

title: Projektauftrag  
author: Jakob Huemer-Fistelberger  
date: 2025-03-03

## Name: ShareStone / CircuitForge

---

### Definitionen

---

**Schematic:** Blaupause in Minecraft

### Projektbeschreibung

---

Eine Plattform für das Teilen von Redstone-Technik und -Kreationen in Form von Posts.

Posts bestehen hauptsächlich aus einer Galerie, Downloads und einer Beschreibung.

Diskussionen gibt es neben Posts. Sie enthalten keine Struktur etc. sondern sind eine reine Diskussion über ein Thema von Redstone in Minecraft. (Integriertes Forum)

Andere Personen können unter Posts und Diskussionen kommentieren.

Das Kommentarsystem ist verschachtelt bis zu voraussichtlich maximal 5 Leveln. (Baumstruktur wie bei Reddit)

### File Uploads

Schematics:

- litematic (Litematica)
- nbt (Minecraft)
- schem/schematic (WorldEdit / MCEdit)

Andere:

- world save file

### Unique Selling Point

---

- Zentralisiertes System für das Teilen, Verbreiten und Beibringen von Redstone-Technik
- Erweiterte Suchoptionen für verschiedene Anwendungsfälle
  - (Tags)
- Community-Driven:
  - Kommentare
  - Posts
  - mögliche Verbesserungsvorschläge
- Freundlich gegenüber Anfängern mit Redstone

### UI/UX

---

Die Website kann komplett ohne Account genutzt werden.

Will man jedoch etwas liken oder Posten muss ein Account erstellt werden.

## Landing Page

Die "Landing-Page" fängt direkt mit allen Posts und Diskussionen an, einer Suchleiste, Filteroptionen etc.

Beim ersten Besuch wird ein Tour-Banner angezeigt für neue Nutzer.

Es gibt folgende Reiter im Header:

- Posts
- Diskussionen
- [Logged in]
  - Profil dropdown mit settings, meine Posts, etc.
- [Logged out]
  - Login
  - Registrieren

### 1 Post Ansicht

Ein Post hat eine Main-Sektion mit der Galerie, Beschreibung, Downloads, etc.

Darunter sind dann die Kommentare.

### 1 Diskussion Ansicht

Eine Diskussion hat als Main-Sektion nur einen speziellen Start-Kommentar. Darunter folgen dann andere Kommentare wie bei Posts.

### Editierbarer Content

User können Editieränderungen für *einen Post* / *eine Diskussion* vorschlagen, der vom Autor akzeptiert werden kann.

Ein User kann eine Editierübungsänderung pro Post vorschlagen.

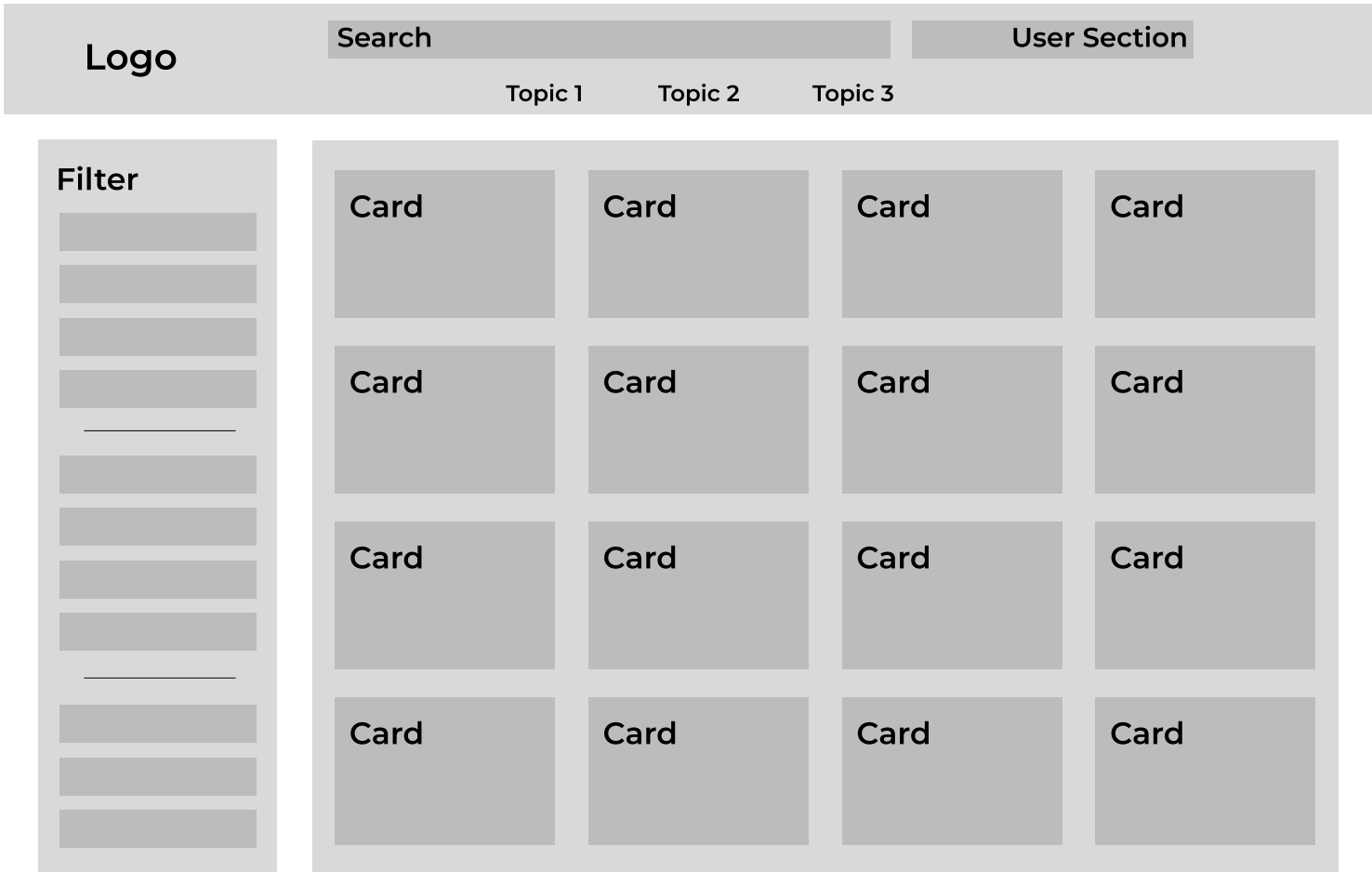
### Markdown Styling

Textinput in Posts, Kommentare, etc. unterstützt markdown Styling wie fett, kursiv, überschrift, liste, etc.

### Lesezeichen

Nutzer können sich Posts, Diskussionen und Kommentare speichern und in ihrem Profil unter Lesezeichen anschauen.

## Grober Wireframe



## Coder Plan

### PHP (Backend)

#### Überblick

Ein REST-Service stellt endpoints für das Frontend bereit.  
Die Filterung von posts läuft hauptsächlich am Server ab

- Login System mit Session Management
- Posts, Kommentare, Diskussionen
- MySQL Datenbank

#### Kommentare

Posts, Diskussionen und Kommentare haben alle eine eigene entity ID welche UNIQUE ist.  
Wird im Datenmodell näher erläutert.

Ein Kommentar referenziert eine solche entity ID als Parent.  
So wird das Nesten einfacher und mehr modular.

#### Likes/Dislikes

Tabelle Ratings besteht aus EntityID (PK,FK), UserID (PK,FK) und rating.  
So wird sichergegangen, dass ein Nutzer nicht gleichzeitig Liken und Disliken kann.

### Vue.js (Frontend)

## Kommentar-Komponente

Eine Kommentar-Komponente erhält ein level und die entity ID des Kommentars. Der Kommentar kümmert sich selbst darum, alle Unter-Kommentare vom Server zu fetchen. (Rekursiv)

## Markdown Styling

Markdown in Posts, Diskussionen und Kommentaren wird mit einer Library wie

[TipTap](#),

[Lexical](#)

oder ähnlichem implementiert.

# 3D-Rendering Schematic

---

## SchematicWebView:

[GitHub](#)

Kann mithilfe einer Client-Jar folgende Files in 3D Rendern:

- .schem
- .schematic
- .nbt

SchematicWebView unterstützt aber keine Block Entities.

## Datenbank: MySQL

---

## Datenmodell

---

