



Blue Sheep

A Mobile Data Collection Application

Final Project Report

CSU San Marcos
Fall 2017 CS 441
Software Engineering

blueHat Technologies

Paul Thomas Rowe
Raul Perez
Mike Yoon
Takuro Iwane





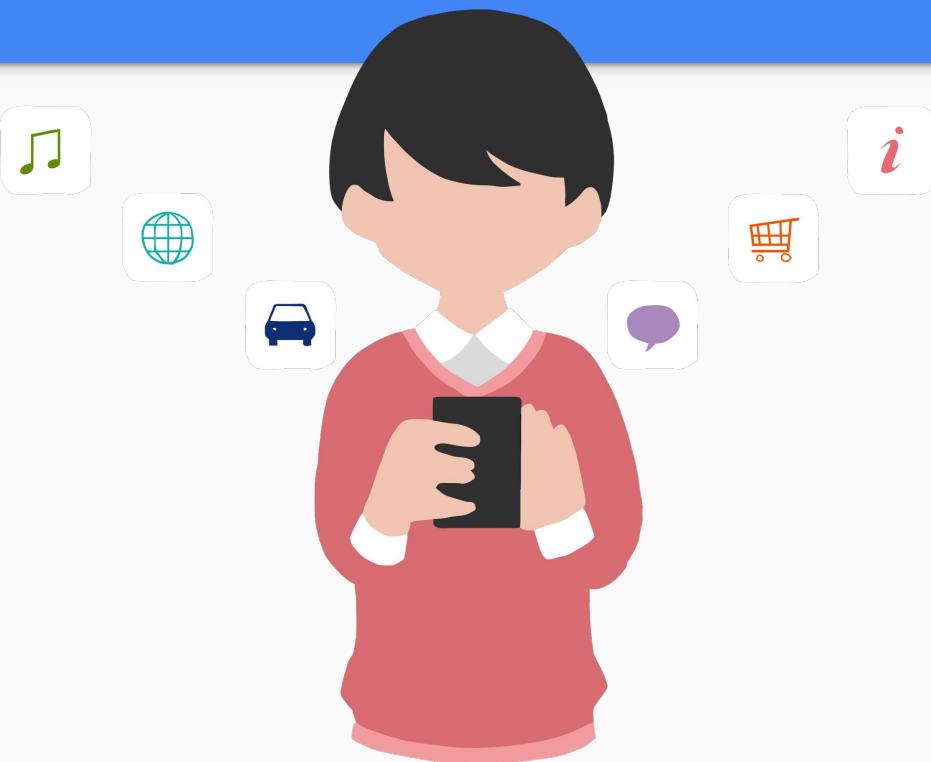
PROJECT DESCRIPTION

Modern mobile devices of today are capable of collecting huge amounts of data through their sensors and software.

A problem that is particularly present amongst users is that a majority will readily grant any application, program, or piece of software full access and permission to their personal devices.,

By doing so, seemingly benign applications are capable of keeping track of a user's location, movements, activities, and preferences to name just a few categories.

We argue that most users of such mobile devices are entirely unaware that even just through regular use, may allow their phones to collect huge amounts of unique data about their day to day lives.





USER REQUIREMENTS

Android running 6.0 or later

iPhone running iOS 8.0 or later

Internet Connection

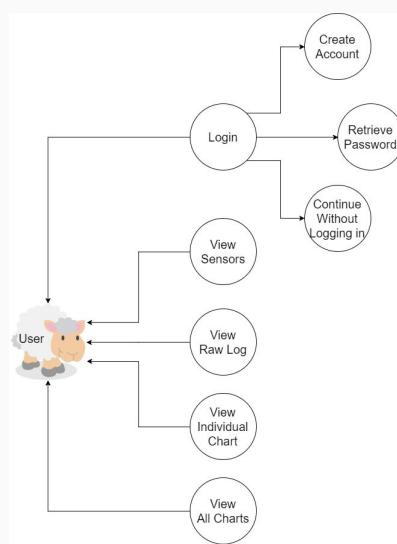
Launch application at least once



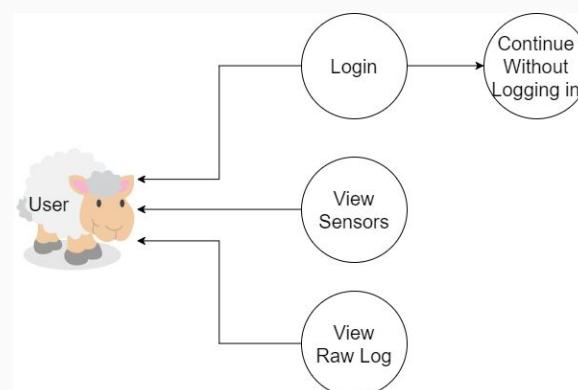


MOST COMMON USE CASE DIAGRAMS

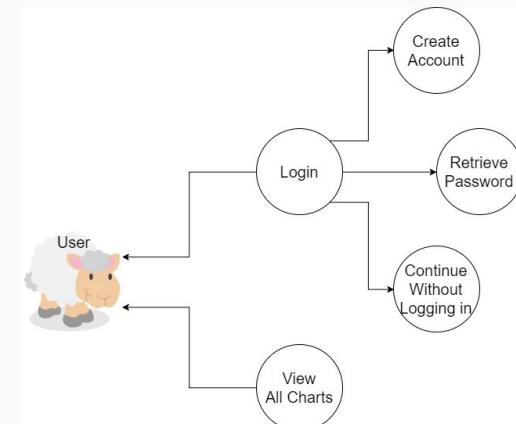
Normal Operation



Offline Mode



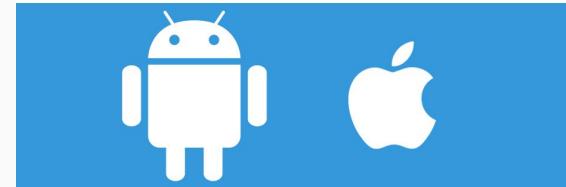
Sensors Disabled





IMPLEMENTATION MILESTONES

1. Xamarin Studio (C#)
2. Detecting Specific Sensors
3. Web Server (AWS)
4. Log File Creation
5. File Parsing
6. Visualizations
7. File Transfer Protocol (SFTP, Rebex)
8. Web Page



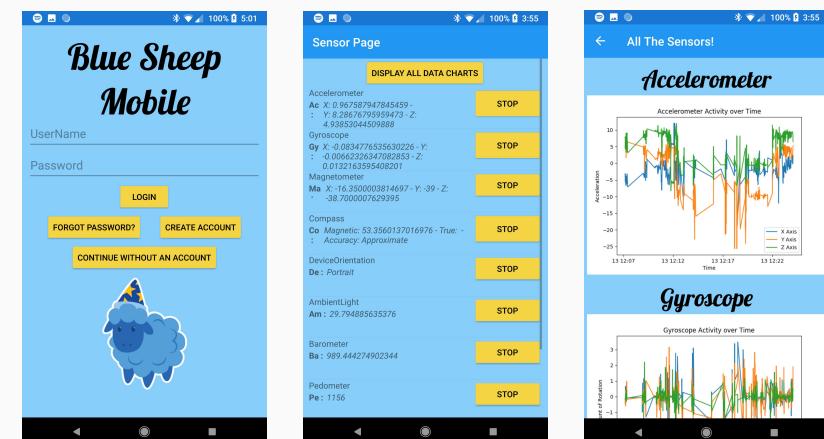


Blue Sheep Mobile Development Cycle

Blue Sheep Mobile 1.0



Blue Sheep Mobile 2.0





Blue Sheep With a Red Hat Development Cycle (Paul Thomas Rowe)

```
Jul 02 03:47:41 server.rheltest.lan systemd[1]: Starting The Apache HTTP
Jul 02 03:47:41 server.rheltest.lan systemd[1]: Started The Apache HTTP
[root@server ~]# _
```

```
[root@server ~]# yum search php
Loaded plugins: product-id, subscription-manager
```

graphviz-php.x86_64 : PHP extension for graphviz

php.x86_64 : PHP scripting language for creating dynamic web sites
libbcmath-x86_64 : A module for BDFL applications for using the bcmath library

QUESTION ANSWERED
DHCP.exe
→ Documentation & Help | Design & Development

adonis.13.gz:Mar 28 07:44:39 144.37.1.253 dhcpcd: DHCPDISCOVER from 00:24:d7:2d:b2:17 via 144.37.188.165
adonis.15.gz:Mar 28 07:44:40 144.37.1.253 dhcpcd: DHCPOFFER on 144.37.188.165 to 00:24:d7:2d:b2:17 |

adonis.13.gz:Mar 20 07:44:49 144.37.1.253 dhcpd: DHCPRQUEST for 144.37.108.168 (144.37.1.254) from adonis.13.gz:Mar 20 07:44:49 144.37.1.253 dhcpd: DHCPACK on 144.37.108.168 to 00:24:d7:2d:b2:17 (17) adonis.13.gz:Mar 20 07:44:49 144.37.1.253 dhcpd: DHCPACK to 144.37.108.168 (00:24:d7:2d:b2:17) via

```
adonis.13.gz:Mar 28 07:45:44 144.37.1.253 dhcpd: DHCPACK to 144.37.188.168 (00:24:d7:2d:b2:17) via  
adonis.13.gz:Mar 28 07:46:56 144.37.1.253 dhcpd: DHCPACK to 144.37.188.168 (00:24:d7:2d:b2:17) via  
adonis.13.gz:Mar 28 07:48:08 144.37.1.253 dhcpd: DHCPACK to 144.37.188.168 (00:24:d7:2d:b2:17) via
```

adonis.13.gr:Mar 20 07:48:22 144.37.1.253 dhcpcd: DHCPREQUEST for 144.37.108.168 from 00:24:d7:2d:b2:0
adonis.13.gr:Mar 20 07:48:22 144.37.1.253 dhcpcd: DHCPCMAX on 144.37.108.168 to 00:24:d7:2d:b2:17 via
adonis.13.gr:Mar 20 07:48:22 144.37.1.253 dhcpcd: DHCPRISONER from 00:24:d7:2d:b2:17 via 144.37.108.168

adenos.13.gz:Mar 20 07:48:23 144.37.1.253 dhcpcd: DHCPREQUEST on 144.37.100.183 to 0:0:24:d7:2d:b2:17 (I
adenos.13.gz:Mar 20 07:48:23 144.37.1.253 dhcpcd: DHCPREQUEST for 144.37.100.183 (144.37.1.254) from

```

adensis.13.gz:Mar 20 07:48:27 144.37.1.253 dhcpd: DHCPACK to 144.37.100.183 (00:24:07:2d:b2:17) via
adensis.13.gz:Mar 20 07:49:59 144.37.1.253 dhcpd: DHCPACK to 144.37.100.183 (00:24:07:2d:b2:17) via
adensis.13.gz:Mar 20 07:50:00 144.37.1.253 dhcpd: DHCPACK to 144.37.100.183 (00:24:07:2d:b2:17) via

```

```
adenis.13.gi:Mar 20 07:50:06 144.37.1.253 dhcpd: DHCPREQUEST for 144.37.100.183 from 00:24:d7:2d:b2:17 via 144.37.100.1  
adenis.13.gi:Mar 20 07:50:06 144.37.1.253 dhcpd: DHCPWIAK on 144.37.100.183 to 00:24:d7:2d:b2:17 via 144.37.100.1  
adenis.13.gi:Mar 20 07:50:06 144.37.1.253 dhcpd: DHCPOFFER from 00:24:d7:2d:b2:17 via 144.37.100.1  
adenis.13.gi:Mar 20 07:50:06 144.37.1.253 dhcpd: DHCPDISCOVER from 00:24:d7:2d:b2:17 via 144.37.100.1
```

adonis.13.gz:Mar 28 07:58:07 144.37.1.253 dhcpd: DHCPOffer on 144.37.108.168 to 00:24:d7:2d:b2:17 |

3c:15;c2:05;10:41 Darcy's phone 17 08:35:0
d4:f4:6f:ed:9c:33 Darcy-HBP 17 08:35:0
17 08:35:4

17 00:35:5
17 00:36:5
17 00:36:5

```
ParseCommands.sh
# Abbreviating the logs:
#!/bin/bash
#DEVICE NAMES
cat ./sourceLogs/DHCP.txt | grep DHCPCACK | sed 's/.*/\n/p' > ./parsed/DeviceNames.txt
#DHCP
cat ./sourceLogs/DHCP.txt | grep "DHCPACK" | cut -d ' ' -f 1 | sort -n > ./parsed/DHCParsedOutput.txt
#WIRELESS
cat ./sourceLogs/wireless.txt | grep "AP-name" | cut -d ' ' -f 1 | sort -n > ./parsed/wirelessParsedOutput.txt
#MAC ADDRESS
cat ./sourceLogs/wireless.logs | grep "suspect"
| uniq -f 4 | grep -v "de-authenticated" > ./parsed/wireless_logsParsedOutput.txt
#WIRELESS LOGS
cat ./sourceLogs/wireless.logs.txt | grep "suspect"
| uniq -f 4 > ./parsed/wireless_logsParsedOutput.txt
#LOCATIONS
cat ./parsed/wirelessParsedOutput.txt | grep -f ./sourceLogs/locations.txt
| cut -d ' ' -f 1,2,7 | sed 's/^\$AP-name/\$/' > ./parsed/wireless_logsParsedOutput.txt
#DEVICE NAMES
cat ./parsed/DHCParsedOutput.txt | grep -f ./sourceLogs/deviceNames.txt
| cut -d ' ' -f 5 | sed 's/^\$/\$./' | awk '{print $1"\\"$2}' | uniq > ./parsed/DeviceNames.txt
#GRAPH
data= pd.read_csv("cpgpa.csv")
data=data.head(30)
pt.plot(data["rollno"],data["suspect"])
pt.scatter(x,y,color="red")
pt.xlabel("X Axis")
pt.ylabel("Y Axis")
pt.title("Graph")
pt.xlim(-10,20)
pt.ylim(-20,20)
#GRAPH
x=[1,2,3,4,5,6,7]
y=[5,-2,3,4,6,9,1]
pt.scatter(x,y,color="red")
```

 Aclog.csv	 Aclog.png	 Aclog.py	 Amlog.csv	 Amlog.png	 Amlog.py	
 Balog.csv	 Balog.png	 Balog.py	 Colog.csv	 Colog.png	 Colog.py	
 reqInvoke.sh	 Gylog.csv	 Gylog.png	 Gylog.py	 looper.sh	 Malog.csv	
 Malog.png	 Malog.py	 Pelog.csv	 Pelog.png	 Pelog.py		

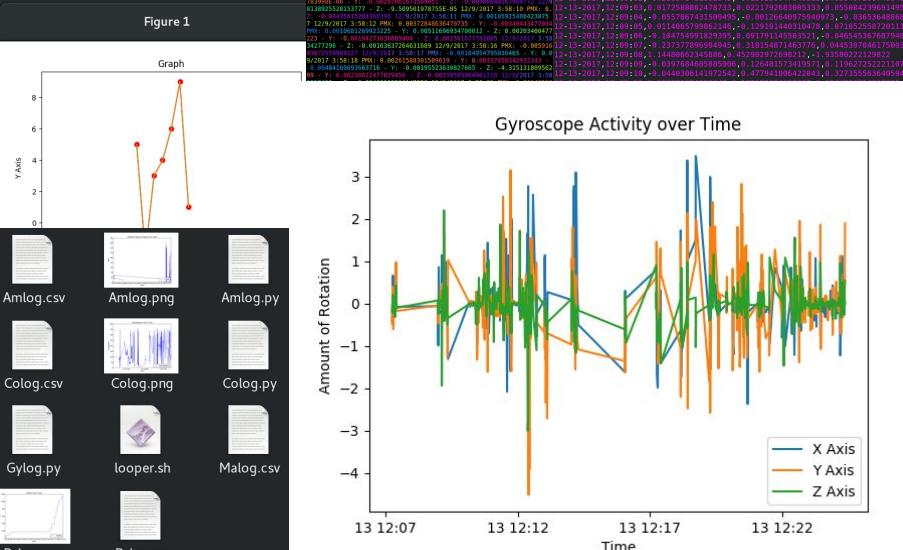
The figure consists of two side-by-side plots sharing a common x-axis representing time.

Left Plot (Graph): The y-axis is labeled "Y Axis" and ranges from 0 to 8. The data points are red circles connected by a solid orange line. The activity starts at approximately (12:00, 5), peaks at (12:05, 7), dips at (12:10, 2), peaks again at (12:15, 6), dips at (12:20, 1), and ends at (12:25, 2).

Right Plot (Gyroscope Activity over Time): The y-axis is labeled "Y Axis" and ranges from 0 to 3. The data points are blue squares connected by a solid blue line. The activity shows several sharp spikes: one at ~12:05 (height ~2.5), another at ~12:10 (height ~2.8), a third at ~12:15 (height ~3.0), a fourth at ~12:20 (height ~2.5), and a fifth at ~12:25 (height ~2.8). A horizontal grey line is drawn at Y=1.0.

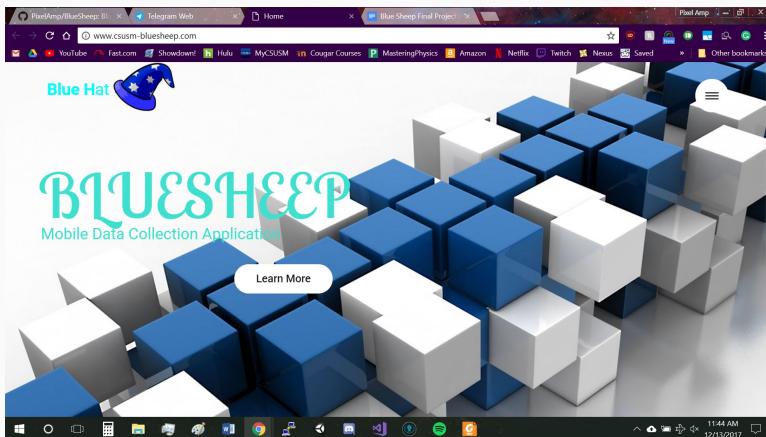
Data Labels:

- Graph: "cpga", color="orange", label="line graph"
- Gyroscope Activity over Time: "cpga", color="blue", label="line graph"





www.csusm-bluesheep.com



ABOUT BLUE SHEEP

Blue Sheep is a Mobile Data Collection Application that allows users to see how much information they unknowingly produce through their phone's various sensors. We will use various algorithms to make the data collected comprehensible through the use of charts and/or models, displaying just the user's personal data, and how their habits fit in against as a whole.

WHY NEED THIS APP?

A problem that is particularly present amongst mobile phone users is that a majority will readily grant any application, program, or piece of software full access and permission to their personal devices, including phones, computers, and tablets.

PixelAmp/BlueSheep | Home | Blue Sheep Final Project | PixelAmp

11:44 AM
12/13/2017

YouTube Fast.com Showdown! Hulu MyCSUSM Cougar Courses MasteringPhysics Amazon Netflix Twitch Nexus Saved Other bookmarks

PixelAmp/BlueSheep | Home | Blue Sheep Final Project | PixelAmp

11:44 AM
12/13/2017

YouTube Fast.com Showdown! Hulu MyCSUSM Cougar Courses MasteringPhysics Amazon Netflix Twitch Nexus Saved Other bookmarks

Problems and Setbacks

- What mobile development environment to use?
 - Xamarin or Android Studio
- Where and how to host our web server?
 - Amazon AWS or Personal PC or Raspberry Pi
- How to collect data inside the mobile device
 - Brute force each sensor or find a plugin that does it all for us
- Data visualization
 - Client side or server side?
- **How to communicate between the Mobile Application and the server?**
 - JSON HTTP post requests or FTP or SFTP
- Scheduling meeting times



CONCLUSION

- Knew this project was going to be very difficult
- None to little experience with the technology
- Ambitious, determined, and optimistic attitude
- Fought every challenge, achieving goals
- Find a solution for every problem
- Proud of our project
- Better Software Engineers

