

**Team Name:** Content Creators

**Team Members:** Tyler Samuels, Virginia Barnes, Derek Li, Gabriel Wild, Kyle Qin

## Project Summary

Our web application, JamSesh, provides an environment for remote musicians to compose music with version control. The intent is for specified musicians to share a common “studio”, a central location where sheet music is kept for a single composition. Once invited to the studio, musicians can create any piece note-by-note and commit this to the studio. Other musicians invited to that studio will see the committed updates and be able to edit and add more.

For example, two violinists share a studio. ViolinistA composes a piece for the first violin and commits it to the studio. ViolinistB sees the commit, makes a few edits to the first violin, creates a piece for the second violin, then commits this to the studio. ViolinistA doesn't like the edits to the first violin that ViolinistB made, so ViolinistA uses version control to get the first violin's piece back to its previous state. Both violinists are now content with their work and have composed music for both parts.

JamSesh allows for sheet music to be played through computer audio, as well. Continuing with the violinist example, both violinists are able to play the music written for each separate violin or both violins at the same time.

For example, both violinists do not like the quality of the computer-generated audio. With JamSesh, musicians also have the ability to upload their own recordings and access previously uploaded recordings.

## Users/Stakeholders

This web application would be useful for composers, instrumentalists, electronic musicians, producers, and music listeners. Composers can create branches as they edit sheet music, allowing them each to create with extensibility. Instrumentalists can listen to their partners' audios and play along remotely. Users can share projects privately or open work to the public for collaboration on a larger scale. The public project feature would be especially useful for smaller artists who don't mind their music being remixed, and gives people an easier way to create remixes, which could possibly give the original artist more exposure. Producers can receive audios from instrumentalists to work with. Conversion of MIDI files and digital sheet music will allow for smooth collaboration between electronic artists and conventional composers. Music lovers could go and listen to music being developed one commit at a time, as well.

### Stakeholders:

Tyler Samuels, Virginia Barnes, Derek Li, Gabriel Wild, Kyle Qin are members of the Content Creators team. These are the designers and developers of the web application, JamSesh. Each individual's grade in Web Systems Development as well as their feelings of accomplishment depends on JamSesh's success.

Dr. Callahan, Priyanshu Tripathi are members of Web Systems Development teaching staff. These are the individuals overseeing and paving the path for project development. Each individual's feelings of pride in their pupils

rides on the back of the Content Creators' learning and accomplishments.

#### Orchestras/Bands

Given the world's current circumstances, many people are unable to meet in person. JamSesh enables musicians who used to play in orchestras and bands to continue collaborating with other musicians' remotely.

#### Record labels

This site allows labels' artists to remotely collaborate during these times.

#### Talent scouts

Scouts can potentially use the site to see more of a specific artist's work.

#### Universities/Schools

Those pursuing a degree in musical arts must compose music with multiple instruments or with other students to receive credit for their degree. JamSesh will allow musicians to easily layer instruments, collaborate with other musicians remotely, and successfully complete their assignments.

## Technology Stack

Valid semantic markup (HTML5)

Valid CSS (browser prefixes excepted)

Client-side scripting (JavaScript)

Server-side programming (PHP)

Database connectivity (MariaDB)

[MusicXML](#) for representing Western music notation

[Open Sheet Music Display](#) for rendering the sheet music in HTML5's canvas

[OSMD Audio Player](#) for previewing the sheet music

[Music OMR](#) for future use

We plan to build this web application using the AMP stack: an Apache web server, a MariaDB database, and PHP server-side programming. We will also use JavaScript for some additional client-side scripting, improving site optimization. Beyond standard AMP stack elements, we will be making use of the MusicXML file-format for storing and sharing the sheet music. HTML5's canvas element supports the use of MusicXML to render sheet music within the web browser via Open Sheet Music Display and VexFlow. OSMD Audio Player will preview the audio of the sheet music, using something similar to MIDI instruments.

## Functional and Non-functional Requirements

Functional requirements:

- The site will host user files as a repository of their music

- The site will allow users to view other users' uploaded content

The site will allow administrators moderator status on different repositories  
The server will track all changes to existing data  
Users are able to collaborate on pieces of music through actions like commits  
Users can manage their own projects and allow other users to collaborate  
Users will be able to download different versions of project files.  
A separate admin page will be built for admins to log in and manage site data.  
The site will authenticate the existing users and provide registration for new users.

Non-functional requirements:

Commits, branches, and other changes to the repository on the site will be backed using a git server.  
User permissions per project will be tracked in a MariaDB table.  
Users' authentication information and new registration will be stored in a MariaDB table.  
Use MusicXML files to store sheet music.  
Use Open Sheet Music Display, an open source JavaScript engine that makes use of VexFlow to render MusicXML files as sheet music via HTML5's canvas element.  
OSMD Audio Player with Open Sheet Music Display to preview our sheet music as audio.

## Estimated Project Schedule

### Sprint 1: September 17th - September 26th

- Assign group roles
- Identify minimum specifications
- Plan overall execution strategy
- Identify what technologies to use and how

### Sprint 2: September 26th - October 3rd

- Draft designs of each site in the application
- Finalize designs of each site in the application

### Sprint 3: October 3rd - October 10th

- Create the database to store user login information
- Complete the HTML/CSS for the homepage/login
- Connect the database to the homepage/login

### Sprint 4: October 10th - October 17th

- Complete the HTML/CSS for the other sites in the application
- Connect the database to these sites if needed
- Complete the Javascript/PHP for the homepage/login
- *Work on Midterm Presentation (due October 16th)*

### Sprint 5: October 17th - October 28th

- Complete sheet music creation's functionality
  - Javascript, PHP, MusicXML

- *Quiz 1 (October 20th)*

Sprint 6: October 28th - November 4th

- Figure out how to play sheet music's audio
- Implement version control

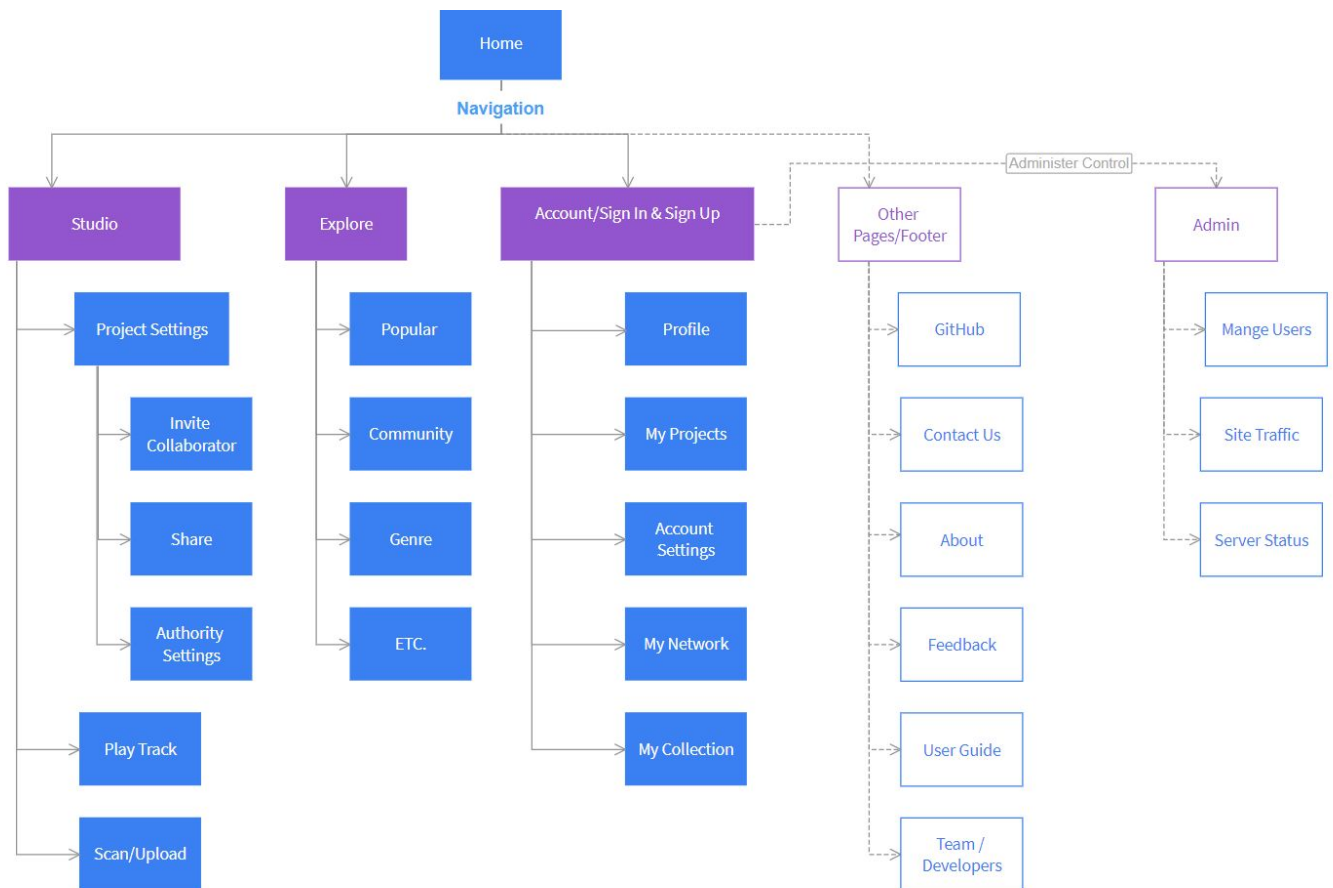
Sprint 7 (Double Sprint): November 4th - November 18th

- Test all of website
- Finalize all HTML, CSS, JavaScript, and PHP

Sprint 8: November 18th - November 24th

- *Prepare for group project previews (November 24)*
- *Prepare for group final presentations (December 1 & 4 & 8)*

## Sitemap



# Wireframe

## Landing Page:

Logo

Why Sign Up | Featured Projects | About | Contact

Search...Go!RegisterSign In

Insert some kind of tagline  
less descriptive than below,  
but more memorable

something like: This is an app where you can  
collaborate with others on sheet music like github.  
You can also post performances of it.

Email

Username

Password

Register

collaborate... something  
something more  
descriptive than above

Snapshot of site

Sample Project Page:

Logo

Why Sign Up | Featured Projects | About | Contact

Search...Go!

RegisterSign In

User: Title

CompositionPerformanceMixesPull RequestSettings

FavoriteFork

SimpleAdvanced

Select File

Select Instrument