KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY

COLLEGE OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

TOPIC**: THE WATCHDOG AND NAVIGATION ANDROID APPLICATION FOR KNUST**

THIS PROJECT WORK IS SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR A DEGREE OF BACHELOR OF SCIENCE OF COMPUTER SCIENCE

UNDERTAKEN BY:

BOAHEN OWUSU ELIJAH 1130113

COBBINAH BEATRICE 1116713

SUPERVISOR:

MR. OPPONG EMMANUEL

MAY 2017

DECLARATION

“We hereby declare that we have wholly undertaken the study report herein submitted”

---------------------------------- --------------------------------------

BEATRICE COBBINAH **Date**

---------------------------------- ------------------------- --------------

BOAHEN OWUSU ELIJAH **Date**

SUPERVISOR

“I declare that I have supervised the students in undertaking the study herein and I confirm that the students have my permission for to present for assessment.”

----------------------------------   ---------------------------------------

Mr. EMMANUEL OPPONG   **Date**

DEDICATION

We dedicate this project to the Lord Almighty for seeing us through successfully, our parents and all our loved ones.

ACKNOWLEDGEMENT

It is amongst the greatest clichés when writing a supervised report to say a “big thanks” to your supervisor. And as much as we despise starting our dissertation with a stereotype, we really do have to express our collective gratitude towards Mr. Emmanuel Oppong. We say a “big thanks” not only for his guidance and patience during this project but also for his mentoring throughout all our brief encounters with the research world in the past.

Many thanks to Ms. Cobbinah whose interest in this project was very beneficial and helped design many vital part of the final software. Also for helping us understand many issues around WatchDog during the early stages of the project and all her helpful suggestions about the report and presentation during the end.

Special acknowledgements to Mr. Castro whose lecturing on software engineering last semester greatly influenced our formal approach towards the project and the abstract analysis on which the final design was based.

1.

Table of Contents

CHAPTER ONE

INTRODUCTION

1.1 INTRODUCTION TO PROJECT

The latest mobile phones like android based mobile phones, also called as smartphones have become a very important part of our life. Smartphones change the ways of communication, it provides an advantage of communicating with anyone virtually through video-conferencing, email, etc., and it also provides a facility to store contact numbers, email id's, in phone memory which reduces the concept of File-System to store personal information. Nowadays, smartphones are acting like a computer, it can be used to store information, documents etc., these documents can be shared with anyone through the internet. These latest smartphones are very helpful for doing business work. Company related information and documents can be viewed anywhere and can be shared with anyone. These days android based mobiles phones/ devices are very popular because it provides a large number of utilities for hand-held devices through which it acts as a computer in a pocket. Because of its open-source nature a large number of utilities have been developed for android operating system and it is getting used in many mobile phones. Kwame Nkrumah University of Science and Technology (KNUST) is a great institution which possess one of the largest infrastructure in Ghana and Africa as a whole. The population of the institution is in its millions with many students coming in and others leaving. The one of the major problems of this institution however, over all these years has been security. Students often wander to unsafe places at unholy hours. Even in broad day light, sometimes students stand a risk of being in danger. This need of the reputable institution is what inspired the creation of this application. Due to the nature and availability of the android phone, we decided that this application should be written in the android language. The application includes an inbuilt map, a database, makes use of SMS and many other functions.

1.2 PROJECT CATEGORY

This navigation and watchdog application is an android mobile application designed to work on android platform for as many devices that make use of the operating system. As such it is expected to run on smartphones, tablets computers, and other mobile devices that have the android platform.

1.3 MOTIVATION FOR PROJECT

When selecting a topic for our project work, we considered a lot of other options, including web applications, e-commerce sites, game development so many others but we settled on this. This was mainly because, we decided that not only did we want to excel at our project work, but we wanted to find a need in our immediate surrounding a be a solution to that problem.

The first time you step on KNUST campus without a doubt, you will get lost and continually you would have to be seeking directions from people every step of the way and sometimes even with these directions you end up at a completely different place after having walked on for minutes and sometimes hours. Not only that but you may end up in a secluded environment with very dangerous people around, especially at night. Also KNUST faces a major challenge of security with constant reports of robbery, rapes and physical violence.

This application seeks to be a solution or at least a help to these two challenges that a student of this university is likely to face.

1.4 OBJECTIVES

The objectives of this system is quite simple. Currently, the only available system in terms of security is locate a security person, hide or just surrender and in terms of guidance and navigation is ask the nearest available person and pray to God they know the right direction.

As such the main objective of this system is as follows:

1. To provide a reliable source of navigation
2. To provide a means to send out SOS messages
3. To provide a means for the SOS message recipients to track you.

1.5 UNIQUE FEATURES OF THE SYSTEM

1. **Inbuilt map**

Making use of the google map, a map of KNUST and the whole world at large was attached to the system. Giving a clear view of every part of the school and the directions to each place.

2. Inbuilt tracking service

A background service has been built-in the application that allows other people to track the sender after a distress message has been sent. It also allows worried family members or friends to track their loved ones to their last known location.

3. SOS Distress message

It allows the user to send a message to alert others of a danger they are in at any point in time.

1.6 APPROACHES TO THE SYSTEM

In designing the software, we had to conduct extensive research into the following

1. Designing and creation of android application

2. Using the Google Map Api
3. Tracking services
4. Design and creation of android interfaces
5. Java development codes

1.7 ANALYSIS OF THE EXISTING SYSTEM

There is almost no existing system in places for navigation, what exist now is making relying on asking people around for your way out. Be it students, sellers, security personnel or whoever. It just mainly consists of finding the most available person and asking them for direction and if it so happens that they themselves are not familiar with the area you are going to then, you simply ask the next person available and so on.

In terms of security, so many measures have been experimented and some are functioning now, and they include “walking in holy pair” which is when students are advised not to go places alone because of safety issues. Security personnel being posted around and though this is so, the personnel cannot be put everywhere as such we still have some areas which have no security man at all especially for those off campus. And the ultimate if it gets too late just don’t move.

In as much as all these measures are good they still have not been able to solve the problem of safety.

1.7.2 FIRMING UP A SOLUTION

This project seeks to address the issues identified with the existing system. It also seeks to make the system as user friendly as possible so that users can patronize it and use it.

1.7.3 PROJECT EVALUATION

Feasibility study is an assessment of the practicality of a proposed project. Though the project seeks to address a major issue, in our estimation the project should meet the following feasibility requirement:

- **OPERATIONAL FEASIBILITY**

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development

This proposed system addresses the issue of security and navigation by

- Providing an application that a user can subscribe to and alert others of any danger he or she might be in

- It provides a user-friendly interface that allows users easy communicate with the systems
- An application that is portable (runs on your mobile phone and tablet), easy to use and available at any time anywhere
- It provides a map for navigation and for searches

- **ECONOMIC FEASIBILITY**

Economic feasibility relates to the cost involved in engineering the project from start to finish. Although this application is free, it can be considered economically feasible in the sense that looking at the cost of bringing up the project- research cost, time factor and others as compared with the benefits it will provide to its users, it can be concluded that it is economically feasible.

- **TECHNICAL FEASIBILITY**

  The *technical feasibility* assessment is focused on gaining an understanding of the present technical resources of the organization and their applicability to the expected needs of the proposed system.

  For this project the technical resources required are internet service, knowledge of android, Java and other programming languages and a good functioning computer that can support the software that runs these programming languages.

  These technical resources are readily available to us and hence this project meets the standards of technical feasibility

CHAPTER TWO

LITERATURE REVIEW

2.1 BACKGROUND REVIEW

In this fast evolving world of technology where time has become an asset of great value and

Information which without doubts has cumulatively helped optimize time to the maximum, it has become necessary to have systems that help to couple the two v ery well.

The idea of the whole document is to represent bounded physical representation of body of information and also to proof whether what we designed has the capacity to communicate or not. This document contains diagrammatic, symbolic and sensory-representational information of the whole project. This document produces an artefact by collecting and representing information. This document projects the comprehension of the user in a very lucid manner. This document explains hardware and software requirements, user view of product use, general constraints and guidelines of the project. Documentation is nothing but a method of communicating what the whole project is about. Objective, realistic and complete are the importance factors kept in mind while writing satisfactory documentation. Moreover, adequacy of the project is not based on length, format, complexity or volume.

No standardized documentation is described to write projects. The whole process of implementation, operation of system and development is based on proper documentation.

Documentation can also be done by embedding proper comments in the executable code of the modules.

Parenthesis, spaces, blank lines along with suitable loops around the comments block were used to improve readability.

2.2 ANALYSIS OF EXISTING SYSTEM

There are couple of mobile security applications currently in use. Typical among them are our traditional ***ihound*** application, ***where is my droid*** application. We will carefully examine these systems that we have in place: ihound antitheft mobile application all around us is a typical instance of an existing system. We all are aware of the stress we mostly have to go through to access even a common incoming SMS patronizing these apps. How we would have to comb our way through to be able to see texts of other folks who patronize the application. Here on campus, we notice how devastated people get if they are unable to familiarize themselves with these already existing apps.

. In my opinion as a user of these anti-theft systems, it is quiet distressing to see the same features every day without amendments or updates or the incorporation of new and relevant features. Some students have actually rubbished these systems already and do not seem to even want to use them again.

2.2.1 PROBLEM IDENTIFICATION

As described, the inefficiencies of the systems we have taken time to study have prompted us to develop this application. Problems identified with the current system includes:

- **Inefficiencies of the existing system:** The Ihound anti-theft mobile application and other related systems are quite inefficient in their pursuit of disseminating information in the sense that in case of emergency, the systems cannot in their entirety generate emergency messages to be sent out, and even if there were able to generate these relevant messages to be sent out, the time taken for these messages to reach their destinations is also another problem to talk about.
- **Technical inability to support authentication**

  Most anti-theft mobile apps we have seen so far on campus often do not have passwords enabled. Anti-theft mobile devices often lack passwords to authenticate users and control access to data stored on the devices. Many devices have the technical capability to support passwords, personal identification numbers (PIN), or pattern screen locks for authentication. Some mobile devices also include a biometric reader to scan a fingerprint for authentication. However, anecdotal information indicates that consumers seldom employ these mechanisms. Additionally, if users do use a password or PIN they often choose passwords or PINs that can be easily determined or bypassed, such as 1234 or 0000. Without passwords or PINs to lock the device, there is increased risk that stolen or lost phones' information could be accessed by unauthorized users who could view sensitive information and misuse mobile devices.

2.2.2 PROPOSED SOLUTION TO SETBACKS IN EXISTING PROBLEMS

This application seeks to resolve the problems associated with the current system in use by accomplishing the following:

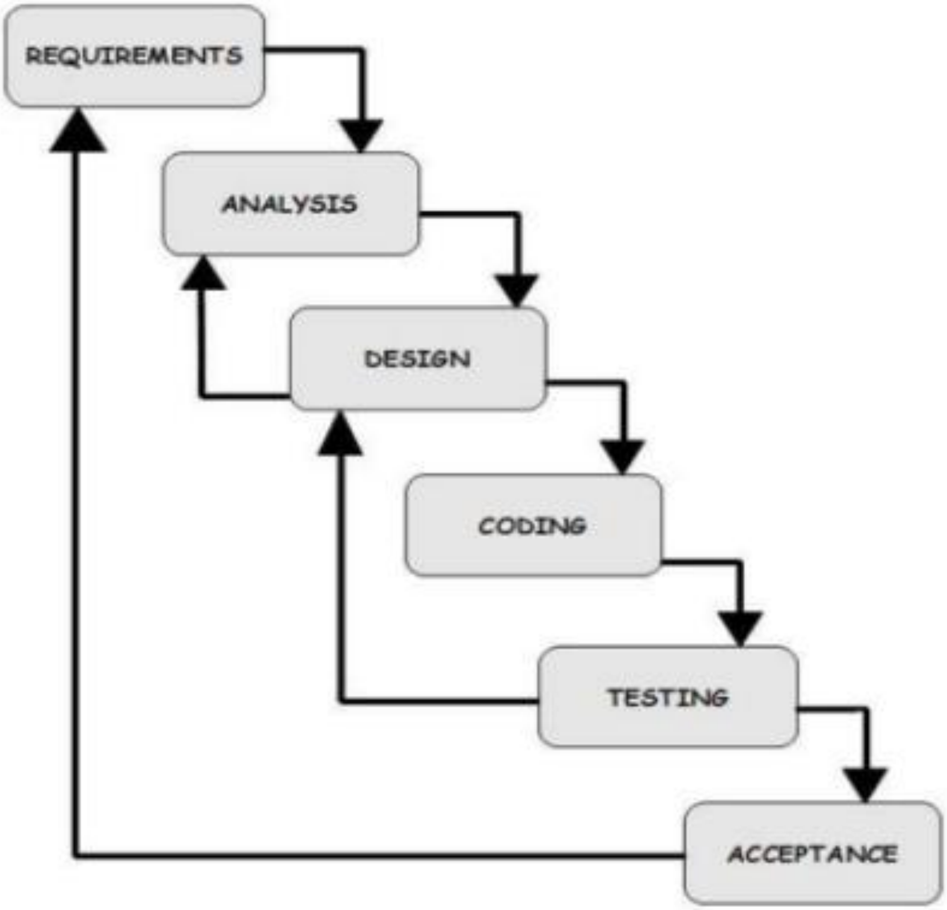
1. Bring up a reliable android application which is easy to use and accessible anytime anywhere at the user’s own convenience.
2. An application that provides amendments and invariably incorporates new features in
    a. this era of technological evolution.
3. Well organized notices which can be searched using its efficient search tool.
4. Users can tag an associated notice with a reminder so they can be alerted when an event’s date is due.
5. Bring up self-updating application which is up-to-date and reliable

6. Develop an application which is efficient and trustworthy because only administrators will have the exclusive right to post notices.

2.3 REVIEW OF PROJECT METHODOLOGY

A number of methodologies can be used for software development in software engineering which include incremental, waterfall approach, prototyping, spiral and many others. After carefully considering all the methodologies, waterfall approach was the preferred. .

The Waterfall method is comprised of a series of very definite phases with each phase intended to be started sequentially only after the last has been completed. One or more tangible deliverables is produced at the end of each phase of the waterfall method of Software development life cycle. Essentially, it starts with a heavy, documented, requirements planning phase that outlines all the requirements for the project, followed by sequential phases of design, coding, test-casing, documentation, verification (alpha-testing), validation (beta-testing), and finally deployment /release.



EXPLANATION OF THE WATERFALL MODEL

Requirement

This initiates the whole process and it encompasses those tasks that go into determining the needs or conditions to meet for the new or altered product, taking account the possibly conflicting requirements of the various stakeholders such as beneficiaries of the project. Requirement analysis is key to the success of every project because if requirements are not clearly known, output may not meet the purpose of the software and hence the users may not use that software you have spent time to develop-all because requirements were not clearly spelt out.

Analysis and software design

This stage produces the logical and physical design specification for the solution by taking the requirements and producing a system design. This stage would define how programmers will implement the requirements document in the development of the software.

Coding and Testing

On receiving system design documents, the work is divided in modules or units and actual coding is started. The system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality. This is referred to as unit testing. Unit testing mainly verifies if the modules meet their specifications.

Acceptance

This phase is virtually never ends. Operation and maintenance of the system is key to its acceptance. Most often, problems of the system developed which are not identified using its development come up when it is put in practical use. Not all problems come up immediately the software is deployed but as and when problems arise from the software, it has to be addressed. The software also needs to be maintained over time to ensure that it is functioning as effectively and effectively as it should be.

2.4 PROPOSED SYSTEM

This software is freely available.

The main advantage of this application is anyone can use it without having much knowledge about the device.

The application meets user’s immediate and long term requirements by providing the images and videos of the thief.

Easy for the user to identify the thief and make him/her get caught and arrested.

This application provides the information about the location of the android based smart phone with the help of e-mail.

The developed anti-theft app will enable user to use his android based smartphone with freedom of getting stolen. It will enhance the security of the android based smartphone.

2.4.1 FEATURES OF PROPOSED SYSTEM

2. The features this application include:

3. Ability to send messages
4. Ability to acquire your exact GPS location
5. Ability to check your current location and your destination on a map
6. Ability to search for the location and direction to specific places
7. Ability to track a required number
8. Ability to save contacts

2.5 DEVELOPMENT TOOLS

Development tools used for the project are:

- **Android Development Tools (eclipse and android studio)**: This is a development platform for android applications. It has tools which aid in developing the interface of android applications but the main programming language it uses to power its interfaces to function is Java. Android studio was used in developing the android part of the application.
- **Java** : This is mainly the language on which the android language is built on.

CHAPTER THREE

REQUIREMENTS SPECIFICATION

3.1 REQUIREMENT CAPTURE

In this chapter, we specify categorically, the details of all the requirements that are needed to be met for the software to run successfully. Requirements capture or gathering is the act of trying to understand a problem by talking to a selection of actual and potential users. It is commonly placed in virtually, all good IT projects. Traditional waterfall projects require that a problem is fully understood, and documented, before beginning to build the solution. Agile projects stipulate that only a “broad brush” understanding of the problem is required to start work, with the gaps in knowledge being filled in as the project progresses. But generally speaking, any type of project, large or small, agile or waterfall, will have some form of requirements gathering component to it.

Requirements can be either functional or nonfunctional in nature. A functional requirement relates directly to a process the system has to perform or information it needs to contain. For example, a process-oriented functional requirement would be that the system must have the ability to search for available inventory. An information-oriented functional requirement would be that the system must include actual and budgeted expenses. Functional requirements flow directly into the next steps of the analysis process (use cases, process models, data model) because they define the functions that the system needs to have. Nonfunctional requirements refer to behavioral properties that the system must have, such as performance and usability. The ability to access the system through a Web browser would be considered a nonfunctional requirement. Nonfunctional requirements may influence the rest of the analysis process (use cases, process models, and data model), but often do so only indirectly; nonfunctional requirements are primarily used in the design phase when decisions are made about the user interface, the hardware and software, and the system's underlying architecture.

3.2 METHOD OF REQUIREMENT CAPTURE

During this project we used the method of direct observation and interviews to gather the necessary information we need.

In the direct observation method we noticed that on campus usually during political semesters there is a chaotic display posters on the notice boards at least for best case scenario. To add more salt to injury the politicians in their quest for attention tend to scribble notices on the walls and pavement. Sometimes in the middle of the street. Usually an audience may be reached with this kind of method but what is the guarantee that the information really gets down to them.

Also during our observation it came to our notice that some of the existing platforms that were used to disseminate information did not have credible sources. An example is the whatsapp platform. Many people on whatsapp have equal rights as the sender of a message or notice to redistribute the messages that they receive. With this enablement people tend to edit the notices and redistribute the “adulterated” one. The target audience will obviously receive the wrong notices we all know the consequences.

In interviews conducted we found out from the university populace that they needed a reliable and credible source of information which they could carry on them everywhere they go. They preferred something on their mobile phones. They also expressed interest in the fact that a simple easily navigable graphical user interface will be appreciated.

"A customizable interface will be great too", one interviewee added.

If there is any one thing any project must have in order not to be doomed to failure, that is an intelligent and comprehensive collection of both the functional and non-functional requirements.

3.3.2 FUNCTIONAL REQUIREMENT

As stated earlier functional requirements are associated with specific functions tasks or behaviors the system must support.

In order to make this application functional, we require the following:

1. **Download mobile application:**

A user should be able to download the mobile an application through either an application store or similar service on the mobile phone. The application should be free to download.

2. **User registration:**

Given that a user has downloaded the mobile application, then the user should be able to register through the mobile application. The user must provide user-name, password and e-mail address.

The user can choose to provide a regularly used phone number.

3. **User Login:**

Given that a user has registered, then the user should be able to log in to the mobile application. The log-in information will be stored on the phone and in the future the user should be logged in automatically.

4. **Dash Board:**

Given that a user is logged in to the mobile application, then the first page that is shown should be the dashboard page. The user should be able to see all the notices that make headlines.

5. **Search Notice:**

The user should be able to search for a notice by entering a keyword. The user can also search by entering the date to bring out all notices that were sent on that date.

6. **Selecting a Notice:**

A user should be able to select any notice from the dashboard. The click on particular notice will take him to notice details of that particular notice. Also selecting a particular category example Religion should take you to all notices about religion.

7. **Navigating back to Notices List:**

The user should be able to navigate back to notices list from the notice details section. This is required to give a good user experience.

8. **Posting Notices:**

The administrator of this application should be able to post the notices. He should be able to add a picture within notices. That picture can be taken either from gallery or by using the camera of the mobile phone.

9. **Administrator Login:**

The administrator should be able to login with his details in order to gain full access to the web application.

10. **Web-master privileges:**

The web-master is the overall controller of the application. With the highest privileges. He can add or delete administrators.

11. **Notification Alert:**

All the registered users should be able to have a ping or notification on their mobile phone whenever a new notice is posted.

12.•Turn on and off Notifications:

Users should be able to turn on and off the ability to receive notifications usually to save data.

13.Reminder feature:

Users can scribble short notes and schedule the notes so as to remind them of an upcoming event. On the specified date the alarm goes off and notification pops up to remind the user.

14.Customize the user interface:

The user can select themes as well as customize the appearance of fonts.

3.3.3 NON-FUNCTIONAL REQUIREMENTS

1. **Usability**: The mobile phone application should easily be very user friendly and as a matter of fact you don't need any extensive training to operate the application.

2. **Portability:** The application can be carried on every android mobile phone that uses Kit-Kat and above. This accounts for about 70% of all android devices.

3. **Simple and Light:** The user interface should be simple and lightly colored. It should give relaxing effect on looking at its GUI

3.4 SYSTEM REQUIREMENT

System Requirement can be categorized into two. Software requirements and Hardware requirements

3.4.1 SOFTWARE REQUIREMENTS

Poke requires an android version of Kit Kat or higher.

3.4.2 HARDWARE REQUIREMENTS

1. Devices should have a minimum of 128 Mb RAM

2. Devices should have access to Internet in order to receive notifications

3. an internal memory of 2gb

3.5 REQUIREMENT ANALYSIS

Requirements analysis in systems engineering and software engineering, encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking account of the possibly conflicting requirements of the various stakeholders, *analyzing, documenting, validating and managing* software or system requirements

Requirements analysis is critical to the success or failure of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.

In our project the requirements that we gathered included the desire of the potential users to be able to edit and send the messages to their fellow students on the application. But this contradicts with the fact that we as a team want to avoid the issue of dilution of information. If students are able to edit and resend the information on the application it implies that credible information sent from us can easily be altered to something else. This might be harmful for consumption and as a result cause more harm than good if care is not taken. An issue is reported once that someone sent a message on whatsapp platform claiming that the dean of students had declared a certain day as a holiday only to find out later that it was just a scam. The origin of the message could not be traced up till now.

In exchange for that desire expressed by users we have included the feature that allows users to share the information received on other platforms at least but we have not allowed them to edit it. This way users can always turn to our application to resolve doubt issues about certain information about events on campus they have seen elsewhere.

The users requested for the ability to view videos and images too on our application. This requirement is beyond our knowledge scope and we are studying assiduously to ensure that we implement this feature in our next version of the application to be released soon.

Aside these two requirements from our potential users the other requirements are expressed in the functionalities of the application.

3.6 USE CASE

A use case is a formal way of representing how a business system interacts with its environment. A use case illustrates the activities that are performed by the users of the system. As such, use case modeling is often thought of as an external or functional view of a business process, showing how the users view the process rather than the internal mechanisms by which the process and supporting mechanisms operate.

A use case depicts a set of activities performed to produce some output result. Each use case describes how an external user triggers an event to which the system must respond. For example, in our system users can view notices, create, edit, delete and schedule reminders. With this type of event-driven modeling, everything in the system can be thought of as a response to some trigger event. When there are no events, the system is at rest, patiently waiting for the next event to trigger it. When a trigger event occurs, the system (and the people using it) responds, performs the actions defined in the use case, and then returns to the waiting state.

In some situations, the process may be “small,” such as the actions that are performed when a video is rented in the previous example. In more complex systems a use case may require several distinct activities, some of which are performed each time the use case is activated and some of which are performed only occasionally (e.g., consider the return of a rented video, which very rarely will be returned with damage). Simple use cases may have only one path through them, while complex use cases may have several possible paths. We create use cases only when they are likely to help us better understand the situation and help simplify the modeling steps that follow. For very simple processes that are well explained in the requirements definition, we often do not bother to create a use case, but simply use the information in the requirements definition itself to build the process and data models .It is important to create use cases whenever we are re-engineering processes or making any changes to business processes that will significantly alter the way people work. Remember that the use case describes what the system will do from the user's perspective. Therefore, it is critical to involve the user in the creation of the use case so that the user understands the activities planned for the new system. Also, the user helps to ensure that no essential steps or tasks are omitted from the use case and that rare, special circumstances are included.