

WebApps Project - Music Collaboration

Ashley Hemingway, Charchris Sloan,
David Liskevich, Edwin Kamulegeya
g1327109

May 28, 2014

1 App Description

The main objective of the application is to allow people to easily collaborate with musicians who may have expertise in separate fields. For example a lyricist, may want someone to sing or even remix his lyrics. This application would allow said lyricist to publicise his project, and ask for users with the requirements he needs. So the lyricist may need a sound engineer with expertise in software such as massive, protools or ableton live (software where many producers and djs produce their sounds). This would be done in forums or if you know a particular user, you may ask them to join. Users will be able to add skills to their profile, which would make it easier for someone starting a project to find people with a certain skill set

This application would allow the lyricist to publicise his project, and ask to collaborate with these specialists to produce some music. It can also be an easy to use storage system of sharing music prototypes, for perhaps band members to showcase their sounds to the group.

The actual project will store music, which can be efficiently organised into sections for example 'intro' 'melody' etc. The leader creates the project and is allowed to request for users to join the project. There is a 'group window' which is music accepted by the project leader, which all the group members can listen to. Also there is a personal window so that each group member has a space to upload music to show to the group.

Of course not everyone is going to be a musician, so we also wanted to accommodate album artists and lyricists in the project, where their art and lyrics can also be added, so their skills could also be showcased and critiqued in the group.

Finally users can show off their finished projects, or ongoing projects on their profile pages, so the public can appreciate the music!

2 User Interactions

Our proposed web app has multiple user interactions which are both vital to the functionality of the website and the user experience. Obviously, the main user interaction comes with the ability for users to collaborate and share in order to compose music together. Further features include the ability to share music among users and to other websites (including soundcloud, Facebook and twitter).

3 Implementation Languages

3.1 HTML/CSS

HTML and CSS were an obvious choice. Their popularity is down to the fact that they are the best basis to design a website on. HTML offers all of the functionality required, and combined with CSS to easily change the look of the website, it is there perfect choice. HTML also neatly interacts with JavaScript and PHP which is a massive advantage. Furthermore, the newly updated HTML5 offers even more functionality like 'drag and drop' and embedded audio which are both requirements of the web app.

3.2 JavaScript/jQuery

JavaScript enables us to add better usability to the website while also keeping things simple. It is widely supported and offers a vast array of frameworks to help us, most noticeably jQuery. It will become very useful when implementing such features like form validation and smoothing easing effects to make the website more user friendly. Furthermore, JavaScript is a requirement when using Ajax, which we will discuss later.

3.3 PHP

Although there are many options to code the back-end in like Node, Django and Ruby, we decided to choose PHP. The main reason behind this was that PHP is widely used and it is therefore very stable, with many tutorials and supporting libraries. Since none of us have any experience with developing the back-end of a web app, we decided that this would give us the best learning experience.

3.4 Ajax

Ajax is a necessary component to our web app since it requires content to be pulled from the database once it arrives. This is useful for features like the user chat and the constant updating of the top featured music. There aren't many alternatives to Ajax available, and since our app doesn't require an immediate update (i.e. it can wait 0.5 seconds) it is the best technology at our disposal.

4 Group Structure

As expected, we split the project into the front-end and back-end development. The back-end is mainly handled by Ashley Hemingway, who will be designing the database structure and writing the database interaction code in PHP. Ashley will be partly supported by Charchris Sloan, who will be predominately designing the user interface, but will also help with the back-end when required. The front-end will be developed by David Liskevich, Edwin Kamulegeya and helped by Charchris. Their main job will be to design the webpages and implement them.