Music Machine

webapp for generation of beats using samples and the Web Audio API.

Philip Kirchner, Ruben Stößer, Matthias Heieck, Maximilian Ott

Gliederung

Überblick

Verwendete Komponenten

Verwendete Libraries

Präsentation der Anwendung

Fazit

Überblick

Webanwendung zum

- Erstellen von Beats mit optionalen Effekten
- Visualisieren des erstellten Beats (Spectrum Analyser)
- Import und Export von erstellten Beats
- Aufnahme des Beats und Speichern als ".wav" Datei

Verwendete Komponenten

AngularJS



Google Material Design (Angular Material)



Web Audio API



Verwendete Libraries

TunaJS

https://github.com/Dinahmoe/tuna

WAAClock

https://github.com/sebpiq/WAAClock

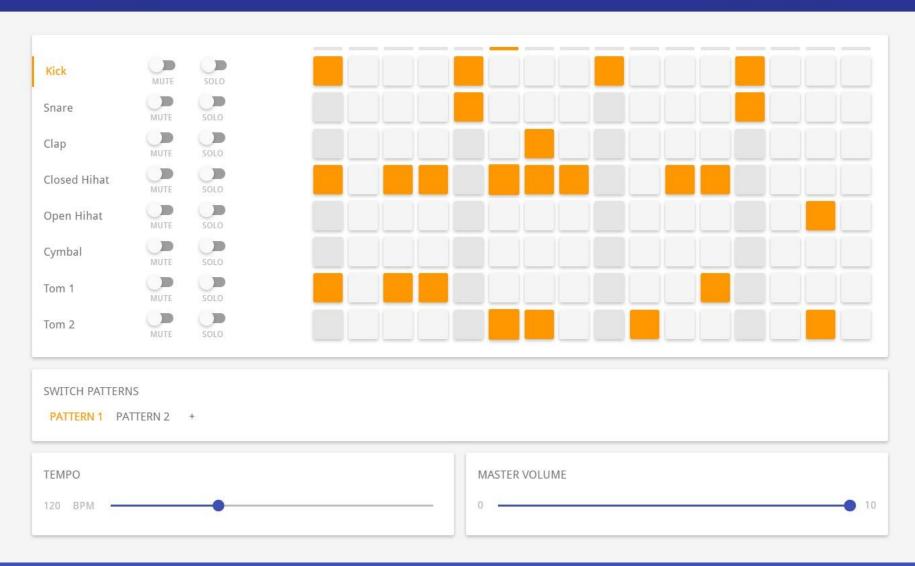
recorderJS

https://github.com/mattdiamond/Recorderjs

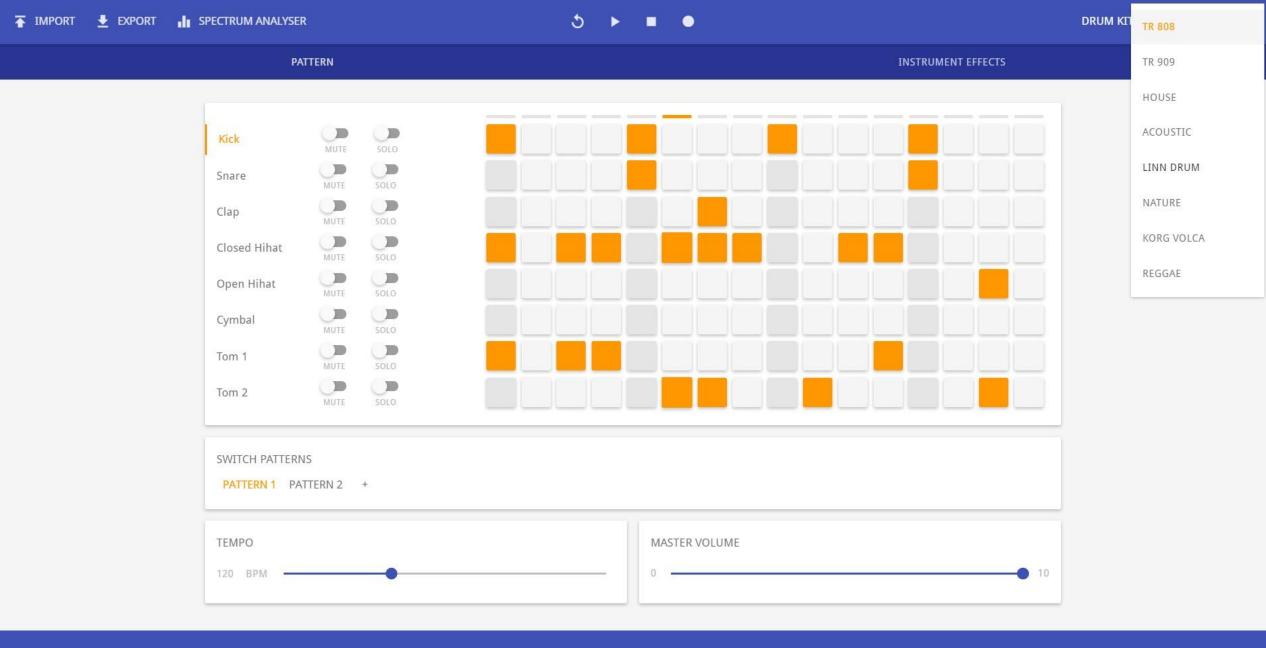
Präsentation



PATTERN INSTRUMENT EFFECTS

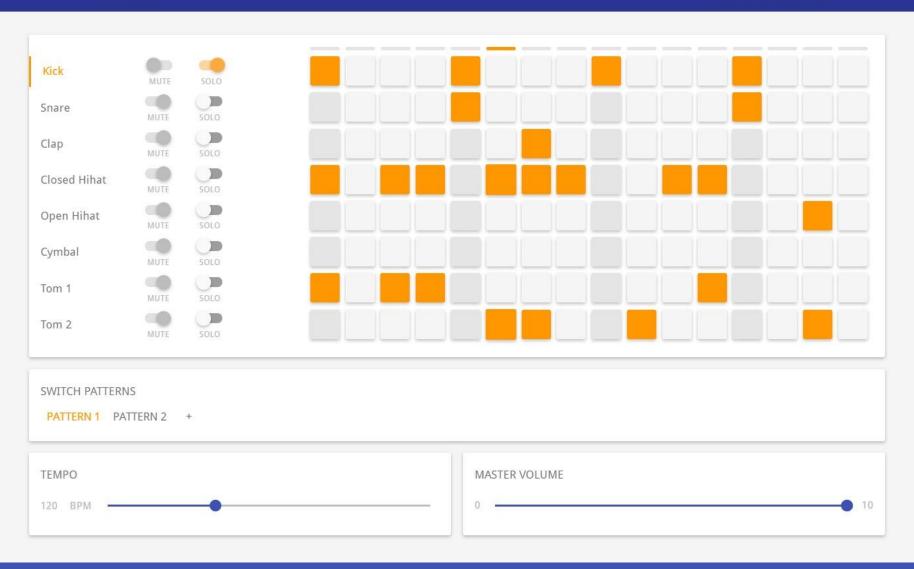


Pattern Screen



Kit Selection

PATTERN INSTRUMENT EFFECTS



Solo & Mute Instruments

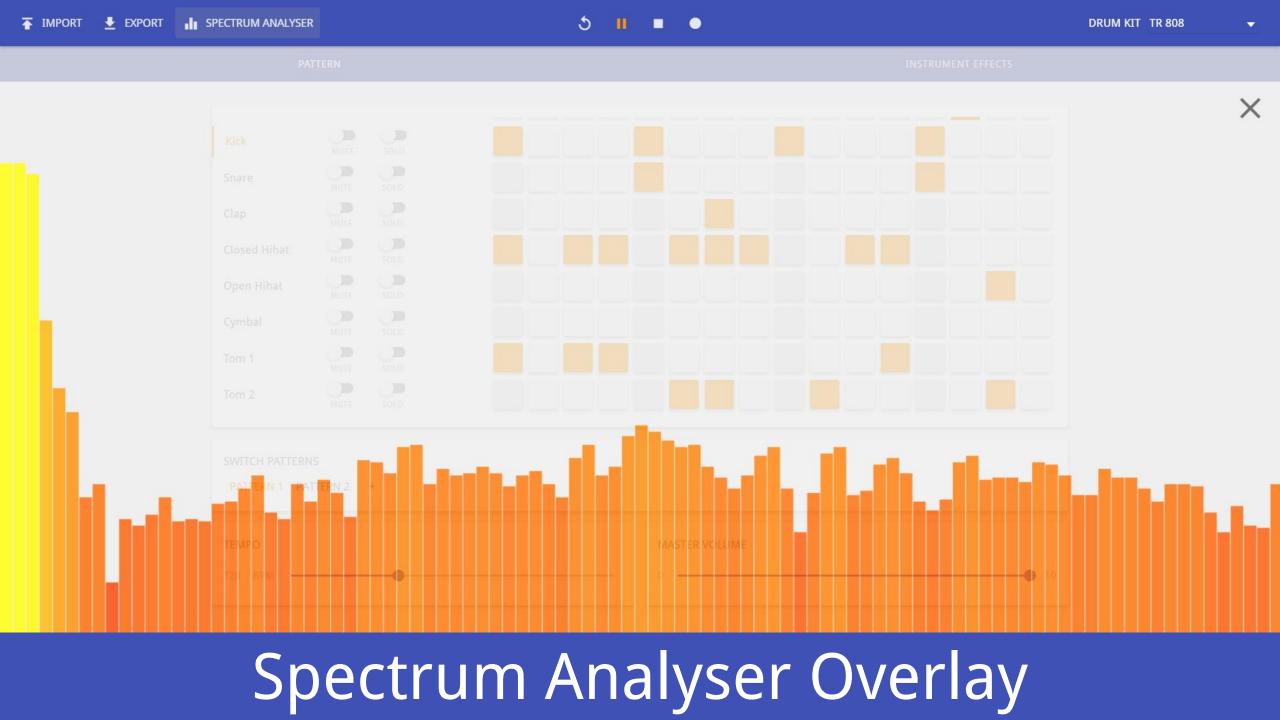


INSTRUMENT EFFECTS

PATTERN

Effects for Kick 0 PITCH VOLUME 0 **PANNING** FILTER LOW PASS 0 DELAY DELAY TIME DELAY FEEDBACK 0 0.8 20KHZ DELAY CUTOFF 20HZ

Instrument Effects Screen



Fazit

- Web Audio API ermöglicht Entwicklung von komplexeren audio-orientierten Anwendungen im Browser
- Browser Support und Performance sind noch stark limitiert
- AngularJS eignet sich für diese Art von Single-Page Anwendungen hervorragend
- Angular Material bietet einfache Implementierung des Google Material Designs in AngularJS Anwendungen

Zukunftsausblick

- Synthesizer (MIDI) Anbindung
- Mehr Effekte (Abhängig von Performance)
- Möglichkeit eigene Drumkits hochzuladen und zu verwenden
- Sequencing ermöglichen. Aneinanderreihung von Patterns in einer bestimmten Reihenfolge
- Eigenen Input Kanal (z.B. für Mikro, Gitarre etc.)

Vielen Dank für Ihre Aufmerksamkeit!