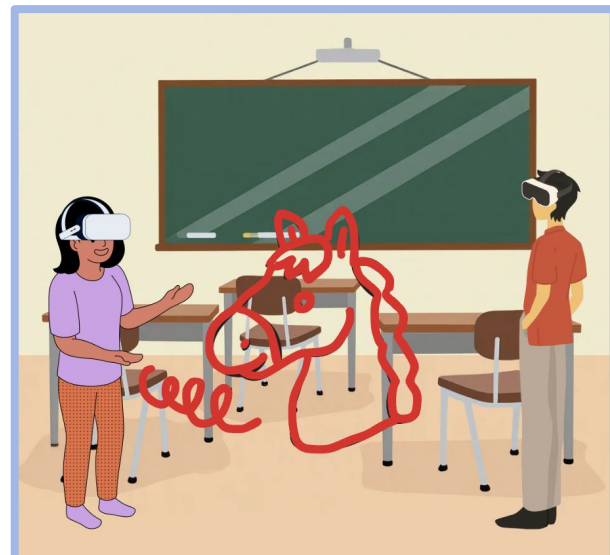


Overview

A collaborative **3D drawing platform** in augmented reality which allows users to create immersive **art** using the **Unity** Game Engine and Microsoft **Hololens 2** headset.

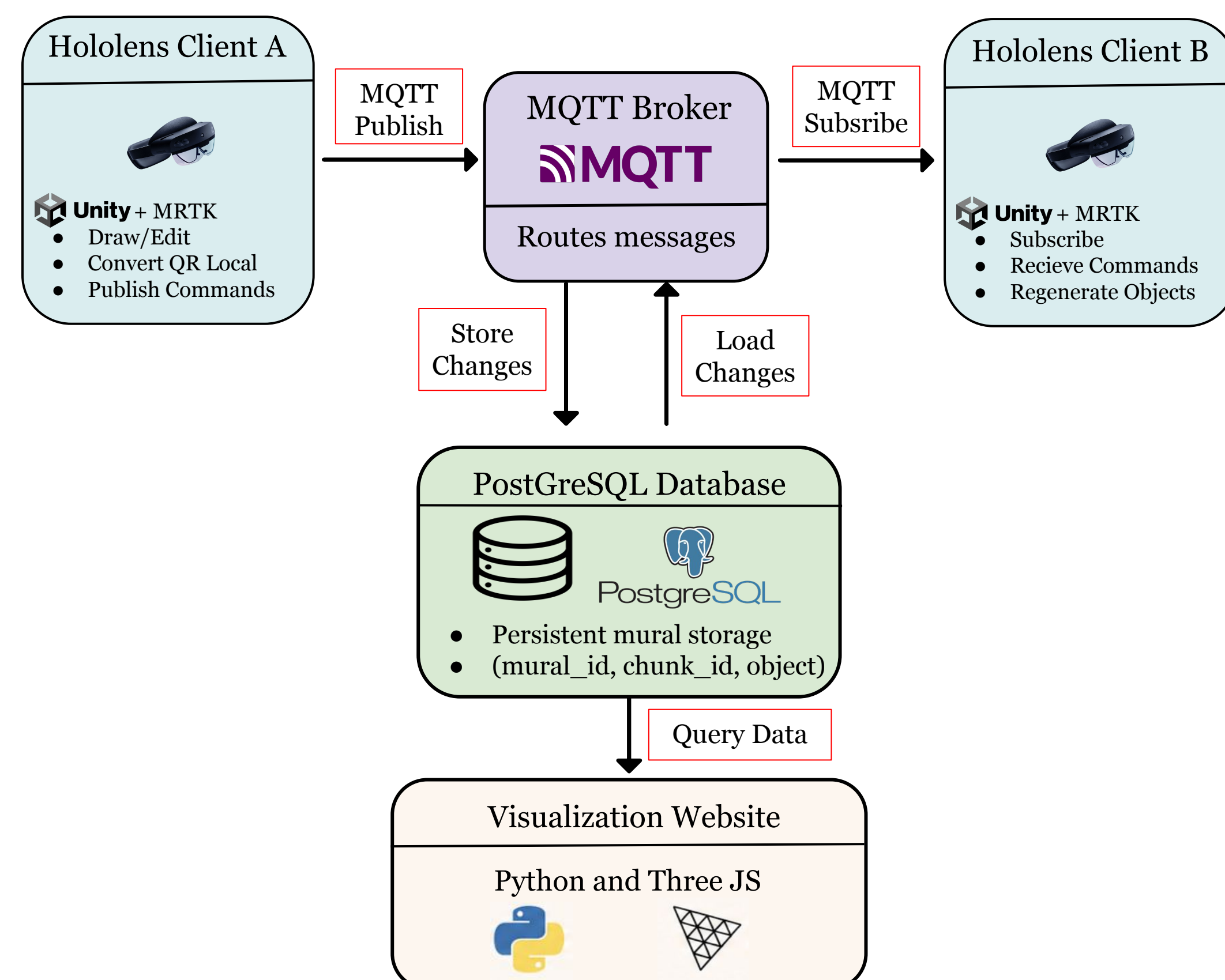


Project Architecture

Publisher-Subscriber Model: Allows for live inter-headset communication via the MQTT broker.

Server/Broker: sends the data to other headsets and stores the data in the PostgreSQL database for persistence.

The scripts are written in **C#** and the Unity development is supported with Microsoft's **Mixed Reality Toolkit 3**.



QR Code Initialization

- 3 QR codes establish a shared 3D coordinate system
- Aligns mural content to the correct real world location
- Keeps avatars and objects synchronized across devices
- Reduces drift and placement inconsistencies
- Enables users in different locations to collaborate in the same augmented reality space



Intended Coordinate Pipeline

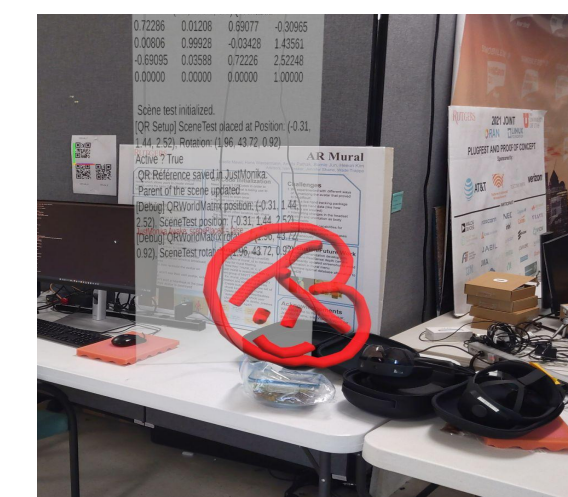
The headset scans the QR codes to create a transformation matrix. Local QR coordinates are converted into shared world-space coordinates. These are then sent to the server, ensuring objects appear in the same location.

Previous Implementation

The QR transformation matrix was offset by 1.6 m. Unity world-space coordinates were sent to the server without any conversion, causing objects to appear in the ceiling and in different locations on different headsets.

Contributions

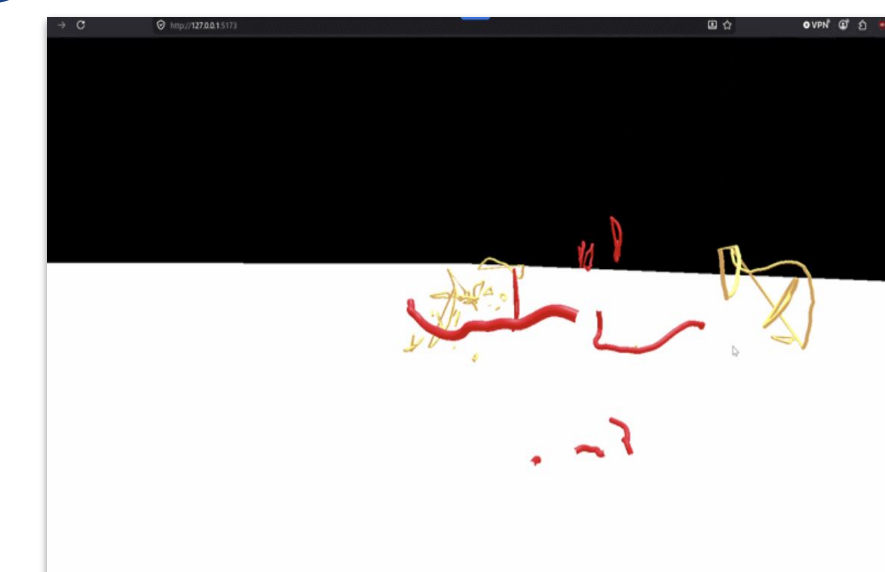
1. Corrected misalignment / QR code initialization
2. Redesigned coordinate transformation pipeline
 - All tools recoded and Avatar positioning
3. Minimized scene switch lag
4. Optimized mural chunk generation
5. Developed error and solution documentation
6. Developed overview of important files [Learn more here!](#)



Debugging Website

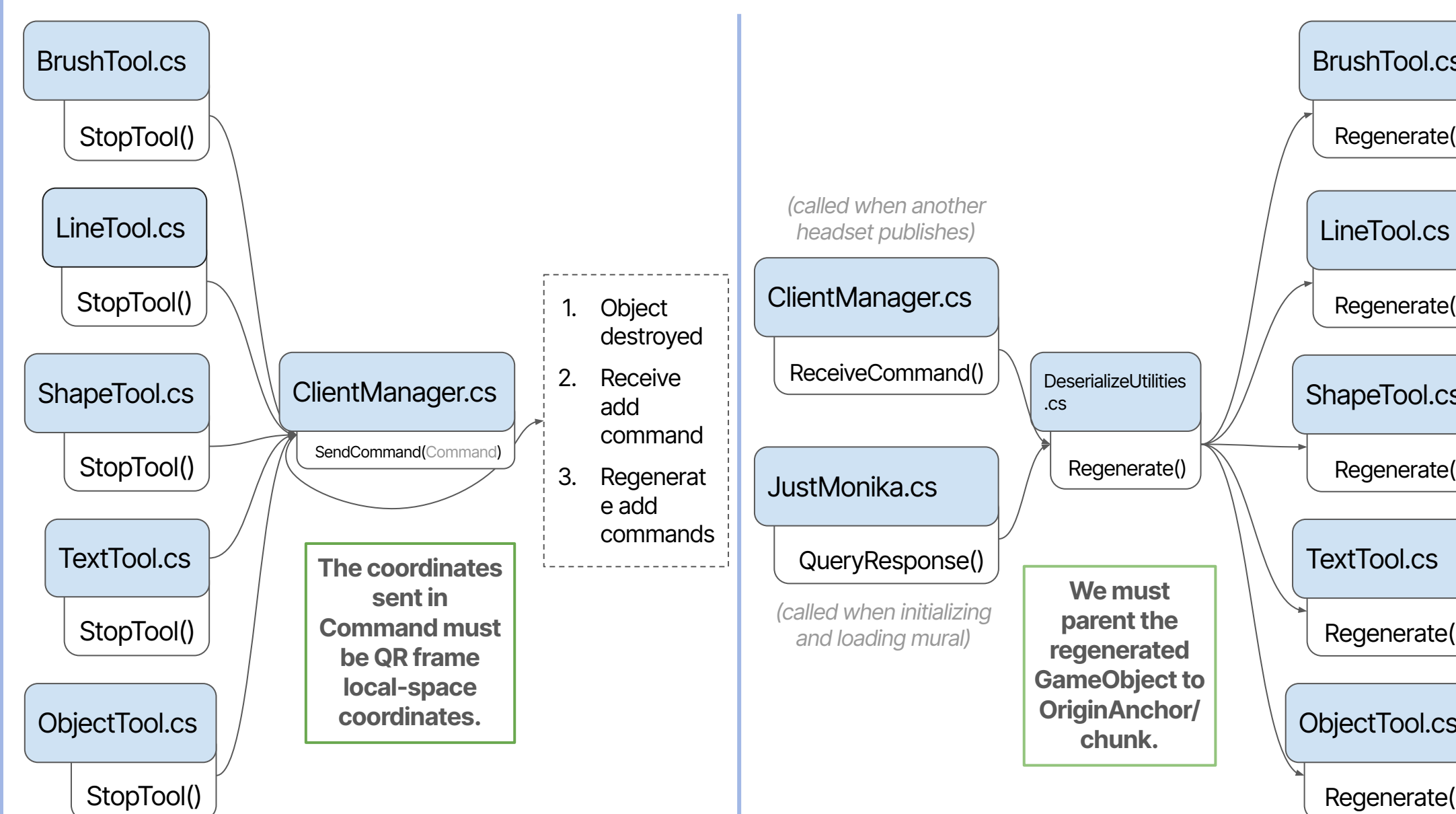
Retrieves data from the **PostgreSQL** database and then displays the 3D objects in a website using **Three.js**

- Allows for faster debugging and visualization of objects

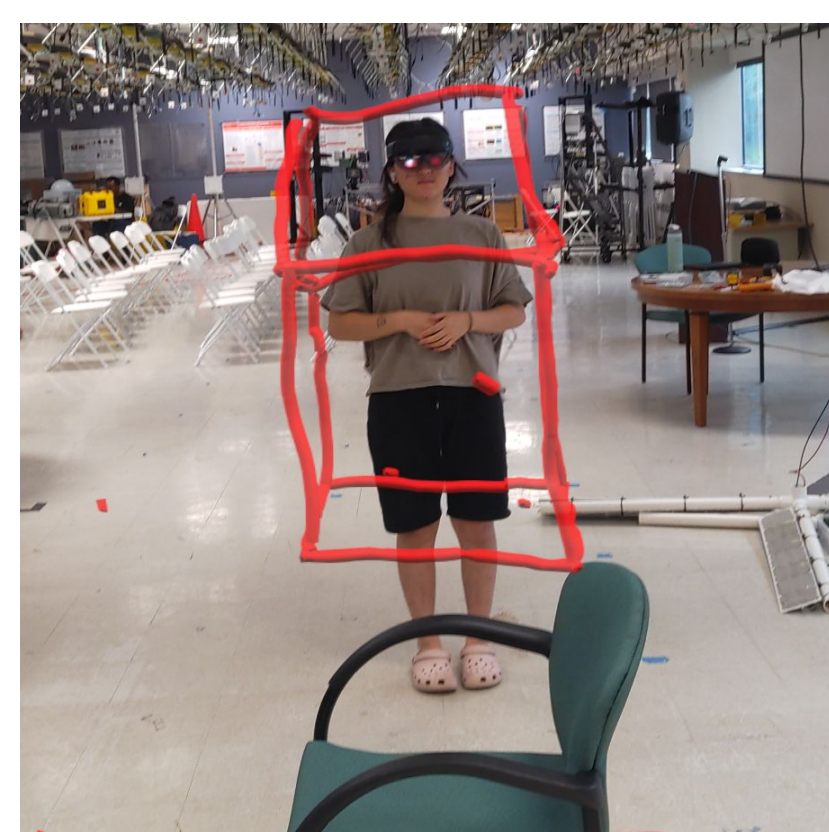
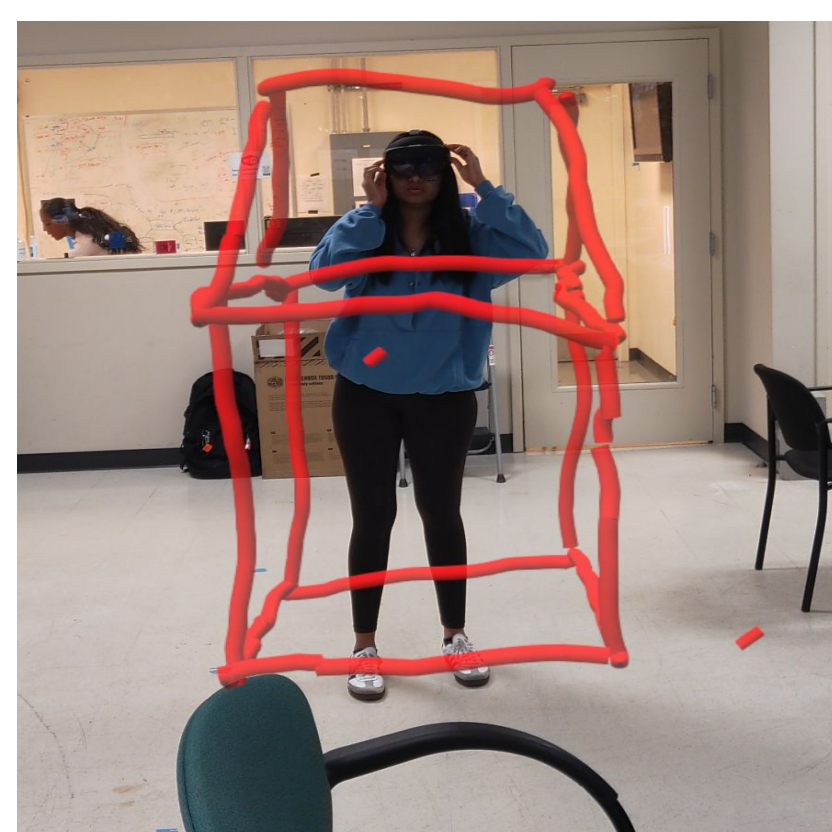


Our Solution

- In every StopTool(), we transform world-space coordinates into QR frame local-space coordinates.
- When objects are regenerated, we set their parent as the QR frame OriginAnchor.
 - Unity automatically transforms them back to world-space.
- Changed offset to zero for MRTK XR Rig (fixed 1.6 m error)



Results



Conclusion/Future Work

- Benchmark speed improvements
- Remove further latency in codebase
- Complete integration with **spatial anchors**
- Allow project to expand beyond room
- Make toolbar features more accessible
- Remove overlap and make them further and larger



Acknowledgements

We would like to thank Ivan Seskar, Jennifer Shane, for their help and guidance throughout this project.