Caitlin Crowley

Final Project Report

CS 443

My Android application features a soundboard that uses pre-recorded sounds that are assigned to specific buttons. When pressing one of the buttons, a sound is emitted much like a musical instrument such as a keyboard or piano.

I wanted to develop a soundboard because I was inspired by the creative and visual side of application building. I myself am an artist and come from a family of artists that inspire me to be creative and in turn I wanted to inspire the user as well. I want to push the user into using their imagination and making something they wouldn’t normally think to create. I chose a soundboard specifically because music is an art that is easy for everyone. I think that music is a versatile art form and it doesn’t matter where the user is from, if they’re young or old, a beginner or an experienced musician. A soundboard can be just for fun, to pass the time, or it can be used professionally by musicians. It’s a simple way to create music on the go and the interface guarantees that even the most inexperienced can enjoy creating a tune.

My soundboard application isn’t the only application of its type available in the Android Market. There are hundreds of soundboards if not thousands available just for Android alone. What is interesting is that there are different types of soundboards. Some are for comedic purposes or sound effects, and some are more professional like sound mixers. My application more closely resembles an electronic tool that a DJ would use called a launchpad. A launchpad’s buttons are programmed to hold sound clips much like a soundboard. A DJ uses this tool at every concert and performance and could be considered an essential device for a proper musical show.

Much like a launchpad, my application’s modest interface ensures that the user can smoothly and easily navigate. My design features custom buttons that are programmed with predetermined sounds from various musical instruments. Each button is labeled so the user is easily able to identify which button corresponds to what sound. These labels are especially important for my design choice. This is because the homepage itself looks like a keyboard where the buttons themselves are seamless. I think this little musical flair makes my application stand out from the others.

The main focus of my design was to make an application that would be visually appealing, user friendly and most importantly simple. This is because having too many buttons could potentially be overwhelming visually and confusing to some. I wanted to make sure the application appealed to all audiences and wouldn’t discourage aspiring musicians.

My soundboard design also features share and save capabilities that are easily accessible at the click of a button. By holding down one of the sounds the user can quickly access a drop-down menu. This menu features two options share as well as save.

When clicking the option “Save” a pop-up window will appear. The new options available will be to save as a ringtone, an alarm, or a notification alert. After clicking the appropriate category, the clip will become an available sound option. For example, when choosing “Alarm” the user can then set the desired sound to an alarm time through the drop-down menu in the Clock application.

The “Share” button is different from the “Save” button in that the user will be prompted to pick a specific application. At the top of the screen a message will appear saying to share the clip using any media sharing applications such as Snapchat or Messenger. These options will appear directly on the screen as icons. After clicking, the new application will appear with the sound clip ready to be shared with friends or family.

My soundboard application runs well, but there is one error that I could not solve. Currently, when the sound clips play they will not play together and must be played separately. I was able to make them play at the same time, but after a certain number of clicks the buttons would stop emitting sound. This was frustrating because I wanted to be able to seamlessly transition from one sound to another.

To play sound clips I used MediaPlayer. MediaPlayer is then handled by my EventHandler class where it must start( ) and release( ). The sounds are then paired with the buttons using my SoundObject[ ] in my SoundBoardActivity class.

In order to save and share I had to get permissions. Ask permissions to write and read external storage using ActivityCompat. This is used to share a sound clip. In order to access the users settings I used a Snackbar that displays an activity that prompts the user to grant permission. This is done by an Intent.

Save and share was accomplished by utilizing longclick. This longclick triggers a PopupMenu. The setOnMenuItemClickListener waits for the user to choose between save or share. Using InputStream and OutputStream it captures the song in a buffer and then to a file in external storage.

If the user chooses to share, then the application starts a new chooser dialog where the user can choose an application. If the user chooses to save, then an AlertDialog will appear as a new window with three options. After clicking the desired option all the information about the audio file is placed into ContentValues called values. If the option clicked was a ringtone then using a switch state I values.put(MediaStore.Audio.Media.IS\_RINGTONE, true), for notification IS\_NOTIFICATION, for alarm IS\_ALARM). Finally, I set the audio as one of the system types using RingtoneManager, for alarm RingtoneManager.setActualDefaultRingtoneUri(view.getContext(), RingtoneManager.TYPE\_ALARM, finalUri).

My design was supposed to feature a favorites option located on the toolbar. To favorite a clip the user would pull up the save and share menu where favorite would be the last option. At the top of the application on the toolbar itself would be an icon that the user would be able to click to bring them to a new page. The specified audio clip, icon, and label would now be under this list as well as the remaining in the original list.

If I had more time I would have also liked to implement a recording system. The recorder would capture the sounds played by the user on the soundboard to make an actual beat. This recorder would be located below the toolbar above the buttons. The user would then be able play and stop the recording as well as save, share, and favorite. The option to play and stop would be located next to the record button with a drop-down menu featuring save, share, and favorite. I also wanted to record from the microphone in order to utilize voice. That would mean that lyrics could be incorporated as well. This proved to be an immense task because the recorder would have to be able to capture the sound clips as well as voice. I wasn’t sure if I would be able to do both at the same time, so this plan was put aside.

Finally, I wanted the user to be able to save their own audio clips to empty buttons. These custom buttons like the favorite action would be on a separate page. Using the built-in sounds users could record then save their beat to a button. These buttons would have the all same functions as the other buttons like sharing with friends. The music clip would be stored in the soundboard and could then be used again in another beat or song.