

Digital Input Computer Keyboard

F.Zeller und E.Hubenschmid

30. Juni 2015

Inhaltsverzeichnis

1	Einleitung	3
1.1	Vorstellung des Projektes	3
1.2	Das Projektmanagement	3
2	Technische Dokumentation	5
2.1	Dokumentation der Storys	5
2.2	Klassendiagramme	5
2.3	Sequenzdiagramme	5
2.4	Beschreibung einiger Methoden	7
3	Benutzerhandbuch	8
3.1	Einleitung	8
3.2	Projekt	8

1 Einleitung

1.1 Vorstellung des Projektes

Unser Projekt das Digital-Input Computer-Keyboard, ist wie der Name schon vermuten lässt ein Digitales Keyboard. Man kann damit mithilfe seines PCs und einer Tastatur Keyboard spielen. Es kann allerdings auch noch mehr. Der Benutzer hat die Möglichkeit verschiedene Samples (Tonspuren die Instrumente darstellen) einzustellen. Somit kann er theoretisch jedes Musikinstrument über seine Tastatur spielen. Außerdem kann der Benutzer gespieltes aufnehmen und abspielen lassen. Somit kann er mit verschiedenen Instrumenten und Tonspuren eigene Lieder zusammensampeln.

Dieses Projekt hat Emanuel und mir sehr gut gefallen, da wir beide ein Musikinstrument spielen und auch allgemein Musik begeistert sind. Ein eigenes kleines Sample-Programm zu gestalten lag also nahe.

1.2 Das Projektmanagement

1.2.1 SCRUMB

1.2.2 GitHub

"Build software better, together."(Motto von GitHub)

Für unsere Projektarbeiten verwendeten wir GitHub. GitHub ist ein webbasierter Hosting-Dienst für Softwareprojekte. Damit Emanuel und ich also bequem von zuhause aus zusammen arbeiten konnten haben wir uns ein Repository in GitHub eingerichtet. Zusammen mit dem Eclipse-Plugin EGit konnten wir unsere Arbeit austauschen und vergleichen. Obwohl es Anfangs Probleme mit der Bedienung und den verschiedenen Funktionen GitHubs und EGits gab, hat es uns doch sehr geholfen und vieles vereinfacht. So konnten wir zum Beispiel gleichzeitig an verschiedenen Problemen arbeiten, indem wir verschiedene Branches (Pfade also Ableger des Projekts) erstellt haben und dann an diesen Branches gearbeitet haben. Sobald dann ein Problem behoben war hat man den Pfad wieder dem Hauptprojekt hinzugefügt und konnte besprechen was genau gemacht wurde und was vielleicht noch verbessert werden muss.

GitHub war uns im allgemeinen eine sehr große Hilfe, aber zu Anfang auch eine große Hürde. Bis wir zurechtkamen mit den Branches, Commits ect. hat es eine Weile gedauert. Man kann aber durchaus behaupten das sich der Aufwand gelohnt hat. Ich würde jedem der ein Softwareprojekt entwickelt empfehlen mit GitHub zu arbeiten, auch wenn er alleine daran arbeitet. Denn EGit zwingt einen dazu seine Änderungen zu dokumentieren und zwischenspeichern. Dies ist zwar etwas nervig, aber man kann sein Projekt immer wieder auf einen beliebigen Standpunkt zurücksetzen wenn etwas komplett schiefgelaufen ist. Dies hat mir oft sehr viel Arbeit erspart.

Gerade jetzt für diese Dokumentation verwenden wir auch GitHub for Windows. Ein Programm für Windows welches einem erlaubt jede Art von Datei über GitHub zu veröffentlichen und zu bearbeiten. GitHub hat uns also sehr viel Arbeit erspart und es uns ermöglicht obwohl wir weit auseinander wohnen ein Projekt auch zusammen zu erarbeiten.

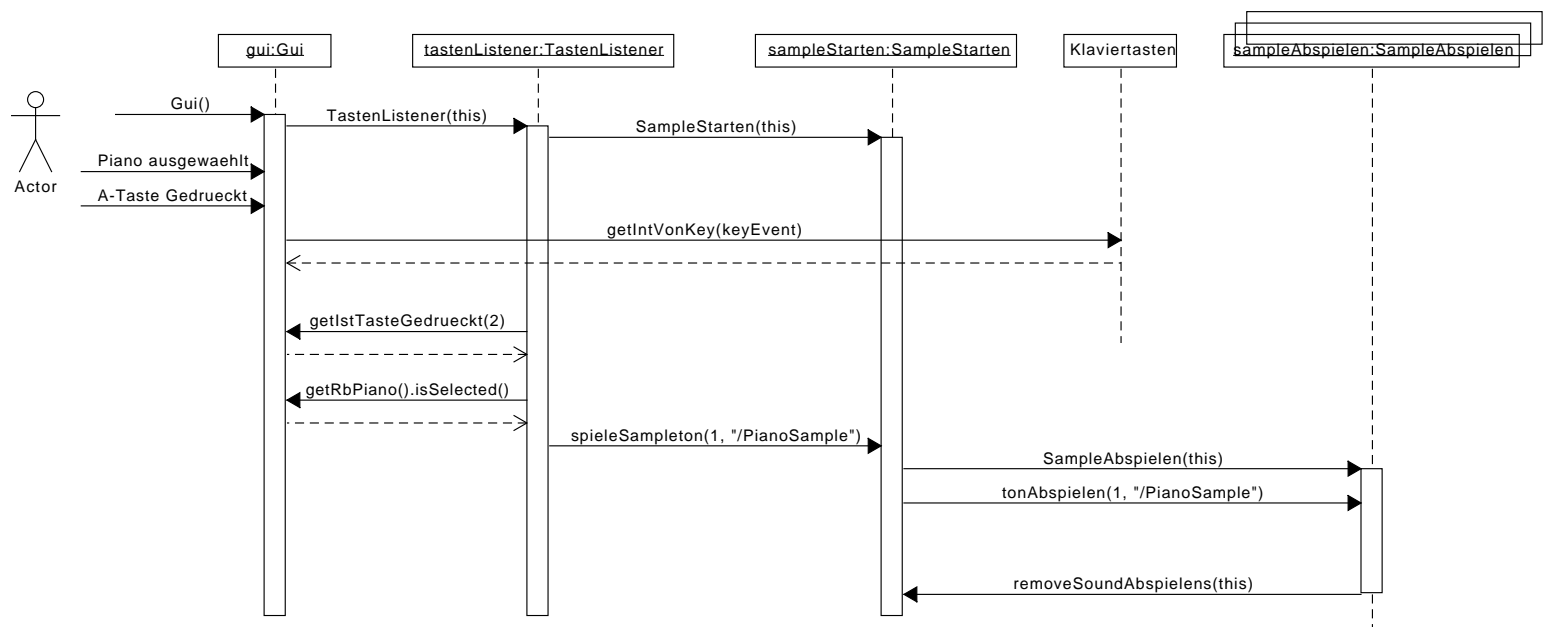
1.2.3 Unser Team

2 Technische Dokumentation

2.1 Dokumentation der Storys

2.2 Klassendiagramme

2.3 Sequenzdiagramme



2.4 Beschreibung einiger Methoden

Wir haben während des Projekts alle Methoden mithilfe von Javadocs dokumentiert. Unser komplettes Project ist also hier einsehbar:

Einige größere Methoden werden wir in dieser Dokumentation allerdings etwas genauer beschreiben.

2.5 Javadocs

Der komplette Quelltext wurde mittels Javadocs auskommentiert. Hier kann nachgelesen werden, welche funktion die einzelnen Methoden erfüllen, sowie ihre Übergabe- und Rückgabeparameter, bzw. Exceptions eingesehen werden.

Die komplette Javadocs Dokumentation kann hier geöffnet werden:

[./JavaDocs/index.html](#)

3 Benutzerhandbuch

3.1 Einleitung

3.2 Projekt

3.2.1 Erstellen

3.2.2 Öffnen

3.2.3 Aufnehmen

3.2.4 Apsielen

3.2.5 Eigene Samples