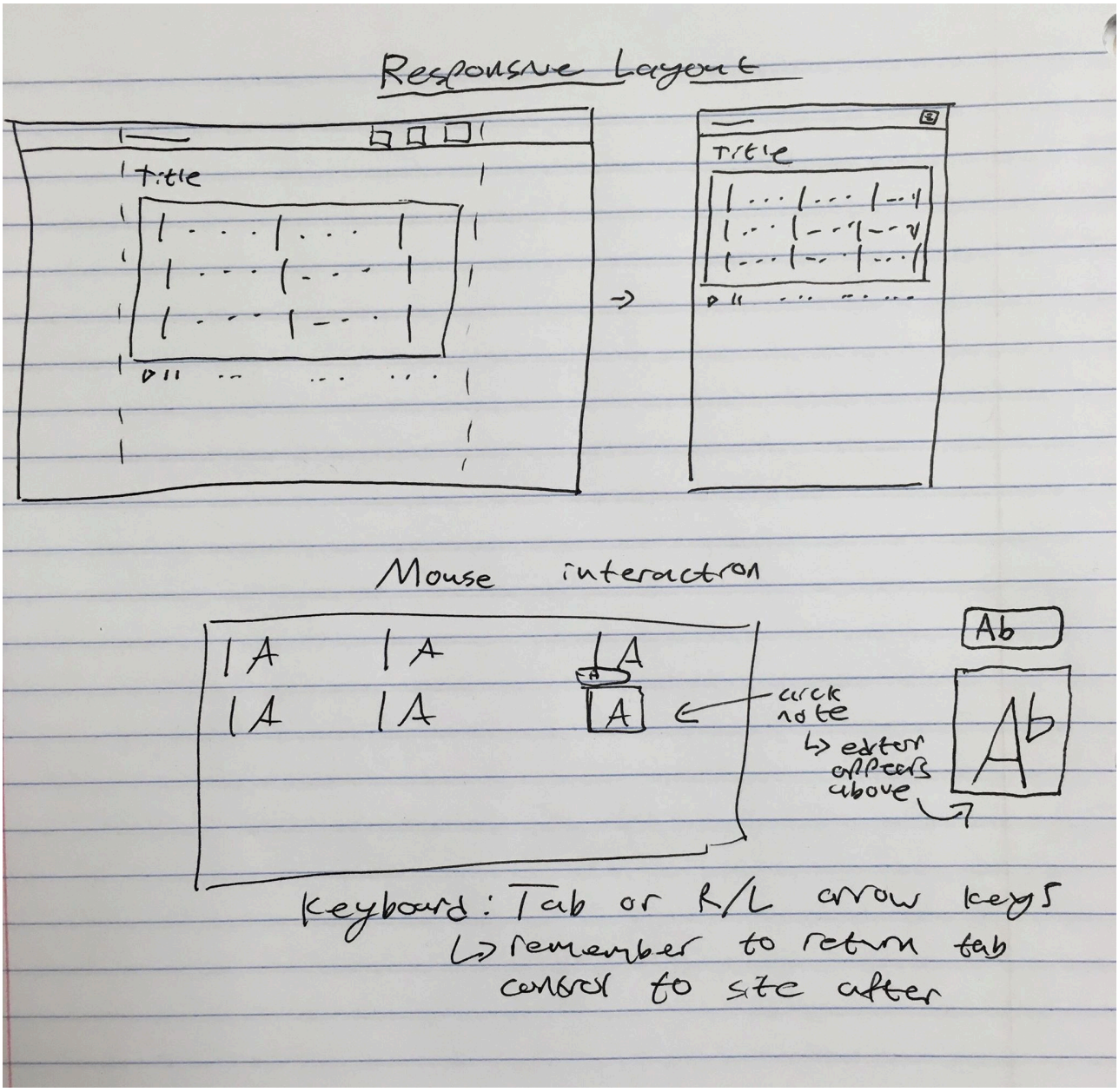
Project 1 Writeup
Jacob Bloom

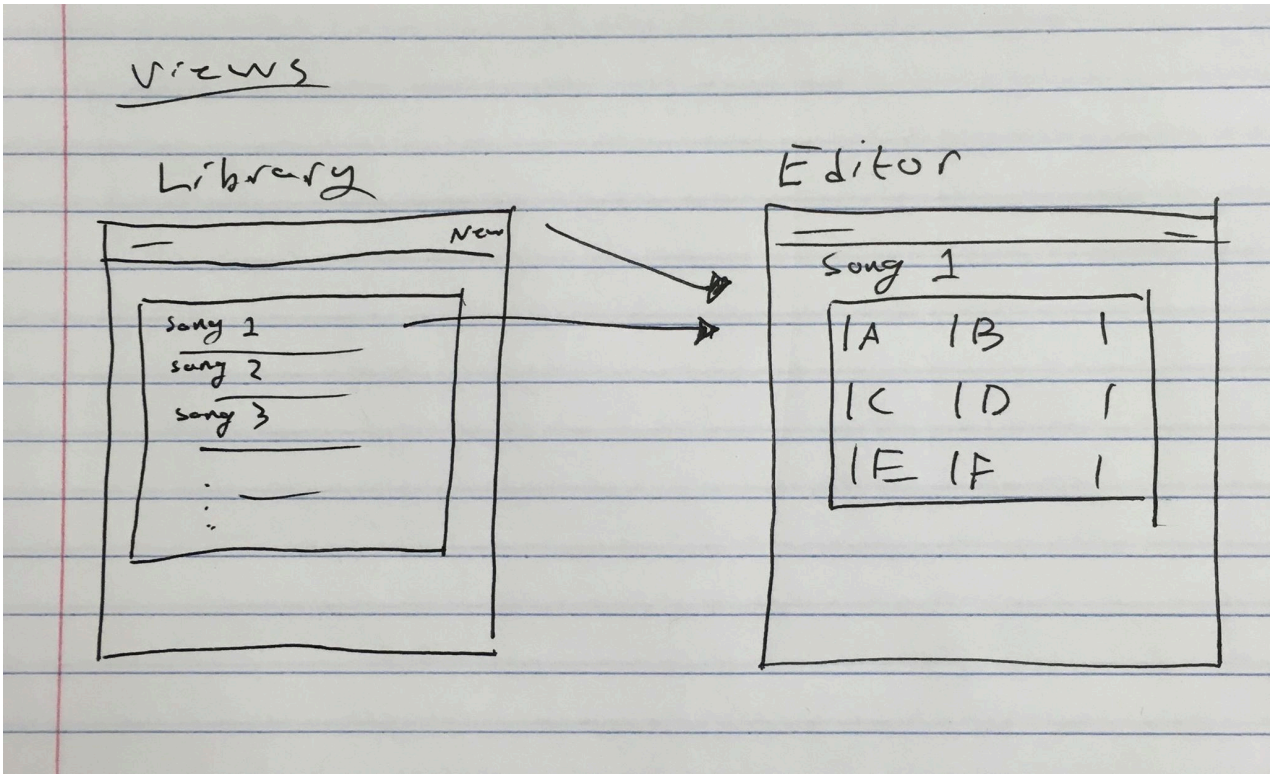
Project Writeup

For this project, I built the GUI wrapper and song library database for a prior long-term passion project of mine. It's a jazz chord player, where you feed it the chords to a jazz song and it does its best to play that song in one of several styles. The rendered "lead sheet" is also navigable (and to an extent editable) using multiple interaction modes (keyboard and mouse) as I'll get to later on.

There's still a long ways to go with this. For example, I set up the song database to save the user's library in browser storage, but I didn't add the ability to modify songs beyond their titles, and the "new song" interface is completely broken right now.

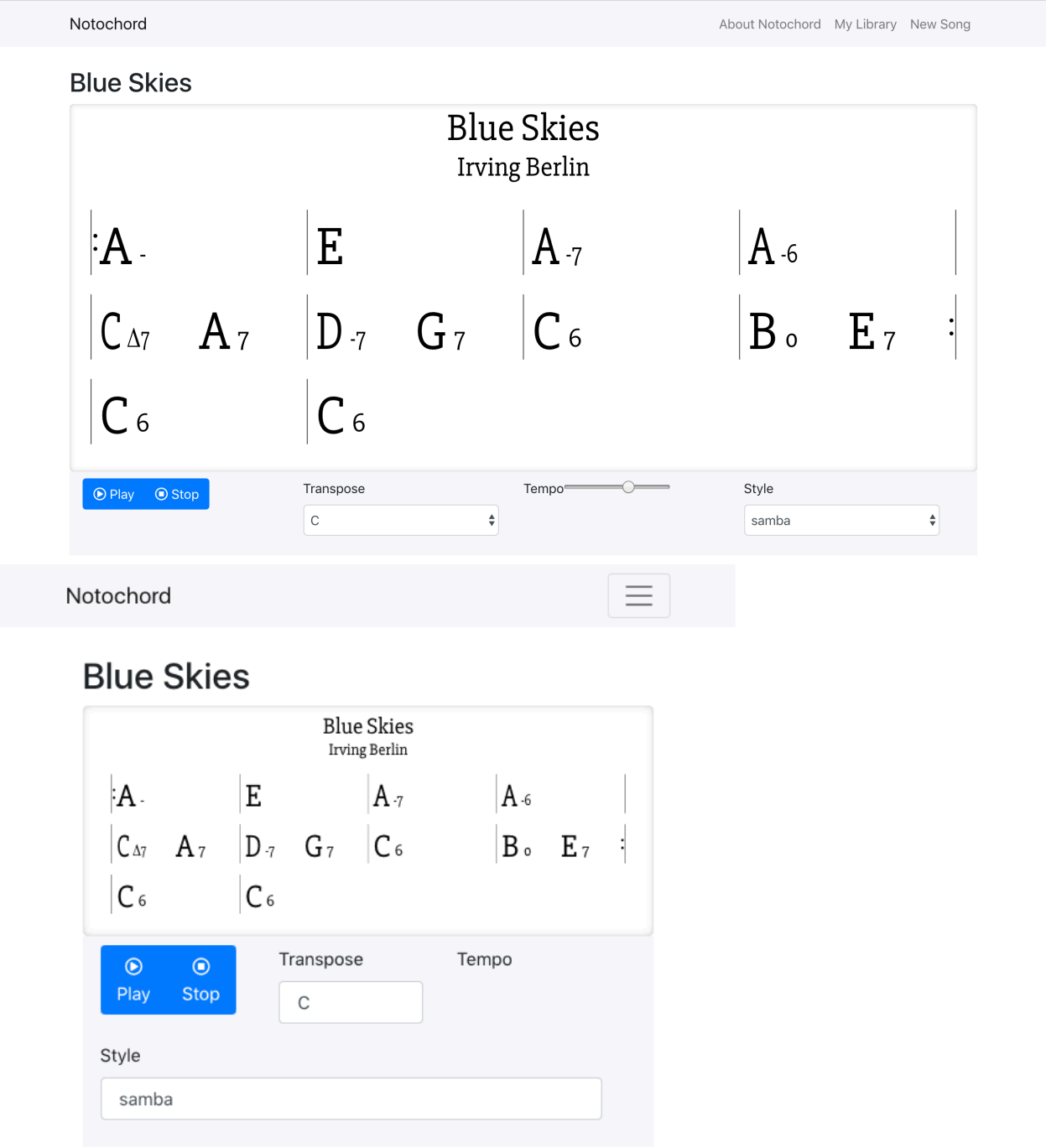
UI Sketches


Responsive Layout
Title
Title
Mouse interaction
A
A
A
A
A
A
click note
editor appears above
Ab
Ab
keyboard: Tab or R/L arrow keys
remember to return tab control to site after

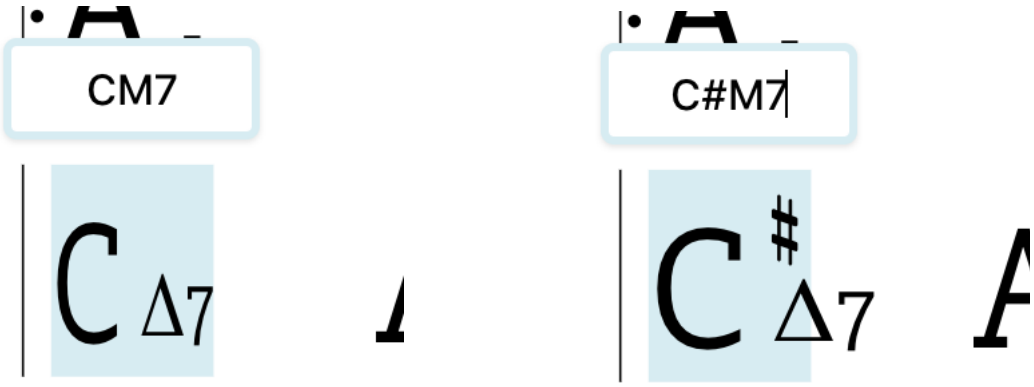


UI Walkthrough

I used Bootstrap to make the GUI responsive. For example, the items on the right side of the navbar automagically turn into a hamburger menu when the screen drops below a certain size. But I did need to make some changes to the player rendering code to make the SVG image scale correctly (specifically, I had to add the “viewbox” attribute):



For the interaction modes, I made sure that the chord editor is tabbable and hands tab control back to the surrounding website when it's done. When you click on a chord, a little editor box pops up above it. Using the left/right keys has one of 2 functionalities: if you're at the beginning/end of the text of the cord, it'll move to the next/previous chord. But if you're inside the text it'll move around within the text. One functionality that still doesn't have a mouse equivalent is that the up/down arrow keys will change the chord by a half-step:



up-arrow ->

...I'm planning to someday flesh out the chord editor with buttons and things instead of just a textbox like we have now.

Demo Link

The deployment service I use shuts down the server if nobody's accessed the site in over half an hour. If you click this link and it doesn't seem like it's loading, that's because the server starting back up -- try again in 5 minutes. Sorry about that.

http://notochord.herokuapp.com

Source Code

Website wrapper:/song database https://github.com/notochord/notochord-gui
Song viewer/player: https://github.com/notochord/notochord

References

To get up-and-running quickly, I used some tools I was already familiar with: create-react-app is a tool to quickly set up a repository to use Facebook's React JS framework. Bootstrap is a set of prebuilt/styled components that make quickly building a wesbite generally easier. Heroku is a service that lets you deploy a website directly from a github repository. Most importantly, there were already tools to tie all these bits together.

I depended a lot on the react-bootstrap documentation: https://react-bootstrap.github.io/getting-started/introduction
I also used this tutorial to get the database up and running: https://developers.google.com/web/ilt/pwa/working-with-indexeddb -- I used Google employee Jake Archibald's "idb" library to work with Javascript's IndexedDB API.

I didn't work too much on the player for this project but its code depends heavily on a couple libraries:
Tonal for the music theory math: https://github.com/danigb/tonal/
Soundfont-player for, well, playing sounds: https://www.npmjs.com/package/soundfont-player