PhonEtech

/fon-é-tek/

CS4624: Multimedia, Hypertext, and Information Access

Professor Dr. Edward Fox

Rachel Egnor & Tress Cochran

Client: Aries Defense

May 8, 2022
# TABLE OF CONTENTS

**TABLE OF FIGURES**  
3

**TABLE OF TABLES**  
3

1. **ABSTRACT**  
4

2. **INTRODUCTION**  
5

3. **REQUIREMENTS**  
6
   3.1 DYNAMIC APPLICATION  
   3.2 CUSTOMIZABILITY  
   3.3 VIEWABILITY  
8  
10  
12

4. **DESIGN**  
13

5. **IMPLEMENTATION**  
16

6. **TESTING / EVALUATION / ASSESSMENT**  
17

7. **USERS MANUAL**  
19

8. **DEVELOPERS MANUAL**  
23

9. **LESSONS LEARNED**  
25
   9.1 TIMELINE & SCHEDULES  
   9.2 PROBLEMS & SOLUTIONS  
   9.3 FUTURE WORK  
26  
27  
28

10. **ACKNOWLEDGEMENTS**  
31

11. **REFERENCES**  
32
### TABLE OF FIGURES

<table>
<thead>
<tr>
<th>Figure Number &amp; Description</th>
<th>Page Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1: This figure is a display of what the user will first see when opening the application -- the Home Page. The display also shows the record button where the user can record their own voice audio.</td>
<td>19</td>
</tr>
<tr>
<td>Figure 2: This figure is a display of all the audio files the user has generated on the application, in list view for easy access and viewing purposes.</td>
<td>20</td>
</tr>
<tr>
<td>Figure 3: This figure is of the screen display when the user wants to edit the audio file name. Present with a save and cancel button for easy customization.</td>
<td>21</td>
</tr>
<tr>
<td>Figure 4: This figure is a display of the text that the user will show towards their environment to aid in communication.</td>
<td>22</td>
</tr>
</tbody>
</table>

### TABLE OF TABLES

<table>
<thead>
<tr>
<th>Table Number &amp; Description</th>
<th>Page Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1: This table showcases testing percentages and variables discovered during the entire Testing process.</td>
<td>17</td>
</tr>
</tbody>
</table>
1. ABSTRACT

The rise of the COVID-19 disease has brought challenges and opportunities that we have not been faced with before. With the new use of masks in our daily lives, reading lips to communicate with others when the surrounding environment may have increased noise levels, is a simple task that we no longer are able to fully utilize. The elimination of masks, however, does not eliminate the overall issue that trying to communicate in a noisy environment creates. To solve this issue, we have created and implemented an Android based phone application, PhonEtech, intended to facilitate and streamline communication between a variety of users, more specifically, when verbal communication is not ideal. Over the duration of a semester-long time frame, our team has designed, developed, and tested this application idea, leading to a usable and functionable application. The finalized result is a phone app that can be developed further to create a more customizable experience with newer features. PhonEtech is not a replacement for verbal communication but instead an aid to a variety of users faced with various communication issues and roadblocks.
2. INTRODUCTION

Our team is composed of two members, Tress & Rachel, working alongside Aries Defense as our client. We were given the project idea of creating a mobile application to aid in communication between two individuals when verbal communication is not ideal. The idea originated around the example of using an application when in a noisy environment where you cannot hear another user. After initial brainstorming efforts, we expanded this solution to include the usage example of when masks are in use. Over the last 2+ years we have been faced with the rise of the COVID-19 disease which in turn created an overwhelming increase in the use of masks. One con that was unexpected with the increased use of these face coverings is that it prevents people who are communicating with one another from being able to read lips. Many of us don’t realize how heavily we have relied on this part of communication until the masks created this barrier for us. This turned into the other side of our problem. Between these two usage examples we wanted to center our design process around how we could solve these problems.

PhonEtech is an application that is intended to make communication between those around the user much easier. PhonEtech was never intended to be a replacement for verbal communication but instead an aiding resource. Through the initial design phase, we were able to narrow down an overarching idea and focus on a few key details that we wanted our application to focus on and that would allow a user to quickly and easily display a text that the user could communicate to others when it may be hard to hear or speak.

Throughout the entire project process, we were constantly coming back to these two main points to ensure that the initial problem was never lost along the extended time frame and
constant design changes and improvements. Our project is the first step in the creation of this application. Although the finalized product is a working and usable application, there are still many features that we feel could be added to further the development and expand to enhance each user’s experience.
3. REQUIREMENTS

PhonEtech, as an application, will require certain attributes that any good application would have in regards to the development. Our three main requirements were to ensure a dynamic application, include the use of customizable features, and finally require that our application is observable and viewable by any user. Breaking down these requirements, in order to ensure a dynamic application, we must meet smaller milestones and goals that include creating easily updatable features and utilizing a base level application design. The idea that our application revolves around language and communication is where the requirement for customized features stems from. Our application would need to be able to fit the users’ needs through various languages or slang terms that may not be apparent based solely on the developers’ language choice. Finally, this application is centered around the user, the person with the application downloaded onto their phone, displaying the application to those around them in order to aid in communication. This requires the application to be easily understood and viewable in regards to both the aesthetic vision but also the observable vision - ensuring that anyone can read the text on display. These three requirements were set by our team to ensure that our end result would be an application that met both our needs and goals.
3.1 DYNAMIC APPLICATION

The overall process of creating an application typically starts as a solution to a problem that the developer sees technology can fix. Another aspect of this idea formation is that the solution the developer creates will last an extended period of time and not simply be a bandaid to a much larger issue. One thing that we, as developers, are aware of, is that no matter how long term you may see your solution as, there will be a need for a product upgrade simply taking into consideration phone updates and newer models. This idea doesn’t even touch on the subject of the need for updates to the application itself. For this reason, an application must always have a goal of encapsulating some level of dynamic capabilities. One of the easiest ways to ensure that an application can later be changed and improved upon, without reworking the entire basis it was built upon, is to create a framework that is not intended to be a finalized product in the first place. One aspect that our application uses that allows for such dynamic capabilities is the recording feature. This allows the user themselves to ‘upgrade’ the application to their own personal preference and liking by including their own voice or their own phrases that the app may not already have. Within the coding process and the writing of functions and files, Rachel, our main developer, ensured that the application utilized the method of calling various files within the main project page as opposed to utilizing a more dense ‘home page’ or starter project page. This will allow for the future files and updates to be imported into the project development. Instead of rewriting the base code, an added call to said newly imported file will be sufficient. At this level of development, this should be a common practice with the future development of not only this application but any mobile application.
we choose to produce. Within the project, there is a player_sheet file; although currently it is not called from any utilized pages, it is an option that any developer could choose to implement and thereby improve upon the application. The creation of such files allows for our app to continually add building blocks and other files to the project which allows for a certain level of flexibility. Thus, with the addition of one simple call to function or pointer to said file, the application could create an entire new experience for the user.
3.2 CUSTOMIZABILITY

With an application such as PhonEtech, the concept of language and communication being the problem that we are solving, customization should be a given. Language is something that every individual has a limited concept / utilization of. Customizability in regards to our application is as simple as allowing the user to record their own phrases and sentences. Where some may like to use the phrase “may I use the restroom?”, others may prefer “can I use the restroom?” or “can I use your bathroom?”. Although minor changes can produce the same overall meaning and question, the slight change in dialect and verbiage can sometimes hold a much larger impact based on the intended user. The purpose of our application is to aid in communication with those around the user. If a user is in a location of a different dialect than the auto generated audio recordings, a need for customized recording presents itself. Our project and application originated in the Southwest Virginia region, where there may be certain slang terms or phrases that are not found in the Northern Virginia region, where our client is located. In order to make our application more widely accessible and utilized by a larger audience, slang and dialect aside, the need for these customizable audio records and visual text displays is evident. Take a user that wishes to explain directions to another user in the middle of a heavily populated area. User A is currently wearing a mask covering the lower portion of their face preventing User B from reading their lips while giving directions, since noise levels prevent either user from hearing the other. User A wishes to utilize PhonEtech in order to facilitate this conversation. When User A first downloaded PhonEtech, it may have come with a pre-generated phrase that the development team felt was useful such as “take the
second exit of the roundabout”. When User A saw this phrase, they realized that they never called a traffic circle a roundabout, and then used the record option to customize this phrase. In this scenario, the customizability of the phrase leads directly to User A being able to properly communicate with User B. Although a simple feature, this provides any user with the ability to freely speak and communicate without any sort of restrictions that we as developers could have accidentally placed on said user. As of now, the application is strictly in English and only has English phrases or words seeing as this is not only our team’s primary language but also our clients’. One aspect of this application that our team is excited about potentially seeing in the future is the idea of adding multiple different languages, thus adding a whole new element to this customizable application.
3.3 VIEWABILITY

PhonEtech is an application that largely depends on the user and those around the user to be able to visually read the text displays at an easy and fast rate. Many will link this to both size, font, and color of the display and overall layout. The details of the chosen formatting will be discussed in the design section, but it was an important phase of the idea planning process to view this as a requirement and not just a visual preference. This requirement not only includes an aesthetically pleasing application to ensure that a user actively wants to use our application but also includes addressing how certain colors are not always visible to certain individuals with disabilities such as color blindness. A further dive into this topic will be included in the design section of the report.
4. DESIGN

In the first stage of the design process, it was important to truly understand the requirements and specific features that we knew we wanted to include and develop. The most basic design concept was the overall layout that we wanted to implement; this needed to be easily navigated. The application is not overly complex, so it is easy to see that the design itself should be straightforward and one that any user, whether familiar or unfamiliar with the application, can understand. It is also important to utilize a certain color scheme and aesthetic in order to ensure that the text displays are easily viewed. Part of the design also included the overall application name; this is what any user would need to type into an app-store in order to download our application. So, it was important that our application name was both representative of what the app did, and a memorable name that if a user heard about the app from a friend or acquaintance, they would then be able to go home and download the application for their own use.

In many cases, the name and color scheme used for any sort of mobile application, can make or break how well the product is received by users. It is expected that a large majority of the application users will not have any sort of development experience so the initial perception of the application will be based solely on superficial features such as the name or colors accompanied with the layout – not necessarily the accuracy and usability. In our case, Aries Defense wanted to make the decision in regards to the application name. Eric, our main point of contact with Aries Defense, let us know that ‘PhonEtech’ would be the desired title after a conversation with other contacts on his end. This was chosen to reflect directly into not only what the application is intended to be used for but also a
slight play on words to enhance the ‘catchiness’ of an app name, using ‘PhonEtech’ as a comparable word to ‘phonetic’, the sound of a certain speech or sentence.

After the name was chosen, another superficial feature that needed to be discussed was the color scheme. The whole premise of our application was to make communication easier between those who may be hard of hearing and struggle in settings that don’t allow for optimal communication environments. Therefore, we wanted to ensure that we were appealing to all individuals who have a variety of disabilities that prevent them from fully communicating. We weren’t turning the application into a replacement for verbal communication by applying this mindset of appealable to all but instead just trying to ensure a wider audience. Any time color is a topic of conversation, color blindness must be considered. As we were researching the most eye-catching color combinations along with the most visually appealing colors paired finally with colors that were the easiest to read and learn from, the most common color was a lighter hue of blue. After doing all of this research our team consulted our own mobile devices and it was easy to see that we were not the first developers to be using the idea of a light blue application. Looking at your phone, you can see that other applications, such as Twitter, Zoom, Skype, Venmo, Safari, the App Store, and many others all utilize the same color scheme in their app design. This was another step further in the confirmation that we wanted to keep this color on our radar. The final push into confirming a blue color scheme was incorporating the Aries Defense logo into the application. The colors are Red, White, and Blue, making it extremely easy and useful to base our color scheme off our client for more than just branding reasons.
The overall design mockups and storyboards were what we eventually used to solidify the final product that we have today. Once again, the design is centered around the idea of future development continuing on to generate a more complex application when Aries Defense feels it needs to reach that level.
5. IMPLEMENTATION

Our application was implemented solely on the premise of utilizing Android Studio [2] for developmental purposes. This development decision was made for the comfort level of our primary developer due to previous experience working with and creating applications for the platform. This platform was also the most cost effective with neither team members needing to purchase any extra packages or tools to fulfill the project requirements.

Android Studio is a very self-explanatory development studio that made it easy to bring on additional team members regardless of prior experience and knowledge. This studio additionally allowed us to make the application easily accessible from a variety of computer types and application pages. Our team members each had a different type of computer and home base in which they would be viewing the application. Android Studio was the clear choice after discussing all the differences between members, whether it was host environment, experience, or preferred coding language.

The implementation of the actual application process was a series of files that performed various functions and tasks to record and rename the different text media that we wanted to display that were then all called from the main function page.
6. TESTING / EVALUATION / ASSESSMENT

We used our own devices to test the app’s functionality. The metrics we tested were speed and error rate; see Table 1. Speed was on average how long it took for an interaction to have a response, and error rate was how many times a click resulted in the app behaving in an unexpected way. For instance, if a button was clicked and nothing happened. On a Samsung Galaxy FE 20, we had an error rate of 0%, as every button behaved as it should have. On a Google Pixel XL emulator, the error rate was about 10%, as either the emulator dropped inputs or was so slow to load that it timed out. As for speed, on a physical device, every button press resulted in $< 0.5s$ response time. Emulators were slower, averaging about 1.4s. Overall, we were pleased with the app’s ability to respond quickly and accurately. While we did want to test with additional devices, unfortunately time constraints led to testing being done primarily on emulators. While the higher unpredictability of an emulator was undesirable, it was at least good enough as a testing device when faced with limited time. Table 1 shows the metrics recorded across three different devices while using the app, showing the app’s compatibility with different devices. The tester performed three actions for each trial, recording a file, playing the file back, and changing the file name.

<table>
<thead>
<tr>
<th>Device:</th>
<th>Error Rate:</th>
<th>Average Speed:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Samsung Galaxy S20 FE</td>
<td>0%</td>
<td>$&lt; 0.5s$</td>
</tr>
<tr>
<td>Google Pixel XL Emulator</td>
<td>10.3%</td>
<td>1.6s</td>
</tr>
<tr>
<td>Galaxy Nexus API 30</td>
<td>6.7%</td>
<td>1.3s</td>
</tr>
</tbody>
</table>
Table 1. Quantitative metrics while testing the app.

In the future, User Experience (UX) developers themselves testing and giving feedback on the app would be a good path to follow to give the most robust testing results.
Figure 1 shows our home page for the recording feature in which the user can make their experience more customizable and utilize their own voice or a caregiver’s voice in the case that the user could be nonverbal.
Figure 2. List view of available audio files
Figures 2 and 3 showcase more customizable features that were included to enhance the overall experience. Figure 2 is a display of the various text media that can be played. Figure 3 shows the screen displayed when a user is creating a custom name for the media. These were designed specifically for not only the customizable requirement that we gave ourselves but also for organization and easily accessible purposes.
Figure 4 displays how our text would be displayed for the individual text files. This would be the screen that the application user would flash to their intended audience to ensure that they are able to understand and communicate without the use of any sort of verbal cues or phrases.
This project is developed in Android Studio Arctic Fox with Android Studio setting minimum API 21. The project has the default Android Studio project layout, with relevant developer files being in the src/main folder. The functional back end scripts are in the java folder, with front end markup files being in the res/layout folder. Additional assets such as images, icons, and other non-script files are in the drawable folder, and should be accessed via Android Studio’s Resource Manager.

Currently, the Java scripts of note for development are:

- AudioListAdapter.java
  - This checks all the audio files in the directory for this application and applies them in a list, as well as giving each of them a unique clickable behavior.

- AudioListFragment.java
  - This is the screen that displays a user’s audio files and allows them to play the files with the audio file name showing up as text.

- MainActivity.java
  - The “main method” for an Android app; this is the first script called in any app.

- RecordFragment.java
  - This handles the behavior of the record button and navigation to the audio list.

- TimeAgo.java
○ This script handles updating the recording timer to let a user know how long
they have been recording,

Relevant XML files for this project include:

● activity_main.xml
  ○ This is the default screen markup that comes with every app. It currently
    contains a pointer to the record_fragment.

● fragment_audio_list.xml
  ○ This contains the ListView item that displays audio files, using the fragment
    from single_list_item.xml to style each item.

● fragment_record.xml
  ○ This contains the recording page’s images, timer, and two buttons.

● Player_sheet.xml
  ○ This is an optional player that could be implemented but is not currently
    used.

● single_list_item.xml
  ○ Style for individual audio list in app

To import this project, install the latest version of Android Studio Arctic Fox and
drag and drop this folder onto the blank software with no other wizards open. It should
automatically detect the Android application and display its project manager within 5
minutes.
9. LESSONS LEARNED

Overall, anytime there is a project of this magnitude with a time frame of this nature, there will be many opportunities for improvement or areas that you can look back on and consider a different approach or plan. A huge portion of our lessons learned came from the overall makeup of the team and outside forces that were not anticipated at the start of the project.
9.1 TIMELINE & SCHEDULES

The initial timeline that we set up ended up skewed along the course of the entire semester. One aspect that we under-anticipated was the breaks that are already in the calendar year such as spring break as well as Easter. Although our team didn’t have Easter off from our university, it is still a time in which many students choose to go home to spend the holiday, if celebrated, with their family. We also greatly underestimated how the health of a team member would affect the resulting timeline and schedules. When timelines are altered, it is important for our members to try to compensate for this with the addition of meeting times or communication outlets to regroup and get back on track. Unfortunately, the team struggled with this aspect in various stages as well.
9.2 PROBLEMS & SOLUTIONS

The team ran into a variety of issues that all stemmed from the number of team members. This is a project that could and most likely should be accompanied with at least one to two additional team members. Rachel was our primary developer and although fully capable of developing an application that works and can be produced on a much larger scale with improvements pending, it is beneficial to have the aid of another primary developer in the case of bugs or roadblocks to be able to have that streamlined communication that more often than not, improves the workflow of a team. The addition of at least one other team member would have also greatly reduced the stress and complications that arose when one team member got sick and fell behind. In these instances, there was truly only one contributor to our team and with a term long project with various moving parts, it can seem impossible to complete alone. The solution to this problem would have been the addition of a team member which wasn’t quite feasible once the semester had kick started. This is something that any future development teams can avoid by ensuring a larger initial team.
9.3 FUTURE WORK

This application, although completed, has many features that could be added in order to increase the usability and customizations. Throughout the entirety of this project, we have been working closely with Aries Defense to not only formulate the idea but also to bring it to fruition into an application. Aries Defense is a client that falls outside of the Virginia Tech scope and community which will in turn, lead to any further development plans to remain solely with them unless a future partnership is made again with students at Virginia Tech. The team to spearhead this project will be chosen by Aries Defense which may or may not include future students working with Dr. Fox. Regardless of who will continue on with the development of this application, there are quite a few features that would be interesting and useful to see created. Currently the application is much more basic and generic than it could be, leaving plenty of room for future improvements and features.

The first suggested feature is one that has already been mentioned in this report before, the expansion and inclusion of foreign languages. Although not the original intent or goal behind our application, after a semester of implementing the current solution to a noisy environment issue, our team realized this application could be a solution to another problem such as being in an area that does not speak your primary language and needing to utilize a few quick and easy phrases. This could be implemented on a much smaller, non-customizable scale at first. Thus, the application could have various tabs for each language desired and within the tab each text display would be something the developer would have needed to implement already that consisted of the English translation text
display and the foreign language text display. This kind of feature would be beneficial in the sense that it provides the user with a quick, undemanding way of communication, when the most obvious verbal communication might not be applicable. Branching even further into the future could mean some sort of partnership or similarities with Google Translate to customize phrases in another language to use instead of relying solely on phrases the developer deems necessary.

Another feature that would be great to implement would be scrolling text displays. We currently don’t have a strong need for the feature as all of our text displays fit on a mobile phone screen at a size that is still clearly visible to any user. We don’t, however, have a way to display longer phrases in a seamless manner other than just swiping to the next word or phrase. A scrolling text display would eliminate this issue and allow for a much wider variety of sentences and phrases to choose from.

One final feature that our team recommends implementing is a better organization of the text displays on the application. The idea is that a user would continue to utilize the application and thus build up a library full of phrases and text terms they will use. We currently have a simple list format but it will reach a point to which this is not the most sensible organization of the texts.
10. ACKNOWLEDGEMENTS

Our team would like to take the time to acknowledge our client Aries Defense for allowing us to be the team that spearheaded this project, and for working closely with us throughout the semester. We would like to thank Eric Donovan for being our main point of contact within the company, and for not only being a constant source of communication but also embodying a supportive and intellectual leader for the project. Eric was the mind behind the name of the application, giving us something to encompass the work that we put into the mobile application. Eric can be reached at eric@ariesdefense.net for any questions on the further development of the project.

We would also like to acknowledge and thank Professor Edward Fox for being a constant support system for the entire team throughout the duration of this project. Professor Fox allowed us to explore and develop the project in our way and our timeline, while still keeping us accountable for deliverables and a final product we could be proud of.

Our final acknowledgements go to the two teaching assistants overseeing the project as well as the entire classroom full of students who have listened to us present and discuss our project while asking thought provoking and challenging questions to, once again, ensure that our project would have a desired end result.
11. REFERENCES

