werden.

- Export als STL: Kann in Unity nicht importiert werden laut verschiedenen Internetquellen, ein Test hat gezeigt dass es tatsächlich nicht funktioniert.
- Export als .obj: Kann einfach exportiert und in Unity integriert werden (ohne speziellen DICOM Importer). Es scheint, als gäbe es auch hier kein Innenleben des Organs mehr (nicht ganz sicher, weil das nicht direkt ohne externe Libraries getestet werden kann)
- MONAILabel: Bietet automatische Herz- Segmentierung basierend auf Deep Learning an. Konnte nicht getestet werden, weil die Installation nicht funktioniert hat (dies erfordert einige Schritte, darunter eine Verbindung zu einem Server).

14.01. Sprint Review 5:

In diesem Sprint lag unser Fokus auf dem DICOM-Datenformat, 3D-Rendering und dem weiteren Vorgehen für die kommenden Semester. Frau Treml hat uns nahegelegt, die DICOM Bilder über ein Preprocessing- Tool in das .obj Dateiformat umzuwandeln. Diese OBJ Daten können mit 3D- Grafiksoftware wie Blender importiert und bearbeitet werden. Mit dem 3D Slicer gab es hingegen Probleme beim Export: STL-Dateien sind für Unity ungeeignet und OBJ-Dateien verlieren scheinbar Details – das konnten wir noch nicht direkt testen. Uns wurde auch ein Projekt auf GitHub namens invesalius (Invesalius, 2007-2025) empfohlen, das wir uns in Zukunft noch genauer ansehen sollten.

Wir haben vereinbart, dass ein solides Fundament für die kommenden Semester oberste Priorität hat. Daher möchten wir basierend auf den Ergebnissen dieses Semesters einen Technologie- Plan aufstellen. Dafür werden wir untersuchen, ob eine Umwandlung von DICOM in OBJ vor der Integration in Unity sinnvoll bzw. umsetzbar ist. Eine andere Lösung wäre, den gesamten Import zu automatisieren. Dabei würde das Herz automatisch aus den DICOM- Daten segmentiert werden und anschließend in ein Unity Objekt umgewandelt werden.

Literaturverzeichnis

- Boyer, A. (2023). Von MarchingNumPy: <https://marchingnumpy.readthedocs.io/en/latest/#> abgerufen
- Dicomstandard. (abgerufen 2024). Von The Medical Imaging Technology Association (MITA): Unity Tutorial Deutsch (Komplettkurs) abgerufen
- Invesalius. (2007-2025). Von <https://invesalius.github.io/about.html> abgerufen
- Meta. (2024). Von Anforderungen an Windows-PCs für Meta Quest Link: <https://www.meta.com/de-de/help/quest/articles/headsets-and-accessories/oculus-link/requirements-quest-link/> abgerufen
- Pixmeo. (2016). *Osirix Viewer*. Von <https://web.archive.org/web/20160903045127/http://www.osirix-viewer.com/resources/dicom-image-library/> abgerufen
- Tutorials, V. (2024). *Youtube*. Von How to Interact in XR: https://www.youtube.com/watch?v=4kGD8q5kEx8&ab_channel=ValemTutorials abgerufen
- Unity Tutorial Deutsch (Komplettkurs). (2022). Von Youtube: https://www.youtube.com/watch?v=VnN5MYQnGak&t=3543s&ab_channel=ProgrammierenStarten abgerufen
- Wikipedia. (21. Dezember 2024). Von Image Segmentation: https://en.wikipedia.org/wiki/Image_segmentation abgerufen
- Wikipedia. (16. Juli 2024). Von Watershed (image processing): [https://en.wikipedia.org/wiki/Watershed_\(image_processing\)](https://en.wikipedia.org/wiki/Watershed_(image_processing)) abgerufen