MusiWrite

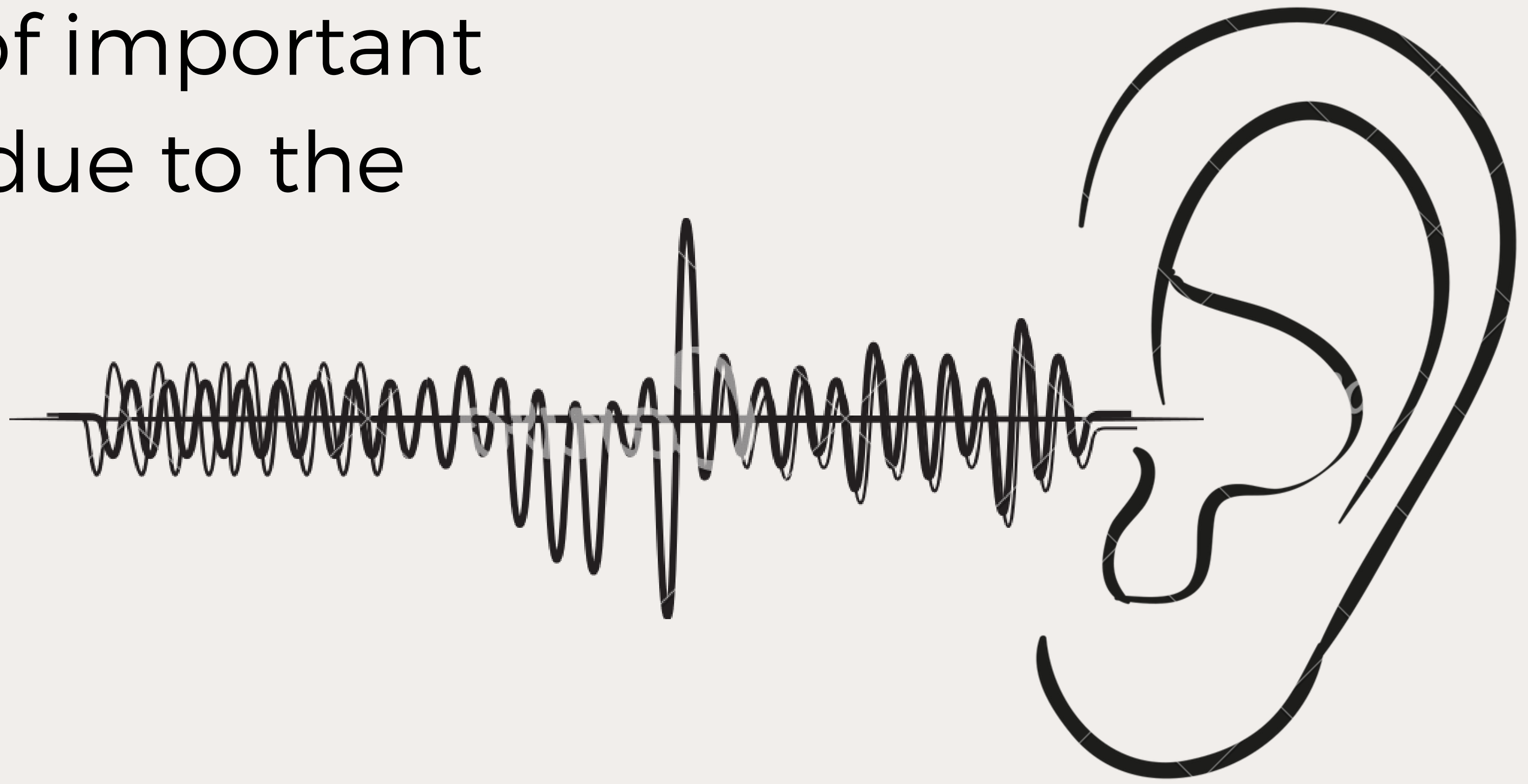
→

RONIT AVADHUTA
DAIWIK PAL
ALINA SHKURIKHINA

Motivation

- People who are hard of hearing are unable to fully experience music because of their lower audio sensory ability

- Many of these individuals are left out of important parts of their culture and modern life due to the disability



PEOPLE PLACED THE ABILITY TO **HEAR PEOPLE INDIVIDUALLY**, [1] IN A **GROUP**, [2] AND **LISTEN TO MUSIC** [3] WITHIN THEIR TOP THREE WANTS

76%

OF PARTICIPANTS BELIEVED THAT *VISUALS ENHANCED USER EXPERIENCE*

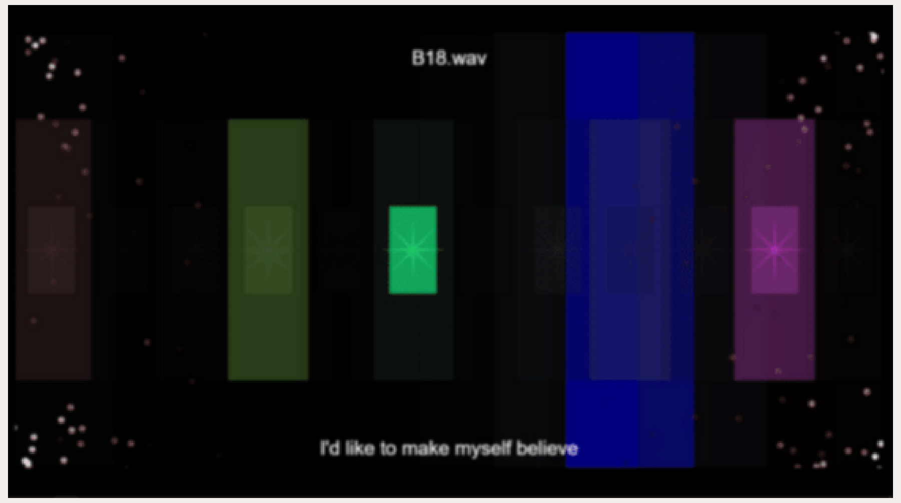
Target Audience

PARTIALLY DEAF AND DEAF INDIVIDUALS
RANGING FROM CHILDREN TO ELDERS
WHO ARE INTERESTED IN MUSIC

Competitors

ViTune

→



aims to enhance the music experience using a piano scroll effect that complements the music

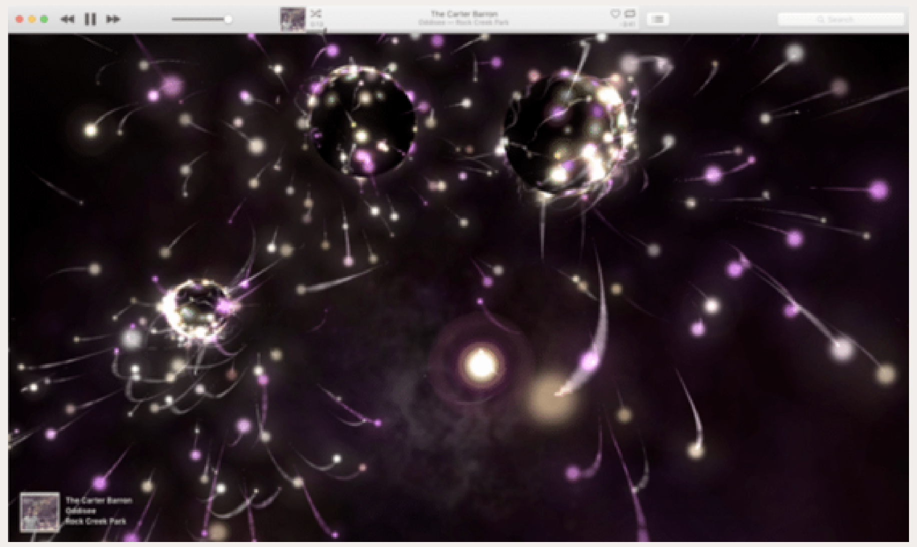
Flaws:

- no ability to search music
- can not save music and create playlists
- evokes little emotion from users

iTunes Visualization

creates an aesthetically pleasing visualization that resembles electrical particles correlated to the audio

←



Flaws:

- little information on how visualization functions
- does not perform well in translating the song's emotion visually

Description of Solution

Song Selection

The App provides the user with a list of songs to choose from (prepared MIDI files) and will eventually be able to take in audio from a MP3 file

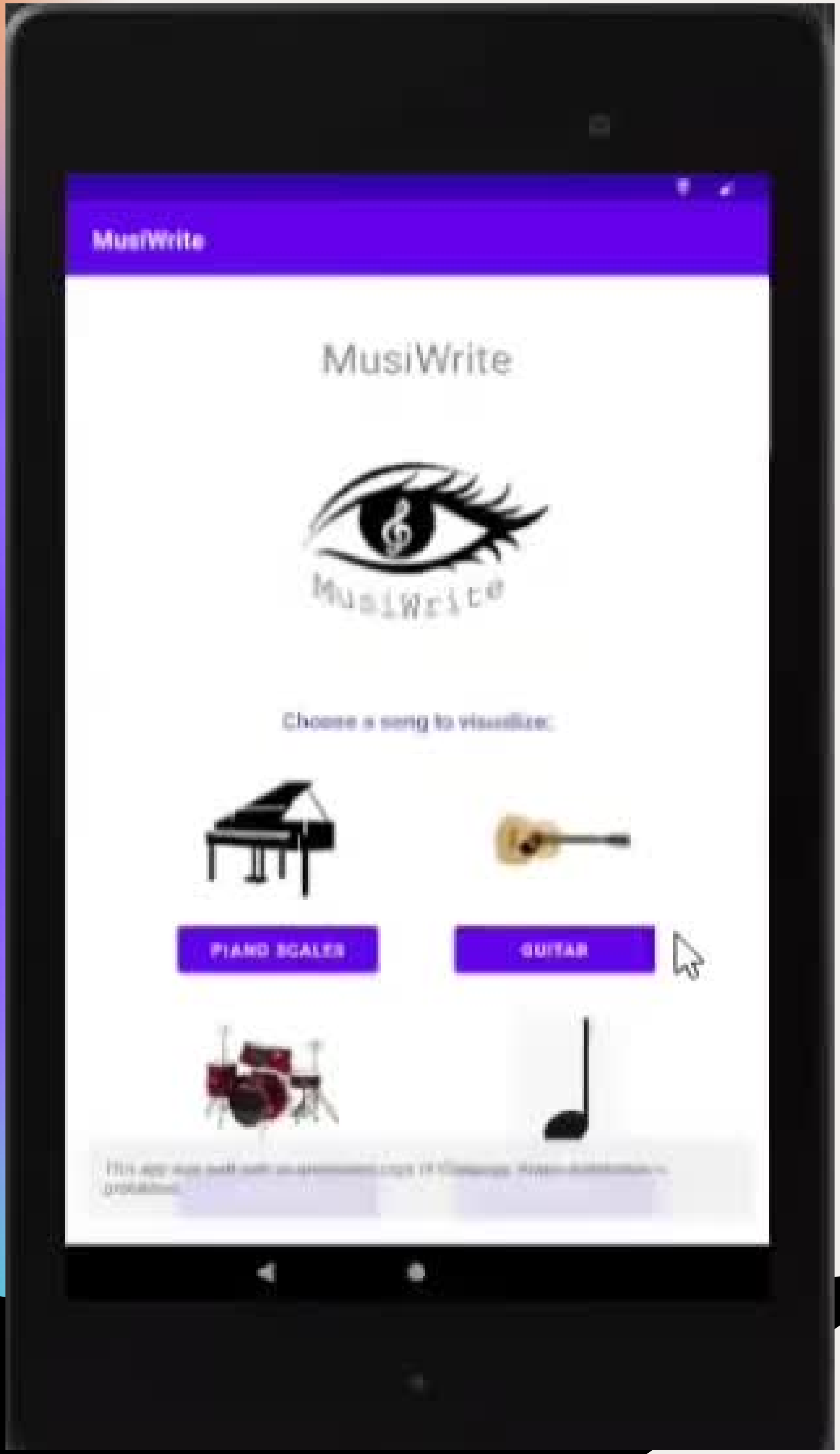
Visualization

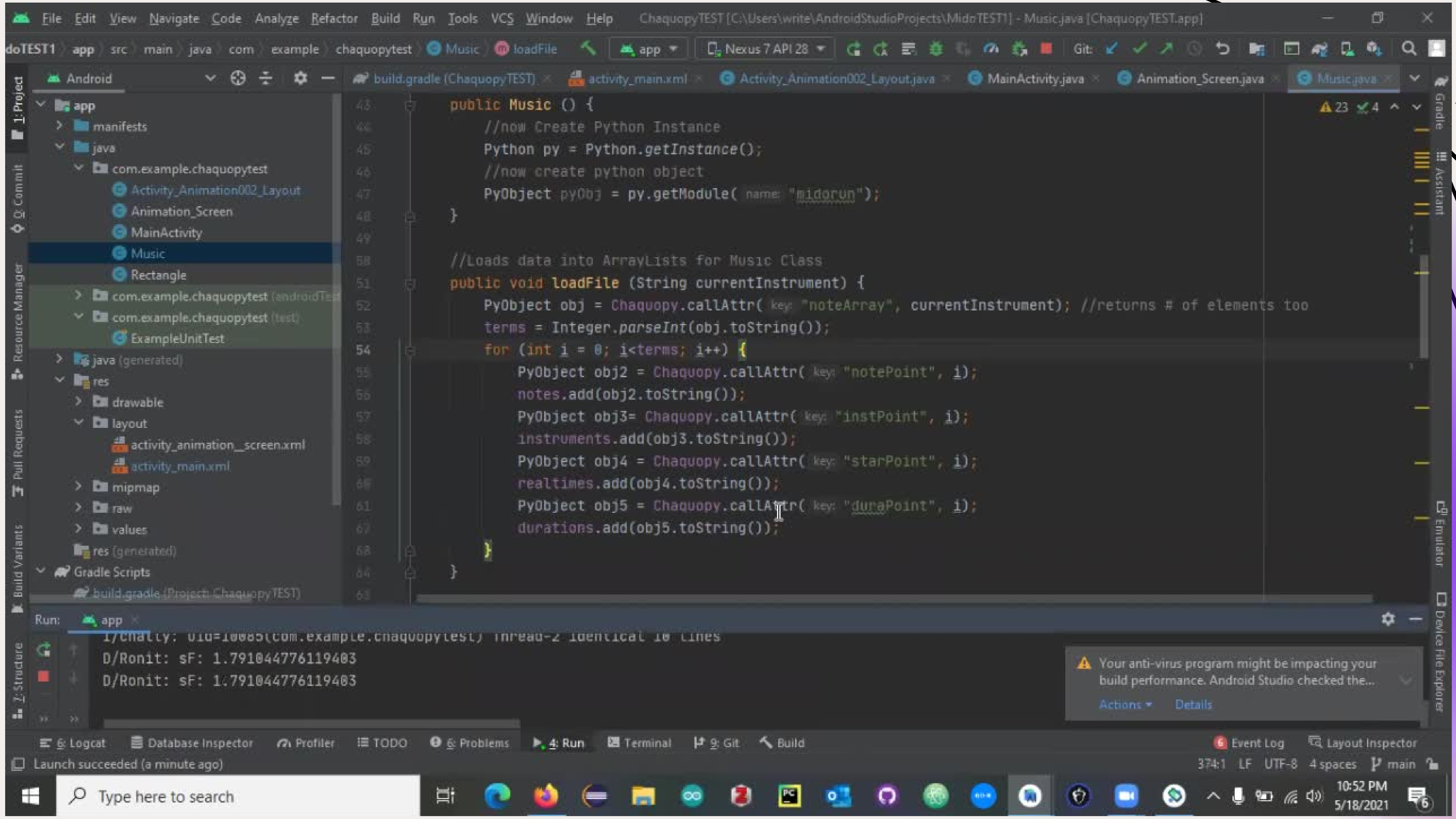
The App visualizes the song in several colorful bars according to Rhythm, Frequency, and Pitch

Playback

The App can play the music along with the bars, with included features of scrolling and pausing

App Demo





```java
    public Music () {
        //now Create Python Instance
        Python py = Python.getInstance();
        //now create python object
        PyObject pyObj = py.getModule( name: "midorun");
    }

    //Loads data into ArrayLists for Music Class
    public void loadFile (String currentInstrument) {
        PyObject obj = Chaquopy.callAttr( key: "noteArray", currentInstrument); //returns # of elements too
        terms = Integer.parseInt(obj.toString());
        for (int i = 0; i<terms; i++) {
            PyObject obj2 = Chaquopy.callAttr( key: "notePoint", i);
            notes.add(obj2.toString());
            PyObject obj3= Chaquopy.callAttr( key: "instPoint", i);
            instruments.add(obj3.toString());
            PyObject obj4 = Chaquopy.callAttr( key: "starPoint", i);
            realtimes.add(obj4.toString());
            PyObject obj5 = Chaquopy.callAttr( key: "duraPoint", i);
            durations.add(obj5.toString());
        }
    }
```

```
I/chatty: uid=10085(com.example.chaquopytest) Thread-2 identical 10 lines
D/Ronit: sF: 1.791044776119403
D/Ronit: sF: 1.791044776119403
```

Your anti-virus program might be impacting your build performance. Android Studio checked the...

Launch succeeded (a minute ago)

Tools & Technologies

Python MIDO Library

This library allows the app to read MIDI music files to obtain data such as the pitch, start time, duration, and instruments for each note

Android Canvas Class

This class was used to draw rectangles and animate them based the data obtained

Chaquopy Enviroment

This tool was used to integrate the Python and Java aspects together

Algorithms

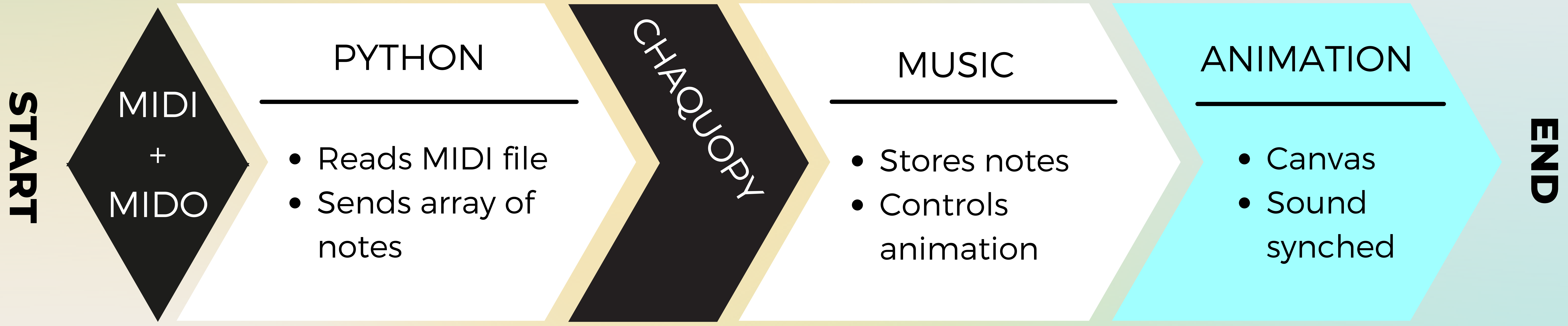
Music Analyzation

1. Python Executable builds lists of notes and attributes (4 lists: notes, start times, durations, and instrument)
2. MIDO Python Library to read MIDI files
3. Chaquopy | Allows Java to run commands in Python
4. Music Class in Java allows for getting and setting 4 ArrayLists

Animation

1. Uses Chaquopy to obtain music data stored into 4 ArrayLists
2. X and Y values for each Note Rectangle are calculated
3. Rectangles are drawn on Canvas
4. Animated by refreshing the Canvas every 50 ms and translating rectangles 5 pixels
5. MP3 audio plays in parallel

App Architecture



Future Extensions

ADD ABILITY TO SEARCH FOR MUSIC

- a feature common in popular music player apps but missing in audio visualizer apps
- would allow the user to choose music outside of the provided local files

ADD ABILITY TO CREATE/ MANAGE PLAYLISTS

- user would not be required to search up their intended song each time
- increases efficiency through easier access of the songs and decreases loading time by saving song data

Any Questions?