ASHESI

ASHESI UNIVERSITY COLLEGE

TRACKING AND LOCATING PROPERTY ON UNIVERSITY CAMPUSES

APPLIED PROJECT

B.Sc. Computer Science

Efua Bentsiwa Bainson

2017

ASHESI UNIVERSITY COLLEGE

Tracking and Locating Property on University Campuses

APPLIED PROJECT

Applied Project submitted to the Department of Computer Science, Ashesi University College in partial fulfilment of the requirements for the award of Bachelor of Science degree in Computer Science

Efua Bentsiwa Bainson

April 2017

DECLARATION

I hereby declare that this Applied Project is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

……………………………………………………………………………………………

Candidate's Name:

……………………………………………………………………………………………

Date: ……………………………………………………………………………………………

I hereby declare that preparation and presentation of this Applied Project were supervised in accordance with the guidelines on supervision of Applied Project laid down by Ashesi University College.

Supervisor's Signature:

……………………………………………………………………………………………

Supervisor's Name:

……………………………………………………………………………………………

Date: ……………………………………………………………………………………………

Acknowledgement

I would like to express my sincere gratitude to God for the strength to complete this project.

To my supervisor and mentor, Dr. Nathan Amanquah and the computer science department, I am thankful for the unfailing support and assistance. I am also thankful to Mr. Osafo Marfo for his indispensable contribution to this project. I am grateful to Mr. Nicholas Tali for his assistance.

To the three life coaches I have, Dr. Kobina Atta Bainson, Mrs. Patricia Antwi, and Ama Bembua Bainson: I owe it all to you. Many thanks!

Abstract

The problem of property crime exists on university campuses worldwide. There is also difficulty in monitoring and determining the location of items. Organizations have sought to rely on tracking devices to curb this issue. However, these devices are either too expensive, require high power consumption or are bulky and cannot be used to track small devices. In this project, an affordable and portable system is developed that monitors devices and alerts users once their items cross boundaries (geo fences).

.

Table of Contents

CHAPTER 1: INTRODUCTION

1.1 BACKGROUND CONTEXT

Each year, millions of students around the world are obliged to live on their own when they enter university. Enabling students to live independently increases the risk of crime for both students and university authorities, especially where security is lax.

Property crime is a worldwide issue, and college campuses are no exception. An analysis of the 2010 Federal Bureau of Investigation (FBI) crime statistics shows that universities are vulnerable to theft. (Ebbett, 2016). According to Ebbett, in 2011 the FBI reported that there were more than two million burglaries reported in universities in the United States of America, making up approximately 25% of all property crimes. This shows how susceptible students and school authorities are, to property crime. This is especially true for new students who have to live with complete strangers who may or may not be criminals. University authorities also have difficulty in locating the position of public goods. Items such as projectors, laptops, and chairs which are available for the public are difficult to track. In Ashesi University in Ghana for instance, student interns and employees have to go around the school looking for the location of assets. Thus, for students as well as school authorities, the security and location of movable assets is a big issue.

The outcome of this lapse in security is affecting students and school authorities as theft cases are consistently on the rise in many universities. On December 6, 2014, concerned students of the University of Ghana appealed to the Vice Chancellor to respond quickly to the security situation on their campus (Ngula, 2014). According to the author of the article, this followed a walk by some of the students to inform the school authorities of their displeasure about the rise in theft cases. Likewise, the number of theft cases continue to rise each year in Ashesi University as illustrated in Figure 1.1. Out of the total number of cases reported, culprits were identified in only 46% of cases. In fact, the problem is much larger because many cases are not reported by the students.

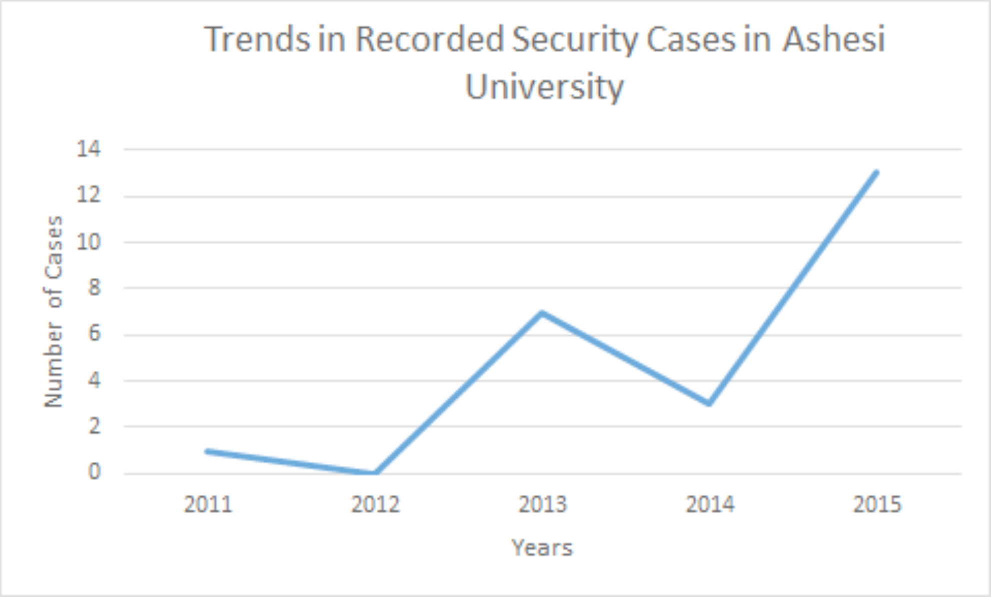


Figure 1.1: Trends in theft cases in Ashesi University

In response to this issue, authorities of institutions put some measures in place to curb it. The introduction of CCTV cameras in Ashesi University in 2016 is a solution which was as a result of the rise in theft cases in the school. Other proposed solutions include advising students to insure their devices, carry their devices with them all the time, take self-defense courses and lock the doors to their rooms. All these measures have helped reduce property crime, but have not been completely effective as indicated by statistics.

1.2 MOTIVATION

Over the years, the number of theft cases on university campuses has continued to rise. Students and school authorities have difficulty in locating items whether stolen or misplaced. With the increase in theft cases on campuses, there is an urgent need to provide an affordable and portable system that will monitor devices and alert users once their items cross boundaries (geo fences). Having identified this problem, this project seeks to develop a solution that could ultimately reduce the theft cases in universities and colleges.

1.3 BENEFIT OF PROPOSED SOLUTION

The proposed solution is an inconspicuous system attached to devices which can track and monitor their location. The system to be developed will incorporate a microcontroller and other electronics coupled with applications. Users can set up geo fences so that they can be notified once their items cross some boundaries. This will make it easy for users to note when items are moved about on the university campus. This will in turn, save users billions of dollars and traumatic experiences from losing valuables such as data. In addition to this, the data collected could be analyzed to provide new knowledge about the behavioral pattern of thieves.

1.4 RELATED WORK AND TECHNOLOGIES

1.4.1 Existing products

A number of systems have been developed to help users to locate their items. This section discusses some of these systems.

1.4.1.1 LoJack

LoJack is a well-known system used to track stolen vehicles (GUHA et al., 2012). The LoJack device which is always in receiving mode is hidden in the vehicle to be tracked. When a user reports a stolen vehicle, signals are sent to the car. If the vehicle is within reading range, the LoJack device is activated and the vehicle is recovered. Unfortunately, this requires that the LoJack device remains connected in order to be activated when a signal is sent (GUHA et al., 2012). This also requires frequent high-power transmission once activated, making it unsuitable for long-term, battery–powered operations (GUHA et al., 2012). Moreover, this device costs $995 which is relatively expensive for students to purchase. (Lojack.com*,* 2016)

1.4.1.2 Spark Nano 5.0 GPS device

This tracking device produced by Brickhouse Security uses Global Positioning System (GPS). This device is able to track items and inform users about their location using satellites. However, they consume a lot of power thus require frequent battery charging. This makes them unsuitable for use in tracking everyday items like laptops and televisions especially in developing countries like Ghana. Additionally, these GPS tracking devices are bulky. The Spark Nano 5.0 GPS Tracker is just about the size of a smartphone. A thief can easily disconnect the tracking device and take the smartphone. (GUHA et al., 2012)

1.4.1.3 AutoWitness

AutoWitness is a system built to fill the gaps created by other companies that produce tracking devices such as Brickhouse, Liveview, and LoJack (GUHA et al., 2012). The AutoWitness tag is embedded inside devices of users and becomes active when it detects vehicular movement (GUHA et al., 2012). The movement and position of the item is recorded and this is used to find the path of movement of the device (GUHA et al., 2012). According to the creators of AutoWitness, the basic requirement that users look out for in a tracking device is size, lifetime, and cost. The tag is small (51mm by 34 mm by 10mm), however, it costs $200 which seems rather expensive for students.

1.4.1.4 Antishoplifting RFID device

Antishoplifting RFID devices are used in shops to prevent shoplifting. With this, items in the shop have tags attached to them which respond to frequency emitted by a Radio Frequency Identification (RFID) reader attached to the exit (Woodford, 2016). RFID uses electromagnetic waves to identify objects attached to tags. Thus, when a customer tries to leave the shop with an item which has not been paid for, the RFID reader picks the signal and an alarm rings. The tag is deactivated when an item is paid for (Woodford, 2016). This is an efficient way of preventing shoplifting, however, the readers are bulky and expensive. The average price for an anti-shopping device on Alibaba.com is $300.

1.4.2 Key Technologies

This section discusses key technologies that can be used to make tracking devices. The technologies discussed below are Wi-Fi, GPS, Bluetooth and Radio Frequency Identification (RFID).

1.4.2.1 Wi-Fi

Wi-Fi is a widely known networking technology that uses radio waves to provide wireless internet (Beal, 2016). Every time a device tries to connect to the Wi-Fi, its MAC address is sent along to the access point to look for available networks (Hoffman, 2014). MAC address is a unique code which is specified in the hardware of a device (Moran, 2014). With some devices such as Android smartphones, even when the Wi-Fi is disabled, it still scans for Wi-Fi routers (Lehmann, 2015). This feature allows Wi-Fi to be used to track devices. Wi-Fi routers can be used to log MAC addresses of devices within range of an access point. With enough Wi-Fi routers joined together, it is possible to keep track of your device's movement (Hoffman, 2014).

1.4.2.2 GPS

Global Positioning System (GPS) is "a constellation of about 30 well-spaced satellites that orbit the Earth" (Peleg, 2016). It allows people to find the geographical location of items. It was initially designed for the US military but is now widely used (Bertagna, 2010). GPS equipment has become relatively low-cost so anyone can own a GPS receiver. However, GPS devices consume a lot of power. This makes them unreliable in tracking everyday devices such as mobile phones and laptops especially in countries which experience frequent power outages (Smith, 2011). Moreover, GPS devices are usually bulky hence not suitable for tracking small devices.

1.4.2.3 Bluetooth

Bluetooth is another radio-wave technology. It is, however, designed to communicate over short distances (Woodford, 2016). Bluetooth enabled devices have built-in radio transmitters and receivers that allow them to send and receive signals from other Bluetooth devices (Fitzpatrick, 2015). They automatically detect and connect to one another (Fitzpatrick, 2015). Bluetooth trackers communicate with phones or other devices of the user and alert him/her when there is a separation between the tracker and the object it is attached to. iBeacon is an example of a Bluetooth tracking device introduced by Apple which is a low-powered transmitter equipped with Bluetooth Smart that can be used to deliver information about location (Girish, 2015). Although it is more power efficient than Wi-Fi, it tends to drain power thus is also not suitable for everyday devices. (Girish, 2015)

1.4.2.4 Radio Frequency Identification (RFID)

Radio Frequency Identification uses electromagnetic waves to identify objects attached to tags. RFID systems consist of three parts: a reader, a tag and an antenna (AB&R, 2016). According to AB&R, digital data encoded in the tag are captured by a reader via radio waves when in range. The tag which contains an integrated circuit and an antenna receives the waves from the reader. It then sends back waves to the reader. The reader is made up of two parts – a transceiver and an antenna – which receives the radio waves and converts them into useful information. This information can be transferred to a computer to be used. There are two types of RFID tags namely; passive and active tags.

Passive RFID Tags

Passive tags have no internal power supply and instead rely on power from the RFID reader. These tags are powered by electromagnetic energy transmitted by the reader. The reader sends energy to the antenna which then converts the energy to radio frequency wave (Smiley, 2016). The tag's internal antenna draws power from the radio frequency wave once the tag is read. The energy powers the chip which generates a signal back to the radio frequency system. The reader detects the change in radio frequency wave and interprets the information. Generally, passive tags have a read range of up to 25 meters because the radio waves must be strong enough to power the tags (Smiley, 2016). The read range is, however, dependent on a number of factors such as the size of the tag antenna and the type of material the tag is attached to. Low-frequency passive tags with a frequency of 125 to 134 MHz read distance of 10 centimeters or less (Smiley, 2016). High frequency and Near-Field Communication (NFC) with a frequency of 13.56MHz read a distance of about 1 centimeter to 1 meter. Ultra High Frequency (UHF) with frequencies of between 865 and 960MHz can achieve over 30 meters of reading range depending on the size of the tag (Smiley, 2016).

Passive tags are used in applications such as access control, race timing, file tracking and supply chain management. (Smiley, 2016)

Active RFID Tags

Active tags are battery powered tags. Active tags provide a much larger read range of up to 100 meters (Smiley, 2016). They are used to check the real-time location of assets and are used in high-speed environments such as road tolling. With active transponder tags, the reader sends a signal to the tag and the tag sends a signal back with relevant information (Smiley, 2016). With a beacon tag, however, the tag sends a signal every 3 or 5 seconds to find a reader in range (Smiley, 2016). Active transponder tags conserve battery life when the tag is out of the reading range. In some tolling systems, as the car approaches the lane, the passive reader in the toll booth emits a signal to wake up any sleeping active transponder tag in the reading zone (Roberti, 2013). The active transponder tag is usually fixed on the windscreen (Roberti, 2013). Other systems use infrared sensors to detect the presence of a vehicle (Rakhi, Anand, Akshay, & Rohan, 2014). The reader receives the signal from the tag and the information is then routed through a local area network to a computer (Roberti, 2013). The long read range of active tags makes them useful in industries where asset location is important. (Rakhi, Anand, Akshay, & Rohan, 2014)

1.4.3 Choice of Technology

In implementing this system, RFID will be used; specifically, passive RFID tags and a reader. These are inexpensive compared to the other technologies mentioned. The tags cost as low as 11 cents. Also, one major advantage of RFID systems over other technologies for tracking is its size. The tags can be as small and light as stickers and cards as seen in Figure 1.1 below unlike GPS devices which are usually bulky thus cannot be used to track small devices. Again, passive RFID tags are powered by the reader thus in developing countries like Ghana where there are frequent power outages, it will be very useful.

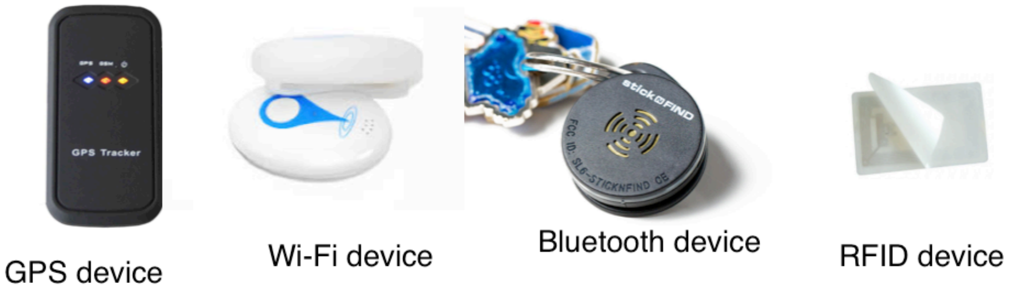


Figure 1.1: Examples of GPS, Wi-Fi, Bluetooth and RFID devices

1.5 Summary of Introduction

This chapter discusses the issue of property crime and the difficulty in monitoring and locating items on university campuses. It proposes to create a small inconspicuous device tracking system. It also discusses research of existing products and key technologies used in making tracking devices. It reveals the reason for choosing Radio Frequency Identification as the technology to use for the proposed system. The next chapter focuses on the high-level structure of the system and the requirements that need to be met.

CHAPTER 2: REQUIREMENT SPECIFICATION

2.1 INTRODUCTION

The proposed system has two main components: the hardware component which identifies and locates the device being tracked and the software component which notifies the user about the location of his/her item.

2.2 OVERALL DESCRIPTION

This system is a low-cost and inconspicuous device which uses RFID technology. The RFID tag, in the form of a sticker, consists of a microcontroller attached to an antenna. The reader which is located at vantage points consists of a scanner with antennas to transmit and receive signals. The antenna in the reader transmits electromagnetic waves which are received by the antenna in the tag. The tag then resends the information received along with information about the device it is attached to, from its memory to the reader. The signal is received by the reader and transmitted to the server for further processing. The server receives this information and identifies the details of the device such as the user and the location of the device. A notification is then sent to the user about the new location of the device. The overall description of the system is seen in Figure 2.1 below.

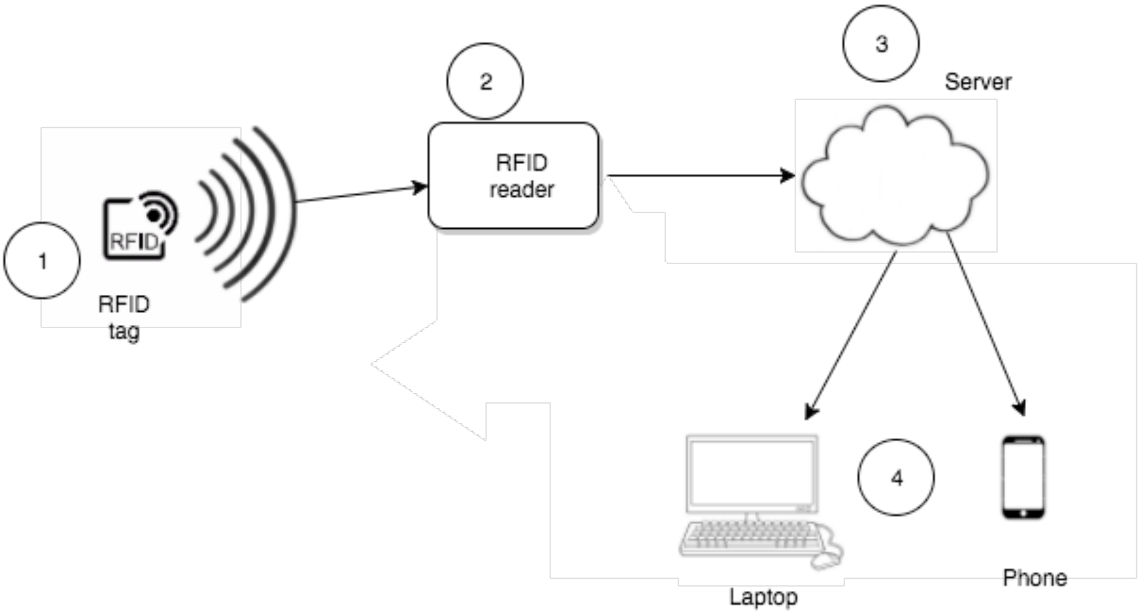


Figure 2.1: Overview of system

2.2.1 Hardware component

The hardware component of the system consists of a microcontroller and other electronic devices used to track the object. It identifies and locates the position of the device being tracked.

2.2.2 Software component

The software component consists of a mobile application which allows users to manage devices to be tracked. This application also allows the user to visualize the location of their devices at every point in time. It, again, notifies users about the location of their devices.

2.2.3 Key features

The key features of the system include:

- Reader and tag communication
- Transmission of information to server for user identification
- Sending notification to user
- Adding a new device
- Visualizing location of devices
- Editing of device information
- Alerting security
- Start tracking
- Stop tracking
- Configuring the reader
- Viewing report

2.2.4 User classes and characteristics

The main users of this system are students and university authorities. Students will use this system mainly to locate the position of their items. The part of the system which students will interface with is the mobile application. This will require no learning curve. Students also need to stick the tracking device which is a sticker to their devices in order for their them to be tracked.

School authorities will use this system to locate devices and equipment that are available to the public for use. It can also help with record keeping and inventory management. Readers will be positioned at the entrance of vantage points in the institution. As stated above, in order to locate the items, the mobile application and the tracking device must be used.

2.2.5 Operating environment

The system will rely on libraries that Arduino, an open source platform for building electronic components, offers to develop the hardware part of this system. The hardware will be connected to the server through the internet. The cross-platform mobile application will be built using HTML, JavaScript, and CSS. The server side of the system will be implemented using PHP. The system will save and retrieve data from a database.

2.2.6 Design and implementation constraints

One constraint of this system is that the reading range of the reader affects the distance within which the tag can be read. Low-frequency passive readers, for instance, have a reading range of up to 10cm which implies that the tag has to be really close to the reader. Active readers, however, have a reading range of up to 100 meters. This system will use low-frequency readers as a proof of concept.

2.3 FUNCTIONAL REQUIREMENTS

This section provides details about the functional requirements of this system.

2.3.1 Reader and tag communication

The reader will be triggered to scan a device when it is within reading range.

Stimulus/response sequences

Step 1: When a tag is within reading range of the reader, the reader transmits electromagnetic waves.

Step 2: The tag receives the signal based on proximity.

Step 3: The tag sends information to the reader

User requirements

- Tag attached to device: The user must attach a tag to the device.

System requirements

- High speed: The system should be able to scan devices that are being moved quickly.

Input/output

- The input is a valid tag.
- The output from this is the signal sent to the tag from the reader.

2.3.2 Transmission to server and user identification

The information received from the tag is sent to the server. The user details, the device details and the location of the device are identified.

Stimulus/response sequences

Step 1: After tag signal is sent to the reader, the information is sent to the server and stored in the database.

Step 2: Retrieve information about device identification number.

Step 3: Search for details of the owner of the device and location of the device.

System requirements

- Secure database system: The system must ensure that information is securely stored and cannot be hacked.
- Search algorithm: The algorithm must efficiently search the database for the details of the owner of the device as well as the device type and name.

Input/output

- The input here is the information sent by the reader to the server.
- The output includes information about the details of the owner of the device and location of the device.

2.3.3. Send notification to user

This feature allows the user to receive notification when their device crosses some boundaries. This is activated when the user activates the start tracking feature.

Stimulus/response sequences

Step 1: Device is read by the reader and determined if it is outside its boundary.

Step 2: Tracking feature is activated.

Step 3: Identify the owner of the device and send a notification is sent to their phone.

User requirements

- A valid phone number: Offline users will be contacted by SMS and email.

System requirement

- SMS and Email notification capabilities: The server must be able to compose and send notifications to users.

Input/output

- Information about the owner and the device are the inputs.
- The notification the user sees on his/ her phone is the output.

2.3.4. Add a new device

This feature allows the user add new devices to be tracked. After a successful login, the user can add a new device to be tracked.

Stimulus/response sequences

Step 1: To track a device a tag must be attached to the device.

Step 2: Enter the details of the device and tag, and add to the database.

User requirements

- Attach tag: Users must attach a tag to the device to be tracked.

Input/output

- The device name, description, image and tag identification the inputs.
- A confirmation message is the output.

2.3.5. Visualize location of devices

This feature allows the user to view the current position of their added devices on a map. After login, the user can visualize the locations on a map.

Stimulus/response sequences

Step 1: Add device and activate start tracking feature.

Step 2: Device is read by the reader and it is determined if it is outside its boundary.

Step 3: Fetch device details and current location from the database.

Step 4: Display this information on a map.

2.3.6. Edit device information

This feature allows the user to make changes to details about the device. After a successful login, the user can edit an already added device.

Stimulus/response sequences

Step 1: Add devices to the database.

Step 2: View all devices.

Step 3: Select edit button on the device to be edited.

Step 4: Enter the changes and add to the database.

Input/output

- The device name, description, image and tag identification the inputs.
- A confirmation message is the output.

2.3.7. Alert security

This feature allows the user to alert the security in the organization when the device changes location. This is done by clicking on the alert security button. When the device is stolen, the security can be alerted to find the device with information about the device and its location sent to them.

Stimulus/response sequences

Step 1: View all devices.

Step 2: Select alert button on the device.

Step 2: Confirm for the alert to be sent.

Output

- A confirmation message informing users that an alert has been sent is the output.

2.3.8. Start tracking

This feature allows the user to start tracking a particular device. When the start tracking feature is activated, notifications will be sent to the user when the device crosses some boundary.

Stimulus/response sequences

Step 1: Add devices to the system.

Step 2: Select device to be tracked and choose the current location to allow the system identify a change in location.

Output

- The output is a confirmation message stating that the device is being tracked.

2.3.9. Stop tracking

This feature allows the user to stop tracking a particular device. The start tracking button allows notifications to be received when the device changes location. The stop tracking button, on the other hand, stops the notifications from being sent.

Stimulus/response sequences

Step 1: Start tracking device.

Step 2: Select device being tracked that no longer needs to be tracked.

Output

- The output is a confirmation message stating that the device is no longer being tracked.

2.3.10. Configure reader

This feature allows the organization to configure readers when they are purchased. Details about the reader and its location are saved.

Stimulus/response sequences

Step 1: Enter location details of the reader and add to the database.

Input/output

- The name, latitude, longitude and type of location are the inputs.
- A confirmation message is the output.

2.3.11. View report

This feature can be accessed only by the organization or person that adds the readers. It gives them a statistical analysis of items being tracked by users, their locations, and the alerts sent to security based on time and location. This information is useful for drawing conclusions about users and the patterns of thieves.

Stimulus/response sequences

Step 1: Add readers which identify when devices change location.

Step 2: Fetch user and location details from the database.

Step 2: Display charts and tables.

2.4 NON-FUNCTIONAL REQUIREMENTS

1. Performance requirement

   The performance of the system is key to ensure the successful tracking of devices. The readers should be able to scan tags all the time that is, 24 hours every day. Also, users should receive notifications within 30 seconds after the device changes location.

2. Security Requirement

   The reader should be secure such that the information stored in it cannot be accessed by anyone except through the phone application. Also, notifications must be sent to the right user with the right information.

2.5 EXTERNAL INTERFACE REQUIREMENTS

2.5.1 User Interfaces

A mobile application will be built to allow users have easy access to the application. The admin side application will, however, be a web application since it will only be used when a new reader needs to be configured and to provide statistical analysis on the system to the organization. The mockup user interface of the system which shows all the devices being tracked is illustrated in Figure 2.2 below. A map showing the most recent location of a selected device is seen in Figure 2.3.

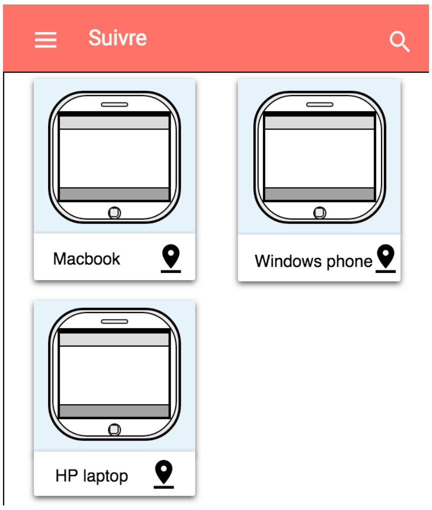


Figure 2.2: List of devices being tracked

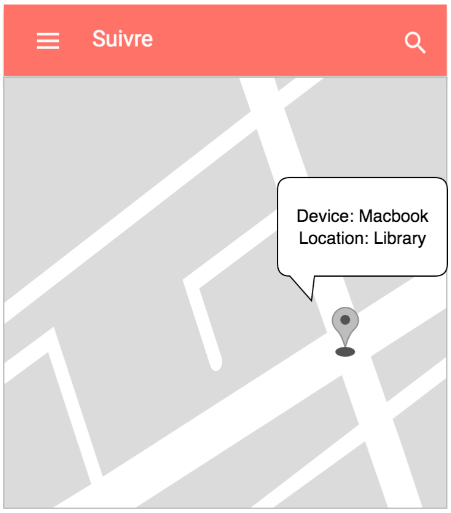


Figure 2.3: A map showing the location of the selected device.

2.5.2 Hardware Interfaces

The reader will communicate with the tag and the server. The reading range and strength of the reader determines how efficient the system will be. Also, transmission of data to the database will depend on bandwidth. There will also be communication between the user's phone and the server. The exchange of data between the client-side device which is the user's phone and the server is necessary for this system to function well. In this communication, TCP/IP protocol will be used. The supported client-side devices are Android, iPhones and Windows phones.

2.6 Summary of Requirement Specification

This chapter discussed the requirements of the system. It contained the high-level overview of the system. The next chapter focuses on the design and architecture of the system.

CHAPTER 3: DESIGN AND ARCHITECTURE

3.1 INTRODUCTION

Figure 3.1 below shows a high-level overview of the system. This activity diagram shows the different activities and the flow from one activity to the other.

After a successful login, users can view all the devices that have been added to the database. Users can then add a new device, visualize the location of the devices on a map, edit device information or start tracking a device. If the tag attached to the device is scanned by the reader after the start tracking button is clicked, it implies the device has crossed some boundary. This reader is built on an Arduino microcontroller as seen in Figure 3.2. Arduino is an open-source platform used for building electronic projects. Information about the tag is retrieved from the reader and transferred to the database.

Afterward, the user and product details are retrieved, and a notification is sent to the user's mobile device via email, SMS or both based on their preference. This notification contains information about the current location of the device. When the user of a device realizes that the device has been stolen, the security of the institution can be notified. Activating the Alert Security button sends a notification to the security with details about the missing item such as its location and a description.

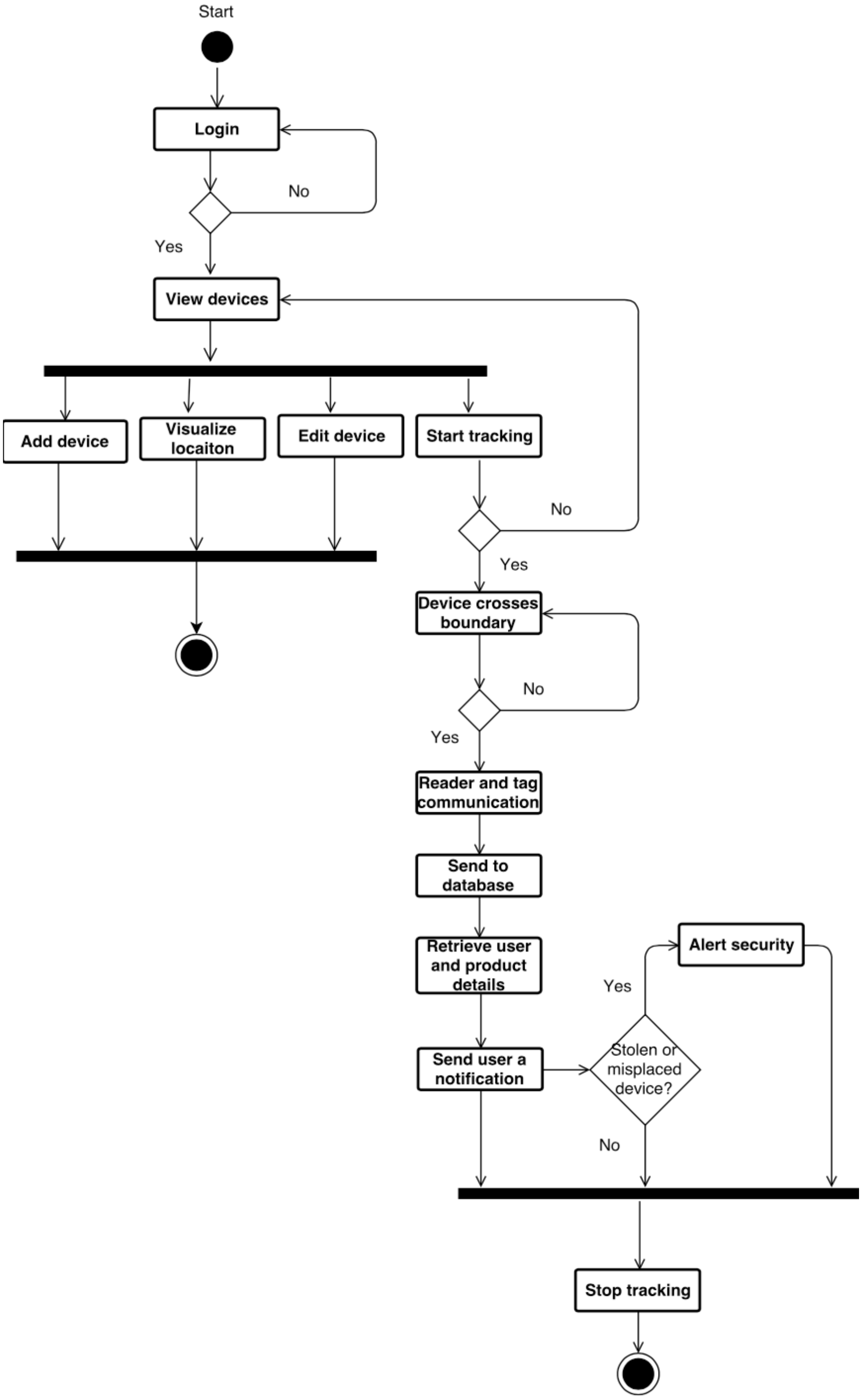


Figure 3.1: Activity diagram showing flow of activities

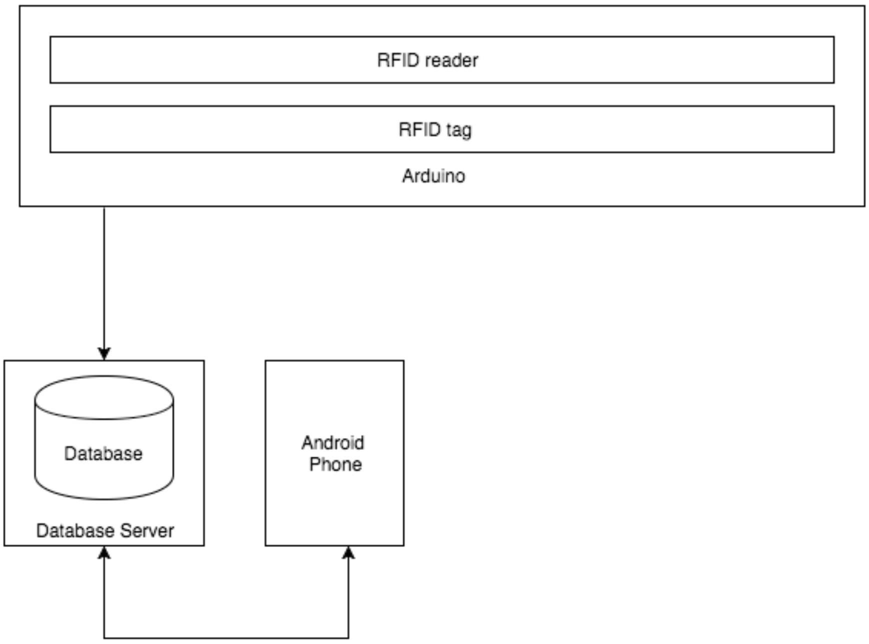


Figure 3.2: Architecture of system

3.2 SYSTEM ARCHITECTURE

This system uses the client-server architecture. According to Sommerville, with the client-server architecture, the user interacts with a program running on the local computer which then communicates with a program running on a remote computer (Sommerville, 2011). The server provides the client with services which the end user can view via the client.

Likewise, with this system, the user interacts with the mobile application to perform functions such as adding a device, visualization the locations of devices, etc. The mobile application, in this case, the client, communicates with the server on a remote computer for services. For instance, in order to visualize the location of devices on a map, the devices and their locations are fetched from the server. This is then displayed on a map by the client device for the end-user.

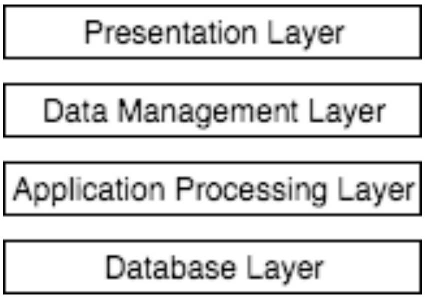


Figure 3.3: Layered architectural model for client-server application

Figure 3.3 illustrates the layered architectural model for this client-server system. Layered architecture is used to achieve separation and independence in order to allow changes to be centralized (Sommerville, 2011). With the layered architecture, the system functionality is divided into layers and each layer relies on a lower layer (Sommerville, 2011).

The presentation layer represents the part of the system that the user interacts with. It is concerned with managing the user's interaction with the system and the presentation of results to the users. This part is implemented using HTML and CSS.

The data management layer is concerned with managing the data that is sent to and from the client. This part of the system is implemented using AJAX. AJAX which stands for Asynchronous JavaScript and XML is used to send data entered by the user to the server. It also receives data from the server. The added feature of AJAX is that updating a page does not require reloading of the whole page.

The application processing layer is concerned with implementing the logic of the system. This is implemented in PHP. It contains the functionality of the application and communicates with the database. The database layer represents the actual database which stores data.

3.3 DESIGN SPECIFICATION

3.3.1 Use Case view

The use case diagram in Figure 3.4 shows a graphical overview of the actors involved in the device tracking system, different functions undertaken by the actors and how these different functions interact.

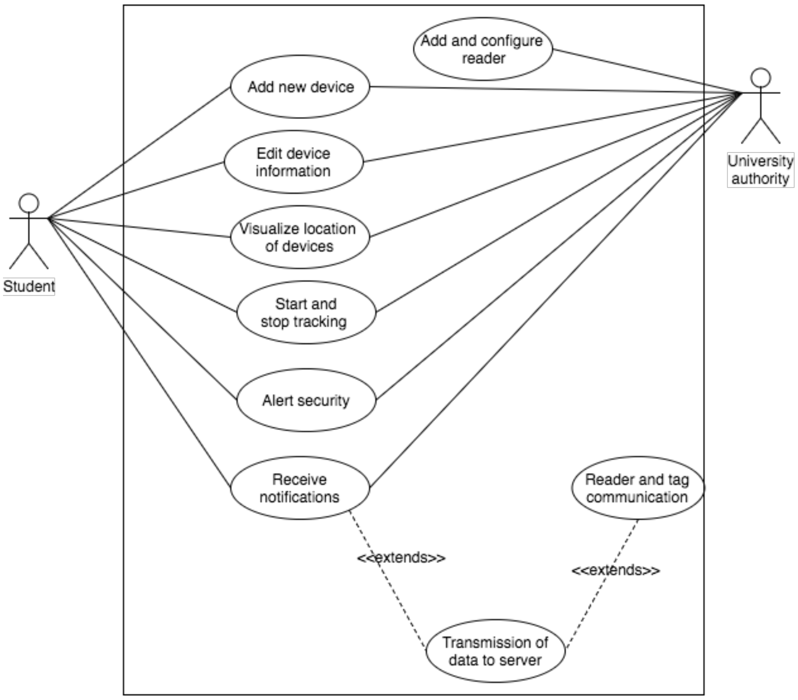


Figure 3.4: Use case diagram for device tracking system

3.3.2 Database Architecture

This section discusses the data model that will be used to store data. The Entity Relation Diagram in Figure 3.5 is used to describe the elements in the data model.

The database contains a user, device, device-location, location, admin and security-alerts tables. The user table contains details about the user useful for login, registration and identifying devices of a particular user. The device table has details about all devices and the tags attached to them. There is a one-to-many relationship between the user and device table. Thus, a user can have many devices.

The device-location table contains information about devices and their locations. This table is updated whenever a device being tracked crosses a boundary with a reader. There is a one-to-many relationship between the device and device-location table. There is also a location table which contains information about the locations which have readers. There is a one-to-many relationship between the device and location table.

The admin table has information about the organization that configures the readers. There is a one-to-many relationship between location table which contains the locations of the readers and this table. There is also a one-to-many relationship between the admin table and the user table. The security-alerts table contains information about alerts sent to the security of the organization when the Alert Security button is triggered. There is a one-to-many relationship between the device and security-alerts table. There also is a one-to-many relationship between the security-alerts and location table, and a one-to-many relationship between the security-alerts and user table.

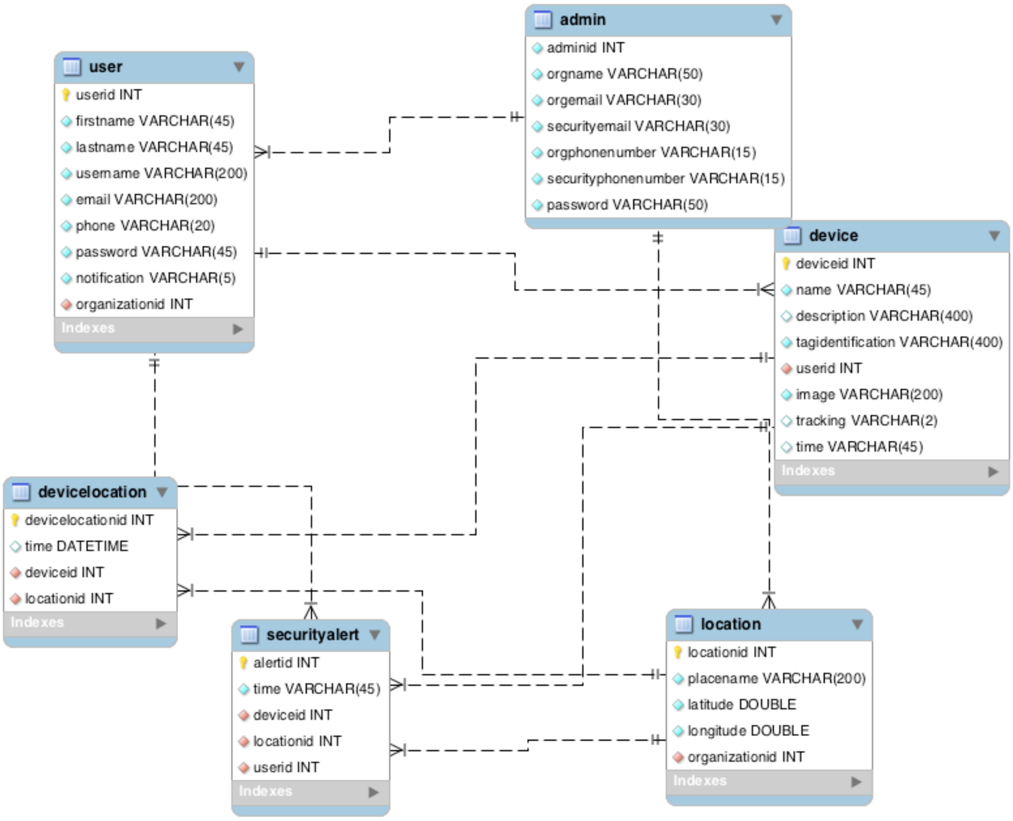


Figure 3.5: ER diagram of database

3.3.3 Logical View

This subsection describes the functionality of the system. Sequence diagrams are used here to model the interactions between the actors and objects in the system. As illustrated in Figure 3.6 below, a notification is sent to the user when a device being tracked is scanned by a reader. When a user realizes that the device being tracked has been stolen, the alert security button when clicked, sends details about the device to the security of the organization to notify them. This is shown in Figure 3.7.

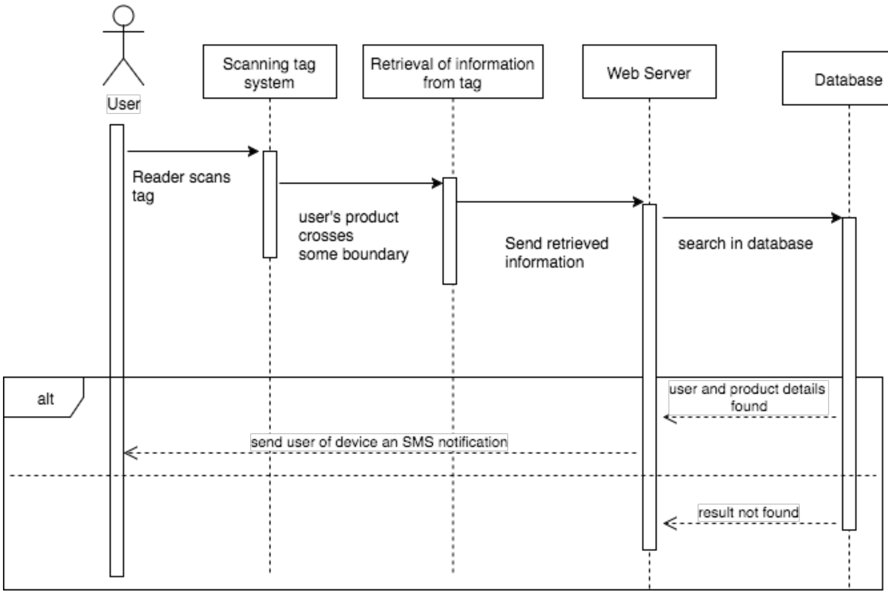

User
Scanning tag system
Retrieval of information from tag
Web Server
Database
Reader scans tag
user's product crosses some boundary
Send retrieved information
search in database
alt
user and product details found
send user of device an SMS notification
result not found

Figure 3.6: Sequence diagram for the send notification requirement

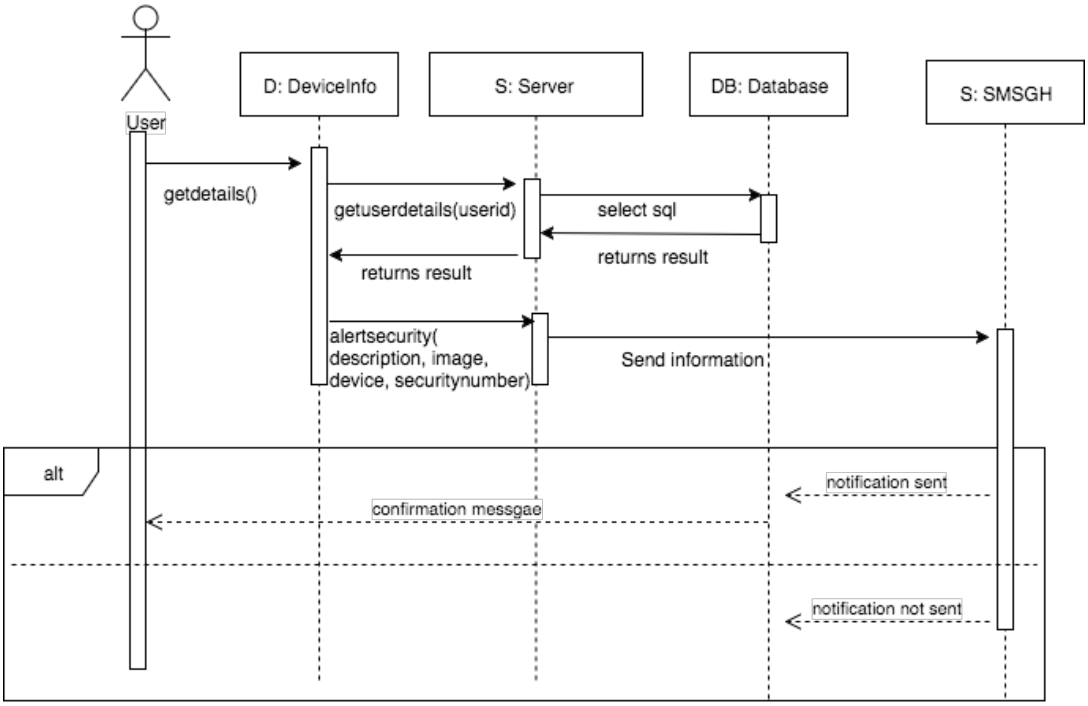


Figure 3.7: Sequence diagram for the alert security requirement

3.4 Summary of Design and Architecture

This chapter discussed the system architecture and design specification. The logical, database and use case views were illustrated. The next chapter focuses on the implementation details.

CHAPTER 4: IMPLEMENTATION

This chapter focuses on the implementation of the system. The first section discusses the details of the implementation and the technologies used. The diagram in Figure 4.1 shows the typical path of a message and the technologies used at each stage. This is further described in this section. The shaded sections represent the parts that I implemented.

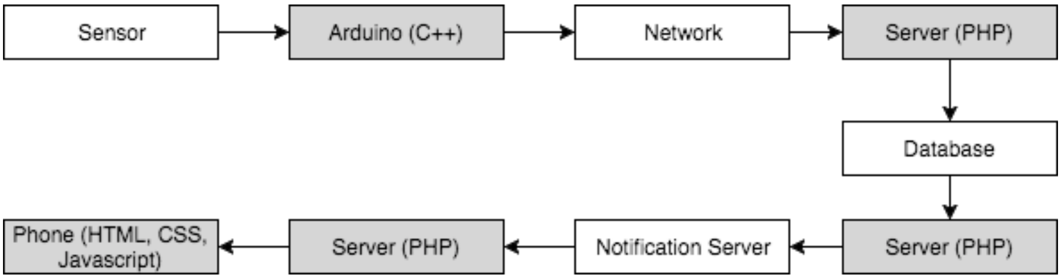


Figure 4.1: Typical path of message

4.1 HARDWARE IMPLEMENTATION

This section discusses the implementation of the hardware component of this system.

4.1.1 Identifying tracked objects

This covers the first three nodes in Figure 4.1 above. In order to track any added device, the Start Tracking button on the mobile application is activated. The user then selects the device to be tracked and the current location of the device. Doing this allows notifications to be sent to the user when the device crosses some boundaries. When the device is moved, its new location is identified by an RFID reader. RFID uses electromagnetic waves to identify objects attached to tags. The RFID readers are located on top of doorposts at vantage points around the university or organization. An MFRCR522 RFID reader is being used for this system as a proof of concept. This is a low-frequency reader with a reading range of 0 to 35mm and frequency of 13.56MHz. There are, however, readers with reading range of over 100 meters.

In order to make the reader work, it is connected to an Arduino board. Arduino is an open-source platform used for building electronic projects. It is used for building digital devices that can sense and control physical devices. It consists of a microcontroller and an Integrated Development Environment (IDE) on a computer. It uses a simplified version of C++.

In this project, an Arduino Uno board is used. This is a microcontroller board which uses ATmega328 micro-controller. It has digital inputs/output pin, analog inputs, a crystal oscillator, USB connection and a power jack. The digital pins are used to take inputs and return output. The analog pins, on the other hand, accept inputs with varying values. Crystal oscillators provide clock signal which controls the timing of the circuit (Arduino.cc, 2017). The USB connection is used to connect the Arduino board to a computer in order to transfer programs onto the board and power the board. The board can also be powered with an AC-to-DC adapter. (Arduino.cc, 2017)

Connecting the reader to the Arduino board is just one step in getting the system to work. After receiving the data from the reader, the data has to be sent to the database via a network. Global System for Mobile Communication (GSM), Ethernet and Wi-Fi can be used to connect to the database. GSM is used because it can be used in areas without Wi-Fi and unlike the Ethernet module, there is no need for a cable. It is, however, slower than the other wireless connections. This system uses the GSM module SIM900. It requires an APN which is obtained from the network provider. The set-up is seen in Figure 4.2 below. Arduino programming which is explained in section 4.2 is, however, required for the hardware to work.

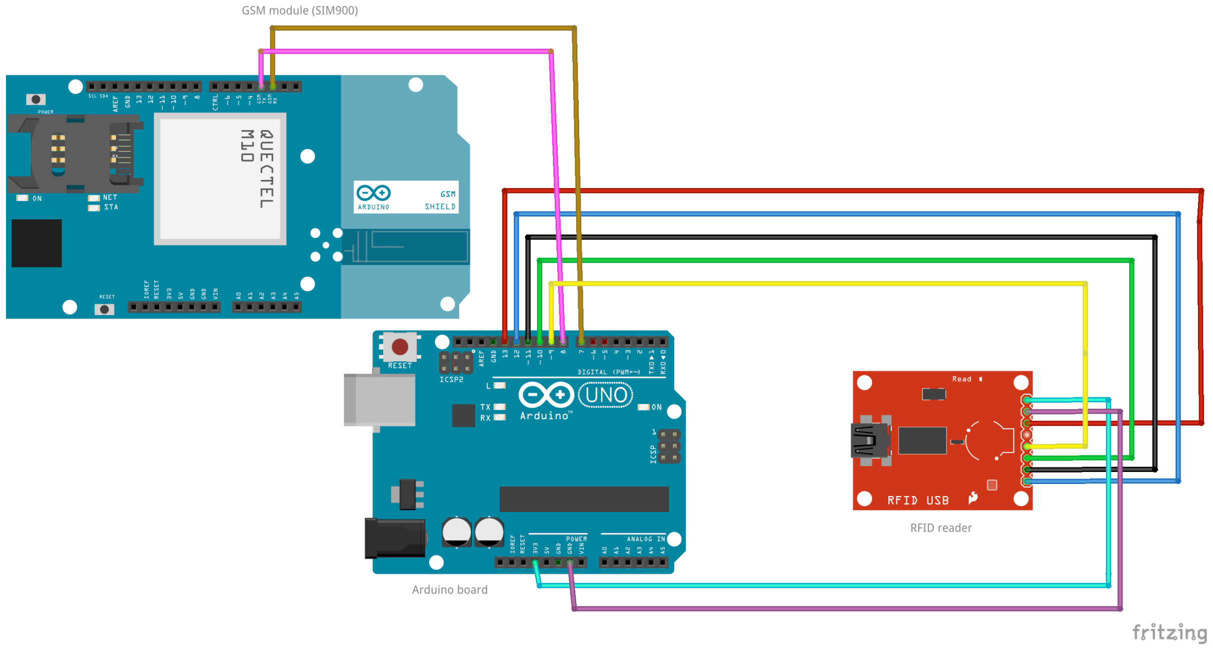


Figure 4.2: Arduino Uno Board, RFID reader, and GSM module connection.

4.2 ARDUINO SOFTWARE IMPLEMENTATION

This section discusses the instructions downloaded onto the Arduino board.

4.2.1 Identifying tags and connecting to the internet

The first node in the path as seen in Figure 4.1 is the sensor. The data from the sensor, in this case, the reader, will be transferred onto the Arduino board. To successfully achieve this, instructions must be uploaded onto the Arduino board. This program identifies a tag within reading range, scans it and records the tag information. The MFRC522 library which contains software packages and programs for the reader, is used. To successfully read, the tag must first be initialized by calling the mfrc522.PCD_Init() method. The tag id is saved in mfrc522.uid and can be retrieved by looping through it and printing the bytes. Listing 4.1 shows a code snippet of how the reader reads a tag.

```
// Init MFRC522 tag
mfrc522.PCD_Init();
//looping through mfrc522.uid
for (byte i = 0; i < mfrc522.uid.size; i++) {
        //printing out the card id which is in mfrc522.uid
      Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
      Serial.print(mfrc522.uid.uidByte[i], HEX);
}
```

Listing 4.1: Code for reading a tag

After the reader scans the tag, the data needs to be transferred to the database. This is done by connecting and passing the sensor value to a PHP script which has a function to add the data to the database. To connect to the network, the Arduino board is connected to a GSM module. The GSM library contains software packages for the GSM module.

Below is the section of the code which connects to the PHP page which is in this case, ajaxPage.php. The function on this PHP page responsible for adding to the database is identified as cmd=12 as seen in the code snippet below. A URL is constructed as an array of characters with all parameters to be passed to the PHP backend included as GET requests. A connection is established to the server on port 80 which is for HTTP, thus an HTTP request is sent. Data and format expected by the HTTP sender is sent by calling the client.print() method. The client.print() function sends requests to the specified server. This is illustrated in Listing 4.2.

```
char server[]= "suivre.000webhostapp.com"; //server name
if (client.connect(server, 80)) {
    // Make a HTTP request:
    client.print("GET ");
    client.print("/ajaxPage.php?cmd=12&tagid=4&locationid=2");
    client.println(" HTTP/1.1");
    client.print("Host: ");
    client.println(server);
    client.println("Connection: close");
    client.println();
  }
```

Listing 4.2: Code for connecting to the PHP page

4.2.2 Data Transfer

The fifth node in the message path is the database. The Arduino board as explained makes an HTTP request to the PHP script via a GSM module in order to add to the database. As stated above, this script has a function which performs the functionality of adding the location and tag identification to the database. This function also connects to the notification server in order for notifications to be sent. This is explained in detail in Section 4.3.1.

4.3 MOBILE APPLICATION

The cross-platform mobile application was built using PhoneGap. PhoneGap is a software developed by Adobe System which is used to build cross-platform mobile applications. This will be used to build the Android, iPhone and Windows versions of the mobile application. Apps developed using PhoneGap use web-development languages such HTML, CSS and JavaScript (Tutorialspoint.com, 2015). In order to successfully build an app using PhoneGap, the developer needs two things; a configuration file and the web content built using the web technologies mentioned above. The configuration file called config.xml is used to configure the necessary settings when building the app. (Tutorialspoint.com, 2015)

PhoneGap accepts a zip file of all resources via a link to a repository on Github. Github is a repository where developers can upload and share their work with others. For this system, the files were pushed to Github and this was called directly to PhoneGap. The application was then built and downloaded onto a phone for testing.

The mobile application for this system was implemented using the NativeDroid framework. NativeDroid inspired by materialize design was deveoped using jQuery mobile. It helps in creating beautiful and faster applications.

In order for the application to communicate with the database, the client-side is connected to the server using AJAX. The AJAX call to perform this function is $.ajax(). This function connects to the specified page which in this case is the PHP script. The PHP script has a number of functions defined to perform different tasks. The parameters passed when the script is called determines the function to be called.

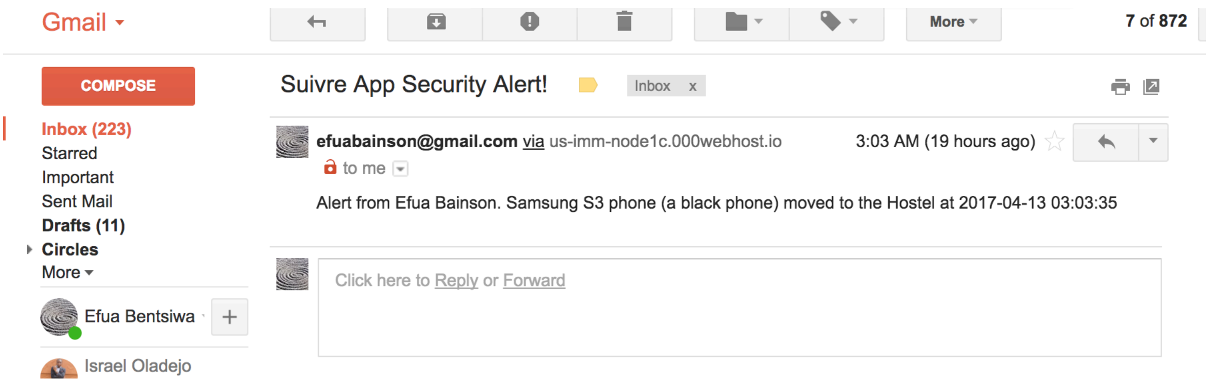
After successfully communicating with the database, JSON is used to pass the returned values from the server to the client. The response is interpreted by JavaScript and feedback is sent to the user.

4.3.1 Notification

The seventh node in the path, as shown above, is the connection to the notification server. For this system, users are allowed to choose between being sent notifications via SMS or email, or both. The mail() PHP function is used to send emails. It takes the recipient's email, subject of mail, message and sender email as parameters. Below in Listing 4.3 is a code snippet showing how the mail() function is called. The result is seen in Figure 4.3 below.

```
mail(‘efuabainson@gmail.com’,’Suivre App Security Alert!’ ,’ Alert
from Efua Bainson. Samsung S3 phone (a black phone) moved to the
Hostel at 2017-04-13 03:03:35’,’efuabainson@gmail.com’);
```

Listing 4.3: Sending email using the mail() function



Suivre App Security Alert!

efuabainson@gmail.com via us-imm-node1c.000webhost.io 3:03 AM (19 hours ago)

to me

Alert from Efua Bainson. Samsung S3 phone (a black phone) moved to the Hostel at 2017-04-13 03:03:35

Figure 4.3: Email successfully sent

The SMSGH API is used to send SMSs to phone numbers from an application. This will be used to send SMSs to phones to alert users about the location of their devices. This requires having an account with SMSGH and using their API. With SMS technology, even in the absence of internet or a bad network, the user will still receive notifications. This is necessary considering the power outages in developing countries which can affect the network.

To send an SMS, an instance of the Message class is created and the message to be sent, is passed to the setContent() method. The setTo() function specifies the recipient while the setFrom() function specifies the sender. The sendMessage() method then sends the message. Listing 4.4 illustrates how to send an SMS. Figure 4.4 shows the message that was successfully sent to a user.

```
$mesg = new Message();//instance of Message class created
//pass message to be sent
$mesg->setContent(‘Samsung phone new moved to the Library at
2017-03-14 06:37:27’);
```

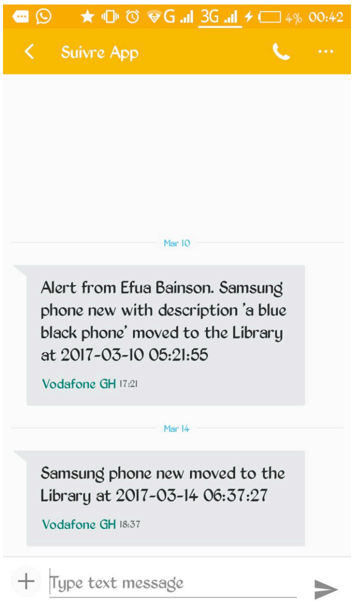
```
$mesg->setTo(“+233573283028”); //recepient

$mesg->setFrom("Suivre App"); //sender

//sends message

$messageResponse = $messagingApi->sendMessage($mesg);
```

Listing 4.4: Code for sending SMS



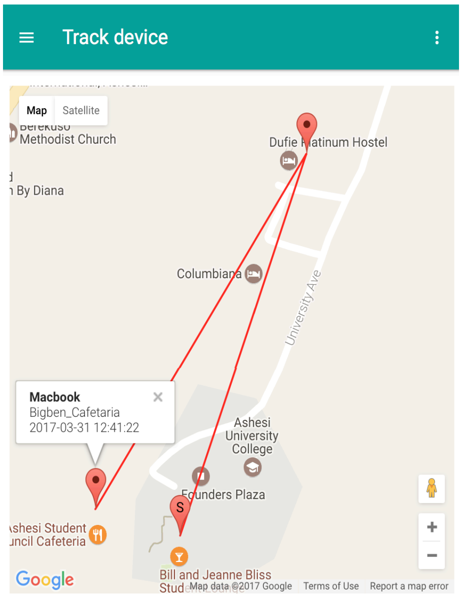
Alert from Efua Bainson. Samsung phone new with description 'a blue black phone' moved to the Library at 2017-03-10 05:21:55

Samsung phone new moved to the Library at 2017-03-14 06:37:27

Figure 4.4: SMS successfully sent to a user

4.3.2 Visualizing device location

Google Maps API is used to provide the user with visual information about where all devices are located at every point in time. This is a mapping service developed to offer satellite imagery, street maps, route planning, etc. The devices and their locations are fetched from the database and displayed on a map. An API key needs to be generated to authenticate the application. This is generated on the Google API Console and added to the link which loads the API. Figure 4.5 below shows the current location of a MacBook which was added by a particular user and the route it took after the Start Tracking button was activated. After the Stop button is activated, the new route of the device is recorded when the Start Tracking button is activated again.



Macbook
Bigben_Cafetaria
2017-03-31 12:41:22

Figure 4.5: Map showing all devices and their locations

To display the locations of the devices on a map, data is requested from the server. The returned results which consist of the latitude and longitude of each location are saved as XML. Each point is then positioned on the map.

4.3.3 Security alert

The security of the organization can be alerted when the owner of the device realizes that the device has been stolen. At the click of the Alert Button by the user, a notification will be sent to the security office’s phone and email address with information about the device such as a description, and its current location. The implementation is similar to that described above for sending notifications to the user.

In order to prevent database hacking, Advanced Encryption Standard (AES) will be used. AES is an encryption technique which is fast and allows various key lengths (Stallings, W, 2002). MySQL has functions that implement encryption and decryption using the AES algorithm. The aes_encrypt() function implements AES with a 128-bit key length. The value to be encrypted is encrypted using a key and returns a binary string as the output. The aes_decrypt() function decrypts the encrypted value using the key and returns the original value.

In the event of a hack, the data is encrypted and thus is not useful. For instance, before the location of devices is added to the database it is encrypted by calling the aes_encrypt function as seen in Listing 4.5 below. To make it difficult to hack, the value, in this case, the location id is concatenated with the current time and encrypted using the device id as the key.

```
insert into devicelocation set deviceid='$deviceid, time='$time,
locationid=aes_encrypt(concat('$locationid, '$time'),'$deviceid');
```

Listing 4.5: AES encryption code for inserting a new location into the database

In order to view the location of the added devices on a map, the data stored in the database is decrypted as seen in the code snippet below. After decrypting, however, the time is replaced with an empty string to get the original value.

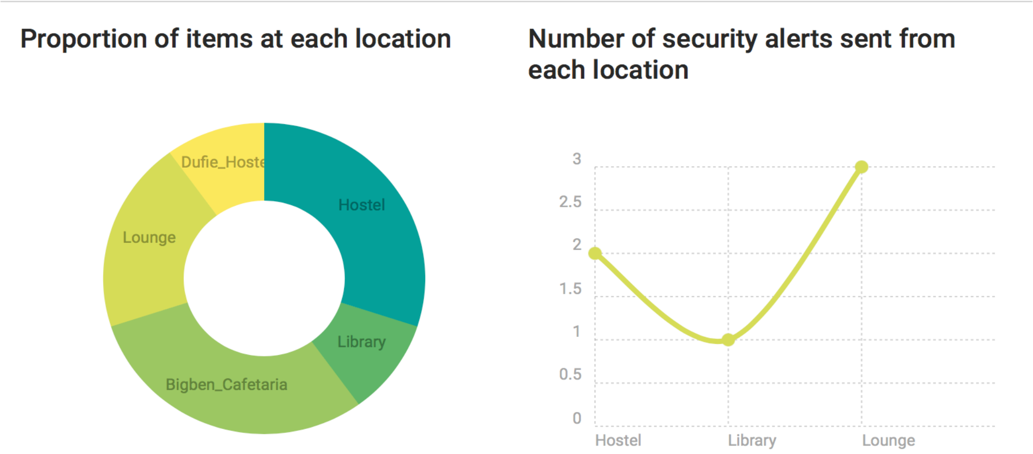
```
select deviceid, time, replace(cast(aes_encrypt('$locationid’,
'$deviceid') as char(100)), ‘$time',''')from devicelocation;
```

Listing 4.6: AES decryption code for fetching locations from the database

4.3.4 View report

This feature allows statistical analysis to be conducted on data collected from the users of a particular organization as seen in Figure 4.6 below. This feature is available only to the administrators, that is, the organization that was registered. The data collected can be analyzed to provide new knowledge about the behavioral pattern of users and even thieves. Chart.js is used to create the charts after the data is fetched from the database.

The first chart in Figure 4.6 informs the administrator about locations where users usually leave their valuable items. This will help them to know where to tighten security. This is especially true if the second chart also reveals that a high number of alerts are sent to the security about missing items in a particular location. This can also reveal the behavioral pattern of thieves such as locations where most items get missing and what time of the day this happens.


Proportion of items at each location

Number of security alerts sent from each location

Figure 4.6: Report using charts and tables

4.4 Summary of implementation

This chapter revealed the details of the implementation and the technologies used. The next chapter discusses the procedures used to test the application and the results.

CHAPTER 5: TESTING AND RESULTS

5.1 Introduction

In this section, the different parts of the system were tested to access its performance. The testing procedures used are described below.

5.2 Testing Overview

This system comprises of different components; thus each component has to be tested. The client side of the application was tested using unit testing. Unit testing involves verifying that each component meets its specification. The MySQL queries were also tested using unit testing. The device identification process was tested by scanning the reader with different tags. It was tested based on the following criteria

- Ability to recognize and read tags accurately
- Ability to send tag information to the database
- Length of time it takes to identify tags and transfer data to the database

Component and user testing are other testing procedures used in accessing this system.

5.2.1 Unit Testing

PHP unit tests were written to test the implemented functions and classes. To set this up, the phpunit.phar file which is a compressed PHP application, was downloaded into the project directory. An instance of the PHP class to be test was then created. The assertEquals() function was used to verify whether the expected result and the actual result are the same. In

Listing 5.1 below, the assertEquals() function verifies whether the response after adding the device is true. This also seen in Appendix A. It returns OK when the test works.

```
$obj=new user();
$this->assertEquals(true, $obj->addDevice("Iphone 6",'A black phone
with a blue-black case', 56, 3,
'/Applications/XAMPP/xamppfiles/htdocs/suivre/img/phone.jpg'));
```

Listing 5.1: Code for PHP unit test

The table in Table 5.1 shows the result of the unit tests conducted for adding a new device.

Table 5.1 Unit Testing for adding a new device

Unit Test point	Testing	Result
SQL statement to add a new device	Write SQL statement ("insert into device set name='$device', description ='$description', tagidentification='$tag', image='$image', userid=$id) **Test input:** device='iPhone', description ='space grey phone with blue cover, tagidentification='002', image='img/iphone.jpg', userid=2 **Expected Result:** Device should be added	1 row affected Device was added
	Test input: device='iPhone', description ='space grey phone with blue cover **Expected Result:** Device should not be added	Error message

	Test input:
	device='iPhone',
	description ='space grey phone with blue cover',
	tagidentification='002',
	image='img/iphone.jpg', userid=k
	
	Expected Result: Device should not be added
		Error message

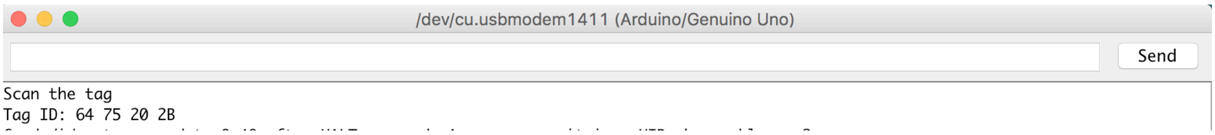
	Test input: device=, description ='space grey phone with blue cover', tagidentification=, image='img/iphone.jpg',userid=2 **Expected Result:** Device should not be added	Error message
addDevice method in the server side (addDevice())	PHP unit test to test the code **Input:** "ajaxPage.php?cmd=3&device='Hp laptop'&description='Purple laptop with stickers'&tag='0045'&image='image/latptop.jpg'" **Output:** OK (1 test, 1 assertion) **Assert Condition:** asserttrue()	Test successful

5.2.2 Component Testing

With component testing, coherent entities are grouped together and tested. For this system, the three components tested were the device identification process, the data transfer and notification process, and the adding and visualizing of devices process.

5.2.2.1 Device identification process

Two different tags were used for this testing procedure. Each tag was scanned once every 2 seconds and this was done 10 times. The first tag was identified 10 out of 10 tries. The second tag was identified 8 out of 10 tries. This implies a success rate of 90%. The limitation, however, is that the tag needs to be within 0 to 40mm from the reader in order for it to be identified. This is because a low-frequency RFID reader is being used. There are, however, readers with reading range of over 100 meters. The diagram in Figure 5.1 below illustrates the output which is the Tag ID retrieved when a device with an attached tag is successfully identified. This output is sent to the PHP script for further processing.



```
Scan the tag
Tag ID: 64 75 20 2B
```

Figure 5.1: Device identification process

5.2.2.2 Data transfer and notification process

In order to test this component, identified tags were transferred to the database and notifications were sent via email and SMS. The data was not sent when the network was bad. The transfer of data was also delayed when there was low bandwidth.

5.2.2.3 Adding and visualizing device process

This component was tested by adding a number of devices and viewing the devices being tracked on a map. Selenium which is a testing framework for web applications was used to test the mobile application since it is written in HTML, CSS and JavaScript (Seleniumhq.org, 2017). The Selenium IDE was used to record user interactions with the application which was used to generate test cases for testing the features (Seleniumhq.org, 2017). Listing 5.2 shows part of the generated Java code for testing the add device feature. The driver refers to an instance of the Firefox WebDriver. The findElement() function uses the id to locate the various form elements on the page.

```
driver = new FirefoxDriver();
driver.findElement(By.id("device")).sendKeys("Samsung S4");
driver.findElement(By.id("submit")).click();
```

Listing 5.2: Test code for adding a new device.

The other part of this component is the visualization of devices. Selenium testing was also used to test this feature. Figure 5.2 shows a list of the devices that were added in testing the add device functionality.

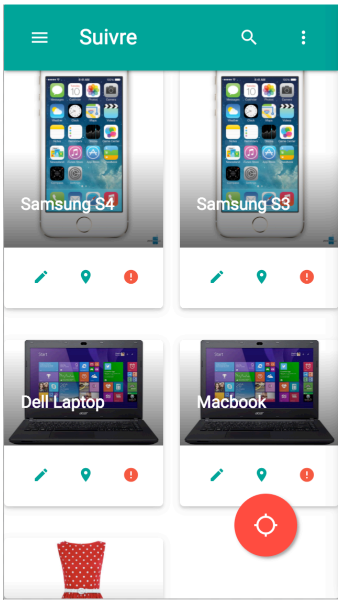


Figure 5.2: A list of added devices

The map in Figure 5.3 illustrates the three devices that were being tracked and their respective location. As seen in the image the Samsung S4 which was added was last seen at the Library.

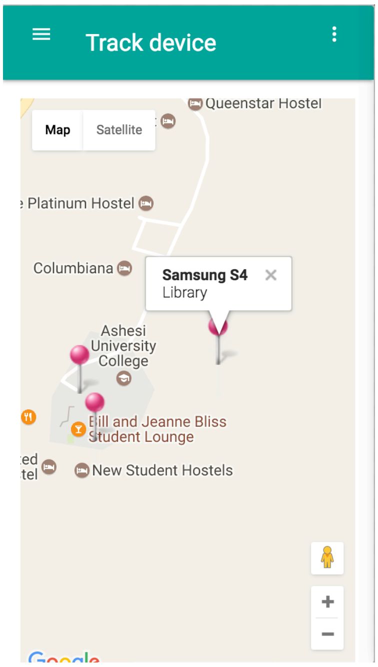


Figure 5.3: A map showing the location of the devices being tracked.

5.2.3 System testing

With this testing procedure, all components are integrated together into a complete system and are tested. This system is able to successfully identify tags attached to devices and transfer the data to the database. However, one challenge of this system is that the tag has to be within less than 40mm of the reader in order for it to be identified.

Secondly, the system is able to successfully notify users when the location of their device changes. This is done via SMS and email. This system also allows users to alert the security of the organization when a device is stolen. Moreover, users can perform functions such as adding, editing and deleting devices.

This system is also useful for data analysis. The charts and tables provided can provide new knowledge about the behavior of people and the devices they own. It will give insight into the behavioral pattern of thieves.

5.2.4 User testing

This testing procedure involves getting actual users to test the system. Six Ashesi students tested the application. The students were observed while they used the application. According to the students, the system will solve the problem of property crime on university campuses. A number of students were excited about the ability to view devices on a map and the tracking route each device takes. The communication between the reader and tag took less than a second for all tests. However, from the test it was noted that alerts to security usually take an average of 19.37 seconds which seems rather long. Also, the transfer of data from the reader to the database took about 40 seconds. Most of the students commented on the inability to identify and understand the Start tracking button. A recommendations was that, instead of alerts being sent to the security of the organization, it should be sent to a person named by the user. The table in Table 5.2 summarizes the result of the user tests conducted.

Table 5.2 Results from user testing

Requirement	Average Time	Success Rate
Device identification process	Less than a second	90%
Data transfer	40 seconds	80%
Security alert	19.37 seconds	100%

5.3 Implemented and tested requirement

This section lists all the requirements and indicates whether the requirement has been implemented and tested. This is shown in Table 5.2 below.

Table 5.2: Table of requirements.

REQUIREMENT	**DESCRIPTION**	**IMPLEMENTED OR NOT**	**TESTED**
Reader and tag communication	The reader is triggered to scan a device when it is within range.	Implemented	Passed test
Transmission of information to server for user identification	The information received from the tag is sent to the server	Implemented	Passed test
Send notification to user	Allows the user to receive notification when their device crosses some boundaries.	Implemented	Passed test
Add a new device	Allows the user to add new devices to be tracked.	Implemented	Passed test
Visualize location of devices	Allows the user to view the current position of their added devices on a map.	Implemented	Passed test
Edit device information	Users can make changes to details about the devices	Implemented	Passed test
Alert security	Allows users to alert the security in the organization	Implemented	Passed test

Start tracking	Allows the user to start tracking a particular device	Implemented	Passed test
Stop tracking	Allows the user to stop tracking a particular device.	Implemented	Passed test
Configure reader	Allows readers to be configured.	Implemented	Passed test
View report	Gives statistical analysis of data	Implemented	Passed test

5.4 Summary of testing and results

This chapter summarized the procedures used tot test the system, and the results generated. The next chapter discusses the limitations of the system and gives recommendations for future development.

CHAPTER 6: CONCLUSION AND RECOMMENDATION

6.1 Introduction

This project set out to develop an inconspicuous and affordable system that will enable users to locate their items and be alerted anytime it crosses some boundaries. The developed system is a tracking device in the form of a sticker, thus diverting attention from the fact that it is a tracking device, which uses Radio Frequency Identification to identify objects. This sticker is attached to the device to be tracked. The developed system is able to identify devices successfully and notify users via SMS and email when their devices cross some boundaries. Users are also able to visualize the location of their devices on a map in order to know where they are at every point in time. This system also conducts statistical analysis using charts and tables to inform organizations about the behavior of their users and their devices. This can also give insight into the behavioral pattern of thieves.

The next section discusses the limitations of the work and suggests direction for future development.

6.2 Limitation

This system is affected by the reading range and strength of the reader. The reader used in this system has a reading range of about 40mm thus the tags have to be within this distance in order to be scanned. This is a limitation as in practice, the reader will be located on top of the doorpost.

The length of time for reading a tag is also a challenge. The tags were read within an average interval of 2 seconds. The reader should be able to read under a second as a number of devices may need to be scanned within a short space of time. The system also takes an average of 30 seconds for the user to receive a notification about the location of device. This is because the transfer of data into the database and the notification delivery time is network dependent.

Moreover, this system does not support keeping track of inventory as the reader may not be able to identify all the individual items in a package if they are moved together.

6.3 Future Work

Based on the limitations discussed above, there is more work that needs to be done to improve the system. An RFID reader with a higher reading range should be used to cater for the short reading range. A passive Ultrahigh frequency reader which has reading range of about 6 meters can be used.

The system should also be designed to allow for inventory management. This will broaden the scope of the system as it can be used for different purposes.

6.4 Conclusion

The issue of property crime and locating items is a worldwide issue which has grave consequences. Measures have been put in place to solve this issue but this problem still persists. Companies have sought to rely on tracking devices to curb this issue. However, these devices are either too expensive, require high power consumption or are bulky and cannot be used to track small devices. The implemented system fills these gaps by providing users with a sticker as a tracking device which is used in identifying devices. Users are notified anytime their device crosses some boundary. Moreover, this system can be used to monitor and determine the behavioral pattern of thieves for data analysis.

This project has indicated that low-cost inconspicuous tracking devices can be developed to help users monitor and locate the position of their devices.

REFERENCES

AB&R (2016). *What is RFID and How Does RFID Work?* Retrieved 3 October 2016, from http://www.abr.com/what-is-rfid-how-does-rfid-work/

Arduino.cc. (2017). *Introduction to the Arduino Board*. Retrieved 9 March 2017, from https://www.arduino.cc/en/reference/board

Beal, V. (2016). *What is Wi-Fi (IEEE 802.11x)?*Retrieved 8 October 2016, from http://www.webopedia.com/TERM/W/Wi_Fi.html

Bertagna, P. (2010). *How does a GPS tracking system work?* Retrieved 9 October 2016, from http://www.eetimes.com/document.asp?doc_id=1278363

Ebbett, S. (2016). *Protecting college students from theft, damage, or loss on campus*. Retrieved 26 September 2016, from http://www.collegexpress.com/articles-and-advice/student-life/articles/college-health-safety/protecting-college-students-costly-theft-damage-or-loss-essential-items/

Finch, C. (2016). *Advantages & Disadvantages of RFID | Techwalla.com*. Retrieved 9 October 2016, fromhttps://www.techwalla.com/articles/advantages-disadvantages-of-rfid

Fitzpatrick, J. (2015). *How to keep track of your stuff with bluetoothtrackers*. Retrieved 12 October 2016, fromhttp://www.howtogeek.com/222869/htg-explains-what-bluetooth-tracking-devices-are-and-why-you-might-want-one/

Girish, D. (2015). *iBeacon vs NFC vs GPS: Which indoor location technology will your business benefit from?* Retrieved 11 October 2016, from http://blog.beaconstac.com/2015/07/ibeacon-vs-nfc-vs-gps-which-indoor-location-technology-will-your-business-benefit-from/

Guha, S., Plarre, K., Lissner, D., Mitra, S., Krishna, B., Dutta, P., & Kumar, S. (2012). *AutoWitness: Locating and tracking stolen property while tolerating GPS and radio outages* (1st ed.). NY: ACM New York. Retrieved from http://delivery.acm.org/10.1145/2250000/2240120/a31-guha.pdf?ip=41.79.97.3&id=2240120&acc=ACTIVE%20SERVICE&key=CE2DA786A4560734%2E3A225D273C2F7FD2%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=669389622&CFTOKEN=75756774&__acm__=1475342356_6d0bdcdb6088c9710d0a38415d72ebf5

Hoffman, C. (2014). *Your devices broadcast unique numbers, and they're being used to track You*. Retrieved 8 October 2016, fromhttp://www.howtogeek.com/196998/your-devices-broadcast-unique-numbers-and-theyre-being-used-to-track-you/

Lehmann, S. (2015). *Tracking Human Mobility using WiFi signals*. Retrieved 8 October 2016, from https://sunelehmann.com/2015/05/26/tracking-human-mobility-using-wifi-signals/

Moran, J. (2014). *What Is a MAC Address?* Retrieved 8 October 2016, from http://www.webopedia.com/quick_ref/what_is_a_mac_address.asp

McDowell, G. (2009). *How Does RFID Technology Work?* Retrieved 3 October 2016, from http://www.makeuseof.com/tag/technology-explained-how-do-rfid-tags-work/

Ngula, J. (2014). *Concerned students Of UG petition the vice chancellor on campus insecurity |Ghana Campus News*. Retrieved 23 September 2016, from http://ugfile.com/concerned-students-of-ug-petition-vice-chancellor-on-campus-insecurity/

Peleg, M. (2016). *What is Global Positioning System (GPS)?*. Retrieved 9 October 2016, from http://searchmobilecomputing.techtarget.com/definition/Global-Positioning-System

Rakhi, K., Anand, P., Akshay, M., & Rohan, K. (2014). *RFID based toll collection system* (1st ed., pp. 2582-2585). Mumbai: International Journal of Computer Science and Information Technologies. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.439.4222&rep=rep1&type=pdf

Roberti, M. (2013). *How so RFID-based toll-collection systems work? Rfidjournal.com*. Retrieved 27 December 2016, from http://www.rfidjournal.com/blogs/experts/entry?10743

Seleniumhq.org (2017). *Selenium*. Retrieved 1 May 2017, from http://www.seleniumhq.org/projects/ide/

Smiley, S. (2016). *Active RFID vs. passive RFID: What's the difference? RFID insider*. Retrieved 27 December 2016, from http://blog.atlasrfidstore.com/a ctive-rfid-vs-passive-rfid

Smith, M. (2011). *How to use a GPS enabled smartphone as a tracking device*. Retrieved 9 October 2016, from http://www.makeuseof.com/tag/gps-enabled-smartphone-tracking-device/

Sommerville, I. (2011). *Software engineering* (9th ed.). Boston: Pearson.

Stallings, W. (2002). The advanced encryption standard (1st ed., pp. 165-188). Retrieved from http://dx.doi.org/10.1080/0161-110291890876

Tutorialspoint.com (2015). *Learn PhoneGap*. (1st ed., pp. 1-15). Retrieved from https://www.tutorialspoint.com/phonegap/phonegap_tutorial.pdf

Woodford, C. (2016). *How do RFID and RF tags work? Radio frequency (RF and RFID) tags*. Retrieved 6 October 2016, from http://www.explainthatstuff.com/rfid.html

Woodford, C. (2016). *How does Bluetooth work?* Retrieved 12 October 2016, from http://www.explainthatstuff.com/howbluetoothworks.html

Valerio, P. (2016). *MIT WiFi Technology Promises Precise Location Tracking*. Retrieved 6 October 2016, from http://www.networkcomputing.com/wireless-infrastructure/mit-wifi-technology-promises-precise-location-tracking/1586154916

Appendices

Appendix A: Unit Test class for the adding a device

```
include_once("../user.php");
class AddDeviceTest extends PHPUnit_Framework_TestCase{
    public function testAddDevice() {
           $tagID=56;
           $obj=new user();
           $this->assertEquals(true, $obj->addDevice("Iphone 6", 'A
           black phone with a blue-black case', 56, 3,
           '/Applications/XAMPP/xamppfiles/htdocs/suivre/img/phone.j
           pg'));

    }
}
```