

An isometric illustration of a city street scene. In the foreground, a green pickup truck is driving on the left side of the road. In the middle ground, a white delivery truck is driving on the right side. In the background, a worker in blue overalls is using a power tool on the sidewalk. A police car with 'POLICE' written on its side is driving in the lower middle ground. White lightning bolts are scattered around the vehicles and the worker, symbolizing noise. The background includes green trees, a street lamp, and a blue building.

Bewusstsein der Lärmbelastung im öffentlichen Raum

Summer School 2023

von Taha Sif Eddin, Clara Storandt, Duc Anh

Agenda

1. Problemstellung
2. Konzept
3. Hardware und Software
4. Durchführung
5. Demonstration
6. Projektergebnis
7. Ausblick
8. Fazit



Problemstellung



Übermäßige Lärmbelästigung



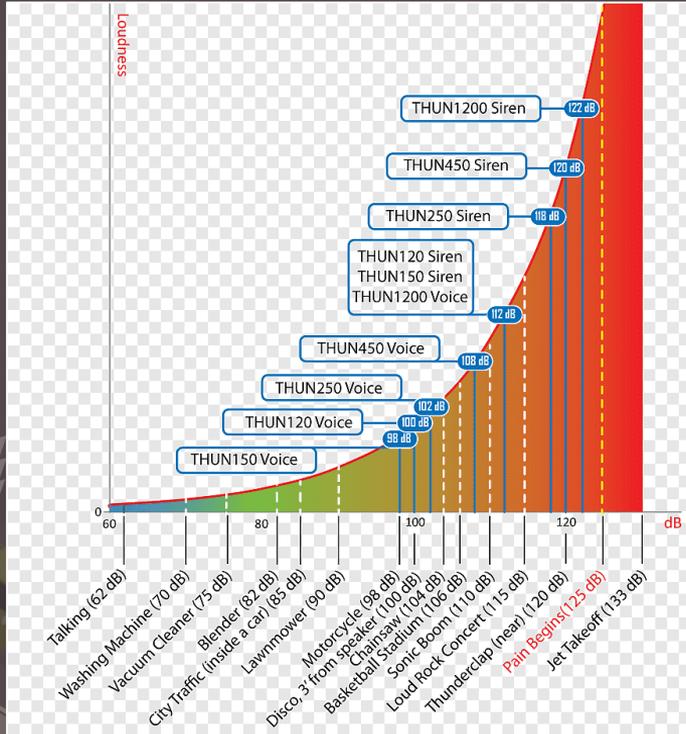
Laustärke – Was ist das?



Akzeptanz im öffentlichen Raum



Konzept



Verständlich Visualisierung der Lautstärke

Selbsterklärende Darstellung

<https://e7.pnggg.com/pngimages/272/226/png-clipart-decibel-sound-meters-a-weighting-noise-others-angle-text.png>

Konzept

User Stories

„Als Person, die sich um meine Gehörgesundheit beschäftigt, möchte ich schnell und einfach erkennen können, wie laut es in meiner Umgebung ist, damit ich weiß, ob ich den Bereich verlassen sollte.“

„Als Person mit Hyperakusis, möchte ich erkennen, in welchem Bereich die Laustärke ist, damit ich weiß, ob meine Reaktion verständlich ist.“

„Als normaler Bürger, möchte ich mehr Aufklärung zum Thema Lautstärke erhalten, damit wir uns aktiv schützen können.“

Konzept

Designideen – Design 1

Hintergrundgedanke:

- Farbverlauf -> steigende Lautstärke
- Farbverlauf von:
 - Grün nach Orange
 - Orange nach Rot
- Zusätzlicher Blinkereffekt bei zu hoher Lautstärke



Konzept

Designideen – Design 2

Hintergrundgedanke:

- „Farbverlauf“ ähnlich zum Design 2
- Farbverlauf hat harte Kannten
 - -> bessere Erkennung der Stufen

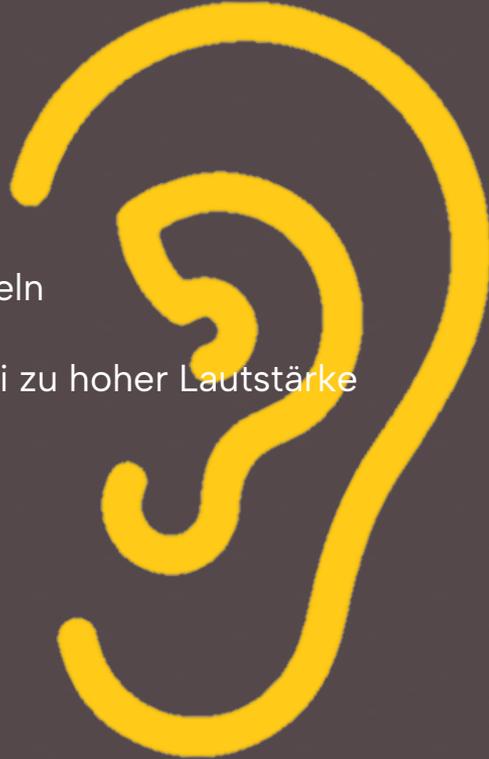


Konzept

Designideen – Design 3

Hintergrundgedanke:

- Kein Farbverlauf
 - Soll einer Ampel ähneln
- Zusätzlicher Blinkeffekt bei zu hoher Lautstärke



Konzept

Designideen – Design 4

Hintergrundgedanke:

- Design für große LED Tafeln
- Zwischenschaltung oder dauerhafte Anzeige
- Zwischenschaltung bei Lautstärke $> 80\text{dB}$
- Kategorisierung, Lautstärkeanzeige, Wiedererkennung, Warnhinweis



130 dB ist
Äquivalent zu
80 Mio bellende
Hunde!

Achtung!

Rettungsfahrzeug in der
Nähe
Lautstärke von **130 dB**



Halten Sie Abstand oder begeben Sie
sich an einen ruhigen Ort
Hörschädigung Gefahr

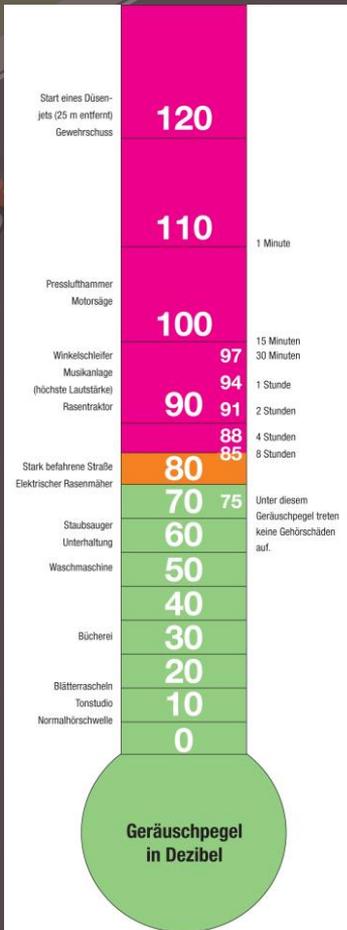
00:29



Hardware

- LED-Matrix (64x32 und 32x32)
- Microcontroller (ESP32 und ESP8266)
- Jumper-Kabel
- DC-Buchse
- Netzteil und Adapter
- Micro-USB-Kabel
- Stromkabel

Software



```
1021 static int dezibel_input = 30;
1022
1023 // Funktion, um das Intervall basierend auf der Dezibelzahl zu bestimmen
1024 int getInterval(int input) {
1025     if (input >= 110) {
1026         return 5;
1027     } else if (input >= 98) {
1028         return 4;
1029     } else if (input >= 86) {
1030         return 3;
1031     } else if (input >= 70) {
1032         return 2;
1033     } else if (input >= 35) {
1034         return 1;
1035     } else {
1036         return 0;
1037     }
1038 }
```

Software

```
1040 // Funktion, um das entsprechende Bild anzuzeigen
1041 void displayImageDependingOnDezibel() {
1042     dezibel_input = changeInput();
1043     int interval = getInterval(dezibel_input);
1044
1045     // Je nach Intervall das entsprechende Bild anzeigen
1046     if (interval == 5) { // Wenn das Intervall 6 ist (ab 110), zeige das letzte Bild
1047
1048         for (int i= 0; i < 5; i++) {
1049             drawImage(current_design[interval - 1], imageWidth, imageHeight);
1050             delay(250); // Verzögerung zwischen den klickenden Bildern (kann angepasst werden)
1051             display.clearDisplay();
1052             delay(100); // Verzögerung zwischen den klickenden Bildern (kann angepasst werden)
1053         }
1054     } else {
1055         // Zeige das entsprechende Bild (außer dem letzten)
1056         drawImage(current_design[interval], imageWidth, imageHeight);
1057     }
1058 }
```

Software

```
68 uint16_t static design_01_1[] = {
69     0x0000, 0x8410, 0xFFDF, 0xFFFF, 0xFFFF, 0xFFFF,
70     0xFFFF, 0xFFFF, 0xFFFF, 0xF7BE, 0xFFFF, 0x0000, 0x0000,
71     0x0000, 0xFFFF, 0xFFDF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF,
72     0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
73     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xF7BE, 0x0000,
74     0x0000, 0xF7BE, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
75     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFDF, 0x0000, 0x0000, 0x0000, 0x0000,
76     0x0000, 0x0000, 0x0000, 0x0000, 0xFFDF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
77     0x0000, 0x0000,
78     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFF9C, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
79     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFDF, 0xFFFF, 0xFFFF, 0xFFFF, 0xF7BD, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
80     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFDE, 0xFFFF, 0xFFFF, 0xFFDF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
81     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0xBDF7, 0xFFFF,
82     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xD6BA, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
83     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0xFF5D, 0xFFDF, 0xFFFF, 0xFFFF, 0xFFFF,
84     0xFFFF, 0xFFDF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
85     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFF9E, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0xFFD, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF,
86     0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
87     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0xFFDF, 0xFFFF, 0xFFFF, 0xFFFF, 0xF7BE, 0xFFFF,
88     0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000,
89     0x0000, 0xFFDF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000,
90     0xFFFF, 0xEF7D, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
91     0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0xFFFF, 0xFFFF, 0xFFFF, 0xFFDF, 0xFFFF, 0xFFFF,
```

Software

```
1060 // zeige das skalierte Bild
1061 void drawImage(uint16_t image[], int imageWidth, int imageHeight) {
1062     display.clearDisplay(); // immer Anzeige zurücksetzen, bevor etwas Neues angezeigt wird
1063     int counter = 0;
1064     for (int y = 0; y < imageHeight; y++) {
1065         for (int x = 0; x < imageWidth; x++) {
1066             display.drawPixel(x, y, image[counter]);
1067             counter++;
1068         }
1069     }
1070 }
```

Software

```
1127 void loop() {  
1128  
1129     displayImageDependingOnDezibel(); // Anzeige eines Bildes basierend auf Intervall  
1130     delay(1500); // Wartezeit zwischen den Anzeigen  
1131 }
```



Durchführung

Kick-Off



Kick-Off:

- Projektauswahl
- Ideen sammeln

Hürden:

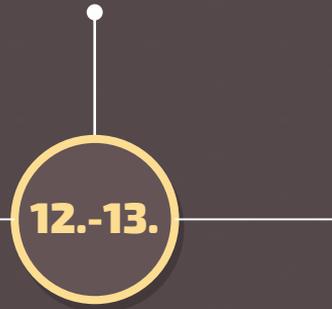
- Fehlende Kenntnisse/Fachwissen

Durchführung

Kick-Off



Recherche



Recherche:

- Fehlendes Fachwissen erarbeiten
 - Selbstrecherche und Material von Sara
- Beginn der Dokumentation

Hürden:

- Keine Hürden

Durchführung

Kick-Off



Recherche



**Arduino -
Einführung**



Arduino-Einführung:

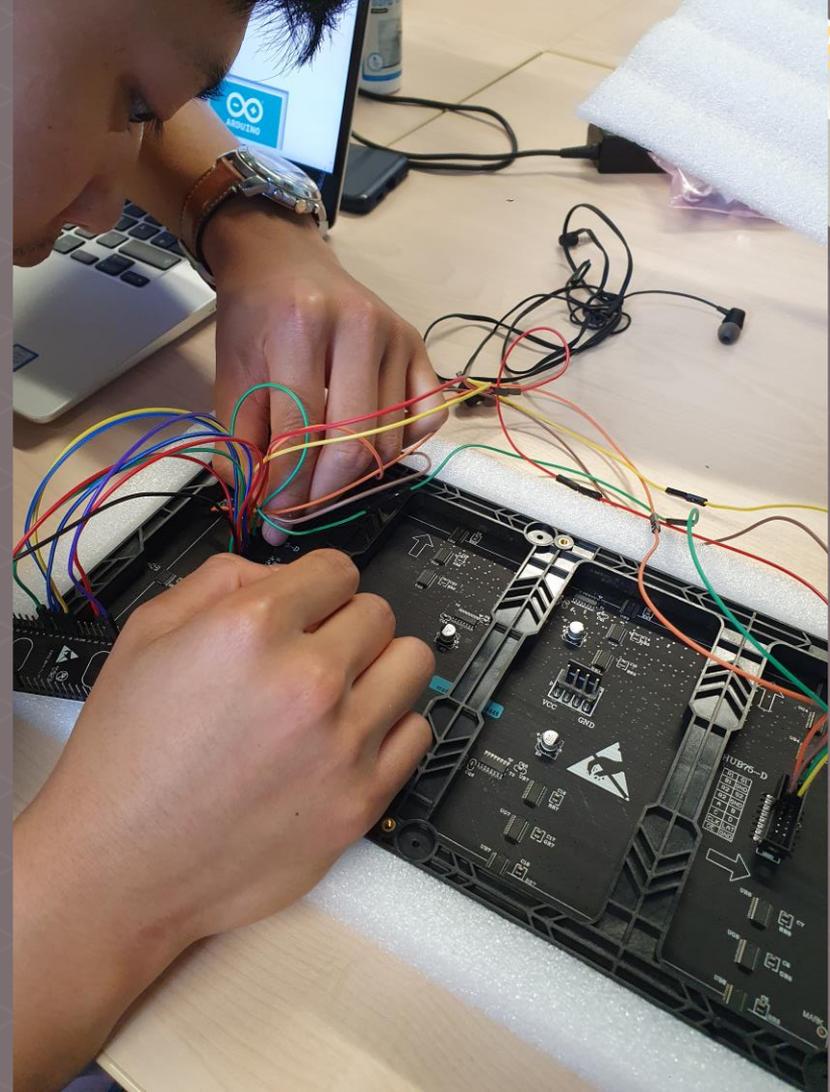
- An der TU mit Sara und Reni
- Einführung in die LED-Matrix

Hürden:

- Minderwertige/nicht funktionierende Hardware



1. Versuch mit der 64x32 Matrix
– Aufbau und
Zusammenstecken



Durchführung

Kick-Off



Recherche



**Arduino -
Einführung**



Prototype

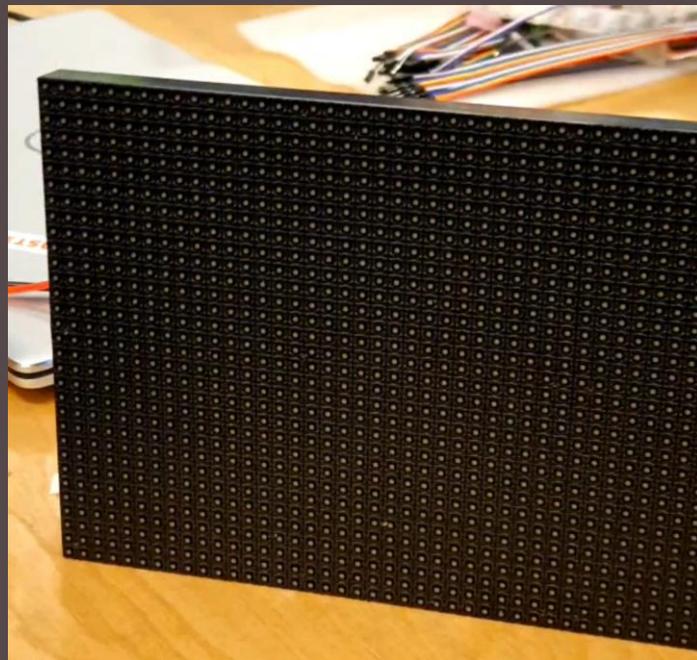


Prototype:

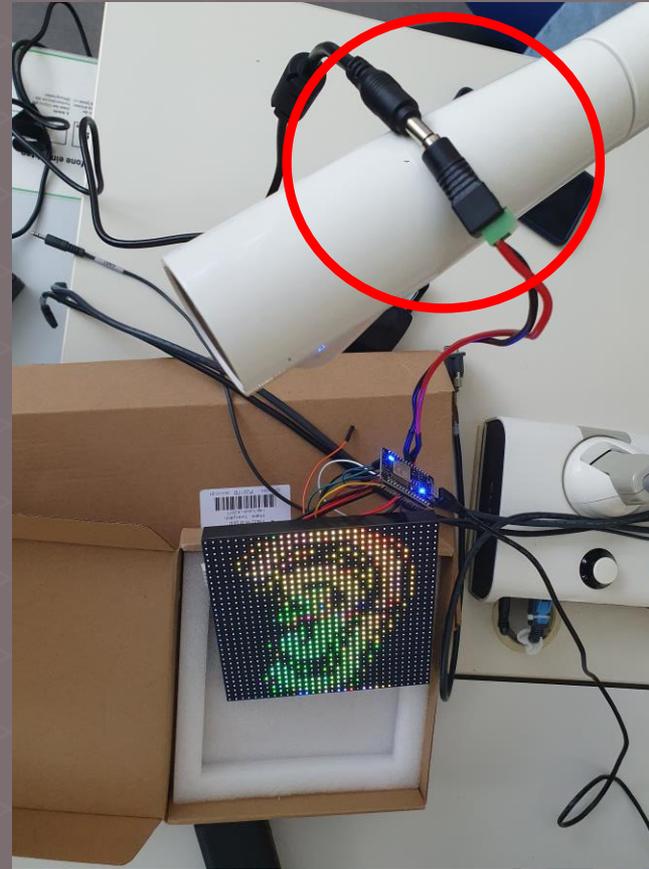
- Erstellen eines Prototyps
 - Implementierung einer Software
 - Arbeiten mit mangelhafter Hardware
- Kudo an zweite Gruppe für das Borgen der Hardware am letzten Tag

Hürden:

- Weiterhin minderwertige/nicht funktionierende Hardware
- Hardware in Brand



1. „gelungener“ Versuch mit der 32x32
Matrix → Designanpassung und
Stromkabel Problem





Auf Kamera festgehalten →
Qualmen der 64x32 Matrix



Durchführung

Kick-Off



Recherche



**Arduino -
Einführung**



Prototype



Goodlab



Goodlab:

- Werkstatt mit Lasercutter für Holz

Hürden:

- Designform in passendes Format konvertieren für den Lasercutter



Erstellung Ohr Design mit dem Lasercutter



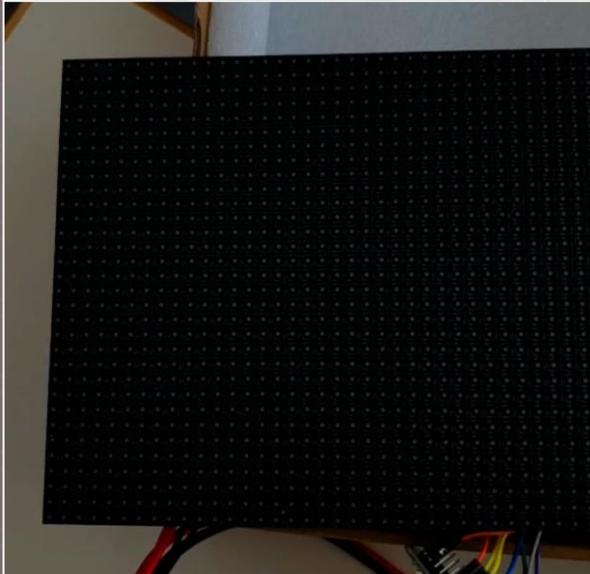
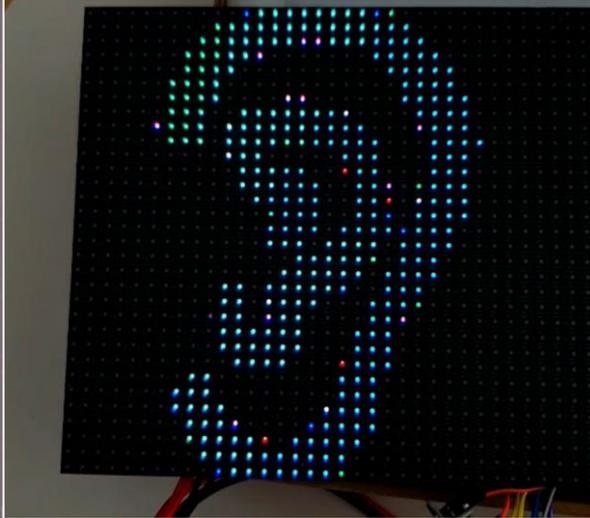
An isometric illustration of a city street scene. In the upper left, a blue building with a traffic light and a yellow car with white lightning bolts above it. In the upper right, a white van with yellow lightning bolts above it. In the lower left, a green building with a traffic light and two green trees. The background is a dark gray grid.

Demonstration

Live Vorführung von Design 1, 2
und 3

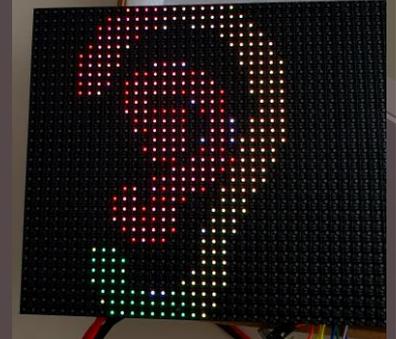


Demonstration



Projektergebnis

- Visualisierung Konzeptideen von Design 1 bis 3
- Mock Up weiterer Design Idee (Design 4)
- Designidee als Holzplatte
- Einsicht zum Thema Lautstärke und Lärm
- Einsicht zum Thema Microcontroller und Hardware



Ausblick

- User Tests – Akzeptanz
- Technische Umsetzung
- Rechtliche / Organisatorische Dokumentation
- Verknüpfung Prototyp mit Lautstärke Sensor und Geräuscherkennung

Fazit

Was lief gut?

- Einarbeitung in das Thema
- Verwendung der Software
- Teaminterne Aufteilung und Arbeitsweise
- Trotz Hindernisse ans Ziel
- Weitere kleine Projekte

Was könnte verbessert werden?

- Zeitmanagement
- Kommunikation (extern)
- Überprüfung Hardware

Danke für Eure Aufmerksamkeit!

Habt Ihr noch Fragen?

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon**, and infographics & images by **Freepik**

