

# Simple Prototype Grid Application

## RESNA Design Brief

### Contributors

The following students from Penn State University worked on this project: Abby Riggio, Brad Beltran, Aron Jalbuena, Guohao Shen, Tzu Hsiang Yeh, William Ye, Yuchun Wu

With help from our mentor team of Dr. David McNaughton and Erik Jakobs and RERC on AAC at PSU (see acknowledgements)

### Background

#### Introduction to AAC

AAC or Augmentative and Alternative Communication is any device, tool, or technique that acts as assistive communication [1, 2, 3]. There are two main categories used to describe the disabilities that AAC users may have: congenital and acquired [1]. Congenital refers to most disabilities that someone may be born with including, but not limited to autism spectrum disorder (ASD), developmental disabilities, and learning disabilities [1]. Acquired refers to disabilities acquired throughout life including, but not limited to traumatic brain injuries, neurodegenerative diseases, and temporary conditions [1]. Because every single individual with a disability may require different levels of communication support, AAC offers a broad range of different forms of assistive communication. Most well-known AAC are software designed for tablets that allow the user to communicate via text to speech techniques, which are known as high-tech AAC [1]. However, there are many different alternative communications that are used today each presenting its own benefits and limitations. Sign language, a no-tech AAC, is commonly used among deaf individuals, but not necessarily able-bodied individuals, which can lead to gaps in communication between different communities [1, 3]. Exchange communication, a low-light tech AAC, relies on images exchanged between the communicator and another [1, 3]. This system is effective because the images can be individualized for the user, but it also may be difficult in certain life settings or not applicable to some people [3]. Disabilities exist on a spectrum and some individuals may need constant communication and social support, while some may be completely independent [3]. Exchange communication may work well for a completely nonverbal individual, but may hinder the development of an individual, who is working towards verbal communication [3]. Speech Generating Devices (SGDs) are a higher-tech form of exchange communication because they are able to speak for their user [3]. Ultimately the grid created for this project will aid in creating new speech generating devices.

Modern technology has allowed individuals with disabilities the opportunity to communicate their needs and desires, which is something that wasn't a priority a century ago [2]. Although technology and understanding about disabilities have both come a long way over the years, there will always be room for improvements as well. This is where our capstone, a project with the goal of creating a customizable communication grid software for research, comes in. Many forms of AAC lack complete customizability or the customizable forms are much too expensive [2]. This gap in individualized communication devices is important because everyone should have a non-robotic voice that sounds like them to communicate as well as freedom over color and design choices within the device.

## **Problem Statement**

Individuals with disabilities often rely on various communication methods that are different from “traditional” verbal communication. Typical communication devices are application-based on tablets like iPads, which allow the communicator to push a button or type what they want to say. This will then be spoken aloud by the tablet for the communicator. The idea of these forms of technology is to speak “for” the person who requires it. These devices are sometimes stagnant and unable to be customized for the individual’s needs. This can lead to very robot-like voices for the audio or frustration in the complexities of overall use. The aim of this specific project was to develop a software application to create and access a customizable grid. This will begin to solve pre-existing issues with some assistive communication devices.

The software created will allow the individual to set background colors, spacing, etc. This project will also be used to support the creation of prototype augmentative and alternative communication (AAC) apps for research purposes. After the software is created it will include: a grid interface with a variable number of rows/columns, each key with a reference to a picture/text/audio file, and customizable settings. The other settings considered will be how rounded the key corners are, the ability to create keys within an app, the selection of a key to produce audio, the ability to animate keys when selected, ability to navigate pages, and access the application with row/column scanning. This grid prototype is designed for research purposes with the hope that more customizable AAC will be created to ensure that individuals with disabilities have the opportunity to have personalized audios and visuals for communication better suited to their wants and needs.

## **Methods and Approach**

### **Objectives**

We expected our design to accomplish a touch screen interface where touching a key on the screen will play an audio saying what the image is. It shows a collection of keys in a grid that can be pressed to play the audio behind the key. The keys will all be spread with a set amount of dead space so that they are not all clumped together in the grid. The grid itself is customizable in terms of how many keys per grid, background color, key corner shape, and key audio files. This software will lay the foundation for future avenues of research and future updates by different groups.

### **Xamarin**

Xamarin is a cross-platform programming software. This was used for ease of future integration into different programs such as ios and the Android Play Store. The following grid was programmed using C# as the programming language because it is commonly used in the software development field.

## **Final Approach and Design**

### **Programming**

List of Commands in C# used:

- Key Roundness: **CornerRadius**
- Key Color: **BackgroundColor**
- Key Text: **Button Text**
- Key Images: **Image Source**

- Followed by name of image and row/column wanted
- Text to Speech on Key: `TextToSpeech.SpeakAsync("desired output")`

### Sample Codes

The following sample was the code used to create a key on the grid labeled "elephant":

```

BackgroundColor="Orange">
    <Image Source="elephant.jpg" Grid.Row="0"
Grid.Column="0"></Image>
    <Button Text="Elephant" Grid.Row="1" Grid.Column="0"
Margin="1" Padding="10"
        CornerRadius="20"
        FontSize="Large" FontAttributes="Bold"
        BackgroundColor="Orange"
        Clicked="Say_Elephant"/>
</Grid>
<Grid RowSpacing="2" ColumnSpacing="1" Grid.Row="0"
Grid.Column="3"

```

This section of code was used to create each of the keys within the grid prototype. The last line of code was changed for each key to ensure each key was in the proper place within the grid.

The following code corresponds to **adding new keys into the grid**.

```

async void Add_button(object sender, EventArgs e)
{
    bool response = await DisplayAlert("Add?", "Would you like to add
a button?", "Yes", "No");
    if (response == true)
    {
        Button n = new Button();
        Item.Children.Add(n);
        n.Text = "Zebra";
        n.Margin = 10;
        n.Padding = 10;
        n.CornerRadius = 20;
        n.Clicked += Say_Zebra;
    }
    else
    {
        await DisplayAlert("Alert", "Add button failed.", "OK");
    }
}

```

## **Final Grid GUI**

A screenshot of the final grid design can be seen in the photos portion of the RESNA submission. This sample grid shows many previously programmed keys for different animals. Each key possesses an image, a corresponding text, and a corresponding audio file. The keys are designed as individual small grids, where the top half of the grid houses an image and the bottom half of the grid houses the text. The add button below results in a popup window that prompts the user on creating a new key. If the user selects “Yes” when asked if they would like to create a new key, a new key will popup in an open space within the grid. In this case, the add key was selected, “Yes” was selected, and a new key “ZEBRA” was created in the open portion of the grid as seen at the top left of the image.

The help, style, and add buttons can be seen in both the project video and additional photo. These buttons appear as keys that lay above the keys used to speak aloud various words.

## **Interface Design**

The user is met with an initial startup screen that loads the main menu and its functionalities. The main menu interface consists of a set of rectangularly shaped “keys” with some background space in between. The keys are initially set to some default color. Three keys that aid the user are seen at the top of the grid labeled: add, help, and style.

The “Add” key can be selected to create new keys. A popup window will appear that asks the user if they would like to create a new key. If yes is selected, the new key can be edited as desired by the user. This includes text, an image, and an audio file either generated from text to speech or recorded by the user.

When the user selects the help button, a window is displayed that describes how to use the application. This is a text box that appears on the screen directly over the main menu. This will be a read only interface where the only button that the user can click is the exit button. Any other background space in the text box will not have any functionality. If the user clicks on the exit button, it will return to the main menu and the help text box and exit button will disappear.

When the user selects the “Style” button, the number of keys shown on one page of the grid changes. There are three different settings each with an increasing number of keys shown on the page. Both the “style” and “help” buttons are functionalities seen in the second photo of the RESNA submission photos.

## **Expected Cost**

This grid prototype did not pose a cost to our group. Xamarin, the program used, was free to download and use.

## **Outcome**

Testing was completed to confirm that each functionality worked as promised. Through the software development process, a series of use cases were defined to assist in the testing process as well. These were used to complete a test script. The five main use cases are seen below.

- Case 1: Select key labeled “x” with picture “x”

- Process: Data structure labeled “x” is contacted.
- Output: Plays audio “x”
- Case 2: Select “Style” Key
  - Process: \*not visible to user\*
  - Output: Number of keys within grid changes
- Case 3: Select “Edit Key”
  - Process: Interface with key creation/editing parameters is opened.
  - Output: Can change the audio files, image files, text
- Case 4: Hit “Add Key”
  - Process: Interface with key creation/editing parameters is opened.
  - Output: New keys can be created and stored
    - Sub-cases include creating a key with only audio or only text or both text and audio
- Case 5: Hit “Help” key
  - Process: New window with help instructions
  - Output: Allows the user to better understand functionalities of the app

### **Future Use**

Our group focused on creating a customizable grid prototype. The software is intended to be further improved upon and updated. Future avenues of even more functionality would focus on even more grid pages and more customizability. The software is intended to be implemented to both the iOS app store and the Android Play Store in the future. Our video showcases the grid prototype on an Android device, but as more groups further develop the software, we hope the grid will be accessible on most touch screen devices.

This grid prototype is intended to be the foundation of a highly customizable speech-generating AAC.

### **Significance**

Communication plays important roles in every aspect of daily life. Humans communicate with one another to negotiate, to complete tasks at work, to form relationships, and much more. Communication is much larger than just spoken word. Millions of people worldwide may have speech impairments or be unable to verbally communicate. These individuals often rely on Alternative and Augmentative Communication. AAC offers a wide variety of communication forms. Many forms or devices are known to be stagnant and lack personalization for the user. These devices may often have robotic voices or lack the option of editing. These devices can inhibit both communication development and social development.

AAC users should have devices that can grow with them. Every person has a right to communication. The main goal of our project was to help create an assistive communication grid that can change as the user may need. This grid will grow with the user and can be modified over time. The speech recording option will allow users to choose what their device sounds like. They will also be able to choose colors to their liking.

This app is not meant to be used by a person with a disability as it is currently configured. Our goal was to write the code that would create an easily customizable prototype app that could be used in research activities to explore new device features (e.g., pop-up text when a button is

selected). There are AAC apps in the marketplace that allow an AAC user to add buttons, but those apps are programmed with proprietary code, and the RERC on AAC wanted to be able to manipulate the underlying code to look at new features for grid displays.

### **Acknowledgements**

We wish to thank our sponsors, The Hintz Endowment for Communicative Competence at Penn State University, and the Rehabilitation Engineering Research Center on Augmentative and Alternative Communication (RERC on AAC). The RERC on AAC is funded under a grant from the National Institute on Disability, Independent Living, and Rehabilitation Research (NIDILRR grant #90REGE0014). NIDILRR is a Center within the Administration for Community Living (ACL), Department of Health and Human Services (HHS). The contents of this poster do not necessarily represent the policy of NIDILRR, ACL, HHS, and you should not assume endorsement by the Federal Government. We also wish to acknowledge the support provided by Erik Jakobs (RERC on AAC), and Dr. David McNaughton (Special Education program, Penn State University)

### **References**

- [1] D. Mahmarian, "A History of Autism and AAC," *Medium*, 04-Feb-2016. [Online]. Available: <https://medium.com/@dmahmarian/a-history-of-autism-and-aac-9a2b321b01f0#:~:text=The%20use%20of%20AAC%20began,treat%20individuals%20with%20severe%20disabilities.&text=He%20was%20inspired%20to%20conceive,a%20bell%20to%20communicate%20with.> [Accessed: 10-Feb-2021].
- [2] "Augmentative and Alternative Communication," *American Speech-Language-Hearing Association*. [Online]. Available: [https://www.asha.org/Practice-Portal/Professional-Issues/Augmentative-and-Alternative-Communication/#collapse\\_1](https://www.asha.org/Practice-Portal/Professional-Issues/Augmentative-and-Alternative-Communication/#collapse_1). [Accessed: 09-Apr-2021].
- [3] H. Fleming and D. Hendricks, *Autism: Introduction to Alternative and Augmentative Communication*. [Online]. Available: <https://vcuautismcenter.org/resources/factsheets/printView.cfm/1076>. [Accessed: 10-Feb-2021].