

trackstudio

A recording and mixing web application

Glen Anderson
Dominic Cabral
Jose Flores
Ramon Meza

GUI Programming II
Professor Heines

Project Memo 1.0.0
March 8, 2016

Table of contents

Table of contents	2
Review and asses goals	3
New issues	3
MongoDB story	3
API story	4
Working features.....	4
Unimplemented features	5
Contingency plans.....	6
Delivery	7
Resource allocation.....	7
Scrapped aspects	8
Final look and feel	8
Marketing Statement.....	8

Review and asses goals

We have broken down our proposal goals into four sub-goals, we currently believe we have or are on track to meeting all of them

1. Create a simplified sound recording application for modern web browsers.
 - Our application currently runs and has been tested on Google Chrome, a modern web browser. It is coded to seek the audio context that is available to the browser, so other modern browsers can use their version of the Web Audio API that works with them. More testing and validation is needed here but we are on track
2. Make an application that musicians and songwriters, of both genders ages 13 and older, can record demos or music without having to download software.
 - Our application allows for our demographic to be able to record music without having to download software, therefore we have met our second sub goal, Beta version improvements will complete exportation which will allow them to get a copy of their music to submit for their purposes.
3. Make an application that focuses on music rather than the technicalities associated with other sound recording applications through a reduced interface.
 - Our application has a minimalistic interface and we are taking care of all the technicalities in our audio object, music can be recorded and played simply, so this sub-goal has been met.
4. To learn and use the Web Audio API
 - We are all still learning the Web Audio API, this will be a project long goal.

New issues

Through oversite when we originally made our project timeline, we did not include some components that will be essential to storing and exporting user data. During the alpha release we addressed these issues by adding two additional task stories to our application

MongoDB story

1. MongoDB Configuration
 - Our MongoDB runs on a separate machine than our deployment server, this allows us to have non instance dependent data.
2. Mongoose wrapper that connects to our database
 - We created a wrapper script to automatically connect our application to the database through a password protected URI.
3. Make Schemas for our collections
 - We have created a *User* schema to allow for users to be added to the system.

API story

1. Create an API wrapper to interact with MongoDB
 - We created a REST API wrapper that allows the application to communicate with our MongoDB installation.
2. Add API URLs with parameters to the Express application
 - Our API can understand GET parameters within the URL.
3. Document API methods
 - All our methods are self-documenting, they take the information passed to them and create a JSON documentation object that is pushed to an application API documentation object array.
 - The API documentation object is displayed through *trackstudio.herokuapp.com/api/get/help*.

Working features

Our Acceptability criteria is described below, alongside their current status in Table 1.

Table 1: Acceptability criteria and statuses

Feature	Expectation	Status
Mixer – Recording	Application can record sounds on all tracks.	Complete
Mixer – Volume – Track	Adjust the volume of each track.	Complete
Mixer – Volume – Master	Adjust the volume of the Master track.	Capability exists, needs to be connected to interface.
Mixer – Transport – Play	Plays the Master track	Complete
Mixer – Transport – Pause	Pause the Master track	Stop capability exists, need to implement resume.
Mixer – Transport – Stop	Stop the Master track	Complete
Mixer – Transport – Forward	Fast forward the Master track	Not started
Mixer – Transport – Reverse	Rewind the Master track	Not Started

Table 1: Acceptability criteria and statuses (continued)

Feature	Expectation	Status
Mixer – Transport – Time indicator	A display that shows time position of the Master track.	Display exists, needs to be connected to all controls.
Mixer – Panning	A knob that allows panning of sound on individual tracks.	Complete
Mixer – Equalization	A series of knob that allows for the equalization of sound on individual tracks.	Complete
Mixer – Muting	A button that mutes a single track during playback.	Complete
Mixer – Soloing	Toggle mute on all other tracks besides the selected track.	Not Started
Mixer – Track – FX slot	Drag and drop of an effect to a single track.	In progress
FX catalog – Effect – Information	Textual description of an effect	Not Started
FX catalog – Effect – Reverb	Effect used to simulate a large space.	Exists but not yet implemented
FX catalog – Effect – Delay	Effect that produces an echo.	Exists but not yet implemented

Unimplemented features

Our unimplemented feature set is described in Table 2, alongside how we plan to implement them.

Table 2: Unimplemented features

Feature	How we plan to implement
Mixer – Transport – Forward	We can start an audio buffer and start at a specific time, therefore we can skip forward to a given time and resume play.

Table 2: Unimplemented features (continued)

Feature	How we plan to implement
Mixer – Transport – Reverse	We can start an audio buffer and start at a specific time, therefore we can skip backwards to a given time and resume play.
Mixer – Soloing	Mute is in place this is a toggle function amongst all mutes except for the selected track.
Mixer – Track – FX slot	This is a jQuery UI module, therefore capability exists. The feature is in progress but slated to be done during the Beta version.
FX catalog – Effect – Information	These are descriptions of the effects, definitions exist within the proposal. They need to be displayed in the application when the effects are added.

Contingency plans

For any of the tasks that cannot be implemented, the following plan will be put in place. Our plans are described in Table 3.

Table 3: Contingency plans

Feature	Contingency plans
Effects	We will scrap the effects idea resulting in the user only being allowed to record and play audio without any effects.
Exporting	The user will only be allowed to play their audio within the application.
Drop-down Partials	A simple menu will be put in place instead of a drop-down partial.
Effects Slot	A drop-down menu will be implemented instead.
Volume Indicator	A numerical value will be displayed instead of a peak meter.
Read & Write Audio from DB	The user will only be allowed to record and play audio in one session and not be allowed to save or load previous projects.

Delivery

Based on the requirements for our acceptability criteria, we are making great progress. The feasibility of the tasks that are not completed have been stated in Table 2 and our current status can be seen in Figure 1. Currently we are only behind on effects, exporting, and drop down partials. Our *Work in Progress* items (WIP) are partially done, and will be completed by the end of the week but will be now part of our Beta version. We only have one *Scrap* item, this is because it was unnecessary to implement.



Figure 1: Alpha release tasks

Any items that were *Not Started* (NS) can be attributed to having resources allocated to the new issue stories of MongoDB and API being added to our task list. Resources and priority have since been allocated to these NS items, this will be discussed in the *Resource allocation* section of the memo.

Resource allocation

We have allocated the remaining items in the Alpha release to the whole team and prioritized them to be completed first before any items in our Beta schedule are begun. We have given ourselves a deadline of March 13, 2016 to complete these items. This queues all features in the Alpha release before we move on to our next tasks in the Beta release and gives us Spring break to start the Beta version.

Scrapped aspects

As shown in Figure 1, we have decided to scrap SSL certification. This step was originally thought of as necessary to record audio but upon research we found that Heroku gives us this capability. Since our application can be accessed through https, and therefore the task deemed redundant and would have taken time away from development.

Final look and feel

Marketing Statement

Track Studio is a simplified multi-track recording application for the artist who wants to bring their ideas to life. Our application strips away the need for technical knowledge of sound recording and allows you, the artist, to start recording with one simple click! Gone are the days of downloading complicated recording software that can ruin your creative flow. Track Studio requires no installation, only your musical ideas!

For a full feature list reference Table 1 and a current user interface reference Figure 2.

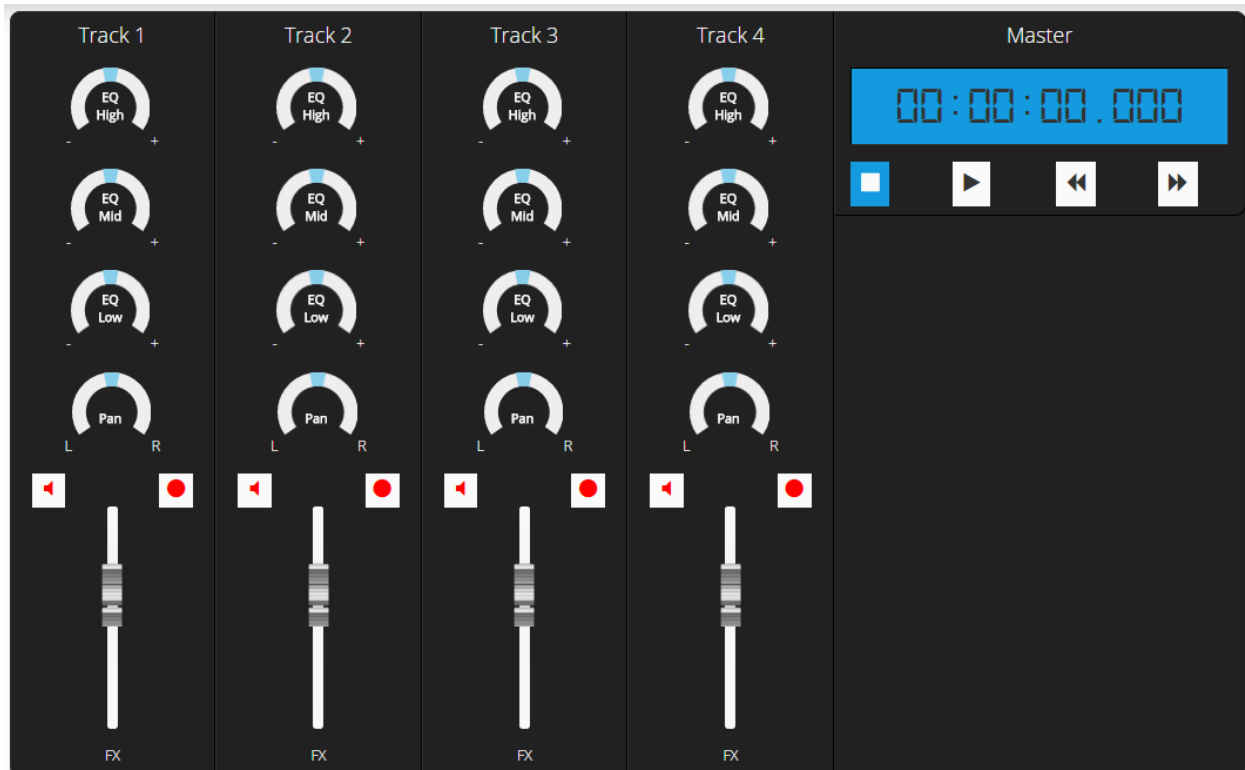


Figure 2: Mixer