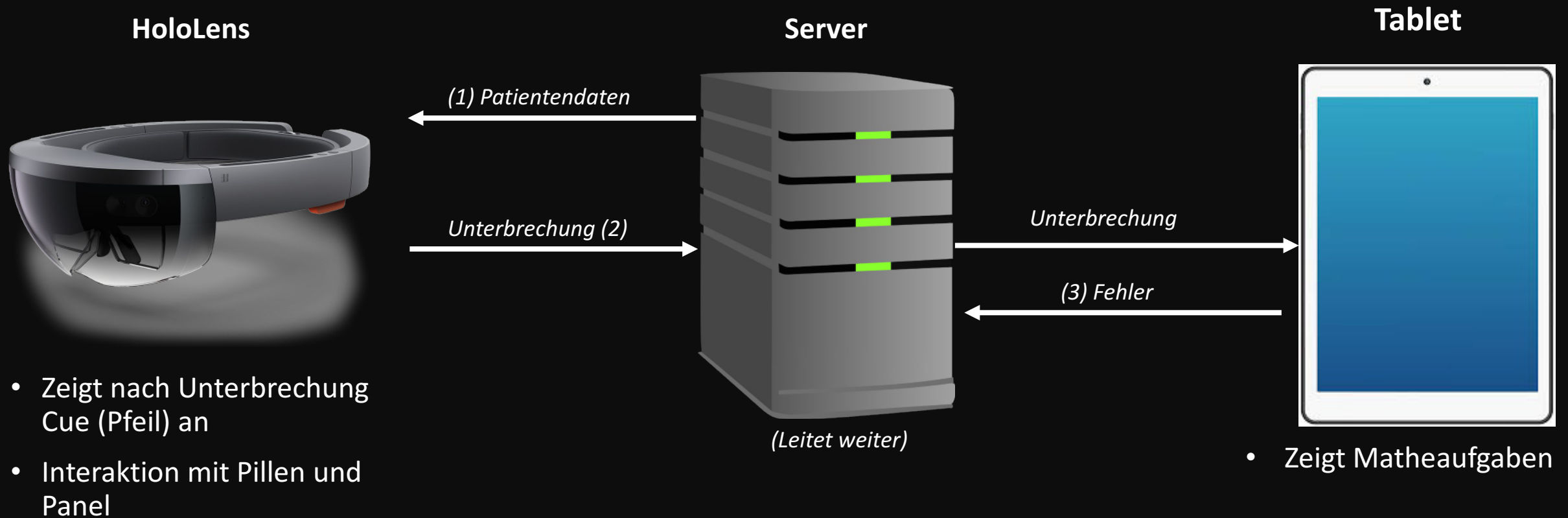




Augmented Reality gegen Unterbrechungen

Einführung in den Programmcode

Versuchsaufbau



Welche Tools?

Server

- Programmiersprache: Kotlin
- IDE: IntelliJ (*Version 2020.3.4*)
- Java 11 um Server.jar file auszuführen

HoloLens

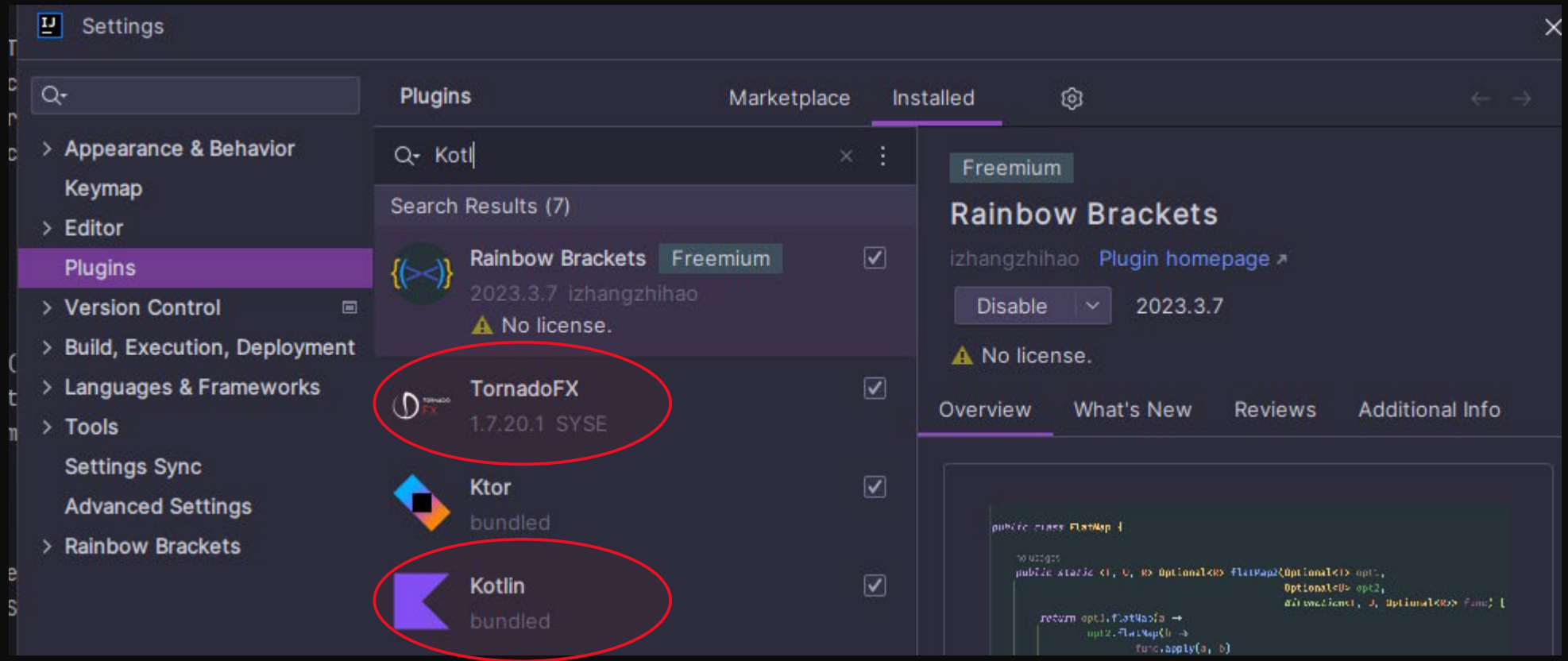
- Unity (**Version 2020.3.14f1!**)
- Programmiersprache C#

• Tablet

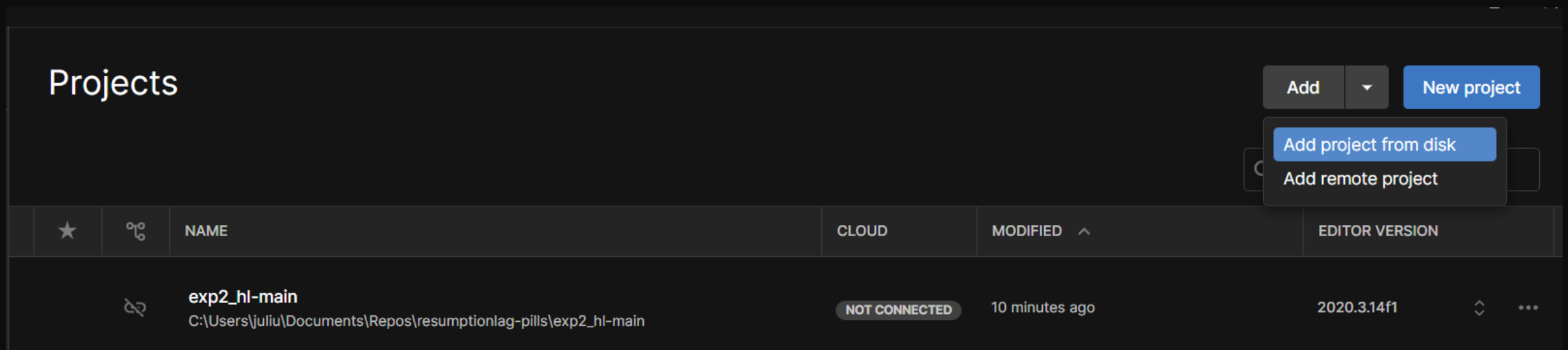
IntelliJ Projekt öffnen

Sollte eigentlich Automatisch passieren wenn Projekt importiert wird – Troubleshooting-guide in Wuecampus falls Fehler!

*Kotlin Plugin +
TornadoFX Plugin
(JavaFX für Kotlin)*



Unity Projekt öffnen



Add project from disk – oben rechts

ACHTUNG: Nicht ganzes Repository öffnen, sondern Unterordner „exp2_hl-main“

Unity Projekt öffnen

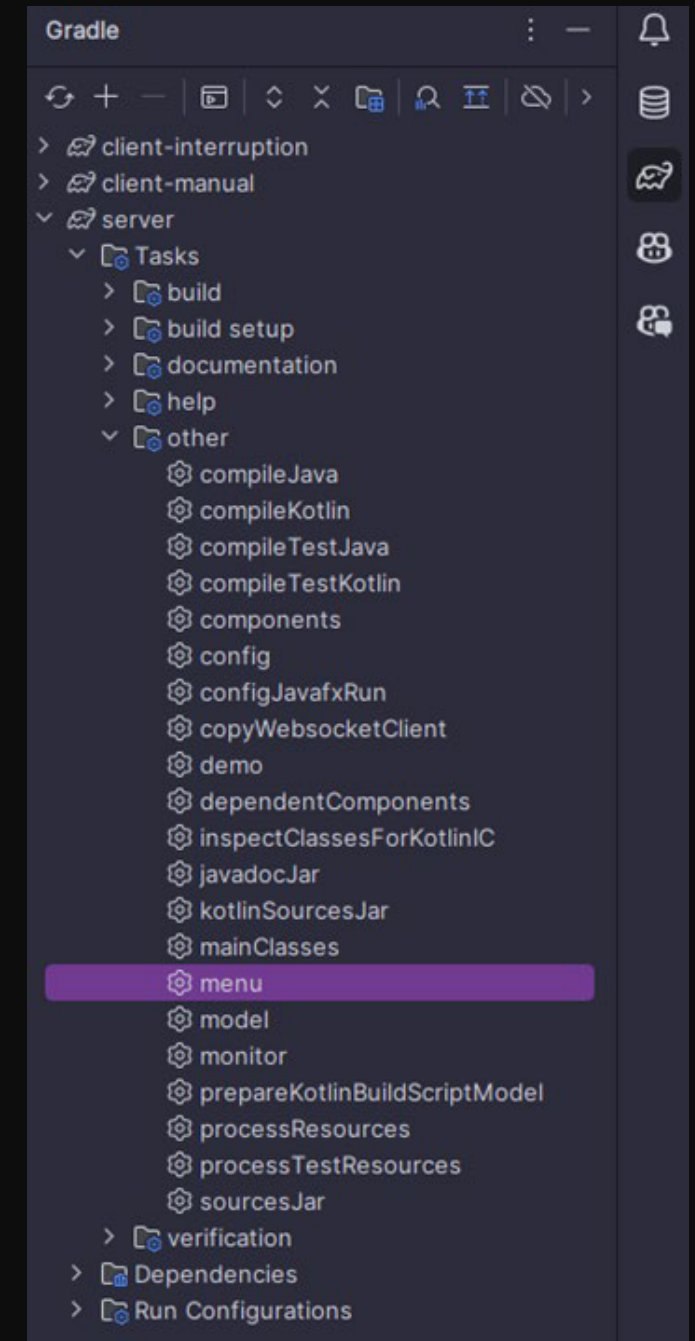
Scenes → SampleScene öffnen



Server setup

Falls Build-Ordner noch nicht existiert, mit Gradle project builden:

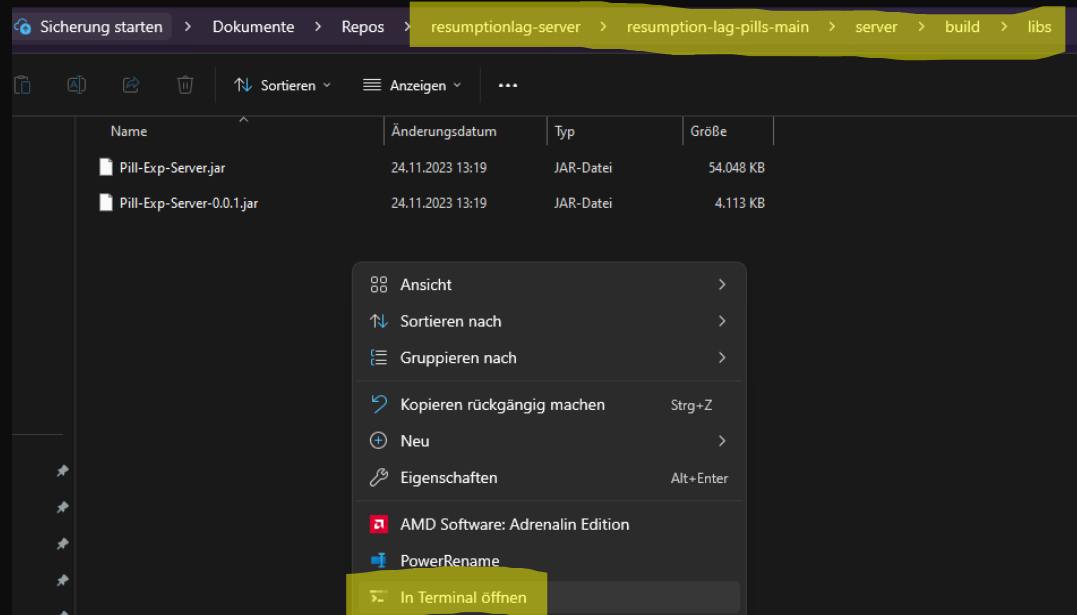
Gradle Menü → Server → Tasks → other → menu



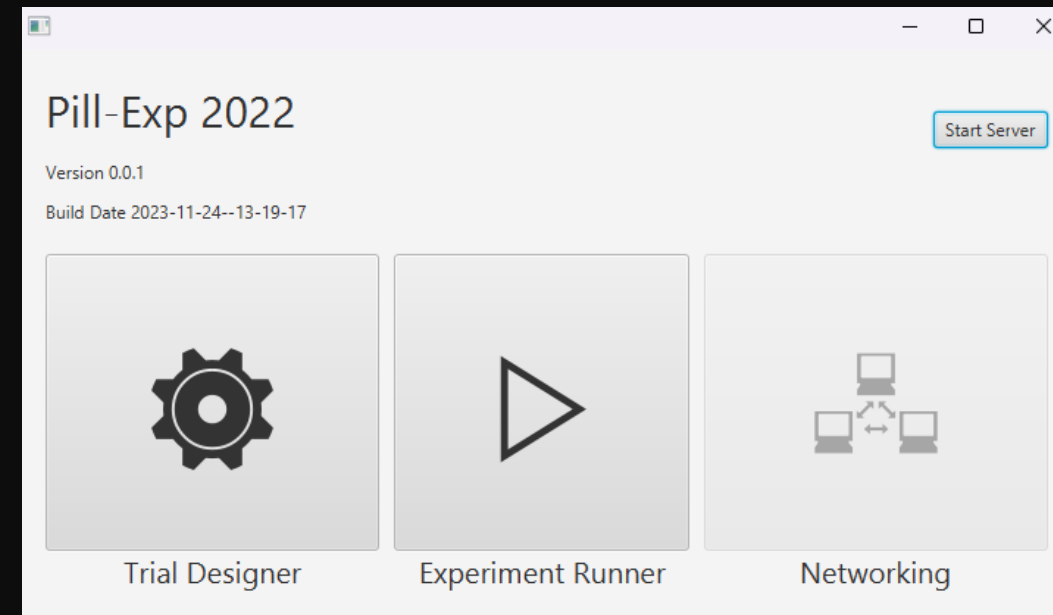
Server setup

1. Starten der Server .jar file

1. (cd) resumptionlag-server\resumption-lag-pills-main\server\build\libs
2. In Kommandozeile öffnen und folgendes eingeben:
3. **java -jar .\Pill-Exp-Server.jar**



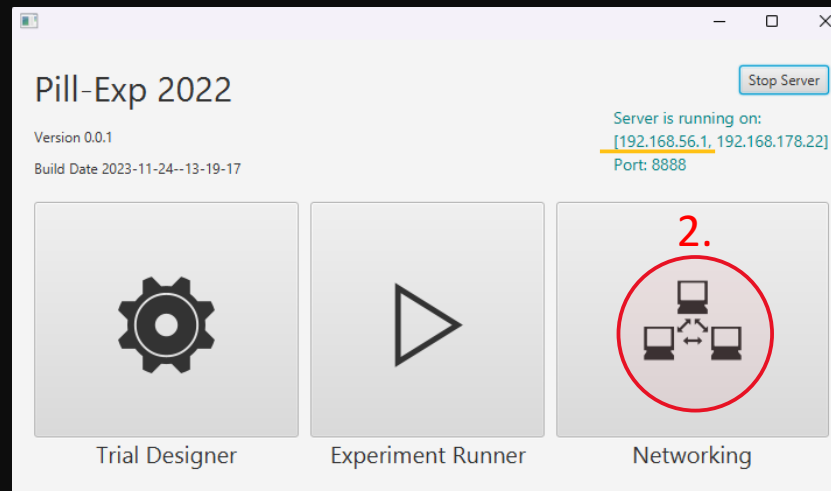
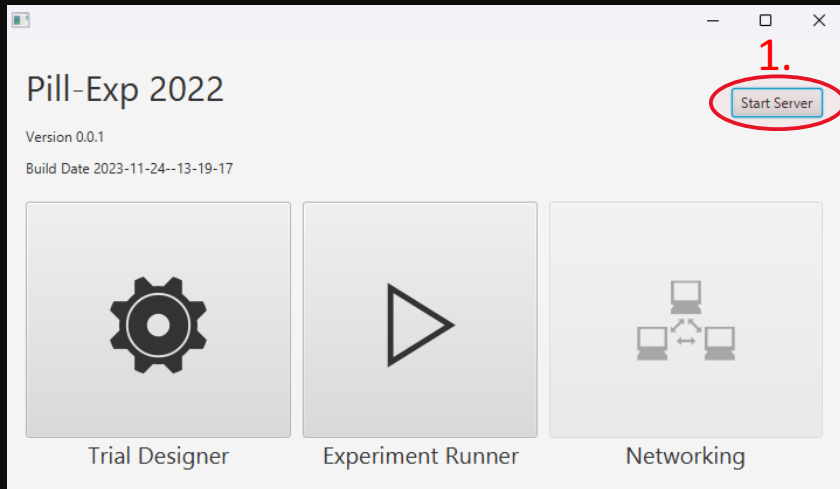
Pfad in Explorer öffnen und „In Terminal öffnen“



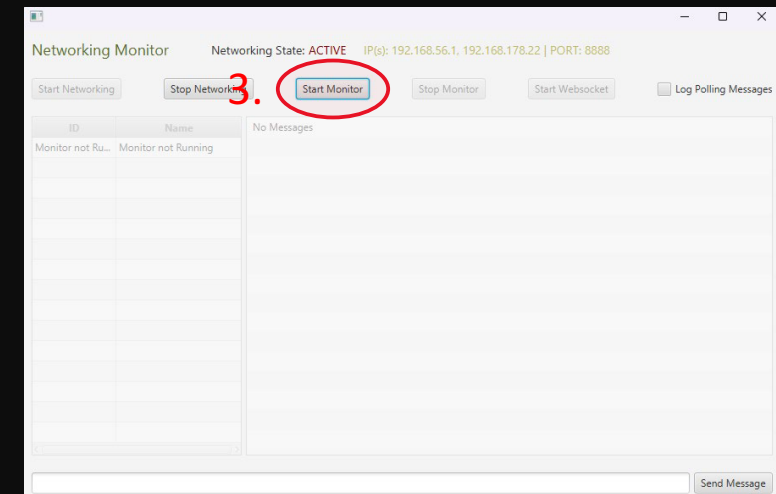
Server Hauptfenster öffnet sich

Server setup

2. Server Starten und Networking beginnen



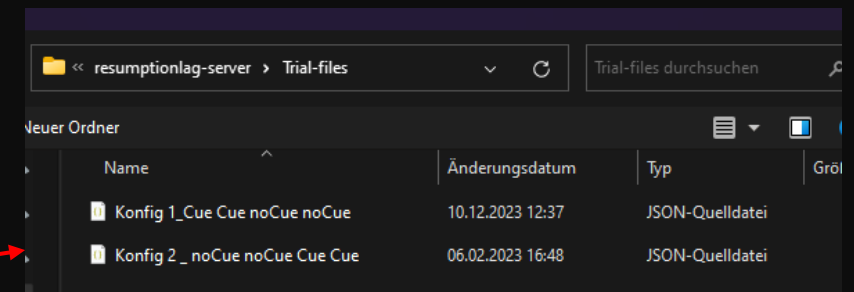
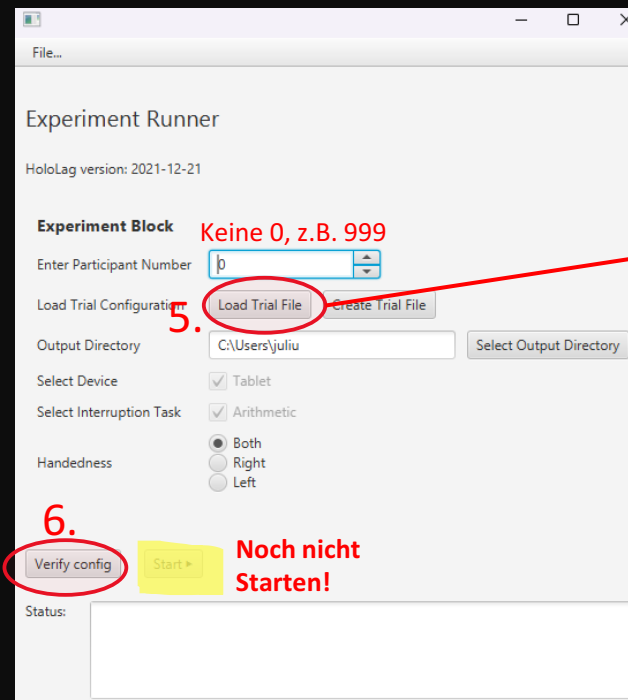
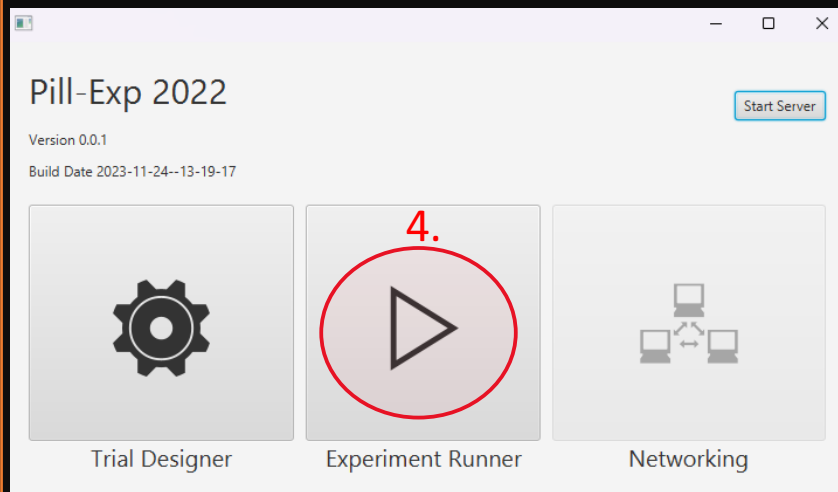
Die IP-Adresse auf welcher der Server startet wird später wichtig!



Neues Fenster öffnet sich

Server setup

3. Experiment Daten in Server laden

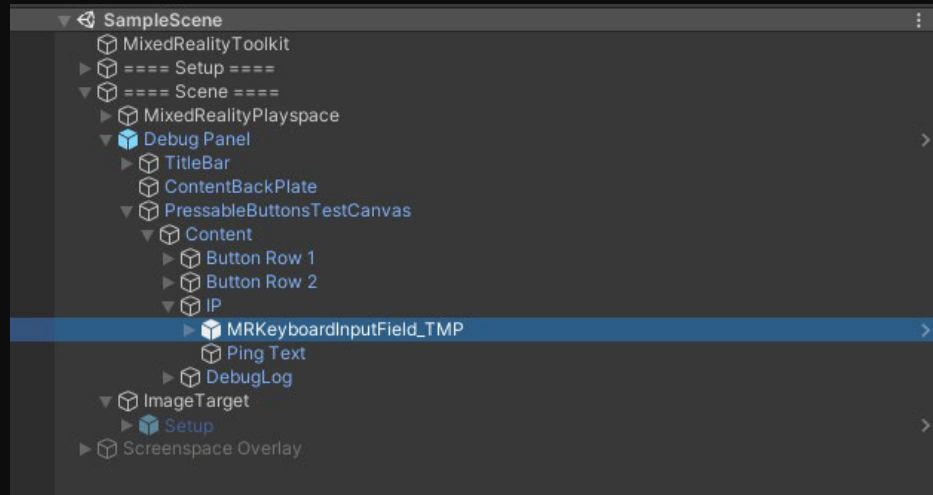


Neues Fenster öffnet sich

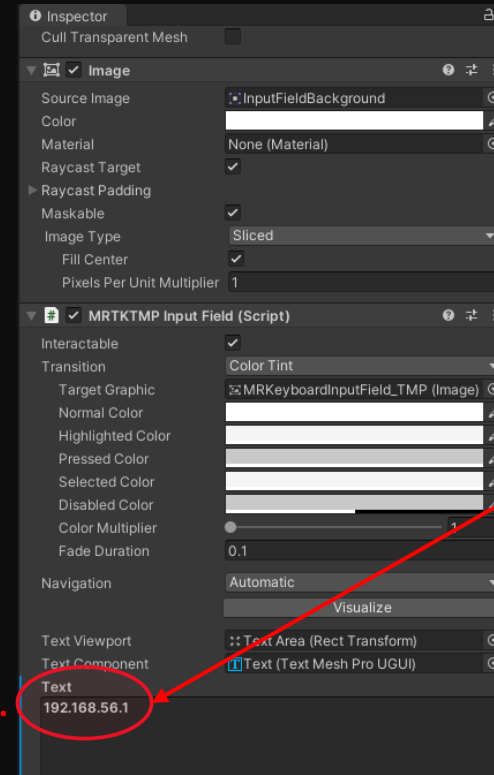
Unity setup

1. IP-Adresse des Servers eingeben

Rechte Seite von Unity (runter Scrollen)



Linke Seite von Unity



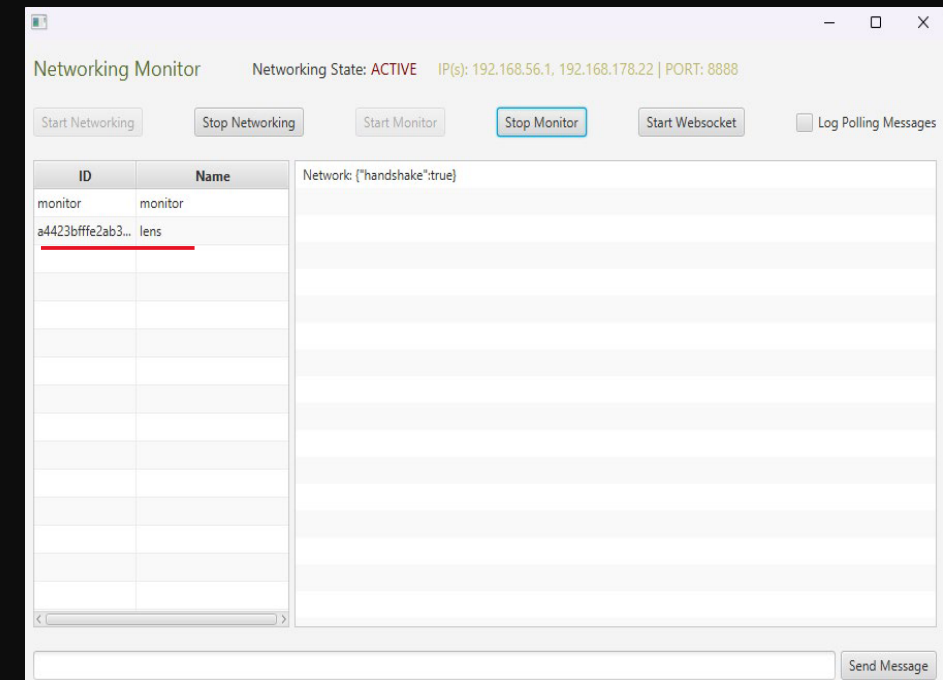
Unity setup

2. Simulation in Unity starten



Es sollte Handshake received im Panel angezeigt werden

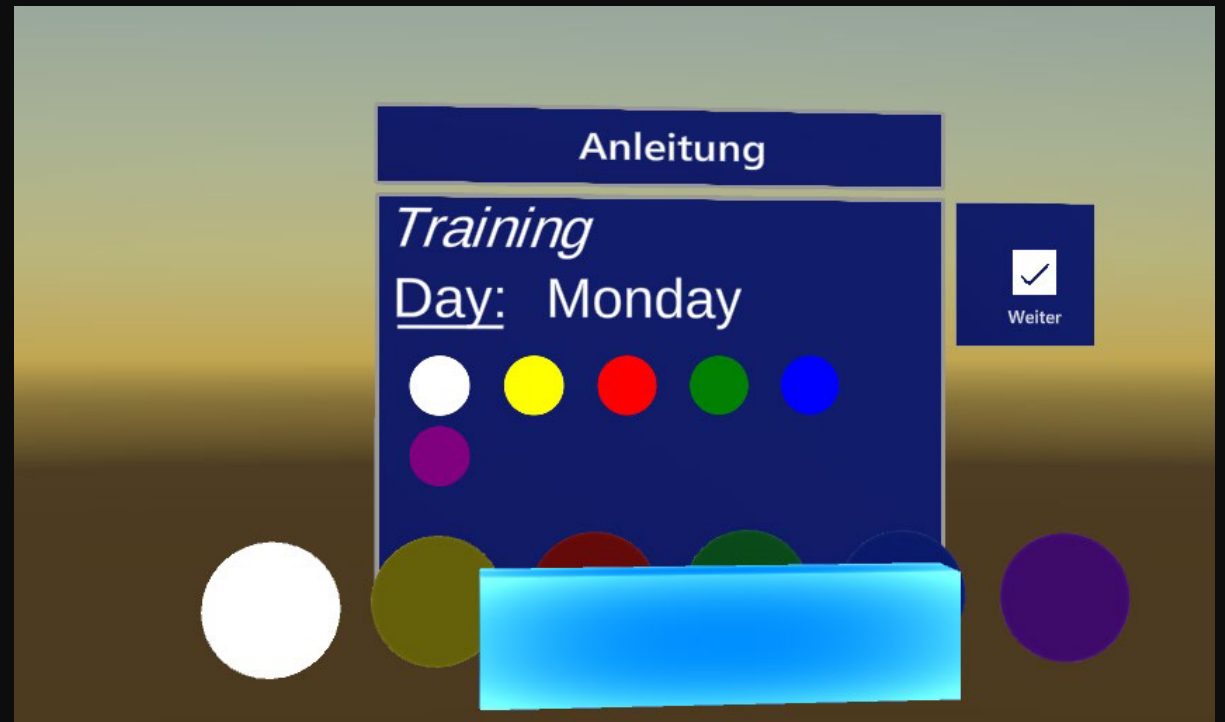
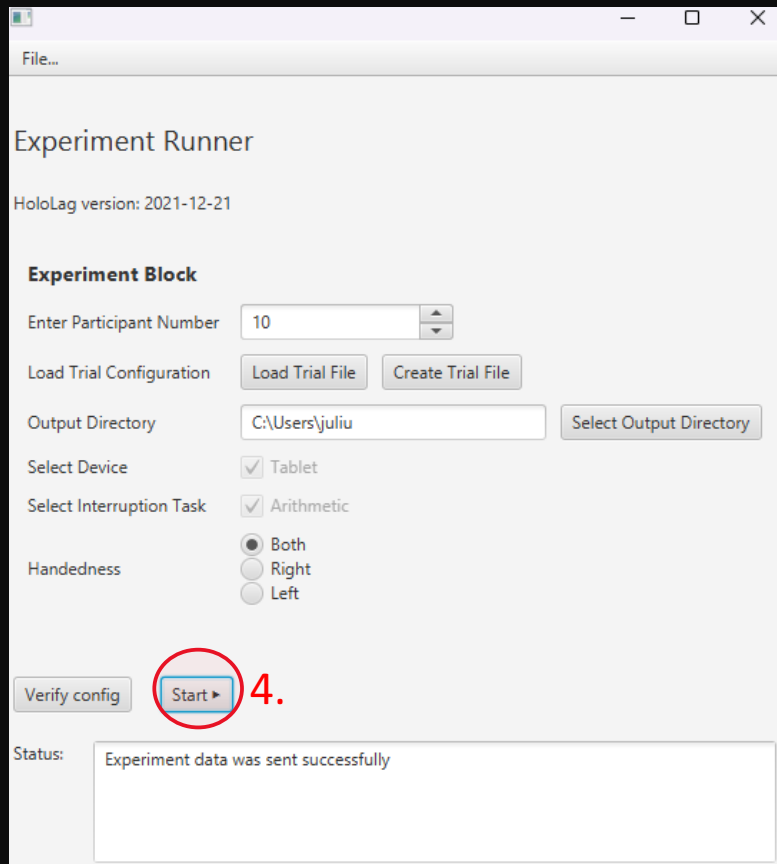
Sicherstellen ob Verbindung zum Server



Beim Server im Fenster „Networking Monitor“ sollte nun auch die HoloLens angezeigt werden

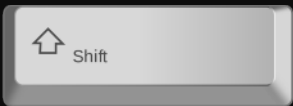
Server setup

3. Daten vom Server an Unity schicken

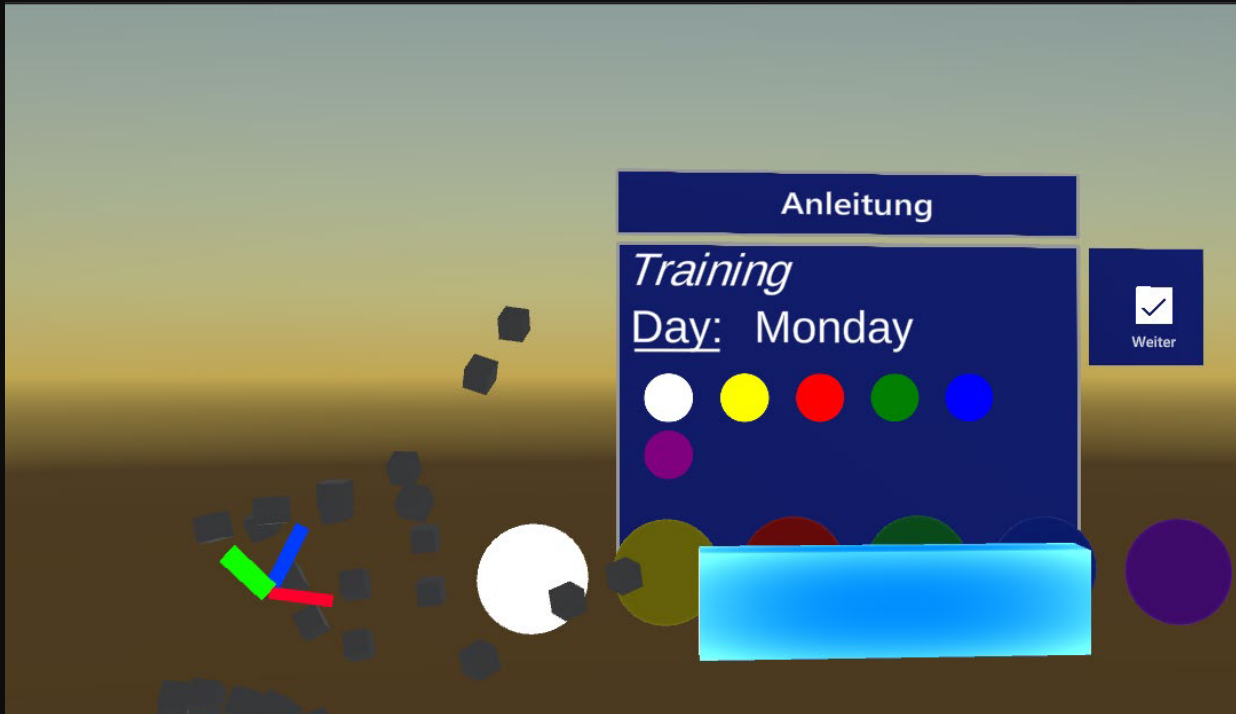


In Unity sollte ein Ton ertönen und der erste Tag sollte geladen sein

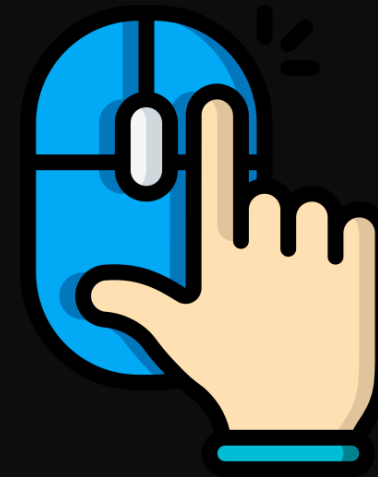
Navigation in Unity



drücken um Virtuelle Hand zu bekommen



Rechts gedrückt
halten: Camera
bewegen



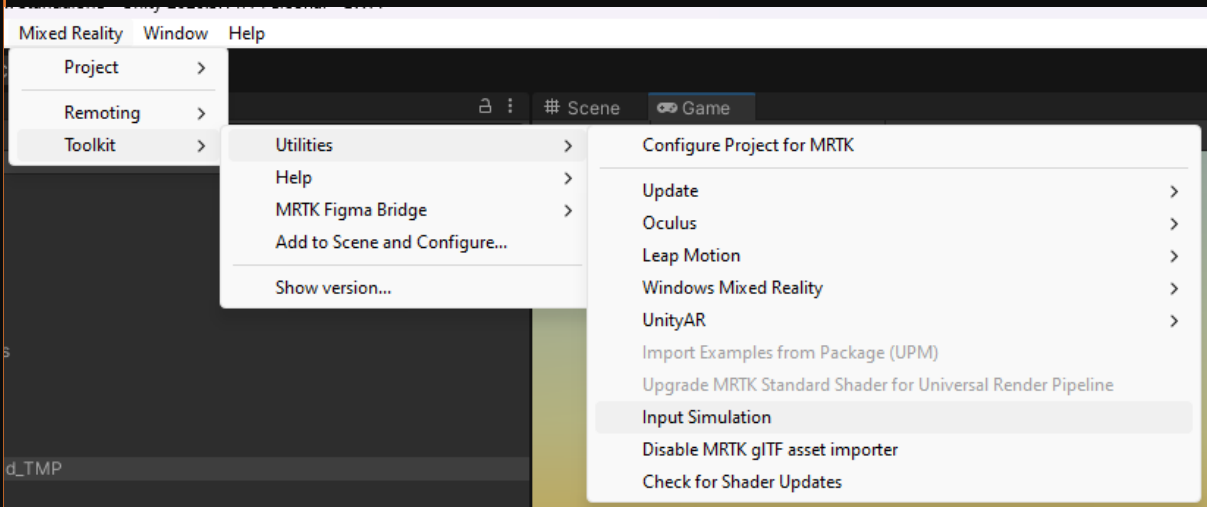
Navigation



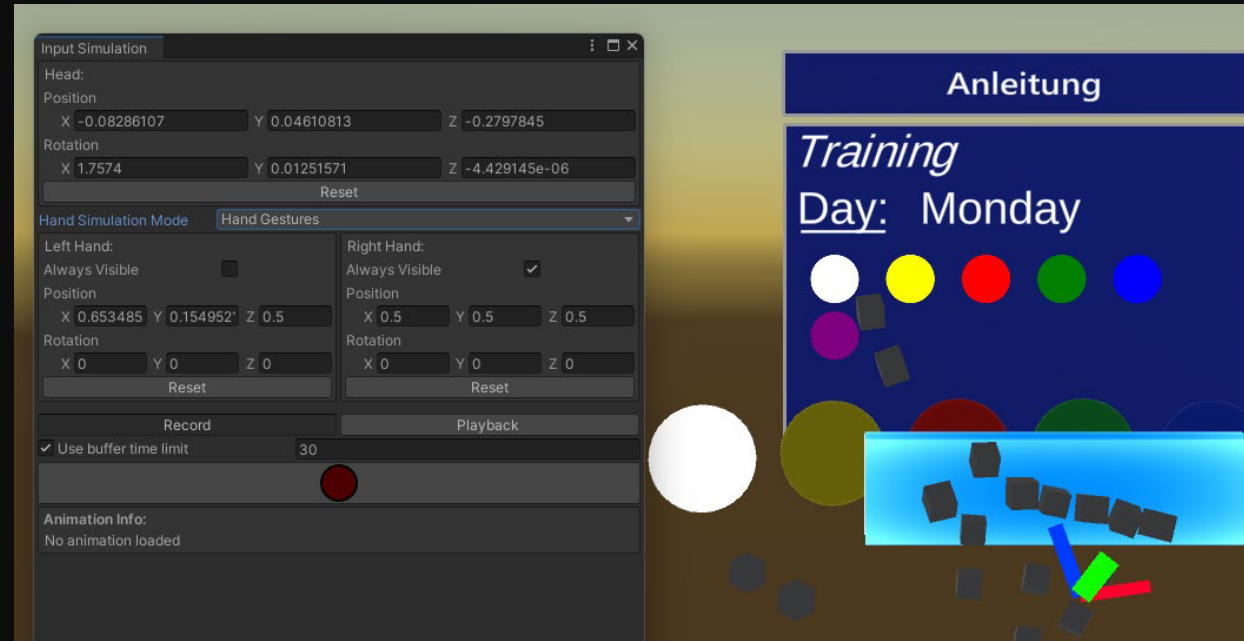
Q: Hoch

E: Runter

Navigation in Unity – Alternative



Alternativ kann im Mixed Reality Tab unter „Input Simulation“ die Hand dauerhaft eingeblendet werden

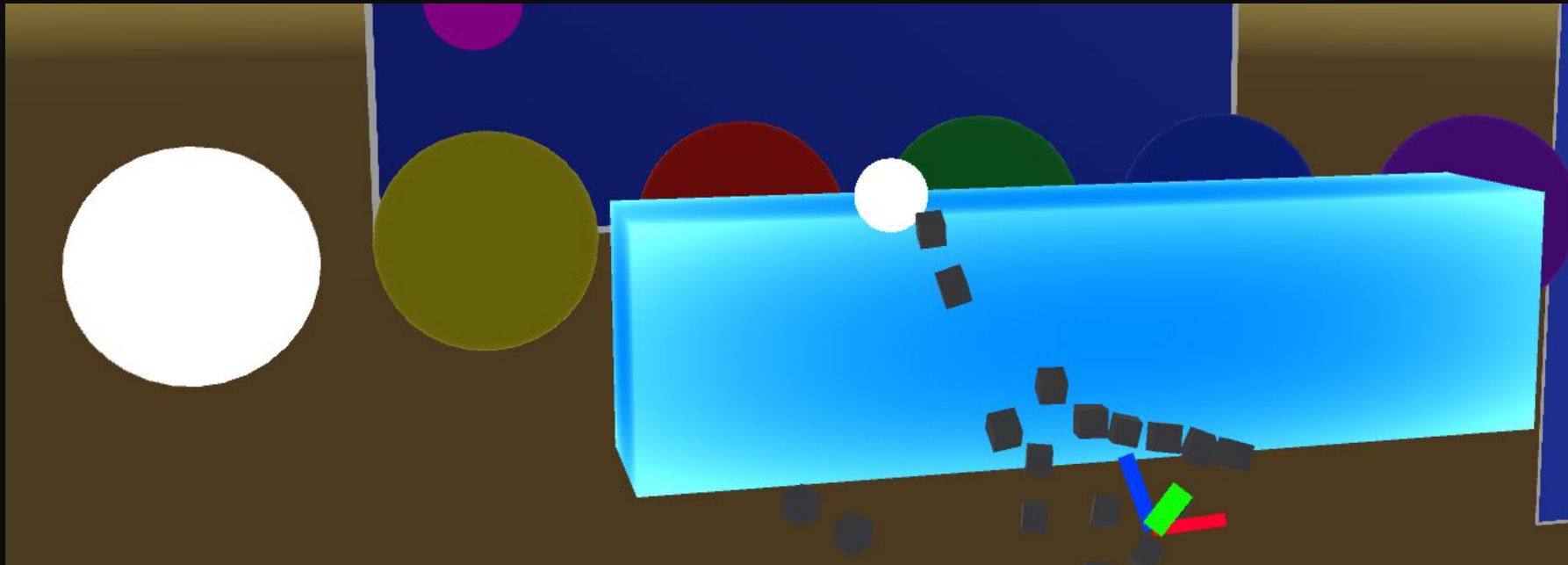


Task

Hand mit



in Kugel (Pillenschale) navigieren (Farbe der Pille wird am Finger angezeigt) und in Blauen Würfel (Pillendose) navigieren



Unity Code

ExperimentController.cs

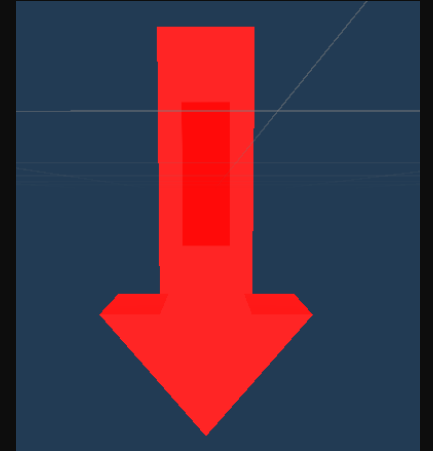


Assets/Scripts

- Cue verschieben
- Unterbrechungen an Server schicken
- Farbe von Kugel an Finger anzeigen

Prefab: Objekt kann immer wieder verwendet werden, muss nicht immer das selbe nehmen!

Pfeilfarbe



Prefabs/Arrow

Unity Code

ExperimentController.cs



Assets/Scripts

Cue verschieben

```
403 void MoveCueToNextTrigger()
404 {
405     if (GetCueType(_currentPatient) == CueType.None)
406     {
407         return;
408     }
409
410     _arrow.transform.position = _setup.GetTriggerPosition(_coordinator.GetExpectedTriggerType()).position;
411     _arrow.transform.position += new Vector3(x:0, y:0.05f, z:0); // move above the center of the sphere
412     _arrow.transform.localScale = new Vector3(x:2.5f, y:2.5f, z:1); // increase its height and width, but not depth
413     _arrow.transform.LookAt(_setup.UserCenter.transform, Vector3.up); //rotate towards the "user"
414
415     _arrow.SetActive(true);
416 }
417
```

Unity Code

ExperimentController.cs



Assets/Scripts

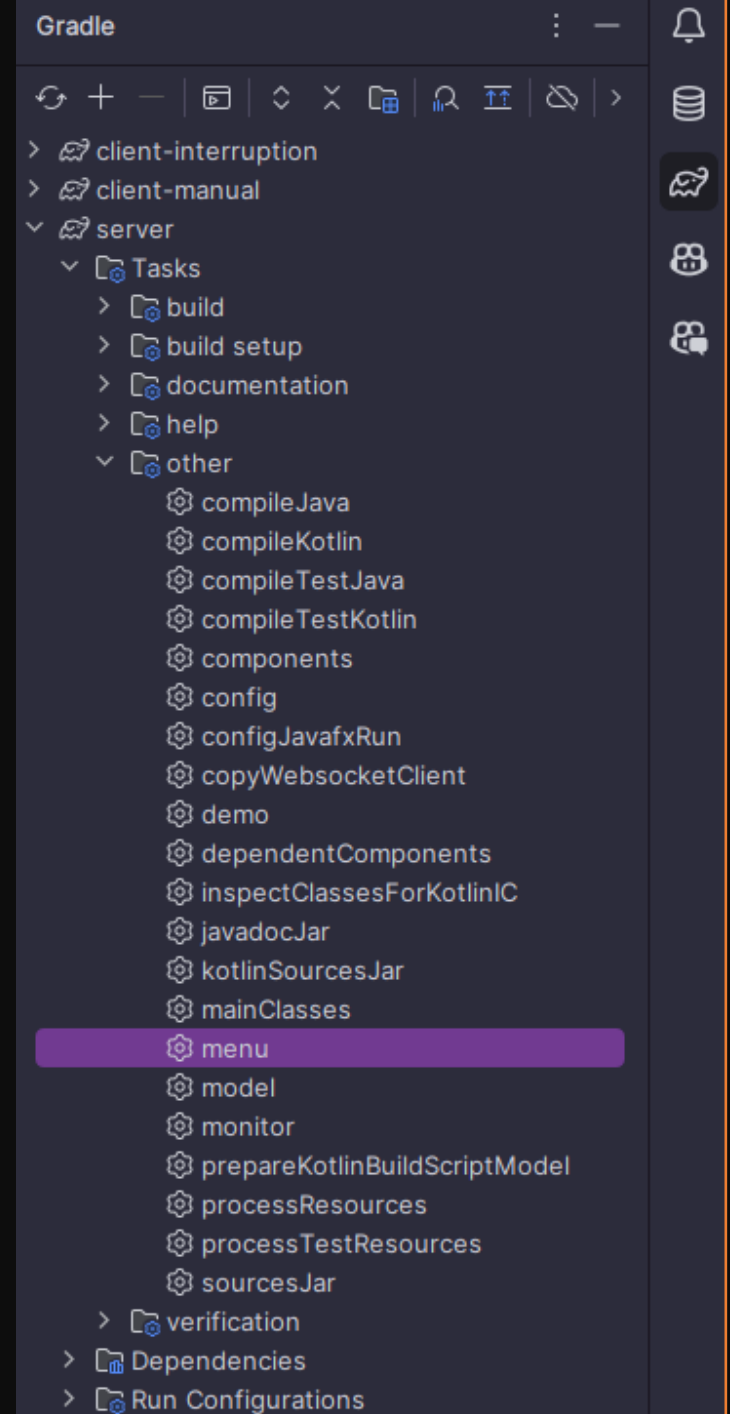
Interruption Senden

```
Frequently called 1 usage Distr0hopper
343 private void TriggerInterruption(TriggerType triggerType)
344 {
345     var intDuration = (triggerType == TriggerType.InterruptionShort) ? shortDuration : longDuration;
346     _wsHandler.SendInterruption(intDuration, interruptionStartDelayInMS);
347     _expLogger.LogEvent($"Interruption Started. Length: {intDuration}s");
348
349     _setup.ManualPanel.Flash(intDuration);
350     StartCoroutine(InterruptionCoroutine(intDuration));
351     _audioSource.PlayOneShot(_audioClips[0]);
352
353     Debug.Log(message: "Interruption Triggered.");
354 }
```

Server Code

**Wenn eine Änderung im Server code gemacht wird,
muss Server re-gebaut werden!**

Gradle Menü → Server → Tasks → other → menu



Server Code



client-interruption\app\src\main\kotlin\cerg\mnv\view\ArithmeticActivity.kt

```
112         // Shows a new equation every 5 seconds
113         Timer().schedule(300) {
114             showFlash(false)
115             setTextVisible(true)
116
117             timer.scheduleAtFixedRate(
118                 object : TimerTask() {
119                     override fun run() {
120
121                         // before every new equation, evaluate whether the input of the user was correct
122                         evaluateUserInput()
123                         showEquation(equationList[equationIndex % equationList.count()])
124                         equationIndex++
125
126                         wasEquationAnswered = false
127
128                     }
129                 }, 10, 5000)
130
131     }
132
```

Code wird aufgerufen,
wenn Server nachricht an
Client (Tablet) schickt

(Code bauen und im
Simulator auf IntelliJ
runnen)