8: Morse code

Due: Noon, March 17

Description

In this assignment, you will create an application that allows user to learn and interact with Morse code. Morse code is a method used in telecommunication to encode text characters as standardized sequences of two different signal durations, called dots and dashes. To learn more about Morse code, you can visit the following wikipedia page:

https://en.wikipedia.org/wiki/Morse_code

As was the case for recent assignments, we will only provide requirements and you must decide how to realize the required functionality while providing a good user experience. As always, good code/project organization and robust code is expected.

Requirements

The app should allow the user to select two different modes:

1. Decoder Mode, which consist of the following:
   A. A rectangle box that a user can tap or long press to imitate a morse code signal.
   B. A button to decode the morse code signal. Upon clicking, color of the button changes to green if the signal has been decoded correctly. It is turned to red otherwise.
   C. A rectangle box that shows all the letters that were decoded correctly.
   D. A button that would clear the text/message in the rectangle mentioned in C.
2. Study Mode, which consists of the following:
   E. This view consist of a flippable card collection (i.e., each card has a front and a back page). The front page consist of a representation of a Morse code. The back page consist of the letter that represents the associated Morse code at the front.
   F. The user can scroll vertically to show different flippable cards.
   G. The user can flip a card by swiping horizontally.

The user should be able to switch modes (e.g., using a back button). A *.json* file is provided that has the letters representation, you can either include the json file in the project OR hardcode it as model data (e.g., a simple dictionary that maps a key string to value string should work); since there is no persistence required in this application.

Bonus (2 pts): Instead of the user having to decode the signal through a button, the app will decode the signal entered by the user after some time (e.g, after 1 second of user not interacting with the rectangle mentioned in 1.A).

Testing

Thoroughly test your app for all possible user interactions you can imagine. Be sure to clean up any compiler or runtime warnings.

Submission

Your submission should be pushed on the master branch. Be sure to verify that your project builds and remove all cruft and compiler warnings.