May 8, 2019
Andrew Baker, Santiago Neira, Pierce Witmer, Hunt Wyman
EE 41440 - Senior Design

# skatEE Final Report

The design requirements we identified for our project are as follows: build a device that mounts on an electric skateboard that allows it to satisfy the requirements of a transportation sharing system, akin to LimeBike; create a mobile application that handles a user interface and is able to send commands to the device mounted on the electric skateboard; build a locking mechanism controlled by the device that releases the electric skateboard controller for use and secures the controller when not in use; implement an information transfer system to receive signals from and send status information to the application user interface. Satisfying these criteria would allow us to create our minimum viable product, to which we can add additional functionality, cosmetic features, and streamlining for user comfort.

## Process:

Our first challenge was learning how to use cellular technology in order to operate our servo remotely. We first experimented with the Adafruit GSM Fona board which had the capability to utilize over-the-air technology but unfortunately ran on the 2G network which is now essentially obsolete so we scrapped it and moved on to the E-Series by Particle.

Since IoT application of cellular technology was not something that we have been previously discussed in classes, we decided that the best idea for our first iteration of the project was to use a kit board available from IoT web developers. This would allow us to work with the front-end, specific to our project engineering aspects rather than worry about the network capacities of a custom board, since the communication protocols for GSM and other radio wave networks were already tested and guaranteed on a board made by a company for the exact purposes we would be using it for. After some searching, we narrowed down our options to two similar products, an AT&T development board called the IoT starter kit and the development board options from Particle, which had a version suited for cellular called the E-Series that adapted their Electron device for ease of use. Staff and user recommendations suggested that the Particle products, web tools, support was reliable and user friendly, so we went with that option.

We had many different objects that needed to be in close proximity in order to function. The solution to combining these various components was a 3D-printed holster that holds the remote and servo lock combination as well as allowing for wiring. The holster and other components are attached via epoxy to avoid damaging the skateboards internal wiring. The particle board itself is lofted on stand-offs to prevent warping or snapping of layered connections. The Particle E-Series includes microcontroller and an antenna, so it can receive functions and store data. The servo is controlled from the Particle E-Series, both are powered by an external 3.7V lipo battery.

One of the major engineering components for our project was the skatEE app. This application for iOS functions as the user interface during a normal “ride interaction”. The three main functionalities that users can perform when communicating with the the skateboard are changing are the locking and unlocking control feature, reading the status of the skatEE device, and locating the board via latitude and longitude coordinates and displaying that location on the Google Maps application.

# Software Architecture:

The interactions between the skatEE app and the skateboard are possible through the use of the Particle E-Series development board and the Particle Cloud Console.  The application itself was built using Apple’s development platform, Xcode, whereas the functions that interact with the E-Series board were written using the Particle Cloud Web IDE.  The two interfaces are then connected using the Particle-iOS SDK (software development kit).  This allows function calls to be invoked from the skatEE App, transferred via the 3G cellular network, and finally received and performed by the E-Series development board which is mounted on the skateboard.

## Google Maps Integration:

To implement the locating function, the skatEE App integrates the Google Maps API. This process required development and integration on both the Particle Console as well as the iOS skatEE App.  The Particle console is responsible for using the Google Geolocation API to triangulate the E-Series board’s position using nearby-cell tower connections.  From this, the latitude and longitude are returned and stored within the skatEE App. Finally, using the Google Maps SDK, the skatEE app is able to return a map depicting the position of the located device.

As of now, the stores all of its key information on the Particle Console. This includes variables such as board status and location, but in the future could be expanded to ride duration and distance traveled, other details that are necessary to calculate the cost of each ride. Ideally, once all of the data parameters were set, they would be stored on a real third party database, such as MongoDB or Amazon's AWS, so that the application could offer secure information storage and reliable retrieval to its users. Although the Particle platform already boasts a billing manager, a third party database would also provide a more effective solution especially if the project scales up.

## Results:

The overall success of this project relies heavily on a few key functionalities. First, the skateboard needs to accommodate the holster and electronics in a compact, efficient way. Although the current design is compact, the efficiency could be improved using custom electronics. Next, the iOS app has to have a way of communicating information to control the servo the right amount. CocoaPods (a dependency manager for Swift) and pulse width modulation directly solve this problem. Both of these solutions were implemented effectively. Lastly, the locking mechanism needs to mechanically allow and disallow access to the holster. The current design is functional, so this functionality also meets the requirements.

# Future Goals:

There are a few major improvements that would allow better integration of the existing systems. First, the electric skateboard would be redesigned from the ground up so that all of the electronics could be integrated into one enclosure under the board. The battery circuitry, cellular module, microprocessor, and power electronics would all be included in the enclosure. Another improvement would be the addition of a proximity sensor that detects when the remote is in the holder. This addition would prevent the rider from stealing a working electric skateboard because the ride only ends when the remote is secured and therefore not usable. Lastly, the GPS integration could be vastly improved to something similar to the Lime Bike GPS features.

This feature would allow the rider to see all skateboards in the area, as well as their charge level so the user knows which are available.