

Wireless Charging SmartBag

by

Cole Skipper & Mihir Patel

Submitted to
the Faculty of the School of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

© Copyright 2020 Cole Skipper, Mihir Patel

The author grants to the School of Information Technology permission
to reproduce and distribute copies of this document in whole or in part.

Cole Skipper

Cole Skipper

4/14/20

Date

Mihir Patel

Mihir Patel

4/14/20

Date

Ryan Moore

Ryan Moore, Faculty Advisor

4/14/20

Date

University of Cincinnati
College of
Education, Criminal Justice, and Human Services

April 2020



Prepared by
Cole Skipper & Mihir Patel

Students of
University of Cincinnati
College of Education, Criminal Justice, and Human Services
School of Information Technology

April 14, 2020

TABLE OF CONTENTS

List of Illustrations.....	iii
TABLES	iii
FIGURES	iii
ACRONYMS AND ABBREVIATIONS.....	Error! Bookmark not defined.
ABSTRACT.....	Error! Bookmark not defined.
1. PROBLEM STATEMENT	Error! Bookmark not defined.
1.1 Background.....	Error! Bookmark not defined.
1.2 Project Description.....	3
1.3 Problem	Error! Bookmark not defined.-3
1.4 Solution.....	Error! Bookmark not defined.-4
1.5 User Profile	5
Project.....	6
Potential Users	6
Software, Interface, and Related Experience.....	7
Experience with Similar Applications.....	8
Task Experience	9
Frequency of Use.....	9
Key Interface Design Requirements that the Profile Suggests.....	9
1.6 Use Case Diagram	10
2. PROJECT MANAGEMENT.....	11
2.1 Budget.....	11
2.2 Objectives/Deliverables.....	11-12
2.3 Project Schedule	13-16
2.4 Gantt Chart.....	17-18
3. TECHNICAL ELEMENTS/DISCUSSION	21
3.1 Network	19
3.2 Application.....	19
3.3 Physical Hardware Design.....	19
4. VISUALS	21
4.1 Network Architecture	21
4.2 Hardware Implementation.....	22
4.3 User Interface	23
5. TEST PLAN	24
5.1 Overview/Methodology	24
5.2 Scope Of Testing.....	24
5.3 Objectives.....	24
5.4 Test Requirements and Procedures.....	24
5.5 Test Result.....	24
5.6 What we Learned during Testing.....	24
5.7 Problems Encountered and Analysis of Problems Solved.....	32
5.8 Future Recommendations/Recommendations for Improvement.....	33
6. CONCLUSION	25

6.1 Fall Semester 2019..... 25-27

6.2 Spring Semester 2020..... 27

APPENDIX A. ADDITIONAL INFORMATION.....Error! Bookmark not defined.

APPENDIX B. REFERENCES.....Error! Bookmark not defined.

List of Illustrations

TABLES

<u>No.</u>		<u>Page</u>
Table 1.	Budget.....	11
Table 2.	Work Breakdown Structure.....	13-16
Table 3.	Usability Test.....	28
Table 4.	Compatibility Test.....	28
Table 5.	Functionality Test.....	29-30

FIGURES

<u>No.</u>		<u>Page</u>
Figure 1.	User Profile.....	6
Figure 2.	Use Case Diagram.....	10
Figure 3.	Gantt Chart Fall/Spring Semester 2019-2020.....	17
Figure 4.	Network Architecture.....	21
Figure 5.	Hardware Implementation.....	22
Figure 6.	User Interface.....	23

ACRONYMS AND ABBREVIATIONS

AT Commands	Attention Commands
API	Application Programming Interface
GPIO	General Purpose Input/Output
GSM	Global System for Mobile communication
GPS	Global Positioning System
IDE	Integrated Development Environment
SIM	Subscriber Identification Module
SIM868	Subscriber Identification Module 868

ABSTRACT

The Wireless Charging SmartBag is the ultimate solution for users on the go, who fail to find time to recharge devices. The combination of proprietary smart device connection cables, adapters, and battery banks into this SmartBag can substantially increase the freedom and organization of users of Qi enabled devices. Current market research shows no viable all-in-one solution that provides this ease-of-use. According to Cisco VNI, the number of networked devices per person is expected to jump to fourteen by 2022. This paper discusses the work, validation, and development of the resulting fully inclusive system utilizing a mobile application to allow users to view battery percentage data for each registered device including the backpack itself, as well as the ability to prioritize device charging based on need and available charging capacity. Additional features include a SmartBag charging dock, GPS tracking, and Global Wifi HotSpot.

1. PROBLEM STATEMENT

1.1 Background

Today, we have numerous purchasing options to choose from when shopping for smart devices. There are several different platforms that all utilize their own proprietary forms of power adapters, requiring specific voltage and amperage ratings, along with various connection cable types. There is a need for the majority of consumers of these products to be able to recharge those devices when no outlets are available, or access is limited. When a user owns devices across different platforms, the number of required items needed to charge is doubled or even tripled. Remembering to bring a battery bank if no outlet were to be available adds another inconvenience to the users of smart devices, not to mention the mess of cabling and wasted space in their bag, purse, or backpack. In order to completely remove these inconveniences, a fully integrated solution that offers multi-platform support is needed. The Wireless Charging SmartBag will increase the freedom of its users by incorporating the multi-platform compatible Qi standard technology into commonly carried bags, purses, and backpacks.

1.2 Project Description

The Wireless Charging SmartBag will incorporate several technologies into common on-the-go items: backpacks, purses, etc. Eliminating proprietary adapters, combining proprietary cables, integrating Qi standard charging coils into pockets, and placing up to sixty-four amp hours of extra power into a commonly carried bag/pack, will solve the problem outlined above. A contact connection AC adapter and optional wall mount to recharge the base units will be incorporated into the design provide a better

overall user experience. Taking into consideration several large groups of users has been a key factor in shaping the design and scope of this project. Hikers, bikers, camping experts, city life enthusiasts, business travelers, and students on college campuses are some of the key user groups. Factors like safety, weight, waterproofing, number of Qi enabled pockets, and battery percentage information, as well as available charge capacity are a product of considerations of these user groups. A mobile app will enable users to view battery levels, as users can prioritize devices to be charged via easy to access switches, based on need.

Prioritization of multiple devices to ensure that the devices that are in critical need of being charged can be given higher priority in situations where there is only enough backpack charge capacity left to charge one device sufficiently. With the inclusion of GPS capabilities, last known location can be used to find the backpack if it is lost or left behind. Global WiFi is also a possibility with the addition of a sim card. A docking station that uses a contact-based charging system will be used when the user is back at home. We would potentially offer a wall mount that incorporates an AC adapter that will charge the backpack/bag, via contact nodes located in the handle/shank of the back or located near the actual battery pack units at the bottom of the bag. The No-Frills Wireless Charging Backpack will also include a Raspberry Pi for the purpose of providing data exchange and processing between the devices, the Qi charging coils and the mobile app.

1.3 Problem

The number of electronics carried by a single individual has increased tenfold over the past decade. According to Cisco VNI, there are around 8 networked devices per person, which is expected to jump to approximately 14 per person by 2022. Changes in

charging port types and the addition of wireless charging capabilities being integrated into all of the newest devices has posed consumers with many ways of recharging these devices. On top of this, companies often tend to push their own proprietary connection types as a means of differentiation and potential profit increases. This is cause for any given user to be required to carry multiple different proprietary cables, with variances in adapter types as well, when the device requires a voltage or minimum amperage specification to charge properly.

The issue is with the sheer number of separate adapters, cables, and solutions for charging that a user would be required to carry out of their home and with them to destinations during the day. Bringing along a battery pack, when no outlet would be available, adds another component to the number of required items a user would need for a normal day away from home.

Based on our marketing research, likely only two other possible solutions have come to light, and they fall far from a solution to our problem. These issues have yet to be fulfilled in the context of our specifications. Although one product with a wireless charging pocket has made it to market, it still boasts issues for the user, because the batteries must be removed to be recharged.

1.4 Solution

Integration of the battery pack and wireless charging coils into pockets of the most commonly used bags for carrying items will eliminate the requirement of consumers to maintain this number of extra items while away from home. It also eliminates the extra thought considerations needed by the consumer in order to keep their devices charged while away. Smartphones, smartwatches, and other devices capable of wirelessly

charging can be thoughtlessly stowed away in any pocket and will begin to recharge. The addition of wireless receivers to non-wireless charging capable devices could pose another level of positive impact to solution goals. This system will include a mobile application that will allow users to view battery percentage data for each registered device including the backpack itself, as well as the ability to prioritize device charging via easily accessible switches based on need and available charging capacity.

Extra considerations for ease of use involve a charging dock for the bags/rucksacks/purses themselves. GPS tracking capabilities give the user extended peace of mind if their backpack/bag was left behind or lost. Global WiFi hotspot capabilities enable the user to access the internet wherever they are with their backpack/bag. The complete elimination of proprietary cords, adapters, and battery banks provides the user with complete freedom while away from home, travelling or on the go.

1.5 User Profile

Figure 1: User Profile, illustrates the top five user groups that were targeted and used during development. This provided the team with insights into specific personas to create key considerations for the product. With these chosen audiences, the team will ensure accurate target execution based on the needs of our users.

USER PROFILE

PROJECT:

Wireless Charging SmartBag

POTENTIAL USERS:

- Anyone with Qi enabled smart devices
- Senior Citizens / Elderly who often forget to charge
- Adolescents where sports and school leave little time for charging
- College Students who are constantly mobile throughout the day
- Travelers / Businessmen always on the go
- Outdoorsmen who would potentially be away for days on end

SOFTWARE, INTERFACE, AND RELATED EXPERIENCE:

The primary target user base will include anyone who owns and operates Qi enabled smart devices. With wireless charging becoming more popular, people will naturally seek to be able to utilize these features within the contexts of their own lifestyles. Here are the top key user classes that will be attracted to the features and product that we offer:

College Students are often carrying more electronic devices than other user groups. Between laptops, phones, tablets, and smartwatches, they will be more likely to have a high number of Qi enabled smart devices on hand. Combined with their mobile and on-the-go lifestyle, it increases the likelihood of having to charge multiple times while on and around campus.

Travelers / Businessmen are a similar user class to college students and are likely carrying many Qi enabled smart devices. The necessity and attraction to our features is the same for this target, as they will be on the go and potentially without a place to charge or time to do so.

Outdoorsmen are a user class that would be away from charging via an outlet for potentially days on end. The high capacity and reliability of our product offers features that satisfying the needs of the target user group as well.

Adolescents have a great deal of electronics as well and with parents wanting to stay connected, would be a key user class. Through school and sports, they will benefit from maintaining charge on multiple devices and the mobility features our product provides.

Senior Citizens / Elderly are another key user class that would benefit from many features of our product such as push notifications and alerts on battery level, in the case that they forget to charge.

EXPERIENCE WITH SIMILAR APPLICATIONS:

The primary target user base will include anyone who owns and operates Qi enabled smart devices. Most users have experienced wireless charging of their Qi enabled devices and have experience with navigating and configuring mobile applications related to a physical product. Here are the top key user classes that will be attracted to the features and product that we offer:

College Students are often carrying more electronic devices than other user groups and will have experience with wirelessly charging Qi enabled smart devices. Many physical products on the market have a related mobile application requiring some configuration. These users will have experience with this process.

Travelers / Businessmen are a similar user class to college students and are likely carrying many Qi enabled smart devices. These users will also have experience with the process of configuring an app for a physical product to utilize its features.

Outdoorsmen may not need some of the more advanced features, but our product offers advantages to their lifestyle, that may be easily utilized. As stated before, owners of Qi enabled devices will have some experience with wireless charging. The configuration of a physical product through a mobile app is becoming commonplace today.

Adolescents are always in tune with the latest tech and will have experience with configuring and operating a product like ours. It is intuitive to learn how to configure app settings to manage the SmartBag.

Senior Citizens / Elderly are likely to have the least experience with some of our more advanced features; however, the simplicity and friendly user experience of both the wireless charging smart bag and mobile application invite any user to take advantage of its features.

TASK EXPERIENCE:

Individuals who own Qi enabled devices will have experience wirelessly charging them. Of the defined user groups, the majority of them, will have experience utilizing mobile application configuration of a physical product.

Individuals using the application will also have experience performing tasks such as:

Searching Bluetooth devices, pairing Bluetooth devices, GPS tracking on a map, Device prioritization management and Battery status management, as these features are available on every smartphone and utilized within thousands of other apps.

FREQUENCY OF USE:

The initial configuration of this product will only need to be performed once but may be continually updated and changed per the user's needs. For our product demonstration and in actual use, the product will be able to charge multiple devices, prioritize charging based on user setting, geo locate the SmartBag, and even initialize a global WiFi hotspot. At our product demonstration, we will be able to show different use cases to appeal to the different user groups outlined above. Demonstrating how to modify the priority of devices via switches, pairing devices, locating the bag if lost, etc.

KEY PROJECT DESIGN REQUIREMENTS THAT THE PROFILE SUGGESTS:

- Visually appealing, clean flow, and easily understood and navigable UI
- Ability to turn features on or off
- Quick setup time
- 4 Qi wireless charging coils/pockets for simultaneously charging devices
- Compliance with QI wireless standards
- Safety features, including water resistance, and heat dissipation
- GPS tracking and geolocation
- Global WiFi hotspot

Figure 1. User Profile

1.6 Use Case Diagram

The following diagram, Figure 2: Use Case Diagram, provides a visual representation of the interaction between the user, mobile app, SmartBag, and each involved technology.

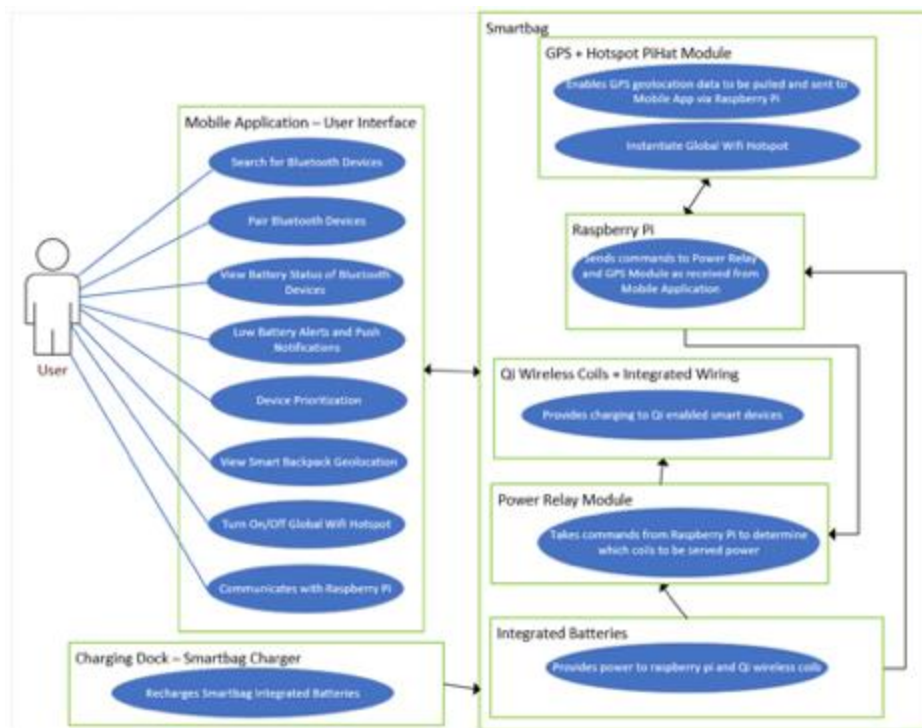


Figure 2. Use Case Diagram

2. PROJECT MANAGEMENT

2.1 Budget

Table 1: Project Budget, portrays the cost estimate for the development and implementation of the Wireless Charging SmartBag.

No.	Item	Unit Hours	Unit Price	Line Item Total
HARDWARE				
1	Estimated Labor	125	\$40.00/hr	\$5,000.00
2	Estimated Hardware	N/A	\$250.00	\$250.00
			TOTALS:	\$5,250.00
3	Actual Labor Cost	248	\$40.00/hr	\$9,920.00
4	Actual Hardware Cost	N/A	\$277.76	\$277.76
			TOTALS:	\$10,197.76
SOFTWARE DEVELOPMENT				
1	Estimated Labor	800	\$55.00/hr	\$44,000.00
			TOTALS:	\$44,000.00
2	Actual Labor	1,256	\$55.00/hr	\$69,080.00
			TOTALS:	\$69,080.00
	Estimated TOTALS:	925	...	\$49,250.00
	Actual TOTALS:	1,504	...	\$79,277.76

Table 1. Project Budget

2.2 Objectives/Deliverables

Our proposed solution seeks to encompass the wireless charging capabilities into as many pockets as possible, so that devices may be charged with little thought of placement. The use of our easily accessible switches for the prioritization of devices, will enable our users to maintain peace of mind that the more important devices stay charged (business phone), if the charge of their backpack has gone too low to supplement all. This is another feature that sets our solution above the rest.

We are developing a backpack/purse/carrying bag that eliminates a user's worry of a phone, watch, or other equivalent wirelessly charged device from completely dying while out and about. The No-Frills Wireless Charging Backpack will eliminate the need for users to remember to bring charging adapters, charging cables of specific proprietary connection types, or external battery packs along with them when leaving their home. By combining these cables, integrating Qi standard charging coils into pockets, and placing up to sixty-four amp hours of extra power into a commonly carried bag/pack, we will solve the problem outlined above. Other possible solutions in the market only offer up to ten-amp hours of charge. Our backpack will enable users to charge for upwards of a week with a fully charged backpack unit. No need to remove batteries or mess with cords or adapters of any kind, which is not currently an option on the market.

A mobile application enables users to view battery levels and decide on the prioritization of their devices to ensure that the devices that are in critical need of being charged can be given higher priority in situations where there is only enough backpack charge capacity left to charge one device sufficiently. With the inclusion of GPS

capabilities, last known location can be used to find the backpack if it is lost or left behind. Global WiFi is also a possibility with the addition of a sim card.

The No-Frills Wireless Charging Backpack will significantly increase the freedom of its users to leave their home, and no longer put any thought into keeping the electronics that power our lives from dying. A docking station that uses a contact-based charging system will be used when the user is back at home. We would potentially offer a wall mount that incorporates an AC adapter that will charge the backpack/bag, via contact nodes located in the handle/shank of the back or located near the actual battery pack units at the bottom of the bag. The No-Frills Wireless Charging Backpack will also include a Raspberry Pi for the purpose of providing data exchange and processing between the devices, the Qi charging coils and the mobile app.

The target features of the No-Frills Wireless Charging Backpack will include:

- Inclusion of wireless coils into each main pocket of the backpack/purse/bag.
- Users will be able to easily dock and charge the bag at home, so it is ready for the next day of work or adventures.
- Optionally, the inclusion of a wall mount with the charging dock functionality.
- User will be able to utilize any pocket of the backpack/purse/bag as a source of wireless charging.
- GPS capabilities
- A mobile application
 - View current charge percentages of each device
 - View remaining charge percentage of backpack/bag

Together, these features will provide users with the freedom they need to leave their home for extended periods of time, without having to remember to bring charging adapters, charging cables of specific proprietary connection types, or external battery packs along with them.

There were no initial goals or design objectives that had to be abandoned during the course of project development. Our design requirements allowed us to meet our goals by incorporating each technology into an all-inclusive system. The incorporation of existing technologies coupled with our mobile app enabled us to implement a unique product that met the goals we set out and completed our design requirements. Our procedures outlined below in our project schedule were key to meeting project deadlines and implementing this solution by the end of the semester. Strict adherence to procedures in our timeline was what enabled us to complete our vast number of design requirements and ultimately bring this solution to fruition.

2.3 Project Schedule

Table 2: Project Schedule and Figure 3: Gantt Chart encompasses the proposed timeline to complete specific deliverables and achieve target milestones.

Task Name	Duration (Days)	Start Date	End Date
1. 0 Project Management and Deliverables	232	8/26/19	4/14/20
1.1 Team Building	7	8/26/19	9/2/19
1.2 Ideas and Brainstorming	7	8/26/19	9/2/19
1.3 Fall Semester Assignment 0: Team Members & Project Name	7	8/26/19	9/2/19
1.3.1 Project Name	7	8/26/19	9/2/19
1.3.2 Project Logo and Branding	7	8/26/19	9/2/19
1.4 Fall Semester Assignment 1: Team Contract	14	9/9/19	9/23/19
1.4.1 Project Approval	14	9/9/19	9/23/19

1.4.2 Gantt Chart	14	9/9/19	9/23/19
1.4.3 Work Breakdown Structure	14	9/9/19	9/23/19
1.5 Fall Semester Assignment 2: Project Abstract for Tech Expo	21	9/23/19	10/14/19
1.6 Fall Semester Assignment 3: Team Contract Resubmission	7	10/7/19	10/14/19
1.7 Fall Semester Assignment 4: User Profile	14	10/7/19	10/21/19
1.8 Fall Semester Assignment 5: Use Case Diagram	14	10/7/19	10/21/19
1.9 Fall Semester Assignment 6: Draft Report	14	10/21/19	11/4/19
1.10 Fall Semester Assignment 7: Final Fall Semester Report	28	11/4/19	12/2/19
1.11 Fall Semester Oral Presentation	35	11/4/19	11/18/19
1.11.1 Presentation Practice	45	11/4/19	11/18/19
1.12 Spring Semester Assignment 1: Testing Plan/Report	28	1/13/20	2/10/20
1.13 Spring Semester Assignment 2: Abstract	7	2/10/20	2/17/20
1.14 Spring Semester Assignment 3: Draft Tech Expo Poster	14	2/17/20	3/2/20
1.15 Spring Semester Assignment 4: Final Poster	7	3/2/20	3/9/20
1.16 Spring Semester Oral Presentation	21	3/16/20	4/6/20
1.16.1 Presentation Practice	21	3/16/20	4/6/20
1.17 Spring Semester Assignment 5: Final Report	21	3/16/20	4/6/20
1.18 Spring Semester Assignment 6: Safe Assign Final Report	21	3/16/20	4/6/20
1.19 IT Expo	8	4/6/20	4/14/20
1.19.1 IT Expo Exhibit and Preparation	8	4/6/20	4/14/20
1.20 Spring Semester Assignment 7: Final Library Copy	9	4/20/20	4/29/20
2.0 Research	42	9/9/19	10/21/19
2.1 Equipment Requirements	22	9/9/19	10/1/19
2.1.1 Determine type of Qi wireless chargers	5	9/9/19	9/14/19
2.1.2 Determine type of Battery Pack	4	9/14/19	9/18/19
2.1.3 Determine type of Raspberry Pi & LCD screen	2	9/18/19	9/20/19
2.1.4 Determine type of Backpack	4	9/20/19	9/24/19
2.1.5 Determine waterproofing options	4	9/24/19	9/28/19
2.1.6 Determine type of wires/cables needed	3	9/28/19	10/1/19
2.2 Software Requirements	22	9/9/19	10/1/19
2.2.1 Determine App Language	5	9/9/19	9/14/19
2.2.2 Determine Bluetooth security	4	9/14/19	9/18/19
2.2.3 Determine AutoMeasure criteria	2	9/18/19	9/20/19
2.2.4 Determine Bluetooth connectivity requirements	4	9/20/19	9/24/19
2.2.5 Determine Bluetooth hub coil shutoff method	4	9/24/19	9/28/19
2.2.6 Determine app notification settings	3	9/28/19	10/1/19
2.3 Safety Requirements	7	10/1/19	10/8/19
2.3.1 Research QI Standards Compliance	7	10/1/19	10/8/19
2.3.2 Determine voltage/Amp for wireless charging	7	10/1/19	10/8/19

2.3.3 Research on maximum weight threshold	7	10/1/19	10/8/19
2.3.4 Research on Ergonomic support	7	10/1/19	10/8/19
2.4 Miscellaneous Research	7	10/14/19	10/21/19
2.4.1 Average number of dead batteries	7	10/14/19	10/21/19
2.4.2 Average weight of Backpack contents	7	10/14/19	10/21/19
2.4.3 Conduct Interviews with Mock Stakeholders	7	10/14/19	10/21/19
2.4.4 Budget Analysis	7	10/14/19	10/21/19
3.0 Design	21	10/21/19	11/11/19
3.1 Create Prototype	7	10/21/19	10/28/19
3.1.1 Create integrated wiring design	7	10/21/19	10/28/19
3.1.2 Create power supply design	7	10/21/19	10/28/19
3.1.3 Create device placement design	7	10/28/19	11/4/19
3.1.4 Create optional wall mount for power supply design	7	10/28/19	11/4/19
3.1.5 Create wireframe Diagrams	7	10/28/19	11/4/19
3.2 Create App Legal Documentation	7	11/4/19	11/11/19
3.2.1 Draft Legal Disclaimers and Privacy Policy	7	11/4/19	11/11/19
4.0 Environment Set-Up	12	9/24/19	10/6/19
4.1 Import Libraries for Development	12	9/24/19	10/6/19
4.2 Setup GitHub	12	9/24/19	10/6/19
4.3 Install Android Studio	12	9/24/19	10/6/19
4.3.1 Set up Android Studio	12	9/24/19	10/6/19
4.3.2 Configure Android version and API	12	9/24/19	10/6/19
4.4 Raspberry Pi Setup	12	9/24/19	10/6/19
4.4.1 Raspbian OS Configuration	12	9/24/19	10/6/19
4.4.2 Bluetooth libraries import	12	9/24/19	10/6/19
4.4.3 Connected devices to Bluetooth relay	12	9/24/19	10/6/19
5.0 Development (Back End and Front End)	123	10/1/19	2/1/20
5.1 Create Layout	14	10/1/19	10/15/19
5.2 Choose Color Scheme	14	10/1/19	10/15/19
5.3 Create Buttons	14	10/8/19	10/22/19
5.3.1 Turn on Bluetooth	14	10/8/19	10/22/19
5.3.2 Turn off Bluetooth	14	10/8/19	10/22/19
5.3.3 List Devices	14	10/8/19	10/22/19
5.4 Create Hamburger Menu	21	10/15/19	11/5/19
5.4.1 List Paired Devices	21	10/15/19	11/5/19
5.4.2 Notification Options	21	10/15/19	11/5/19
5.4.3 AutoMeasure Options	21	10/15/19	11/5/19
5.5 Raspberry Pi Development	21	10/15/19	11/5/19
5.5.1 Raspberry Pi Bluetooth hub communications	21	10/15/19	11/5/19

5.5.2 Bluetooth Hub Data Mapping	21	10/15/19	11/5/19
5.5.3 Raspberry Pi Hub to Coil Shutoff	21	10/15/19	11/5/19
5.5.4 Coil Off/On via Prioritization	21	10/15/19	11/5/19
5.5.5 Validate Power Requirements (Hub/Pi/Coils)	21	10/15/19	11/5/19
5.6 Develop App Features	109	10/15/19	2/1/20
5.6.1 Develop Bluetooth functionality	26	10/15/19	11/10/19
5.6.2 Develop Pair devices functionality	34	10/15/19	11/18/19
5.6.3 Develop Notification functionality	16	11/29/19	12/15/19
5.6.4 Develop AutoMeasure interval functionality	95	10/29/19	2/1/20
5.6.5 Develop Device Prioritization functionality	109	10/15/19	2/1/20
6.0 Testing	73	2/1/20	4/14/20
6.1 Functionality Test	30	2/1/20	3/2/20
6.1.1 Turn on/off Bluetooth	30	2/1/20	3/2/20
6.1.2 Pairing devices functionality	30	2/1/20	3/2/20
6.1.3 List devices	30	2/1/20	3/2/20
6.1.4 Turn off wireless charging functionality	30	2/1/20	3/2/20
6.1.5 Battery level display of devices	30	2/1/20	3/2/20
6.2 User Pilot Test	30	3/2/20	4/1/20
6.2.1 Select user groups (Hikers, bikers, camping experts, city life enthusiasts, business travelers, and students)	30	3/2/20	4/1/20
6.2.2 Conduct usability test for each group	30	3/2/20	4/1/20
6.2.3 Perform multi-device use case	30	3/2/20	4/1/20

Table 2. Project Schedule

No-Frills Wireless Charging Backpack Gantt Chart:

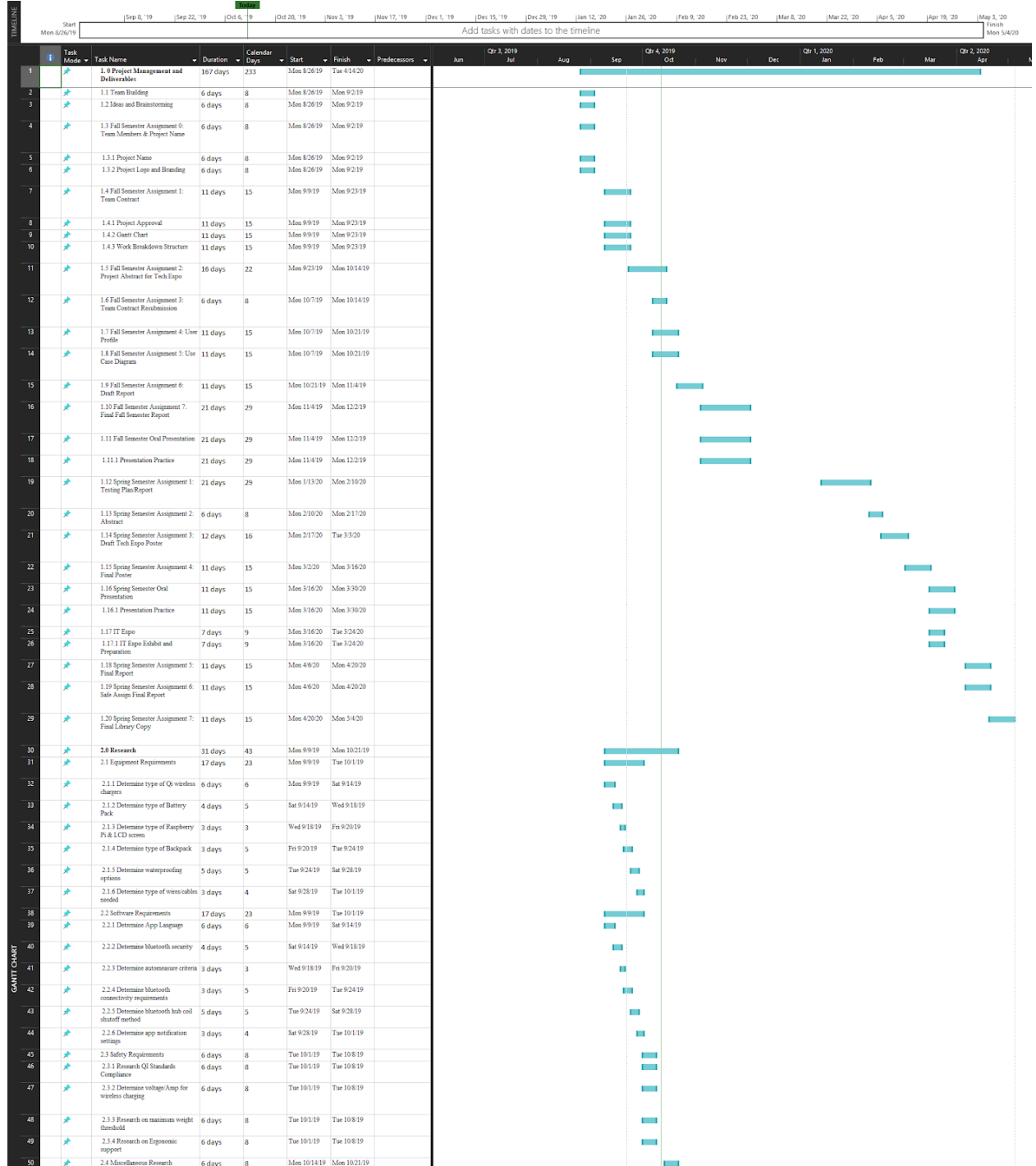




Figure 3. Gantt Chart

3. TECHNICAL ELEMENTS/DISCUSSION

3.1 Network

The networking elements include device to device Bluetooth communication. Retrieval of battery level data, and prioritization of devices will be handled using Bluetooth technology and on/off switching of the coils per the users needs. The main phone of the user, which contains the SmartBag mobile application provides data for each device in use. Bluetooth will also be used to send commands to a Raspberry Pi Zero Wireless module that can relay information to the application. The Raspberry Pi Zero Wireless will also communicate with 3G/4G cellular towers to retrieve GPS location data, and to broadcast a global WiFi hotspot. This is done with serial communication using the GPIO header pins on the Raspberry Pi Zero Wireless with an attached SIM868 module. The module which is directly attached to the Raspberry Pi Zero Wireless, has antennas for communicating GPS data and GSM with cellular networks via AT commands.

3.2 Application

The mobile Application will be built using android studio, an integrated development environment (IDE) used for android app development. The mobile application will be created using java programming language. The app will be using Bluetooth features, as a result, Bluetooth permissions are required. These permissions include Bluetooth, Bluetooth_Admin and Access_Fine_Location. By enabling these permissions users will effectively be able to discover nearby devices based on their

location. The application will be supported by devices with API level between 16-28, this means 99.4% of android devices will be able to use the app.

3.3 Physical Hardware Design

The hardware elements of the project include the use of multiple smart devices. The integrated design of the SmartBag itself contains two high capacity battery cells for a total of sixty-four amp hours of power and two backup batteries powering the Raspberry Pi and its peripherals. These batteries are wired to the Raspberry Pi Zero Wireless, and four Qi wireless charging coils enabling the simultaneous charging of up to four devices. Between the connections from the battery to the Qi wireless charging coils sits power switches, that can be used for powering on and off specific coils as the user sees fit. The Raspberry Pi Zero Wireless module itself has a communication module soldered on top via GPIO pins. This module contains the microchips capable of communication with an inserted, working SIM card, to send and receive data via the attached antennas. The base of the SmartBag has a port for recharging the high capacity battery cells, and those of the Raspberry Pi.

4. VISUALS

4.1 Network Architecture

Figure 4: Network Architecture, illustrates how the various network components will communicate with each other.

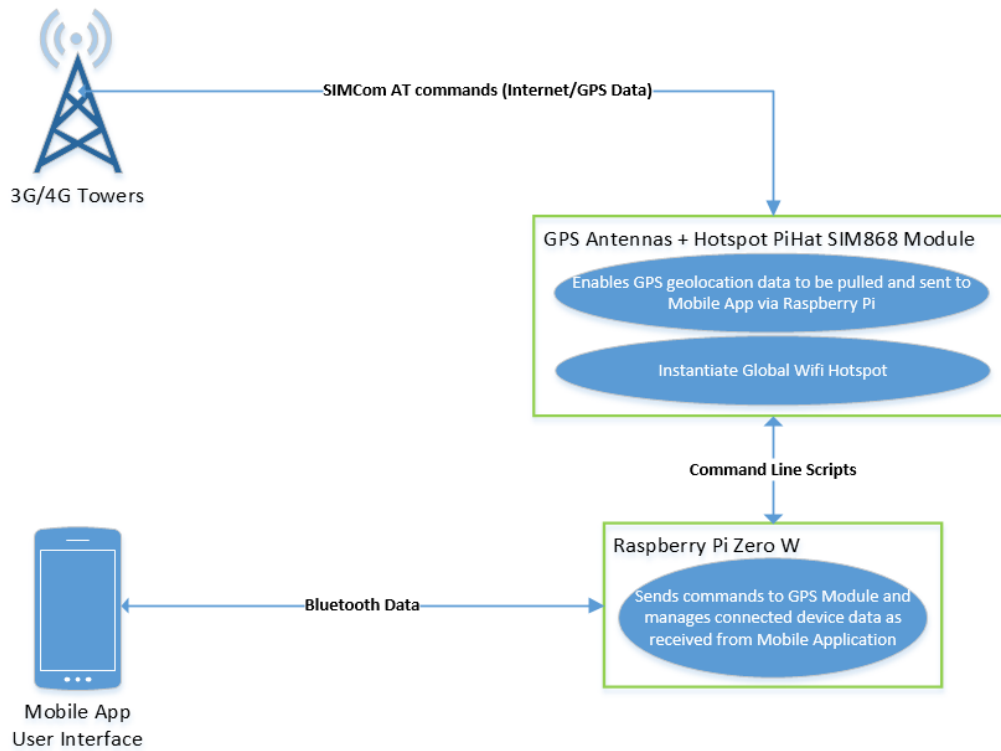


Figure 4. Network Architecture

4.2 Hardware Implementation

Figure 5: Hardware Implementation, illustrates how the various component inside the SmartBag will communicate with each other.

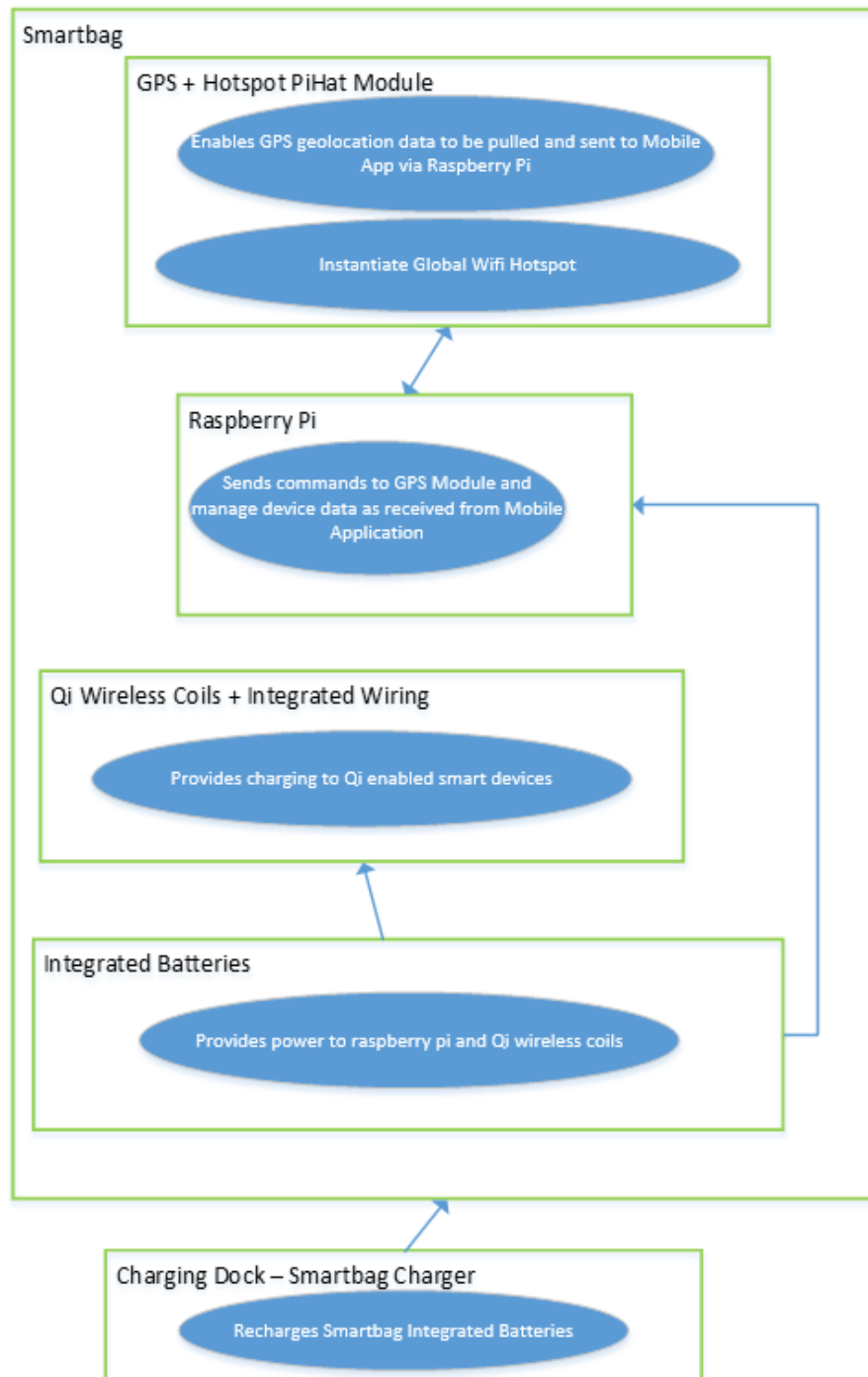


Figure 5. Hardware Implementation

4.3 User Interface

Figure 6: User Interface, provides a visual representation of how user can use different features of the mobile application.

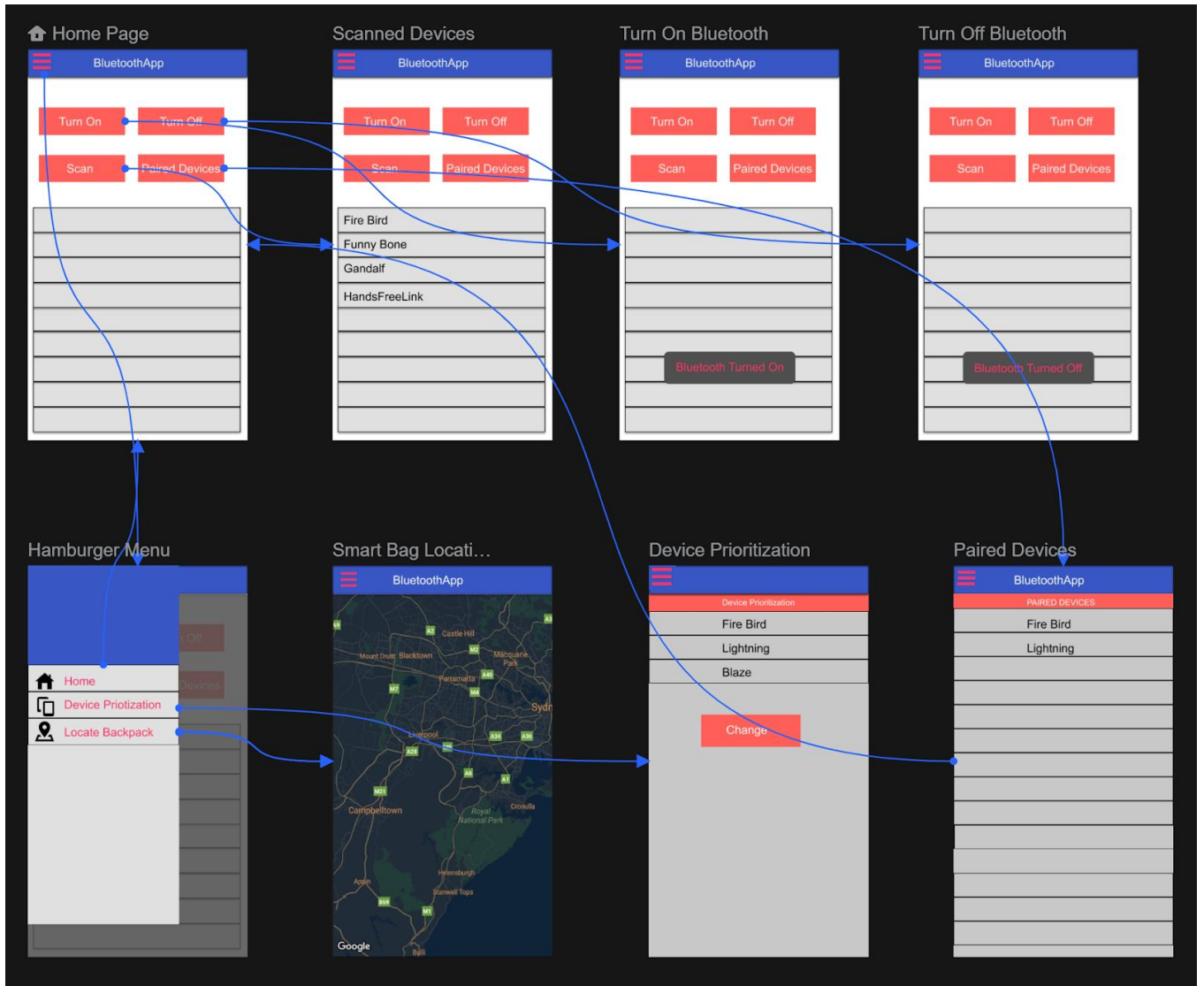


Figure 6. User Interface

5. TEST PLAN

5.1 Overview/Methodology

Our test plan will be designed to confirm mobile application functionality and interactivity with the SmartBag, as well as collect user experience data. Testing is very essential in development of any sort of application, whether it is a desktop application, mobile application or web application. The purpose of testing is to ensure that the requirements are met and the final product is acceptable to the end user. The following is a reference regarding the processes and methods used to test the functionality of the Wireless Charging SmartBag and the SmartBag Manager Mobile App. The app is designed to search for Bluetooth devices and pair with them in order to create a personal area network. With each desired device added, a user can use their phone to view the battery level of each device, including the SmartBag. The app offers the ability to geolocate the wireless charging SmartBag, set notifications, and users can prioritize the charging of devices via switches based on device info in the app. The SmartBag Manager Mobile App will only be available for Android Devices. The Wireless Charging SmartBag allows simultaneous charging of up to four devices. The following groups will be able to use this section as a guide:

- Developers
- Beta Users - College Students, Travelers/Businessmen
- UI Designers
- Team Members

The following approaches to testing that will be performed to ensure full functionality include:

- 1) Usability Testing – Users will provide feedback on ease of use with the mobile application and their interactivity with the SmartBag
- 2) Compatibility Testing – Users will use various devices with different android API levels to ensure testing meets end user requirements with this included API variability and different sizes of devices.
- 3) Functionality Testing – Users will test all features and buttons of the app to ensure they meet the proposed requirements and that the SmartBag works as intended.

5.2 Scope of Testing

The scope of this testing encompasses adequate charging of devices via wireless coils in each main pocket of the SmartBag and providing the ability to manage the entire personal area network of included devices. This scope involves: Ensuring that users will be able to easily dock and charge the bag at home, so it is ready for the next day of work or adventures. The ability to utilize any pocket of the SmartBag apart from the main storage compartment as a source of wireless charging. A SmartBag Manager Mobile App using GPS capabilities to locate the SmartBag, view current charge percentages of each device, view remaining charge of the backpack/bag, prioritize device charging according to device data in the app, and set up specific notification settings.

5.3 Objectives

The objective is to confirm feature functionality and user requirements to the proposed project solution. Scores for user experience, ease of use, and functionality without error will provide a basis for whether or not feature requirements were met. The purpose of testing is to ensure that every proposed function works as expected, the entire product is easy to use, and to remediate any issues that may be discovered. In order to have a complete analysis of the Wireless Charging SmartBag and the SmartBag Manager Mobile App, we will be testing each feature separately, conducting compatibility testing for various Android devices varying in API level, and recording results of usability testing. This will ensure that the requirements of the SmartBag and app are met so that the final product is in an acceptable form to present to the end user.

5.4 Test Requirements and Procedures

As outlined above, we took three approaches to our testing which included, usability testing, compatibility testing, and functionality testing. Any necessary fixes or changes that were discovered during this testing were fulfilled.

Each approach serves a specific purpose in completing a robust analysis of our unique product. Usability testing holds substantial weight because it determines whether or not we have a viable product that people are willing to use. Compatibility testing is required for ensuring that our application works with many different mobile operating systems on the Android platform classified by API level. It also ensures compatibility of

our SmartBag pockets with various sizes of devices. Functionality testing is paramount to the proper function of the whole system.

Each team member was required to work on all fixes or changes to improve the product in any of the three approaches.

The following requirements must be met to perform proper testing:

- Outline the pass/fail conditionals for each applicable function
- Generate record tables for each test attempt and subsequent results
- Ensure that various devices are included in testing and that these devices are of different Android API levels
- At least one user from each of our user profiles is included in the testing

Tests that will be conducted:

1) Usability Testing

- Potential users will test the app and rate it on a scale of 1 to 5, 1 being hard to navigate through the app and 5 being easy to navigate.
- The SmartBag will be tested using the same rating scale as above to rate users' interactions with the pockets, adding different devices to charge, etc.

2) Compatibility Testing

- The app will be installed on devices with different android API levels and all of the usability tests will be performed to ensure testing meets end user requirements with this included API variability.

3) Functionality Testing

- All the features and buttons of the app will be tested to ensure they meet the requirements and serve the end user needs.

- Each individual function of the application will be tested separately using a pass/fail conditional in conjunction with the SmartBag.
- The SmartBag will be tested using a pass/fail conditional for functionality where applicable.

5.5 Test Results

Table 3: Shows the Usability test result among different user groups.

USABILITY TEST	USER 1	USER 2	USER 3	USER 4	USER 5
RATING 1 -5					
How easy was it to connect to Bluetooth devices?	5	5	4	5	5
How easy was it to pair Bluetooth devices?	4	5	5	5	5
How easy was it to navigate through the App?	5	4	5	5	5
How easy was it geolocate wireless Back Pack?	5	4	4	5	5
How did your devices fit into the pockets?	5	5	5	5	5
Was it simple to place your devices in the pocket?	5	5	5	5	5
How comfortable is the Smart bag?	5	4	4	5	5

Table 4: Compatibility test between different android API levels

COMPATIBILITY TEST	INSTALLATION	SMARTBAG POCKET
Motorola Droid Turbo	✓	✓
Samsung Galaxy S9 Test	✓	✓
Samsung Galaxy S7 Edge Test	✓	✓
Samsung Galaxy S6 Test	✓	✓
Samsung Galaxy S4 Test	✓	✓
Samsung SM-G550T Test	✓	✓

Table 5: Functionality Test of the Application between different Android Devices

FUNCTIONALITY TEST	FUNCTION	DESCRIPTION	PASS/FAIL
Motorola Droid Turbo	App Start-Up	Fast, no issue	✓
	Menu Navigation	Smooth transitions	✓
	Scrolling	Working fine	✓
	GPS Location	Takes a minute but works	✓
	Notifications	Great info and detail	✓
	Pair BT	Works	✓
	View paired BT	Working	✓
	Enable Discovery	Working	✓
Samsung Galaxy S9 Test	App Start-Up	Works	✓
	Menu Navigation	Simple and easy	✓
	Scrolling	Working well	✓
	GPS Location	Found GPS coords and put it in a google maps in a text!	✓
	Notifications	Working well	✓
	Pair BT	Working	✓
	View paired BT	Working	✓
	Enable Discovery	Working	✓
Samsung Galaxy S7 Edge Test	App Start-Up	No issues	✓
	Menu Navigation	Smooth and easy to use	✓
	Scrolling	Smooth	✓
	GPS Location	Like the google maps text	✓
	Notifications	Good info, more than expected	✓
	Pair BT	Works just fine	✓
	View paired BT	No issues seeing devices	✓
	Enable Discovery	Works great	✓
Samsung Galaxy S6 Test	App Start-Up	No issues starting	✓
	Menu Navigation	Great layout and flow	✓
	Scrolling	Works fine	✓
	GPS Location	Takes some time but excited to know I can find this bag anywhere!	✓
	Notifications	Great variety of information about each device!	✓

	Pair BT	Works well	✓
	View paired BT	Can see all paired devices	✓
	Enable Discovery	Works as intended	✓
Samsung Galaxy S4 Test	App Start-Up	No issues running	✓
	Menu Navigation	Easy to navigate	✓
	Scrolling	Smooth and functional	✓
	GPS Location	Cool feature, works great.	✓
	Notifications	Working fine, and nice to have info on each device.	✓
	Pair BT	Pairs quickly	✓
	View paired BT	Can see paired devices	✓
	Enable Discovery	Working well.	✓
Samsung SM-G550T Test	App Start-Up	No startup issues	✓
	Menu Navigation	Easy enough to use	✓
	Scrolling	No issues with scrolling	✓
	GPS Location	Works but took some time	✓
	Notifications	Favorite feature here, lots of info for each device and graphs available showing all devices usage.	✓
	Pair BT	Working well	✓
	View paired BT	Working well	✓
	Enable Discovery	Working well	✓

5.6 What we learned during Testing

We learned that our thoroughness through development paid off. We made sure to include as many devices as possible when completing design requirements. This was most important during mobile application development. Most of the issues that arose during the development of our mobile application were a compatibility problem. Developing with multiple Android operating system levels and various phones throughout the entire start to finish was paramount to the successes we have seen during our final product testing. At each step of the way and each new feature we implemented, we would install and test our mobile application on as many devices as we had available

to us. This resulted in a robust and fully compatible code that works on every device we have tested thus far.

We also learned that incorporating multiple technologies into one product can be cumbersome and that focusing on the standards behind each technology makes for easier development. Bluetooth for instance, is regulated by an organization and has strict protocols, references, and up to date information that adheres to the latest security standards. By utilizing each technology stack at its core, development progressed at a much smoother pace. The toughest lesson learned is with the realization of development constraints caused by budget. Maintaining a low budget and overall cost for our project allowed us to be more creative but proved that a healthy budget could have increased our potential in development on both hardware and software elements.

The main lesson learned is that you can only have so much foresight when you begin a project and must constantly reevaluate project variables on each day moving forward. This is what has led us to a successful implementation of our proposed solution. Maintaining a high level of adaptability has proven to ease issues that could have arose further down the road. Our compatibility testing went off without a hitch due to our tenacity to improve each aspect of our design and wide range of considerations early on, as well as throughout the entire development process.

5.7 Problems Encountered and Analysis of Problems Solved

We had issues with the connection on the PiHat module which was a big setback and very disappointing. It was unclear for a while that the issue was actually with hardware, and so we spent a lot of time wiping microSD cards and start back from

scratch. As stated above, some good did come of this (completely thorough knowledge of the onboard chips functions, how to manipulate them, kernel device tree tweaking, and very thorough documentation). The new boards and hammer headers that arrived got us ready to get our reliable module up and running so we could complete our work with the mobile app and testing involved with the app to the device.

The other issue had been with the prioritization function. We had to diverge to a solution that enabled users to manually switch on and off the charging coils. More importantly, determining which paired device needed the most charge via the app was essential. This adaptation was critical to the success of our product. An integration using Docker on our Raspberry Pi brought together all of the data we needed to house specific device info for each device in one place, which is now available in our mobile application.

Another issue was with our app being able to show the battery level of the SmartBag batteries. We do not believe there are any communication protocols to read this information. We were not sure if there is a way to determine the battery level from the two battery banks through power cables. We found a solution through a company in the United Kingdom that created a PCB called MoPi. This allowed us to configure a specific array of rechargeable batteries in series. The software developed for the board takes into account the battery chemistry, number of cells, and high, good, low, and critical points of voltage to determine charge level.

5.8 Future recommendations/Recommendations for Improvement

Our future recommendations involve contact with experienced Android and iOS developers. This was one of the most challenging pieces of our product to implement due to the vast amount of involved technologies and communication protocols. Restricting ourselves to Android only was a decision made in the beginning of our project. We were able to implement the necessary features but are certain that our code is not as efficient as it could be. If we had to do it all over again, we would be able to complete a working product in a short amount of time. With a team of mobile developers, we could create a cleaner mobile application in terms of code efficiency and UX. The amount of knowledge and lessons learned through this extremely iterative design process have allowed us to make more considerations each day.

Our product could keep growing and becoming even more feature rich if we had more time. With every new consideration and discovery, we have had dozens more ideas for improvements. Each piece of this product can be improved with time. If we had access to print our own PCBs and 3D print enclosures for our hardware, we would be able to eliminate certain integrations of other company's products. It has become apparent that with more time this product will continue to improve.

Some suggestions we have received from others were to include the ability to remove the batteries. This will certainly be an option in a proposed version 2 of our product. Concerns for battery removal on airlines was the reasoning for this suggestion. Other suggestions included offering various bag types. This has always been in our scope, although we had to focus our time and energy into a solid working prototype. Our main focus was with our backpack model. We want to offer purses, duffle bags,

briefcases, and other various carrying cases in future designs, with various offerings on battery capacities.

Our plans for this project are to use our suggestions and knowledge to reach out and get a team to back us. As we outlined above, experienced mobile developers and manufactures could enable us to eliminate 3rd party integrations, giving us full ownership over each piece of the product. With the right resources, we hope to potentially push this product to market someday.

6. CONCLUSION

6.1 Fall Semester 2019

Since our decision to move forward with this project, our team has thoroughly gone through as many considerations as we could with the user in mind. We have faced challenges, but were willing to push through, learn what we needed to, and make the progress needed to stay on track. Since this started as a more electrical engineering-based project, we knew that we should go ahead and map it out, so that we can focus our time and efforts on preparing the IoT communications side of things. We have figured out how each component should interact with each other to enable full functionality of our outlined scope.

The only worry we had in the beginning was with mobile app development. Neither of us were very knowledgeable on this subject. Through researching each of the methods of data communication involved, the data we were handling, its security considerations, and the other systems we needed to interact with, we were able to quickly grasp the concepts to gain a clearer idea of our timeline. We have made great progress on our mobile app and hardware implementation.

The further we pushed through development, the more considerations that were realized for our project. We knew that there would be unknowns and we were confident that we could take them as they come. It has been a great learning experience getting these sorted out early. One such consideration was how we were going to either put the Raspberry Pi in low power mode or turn it on and off. Research showed that it can run on a substantially lower source of power than in the actual specifications. The integrated batteries are able to run the Pi at its recommended voltage and amperage, but we may

underclock it, depending on our testing results, and what comes of our solution to low power/ turning it off and on. Another consideration was whether we should use one of those clear backpacks that they force some High School students to use, or if we stick to one like a swiss gear. We ultimately decided that a clear backpack would be perfect for our prototype and subsequent demonstrations.

Software development has been moving along nicely, as stated previously this was one of our biggest concerns, but we have made good headway. We have at this time successfully developed an app that can enable Bluetooth permissions and discover devices. We are collaborating using GitHub to stay organized and have some version control. We are in the stages of adding the next step and pages to the mobile app which will allow users to pair/bond devices. Bonding is the exchange of long-term keys after pairing has already occurred, and it stores the keys to be used later. Pairing is not the creation of permanent security between devices, which we did not know about until we started our work on this. Pairing is the mechanism that actually allows bonding to occur, which is pretty neat. As stated previously, we HAVE put our noses to the grindstone on this and are feeling much more comfortable now that we are well underway.

Testing the app on multiple phones with various Android OS levels has shown us that we could improve compatibility tremendously, and we have already done so. It is also recommended by the Android Dev community to not allow too low of a minimum as it can create problems with the app itself. We realized we should work on this part now and get it sorted out, before we start writing code for our other more complex features. This is going to require some tweaking but should be fairly easily sorted out. Some

phones were not acting the same with the app installed, so this will enable us to create a much more compatible app through testing.

We have begun working with pre-paid SIM cards as well which are relatively cheap on Amazon. Similar in price to a monthly contract which we found through T-Mobile for around \$3/mo. These cards are cheaper because they can utilize older technologies like GSM with 2G and 3G and still work for GPS purposes. Verizon plans to turn off 2G and 3G at the end of 2019 so we need to look more into that, so we are not building out something that will become obsolete. 4G is possible, but not optimal for our costs. T-Mobile will turn off 2G and 3G at the end of 2020. AT&T has already turned off 2G. AT&T and Sprint will keep 3G running until 2022. Sprint, Verizon, and US Cellular use CDMA. AT&T and T-Mobile use GSM. We found some excellent tutorials online from users that have already paved the way with GPS GNSS GSM Arduino and Raspberry Pi add-ons. This will help us complete this feature quickly and learn from other's mistakes.

6.1 Spring Semester 2020

Throughout development we have improved our documentation skills. Constantly creating backups and cloning our MicroSD cards as well as maintaining our GitHub repository for our mobile application code was a must. We have added to our documentation the extra linux modules used, in addition to the steps required to implement and install the proper Raspbian operating system, updates, and modules necessary for our GPS and SMS connection. With each update to documentation, there was new knowledge gained or a lesson learned. Installing the Raspbian Lite version, which is CLI only and performing the apt-get update, upgrade, and rpi-update is standard.

One non-standard Linux lesson was the raspberry pi kernel hacks needed to edit the device tree code directly. These are called DTOverlays and enabled us to integrate our specific hardware into this product and allow correct communication to the other components.

Additional modules such as GAMMU and GAMMU-SMSd were necessary for location functionality. These two modules allow our raspberry pi stack to continually listen for incoming sms messages, which in turn can trigger virtually any system function based on the message and how we design it to interpret that message. This is how we were able to enable the GPS location feature. Another huge improvement was resolving multiple usage of a single serial connection. The Raspberry Pi Zero Wireless has only one serial connection port, we have written shell scripts and included them as system services. The gammu-smsd daemon is required to be stopped, as this is what listens for sms messages using the serial port. Upon receiving a text, another script is run that starts a separate system service. Any subsequent processes that had started under the daemon are attached as a PID of the daemon. We were able to create the services that run these shell scripts which in turn will detaches the scripts and processes that need to be run outside gammu-smsd. This is all necessary to free the serial port from the previous daemons and processes for use with multiple applications. This is the kind of information you cannot read in a book or find directly online. Each piece of development was like another piece of the puzzle.

The other addition since the fall was the simbamond API for use with the MoPi 2 board. It was fairly simple to install and configure and was a major breakthrough for our project at a point when we were stuck. This PCB and Linux module are the pieces that

indicate the remaining charge for the two batteries that power the Raspberry Pi stack. Aside from these additions, we completed every initial feature that we originally set out to provide. The backpack is a fully working prototype that has four charging coils and can be easily recharged and managed via our mobile application. This semester, we completed each piece of this product that we had started previously.

Overall, we have developed the ability to adapt continually, develop like we are completing a puzzle, and enhanced our problem-solving skills in technical areas in which we had no previous experience. Mobile application development was an area we had limited knowledge and came out the other end with a successful application. We have truly learned what an iterative design process means. We have learned from start to finish just about everything there is to know about Unix systems during the course of developing this product.

APPENDIX A. ADDITIONAL INFORMATION

The team contacted the following professionals for clarification and additional insight

1. Gary Skipper, Electrical Engineer - Michigan State University
2. Ryan Moore, Instructor - University of Cincinnati



WIRELESS CHARGING SMARTBAG

College of Education, Criminal Justice, & Human Services
School of Information Technology
University of Cincinnati

TEAM 42: Cole Skipper & Mihir Patel

Technical Advisor: Ryan Moore



THE PROBLEM

- According to Cisco VNI, today 8 networked devices per person is expected to reach 14 by 2022.
- The issue is with a tangled mess of separate adapters, cables, and solutions for charging.
- Currently no known solution exists in the market that features full integration of these items that also caters to a multi-platform userbase.

OUR SOLUTION

A Wireless Charging SmartBag that will eliminate the need for users to remember to bring charging adapters, charging cables of specific proprietary connection types, or external battery packs along with them when leaving their home.



Wireless Charging SmartBag is a Smart Device charging system utilizing a mobile application to view battery levels, prioritize device charging, geolocate, stay connected via Global Wifi HotSpot, and easily dock to recharge.

TECHNOLOGY + DESIGN

SMARTBAG	MOBILE APPLICATION
  	

KEY BENEFITS

 MULTI-PLATFORM QI TECHNOLOGY	 MOBILE APP DEVICE MANAGEMENT	 GPS GEO-LOCATION	 IOT DEVICE INTERCOMMUNICATION	 3G/4G GLOBAL WIFI HOTSPOT
---	---	---	---	--

APPENDIX B. REFERENCES

Cisco VNI. "Cisco Visual Networking Index: Forecast and Trends"
February 27, 2019. Accessed October 09, 2019.
[https://www.cisco.com/c/en/us/solutions/collateral/service-provider/
visual-networking-index-vni/white-paper-c11-741490.html#_Toc532256809](https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html#_Toc532256809)