

Open Source Gamified Remote Labs in Photonics Education

Johannes Kretzschmar, Clara Henkel, Christian Helgert & Thomas Pertsch

Friedrich Schiller University Jena, GERMANY

Abbe School of Photonics Jena, GERMANY

johannes.kretzschmar@uni-jena.de



FRIEDRICH-SCHILLER-
UNIVERSITÄT
JENA



**Institute of
Applied Physics**

Friedrich-Schiller-Universität Jena



**Abbe School
of Photonics**

Friedrich-Schiller-Universität

JENA



LICHTWERKSTATT JENA

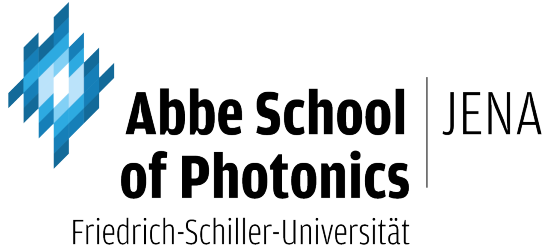
OPEN PHOTONICS MAKERSPACE

NPSE24 New Perspectives in Science Education
14th-15th March 2024, Florence, Italy

REMOTE STUDYING

at Abbe School of Photonics

- M.Sc. Photonics Program
- 80 Students p.a.
- 80% international
- Remote Access



Challenge:

Experimental Lab Courses

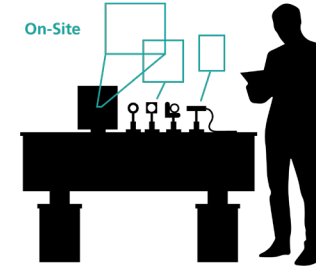
(Optics & Quantum Technologies)

Conventionell Remote Access



Individual Preparation & Performing of Remote Experiment

Augmented Reality Access

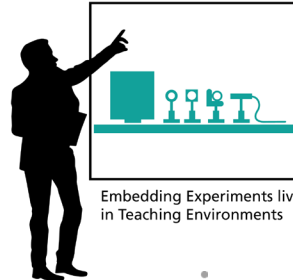


Perform Experiment with Augmentation of Educational Content

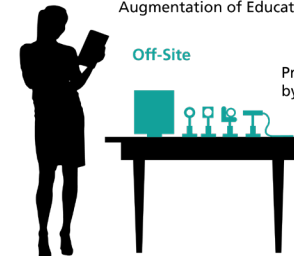
Virtual Reality Access



Prepare and Perform Experiment by means of Virtual Model (Digital Twin)



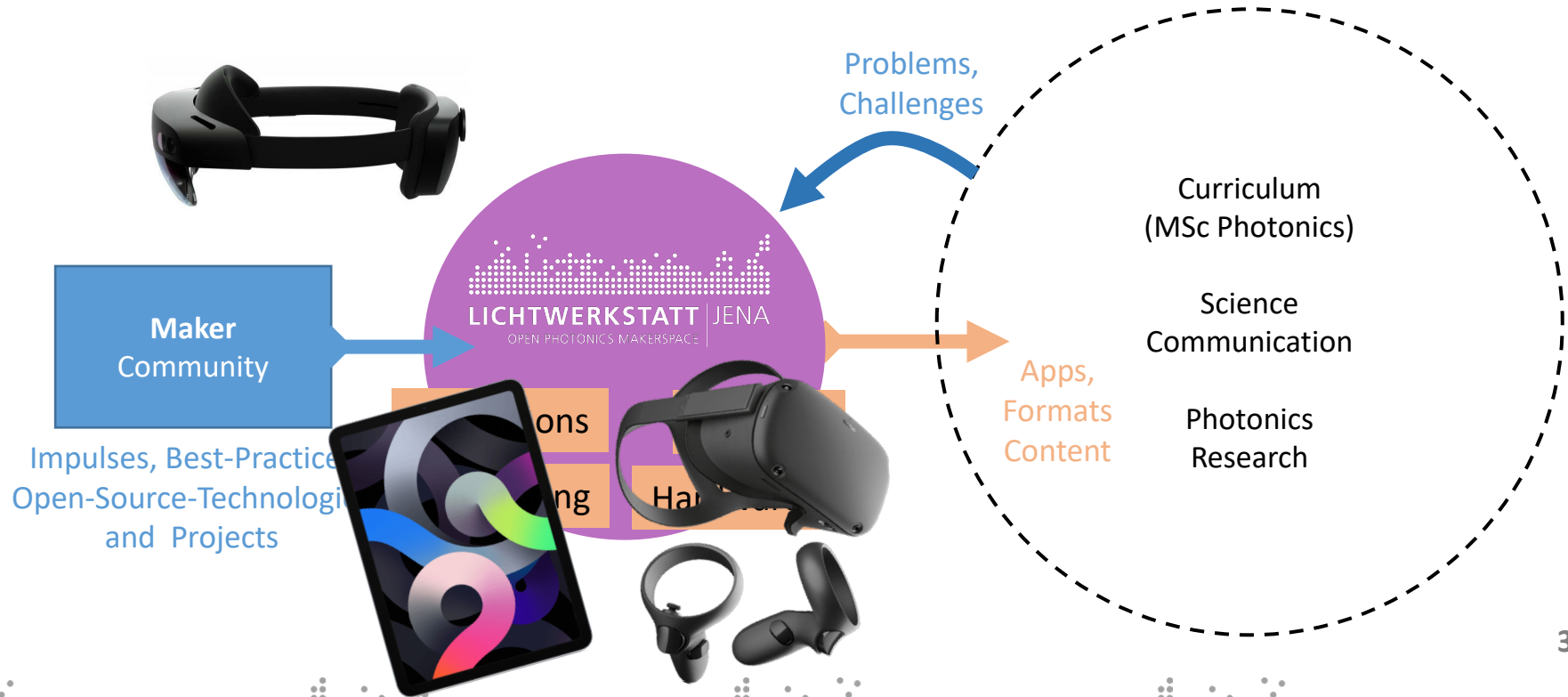
Embedding Experiments live in Teaching Environments



XR TwinLab Application

COMMUNITY DRIVEN DEVELOPMENT

Within the Lichtwerkstatt Makerspace



REQUIREMENTS

Experiments

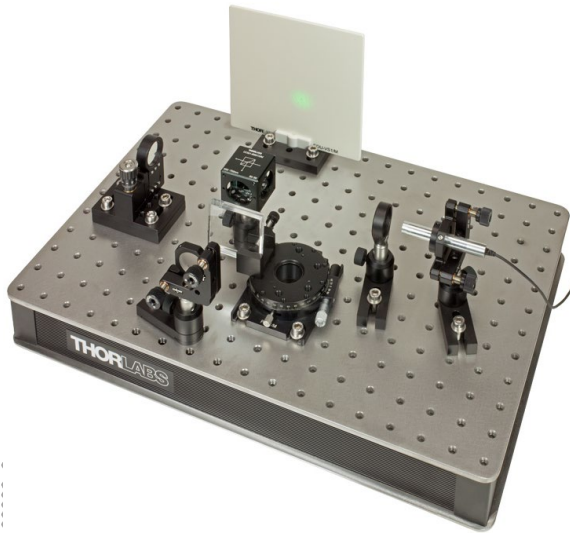
- Based on existing setups
- Standard Components (Thorlabs)
- Interface to whole setups or single components

Virtualization

- Open Source Technology
- Platform-independent
- Low-threshold
- Uniform Codebase
- Ad-hoc Networking (IoT)
- Modular & Reusable

Learning

- Collaborative Group Work
- Visualization
- Immersion
- Communication



APPROACH

Experiments

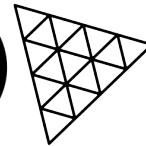
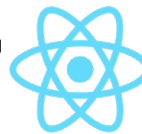
- 3D Printing
- ESP32 Controller
- ESP32 Cam
- PCB

Virtualization

- JavaScript
 - nodeJS
 - reactJS
- Socket.IO
- WebRTC
- Backend Database
- GitHub

Learning

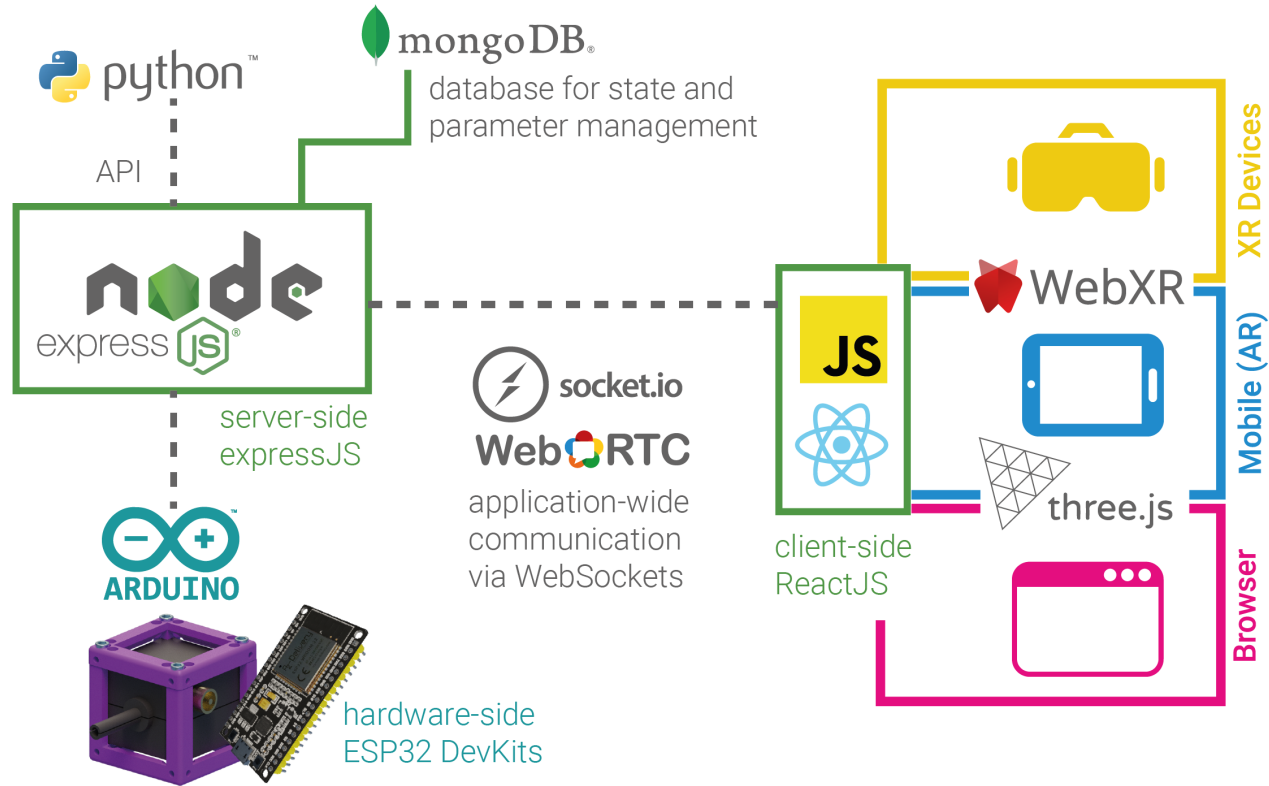
- WebCam
- WebGL Rendering
 - Three.JS
 - reactFiber
- WebXR
- Educational Content
- Gamification
- LEDs Status Display



WebXR

ARCHITECTURE

XR TwinLab



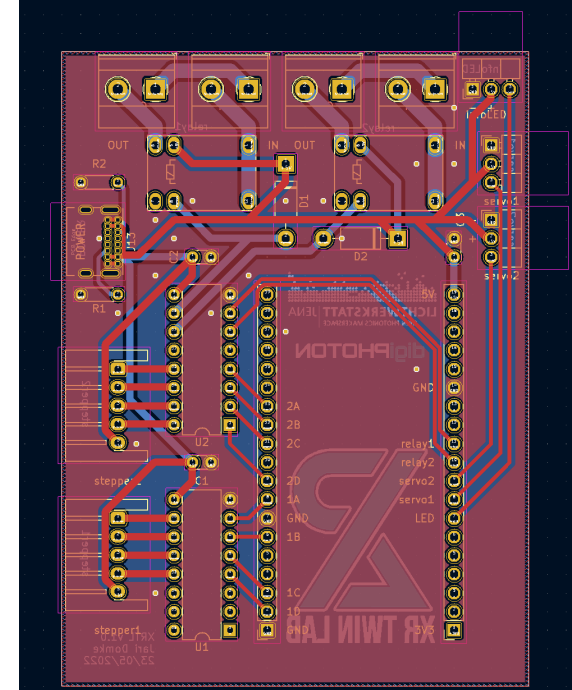
HARDWARE

ESP32 / PCB

1.418,32 €



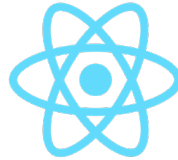
45,00 €



SOFTWARE

reactJS Single Page Application

```
const App = () => {
  process.title = 'XRTLApp';
  return (
    <AppContextProvider>
      <SocketContextProvider>
        <VirtualLayer />
        <PopUpContextProvider>
          <ExperimentUILayer />
          <Console />
          <Chat />
          <NavBar />
          <Login />
        </PopUpContextProvider>
      </SocketContextProvider>
    </AppContextProvider>
  );
};
```



reactCtx



UI Control Elements

Control Components

Components

Experiment Components

UI Elements

SPA UI Components

Background

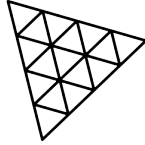
Illustration

Video Feed

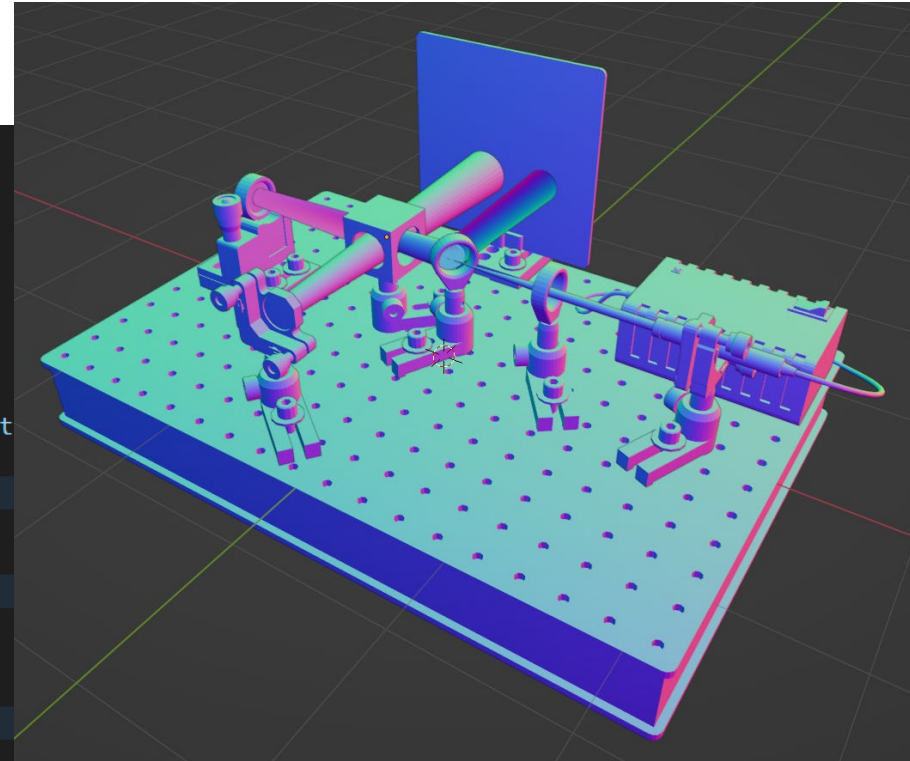
OpenGL Rendering

SOFTWARE

React Three Fiber

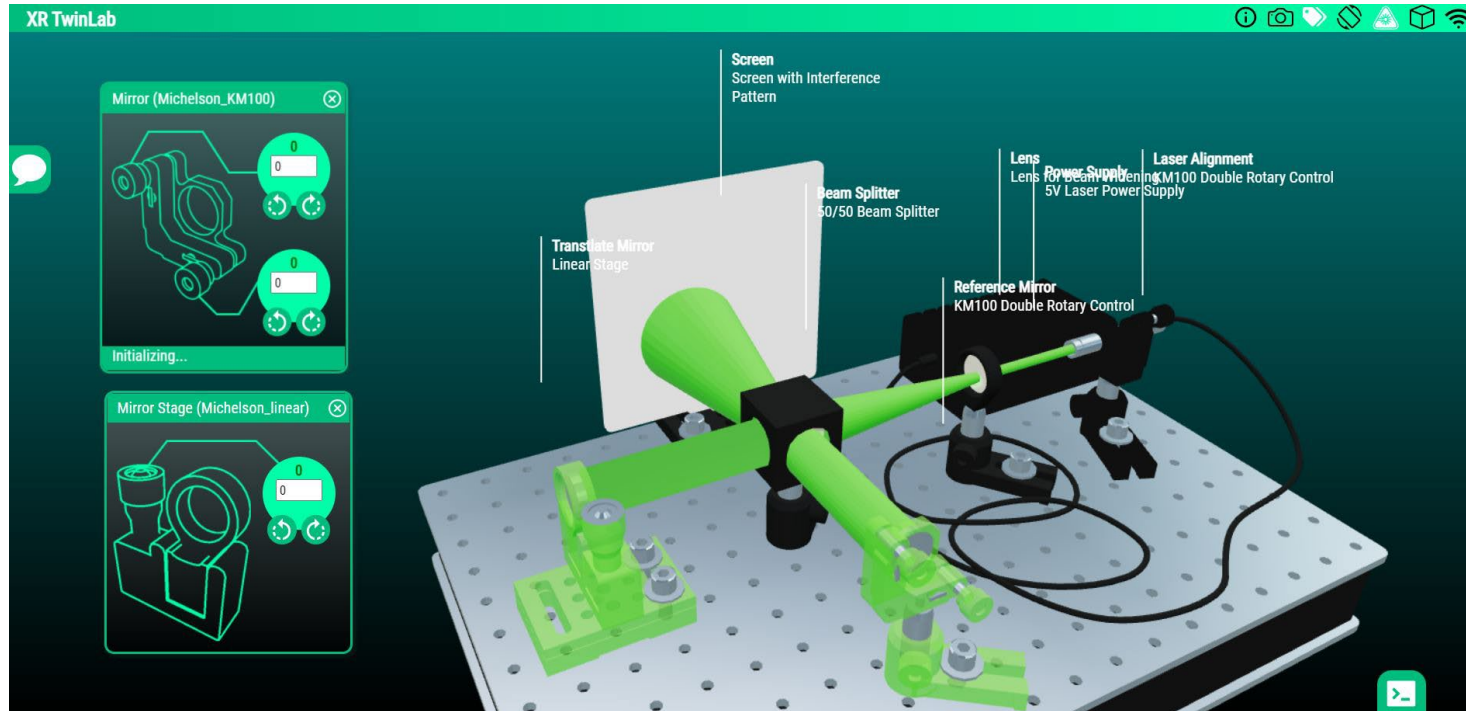


```
</group>
<group
  name="Mirror1"
  position=[[-0.53, 0.67, 0.94]]
  onPointerDown={(e) => {
    e.stopPropagation();
    props.toggleSelect("Michelson_KM100");
  }}
>
  {props.showTags && <DescriptiveTag position=[[0, 1.3, 0]] tit
  Rotary Control" />}
  <mesh ...
  </mesh>
  <mesh
    name="SideMirrorMesh_1" ...
    material={materials.ShinyParts}
  />
  <mesh
    name="SideMirrorMesh_2" ...
    material={materials.Mirror}
  />
```



SOFTWARE

Single Page Application





Socket.IO

```
//Returns the status of a experiment component
socket.on('status', payload => {
  var today = new Date();
  var time = today.getHours() + ":" + today.getMi

  if (componentList.includes(socket.id) === false)
    componentList.push(socket.id, time, payload)
  }
  else if (componentList.includes(socket.id) ===

  } else {
    componentList = [socket.id, time, payload.c
  }
  socket.to(GUIId).emit("newLog", "New Status" +
  socket.emit("newComponent", componentList);

  socket.broadcast.emit('status', payload);
  console.log("New status: ", payload)
});
```

- Rotary Control (stepper)

```
{
  "userId" : "user1234", <- Id of user, who initiated command
  "componentId": "km100_1", <- Component Identifier
  "command" : {
    "controlId": "top", <- Control Element of Component (top/bottom for KM100)
    "val" : 100 <- number of steps to turn
  }
}
```

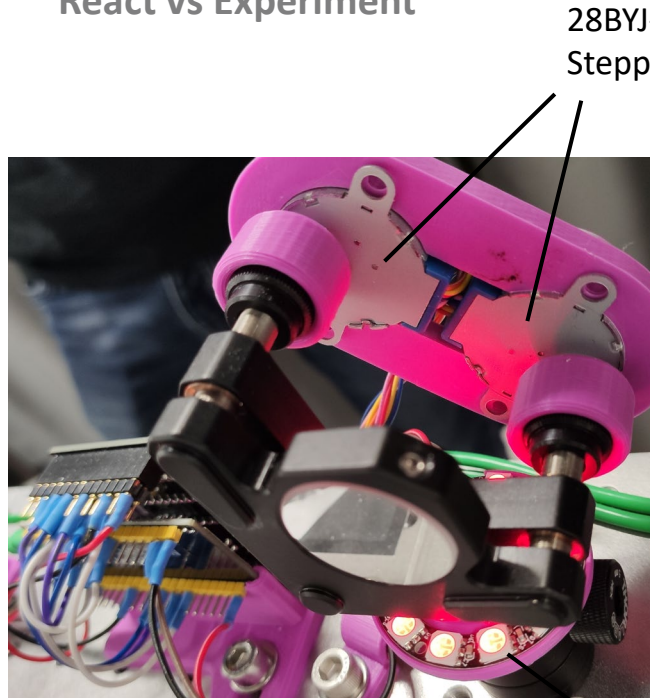
```
{
  "userId" : "user1234",
  "componentId": "KM100_1",
  "command" : "stop" <- aborts all rotations in progress
}
```

- Rotary control (servo)

```
{
  "userId" : "user1234",
  "componentId" : "ESP32Cam_1",
  "command" : {
    "controlId" : "tilt",
    "val" : -30 <- target value of servo
  }
}
```

LOOSE COUPLING

React vs Experiment



Experiment KM100

28BYJ-48
Stepper

Status LED

```
{
  "userId": "user1234", <- Id of use
  "componentId": "km100_1", <- Compon
  "command": {
    "controlId": "top", <- Control El
    "val": 100 <- number of steps to
  }
}
```

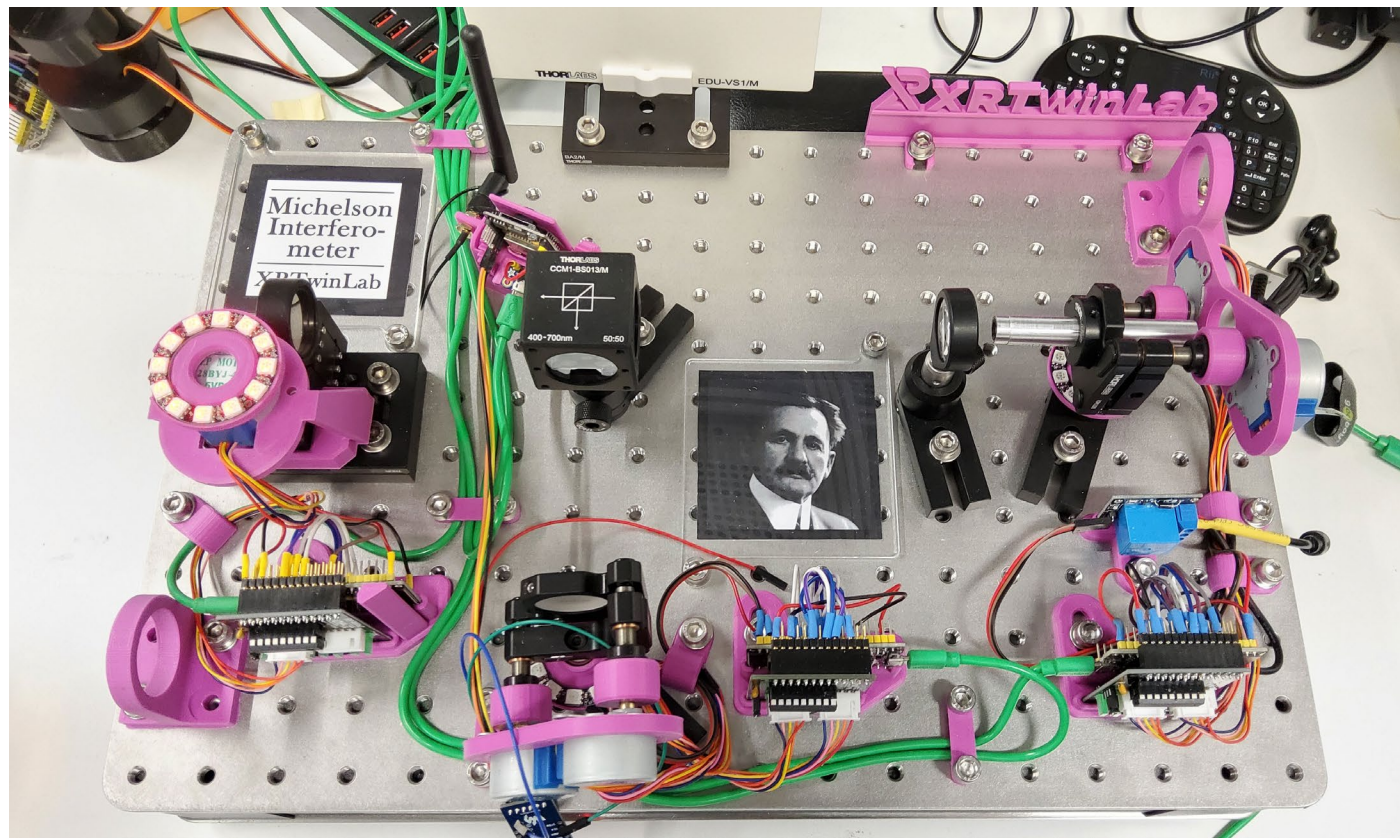
React.Window

React.RotaryCtrl



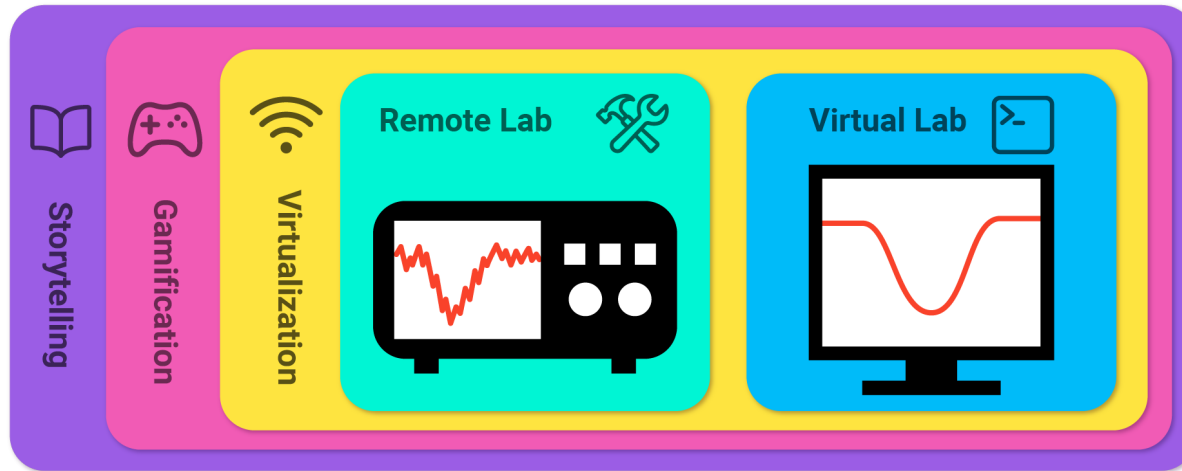
React.KM100

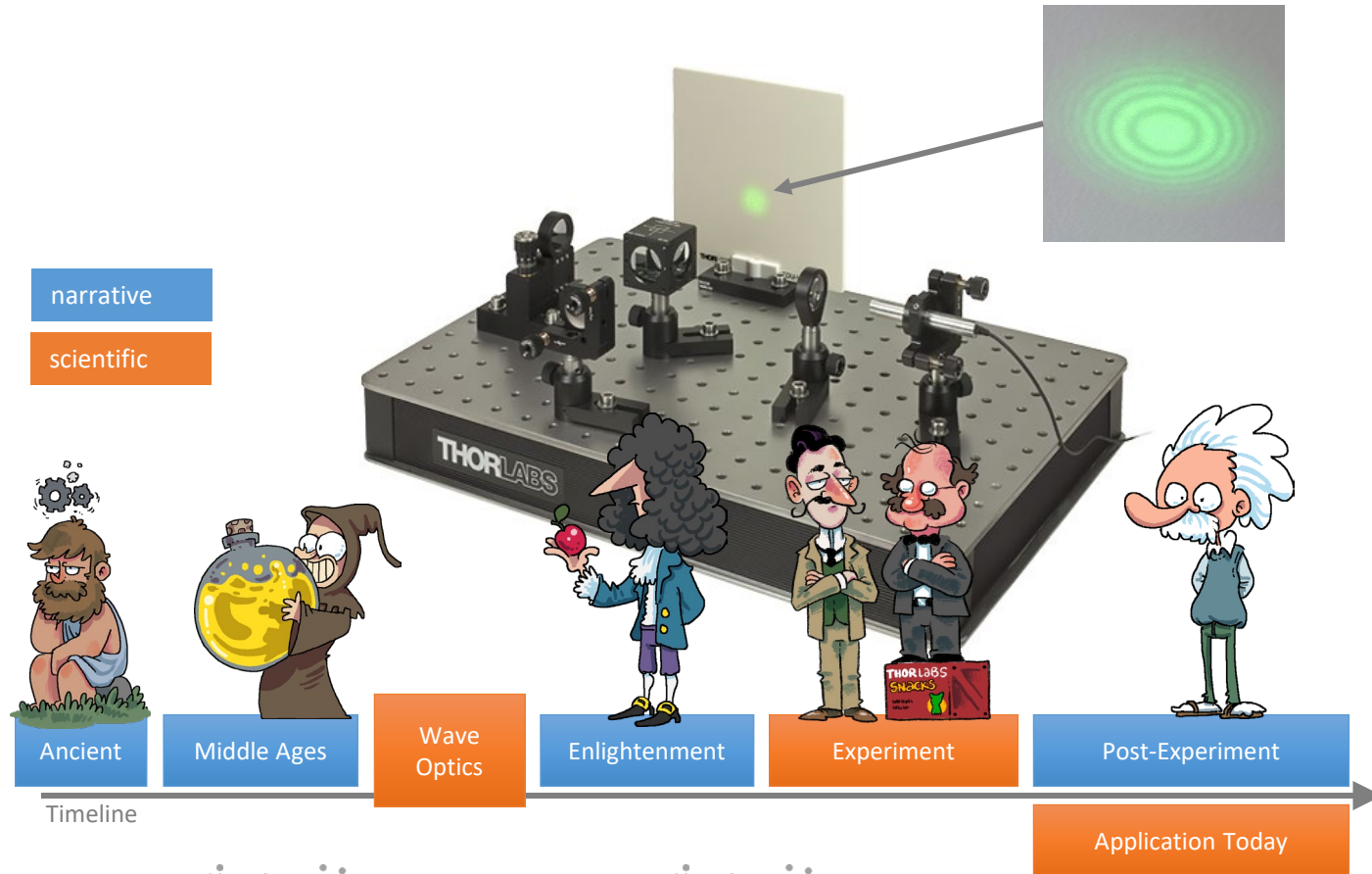
XR TWINLAB

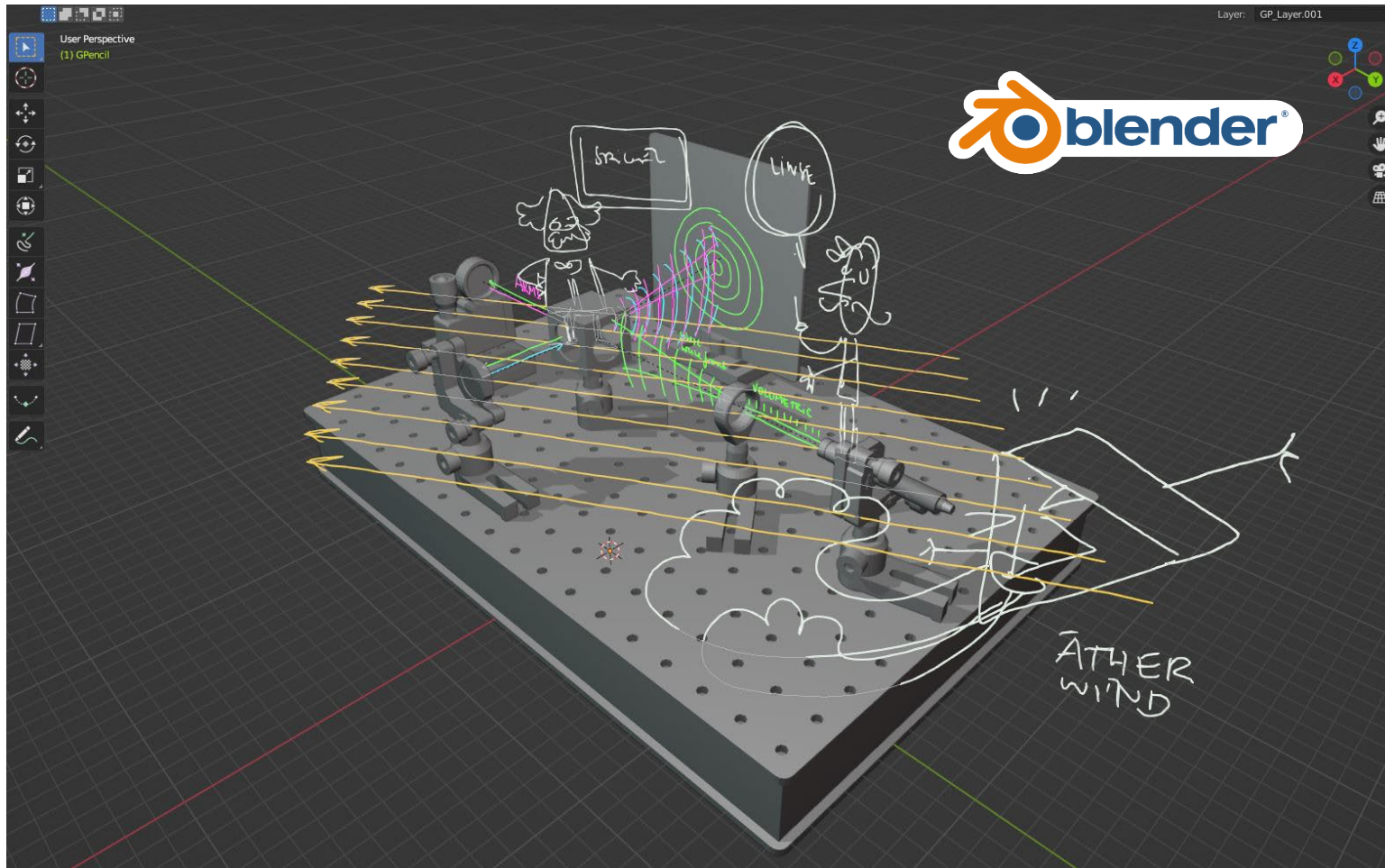


Thorlabs
Michelson Interferometer
Educational Kit + XRTL

XRTL GAMIFICATION & NARRATION FRAMEWORK







KTHXBAI!

XR TwinLab & Comics & Games!

- Augmented Games and Storytelling on virtualized remote experiments is educational and fun
- Endless possibilities with easy to learn open technology and software
- Collaborations with EU-funded BioQantSens Project 101120051).
- We are looking for partners for new experiments to build!
- GitHub: https://github.com/Lichtwerkstatt/XRTL_SPA



XRTL Team
Jari, Falko
Clara und
Johannes



Funded by
the European Union



BioQantSense

Dipl.-Inf. Johannes Kretzschmar
johannes.kretzschmar@uni-jena.de
+49 (0) 3641 9-47577