The sounds for the app are in the “Edited Sounds” folder. I’ve included most of the raw sounds in their own folder in case we need to go in and create a new edit.

The folders are arranged according to the screens in the app. The notable exception is the “Generic” folder, which holds the few sounds that apply to buttons and elements that appear on multiple screens.

0. Generic

**sfx\_digital-button.wav**

This sound is used on all “on screen” selections. This includes tapping on the planets, and any “digital” buttons. The first click should begin as close to when the user puts their finger on the screen as possible.

**sfx\_digital-typing.wav**

This sound is used whenever text is being typed out. The sound should be cut off as soon as the text stops. This sound loops well.

**sfx\_error.wav**

This sound plays on the rover build screen when the user attempts to place a tool on a port where it doesn’t fit. The click should begin as close to when the user removes their finger from the screen as possible. It also plays on the navigation screen when the user attempts to enter a command that has the rover running into an obstacle. The sound should begin as close to when the user puts their finger on the screen as possible.

**sfx\_physical-button-down.wav**

This sound is used on all “physical” buttons. This includes the start button, exit “X” button, rover build buttons, and navigation buttons. The click should begin as close to when the user puts their finger on the screen as possible.

**sfx\_physical-button-up.wav**

This sound follows the above sound as soon as the user removes their finger from the screen.

1. Title Screen

**mus\_intro.wav**

This sound plays on a loop in the background.

2. Planet Select Screen

**amb\_outerspace.wav**

This sound plays on a loop in the background.

3. Rover Build Screen

**amb\_workshop.wav**

This sound plays on a loop in the background.

**sfx\_clickwheel-1.wav** through **sfx\_clickwheel-9.wav**

This one is a little tricky. The objective is to make the clickwheel sound like a winding clock as you drag the wheel or the tools onscreen from left to right (or right to left). I pulled a sound of a winding clock that had the clicks rise in tone from the 1st click to the 9th click. I chopped up the audio file into nine parts so that if a user were to drag their finger slowly, the clicks would play slowly and if they were to drag their finger fast they would play fast. From a programming perspective, I would imagine that you would divide the clickwheel and the touchscreen into nine parts respectively. As you drag your finger, each subsequent click plays as you move past 1/9th of the field.

**sfx\_tool-drag\_pickup.wav**

This sound plays when the user picks up a tool from the selection bin or off of the rover. The sound should begin as close to when the user puts their finger on the screen as possible.

**sfx\_tool-drag\_equip.wav**

This sound plays when the user places a tool on the rover. The sound should begin as close to when the user removes their finger from the screen as possible.

**sfx\_tool-drag\_return.wav**

This sound plays when the user drops a tool back into the selection bin. The sound should begin as close to when the user removes their finger from the screen as possible.

4. Journey Animation

NOTE: It may make more sense to mix and “flatten” these sounds into a single audio file in order to have them correspond to the animation on screen. We may need to adjust the animation slightly to fit the pace of the audio files and vice versa.

5. Navigation Screen

**amb\_control-room.wav**

This sound plays on a loop in the background.

**sfx\_hovercraft\_start.wav** and **sfx\_hovercraft\_loop.wav** and **sfx\_hovercraft\_end.wav**

These sounds play whenever a hovercraft-equipped rover is moving. The start and end snippets are distinct from the main looping section. Depending on how long the rover is moving, you should be able to loop and cut off the center piece of audio so that the entire string of three files equals the amount of time you need. If that doesn’t work well, let me know if there’s a better way to chop up the files to achieve the same effect.

**sfx\_wheels-and-treads\_start.wav** and **sfx\_wheels-and-treads\_loop.wav** and **sfx\_wheels-and-treads\_end.wav**

These sounds play whenever a wheels- or treads-equipped rover is moving. The snippets operate similarly to the hovercraft version, however the start and the end sounds are not particularly distinct. Instead, they are just the same noise faded in and out. If it’s easier to just use the center piece of audio and fade it in and out using the software, feel free to do so.

**sfx\_low-battery.wav**

This sound plays three times when the rover reaches 25% battery remaining. The chimes should line up with the flashes of the indicator bar. The bar should continue to flash without the chimes until you have only 5% battery remaining, at which point it should play along with the flashes until the end of the game.

**sfx\_robot-arm.wav**

This sound plays when the rover activates the robot arm in order to move an obstacle out of its way. The action and sound should take place one or two beats after the rover comes to a complete stop in front of the obstacle.

6. Victory Screens

**sfx\_failure\_battery-drained.wav**

This sound plays when the battery drained “victory” screen pops up. India/Jacob’s text will type in afterwards.

**sfx\_failure\_wopwop.wav**

This sound plays one or two beats after India/Jacob’s text finishes typing out on the battery drained “victory” screen.

**sfx\_moon\_thermometer.wav**

This sound plays when the thermometer victory screen pops up. We may need to adjust the sound file to line up with the animation or vice versa. India/Jacob’s text will type in afterwards.

**sfx\_mars\_rock-grinder.wav**

This sound plays when the rock grinder victory screen pops up. We may need to adjust the sound file to line up with the animation or vice versa. India/Jacob’s text will type in afterwards.

**sfx\_titan\_laser-drill.wav**

This sound plays when the laser drill victory screen pops up. We may need to adjust the sound file to line up with the animation or vice versa. India/Jacob’s text will type in afterwards.

**sfx\_pluto\_camera-servo.wav**

This sound plays when the camera victory screen pops up. The flash of the camera should line up with the shutter sound of in the audio file. India/Jacob’s text will type in afterwards.

**sfx\_success\_chime.wav**

This sound plays one or two beats after India/Jacob’s text finishes typing out on the final (aka 3rd) victory screens.

7. Miscellaneous

**sfx\_transmission.wav**

This sound plays on loop on the transmission screen.