What is the Project title?
J.A.C.K.: Joint Audio Correction Kit

Who is in the group?
Cooper Barth, Andrew Finke, Jack Wiig

Give a high-level description of what you want to do. (a couple of sentences) Are you solving a problem? Duplicating a system you think is cool? Running an experiment?

The objective of our project is to create a two-part audio correction kit consisting of two parts: an interface for noise reduction in audio files and a software providing an active noise-cancellation solution. The first portion of this project will mimic other existing noise-reduction solutions but also provide a clean one-click interfaces for users to easily utilize. The second portion of this project is solving an existing problem, creating a software solution for a space which has previously only been explored in the hardware.

Explain why this is this an interesting/useful/cool thing to do (a paragraph)
The first part of the project, the audio file noise reduction, will act as an exercise in meaningful learning through development. Noise reduction software that works on audio files has already been integrated into software like Adobe Audition. However, building this software ourselves will give us a deeper understanding of the process behind audio noise detection and reduction. The implementation of a clean one-click interface will also enable users without experience with more complex software to accomplish noise reduction much easier. The second part of the project, software-based active noise cancellation, is an interesting problem simply because it's never been done before. Noise cancellation is typically accomplished via hardware, as lag is a huge inhibiting factor. Developing an effective software for active noise cancellation would be very impressive for us as students if we could pull it off.

What prior art is there?
Noise reduction and cancellation are important qualities in higher quality headphones. Modern solutions perform this in real time but are often much costlier than their non-cancelling counterparts. Research on noise cancellation and reduction in real time has been ongoing, with some early sources being a paper that attempts to isolate the voice in a transmission from the high-ambient background[1]. This paper was published in 1980 and found that using different noise suppression factors and pre-filtering helped isolate the noise. In more modern times, researchers are looking for predictive noise cancellation[2]. Active noise control systems are not quick enough to do real-time cancellation, so they must be use predictive methods to be useful. We then looked at the hardware used noise cancelling headphones to understand what existing solutions use. The paper we found explored the circuit that controls the active noise cancellation[3]. Lastly, we looked at a filtered least mean squares algorithm for noise cancellation[4]. The aim of this study was to make the noise cancellation process using filtered least mean squares more efficient in respects to spatial regions.

[1] Speech enhancement using a soft-decision noise suppression filter
McAulay, Robert, and Marilyn Malpass. "Speech enhancement using a soft-decision noise suppression filter." IEEE Transactions on Acoustics, Speech, and Signal Processing 28.2 (1980): 137-145.

[2] Prediction filter design for active noise cancellation headphones
Guldenschuh, Markus, and Robert Höldrich. "Prediction filter design for active noise cancellation headphones." (2013).

[3] Controller design for active noise cancellation headphones using experimental raw data
Yu, Shiang-Hwua, and Jwu-Sheng Hu. "Controller design for active noise cancellation headphones using experimental raw data." IEEE/ASME transactions on mechatronics 6.4 (2001): 483-490.

[4] Sparse complex FxLMS for active noise cancellation over spatial regions
Zhangg, Jihui, et al. "Sparse complex FxLMS for active noise cancellation over spatial regions." 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016.

What is the EXACT TASK that your system will do?
For noise reduction, we will take an audio file as input on our website. It is assumed that the audio file will have some level of noise that the user wants reduced. The website, with a python backend, will then analyze the audio track for noise, and attempt to remove that audio from the track. After doing so, we will display the waveforms for both the original audio track and the new, noise reduced track, allowing the user to see the effect of the algorithm.

For noise cancelling, we will take an audio file as input on our website. Then, as the user plays the audio track, we will take samples of the room's background noise from the headphones' microphone and cancel out the received noise.

What measure will you will use to evaluate performance of what you build?
For both noise reduction and noise cancelling, much of the real world performance (unlike the box example from the homework), won't have exact answers for precision. Therefore, we will primarily be relying on qualitative analysis. For noise reduction, we will compare our results to tools such as Adobe Audition, and for noise cancelling we will compare our results to using noise cancelling headphones.

Is there a data set that your system can be tried on? If so, what is it (give links). If not, explain why not?
The data set will be any audio track the user uploads. Therefore, we don't need an existing data set.

What is the baseline approach you will compare your system to?

We will be comparing our noise-reduction system to existing technologies that exist to apply noise reduction, such as Adobe Audition. We will be comparing both effectiveness of noise-reduction as well as ease of use for the user (e.g. upload the file + one click to remove noise). For the real-time noise cancellation, we will be comparing our software solution to existing hardware solutions, such as noise-cancelling headphones. This can be accomplished by playing an audio file with the headphones' noise-cancelling hardware turned on, then running our software with the noise-cancelling hardware off and comparing.

Describe any software will you need to write.
We will need to write the web interface for uploading the audio track, the interface for displaying the original track and the adjusted one, and the python backend, for which we will need to implement a noise reduction algorithm and an algorithm for soft noise reduction.

What are potential obstacles to success?
One obstacle in our path is the difficult implementation of an accurate and effective noise-reduction algorithm for audio files. This may be difficult to deploy, especially on the level of software which has already been created by others. Another obstacle in our path to success is the challenge of tackling real-time noise cancellation through software alone. This is a problem that has not been solved effectively before, and takes a wealth of knowledge and experience to handle effectively. This, along with the lag time caused by Python's slowness, will make true noise cancellation very difficult.

How will you QUICKLY determine if these obstacles will stop you?
We plan to execute our plans for development very early in the building of the product. As such, we will know if we hit any serious roadblocks very earlier, and will be able to reach out for help as soon as possible if need be.

What other tasks (besides coding) will you need to do?
The main task that we will need to work on is a comprehensive reading of literature. As we will likely be applying many existing algorithms, we will need to source from other research which has developed these algorithms and software and base components of our project off of it.

Milestones
By the first meeting we will have completed a basic web interface that supports uploading audio tracks and displaying their waveforms. This will also include the python backend (which we will use for creating the waveform images and later implementing the algorithms)

By the second meeting we will have completed our noise reduction algorithm which will allow the user to upload audio tracks with noise and have that noise reduced on the fly. We will also have the interface implementation that allows the user to play the resulting audio file and see it's waveform and spectrogram.

By the final presentation we will have attempted to implement a soft noise cancellation algorithm. This would allow the user to play an audio track and then, using the last few seconds of microphone input, attempt to cancel out the audio of their environment during playback.

Given the tight integration of each of the components, all three team members will share equal responsibility of each milestone, programming each part as a group.