

Magic Table

Interaktiver Tisch mit [projection mapping](#) system als [Augmented Reality](#) Installation.

Verwendungszwecke

- Entwicklungsumgebung für projection mapping
- Gaming Table
 - Table Top (TBD)
 - Visuelle Programmierumgebung für die Früherziehung von Softwareentwicklung

Installation im Lab



Rechner, Beamer und Kamera sind über dem mittleren Tisch vor der Küchenzeile montiert.

Der Rechner hat die IP 192.168.1.18 und den Hostnamen magictable.flka.space.

Username: fabi

Password: <Standard>

The project is installed at `/home/fabi/projects/magic-table`

Programmierungsumgebung

Quellen

[Github Repo](#)

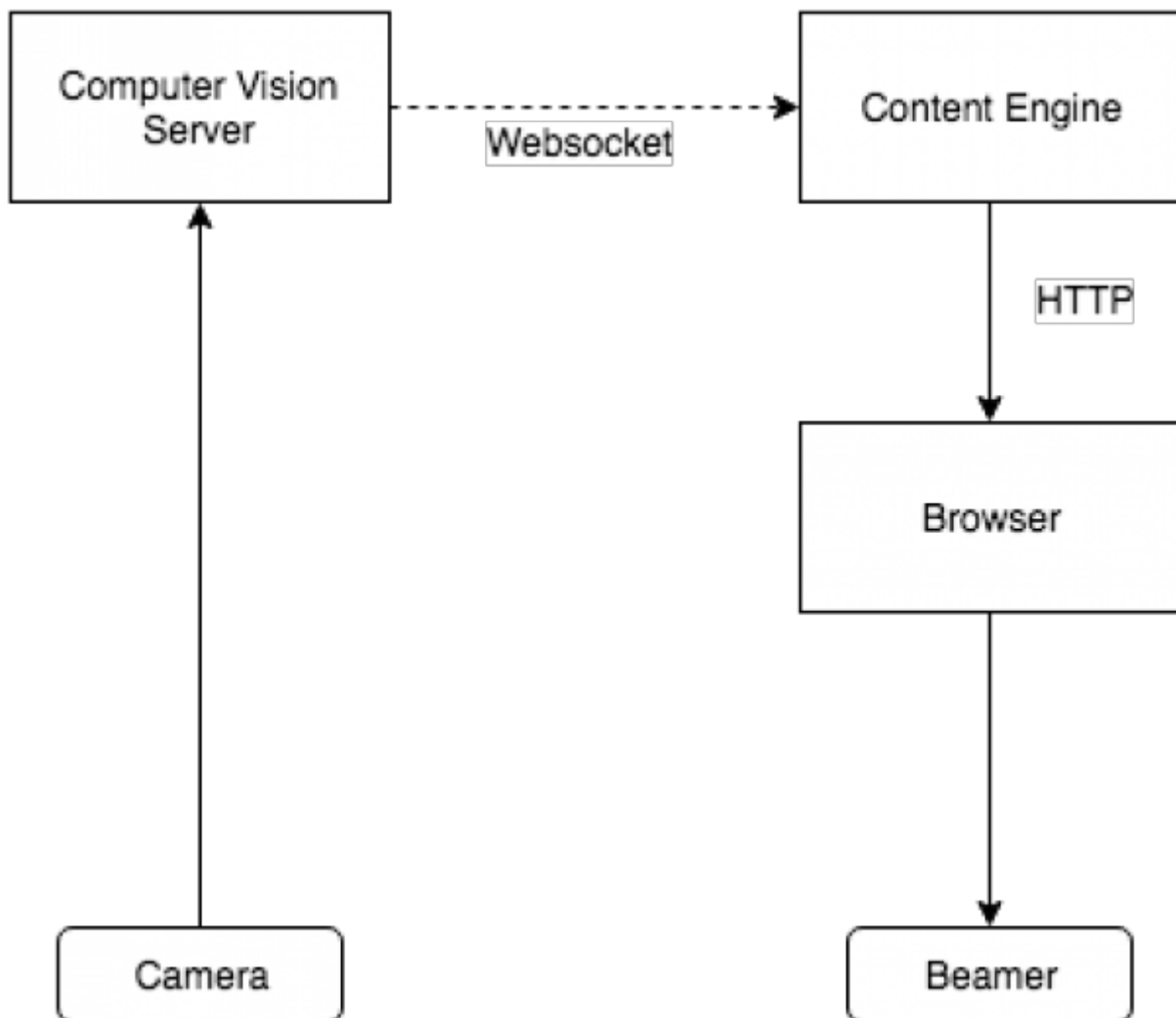
Architektur

CV-Server

Der CV-Server nutzt eine kalibrierte Webcam um Marker mittels [OpenCV](#) zu erfassen und sendet deren Positionen und Orientierungen relativ zur Kamera via [Websocket](#) an alle verbundenen clients.

Content Engine

Die Content Engine ist ein Client für den CV-Server. Sie stellt einen Webserver bereit welcher eine Webapp bereitstellt die sich via [Websocket](#) an den CV-Server verbindet und anhand der Daten ein Bild generiert welches mittels Kalibriertem Beamer wieder projiziert wird.



CV-Server Websocket Interface

Interface um marker updates zu übermitteln. Clients die sich mit dem Server verbinden bekommen messages mit dem folgenden Format:

```
[
  {
    ids: number[];
    marker: number[][][];
```

```
    transform: number[][];  
  },  
  ...  
]
```

Das JSON-Array enthält eine Liste aller Messages die auf einmal vom Server verschickt wurden (sollte meistens genau eine Nachricht enthalten).

Jedes Objekt darin hat

ids - Liste aller Marker ids in dieser message. Eine Marker ID ist eine nummer von 1-255 welche im Marker eindeutig encodiert ist.

marker - Liste aller Eckpunkt-Koordinaten der Marker (`[<marker index>][<eckpunkt index>][x, y]`)

transform Liste der Marker Matrizen welche die Orientierung der Marker im dreidimensionalen Raum relativ zur Kameraposition beschreiben.

Web Interface (HTTP)

Die Content Engine veröffentlicht eine Webapp die mit einem Browser erreichbar ist.

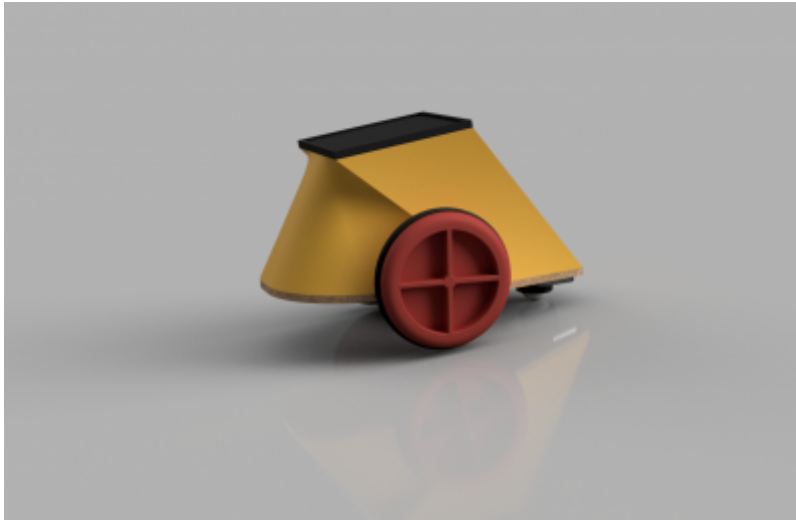
Visuelle Programmierumgebung für Kids

Spielerisch programmieren lernen.

Idee

Das Spiel projiziert mittel Beamer ein Spielfeld auf einen Tisch. Ein kleiner **Roboter** fährt sobald er eingeschaltet wurde an den Startpunkt der Karte. **Programmierblöcke** die auf den Tisch gelegt werden werden vom Spiel ausgewertet und interpretiert in dem der Roboter versucht den Anweisungen zu folgen.

Roboter



ESP8266 basierter Roboter mit 2 Rädern.

BOM

- Wemos D1
- Wemos Battery Shield
- Wemos Motor Shield
- 3V Getriebe DC Motor
- Lipo Akku
- 2 LEDs
- Lasercut & 3D Printed Parts: [3d Model](#)
- 3cm O-Ring

Firmware

<https://github.com/fablab-ka/magic-table/tree/master/robot/firmware>

registriert sich unter `magicrobot.flka.space` und kann via HTTP oder Websocket angesprochen werden.

[20190113_152549.mp4](#)

Websocket Interface

Message format: `#<direction><velocity><amount>`

direction: Eine Nummer mit folgenden Werten und Bedeutungen.

- 0 - Stop
- 1 - Rotate Left
- 2 - Rotate Right
- 3 - Forward
- 4 - Backward

velocity: The Velocity of the movement. Values are between 0 and 1023.

amount: The Amount of the Movement. Values are between 0 and 1023. For rotations 1023 means a full rotation. For linear movements 1023 means 10cm.

Programmierblöcke

Table Top Gaming

TBD