

Preface: This project was built using a programming language called Pure Data, which differs from traditional programming languages like C++ and Java since it is entirely a visual programming language. Pure Data was specifically created to allow for sound design. As a result, most of the input and output for the project involves taking in audio from some form of listening device and then manipulating the input to create different audio waves as output. Since we used a microphone to get our input, the incoming audio waves cannot be precisely defined. This means that the output is somewhat ambiguous. We got around this fact by examining our output manually and determining if our module plays back the sound we expected to hear. This means that traditional unit testing was not feasible. Instead, we isolated each portion of the project and then manually tested it by checking that manipulations were working and using our ears to check the output.

Sprint 1: System testing was not deemed valuable or applicable, since sprint one was dedicated entirely to education and did not have any deliverable products.

Sprint 2:

1. As a guitarist, I would like an assortment of effects to work with. (Delay)
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in puredata preferences.
 - c. If you are using a mic as opposed to a guitar, be sure to use headphones so you don't end up with a nasty feedback loop.
 - d. Start delay.pd either with the pd gui, or from the command line with 'pd delay.pd'.
 - e. Play with parameters and ensure that the output sounds like an echo.
2. As a guitarist, I would like an assortment of effects to work with. (Wah)
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in puredata preferences.
 - c. If you are using a mic as opposed to a guitar, be sure to use headphones so you don't end up with a nasty feedback loop.
 - d. Start wah.pd either with the pd gui, or from the command line with 'pd wah.pd'.
 - e. Play with parameters and ensure that the output sounds correct. It should sound like an oscillating 'auto wah' effect. The name is onomatopoeic.
3. As a guitarist, I would like an assortment of effects to work with. (Distortion)
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in puredata preferences.
 - c. If you are using a mic as opposed to a guitar, be sure to use headphones so you don't end up with a nasty feedback loop.
 - d. Start distortion.pd either with the pd gui, or from the command line with 'pd distortion.pd'.
 - e. The output should sound like a harsh audio clipping such as when you turn the volume too high on a digital device.
4. As a guitarist, I would like an assortment of effects to work with. (Tremolo)
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in puredata preferences.

- c. If you are using a mic as opposed to a guitar, be sure to use headphones so you don't end up with a nasty feedback loop.
 - d. Start tremolo.pd either with the pd gui, or from the command line with 'pd tremolo.pd'.
 - e. Play with parameters and ensure that the output sounds like there is a stutter.
- 5. As a guitarist, I would like an assortment of effects to work with. (Vibrato)
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in puredata preferences.
 - c. If you are using a mic as opposed to a guitar, be sure to use headphones so you don't end up with a nasty feedback loop.
 - d. Start vibrato.pd either with the pd gui, or from the command line with 'pd vibrato.pd'.
 - e. Play with parameters and ensure that the output is experiencing fluctuations in pitch.
- 6. As a guitarist, I would like an assortment of effects to work with. (Volume expression)
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in puredata preferences.
 - c. If you are using a mic as opposed to a guitar, be sure to use headphones so you don't end up with a nasty feedback loop.
 - d. Start volume.pd either with the pd gui, or from the command line with 'pd volume.pd'.
 - e. The parameters are a simple level control. It should just control the volume.
- 7. As a guitarist I would like to be able to control these effects with my feet.
 - a. Plug in external sound source
 - b. Plug in external midi device
 - c. Ensure that sound source and the midi-in sources are correct in puredata preferences.
 - d. Start delay.pd either with the pd gui, or from the command line with 'pd volume.pd'.
 - e. When you use the external midi device to send the midi cc signal number 22 with a control value between 0 and 127 the volume of the output should adjust accordingly.

Sprint 3 (Focused partly on infrastructure, hence lack of effects):

- 1. As a guitarist, I would like an assortment of effects to work with. (Looper)
 - a. Plug in external sound source
 - b. Ensure that sound source is correct in Pure Data preferences
 - c. Start looper.pd either with the pd GUI, or from the command line with 'pd looper.pd'
 - d. Record sound for up to ten seconds by toggling a button to the "on" state
 - e. Disable the recording toggle and verify that the sound plays back on a loop and is accurate to the given sound input (i.e. does it play back at the same rate, is it at the right pitch, etc.)

- f. Press the clear button and verify that the loop is cleared.
2. As a guitarist, I would like an assortment of effects to work with. (Phaser)
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in Pure Data preferences.
 - c. If you are using a mic as opposed to a guitar, be sure to use headphones so you don't end up with a nasty feedback loop.
 - d. Start phaser.pd either with the pd GUI, or from the command line with 'pd phaser.pd'.
 - e. Play with parameters and verify that the output sounds like it's being put through a phaser effect (sound is a bit more "wavy", kinda like surf rock).
 - f. Play with parameters further and verify that the output is modified as expected.
3. As a guitarist, I would like an assortment of effects to work with. (Depth in Tremolo)
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in Pure Data preferences.
 - c. If you are using a mic as opposed to a guitar, be sure to use headphones so you don't end up with a nasty feedback loop.
 - d. Start tremolo-2.pd either with the pd gui, or from the command line with 'pd tremolo-2.pd'.
 - e. Play with parameters to confirm that the level in volume of the effect changes as one of the parameters is manipulated.
 - f. Play with parameters further and verify that the output is modified as expected.

Sprint 4:

1. As a guitarist, I would like to be able to customize the parameters of effect with my feet.
 - a. Plug in external sound source.
 - b. Ensure that sound source is correct in puredata preferences.
 - c. If you are using a microphone as opposed to a guitar, be sure to use headphones so you don't end up with a feedback loop.
 - d. Plug in a guitar effects midi controller.
 - e. Start main.pd either with the pd gui or from the command line with 'pd main.pd'
 - f. Use an effect selector on the midi controller to switch between each audio effect. The effects are hard code in as midi control change values between 20 and 29.
 - g. Once an effect is selected, use the control change expression pedals on the midi controller to manipulate the parameters for the effects.
 - h. After performing a manipulation listen to see if the audio has change in correlation to the change in parameters. Most parameters involve the speed or depth of the effect. By testing the effect with the min and max of each parameter a clear change in audio output should be expected.