

Projektkonzept - Music Controller

Moritz Stoll - 2288654

Simon Ruisinger - 2298552

Einleitung

Im Rahmen der Wahlpflichtveranstaltung Audio-Video-Programmierung setzen wir das Projekt "Music Controller" um. Das Projekt ist die Abschließende Prüfungsleistung in AVPRG. Da die Veranstaltung in einen Audio- und einen Video-Teil aufgeteilt wurde, hatten wir die Möglichkeit unsere Projekte nur in einem der beiden Teile, oder in beiden zu machen. Wir haben uns entschieden unser Projekt im Audioteil der Veranstaltung umzusetzen, da wir so unserem Interesse an Web-Technologien nachgehen können und wir uns für die Produktion mithilfe von Software interessieren.

Zum Abschluss des Projektes erfolgt eine Präsentation, für die wir ein Plakat erstellen werden und eine Lauffähige Version des Projektes zeigen wollen, die die anderen Kursteilnehmer und eventuelle Besucher der Abschlusspräsentation testen können.

Projektziele

Als Nutzer soll es möglich sein mit einem Gamepad-Controller (Playstation oder xBox) zu musizieren. Es ist möglich jede Controller-Taste und Joystick mit einem Sound (Ton, Akkord) oder Synthesizer Funktionen (Gain, Filter, Reverb...) zu belegen.

Zusätzlich wird es eine feste KI-Taste geben. Wenn diese gedrückt und gehalten ist, wird der nächste gespielte Ton / Akkord an die KI gesendet damit eine Melodie über einen Takt erzeugt wird. Diese wird solange gelopped, wie die Taste gedrückt ist. Auf diesen Grund Loop kann man nun mit den frei belegbaren Tasten eigene Töne und Akkorde dazu spielen.

Wir wollen auf der gemeinsamen Abschlusspräsentation ein Notebook aufstellen, mit der die Besucher das fertige Projekt live bedienen können. Das Prinzip der Anwendung ist auf einem Plakat möglichst kurz und anschaulich dargestellt.

Anforderungsanalyse Moritz

Im Folgenden beschreiben wir die konkreten Anforderungen an unsere Software. Hierbei unterscheiden wir in funktionale und nicht- funktionale Anforderungen.

Funktionale Anforderungen:

- Eine erreichbare Website
- Erkennung eines angeschlossenen Gamepads
- Anzeige/Feedback ob Gamepads angeschlossen wurde oder nicht
- Verarbeitung der Gamepad-Eingaben
- Belegung jedes Button auf dem Gamepads aus verschiedenen Funktionen frei wählbar
- Zur Belegung der Buttons stehen entweder die Funktionen "Note", "Akkord" oder "Synthesizer-Effekt" bereit
- Jeder Button des Gamepads kann entweder eine Note oder einen Akkord spielen, oder das Audiosignal mit einem Effekt des Synthesizers verändern.
- Es gibt ein Vordefiniertes Mapping, das der User direkt benutzen kann
- Es gibt eine KI Taste
- Ist eine Ton, oder Akkord erzeugende Taste gedrückt und wird die KI Taste gedrückt, wird mit einer KI eine zu dem Ton oder Akkord passende Tonfolge erzeugt
- Die erzeugte Tonfolge wird dauerhaft geloopt
- Die erzeugte Tonfolge passt auf das eingestellte Tempo und den eingestellten Takt
- Es können mehrere Tasten gleichzeitig gedrückt werden (so kann sich der User wie auf einem Klavier selbst begleiten)
- Die Latenz sollte so klein sein, dass sie das Musizieren nicht behindert (wie hoch sie dafür maximal sein darf, wissen wir noch nicht)
- Der Synthesizer, der die Töne erzeugt lässt sich einstellen und somit der von den Tasten gespielte Ton verändern
- Für Tastenbelegung des Controllers die Taste gedrückt halten und aus einer Liste die entsprechende Funktion oder den Ton auswählen
- Belegung der Gamepad-Tasten wird samt der aktuellen Funktionen grafisch dargestellt (siehe Anhang Mockup)
- Es gibt eine Record-Funktion, mit der das gespielte aufgezeichnet und als .wav-Datei heruntergeladen werden kann

Nicht-Funktionale Anforderungen:

- Die Benutzeroberfläche muss sehr schlicht sein, um den User nicht mit Optionen zu überfordern
- Kompatibilität mit Chrome, Firefox und Safari
- Die Anwendung ist durch modularen Aufbau leicht erweiterbar und Änderungen sind leicht umzusetzen

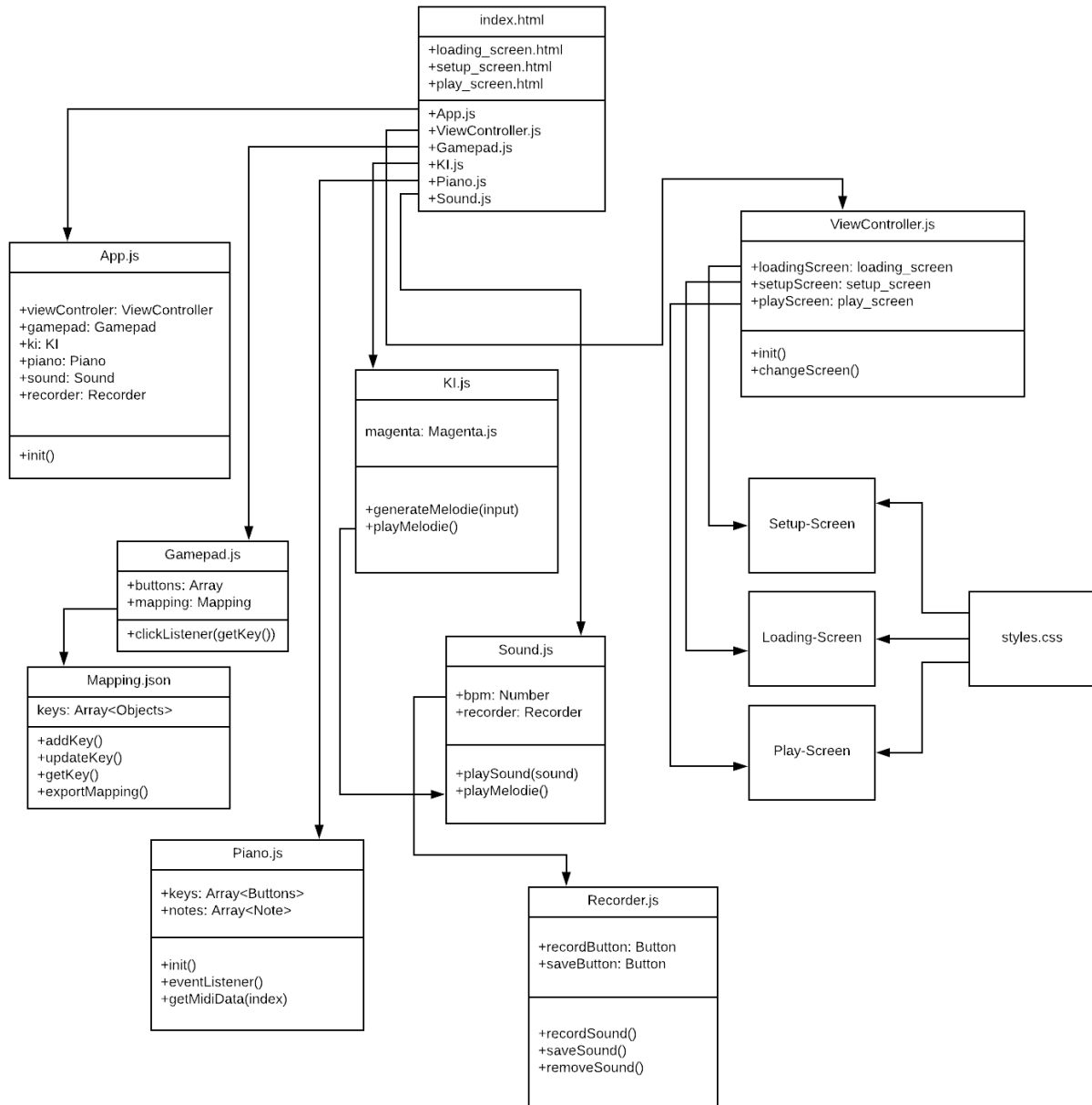
Technische Umsetzung Simon

Das Projekt wird durch den Einsatz von Webtechnologien realisiert.

Technologie	Einsatz
HTML	Standard um Struktur einer Webanwendungen für Browser zu definieren.
CSS	Standard um design einer Webanwendung zu erstellen.
Javascript	Für die Funktion der Webanwendung. Mit Javascript werden Web-Technologien einsetzbar.
Web Audio API	Projektanforderung. Um über den Browser Audio verfügbar zu machen. Einfach zu bedienen und durch W3C Standard hohe kompatibilität.
Web Gamepad Api	Leicht zu bedienen und durch W3C Standard hohe kompatibilität.
Magenta.js (Tensorflow.js)	Framework um mit einer KI Musik zu erzeugen. Viele Beispiele und bereits trainierte Modelle verfügbar.
Javascript Libraries für Musiktheorie	https://github.com/danigb/tonal https://github.com/saebekassebil/teoria

Technisches Konzept Simon

Das Diagramm zeigt den grundlegenden Aufbau der Web-Anwendung. Die “Klassen” können als Komponenten betrachtet werden. Grundlegende Funktionen der Komponenten sind ebenfalls dargestellt.



Bedienkonzeptes Moritz

Das User Interface ist in drei Screens gegliedert. Den Start-Screen, Setup-Screen und Play-Screen.

Start -Screen

Der Ladescreen wird solange angezeigt bis alle benötigten Komponenten, Sounds und Modelle geladen sind. Auf ihm ist eine Animation zu sehen.

Loading...



Setup-Screen

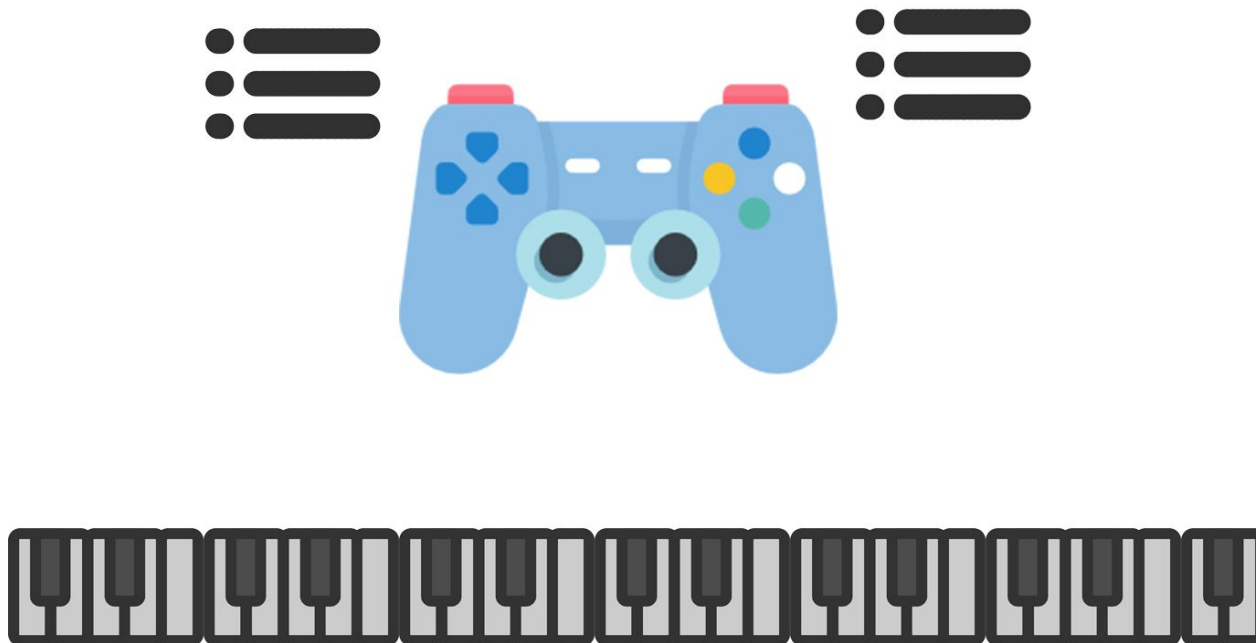
Der Setup-Screen ist deutlich komplexer aufgebaut. Auf ihm ist zentral eine schematische Darstellung des Gamepads zu sehen. Ist der Gamepad nicht angeschlossen, wird die schematische Darstellung ausgegraut und rot umrandet. Ist er angeschlossen, ist die Darstellung voll gesättigt und grün umrandet.

Drückt der User einen der Buttons auf dem Gamepad, wird dessen entsprechende Darstellung auf der schematischen Darstellung hervorgehoben. Gleichzeitig öffnen sich zu beiden Seiten des Gamepads zwei Listen. Eine um eine Note oder einen Akkord auszuwählen, die andere um eine Synthesizer-Funktion auszuwählen.

Um zwischen Note und Akkord zu differenzieren befindet sich über der entsprechenden Liste ein Switch. Wird dieser auf "Chord" gestellt öffnet sich neben der Liste mit den Noten eine zweite, kleinere. Hier lassen sich die Tonalitäten wie "Dur" oder "Moll" auswählen.

Drückt der User im Setup-Mode eine belegte Taste, wird diese hervorgehoben. Gleichzeitig wird der entsprechende Ton gespielt. Unten auf dem Setup-Screen befindet sich ein virtuelles Piano-Keyboard. Wird ein Akkord, oder eine Note gespielt, werden auf dem Klavier die dazugehörigen Tasten hervorgehoben.

Auf dem Gamepad befinden sich eine Start- und eine Select-Taste. Diese sind auf der schematischen Darstellung mit Play und Record Symbolen gekennzeichnet. Drückt der User eine der beiden Tasten wird die Darstellung reduziert und er gelangt somit zum Playscreen.



Play-Screen



Auf dem Play-Screen sind nur noch der Controller und das Piano-Keyboard zu sehen. Nach wie vor werden die Tasten des Keyboards, so wie die schematisch dargestellten Tasten bei einem Tastendruck hervorgehoben. Jedoch lässt sich nun keine neue Funktion mehr auswählen und der User kann nun frei mit der von ihm erzeugten Tastenbelegung musizieren.

Drückt der User nur die Play Taste, ist oben rechts in der Ecke ein Play Symbol zu sehen. Drückt er die Record Taste, startet er die Aufnahme und kann seine Musik nun aufnehmen. Drückt er sie erneut, stoppt die Aufnahme und über dem Keyboard erscheint ein Musikplayer mit einer Zeitleiste, einem Play/Paus-Button und einem Download symbol. Er kann die Aufnahme nun als .wav Datei herunterladen. Drückt der User nun erneut die Play Taste kehrt er zum Setup-Screen zurück.

Unsicherheiten

Wir sind uns nicht ganz sicher, wie gut der durch die KI-Taste erzeugte Loop vom Tempo und dem Takt in das Gespielte passt. Gegebenenfalls müssten wir hier mit einem Metronom arbeiten, das dem User den aktuellen Takt angibt.

Außerdem sind wir nicht sicher, wie schnell die Generierung einer Tonfolge ist. Da wir die Tonfolge live einspielen wollen, darf die Ladezeit hier wirklich nur sehr gering sein. Da wir noch nicht so vertraut mit Magenta.js sind, wissen wir nicht, ob eine Erzeugung in Echtzeit möglich ist. Als "Back-Up" haben wir uns überlegt, die Tonfolgen für jeden Ton und jeden Takt vorzuladen.

Zeitplan Moritz

Wir haben uns dazu entschlossen, das Projekt nach Scrum zu entwickeln und arbeiten somit mit einem Backlog und 1-Wochen-Sprints. Hierbei gibt es folgende Termine mit Zielen, die wir erreichen wollen:

30.10.2018	→	Konzeptvorstellung
5.11.2018	→	Einfacher Synthesizer, der MIDI-Signale abspielen kann
10.11.2018	→	Anbindung Gamepad und Auslesen der Inputs
16.11.2018	→	Funktion des Controller mit Noten (MIDI), Erzeugung der Töne
21.11.2018	→	Steuerung der einiger Synthesizer-Funktionen über Gamepad.
26.11.2018	→	KI-Taste - Melodien auf Basis eines gleichzeitig gedrückten Knopfes
27.11.2018	→	Vorstellung eines Prototyps
04.12.2018	→	Implementierung der fehlenden Funktionen
11.12.2018	→	Bug-Fixing
18.12.2018	→	Präsentation
10.1.2019	→	Fertiges Projekt
13.1.2019	→	Projektabschluss

Aufwandsschätzung

Task	Zeit min	Zeit	Zeit max
Grundlagen der Webseite erstellen: index.html, styles.css App.js	1	2	3
Repository, Readme, Github Einrichtung	1	2	3
Loading-Screen	4	6	8

Lade-Animation	2	4	6
Status-Anzeige	2	4	6
Setup-Screen	6	8	10
Gamepad-Erkennung	4	6	8
Gamepad-Darstellung	8	10	12
Gamepad-Mapping	8	10	12
Piano-Darstellung	8	10	12
Piano-Funktion	8	10	12
Play-Screen	8	12	15
View-Controller	10	15	20
Sound	20	40	60
Synthesizer	20	30	40
Recorder	10	14	18
KI	20	30	40
Test	5	10	15
Bug-Fix	10	15	20
Plakat	2	4	6
Dokumentation	10	15	20
Gesamt	167 Std.	253 Std.	338 Std.

Teamplanung Simon

Wir arbeiten nach mit dem Prinzip der Agilen Softwareentwicklung. Hier verwenden wir das Scrum-Modell. Wir erstellen einen Backlog mit allen Aufgaben. Aus diesem werden für jeden Wochen Sprint die Tasks in die nächste Sprint-Planung aufgenommen. So kann jeder ein Ticket bearbeiten ohne Aufgaben im Vorfeld vergeben zu müssen. Dadurch erwarten wir eine schnelle Entwicklung und frühes Testen der Anwendung. Nach jedem Sprint gibt es eine lauffähige Version der Anwendung.