Hearing is a necessity. It allows us to communicate and interact with the outside world. Hearing loss is a serious and prevalent problem in our society today. The main catalysts for this epidemic, however simple they may seem, are listening to personal music players at high volumes and enduring loud noise in the workplace. Unfortunately, today's healthcare system does not provide for an annual or routine hearing test. A routine checkup would encourage people, if they do have a hearing deficiency, to alter their lifestyle in order to avoid eventual deafness.

Currently, the device used to analyze a person's hearing is called an audiometer. An audiometer evaluates a person's hearing by analyzing their ability to hear a specific set of tones with ranging frequencies, or pitches of sound, and amplitudes, or volumes of sound. Commercial audiometers today, however, are not beneficial for personal ownership and usage because they are highly sophisticated. Thus, they tend to require a hearing specialist with thorough and professional training using the device. Also, they are very expensive, cumbersome, and require annual calibration to ensure accurate performance. I realized that the best way to counter the problem of growing hearing loss would be to create a mass market audiometer that could be used by anybody, anywhere, anytime, and at a low cost.

I decided, considering the recent increased popularity of computer applications, it would be most logical to create a computer-based audiometer application. With it being a software application, it would be easy to use because many people are accustomed to using a computer and it could theoretically be used on any computer because applications are easy to distribute. It would also be an inexpensive alternative because of the low cost of computer applications. This product would be able to meet the guidelines I had initially established.

The next step in my engineering process was to set specifications for my application, which were the basic functions I wanted my device to be able to execute. Before I did this, however, I conducted preliminary research in order to try and augment the features of the current audiometers on the market today. One of the more popular audiometers I based my specifications on was called the Welch-Allyn AM 182 Manual Audiometer. One of the major disadvantages of this audiometer, however, is that it is operated manually. This means a person, in order to use it, must operate various knobs, switches, and buttons. For most people without specific knowledge of what these devices do, this hearing test can be very difficult to complete. To eliminate this aspect of the audiometer, I decided to make my audiometer fully automatic in its function. For example, a person using my application would solely respond “Yes” or “No” to hearing a sound without having to worry about operating a myriad of random switches. This specification would ensure the audiometer’s user-friendliness.

Another advantage to my audiometer is that it utilizes a visual display to convey results to a user rather than a textual display. This makes it much easier to understand results. In clinical audiometry, a visual display of results is called an audiogram. An audiogram is also divided into colored sections which signify the severity of a person’s hearing loss. For example, the top of an audiogram is typically colored green signifying that a person who obtains results in that section has no hearing loss and the bottom of an audiogram is typically colored red signifying that the person has severe hearing loss.

The final specification I set based on my initial research was the frequency range for my audiometer. The frequency range is the range of pitches that an audiometer has been programmed to test a person’s hearing in. Most clinical audiometers’ frequencies range from 100Hz to 8,000Hz, which is the threshold of human speech. I decided to program mine to test frequencies up to 20,000Hz, which is the threshold of human hearing. These higher frequencies are important to test because they results they yield can help predict when a patient’s middle frequency range might be lost as well. Given that most sounds we hear on a daily basis are of the middle frequency range, it is vital that the higher frequencies are tested as well so that the middle frequency range can be kept intact.

I developed this application, which I called SoundWorks, in C# programming language in Microsoft Visual Studio on a laptop running Microsoft Windows 7. I continued to add features to SoundWorks based on my ongoing research. For example, when I met with the hearing specialist at the local Costco, I learned that they used masking white noise and pulse tones in their audiometers for more accurate results. I decided to incorporate these features into my own application. I also met with the audiologist at the local Hearing Center and learned about how she sometimes uses a speech recognition test to evaluate a person’s hearing loss. I decided to create a speech recognition test in my own application for extensive hearing loss evaluation of my users. Other features of this application include multilingual capability, sound and text prompts to help the user progress through the test and a test report with all results that can be easily accessed.

To begin the test, the user will first enter personal information that will be displayed in a test report created at the end of the hearing test. The user will then select features they would prefer to have in their test environment including background white noise and/or pulse tones. After selecting features, the test will begin by exposing the user to various sounds in the left ear. The application is programmed to test the user in every frequency until they can hear that specific frequency. As the user progresses trough the test, an audiogram on screen will update to convey results. Once the user has progressed through all the frequencies in the left ear, they will repeat the same process for their right ear. At the culmination of testing both ears, a user will be able to access a test report outlining all of their results and analyze their audiogram. They can take their report to a doctor for further interpretation or use it to keep track of their progress over a period of time. Given that this application evaluates a person's hearing loss automatically, it removes any burden on the user.

The next step of the engineering process was verifying the development. I made sure that when I ran the program, no errors would appear. This means debugging any errors if necessary in order to guarantee that the application can run without fault. I also calibrated the sounds coming out of my application, which is a step required for all audiometers. My calibration setup included the laptop running the application, headphones connected to the laptop and a decibel meter close to the headphone buds. A decibel meter is a device that measures the sound pressure level (noise level) of the environment or in this case, the sound coming from the headphones, which was generated by the running application. The purpose of this calibration process is to make sure that the sounds being produced by the computer are the correct amplitude levels. I also plan to get this product officially calibrated by a medical agency in order to assure that it can be used by the everyday consumer.

Whether it's due to consistent use of personal listening devices at high volumes or loud noise in the workplace, hearing loss is affecting more and more people every day without them even being aware of it. I believe, however, that the application I have developed will be a major step in countering this dilemma of hearing loss. Future extensions for this product include porting onto mobile devices like smartphones and tablets, calibration by a certified medical agency, and capability to test bone conduction. Due to the application's simplicity, portability and inexpensive effectiveness, I can imagine it becoming a marketable product making its way into every household in the world, and preventing a hearing loss epidemic.