SecurePack: Utilizing Internet of Things for Physical Security

Collin Kemner, Travis Ellerbrock, and Dr. Saeed Al-Haj
Electrical & Computer Engineering and Computer Science Department Ohio Northern University, Ada, OH 45810

Abstract

This paper will propose the creation of a new product named SecurePack, an IoT backpack with a focus on personal security. This backpack will include features such as an IoT kit, a built-in camera, audio speakers, GPS tracking, Bluetooth/Wifi capabilities, LED lights for bikers, communication with other backpacks, the normal storage features of most backpacks and an iOS app. SecurePack will strive to be as resilient to physical and cyber attacks as possible while still remaining cost efficient.

The SecurePack's intention is to protect the user from potential dangers most common in urban environments, as well as functioning as a normal backpack that can carry books, notepads, laptops, etc. The specific dangers the SecurePack will have in mind are the vehicular dangers in a city as well as dangers from potential attackers such as muggers. This backpack will have LEDs to warn nearby traffic of the user's presence by making the user more visible. The SecurePack will protect the user from a potential attacker by allowing the user to quickly shine a high lumen flashlight in the attacker's eyes as to temporarily blind the attacker and audio speakers to alert anyone within earshot. It will also come with a built-in camera to record any dangerous situations happening in the vicinity, and Bluetooth/WiFi capabilities so that evidence may be stored and sent to trusted devices or to the authorities. Another feature the Secure Pack will include is a GPS tracking device so that the user may send their location to a trusted person in the event that they are in danger. For the user's everyday convenience, the SecurePack will also have functionality that allows them to charge their devices with the backpack's battery pack.

The purpose of these features is to protect the user from dangerous situations and if these measures fail, then the SecurePack will allow for the easy transfer of evidence and potentially life saving information like video evidence and GPS coordinates. One of the major goals of this project is to allow for the SecurePack to communicate with users' phones and then allow for the phone to transfer the data over well-established cellular communication networks while the emergency mode of the SecurePack will be activated by a conveniently placed button. The user will also be able to activate the various features of the backpack at their convenience using their smartphone.

Introduction

The world is an unpredictable place, especially in areas where crime is rampant. According to the Chicago Police Department, there has been a 5% uptick in crime since 2015, with sexual assault at an outrageous 92% increase and a 15% increase in shooting incidents[1]. Despite technological advancements allowing for a quicker detection of crime, there are still incidents that occur in which police are not alerted or where they arrive to the scene too late. If there could be a way for potential crime victims to report suspicious activity or crimes in a quick and efficient manner, it's possible that more can be done to get help and attention to the scene in question.

Introduction to IoT

Internet of Things (IoT) technology are any devices that have the ability to connect to a network, and in turn, to the internet. For example, smartphones, wireless printers, smart homes, and smartwatches are all IoT devices. Smartphones can visit webpages and components (such as door locks) in smart homes can be controlled via smartphone apps. Wireless printers print documents sent over a network, and smartwatches can not only take phone calls, but can also connect to apps that require an internet connection. Many of the uses for IoT devices are in a recreational or business oriented environment, while smart homes are focused on the security of a house. Despite this, there seems to be a fairly limited market on using IoT for the security of an individual.

Problem Statement

This capstone project will introduce a new device intended to bring more security and peace of mind: The SecurePack. The SecurePack is a backpack that implements and utilizes IoT technology in order to allow crimes and dangerous situations be reported to loved ones and the authorities in both a stealthy way and in a way that can bring attention to the local area through the use of noise devices and lights. Through the use of such lights, nightlife may become more safe due to a better sense of awareness by others.

Background

Competition

Given that the idea of a personal defense backpack is a new entry into the market, the competition is minimal. Each product that would be a competitor to our solution only covers one aspect of what our product plans to do. For example, for personal defense against crime, a Gallup poll[2] showed that 48% of Americans simply avoids certain neighborhoods, 14% carry mace, 12% carry a knife, and 12% carry a firearm. While these are solutions for personal safety, they do come with drawbacks; namely that the latter two are violent and may place the user in harm's way. These methods of personal protection also depend on the user's skill with the weapon. In the case of mace, the same study found that women are far more likely to carry mace than men; 21% vs 7%, respectively. This result may be troublesome as women are more likely to carry mace in a purse or handbag for convenience. This location may cost precious seconds in the event of a physical attack. Our solution to this problem is to have a quick activation time requiring only 2 button presses on the shoulder strap of the backpack. Our solution is also non-violent, which is more palatable to most of the population.

In terms of other IoT backpacks on the market, the competition consists of backpacks that charge the user's devices, such as the TYLT Traveler Power Bag[5] and backpacks that can activate LED's to give turn signals from bikers to vehicles that are behind them, such as the Cycling Backpack[3]. Both of these features are planned to be incorporated into our solution. Researching the competition informed us what features could be advantageous to put in our product as well as informing us what the biggest flaws may be in personal protection. Our solution will be better because it will incorporate many of the features of our competitors into one product as well as seeking to resolve some of the flaws we see in them.

Stakeholders

There are many potential stakeholders for this product. SecurePack is targeting people who live and/or work in urban settings and want to secure themselves against threats such as muggings and sexual assault. The stakeholders that may be interested in this product include students, parents, law enforcement, and people in urban environments.

Students would be interested in our product because they are highly likely to carry backpacks. Parents would likely value the GPS tracker in the system to be able to know where their children are. Law enforcement would be interested in our product because video captured from the camera on the backpack is intended to be used as evidence in the event of an attempted crime on the user or against others. People living in urban areas are also stakeholders because this product is intended to help ensure their safety as the backpack is intended to ward off potential attackers that are more common in urban areas. Another group of people who may be interested in this product are those who wish to keep their devices charged on the go. This is because an intended feature of this backpack is to be able to charge devices from the battery pack in the backpack.

Constraints, Requirements, and Selected Components

This product will be evaluated by how quickly (in seconds) the defense and recording mechanisms can be activated, how far away help can be called (in miles), how heavy the backpack is (in pounds), how long can the entire system be active (the battery life in hours), and how much does the backpack cost (in dollars). Generally, the product needs to be as efficient as possible, as reliable as possible, as light as possible, able to withstand everyday life, and be as cost efficient as the materials allow.

As a standard, the SecurePack should be able to respond in less than a second. It should be able to send signals anywhere with a cell signal. It also should weigh less than 5 pounds. The SecurePack will also be able to last for more than 16 hours on its battery life and the prototype of the SecurePack is aimed to cost no more than $300.

These values were chosen in order to attempt to meet or exceed customer expectations. A response time of less than a second is necessary for the SecurePack to serve its function. A 16 hour battery life is reasonable to allow the user to walk around with the device for a day without charging. Five pounds is reasonable as to not burden the user with excess weight and a cost of no more than $300 means that consumers may be more willing to purchase the product. It is also worthwhile to note that mass production of this product would likely reduce the cost of a single SecurePack

The chosen components include a waterproof backpack, a high lumen flashlight, a Raspberry Pi 3 B+, USB speakers, LED light strips, a Raspberry Pi GPS Module, a Raspberry Pi camera, and a power bank. The SecurePack will be equipped with an emergency button which, when activated, will begin recording with the camera and activate the GPS tracker. This data will then be transferred to the Raspberry Pi and be sent to the user's phone and then from the users phone to a trusted third device such as a relatives phone or to a personal computer. Once the camera is activated, the bright light can also be activated potentially disorienting a potential attacker. The speakers on the SecurePack will serve as an audible alarm which will alert anyone within earshot of a potential threat. It is of note that this alarm could be a loud noise or pre recorded message. If there are other users of the SecurePack within the Bluetooth signal range they will be alerted that someone nearby is potentially in danger. Additionally, in everyday use the user will be able to charge devices such as cell phones and tablets via the backpacks battery. The user will also be able to activate LED's on the backpack to make the user clearly visible in the dark. The iOS app is included with a code which needs to be redeemed for set up on their iPhone. Based off of the processing speed of the Raspberry Pi 3 B+ and the lack of any slow mechanical components, the response time of the system should easily be able to be under one second.

Power Requirements

The various devices in the SecurePack consume varying amounts of power in both active and idle states. Table 1 shows the current draw for each device. Note that some of the idle current draw figures are estimates based on similar products and not the exact device used.

Table 1: Current Draw

Device	Idle Current Draw (mA)	Active Current Draw(mA)
Backpack Charger	1	1,000
Raspberry Pi 3 B+ with Camera	400	520
GPS Module	1	20
High Lumen Flashlight	16	1,600
Speakers	10	1,000
LED's	1	60
Total Current Draw	**429**	**4,200**

The power bank that was chosen is rated at 13,000mAh, therefore by using Equation 1 the duration of the SecurePack's battery can be calculated.

Equation 1: $Battery\ Amp\ Hours \div Current\ Draw = Time\ Until\ Battery\ is\ Depleted$

This means that the power bank can last for about 30 hours while inactive or a little over 3 hours if the system is completely active. It is reasonable to assume that if an emergency were to occur the system would not need to be fully active for more than an hour. Given this information it can be calculated that the system could run for one hour fully active and about 20 hours and 30 minutes idle. This means that the system should fully exceed its evaluation metric of being able to last for more than 16 hours on a single charge.

Weight Requirement

The weights for each component of the SecurePack needs to be obtained in order to find the total weight. Based off of the results from Table 2 the total weight of the secure pack will be 4.12 pounds. This meets our evaluation metric of being less than 5 pounds.

Table 2: Weight of the SecurePack

Device	Weight
Waterproof Backpack	1.6(lbs)
USB Speakers	11(oz.)
Adafruit GPS Module	0.6(oz.)
Raspberry Pi 3 B+	2.03(oz.)
Raspberry Pi Video Camera	0.64(oz.)
Power Bank	9(oz.)
LED's	11.1(oz.)
Flashlight	6(oz.)
Total Weight	**4.12(lbs)**

Cost Requirement

In order to determine the cost of materials for the SecurePack, the price of each component must be determined. Table 3 shows the price of each device and the total cost of materials for the SecurePack. Based off of the results of Table 3, the cost of the materials is $196. This meets the evaluation metric of being less than $300.

Table 3: BOM Raspberry Pi Option

Item	QTY	Price

Backpack (waterproof)	1	$30
Raspberry Pi 3 B+	1	$35
High Lumen Flashlight	1	$8
Speakers (2) (USB)	1	$10
Power Bank	1	$35
Raspberry Pi Camera	1	$13
GPS Module	1	$45
LED’s	1	$17
USB to Micro USB Cable	1	$3
Total	**N/A**	**$196**

Methodology

There are a large number of connections that need to be made in order to be able to control all of the devices at once. Every device will have signal connections with the Raspberry Pi. The waterproof backpack that will be purchased includes a built in USB cable that is capable of charging smart devices. This USB cable will be connected to one of the two USB Ports located on the Power Bank. The Raspberry Pi will be connected to the last USB port on the power bank via a USB to micro USB cable. Every other device will receive its power from the Raspberry Pi with the exception of the high lumen flashlight. The flashlight will receive its power from the batteries that come standard with the flashlight. Signal wires coming from ports on the Raspberry Pi will replace the flashlight’s built in pushbutton and mimic it in order to control the flashlight. The Raspberry Pi, GPS module, and camera are all built by adafruit and so the GPS module and the camera are both designed to be connected directly to the Raspberry Pi. The speakers that will be purchased are USB speakers and will thus be connected via one of the USB ports on the Raspberry Pi. The LED strip will receive its power and signals via the Raspberry Pi. An example of the connections is shown in Figure A.

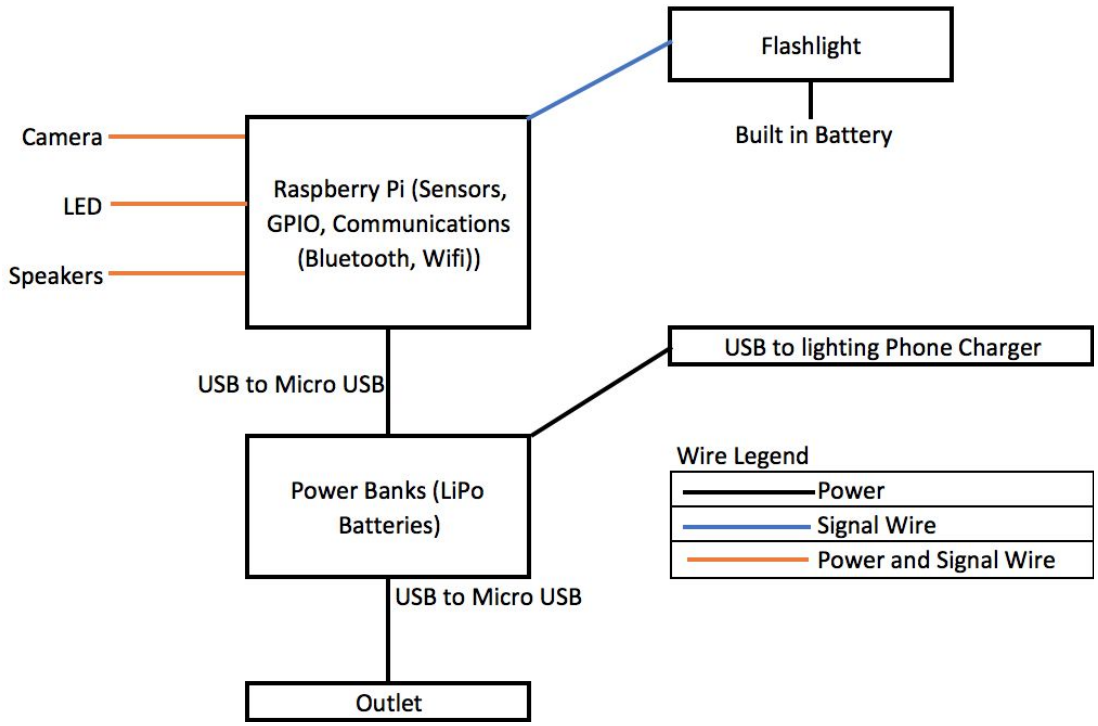

Figure A: Power and Signal Wiring Diagram

Component Usage

The high lumen flashlight will be useful in multiple aspects. The primary function of the high lumen flashlight will be to use it as a disorienting bright light against potential attackers. This method is incredibly effective at night or in dark places such as an alleyway. While it is not as effective during the day, the high lumen flashlight is still blinding in broad daylight. According to PoliceMart, a distributor of tactical flashlights, 300 to 500 lumens is enough to blind an assailant for a few minutes in broad daylight[4].The flashlight being used for this project is advertised to have a maximum brightness of 2000 lumens. This flashlight also serves as the purpose of ensuring that while the camera is recording, there will always be light available. The backpack will also allow the option of being able to just turn on the flashlight and use it for traveling in the dark.

The camera on the backpack serves the purpose of being evidence in emergency situations. The camera will be able to record and store this evidence within the backpack as well as send it to a trusted device, such as a loved one's phone or a personal computer.

The Bluetooth module serves the purpose of allowing the backpack to communicate with the user's phone and thus be able to send that information to any device connected via cell tower. The Bluetooth also allows the backpack to potentially send signals to other backpacks nearby, which could inform them of the danger or trigger their cameras to start recording. The GPS module serves the purpose of being able to track the location of the user. This will be useful in an emergency situation in which the user can send their location to a trusted device or law enforcement. This device will also serve the purpose of informing the micro-controller when the user is moving so the device knows when to activate itself.

The backpack itself serves two main functions. The first being a waterproof casing to prevent water damage to the electronics inside of the backpack. The second being a backpack to store whatever the user wants to store inside of it. The LED light strips will serve the purpose of lighting up the user to make their presence known to traffic. The speakers will serve the purpose of being a warning for any potential attackers to back off. The audio clip played will have the intention to defuse the situation. If this fails, then the speakers will play a loud alarm sound file. This will have the intention to alert anybody within earshot that the user is in danger and needs assistance.

The power bank will be the power supply of the backpack ensuring that the backpack can handle the power requirements of the various devices. This includes charging devices such as the user's smartphone or tablet. Next, the Raspberry Pi will serve as the brains of the backpack.

Lastly, an iOS app will be developed to be able to communicate with the system as well as provide useful messages for users of SecurePack. The features of the app are currently planned to include a message board, an alert system for informing users of an assault or other crime in a particular area, information regarding where the backpack is by means of GPS (in case it gets stolen), a way of transfer to import pictures and/or video from the camera (including audio), a button to turn on the system as well as its components (i.e. to start filming with the camera), a button to turn off the system, and the user's profile. As this product is a prototype, the Apple iPhone is the focus of the app, rather than Android. However, if this product were to grow in user size, an Android version of the application could surely be developed within a short time as the blueprints for the app are already laid out in the current version.

Current Progress

For the project, progress is being measured by the different components that work and can be communicated throughout the system. For example the power bank has successfully been connected to the raspberry pi and it can thus be powered via the power bank. The speakers are another example of a component that works through the use of Python code directing the Raspberry Pi to play a sound sample. An example of this code can be found in Code Example 1 in the appendix.

The Raspberry Pi camera functionality is now complete and operates by the "picamera" Python library being imported. The PiCamera() function has preview and recording methods which allows the camera to record for a specific amount of time (using time.sleep() to specify the amount of time). On top of this, the video, which is recorded in .h264 video type, is able to be saved to the Pi's SD card with a unique name. The naming technique uses the current date time down to a nanosecond, which guarantees the no two videos have the same name.

The LED strip has successfully been controlled via the Raspberry Pi's GPIO ports. The circuit to help control the strip consists of 3 NPN mosfets in order to protect the Raspberry Pi and to control the Red, Green, and Blue pins on the LED strip. The LED strip is also able to be controlled via Python codes, which controls the GPIO pin signals. An excerpt of this code can be found in the appendix under Code Example 2.

The flashlight bought for this project utilizes a button located on the end of the flashlight. This button can be mimicked by connecting a wire from batteries to the metallic casing. Therefore, this flashlight can be controlled via a simple transistor-relay circuit. Voltage from a GPIO port can trigger a transistor which will then switch a relay and thus connect the battery from the flashlight to the metallic casing. This process then behaves as if the button was pressed on the flashlight.

The GPS communication also works with Python code. For each time the SecurePack is turned on, the GPS module needs to be turned on by the system program which is done by Python. The program uses threading to allow all of the information that can be processed by the GPS to be printed to a file and then sent elsewhere. A GPS Poller class is ran so that the GPS is on continually and doesn't stop checking where the the Pi's location is.

The core of the system, Bluetooth, is also able to send files from one Raspberry Pi to the other manually via Bluez/Blueman. A Python code has also been written for Bluetooth communication though it still has some kinks that need to be worked out for it to be fully operational.

Finally, the iOS app is currently being created using Xcode as the IDE and Swift as the programming language. The app is also using Firebase to as the method to store user data (usernames, passwords, etc.). The functionality to Sign Up/Sign In to an account on the app is done as well as the visual interface for that to happen. The "home" page is a simple blue screen with two text fields to enter email and password and then a button to send the data to be checked. If a particular user does not exist yet and those fields are entered, the user will be taken to a screen where they can upload a profile photo and create a username. After that is done, they are sent to the main area of the app, which is a Twitter-style posting page. This is where users are able to share about any suspicious activity in a certain area so that others are aware. This is also where an automated message will pop up if a panic button is hit on the backpack, sharing GPS data and a warning to users in a particular area. A few example screenshots of the progress so far can be found in the appendix with Figures B, C, and D.

Future Work

While Each of the components of the SecurePack have been tested individually, they still need to be tested to see if they work together. Additionally each of the components is in the process of being attached onto the backpack itself.

The majority of the future work is on the iOS application. Sections of the app that will be worked on in the coming days are a page to modify profile information, GPS location from the Pi, and Bluetooth communication between the app and the Pi, so that the app may control some functionalities of the SecurePack as well.

The Bluetooth connectivity between multiple RPi's also will be finished in the coming days. There are some kinks and inconsistencies with using the current Python solution for communication, so a better, more reliable method is being researched and will be implemented. On top of this, the Raspberry Pi needs to take advantage of threading/processes in the Linux terminal so that all functionality of the system can be fired at the same time, rather than in sequential order. Other than these few things, the SecurePack system is nearing the end of R&D.

Conclusion

In conclusion, the creation of SecurePack is viable and may likely be in demand in the future. As laid out in the paper, the SecurePack meets or exceeds all of its constraints, allowing it to perform its main function of protecting the user effectively. Each of the components on the

SecurePack are intended to protect the user in an emergency as well as providing useful features for day to day life. The camera can be utilised to record criminal activity as well as take a family picture. The flashlight can be used to ward off an assailant and light any area in the dark. The features are intended to allow the SecurePack to be useful even if the user never needs to use its main emergency function. On top of all of this, the iOS application will allow the SecurePack to connect other users and to keep them in the loop of any happenings in a specific area. All of these functionalities and features will allow this product to be reliable, flexible, and desirable.

References

[1]. Chicago Police, "Chicago Police Crime Statistics" Home.chicagopolice.org, [online] Available: https://home.chicagopolice.org/wp-content/uploads/2019/01/1_PDFsam_CompStat-Public-2019-Week-3.pdf [Accessed 30 Jan. 2019].

[2]. Gallup, Inc, "How Americans Protect Themselves From Crime," *Gallup.com*, 26-Oct-2007. [Online]. Available: https://news.gallup.com/poll/102418/how-americans-protect-themselves-from-crime.aspx .[Accessed: 29-Sep-2018].

[3]. "iHuniu LED Turn Signal Light Reflective Vest Backpack/Waist Pack/Business/Travel/Laptop/School Bag Sport Outdoor Waterproof for Safety Night Cycling/Running/Walking" *Amazon*. [Online]. Available: https://www.amazon.com/iHuniu-Reflective-Backpack-Business-Waterproof/dp/B013N30PVO

[4]. "Tactical Level Lighting," *Police Magazine*. [Online]. Available: http://www.policemag.com/channel/patrol/articles/2012/12/tactical-level-lighting.aspx. [Accessed: 29-Sep-2018].

[5]. "TYLT Energi Pro Power Backpack with Charging Station - Charge Up to 3 Devices at Once via USB or USB Type-C Ports, TSA Approved Laptop and Mobile Device Travel Bag with Built-in Power Bank" Amazon. [Online]. Available: https://www.amazon.com/s/?ie=UTF8&keywords=tylt+backpack&tag=googhydr-20&index=aps&hvadid=241929106783&hvpos=1t1&hvnetw=g&hvrand=7001158769568149193&hvpone=&hvptwo=&hvqmt=e&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9015966&hvtargid=kwd-493435852072&ref=pd_sl_1ol7k6qy8z_e_p38

Appendix

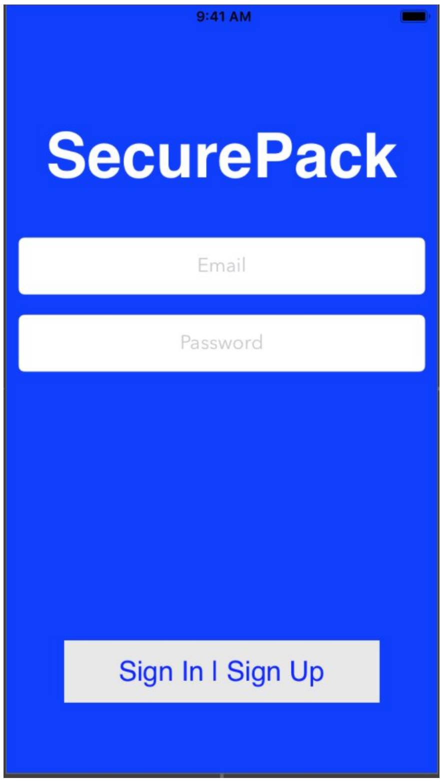

Figure B: App Sign In/Sign Up Screen

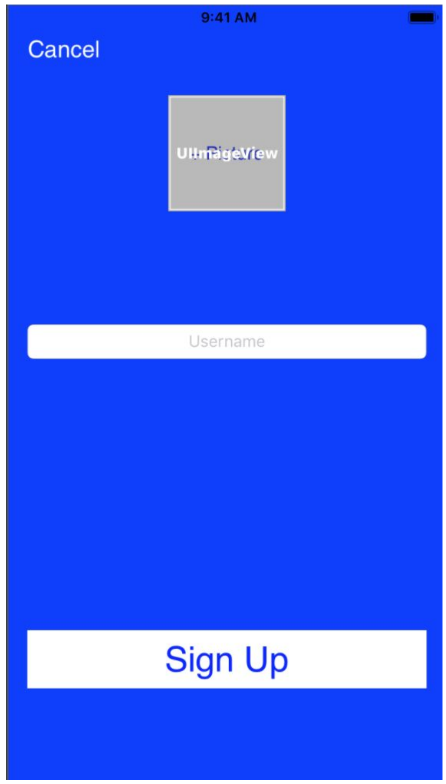

Figure C: Profile picture and username screen.

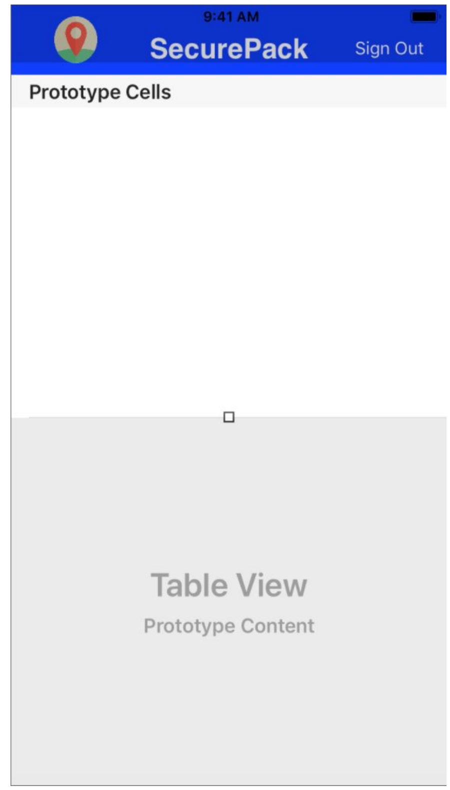

Figure D: Beginning stages of the feed screen.

Code Example 1:

```
pygame.mixer.init()
pygame.mixer.music.load("Siren_noise.mp3")
pygame.mixer.music.play(X)
while pygame.mixer.music.get_busy() == True:
        continue
```

Code Example 2::

```
import RPi.GPIO as GPIO #Import Raspberry Pi GPIO library
import time #import time library

GPIO.setwarnings(False) #Ignore warning for now
GPIO.setmode(GPIO.BOARD) #Use physical pin numbering

GPIO.setup(3, GPIO.IN, pull_up_down=GPIO.PUD_DOWN) #set pin 3 as an input (button)
GPIO.setup(10, GPIO.IN, pull_up_down=GPIO.PUD_DOWN) #set pin 10 to be an input pin and
set inital value to be pulled low (off)
```

```
GPIO.setup(5, GPIO.OUT) #set pin 5 as the green LED
GPIO.setup(7, GPIO.OUT) #set pin 7 as the yellow LED
GPIO.setup(12, GPIO.OUT) #set pin 18 as the red LED

def buttonPress(channel):
    print "Green LED on"
    GPIO.output(5, GPIO.HIGH)  #turn on LED
    time.sleep(1)
    print "Green LED off"
    GPIO.output(5, GPIO.LOW)   #turn off LED
    time.sleep(1)

    print "Red LED on"
    GPIO.output(12, GPIO.HIGH)  #turn on LED
    time.sleep(1)
    print "Red LED off"
    GPIO.output(12, GPIO.LOW)   #turn off LED
    time.sleep(1)

    print "Yellow LED on"
    GPIO.output(7, GPIO.HIGH)  #turn on LED
    time.sleep(1)
    print "Yellow LED off"
    GPIO.output(7, GPIO.LOW)   #turn off LED
    time.sleep(1)

def secondButtonPress(channel):
    print "Christmas lights!"
    GPIO.output(5, GPIO.HIGH)
    time.sleep(2)
    GPIO.output(12, GPIO.HIGH)
    time.sleep(2)
    GPIO.output(7, GPIO.HIGH)
    time.sleep(2)

    GPIO.output(5, GPIO.LOW)
    time.sleep(2)
    GPIO.output(12, GPIO.LOW)
    time.sleep(2)
    GPIO.output(7, GPIO.LOW)
    time.sleep(1)

GPIO.add_event_detect(10, GPIO.RISING, callback = buttonPress, bouncetime = 300)
GPIO.add_event_detect(3, GPIO.RISING, callback=secondButtonPress, bouncetime = 300)


message = input("Listening\n\n")  #starting message

GPIO.cleanup()
```