

## TUTORIAL

### Generalized Web Augmented Reality (WebAR) features from the experience *from hostile to hospitable*

Software to install:

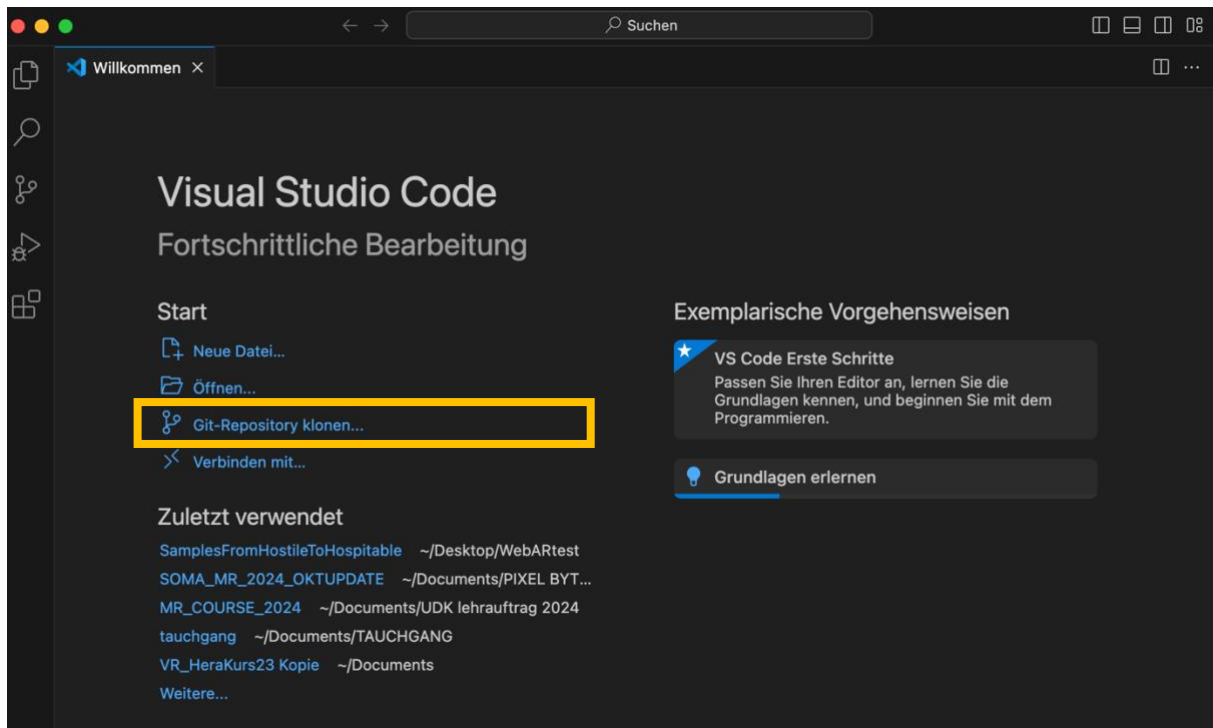
[Visual Studio Code](#) or similar editor.

Create free account for [Netlify](#).

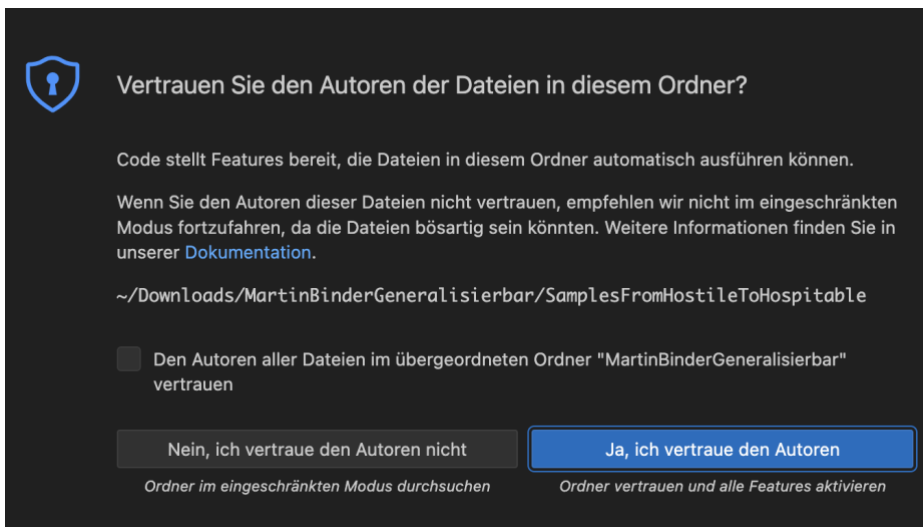
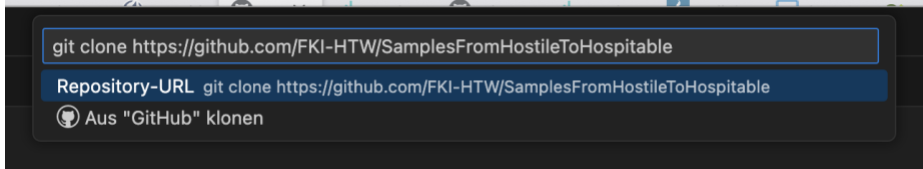
GitHub link to project:

<https://github.com/FKI-HTW/SamplesFromHostileToHospitable>

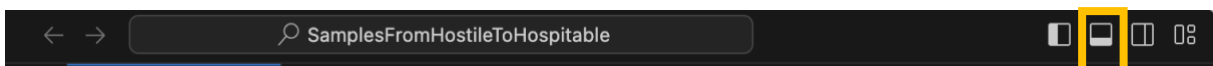
Open Visual Studio Code and click Git-Repository klonen...



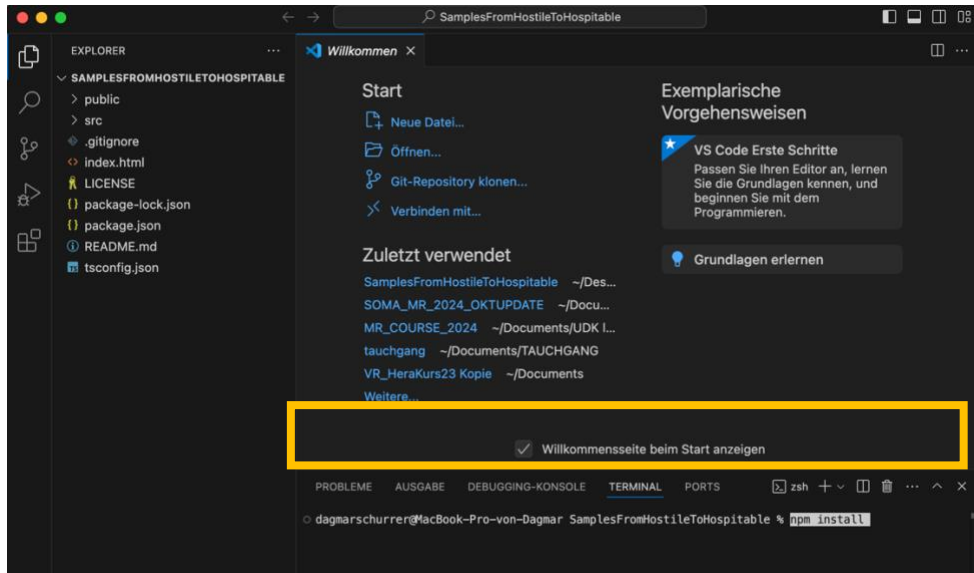
## Paste GitHub Link to project and trust authors



In Visual Code open the terminal window if it is not open yet by clicking the icon top right.



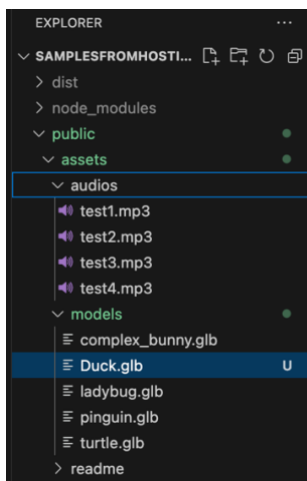
The first time you open the project type in the terminal:  
*npm install* and click enter



## Add your own 3D models and audio files in Visual Studio:

Supported file format for 3D models is glb / gltf, for audio mp3 / wav / m4a.

Drag & drop them in the corresponding folder:  
Public > assets > audios / models



Navigate to *database.json* and exchange the file names with your own models and audio files by copy and paste. You can add and delete as many ID-blocks as you need.

The »time« corresponds to the start time to play the audio in the timeline in seconds.

```

1  [
2
3  {
4    "ID": 0,
5    "contentName": "Spot 0",
6    "pathModel": "assets/models/ladybug.glb",
7    "pathAudioFiles": [
8      {
9        "path": "assets/audios/test1.mp3",
10       "time": 2
11      },
12     {
13       "path": "assets/audios/test2.mp3",
14       "time": 8
15     }
16   ],
17 },
18 {
19   "ID": 1,
20   "contentName": "Spot 1",
21   "pathModel": "assets/models/pinguin.glb",
22   "pathAudioFiles": [
23     {
24       "path": "assets/audios/test3.mp3",
25       "time": 2
26     },
27     {
28       "path": "assets/audios/test4.mp3",
29       "time": 10
30     }
31   ]
32 },
33 ]

```

### Customize button(s):

To add and label a new button, go to *GameScene.ts* and adapt the following syntax:

Adjust the button text by replacing e.g. 'Content A'

Adjust the number to the corresponding ID-block in *this.prepareData (0)*

Adjust *Offset* and *Vertical distance* of the buttons as needed

```

src > scene > TS GameScene.ts > ...
10 class GameScene {
34   private constructor() {
57     // debug 0:
58     this.ui_singleton.createVerticalButtonLayout(
59       {
60         { id: 'btn', text: 'Content A', onClick: () => this.prepareData(0)
61         { id: 'btn', text: 'Content B', onClick: () => this.prepareData(1)
62         { id: 'btn', text: 'Content C', onClick: () => this.prepareData(2)
63       },
64       40, // Top-Offset
65       60 // Vertical distance between buttons
66     },
67   },
68   this.timeline = new Timeline(this.scene, this.camera, this.renderer);
69   this.arManager = new ARManager(this.renderer, this.eventManager);
70
71   window.addEventListener("resize", this.resize, false);
72
73   this.eventManager.subscribe("placeObject", () => this.startTimeline());
74   this.eventManager.subscribe("stopScene", () => this.timeline.resetTimeline());
75
76   this.renderer.setAnimationLoop(this.update);
77 }
78
79 private async prepareData(index: number) {
80   this.loadedJSON = await this.loadExternalData(index);

```

**To test the WebAR experience on your Android smartphone (not possible on iPhone)**

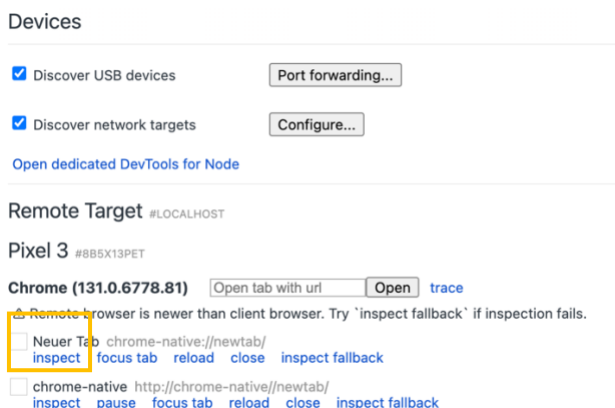
Type `npm run dev` into the terminal and click enter

Connect your Android Device with your computer by cable

Open Chrome Browser on your computer and copy paste `chrome://inspect/#devices`

Allow USB debugging on your smartphone and make sure you have the developer mode enabled

In the Chrome Browser click on *inspect* under Neuer Tab / New Tab

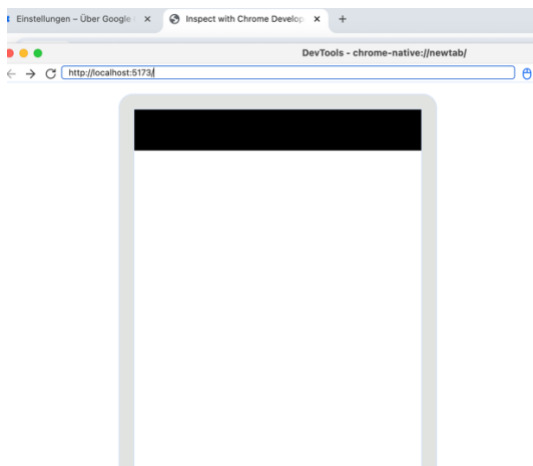


Copy the Local URL in the Visual Code Terminal

```
PROBLEME  AUSGABE  DEBUGGING-KONSOLE  TERMINAL  PORTS
o dagmarschurrer@MacBook-Pro-von-Dagmar SamplesFromHostileToHospitable % npm run dev
> generalisedfeatureswebar@0.0.0 dev
> vite

VITE v5.4.8  ready in 128 ms
→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```

And paste it in the Chrome Browser

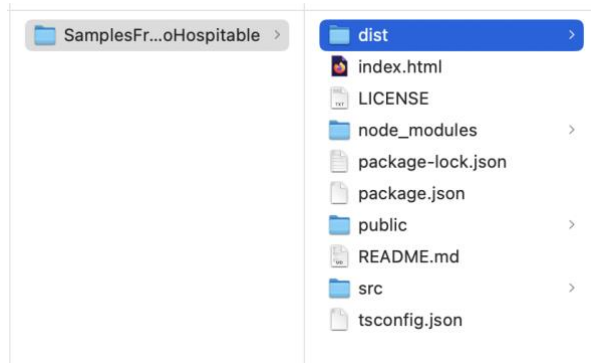
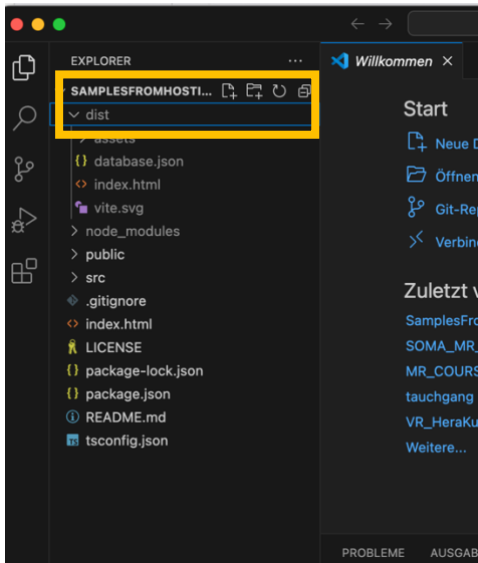


### To build the WebAR experience:

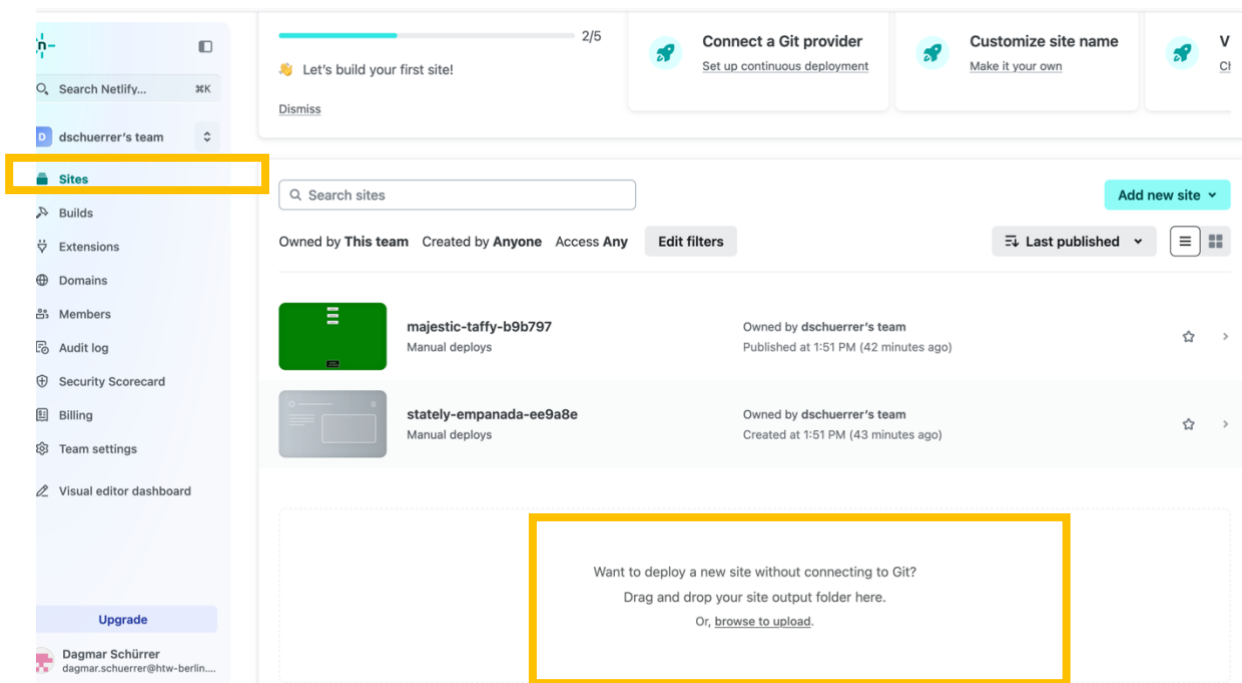
Go to the Visual Code Studio terminal. (If you are currently testing the app, close the test by typing `cntrl-C`)

Type `npm run build` in the terminal and click enter

- a `/dist` folder is created



Go to Netlify in your browser  
Make sure Sites is selected in the menu, drag & drop the */dist* folder in the white area on the bottom of the page, or click on *browse to upload*



When uploaded click on *Open production deploy* to create the URL of your WebAR experience, which you can now share.

### Published deploy for magical-kitten-918a8e

Today at 3:06 PM

Production    ↓ Download

[Open production deploy ↗](#)    [Lock to stop auto publishing](#)

[Options ▾](#)

If you make changes to your project, you must make a new build.

### Supported Browsers:

You have to use [XR Browser](#) for iPhone (WebAR is not running on Safari or Firefox yet)

Google Chrome is recommended for Android