# Coding Project

Spring 2020

Diane Brauner

dianebrauner@me.com

## Background

Mainstream classrooms introduce coding concepts to students in preschool and kindergarten classrooms. The Hour of Code and other coding activities are embedded into the curriculum and coding and robotics clubs are extremely popular. There are numerous coding concept apps available for early elementary students as well as apps that introduce students to basic coding commands. Unfortunately, these mainstream apps are not designed with accessibility in mind. Traditionally, app developers do not incorporate accessibility features that enable students who are visually impaired to be able to access buttons or use commands such as drag-and-drop commands – which can easily be programmed to be compatible with screen readers. Introductory coding apps for elementary and middle school students are often very visual in nature in which the created code animates a character or object that the blind student is unable to see.

In addition, blind students have unique needs when transitioning from traditional tactile graphics to digital graphics and a screen reader. Teachers of the Visually Impaired (TVIs) understand these unique needs; however, the majority of TVIs do not have a background in computer science and are uncomfortable teaching coding. Computer science teachers, who clearly understand coding, do not have experience with teaching students who are visually impaired. Part of the goal of this coding app is to help both TVIs and computer science teachers support blind students learning introductory coding.

## Coding Project Description

The goal of this project is to create a coding app that goes beyond a basic coding concept app - using arrows to create a path through a grid – to moving a Sphero (robotic ball). This fully accessible app would bridge the gap by providing an accessible way to control a robot through the specified course using drag-and-drop commands. Ideally, the training portion of the app will include an accessible 3x4 grid that reflects a basic course design; this grid will encourage students to develop a mental map of the physical course. Students will determine a course through the grid using drag-and-drop commands. The app will then guide the student through various activities designed to teach basic coding skills (such as loops and if/then commands). In the second part of the app, students will learn how to create physical courses and then navigate the Sphero through the course using the available drag-and-drop commands. The robot will ‘act out’ the commands, enabling the blind student to know if he wrote his code correctly.

Note: Currently students with visual impairments are severely under-represented in the computer science field; however, computer science is an excellent career choice for students who are visually impaired!

## Current Accessible Coding Apps

### Accessible Coding Concept Apps

Previously, a UNC student team created CodeQuest, the first fully accessible coding concept app along with 3D printed grids and characters. CodeQuest uses *arrow commands* to navigate a character through an accessible grid. This proof-of-concept app led to an app developer (Sonokids) to develop three simple accessible coding concept apps. (Here is a link to the [Ballyland Code 1: Say Hello app](https://www.perkinselearning.org/technology/blog/ballyland-code-1-say-hello-ios-app). CodeQuest has been adopted by the American Printing House for the Blind and initial field testing of the beta version will begin in January 2020. Videos of the original CodeQuest are included in this [blog post](https://www.perkinselearning.org/technology/blog/hidden-resources-collaborative-projects-universities#comment-12093).

SAS CodeSnaps app incorporates drag-and-drop commands, but an update in 2018 broke the drag-and-drop accessibility. With some modifications, blind students can use the printed and braille labeled QR codes for some basic Sphero activities. For more information go to [CodeSnaps Activity Part 2.](https://www.perkinselearning.org/technology/blog/codesnaps-activity-part-2)

### Accessible Coding App

Apple’s Swift Playgrounds is an accessible coding app; however, it requires pre-requisite skills that most visually impaired students have not yet acquired; most TVIs are not experienced with or comfortable with Swift Playgrounds. Here is [introductory Swift Playgrounds post](https://www.perkinselearning.org/technology/blog/understanding-swift-playgrounds) created by a high school student who is visually impaired about Swift Playgrounds. Tyler describes the Swift Playgrounds ‘world’ layout and demonstrates how a 3D printed and tactile (braille paper) graphics helped him understand the digital layout.

## Coding App Goals

* **The goal of this coding app is to provide a fully functional and accessible coding app that incorporates drag-and-drop commands which will drive the Sphero robot.** This app should provide structured opportunities to build the necessary skills – mental mapping, spatial concepts, tech skills and coding skills – that blind students need in order to be successful in computing. This app will help bridge the gap between basic concepts (such as the Ballyland app) and prepare the student for more complex coding activities, such as Swift Playgrounds. While the need for fully accessible digital materials is certainly critical, the content of this app is the next logical step to enable more visually impaired students to learn to code.

## Purposes

There are three purposes for this coding app project:

* Provide students with visual impairments an accessible coding game using *drag-and-drop commands*.
* Support and encourage TVIs to engage and support their students to actively participate in coding activities.
* Create a proof-of-concept app for app developers, textbook publishers and high-stakes online assessments.