2020

Improved Signal Performance for Smartphones

Devin A. Ferrell
Georgia Southern University

Improved Signal Performance Data for Smartphones

An Honors Thesis submitted in partial fulfillment of the requirements for Honors in Electrical and Computer Engineering

By
Devin Ferrell

Under the mentorship of Dr. Fernando Rios-Gutierrez

ABSTRACT

The research project "Improved Signal Performance Data for Smartphones" involves collecting and analyzing data from a smartphone's available connections and making the information available in a simple application interface. Due to the data collection/analysis and iOS system framework, several ethical and professional issues will be encountered ranging from individual privacy to company standards/guidelines. These conflicts are addressed, and solutions are provided, along with researching into similar situations that others have reported. By making comparisons between this project and similar approaches, professional and ethical areas of concern can be addressed early into development, or completely avoided. The research project course was properly setup to provide a time frame, due dates, and guidelines for implementing each stage of the project. Project goals are explored in detail. All required components for the project implementation are listed, along with their functionality and availability for use. The final product was a program that accurately derived various characteristic from connections and displayed this data in tables and graphs for consumers to make use of.

Thesis Mentor:_____

Dr. Fernando Rios-Gutierrez

Honors Director:_____

Dr. Steven Engel

April 2020
Electrical and Computer Engineering
University Honors Program
Georgia Southern University

ACKNOWLEDGEMENTS

Special thanks to Dr. Fernando Rios-Gutierrez for his guidance through this project, along with all the Electrical and Computer Engineering faculty at Georgia Southern University for their educational instruction. In addition, special thanks to the staff at Zach S. Henderson Library for their assistance with the on-campus Mac computer systems and adding the necessary programs like Xcode to them.

INTRODUCTION

The proposed project involved analyzing available data connections (Wi-fi, LTE, 4G, 3G, etc.) and providing this information in an easy-to-read format through a program application. Important connection details included data packet transmission rate (sending vs: receiving), bandwidth/frequency, packet loss, and comparison to other available sources (Wi-fi, 3G, 4G, LTE, etc.). Due to recent events restricting access to a Mac computer, a service called MacinCloud.com was being utilized to create a virtual Mac environment on a Windows computer. The benefit of this system is the access to Xcode. Xcode is an Apple specific software that allows for the straightforward design and virtual testing of an application. It used the Swift programming language, which is similar in structure to Object Oriented Java, to code the application. Anytime throughout the process, the program can be simulated in a virtual iPhone environment. After the application was complete, it was submitted several times through the Apple Development Program for analysis, review, and edit suggestions.

LITERATURE REVIEW

Due to the project's involvement with digital technology and data collection, there are many people whose projects have a similar scope. These other situations have faced their own ethical or professional issues that this project may face, or at least be related to.

Experimentation has been done in improved methods of collecting health information from individuals. A common avenue of obtaining information has been from smartphones and wearables such as smartwatches. Devices such as these "continually collect and analyze a wide range of data types" obtained from built-in sensors [1]. Since this data is obtained without interaction from the individual, its known as passive data. Although convenient, passive data collection of health information provokes concern over the exact data that is being collected. Since the passive data collection is usually automatic, "participants may not be aware of the type, amount, or implications of the data that is collected" [1]. If not constantly under scrutiny, these services could be collecting irrelevant health information for the prescribed treatment. In addition, this information could easily be distributed to companies, where they could personalize advertisements or refuse service based upon the obtained health statistics.

Due to the increasing role of the internet and communication, data has become one the most valuable resources for any organization. It has become so valuable, that many online services have been labeled as free. However, they are not truly free, since the currency cost has been replaced with data collection. Many such services are required to generate "a request for permission to track certain information" where the consumer must "voluntarily allow it as a convenience" [2]. This is usually done in the form of a terms of service agreement. Unfortunately, many consumers neglect to read the

agreement and instantly sign away their privacy rights. Technically this process eliminates the legal side of the data collection ethical issue, although one could argue that the consumer was unaware of what they were agreeing to. Even when "unintentionally giving away permission to data privacy," a signed agreement relinquishing responsibility for an individual's privacy adhere to professional standards for digital interactions [2].

Backlash due to the improper management of professional and ethical issues can have a unique effect on the future success of an organization or individual. For instance, the USA population is vocally against the collection of personal data by the government, but "have a more favorable view of businesses' use of data collection and analysis" [3]. In respect to this project, consumers are more likely to give their consent for the data collection that is required, rather than if it was government sponsored. Due to this, the overall consumer impact can be predicted to be much greater, since more people are willing to agree to the mandatory privacy policy and terms of service.

BACKGROUND

The Swift programming language is the standard upon with most iOS systems rely on. An object-oriented language, Swift runs off a structured block layout, where each block of code references a GUI interface object. Some key features of this language include "property wrappers for customized access to property values, support for handling collections of related types, and a text-based interface" that allows for easy code to interface development [4]. The program syntax was built to be able to understand easily by beginning programmers. Therefore, Swift is an extremely high-level language, in that its as close to human language as possible.

```
import UIKit

class ViewController: UIViewController {

    @IBOutlet weak var myLabel: UILabel!

    @IBAction func myButton(sender: AnyObject) {

        myLabel.text = "It worked!"
    }
    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view, typically from a nib.

        print("Hello world!")
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
        // Dispose of any resources that can be recreated.
    }


}
```

Figure 1: Example Swift Structure

One disadvantage between Swift and other programming languages is that variables are not allowed to hold NULL values, so every variable must be defined upon initialization. In addition, it was developed relatively recently compared to other languages, so not many programmers are fluent in Swift. Due to the language being specialized for iOS systems, it has very little compatibility with third party applications and other systems. Despite these drawbacks, Swift is the best language for iOS development and is relatively easy to learn.

PROFESSIONAL ISSUES

Since the proposed application was executed through the Xcode environment and is aimed at iPhone users, it must follow professional guidelines that have been set forth by Apple. These guidelines can be broken down into five sections: Safety, Performance, Business, Design, and Legal. Users of this application should be comfortable using this

app, and no encounter rude or damaging content such as “offensive, device damaging, physical harm, or discriminatory” elements [4]. In addition, the appropriate age rating should be clearly visible. This application doesn’t include such content, so is safe for operation by all age groups.

Any applications that are submitted for the review process should be functioning well and performing as expected during tests. Incomplete applications such as “demos, betas, and trial versions” are excluded from being reviewed by the Apple Developer Program [4]. Hardware and software requirement must be met, and any reliance on third party sources is invalid. By utilizing the virtual simulation environment in Xcode, this application was tested multiple times to verify its functionality before final submission. In addition, minimal physical operation tests were performed using an iPhone and connecting cable, since outside variables may not be considered during the virtual simulations. These measures made sure that the final application was functional.

For any exchange of currency, a detailed description must be given to the consumer so that they fully understand what it is that they are purchasing. Any misrepresentation of a product can be met with legal problems and rejection/removal all Apple affiliated devices. Also, advertisements in an application must be age appropriate and are subject to the same currency guidelines as the program itself. The proposed application is not intended to be profitable and is an open source code available for anyone to access or improve upon. In addition, no advertisements are displayed while the app is operating. Due to this set-up, currency exchange and advertisement restrictions are not applicable issues that this project faced.

Applications are encouraged to be “simple, refined, innovative, and easy-to-use” in order to improve the overall quality of the consumers’ experiences [4]. It must continue to meet these standards while its available on any form of device. Blatant copying of existing applications should be avoided, as it infringes on copyright laws. While well-designed and innovated apps are encouraged, they should remain easy-to-use for the average person. Most of this application’s operations are performed in the background, out of sight from the consumer. They see a well-designed interface that lists the specified connection name, along with the resulting characteristic values as data tables or graphs.

Any data or contents accessed by an application must be clearly stated in a privacy policy, and the collected data must be used only for its specific purpose. Illegal activities such as “gaming, gambling, and lotteries” must be officially sponsored by the app developer, or valid organization [4]. Any data that the proposed application accesses will be clearly stated in the privacy policy to avoid such legal issues. None the of collected data will be distributed to third-party platforms, and its use will be strictly for the application’s performance. No currency exchanges will be included in this application, so any monetary restrictions are inapplicable.

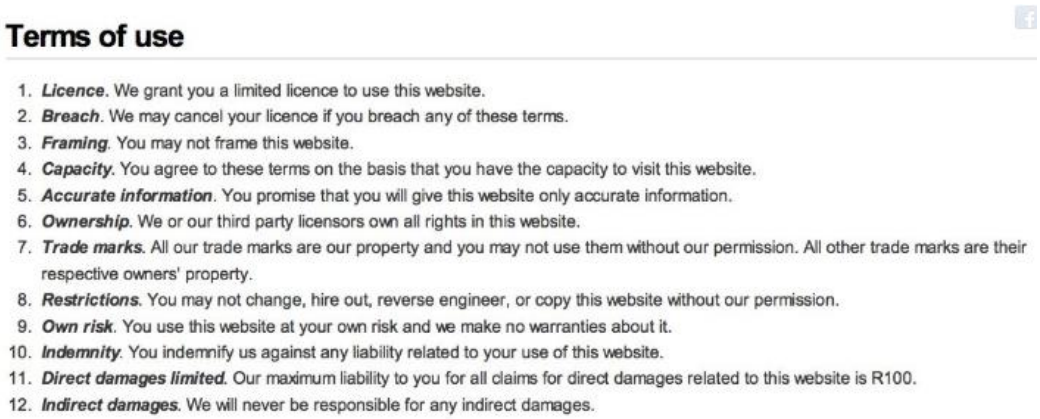
ETHICAL ISSUES

Due to the data collection and online connectivity of the application, the project does intrude into some major ethical issues involving the privacy of individuals’ information. Some standard ethics guidelines for engineer developers can be see in the below figure.

Public	Software engineers shall act consistently with the public interest.
Client and Employer	Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest
Product	Software engineers shall ensure that their products and related modifications meet the highest professional standards possible
Judgment	Software engineers shall maintain integrity and independence in their professional judgment.
Management	Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
Profession	Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
Colleagues	Software engineers shall be fair to and supportive of their colleagues.
Self	Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

Figure 2: Ethics Code

The main ethical concern results from the signal connectivity analysis the application will perform. Since performance data from the individual's iPhone and available connections will be required for the application, each individual user's consent must be acquired for the application to function as described. In order to do this, the most reliable approach was to create a terms of service agreement. The agreement lists the specific data that is being utilized from the user's device, along with any privacy issues such as third-party access that must be addressed. Below are some areas of terms of use that were considered for this application.



Terms of use

1. ***Licence***. We grant you a limited licence to use this website.
2. ***Breach***. We may cancel your licence if you breach any of these terms.
3. ***Framing***. You may not frame this website.
4. ***Capacity***. You agree to these terms on the basis that you have the capacity to visit this website.
5. ***Accurate information***. You promise that you will give this website only accurate information.
6. ***Ownership***. We or our third party licensors own all rights in this website.
7. ***Trade marks***. All our trade marks are our property and you may not use them without our permission. All other trade marks are their respective owners' property.
8. ***Restrictions***. You may not change, hire out, reverse engineer, or copy this website without our permission.
9. ***Own risk***. You use this website at your own risk and we make no warranties about it.
10. ***Indemnity***. You indemnify us against any liability related to your use of this website.
11. ***Direct damages limited***. Our maximum liability to you for all claims for direct damages related to this website is R100.
12. ***Indirect damages***. We will never be responsible for any indirect damages.

Figure 3: Areas of Concern for Terms of Service Agreements

However, this project didn't involve distributing an individual's data to others, so the involuntary sharing of user data was not a major concern. By providing a mandatory terms and conditions agreement before allowing access to the application, this project avoided the issue of unauthorized data access. Also, no ethical third-party clause was violated since the project is isolated to just the operation of the application.

Many ethical issues can stem from monetary gain, such as improperly advertising services/subscriptions or not maintaining valuable functions that consumers pay for. While such practices are discouraged, confronting the source of such problems is generally out of reach for the average consumer. Therefore, these practices exploit the economic and social position of consumers. To avoid such an ethical dilemma, it has been decided that this application's source code will be provided to the public free of charge, and without the presence of advertisements. Th goal of the project is to improve the quality of life for users utilizing transmission signals, not to gain monetary value from situations where they are suffering. Therefore, this profiting area of ethical concern will be avoided entirely in the scope of this proposed project.

PROJECT OBJECTIVES

SMART goal setting was a guideline for planning out the project by analyzing the details of each research component. The five components of this method are “specific, measurable, attainable, realistic, and timely” research goals [5].

Setting specific goals for a project involve “answering six questions: Who is involved? What do I want to accomplish?, Where is it happening?, When will it happen?, Which parts are required?, Why is this happening?” [5]. In response to this current project, I was the sole researcher/contributor to the overall deliverables, the schedule of which can be seen below in the form of a Gantt chart.

Project Name	Improved Signal Performance for Smartphones

Approved By	Dr. Fernando Rios-Gutierrez

Design Team	Names
Team Leader	Devin Ferrell
Project Designer 1	Devin Ferrell
Project Designer 2	Devin Ferrell
Project Designer3	Devin Ferrell

Tasks	Start Week	Duration in Weeks
Gather/Set-up Materials for Testing	1	1
Research/Test Methods of Measuring Signal Data	2	1
Compile Automated Programs for Data Collection	3	5
Set-up Basic App Structure	8	2
Create Interface Between Data and App	10	4
Debug Interface	14	4
Final Version of App Compiled/Tested	18	2
Research Paper	20	4

Figure 4: Gantt Chart Schedule

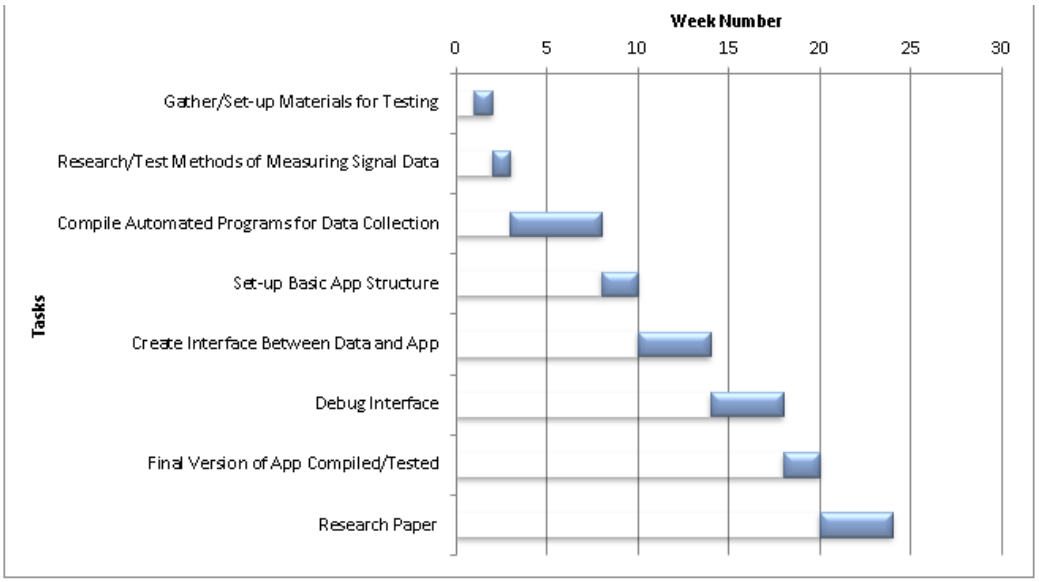


Figure 5: Development Schedule

I wished to accomplish the goal of creating an Apple compatible application that provides the user with detailed network performance data that the current iOS system doesn't offer. This project took place at Georgia Southern University, through the Senior Design class and Electrical Engineering Department. Due the university's temporary closure, the final stages of the project were completed at home while under lockdown. The successful project has a programming environment that is suitable to an IOS system, and fully functioned as a signal data collection process.

Measuring the progression of a project involves keeping track of what has been completed and current work, along with due dates and presentations. This process has followed due dates are provided by the affiliated professor. His guidance has served as a guide to the project progression by measuring how tasks are proceeding, along with when they are due.

Most projects are not able to be started right away. Usually time must be spent completing background research and learning new skills to begin work on the actual project. In addition, financial position must be in good standing to fund all aspects of the project, or loans must be obtained for funding.

Although many programming languages have been taught throughout the Electrical Engineering course such as Java and C++, programming for IOS devices is restrictive to specific languages due to oversight by Apple. To program a compatible application, the most straightforward approach was to use the Swift programming language, which knowledge was lacked in. Therefore, time had to be spend becoming familiar with the basic intricacies of the language, so a rough draft of the program could be written. No loans had to be taken for this project, as the only money expenditures included the rental of the Mac iCloud server and the purchase of the Apple Developer package. These two products allow for the creation, testing, and final submission of the application.

To set a realistic goal, one must consider how much they are able to work on the project, and just how high the goals should be set. Multiple factors can affect the ability to get work done such as team project size along with additional responsibilities outside of the project scope. Due to only one person working on the project, the overall goals had to been set wisely, since there are no team mates to support the workload.

METHEDOLOGY

Due to this being a solo project, all the responsibilities of the project execution were handled by myself. From time management to programming, to the final implementation.

The main stages of the project followed the development of the application's logic. The first stage was to gather information on how to obtain signal data from an application standpoint, especially when dealing with iOS systems. Then, a basic understanding of the swift programming language had to be developed, due to the fact that no courses at Georgia Southern University offered Swift as a learning objective. After spending several weeks completing self-study of the Swift programming language, an attempt at the first draft program was made, an comprised nearly a month of work to get functioning. One benefit of using the Xcode compiler was that the application interface is developed alongside the program, so it can either be hardcoded, or imported GUI elements automatically add a default method field to the program. This saved a large amount of time by being able to work on both aspects of the application at once. After the initial draft of the program was completed, I took some time off programming to further learn more about the Swift language and its compatible functions with iOS. Using this new information and my experience with creating the original program, the next three months consisted of debugging, code rewrites to improve efficiency, and overall clean up of the interface and code presentation.

The required materials for this project consisted of a laptop, Mac environment, access to the Xcode compiler, an iPhone, and the Apple Developer program. My laptop and the laboratory computers run on the Windows operating system, so most of the

project development took place inside the Georgia Southern University library, which owns about a dozen Mac computers. A maintenance request had to be made in order to have Xcode installed on the Mac computers. By utilizing the library computers, access was gained to the Xcode program, which is a compiler that allows easy testing for iPhone applications in a virtual environment. The simulated environment that Xcode provides can be seen below.

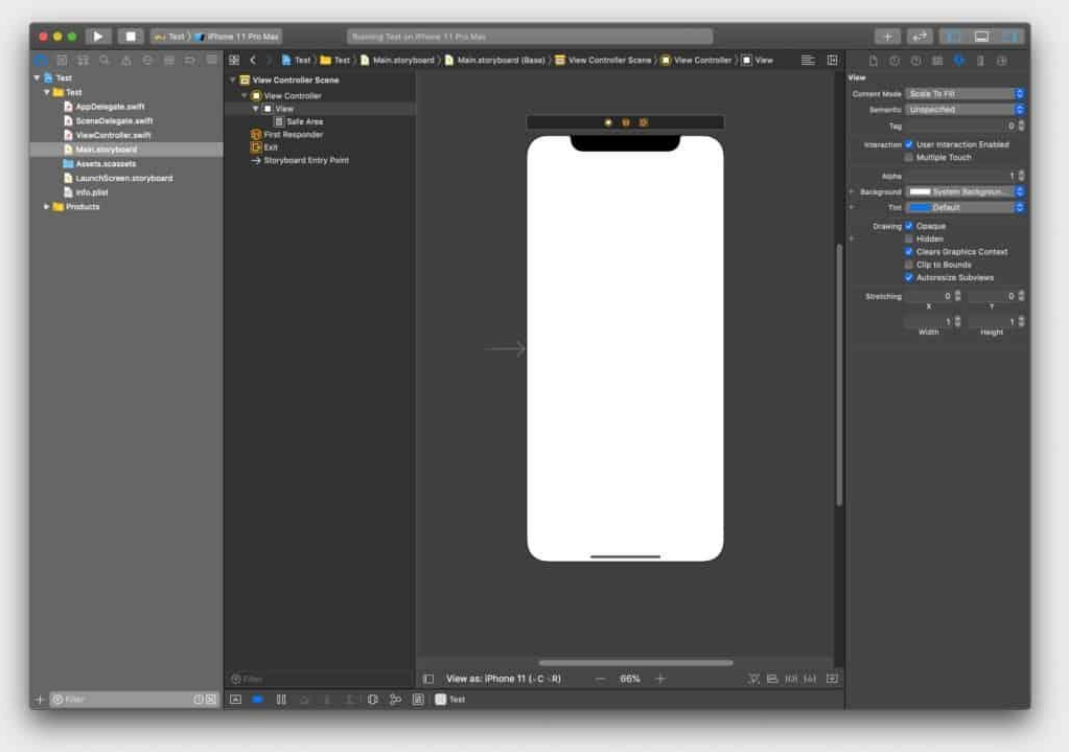


Figure 6: Xcode iPhone Simulation

However, the closure of Georgia Southern University in March 2020 eliminated my access to Mac computers. To circumvent this issue, a temporary subscription to MacInCloud was necessary. This service provides a virtual machine that runs off a collection of Mac computers, allowing access to the Xcode compiler on my Windows

laptop. However, physical testing was not a capability that the remote connection allowed, so the simulated environment Xcode provided was relied on for program testing purposes.

Lastly, the Apple Developer Program provided feedback and proposed improvement on written applications, and even can add applications to the Apple store. Although the original scope of the project included this upload to the store, staffing issues causing little to no communication from the developer program and the inability to perform further physical tests to satisfy the necessary requirements due to the pandemic prevented this. I would like to continue this work and successfully have the application uploaded once quarantine has ended.

RESULTS/DISCUSSION

Due to iOS restrictions, users must grant permission for background refresh to occur. Otherwise data can only be collected when the application is open. If connections are available, test methods are executed to determine signal data. Retrieved data is stored for a time period, allowing table and graphical displays to be generated to represent overall signal performance. The product was able to successfully generate signal characteristic data from a provided connection. A single connection test data table can be seen in the below figure.

Signal Connection: Amp-ed_B1912_C288

Signal Description	Data Value
Signal	Amp-ed_B1912_C288
Signal Type	WiFi
Download	53.11 Mb per second
Upload	23.47 Mb per second
Bandwidth	22.17 MHz
Frequency	5.78 MHz
Packet Loss	2.35%

Figure 7: Connection Test Data Table

This information is stored for a desired length of time, during which it can be graphically compared to other connection instances. These graphical displays or data tables are generated based upon user preference, providing displays comparing various signal characteristics by showing multiple data collection events side by side.

The project was completed without extraneous spending. Due to recent events causing a loss of access to university Mac computers, a remote server connection was required. An affordable monthly payment for remote Mac access allowed continue project development. Figure 8 contains the expenditures this project encountered.

Component	Average Cost
Laptop	N/A
Remote Cloud Server	$20/month for 3 months
Xcode	Free
IPhone 8	N/A
Apple Developer	$99
Total	**$159**

Figure 8: Expenditures

Many useful capabilities were discovered through the development of the project, other than just the program itself. This project process introduced me to the Xcode compiler. The Xcode environment provided many useful capabilities for designing application interfaces, such as being able to position GUI elements in relation to one another prevented misaligned text boxes and displays when sizes were altered. In addition, the simulated iOS environment that the compiler provided was extremely useful for the quick testing of processes, although it wasn't as reliable as physical testing.

Although programming in Swift promoted iOS compatibility, the process of calculating data statistics for a connection proved to be a greater challenge than in languages like Java or C. Reliance on multiple individual libraries was required to complete this process. In addition, most of the Swift syntax is drastically different than other languages that I've learned here at Georgia Southern (Java, C++, MATLAB, etc.).

Several deliverables were presented through the development of the research project. Initial project proposals, ethics reviews, formal presentation, and research reports were generated and displayed for my peers to review. This information was uploaded to a website created for this research project, along with access to the public source code materials.

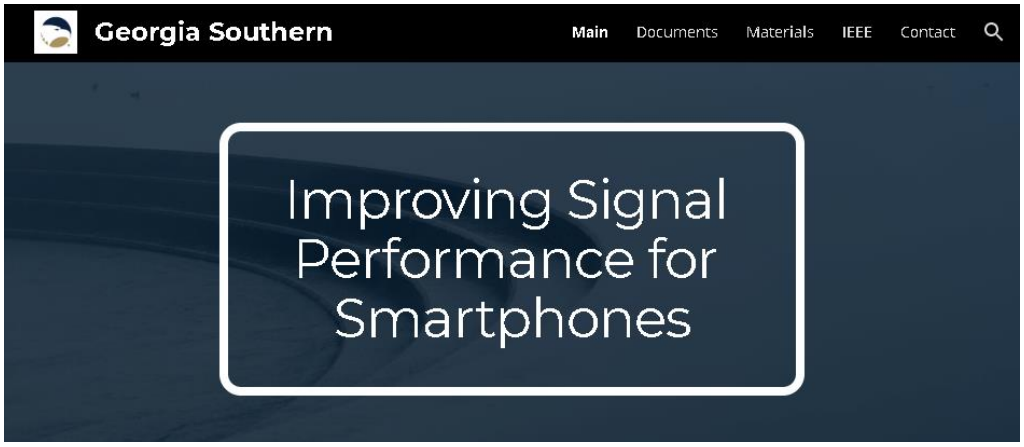


Figure 9: Project Website

The basic structure for a website was completed using Google Sites. This provided the public with information relative to the project such as its purpose, goals, and current progress. In addition, explanations of all related components are provided, along with graphs, data, and program code that has been generated. The website can be found at https://sites.google.com/view/devinferrell.

CONCLUSION

Through analyzing Xcode and Apple's professional guidelines and ethical concerns that this proposed application faces, all issues that might arise are confronted, and applicable issues are met with a proposed solution. The main issues applicable to this project were privacy, data usage, performance, and design. Privacy and data usage issues were dealt with by providing a clear privacy policy. In addition, a terms and service agreement were mandatory for every user before they can use the application. These measures satisfied the privacy and data needs. For design and performance, extensive testing was conducted utilizing multiple simulations to verify the applications functionality before final submission. By making sure to perform these assignments, the project faced no professional or ethical issues along with their accompanying ramifications. Although the program application has proven to compile and perform successfully after many tests, recent events have barred access to a physical Mac computer for testing and uploading purposes. As an alternative approach, a remote connection to a Mac environment was obtained, although this drastically limited testing option to mainly simulations of the runtime application interface. Nevertheless, the final implementation demonstrates the ability to derive signal characteristic data, store the data, and display the performance statistics to the user.

In the future, I hope to perform further work on this project by exporting it as an official Apple application. The unavoidable loss of access to a physical testing environment and cutbacks from the Apple Developer Program as a result of current events forced the project's focus to be redefined.

WORKS CITED

[1] Maher, N., Senders, J., Hulsbergen, A., Lamba, N., Parker, M., Onnela, J., Bredenord, A., Smith, T. and Broekman, M. (2019). Passive data collection and use in healthcare: A systematic review of ethic...: Discover @ Georgia Southern. [online] Eds.b.ebscohost.com. Available at: https://eds.b.ebscohost.com/eds/detail/detail?vid=0&sid=d2a77c55-23ad-422c-96ff-7fd5a40c605d%40sessionmgr101&bdata=JkF1dGhUeXBlPWlwLHNoaWI%3d#AN=S1386505619302527&db=edselp [Accessed 18 Nov. 2019].

[2] Allen, J. (2019). The Rapid Change of Privacy in the Digital Age.: Discover @ Georgia Southern. [online] Eds.b.ebscohost.com. Available at: https://eds.b.ebscohost.com/eds/pdfviewer/pdfviewer?vid=1&sid=aaa63648-f0b1-4a90-9b59-54e3d3a009ab%40pdc-v-sessmgr02 [Accessed 18 Nov. 2019].

[3] LaBrie, R., Steinke, G., Li, X. and Cazier, J. (2019). Big data analytics sentiment: US-China reaction to data collection by business and government. [online] ScienceDirect. Available at: https://www.sciencedirect.com/science/article/pii/S0040162517308612?via%3Dihub [Accessed 18 Nov. 2019].

[4] Developer.apple.com. (2019). App Store Review Guidelines - Apple Developer. [online] Available at: https://developer.apple.com/app-store/review/guidelines/ [Accessed 18 Nov. 2019].

[5] Topachievement.com. (2019). Creating S.M.A.R.T. Goals — Top Achievement. [online] Available at: http://topachievement.com/smart.html [Accessed 7 Dec. 2019].

[6] GeeksforGeeks. 2020. Introduction To Swift Programming - Geeksforgeeks. [online] Available at: <https://www.geeksforgeeks.org/introduction-to-swift-programming/> [Accessed 10 January 2020].