1️⃣ Dev Doc – *HoloHuman XR: Down the Rabbit Hole Into the Human Body*

1. Project Name & Theme

Working Title:

HoloHuman XR: Down the Rabbit Hole

Tagline:

Step into the human body, peel back its layers, and explore real medical scans from the inside.

Theme Fit – "Down the Rabbit Hole"
Instead of falling into a fantasy world, the user falls into the **inner world of the human body**:

- Skin → muscle → bone → organs → medical imaging
- A glowing "portal" on the chest leads into CT/X-ray space
- The experience feels like diving deeper and deeper into a hidden reality

---

2. Vision – What This Project Is About

High-Level Vision

HoloHuman XR is an extended reality (XR) experience where:

- You see a **life-sized human body** floating in front of you (Quest 2).
- You can **peel back layers** (skin, muscles, skeleton, heart).
- You can toggle a **"scan mode"** that uses **SecureMR** to show real medical imaging (X-rays, CT slices) inside the body.
- You step through a **glowing portal** into a space made of floating scans – walking *inside* the medical data.
- You **feel** the heartbeat and fractures through **Afference** haptic rings.

The long-term vision is a **clinical collaboration tool** where doctors, radiologists, and students explore a shared "patient twin" in XR.

---

3. Problem & Motivation

Today:

- Medical data is scattered:
    - anatomy in textbooks/simulators
    - imaging in PACS viewers on 2D screens
    - vitals on bedside monitors
- Understanding complex cases requires **mental 3D reconstruction** in each clinician's head.
- Patients and students struggle to understand what's going on inside the body.

We want to:

- **Fuse anatomy + imaging** into one spatial, explorable XR workspace.
- Create an **engaging, emotionally powerful** way to understand injury and disease.
- Prototype what **future medical visualization** could look like:
    - immersive
    - collaborative
    - AI- and data-driven

---

4. Target Users & Use Cases

Primary (long-term):

- Radiologists
- Orthopedic surgeons
- Cardiologists
- Medical students / residents
- Patients receiving explanations of their condition

Hackathon framing:

- Use case 1 – *Fracture explanation:*
  A doctor shows a patient where their bone is broken using a full-body XR skeleton + X-ray overlay.

- Use case 2 – *Education:*
  Students peel back layers, then walk into "scan mode" to connect anatomy with imaging.

---

5. Core Experience Flow (User Journey)

1. Entry – "Meet the body"

- User puts on Meta Quest 2.
- They see a full human body model floating in a dark, slightly stylized "rabbit-hole" environment (subtle depth, particles).

2. Layers – "Peel back reality"

- UI shows buttons/toggles for:
  - Skin
  - Muscles
  - Skeleton
  - Heart
- User taps:
  - Skin off → muscles + skeleton visible
  - Muscles off → skeleton-only view
- Transparency sliders let them fade between layers.

3. Heart – "Feel the pulse"

- User focuses on chest.
- Heart model is visible in context.
- A **Heart Rate slider** (e.g., 60–120 bpm) controls:
  - heart beat speed
  - subtle heart glow
- **Afference ring** pulses in sync with the heartbeat.

4. Imaging Portal – "Down the Rabbit Hole"

- A glowing portal appears on the chest (or over a bone with a fracture).

- User taps the portal → it opens as a circular gate.
- Stepping through it transitions them into a **scan space**:
    - floating X-ray or CT slices powered by **SecureMR**
    - each slice is attached to the skeleton / region it corresponds to (e.g., forearm or chest)

5. Fracture Highlight – "Inside the injury"

- They select a predefined fracture case (e.g., distal radius fracture).
- X-ray appears with fracture area highlighted.
- The corresponding bone in the XR skeleton glows red.
- When they reach out to touch the glowing fracture:
    - Afference ring gives a "crack" vibration pattern.

6. Exit & Future Vision

- User exits the portal back to full-body view.
- We explain: in the future, doctors in different departments could explore this same view together on different headsets.

---

6. Feature Set

6.1 Vision Feature Set (Beyond MVP)

These are **future** capabilities (for "Future Plan" section, not 36h scope):

- Real-time connection to **live heart rate / ECG** and vitals
- Multi-user XR sessions (multiple doctors in the same body)
- Real CT/MRI volumetric reconstructions, aligned to anatomy
- AI to detect fractures or lesions automatically and highlight them
- Patient-specific body meshes instead of a generic model

6.2 MVP Feature Set (Hackathon Scope)

Concrete MVP target for 36 hours:

1. **Interactive Layered Body**

- 3D human model with at least:
  - skin shell
  - muscles (optional simple layer)
  - skeleton
  - heart mesh
- Toggle layers on/off.
- Rotate, scale, move around the body.

2. **Heart Rate-Driven Animation (Simulated)**
   - Heart model beats faster/slower based on a **slider-controlled heart rate**.
   - Simple UI: HR value (bpm).
   - No real ECG parsing; just drive animation speed.
3. **X-ray Fracture Overlay**
   - One or two sample X-ray images (e.g., arm fracture).
   - Use SecureMR or preprocessed images to:
     - load the X-ray
     - highlight the fracture region (predefined mask/bounding box)
   - Skeleton bone corresponding to the fracture glows red.
4. **Scan Portal Mode (SecureMR Integration)**
   - Press a button or tap the chest → open a glowing ring portal.
   - Inside the portal:
     - 1–3 CT or X-ray slices are displayed on floating panels.
     - Panels are labeled (e.g., “Forearm X-ray: Fracture here”).
   - These slices are loaded via SecureMR APIs or sample data.
5. **Afference Haptic Feedback**
   - Heartbeat sensation:
     - Afference ring pulses in sync with heart rate slider.

- Fracture touch:
    - Strong short vibration when user touches the glowing fracture bone.

6. **Basic Scene & UI Polish**
    - Simple environment (dark void, grid, or subtle “tunnel” look).
    - Minimal but clean UI (toggle buttons, sliders, small labels).
    - Stable performance on Quest 2.

---

7. Tech Stack

7.1 Hardware (Confirmed)

- **Meta Quest 2**
    - Primary XR headset / dev + demo target.
- **Afference ring(s)**
    - For haptic feedback (heartbeat + fracture feedback).

(We will design architecture so it’s portable to XREAL, Vision Pro, Raven, etc., but we focus on Quest 2 first.)

7.2 Software & Services

- **Engine:** Unity (2021+), C#
- **XR Framework:** Unity XR Interaction Toolkit + OpenXR backend
- **3D Assets:**
    - Base human body model (from a medical/anatomy asset or generic humanoid pack)
    - Skeleton + heart meshes (asset store, open source, or quickly stylized via Meshy)
    - Optional stylized portal / environment assets (Meshy or kitbash)
- **Medical Imaging:**
    - **SecureMR SDK / API** for:
        - loading sample X-ray/CT data safely

        - possibly getting orientation or basic metadata
    - Local fallback: pre-saved PNG/JPEG slices (if integration is heavy)
- **AI / 3D Asset Generation (optional but strong):**
    - **MeshyAI**:
        - to generate stylized organs, portal structures, or fractured bone props
- **Version Control / Repo:**
    - GitHub public repo (MIT license)
- **Devpost & Media:**
    - Devpost project page
    - 30s vertical demo video (TikTok-style per rules)

---

8. System Architecture (Conceptual)

1. XR Client (Unity on Quest 2)

- Handles:
    - rendering body model and environment
    - XR input (gaze, controllers, teleport/locomotion)
    - UI (layers, heart rate, scan mode toggle)
    - Afference ring communication (via their SDK or BLE bridge)
    - mapping SecureMR-provided images onto in-world panels

2. Anatomy Module

- HumanBodyManager:
    - references SkinMesh, MuscleMesh, SkeletonMesh, HeartMesh
    - exposes functions like SetLayerVisible(layer, bool) and SetTransparency(layer, alpha)
- HeartAnimator:
    - has param heartRateBPM

    - controls animation speed or scale pulsation
    - emits events (“OnBeat”) to drive Afference haptics

3. Imaging Module (SecureMR Integration)

- ImagingLoader:
    - loads sample scans via SecureMR or from local path
    - exposes textures + metadata (e.g., “region: forearm”, “modality: X-ray”)
- ScanPortalManager:
    - spawns portal at chest position
    - when user enters portal:
        - activates floating scan panels
        - attaches textures to quads
- FractureHighlighter:
    - for a given case, knows:
        - which bone mesh to glow
        - which region of the X-ray is the fracture
    - uses a material / shader to highlight area

4. Haptics Module (Afference Integration)

- HapticsController:
    - receives events:
        - OnHeartBeat(int bpm)
        - OnFractureTouched()
    - translates them into Afference API calls for:
        - continuous pulses in rhythm
        - single sharp pulses

---

9. Implementation Plan (High-Level 36h Breakdown)

Phase 1 – Core XR & Anatomy (6–8 hours)

- Set up Unity project for Quest 2.
- Import human model, skeleton, heart mesh.
- Implement layer toggles + basic interaction (grab, rotate, teleport).

Phase 2 – Heart Animation + Haptics (4–6 hours)

- Create simple heart animation (scale/position curve or pre-made animation).
- Add HR slider UI + data binding.
- Integrate Afference SDK → basic heartbeat vibration.

Phase 3 – Imaging & Portal MVP (8–10 hours)

- Integrate SecureMR or load sample scan textures.
- Create portal object and transition (fade, position, etc.).
- Build "scan space" with floating slice panels.
- Wire up at least one X-ray with a fracture highlight and matching bone glow.

Phase 4 – Polish & Theme (6–8 hours)

- Add subtle visual polish: particles, lighting, smoother transitions.
- Add simple text labels and tooltips.
- Optimize for performance on Quest 2.
- Record 30-second vertical video.
- Write Devpost "What it does / How we built it / Challenges / Next steps."

Phase 5 – Buffer (remaining time)

- Debug, rehearse pitch, refine flow.
- If there's extra time: tiny stretch (e.g., simple CT slice scrolling, extra haptic patterns).

---

10. Risks & Mitigations

- **Risk: SecureMR integration too complex.**
    - **Mitigation:** Use pre-exported PNG/JPEG from SecureMR ahead of time; still credit them.

- **Risk: Afference integration time-consuming.**
    - **Mitigation:** Minimum integration: one simple vibration pattern on OnHeartBeat(). Extra patterns only if time allows.
- **Risk: 3D assets too heavy/slow.**
    - **Mitigation:** Use lower-poly models, limit number of layers.
- **Risk: Time crunch.**
    - **Mitigation:** Prioritize:
        1. Layered body
        2. Heart + Afference
        3. One solid imaging portal + fracture case

---

11. Future Plan (Post-Hackathon / Immersion League)

After the hackathon, this project is a great candidate for the **Immersion League / continuing track** mentioned in the slides.

Future directions:

- Real-time ingestion of live vitals (HR, SpO_2, BP) from devices.
- Proper ECG waveform parsing + synchronized heart mechanics.
- True CT/MRI volume reconstruction (3D textures, marching cubes).
- Multi-user sessions (Quest 2, XREAL, Vision Pro, etc.) for cross-department case reviews.
- AI models for automatic fracture / lesion detection and severity scoring.
- Patient-specific anatomical models derived from DICOM segmentations.