**Touch Control**

**Rafael de Comas and Jon Parcien**

**Embedded System Design Final Comprehensive Report**

**Abstract**

Touch Control will serve as a picture taking application and will be used in the in the event of the user wishing to take a picture without having to physically be touching the phone. Touch Control will offer users the choice of a button which can be used in replacement of the touch screen or buttons on their phone in order to allow them to take photos from a reasonable distance away from their phone. Touch Control will offer an easy solution to picture taking. Jon Parcien and I utilized the Raspberry Pi 3 to operate the application from a distance through the input of a button. We utilized the compiler Xcode and the JavaScript language in order to create an application which can be operated on the mobile Ios operating system. This makes the application work on the iPhone in which the audience we wish to attract use. Jon and I worked as a unit in order to provide an application that complimented the hardware effortlessly and allows the user to exert as little energy necessary in order to get the output they want. The dedication to provide that for our users means the process is not only easy but the photo that is taken is easily accessible and provided in the best quality their phone has to offer. We worked on a weekly basis, ensuring that as each week passed by we had progress to offer whether towards the hardware or the software end as we have now come to the final project. We have provided a simple button that can open users to a whole new world of photography.

**Table of Contents**

1. **Introduction 2-3**
2. **Requirements for Proper Use 3-4**
3. **Background Information and Methodology 5**
4. **Problems and Solutions 6**
5. **Conclusion 6-7**
6. **Introduction**

Touch Control will be extremely useful for not only photographers, but also everyday smart phone users who enjoy taking a good photo of themselves from angles otherwise inaccessible when limited to having to hold your phone. We will be providing users with this privilege for their photo taking needs. With the use of easily accessible hardware and an extremely simplistic design, Touch Control should be easily set up and used by smart phone users of all ages. Jon Parcien and I have designed an application with the goal to combine easy use, simplicity and implementation of all the camera of a smart phone has to offer. The programming done in JavaScript was no easy task, as Jon and I struggled immensely while trying to create a great camera application for our users that interacted seamlessly with the Raspberry Pi’s button on the breadboard. Each week, as we continuously extended the many lines on our program as we attempted to erase errors we would simply be greeted by more. We decided to follow through with Touch Control because we honestly have seen the huge market in social media and more specifically the sharing of photos through these social media applications. We would hope that Touch Control would be very attractive to avid social media users, whether it be Instagram, Facebook, Twitter or any other popular application.

1. **Requirements for Proper Use of Touch Control**

For our users to be able to successfully operate the Touch Control application they simply need a smart phone and the space available to download the application. The application can be used in both the traditional camera application format in which you can take pictures from the touch screen input without any need for the Raspberry Pi and external hardware we designed specifically to operate alongside the application. However, without the breadboard this would simply make our application deliver nothing less than the base camera applications that come along with every smart phone with an operating camera. In order to fully utilize the Touch Control application the user would require a Raspberry Pi 3, a power source for that Raspberry Pi and a button to act as an input. The input button will provide users the ability to take pictures either with flash or without and save that picture taken to both the photos within the phone and also a gallery within the application itself will store all photos taken from the application. Touch Control will operate from a relatively large distance away from the smart phone in use but will experience faultiness if outside of range of the API’s connection. Touch Control’s button operation will not operate unless the Raspberry Pi is powered but will remain able to take photos through input via the touch screen of the phone. Use of the Touch Control application will involve pressing the input button once or twice for a picture taken either with or without flash. The phone will be connected to the Raspberry Pi via an API and the input button will send a signal to the Raspberry Pi and then the smart phone to allow the signal for the picture to be taken to reach the application. The figures below show a small portion of the code within the application in JavaScript and the breadboard with the Raspberry Pi and the button input.

A circuit board

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a cell phone

Description automatically generated

1. **Background Information and Methodology**

Touch Control was innovated as a final project for an Embedded System Design course at Florida Atlantic University in late 2019. It was designed with the requirements of creating a project which would incorporate a microcontroller as a median for input into a final project which would be approved by the professor. Rafael de Comas and Jon Parcien had chose the Raspberry Pi as their microcontroller and utilized the Iphone for their application to run on. The code was implemented on Xcode and Visual Studio Code in order to create a perfect application to coincide with the microcontroller. The framework of our project is React Native and helped provide a base for us to implement our code upon. The code was constructed over many weeks as the application was finally brought to life and following was the code for communication between the application and the Raspberry Pi. The coding language Python was used in order to implement this communication.

1. **Problems and Solutions**

We encountered many problems with the implementation of the code with errors and overall compilation errors with Xcode. The problems were both with Xcode and the code in of itself and had caused for multiple weeks with nothing to show for the application due to many lines of code being added with errors constantly approaching. The original application would eventually be scrapped in favor for starting from scratch on a new application once we decided that the original was encountering too many problems. This worked heavily in our favor as within two weeks the new application was quickly finished and working perfectly. We had also encountered very serious issues when attempting to make a button input which would not be wire connected to the Raspberry Pi and therefore much easier and comfortable for users. Unfortunately, this was not achieved and scrapped in favor of a button on the breadboard which would be powered by and inputted via wire into the Pi. The learning experience provided through having to face and solve these problems is one that has serious shaped us as future engineers.

1. **Conclusion**

In conclusion, we believe that Touch Control was a successful project that fulfilled our plans and ideas. It was a very fulfilling experience to turn our ideas into reality through hard work and teamwork. Though not all objectives were met but our application went above and beyond our original intuition with the button input communicating with the application seamlessly. We are very happy with our project and only wish we could have achieved the input button to be wirelessly communicating to the Raspberry Pi and therefore the application. Looking back I wish we would have scrapped the original application earlier to allow ourselves more time to play with the APIs and create more as it was extremely difficult to test without any application to view results.