Final Report: Bekesy Test for Mobile

Team 2: Samir Dahmani, Nihit Mody, Joseph Wolanski

Client: Douglas Oliver, 263 Farmington Avenue, Farmington

December 7th 2012
Table of Contents

- Abstract
- Introduction
  - Background
  - Purpose
  - Previous Work Done by Others
- Project Design
  - Introduction
- Optimal Design
  - Objective
  - Subunits
- Realistic Constraints
- Safety Issues
- Impact of Engineering Solutions
- Life-Long Learning
- Budget and Timeline
Abstract

We are designing a mobile phone application that will accurately model professional Bekesy audiometry, for Dr. Douglas Oliver. The Apps that are currently available on the various online stores bear little resemblance to the accuracy of a professional audiogram. Thus far, there is no mobile phone application that has the ability to reliably identify hearing loss in a subject. This is due to a lack in normalized threshold over which to accurately measure a subject’s hearing. Essentially there is no App that reproduces Pure Tone Audiometry (PTA), which is used to identify degree, type and configuration of a hearing loss.

1 Introduction

1.1 Background

At the most basic level an audiogram is a screening test like that which is used in schools. A series of tones at fixed volumes are presented to the listener who then indicates which ones he or she can detect. PTA is a subjective and behavioral measurement of hearing threshold, because it relies on patient initiation of a response to acoustic stimuli. PTA is designed for clinical use on adults and children old enough to understand and execute the test procedure. Like the majority of clinical tests, calibration of the test environment, the equipment and the stimuli to ISO standards is pivotal, as PTA measures thresholds of hearing only (in contrast with other methods which delineate sound localization). Our client is Dr. Douglas Oliver, a professor of neuroscience at the UConn health center.

Clinicians measure sound intensity in dB HL (decibels Hearing Level), in other words, decibels relative to the quietest sounds that a young and healthy individual can detect. In a clinical audiogram test, pure tones between frequencies of 250 and 8000 Hz are presented at varying levels, to determine the quietest audible sounds a patient can hear. These are also called pure tone detection thresholds, and are taken for the left and right ear. Thresholds between -10 and +20 dB HL are considered well within the normal range, whereas any above 20 dB HL would yield a diagnosis of mild, moderate, severe or profound hearing loss. This is demonstrated in figure 1.
Table 1 below, lists a range of normal hearing thresholds.

**Table 2: Audimetric hearing thresholds of normal ears: conversion of dB SPL into dB HL**

(Extracted from ISO, 2003)

Pure-tone audimetric thresholds are expressed in dB HL (hearing level). The differences between dB HL and dB SPL arise from isophonic curves.

<table>
<thead>
<tr>
<th>Frequency [Hz]</th>
<th>dB SPL</th>
<th>dB HL</th>
</tr>
</thead>
<tbody>
<tr>
<td>250</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>500</td>
<td>&gt; 5</td>
<td>0</td>
</tr>
<tr>
<td>1000</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2000</td>
<td>- 2</td>
<td>0</td>
</tr>
<tr>
<td>4000</td>
<td>- 5</td>
<td>0</td>
</tr>
<tr>
<td>8000</td>
<td>13</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 1 (Fig. 3)
Pure tone audiometry is useful for detecting many types of hearing loss. These include: conductive hearing loss, which is a loss that takes place when the transmission of sound to the inner ear is compromised due to an ear infection, calcification etc, a relative db HL plot is displayed below in figure 3.

![Graph of db HL of patient with conductive hearing loss]

**Fig. 3 (Graph of db HL of patient with conductive hearing loss)**

Sensory-Neural Hearing loss, which is the most common neural hearing loss, and results from damage to sensory hair cells in the cochlea. This happens when the outer hair cells become damaged by exposure to excessively loud sounds or wear out in old age.

Patients with noise damage have characteristic concentrations of damage, particularly the 4 kHz frequency range, as the human ear is especially efficient in detecting that range using the outer and middle ear.

**1.2 Purpose**

The purpose of this project is to develop a software application for mobile devices that accurately determines the extent of hearing loss in the user. The primary goal is assessing whether or not the subject should visit a local doctor and get his/her hearing tested. The goal is not however to create an app that does nothing more than functionally perform a hearing test. The ultimate intent is to design one that is calibrated. The importance of calibration cannot be understated; calibration is what sets apart a typical mobile phone app based hearing test apart from a professional audiogram.
1.3 Previous Work Done by Others

The Apps that are currently available on the various online stores bear little resemblance to the accuracy of a professional audiogram. Thus far, there is no mobile phone application that has the ability to reliably identify hearing loss in a subject.

Patent search yielded nothing of note. Patent searches are typically foregone when developing software due to the cost of hiring attorneys to search the mountains of software patents. This cost can easily eclipse that of a lawsuit from a patent-holder (software developers typically just cross their fingers). As a result the cost of determining if a particular piece of software infringes any issued patents is too high and the results are too uncertain.

However this should not be a problem in any case as we are developing on an open source platform (Android) and we plan to make the software open source, which is protected under F/OSS (Free and open-source software model).

The design that will be detailed further in the proceeding sections was chosen because it most closely reflects the Bekesy test and will likely generate more accurate results. This design is also easy to learn and use, which will result in reduced operator error and increased accuracy/reliability.

1.4 Map for the rest of the report

Included in this report will be, other alternative designs, as well as the basis for our optimal design selection. The optimal designs chosen, will be explained in further detail. This includes all subunits of the projects as well as how the will work together to form our final product.

The prototypes developed will be shown to have met the criteria set forth by our clients. Other considerations such as economic, engineering standards, environmental, sustainability, manufacturability, ethical, health and safety, social, and political considerations will also be discussed.

Scrupulous notes were maintained in regards to the budget as well as time spent on each of our projects. This will also be discussed within our report. Each team member will show their hourly contribution to each of our projects, stating what they have done, and how they have worked together to create our final product.

Finally, our report will be concluded with an appendix, acknowledgements, references, as well as a report summary.
2 Project Design

2.1 Introduction

Design 1

The user will begin by identifying the type of headphone he/she is using, choosing from a list of headphone varieties and possibly models i.e. over-ear, ear-buds. Next the user will be prompted to choose from a list of frequencies that he/she will have the option of playing and attempting to hear. Each frequency will be assigned an age range. The age range will reflect the average maximum frequency of sound that individuals with healthy hearing can perceive. The frequencies will be scaled and calibrated contingent upon the type of headphone selected in the starting menu. The user will then be able to determine whether or not he/she is suffering from hearing impairment based on their test results. Some limitations of the app depend on the operation of the app. The app will work optimally in a quiet environment, using a pair of Sony MDR-V6 headphones, or otherwise similar digital monitoring headphones.

Design 2

When the audiometer is turned on, a pure tone at a midrange frequency will come through the earphone. The subject will control the intensity of the stimulus by pressing a button while listening to a pulsing (0.5s) pure tone whose frequency slowly moves through the entire audible range. This is what is called an “interrupted” tone. The intensity diminishes as long as the button is depressed. When the intensity is too low for the subject to hear the tone, the button will be released and the intensity will start to increase. When the subject again hears the tone, the button will be pressed again, producing a zigzag trace. The test will involve diagnostics on each ear, after which the tracings of both the left and the right ear will be compared. The test will be usable to differentiate between cochlear and neural hearing losses.

Because of the importance of having the calibration of the headphone and output jack be constant across all users of the app, the mobile platform chosen for the development of the app is Android Platform, using the wildly popular Google Nexus 7 Tablet. Due to the widespread use of Apple ear buds, the app will be calibrated according to the ear buds and Nexus 7 combination. Such uniformity is not so apparent in Android phones, due to the variety of manufacturers (and likewise headphones and/or output gains).

Design 3

A pure-tone air conduction hearing test determines the faintest tones a person can hear at selected pitches (frequencies), from low to high. During this test, earphones are worn so that information can be obtained for each ear. The person taking the test can respond indicate whether or not they hear a sound by
clicking a sound. A “toast” would then enter the screen to indicate that a sound has been played (like above). The app would compile user data and create a graph of each ear at the end over a normal hearing threshold.

2.2 Optimal Design

2.2.1 Objective

The fundamental concept behind the Bekesy audiometer is that the patient records his or her own threshold automatically on an audiogram black. When the audiometer is turned on, a pure tone at a midrange frequency will come through the earphone. The subject will control the intensity of the stimulus by pressing a button while listening to a pulsing (0.5s) pure tone whose frequency slowly moves through the entire audible range. This is what is called an “interrupted” tone. The intensity diminishes as long as the button is depressed. When the intensity is too low for the subject to hear the tone, the button will be released and the intensity will start to increase. When the subject again hears the tone, the button will be pressed again, producing a zigzag trace. The test will involve diagnostics on each ear, after which the tracings of both the left and the right ear will be compared. The test will be usable to differentiate between cochlear and neural hearing losses.

Due to the importance of having the calibration of the headphones as well as the output jack remaining constant across all users of the app, the mobile platform chosen for the development of the app is Android Platform, using the wildly popular Google Nexus 4 Phone. Because of the largely popular Apple ear buds, the app will be calibrated according to those ear buds as well as the Nexus 7. Such uniformity is not so apparent in Android phones, due to the variety of manufacturers.

The user will then be able to determine whether or not he/she is suffering from hearing impairment based on their test results

2.2.2 Subunits

Hardware: Android Phone/Tablet

1. Hardware used to build and deploy:
   - Macbook Air 13.3”
   - Dell Optiplex 745
   - 3D Connexion 3-D joystick (for pre-implementation graphics modeling and vector visualization)

2. User interface: Touch Screen

3. Hardware interface: Google Nexus 4, apple earbuds
Specifications

- Tests frequency range of 250 Hz – 8000Hz
- Runs on a mobile phone
- Audio output gain and headphone frequency range are calibrated and taken into account
- Performs accurate model of Bekesy audiogram
- Accurately and effectively stores data and displays plot of hearing range to the user
- Provides user with access to previous tests
- Has a feature that directs the user to local audiologists in the surrounding area

Fig. 4  *headphone dB output values*
Fig. 5 Preliminary UI Layout

Fig. 6 Preliminary UI Layout
The output gain of the headphone jack, as well as the frequency range of the earbuds will be measured and taken into account when during development. This will be the source of inherent calibration. The App will open to a readme which will feature background on Bekesy audiometry, as well as the standard ISO procedure for instructing the test taker how to perform the audiogram on his/herself that is required to be read to all subjects. The app will then open to the home screen pictured below (this is a prototype that our team has developed that is running in the Blue Stacks android emulator). The current prototype is written in openGL (which is a cross-platform open graphics library). Part of the reason we are writing it in openGL despite the fact that it is significantly more complex and temperamental, is that this will allow us to more easily port the application to iOS. This will allow us to release the app for iPhone, which is another extremely popular phone which features a standard output gain across models starting from the iPhone 4 model ranges, should we decide to do so.

The preliminary UI layout for design 1 (figures 6 and 7) featured a start button and a toggle button for user-attenuation of the tone. We did away with the start button in the first prototype to streamline operation (figures 8 and 9). The functionality is the same, however the red button doubles as the start button. There will not be a lot of client facing information to make it straightforward and easy to use. This design will also feature a running display of frequencies being played: 250 kHz, 1000 kHz, 4000 kHz and 8000 kHz etc. The app will compile user data and create a graph of each ear at the end over a normal hearing threshold. We will use the AudioManager component of the Google API to generate audio, openGL to generate 2D graphics. The API is typically used to interact with the integrated GPU, to achieve hardware-accelerated rendering. This will result in a fast and visually pleasing user experience, which is important to us, as it is something that many of the audiometry apps on the market lack. The app as described in figure 7, will direct the user to local audiologists at the conclusion of the test. For this we will use the Google Maps geolocation API for map tiles and listings.

---

![Diagram](image)

Fig. 7 Basic UML
The home screen above will follow the readme mentioned earlier in the report. The triangular button with a plus sign at the center initializes the next screen, which opens to the testing screen (figure 9). The triangular button with three lines, loosely depicting a fluttering stack of papers, directs the client to past usage statistics, a shortcut to local audiologists, a copy of the readme (background usage etc), as well as information about the developers and the creative commons license under which the application will be released.

A little more about the design-- currently our prototype is sporting a minimalist and utility driven aesthetic. The app is modeled after a laser focus design principle, which puts visual focus on a single obvious task to do once the user opens the application, instead of providing several options of equal weight/importance. The main attributable benefit of this design approach is of course simplicity. The user instantly knows what the application does and how to do it. This type of approach is best for applications that have a single most important function, like ours. The main function of course is to quickly and accurately test the user’s hearing, and do so using one and only one test in particular.

Our application will also feature collapsed content. This will hide nonessential options and widgets inside a single link that expands and collapses on request by the user. Reducing
clutter is something we took very seriously (a shortcoming we found was common among currently available audiometry applications). Few of the currently available applications we tried were particularly inviting to the user. We typically did not feel inclined to understand the application because of the unintuitive directions and layout of menus, as well as an ill-defined purpose to the application. It was not always clear what kind of test said app was designed to perform, and how to take that test.

Great care is being taken to ensure that all animations are carefully placed and very well executed. Instilling a sense of effortlessness and ease of navigation when using the app will be the ultimate goal while developing the user experience. Swiping backwards and forwards between menus is a navigation feature we will explore. For the most part, our application will keep it brief, using short phrases and simple words. People are likely to skip sentences if they’re long. We will also use pictures to explain ideas, particularly in the readme section, as this gets people’s attention and can be much more efficient than words. The app will also only show features that the user needs when they are actually needed. We don’t want the user to become overwhelmed when they see too much at once. Tasks will be broken into small digestible bits, and options that are not essential at a given moment in the procession of the application will be hidden. Essentially the user will be taught as he/she goes.

Fig. 9 Testing Screen (Prototype 1 “Navajo” running in BlueStacks)
Our navigation experience will try to instill a sense of confidence that the user knows he/she way around. This will be done by making different areas of the app distinct and a tap away. This is an example of implementing a flat UI hierarchy (pictured in figure 10 below). In other words the finished product will ideally not be navigationally very deep, involving mostly lateral movement between menus where the user will be able to return to the home screen within two taps.

![Flat UI Hierarchy](image)

**Fig. 10 “Flat” UI Heirarchy**

### 3 Realistic Constraints

#### 3.1 Engineering Standards

All engineering standards will be met in the development and production of the application. Many safety issues have been addressed already and improved upon with the optimal designs dictated in this report. The volume will be stable and volume and the volume of the pure tone will be capped at a reasonable volume, so as not to (further) damage the users hearing. Piercing tones at a high volume can quickly induce migraines, our app is designed such that it will be unobtrusive to use for extended periods of time.

Some limitations of the app depend on the operation of the app. The app will work optimally in a quiet environment, using a pair of Apple ear bud headphones.

**Economic constraints:**

Our budget is capped at 1000 dollars and so far and the group is under budget for the application: the Google Nexus 4 has been ordered at a cost of $349.99. The 3D connexion 3D joystick used for development is something that one of our group members owns already.
4 Safety/ Health

Volume will be attenuated appropriately to avoid any chance of hearing damage. Interfaced designed around reduced button taps means reduced carpal tunnel and other repeated motion afflictions.

5 Impact of Engineering Solutions

The code used will be recyclable and environmentally friendly and uses products/materials that are readily available. There will perhaps be a global impact if the application is successful and becomes widespread. Highly sustainable, as the app can be updated, and will be designed for a lasting professional platform and comparatively future-proof. Since we are developing on a Google device, and Google owns the operating system as well, the device will be updateable for a long time, and always as soon as the update is released. This will result in increased stability. Patches will come quickly as a result of the integrated hardware and software platforms. It is easily reproducible (download distribution) and free hosting. As far as ethical constraints, we have made sure the design created was done so on our own, and that we are not reproducing someone else’s idea generic idea (a Bekesy hearing test for mobile phones).

6 Life-Long Learning

Will become proficient in android development, will gain experience in the software development cycle. The team has already learned a great deal about user experience and design. This project has already taught us a great deal about application development. Joe has never used Java before and is making great strides in learning to develop some of the application logic behind the audiometry itself. Nihit who had some background in development after having taken an introductory level course in object oriented modeling at UConn is also coming along well. Samir who is also majoring Computer Science is learning a lot by implementing the theory being taught in his course-work to a real life client-based application. He is learning a lot of practical skills as a result.

7 Budget and Timeline

7.1 Budget

<table>
<thead>
<tr>
<th>Project</th>
<th>Part</th>
<th>Quantity</th>
<th>Price ($)</th>
<th>Shipping</th>
<th>Total Price</th>
<th>Part #</th>
<th>Purchase Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android application</td>
<td>Google Nexus 4 Phone</td>
<td>1</td>
<td>349.99</td>
<td>0.00</td>
<td>349.99</td>
<td></td>
<td>Google Play Store</td>
</tr>
</tbody>
</table>

Table 2 Budget
### 7.2 Timeline

We had some technical difficulties with our timeline so we exported the contents to excel, and is viewable below in table 2.

<table>
<thead>
<tr>
<th>ID</th>
<th>Active</th>
<th>Task Mode</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>34</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>Visit Doctor Oliver</td>
</tr>
<tr>
<td>35</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>Get revised and clarified specifications</td>
</tr>
<tr>
<td>36</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>do research on Bekesy audiometry</td>
</tr>
<tr>
<td>37</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>plan out how the app will work (core functionality)</td>
</tr>
<tr>
<td>38</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>determine preliminary UI</td>
</tr>
<tr>
<td>39</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>research different options for platforms</td>
</tr>
<tr>
<td>40</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>decide on android platform</td>
</tr>
<tr>
<td>41</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>research different options for devices</td>
</tr>
<tr>
<td>42</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>decide on android device Google Nexus 7 Tablet</td>
</tr>
<tr>
<td>43</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>research different headphone options</td>
</tr>
<tr>
<td>44</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>settle on apple earbuds headphones</td>
</tr>
<tr>
<td>45</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>visit Dr. Bernstein learn/take hearing test UCHC &amp;calibration approach</td>
</tr>
<tr>
<td>46</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>download android SDK</td>
</tr>
<tr>
<td>47</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>acquire android development book</td>
</tr>
<tr>
<td>48</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>begin learning android development</td>
</tr>
<tr>
<td>49</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>begin doing practice development apps</td>
</tr>
<tr>
<td>50</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>continue practicing and developing practice apps</td>
</tr>
<tr>
<td>51</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>begin developing core functionality of app</td>
</tr>
<tr>
<td>52</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>begin practicing openGL</td>
</tr>
<tr>
<td>53</td>
<td>Yes</td>
<td>Manually Scheduled</td>
<td>begin practice development of apps using openGL</td>
</tr>
</tbody>
</table>
Table 3 Timeline

8 Team Member Contributions

With the three projects we have been specified for this semester, we have split them into parts, and given the responsibilities of each project to each of our three team members. I have been fully in charge of the design and modification of the hearing test application.

Nihit and Joe have been in charge of the walker and the stairway lift systems. Nihit being the director and manager of the walker and Joe being in charge of the stairway lift system.

My contributions include learning how to use the Android SDK (Software Development Kit) and ADT (Android Developer Tools), designing and developing prototypes, designing and implementing data mining algorithms, assessing and establishing a user experience, designing the user interface, and applying Android development best practices.

9 Conclusion

So the goal of the project is to create a portable software application that provides a reliable assessment of decibel hearing level. The challenges of creating a test that can be comparable to a professional hearing exam have up until this point not been adequately confronted, and there as yet does not exist a satisfactory solution on the market today. We are pleased so far with the prototype and will be conducting audiometric assessments on the phone and finalizing the calibration through real-world testing when the order arrives (the phone is on back order due to its popularity). We hope to reach a lot of users with the application and hope to enable users to seek treatment early and at their own convenience to prevent the onset of debilitating hearing loss. Overall we hope that this application will save users’ time as well as money by eliminating the need to travel to and pay for a preliminary hearing screen.
10 References

http://developer.android.com


Keidel, W D (1973), "[In memoriam Professor Dr.Phil, Dr.Med.h.c. Georg von Békésy]", Zeitschrift für Laryngologie, Rhinologie, Otologie und ihre Grenzgebiete 52 (1): 1–6, 1973 Jan, PMID 4567951


11 Acknowledgements

Our group would like to thank Dr. John Enderle and teaching assistant Sarah Brittain for their continued guidance throughout the design process. As well as Dr. Bernstein and Dr. Oliver for their correspondence and help with planning and development of the application. Thank you.

12 Appendix

Hardware
   Hardware: Android Phone/Tablet
   User interface: Touch Screen
   Hardware interface: Headphones

Software
   Android 2.2.1 or higher
   Maintenance
      Bug Fixes and updates
      Stability improvements
      Processing thread optimization