**Synquencer Sprint 1 Retrospective**

Aidan McDonnell, Mitchel Bridwell, Mohini Roplekar, Zachary Heskett

Team 37

# **What went well?**

We were able to complete every user story we planned for the first sprint. We completed the basic layout of the site, including the homepage, about page, and sequencer view. We were able to create a system for generating unique URLs for each sequence. Within the sequencer itself, we completed most of the basic functionality needed for editing MIDI sequences with both mouse and keyboard. Additionally, we created a system for saving sequences to the database from the UI, as well as a system for converting our sequences to MIDI files and saving them to the user's computer. The work we have done in Sprint 1 lays a good foundation for the features we will be working on in future sprints.

**User Story #1**

As a user, I would like to be able to create a new, empty sequence

* Implementing the sequencer UI took up the vast majority of the time for the first sprint, since so many of the other user stories for this sprint were completely reliant on the ability to create a new sequence. With the sequencer UI in a workable state, we will be able to devote much more time to implementing other user stories for the next sprints.

**User Story #2**

As a user, I would like to add a note to the sequence by clicking an empty spot

* This user story is fully implemented both on the frontend and the backend, meaning that added notes are reflected both visually and in the database. The implementation of synchronizing changes to the sequence across different clients simultaneously remains to be done.

**User Story #3**

As a user, I would like to edit or remove a note from the sequence.

* As with adding notes to the sequence, all editing and removal of notes occurs both at the level of the UI and at the level of the underlying database which stores sequence data. This functionality is not at a point where these changes can be synchronized across multiple clients simultaneously; this will be done in future sprints.

**User Story #4**

As a user, I would like to change the velocity of a note.

* This user story was implemented during this sprint, but it has no effects that are visible to the user right now, except when exporting a MIDI, because the application does not have sequence playback functionality right now. This functionality will be implemented during the next sprint.

**User Story #5**

As a user, I would like to change the tempo of the sequence

* Unless the user exports the MIDI, the effects of this user story are not directly visible to the user, since playback functionality has not been implemented. Implementing this functionality, as well as the effects of tempo on playback, is a particularly high priority for the coming sprint.

**User Story #6**

As a user, I would like to lengthen or shorten the duration of a note.

* Users can lengthen or shorten note duration by clicking and dragging on the side of the note. This was tricky to implement since we are using a custom canvas, and building our tool mostly from scratch. We have to do the math based on pixel and note positions to get a properly corresponding position relative to a note.

**User Story #7**

As a user, I would like to move or copy a note somewhere else.

* The moving of the notes was accomplished using dragging, and copying was done using the middle-click function. Both of these systems work perfectly to move notes and paste notes into new positions.

**User Story #8**

As a user, I would like to store my sequences on the server and access them later

* The code for storing and retrieving sequences from the server was fully implemented, but the Fluid framework data structures for sharing sequences between clients was not. The implementation of those data structures has to be implemented in the sprints to follow.

**User Story #9**

As a user, I would like to enter notes using only my keyboard.

* This story, while complete, is not as great as it could have been. In our future user stories that include the keyboard, we will flesh out the keyboard shortcut system to make it easier to use and less cumbersome, as well as increase the overall functionality of the keyboard.

**User Story #10**

As a user, I would like to export my sequences to MIDI files and store them locally

* This part of the application works nearly flawlessly. Once we add instruments to the code base, this will need to be changed a little bit, however. Overall though, the MIDI export works exactly as expected.

**User Story #11**

I would like to change the grid resolution of the piano roll

* This was a story we added relatively late into the project. We replaced a previous story, playback of the sequence, with this one. The grid resolution was a much simpler story to implement with our current code base relative to the previous one, while still providing important functionality to our app. The grid resolution changes how notes are placed, moved and lengthen/shortened.

# **What did not go well?**

## **General Retrospective**

None of the members of our team had ever worked with Next.JS, React, or even Typescript before. This introduced a massive hurdle as all four of us had to learn these tools on our own, on top of developing our application. Although we were able to implement all of our user stories, a significant portion of our time was taken up just by setting up and learning to use the tools we worked with during this sprint. Since we now have a good deal of experience with these tools, we should be able to work more efficiently in the following sprints.

We ended up running short on time during our sprint as we tended to underestimate the difficulty and time commitment required by a lot of our stories. With our current knowledge of the tools we’re working with, we should be able to prevent this from getting as bad as it did in sprint 1. With that said, our team is very proud of the work we’ve put in, and are hopeful for the future of the project.

## **Failed User Story Tasks**

All of our failed tasks are related to testing or creating unit tests for various parts of the project. We needed to create unit tests for user stories 2-7, but we simply didn’t have the time. All members of the group spent well over 10 hours each week on the project, and ultimately we had to prioritize functionality over testing. In the upcoming sprint we will finish the unit tests for these stories (as well as new stories) to ensure the robustness of our application. We are also likely to split unit testing up differently among the group members for the next sprint, so that the entire team contributes to creating unit tests.

**User Story #2**

As a user, I would like to add a note to the sequence by clicking an empty spot.

| 4 | Create unit tests to ensure functionality of note creation functions | 1 hr | Mohini |
| --- | --- | --- | --- |

**User Story #3**

As a user, I would like to edit or remove a note from the sequence.

| 5 | Create unit tests to ensure functionality of note deletion functions | 1 hr | Mohini |
| --- | --- | --- | --- |

**User Story #4**

As a user, I would like to change the velocity of a note.

| 3 | Create unit tests to check the functionality of velocity changing methods | 1 hr | Mitchell |
| --- | --- | --- | --- |

**User Story #5**

As a user, I would like to change the tempo of the sequence

| 3 | Create unit tests to check the functionality of tempo changing methods | 1 hr | Mitchell |
| --- | --- | --- | --- |

**User Story #6**

As a user, I would like to lengthen or shorten the duration of a note.

| 3 | Create unit tests to check the functionality of note length change method | 1 hr | Mitchell |
| --- | --- | --- | --- |

**User Story #7**

As a user, I would like to move or copy a note somewhere else.

| 7 | Create unit tests to check the functionality of time and pitch change methods | 1 hr | Mitchell |
| --- | --- | --- | --- |

# **How can we improve?**

Many of the problems we faced as a team were related to time constraints and ineffective time management. We ended up finishing sprint 1 of our project at 5:00 a.m. the day of the review. This, of course, is not something we would like to repeat in the future.

Because of the work we have done so far on this project, we are much better equipped to understand our capabilities, how long things typically take, and how to go about tackling a new task. With the knowledge we’ve gained from our experiences, we will distribute our tasks more evenly throughout the weeks (assuming this is possible on any given week), as well as give ourselves more time on stories we know are certain to take a long time.

We had a few issues pop into our application due to poor or miscommunication. For instance, our database code did not properly align with the data structures we were using for the Fluid Framework, and instead were modeled off the metadata related to those data structures. In the future, this will cause problems with synchronization between clients and the database. This is something that while technically functional at this point, will need to be fixed in the future sprint.

To prevent something like this from happening again, we will be more diligent about attending meetings at scheduled times. Oftentimes, we would make our meetings incredibly short because we just assumed there was nothing that we needed to talk about. If we spent the time to ask each other questions instead, we could have figured out issues with our code before they arose. For this reason, it is important we improve by using our scheduled meeting times as effectively as we can, and not avoiding any meetings.