



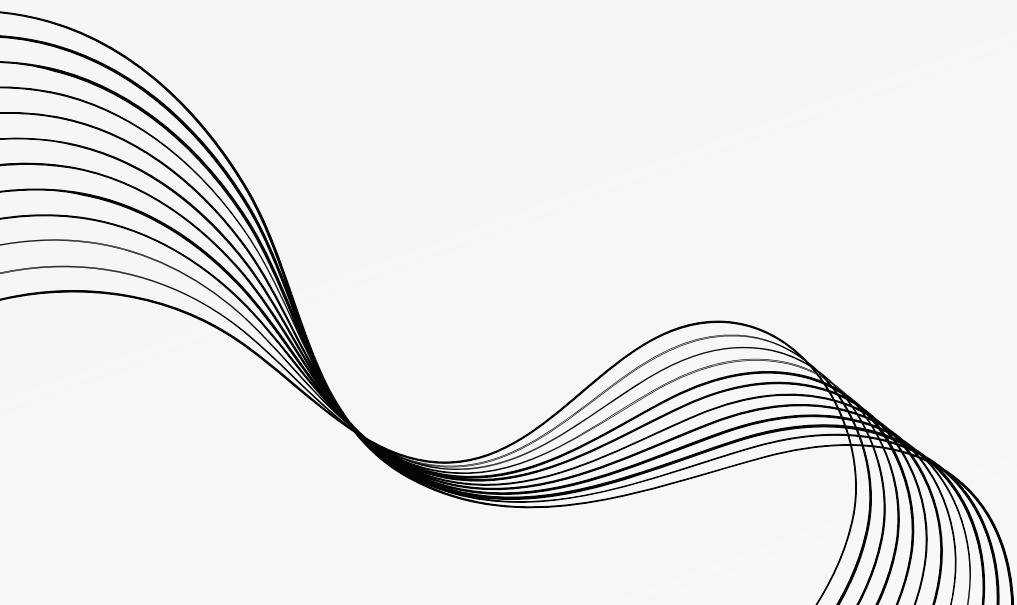
**TEAM 4**

**SOUNDLINK**

# THE PROBLEM

It is difficult for deaf/hard of hearing people to recognize loud and potentially dangerous sounds in the environment

Existing notifications do not give context about the sound - making it confusing and irrelevant



# IS THIS PROBLEM A PROBLEM?

I can't hear my apartment's fire alarm either. But my cats do, and one of them hates it. So when she starts slinking away from her usual nap spot, I know to check the hall for the fire alarm lights.

## **Fire Alarm for the hearing impaired? Need some Help on this**

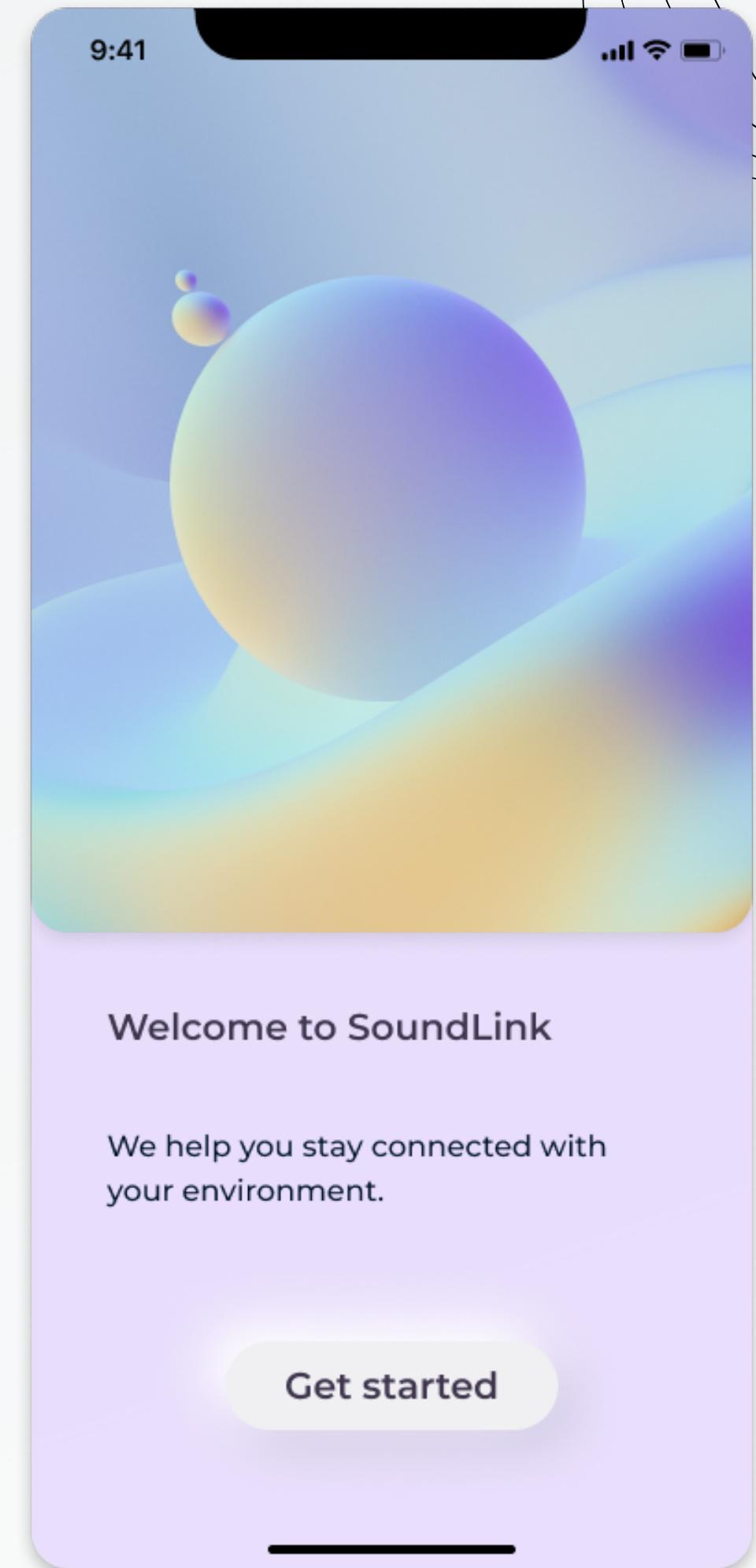
A family friend of mine is deaf in her right ear. The other night she was sleeping in her dorm when a fire alarm had gone off and she was unable to hear it. Fortunately everything is fine, but for the future I was wondering if there are any alternatives or anything that could assist in the future the next time a situation like this is encountered. I looked around Google but wasn't quite sure exactly what I was looking for. Any help would be very much appreciated, thank you.

I'm hearing, but both of my parents are/were profoundly (completely) Deaf. Every time I visit, there is inevitably something making a noise they're unaware of. An alarm beeping, a pipe knocking, a light buzzing. They once drove round for an entire day with their car alarm blaring!

# INTRODUCING SOUNDLINK

We help you stay connected with your environment.

Get notified when there is a loud sound nearby, when we get a report of crisis in your area or when someone repeats your name.



# IS THIS VALUABLE?

Now, I know apps to check noise levels exist - decibel measurers etc. That's not what I mean - my mum doesn't need to know how loud something is. She needs to check if a noise is being made, a description of that noise, and its location. I can't seem to find anything like that.

Problem with Android is the phone goes into sleep mode with the screen off and most sensors are switched off. It was originally meant to save battery but now it's more of a security issue as it's a way to unlock the phone.

Deaf guy here. There was a app for that.

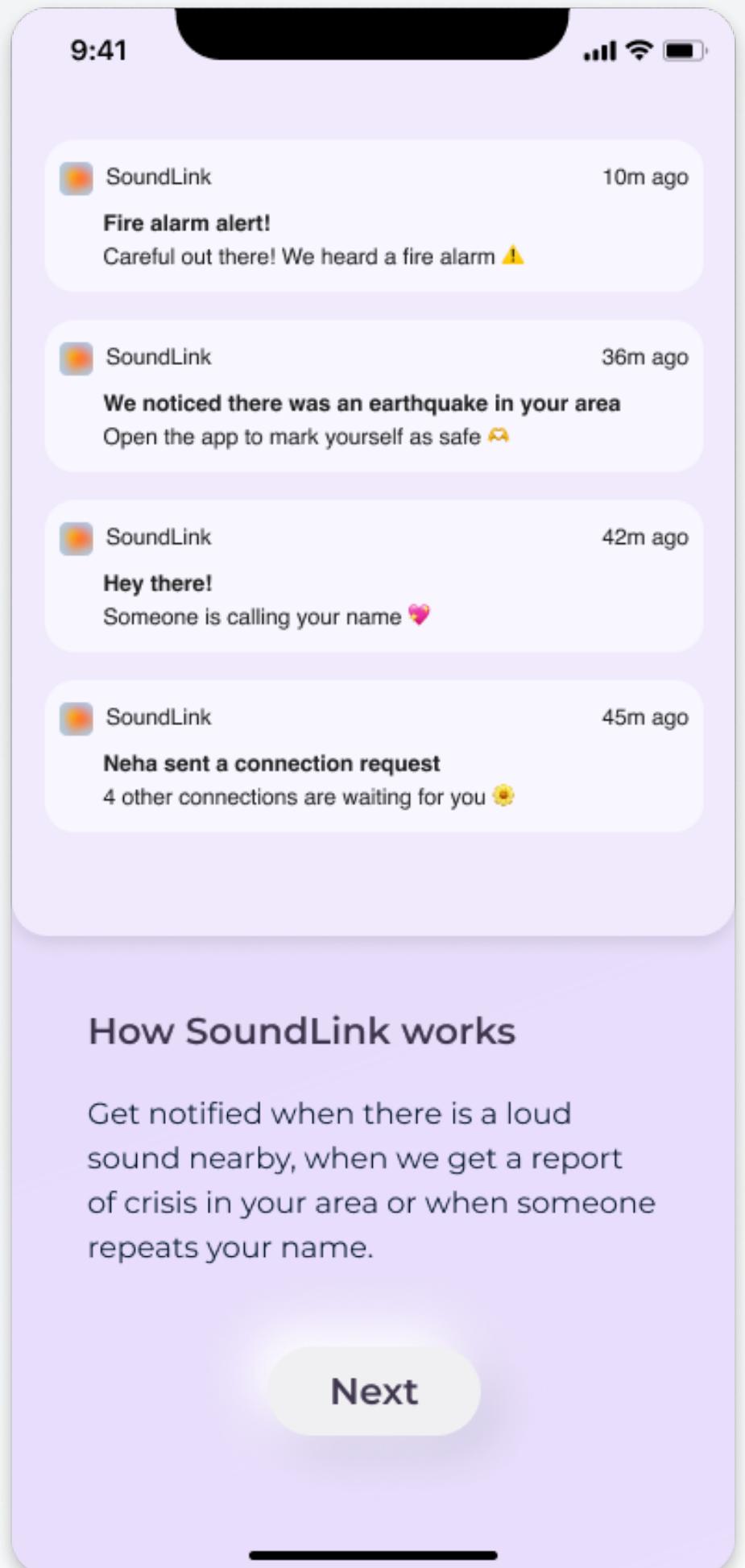
Unfortunately it was terrible!!! It vibrates at any noise no matter what. So it was basically vibrating nonstop and I'd have no idea what was making the noise and from where.

The Noise app measures the ambient sound levels in your environment using the microphone and duration of exposure. When Apple Watch detects that the decibel level has risen to a point where hearing could be affected, it can notify you with a tap on the wrist.



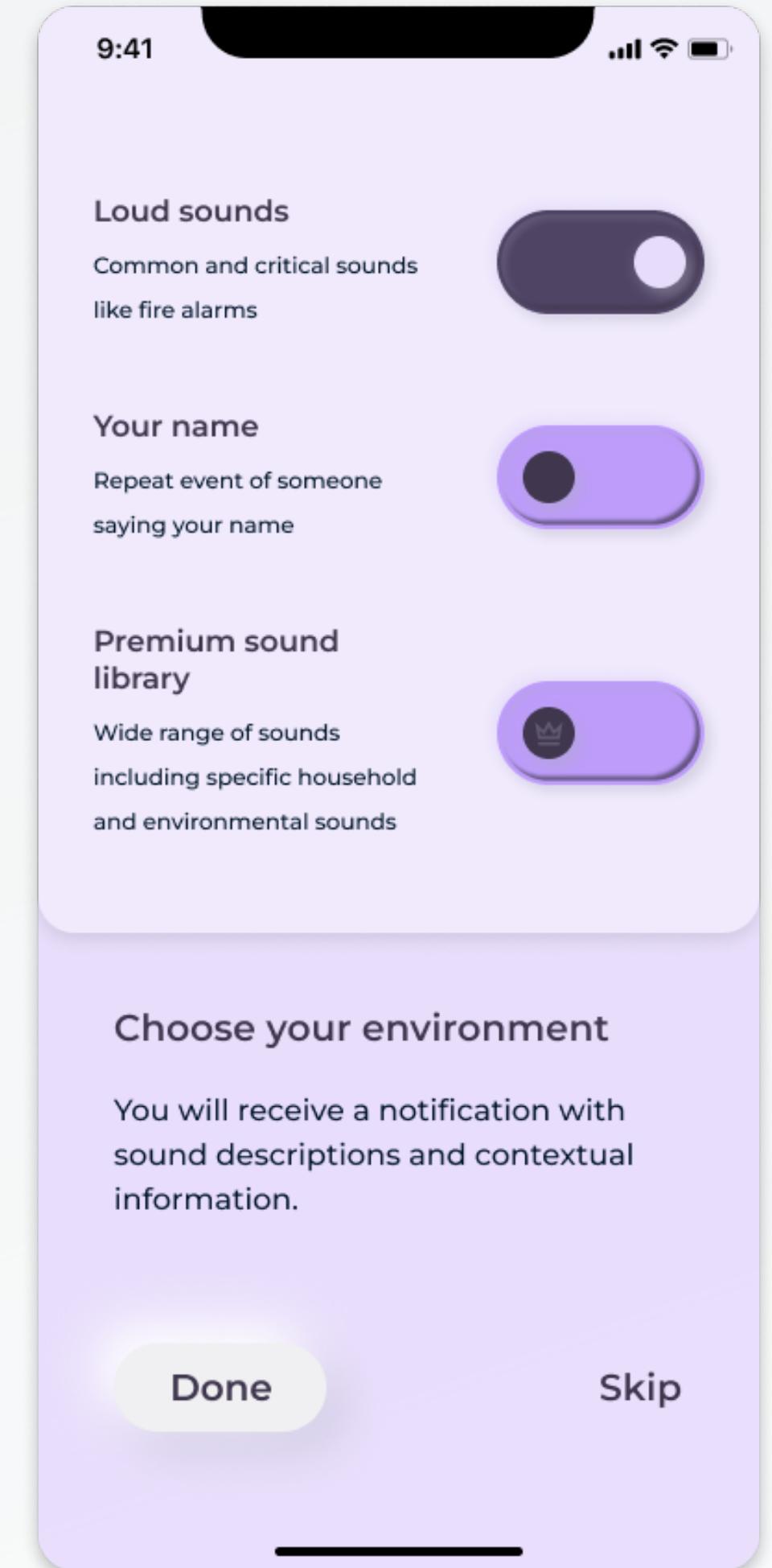
Apple Support

<https://support.apple.com/guide/watch/watchos> :



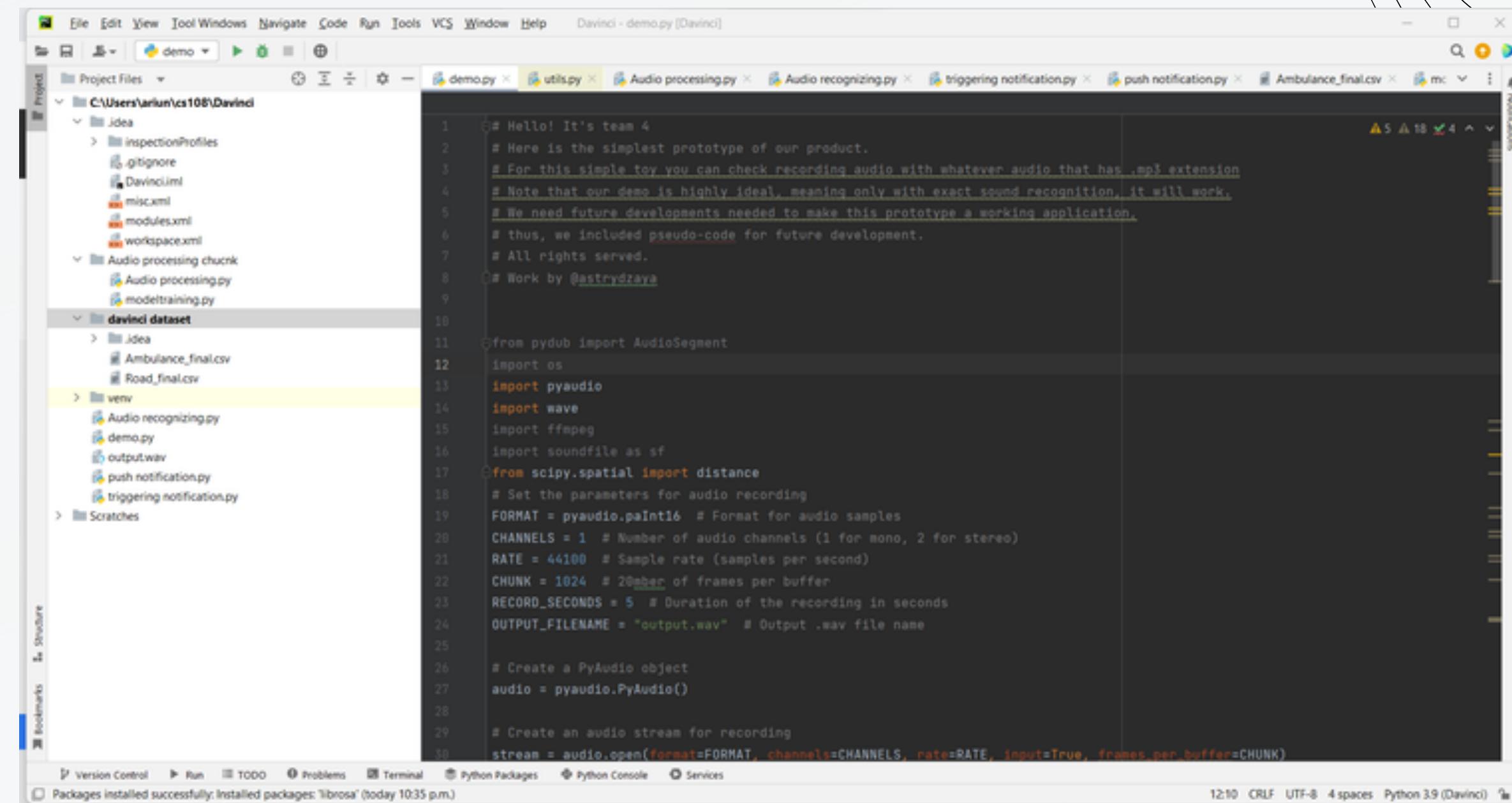
# BUSINESS MODEL

Freemium model



# BEHIND THE SCENES

Providing accurate contextual information about loud sounds in the user's surroundings



The screenshot shows a code editor interface with a Python script named `demo.py` open. The project structure on the left includes files like `inspectionProfiles`, `Audio processing chucnk`, and `davinci dataset`. The code itself is a prototype for audio recognition:

```
# Hello! It's team 4
# Here is the simplest prototype of our product.
# For this simple toy you can check recording audio with whatever audio that has .mp3 extension.
# Note that our demo is highly ideal, meaning only with exact sound recognition, it will work.
# We need future developments needed to make this prototype a working application.
# thus, we included pseudo-code for future development.
# All rights served.
# Work by @astryzaya

from pydub import AudioSegment
import os
import pyaudio
import wave
import ffmpeg
import soundfile as sf
from scipy.spatial import distance
FORMAT = pyaudio.paInt16 # Format for audio samples
CHANNELS = 1 # Number of audio channels (1 for mono, 2 for stereo)
RATE = 44100 # Sample rate (samples per second)
CHUNK = 1024 # 20ms of frames per buffer
RECORD_SECONDS = 5 # Duration of the recording in seconds
OUTPUT_FILENAME = "output.wav" # Output .wav file name

# Create a PyAudio object
audio = pyaudio.PyAudio()

# Create an audio stream for recording
stream = audio.open(format=FORMAT, channels=CHANNELS, rate=RATE, input=True, frames_per_buffer=CHUNK)
```

# TEAM

1st year CS  
student  
UTSC

PETER

4th year  
Business student  
UTSC

NEHA

2nd year Stats  
student  
UTM

ASTRYD

**THANK YOU :)**