

## **Process - Short Sounds**

**CSE 403 Spring**

**4/13/15**

Joel Sigo	sigoj
Justin Yoon	yoony4
Nick Jones	ntjones
Neil Harlow	nharlow
Seth Hampson	shampson
Mattie Carlson	mlc327
John Buscher	jbuscher
Casey Fischer	caseympf

## **Software Toolset**

Since Short Sounds is an Android application, the primary programming language we plan on using is Java. XML will also be used for some front end development. The group will also consider using C++ if we find that java does not contain adequate libraries for manipulating sound, however this would be as a fallback plan only.

After initial talks about the application, there would be no need for data sources in the initial design.

For version control, we plan on using Git, as most group members are familiar with it. Github also has a bug tracker on it, so we plan on using that for our group bug tracking.

Everyone will be coding using Android Studio IDE for consistency. It also has a built in emulator for the group members who wish to test their code, but do not own an android phone to load the application onto.

One major software component we will attempt to use “off the shelf” are Google’s sharing API’s. This will make it simple for clients to share their Short Sounds in their preferred method, whether it be email, Facebook, etc. Another component we do not plan on implementing from scratch is an audio manipulator. As of now, we plan on using the jSyn library for adding effects to sounds. We also plan on using Android’s built-in audio recording functionality.

## **Group Dynamics**

Everyone's roles have been divided to front end, back end, and full stack so far. The specifics of each are still being determined, and will be divided when the team has a better idea exactly how the project will be partitioned.

Nick is currently the project manager. He will also be working full stack so he can understand every aspect of the project. Neil, Justin, Mattie, and John will work on the back end, Joel will take the lead on the front end. Casey and Seth will work full stack, wherever they are needed. Testing will be handled by the developers themselves, whether they wish to use test driven development, or if they decided to write tests after implementation is left up to them.

The roles have been chosen based on everyone's interests with the project. They are also split up in such a way that the group will have multiple people tackling the difficult backend side of this project. Documentation will be divided up between team members.

Communication will be handled primarily through a group Facebook page, monitored by the project manager.

Disagreements will be settled through voting. In the event of a tie, Nick is the tie breaker.

### **Schedule / Timeline:**

By the alpha release, some of the main features should be working, although maybe not quite finished yet. That means that the application should be able to record a sound by this point. The recording of a sound is a key feature that all of the other features rely on, so it is essential that this is the first feature that is completed. The ability to record a sound is used in a lot of applications, and we do not expect this task to be exceptionally difficult. At this point, ShortSounds should also be able to save a sound recorded by the user. The ability to save a file is a stepping stone on the way to sharing a file, so it is also important that this feature get implemented early, paving the way for the sharing feature.

By the beta release, the main features should all be implemented. This means that ShortSounds should be able to layer different sound recordings on top of each other in order to create a new sound. The sound manipulation should involve changing the volume as well as at least one or two other effects. The sound manipulation will be the most difficult feature to implement, so it is scheduled to be complete by the beta release. This gives us a bit more time to work on it as opposed to scheduling it to be completed by the alpha release, but it also leaves a buffer before the feature complete release in case the sound manipulation takes longer than anticipated. ShortSounds should now be able to save a sound to its library and

share a sound with another person. The sharing feature should not be too much of a stretch after we are able to save a file, so it is reasonable to expect that this can be complete by the beta release.

### **Risk Summary:**

One major risk for this project is that we will be working with audio files and most people in our group are pretty unfamiliar with this. We don't know which libraries we should use, and whether or not they are capable of the sort of sound blending and manipulation that we need in order to create ShortSounds. To mitigate this risk, one of the first things our group will do is research different libraries that are available for manipulating sound in order to determine the best possible way to implement ShortSounds. Once a library (or libraries) has been found to help with this task, we will make small test programs that test the different sound manipulation that ShortSounds will have to be capable of performing, including blending two sounds and adding effects to sounds.

In order for ShortSounds to be successful, it has to have an intuitive design that will appeal to beginners. This is a major risk because the application does have a lot of complicated features and we need to make sure that we present them in a way that is simple and easy to use. In order to make sure that we are responding to this risk properly, we are going to make paper prototypes and get user feedback before we decide on a final design. In addition, we will try to get a prototype finished as soon as possible in order to get feedback on the user interface and usability. Because we want to make an application that is intuitive to use, the most helpful times to receive user feedback would be after creating paper prototypes, and then again after we create working prototypes of the application. We will get feedback by asking people who are unfamiliar with the design to try to use the application. We will proceed to observe how they use the application and ask them questions about their experience afterwards.

The last major risk that our group faces in working on ShortSounds is the fact that only one of us is familiar with Android development. Since we only have a short time to work on the project it is crucial that we all become familiar with how to make an Android application quickly. Thankfully, there is a lot of really in depth documentation on how to make an Android application, so group members will be expected to read this documentation. In addition, since Nick has already done some Android development, he will be crucial in helping everyone else

get started up. Lastly, we will try to set up a skeleton to work on in the beginning, so we will not waste as much time individually trying to figure out how to structure the code.

We are most worried about not being able to find a library to manipulate and blend the sounds as we envision because that is the most crucial aspect of the application. If we are unable to find a way to blend sounds together, we will have to switch to just having a single sound that can have effects added to it. If we are having a hard time adding effects, we might have to resort to having less special effects than we had previously envisioned on the final application. Another option would be to check out other languages besides Java to see if we can find the sound manipulation tools that we need.