Adventist University of Central Africa (AUCA)

SMART CRIME SCENE EVIDENCES ANALYSIS

Case Study: RWANDA INVESTIGATION BOARD (RIB)

A project

Presented in the partial fulfillment of the requirements

For the degree of

BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY

Major in

SOFTWARE ENGINEERING

By

MUGABO Yves

May, 2022

DECLARATION

I, **MUGABO Yves** hereby declare that to the best of my knowledge this work has been done by me and has not received any previous academic credit at this or any other University or institution.

Student

MUGABO Yves

Signature: …….

Date: ……/……/……...

DEDICATION

I dedicate this work to my family
For their support during all the time of my studies.
I also dedicate it to all my friends, especially students at AUCA.
Without forgetting, also I dedicate this work to my supervisor for his guidance.

APPROVAL

I, **Mr. MANIRAHO Laurent** hereby certify that this project report has been done under my supervision and submitted with my approval.

Signature: …….

Date: ……/……/…….

PROJECT ABSTRACT

Project for Bachelor Degree in Information Technology

Emphasis in Software Engineering

Adventist University of Central Africa

TOPIC: SMART CRIME SCENE EVIDENCES ANALYSIS

Name of this researcher: MUGABO Yves

Name of faculty Advisor: Mr. MANIRAHO Laurent

Date Completed: May, 2022.

Crime is a human experience and is as old as the human race. Crime is geographical. It occurs at a specific place, specific time and for a specific reason. It can affect everyone and anyone at any time.

The main purpose of this project is focused on the design of Smart Crime Scene Evidences Analysis projects which will be a good tool to analyze the evidences, make decisions and to easily find out the primary suspects based on the new evidence and criminal record of the suspect.

Briefly this system will help RIB stations to record the suspects to a certain case and attach evidences found at the crime scene then based on past criminal record and present evidences of the specific suspected person, the system will present primary suspect to RIB user logged into the system

These are following Technologies and tools used:

In order to run Smart Crime Scene Evidences Analysis System on server side we need:

- ✓ Python 3.7 installed in environment variables in local computer.
- ✓ Django 2.2 installed.
- ✓ PostgreSQL for backend database, which is able to serve as data store
- ✓ Visual Studio Code as Integrated Development Environment (IDE) for many programming languages.

ACKNOWLEDGEMENT

My praise and sincere thanks goes to my God Almighty who blessed me with his grace and blessings throughout my life.

I sincerely have deep recognition to my supervisor **Mr. MANIRAHO Laurent** who guided me in the accomplishment of this work, especially for his substantial advice, professional assistance, guidance and precious ideas.

I thank the administration of Adventist University of Central Africa as well as the staff of department of Information Technology.

I am also very grateful and extend my sincere thanks to the General Secretary of Rwanda Investigation Board for their cooperation by sharing the information to make me have time to work on this project and throughout my study.

Finally, special thanks to my family, colleagues and friends for their support, encouragement, friendship and those who have helped and given suggestions, supports and participated in one way or another in the accomplishment of this work.

God bless you all.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AUCA - Adventist University of Central Africa

RIB - Rwanda investigation Bureau

SCSEA - Smart Crime Scene Evidences Analysis

DBMS - Database Management System

GUI - Graphical User Interface

HTML - Hyper Text Mark-up Language

CSS - Cascading Style sheet

HTTP - Hyper Text Transfer Protocol

ICT - Information Communication Technology

IDE - Integrated Development Environment

IT - Information Technology

OS - Operating System

RAM - Random Access Memory

SQL - Structured Query Language

UML - Unified Modelling Language

XML - Extensible Mark-up Language

DEFINITION OF KEY TERMS

System

System is an organized, purposeful structure that consists of interrelated and interdependent elements (components, entities, factors, members, parts etc.…). These elements continually influence one another (directly or indirectly) to maintain their activity and existence of the system, in order to achieve the goal of the system. All systems have inputs, outputs and feedback mechanisms, maintain an internal steady-state (called homeostasis) despite a changing external environment and have boundaries that are usually defined by the system observer. (John Onunga, 2005)

Information system

Information system is an integrated set of components for collecting, storing and processing data and for providing information, knowledge and digital product. Business firms and other organizations rely on information systems to carry out and manage their operations, interact with their customers and suppliers and complete in the market place. Information systems are used to run inter-organizational supply chains and electronic market. (Jessup, Leonard M. Valacich, Joseph S, 2008)

Database

A database is a collection of related data. This database will help to store all information related to design and implementation of electronic examination system; a database is an organized collection of data. The data are typically organized to model relevant aspects of reality in a way that supports processes requiring this information (Robert J. Robbins, 1995).

CHAPTER 1

GENERAL INTRODUCTION

Nowadays society is characterized by increasing levels of global social mobility and uncertainty relating to levels of risk posed by internal and external security threats. Within this climate security driven by technology is increasingly being used by governments, corporate bodies and individuals to monitor and reduce risk.

Generally, Automation plays an important role in the global business, data analysis and in daily experience. As part of its ongoing objective to the improvement of data analysis through information and communication technology (ICT), engineers are working so hard to combine the automated devices with the organizational tools to create complex systems for a rapidly expanding range of the applications.

Both Information Technology (IT) and crime are complex and constantly changing. Social and technological changes introduce new targets, tools and motives for crime. E-crime targets include confidential information, technical infrastructure and denial of service. Improved and faster technology and worldwide communications make it easier to both organize and hide crime. Above cybercrimes there are many crimes that are not Internet related and the use of technology is a solution to crime control complexity.

This project deals with data analysis of the crime scene evidences where it refers to the presented new evidences of the suspect and with the suspect criminal records for it to present the primary suspected person.

Thus for SMART CRIME SCENE EVIDENCES ANALYSIS information technology can be the most advantageous way to be used in crime analysis.

Background of the Study

Crime is still the one of most challenges that most governments around the world are struggling with. Families and businessmen are still being directly or indirectly affected by robberies and other many types of crimes. So, after seeing the RIB existing system takes almost 72 hours to present a primary suspect and sometimes reports comes with errors due to human justice which can be easily diverted in one way or another.

This project has been conceived in order to propose a model of analysing crime evidences in order to facilitate and simplify these actions one of the major tools is to have automated crime scene evidence analysis system which consists of tasks such as registering suspects, revising his/her past criminal report compare it with the evidences and report the summarized report.

Automation is the utilization of technology to replace human with a machine that can perform more quickly and more continuously.

Statement of the Problem

One major problem Hindering faster evidences analysis to present the primary suspect is that investigation officers are still using a physical log book to record crime scene evidences and their human judgement in analysing the evidence as well as considering past suspects criminal records in the process of presenting primary suspected person.

Nowadays crime rates are increasing day and night which resulted in having a queues of cases to be analysed and then the reporting of suspected person will result to:

- Errors in reporting.
- Taking too long.
- Injustice due to poor of following up criminal.

Choice and Motivation in the Study

After observing the process of finding the suspected person among others where some people last in RIB stations for 72hours yet they are innocent. I found that it was not an appropriate way to do so.

Indeed, that pulled my attention and thus thought I could develop a system that would improve it. This was a big occasion for me, as an IT student to put in practice what I have been studying throughout my three years at University.

For my university, it will be a way of achieving one of AUCA's objectives which is to help students to become useful member of society; empowered by the desire to develop the society not only with theoretical intellectual skills but also with a demonstrated desire of practical research Endeavour's that lead to country's development.

Objective of the Project

General Objective

The Smart crime scene evidences analysis system is generally aimed to exclude human-based crime scene evidences analysis which is done manually. So that this new system is intended to provide better crime registration and better primary suspect reporting system on time and fewer loss of time due to fast evidences analysis using information technology.

Specific Objectives

Specifically, this project is expected to solve problems found as follow:

1. Analysis of the already existing system of crime scene evidences analysis system which is paper-base, knowing the problems with that system and the impact there are having on crime control and primary suspect.

2. Develop a crime scene evidences analysis system which will be a web-based solution to those problems as it will provide better on time crime reporting and computerized system provides accuracy, security, and reduced redundancy, reduction of work load, backup facilities and faster criminal information retrieval, it will also provide good crime reports to the investigation officers.

3. Test of the application to be sure that it has solved the problems that was there before and make sure that it is able to provide a good crime reporting way to RIB Station that exclude the time spent to investigation offices and other related expenditures and provide crime reports to the investigation officers.

Scope of the Study

This project in mainly focuses on providing primary suspected person reports about crime scene' evidences, past criminal record reports of the suspect so that RIB must have access to complete, accurate and timely information about the suspects and evidences. One of the benefits of automated SCSEA is that the suspect record system will simplify retrieval of required information and is a great instrument for RIB improvement by taking measures from information acquired then the system will analyse the evidence and past criminal records of the suspects and generate reports of the primary suspected person to the RIB station user. This will solve the problem of late and high process of getting primary suspected person from others.

Methodology and Techniques Used in the Study

After knowing how the current system is working, the study has got started by researching how the crime scene evidences analyses are being done from the crime scene to investigation bureau are structured hierarchically according to the roles of investigation officers, the study has been carried on how different functionalities should be added for better use and improvement in such conditions to minimize the time it takes them to find out the suspected person.

It is not to imagine a system of information without having complete information of what happens within the enterprise or organization in its activities of every day**.**

To reach this, it was necessary to use different methods to collect the necessary information coming from users (RIB stations). In this research, three techniques of data collection which are the interview, documentation and the observation where used.

Observation

Observation is a systematic data collection. It is the action or process of closely observing or monitoring something or someone. To observe, means going deeper in data collected or prolonged engagement in a setting or social situation. Observation is one of the methods used to collect information over the existing situation, so it has been used to help in this study to collect a lot of information and based on it became very easy to know the way the existing system were working. As mentioned in the definition above one of ways used in this study in understanding the flow of

work of existing system, an observation in Rwanda Investigation Bureau looking on how they work during the crime reporting was made. (technopedia, 2021)

Documentation

Documentation is a material that provides official information or evidence that serves as a record. It is also a set of documents provided on paper, online, on digital or analogue media, such as audio tape or CDs. This method was used when collecting data from files and official documents of Rwanda Investigation Bureau relevant to this work. The documentation helps in the study to get more familiar with the various tools and techniques to be used to develop the application. Reading more documents about how crime reporting is made, viewing all the process that is taken to do it has been a big benefit to in the journey of making a computerized crime reporting system. (technopedia, 2021)

Interview

An interview is a conversation between two or more people where questions are asked by the interviewer to elicit facts or statements from the interviewee. Thus, in order to attain highly personalized information data about the operation of the existing system and to better understand process of crime scene evidences analyses in Rwanda Investigation Bureau (RIB) an interview with the Investigation Support division was taken and they have explained to me all the process of evidence collection and analysis in (RIB) step by step and some of the issues they face based on paperwork.

In an interview with Investigator Charles who is in Investigation Support division, I have asked him the processes that are followed in crime scene evidences analysis and the answer was this; “First the reporter has to go/send a call to the nearer RIB station, then the investigation team will have to show up to the crime scene to conduct the crime scene evidences, then investigator will have to analyse the case evidence with the suspects and categorizes the case according to the crime case category. After all, the reporter should fill the investigation form where he/she puts there his/her names, name of the suspect and place where the crime has been occurred to Then the investigation would start as well as analyzing the evidences”.

I have also asked him if there is any problem that they are face while using this system which is manual, and he has answered to me in these words “the first problem we are facing is time consuming while recording crime cases and take another time to revise the case history of the

suspects. Not only this problem but also there are other problems with this manual system that we are using which include updating forms when there is a mistake in writing which requires us to overwrite the whole information given.

Expected Results

The new system will allow the following results:

- ✓ Human judgement will be turned to Automation with the use of Evidences analysis.
- ✓ The automation of the already existing system that will ease the entry of records.
- ✓ Quick and easy primary suspect reporting.
- ✓ Allow the Investigator to manage the crimes.
- ✓ Help the Investigation officers to view crime report online and may download it when needed.
- ✓ Generate reports about evidences reporter.
- ✓ Minimizing human errors in decision making.
- ✓ Generate reports about crime cases status.
- ✓ The system will report all cases that are in each RIB Station.
- ✓ Easy to review the past criminal records of the suspects.
- ✓ Generate reports on places where the crime has taken place.
- ✓ Generate reports on past criminal records of the suspects.

Organization of the Report

This study contains five chapters:

Chapter one: talks about General Introduction. It will provide basic information on the research project, will present problems in to be solved, objectives, scope of the study, methods and techniques of the study, expected results and organization of the report.

Chapter two: entitled Analysis of the existing system, will focus on the analysis of the existing system where we will describe the operation of the current system, its history, vision and mission, deeply discover issues, and suggested solutions proposed on those problems.

Chapter three has the title of Requirements analysis and design of the new system, is normally the logical conception of the new system. It will portray the conceptual process of the solutions proposed to solve the problems of the existing system.

Chapter four will emphasis on the technical realization of the application and the presentation of screenshots as forms of data entry and reports, where we will explain the new system and how the application has been conceived, as well as the technologies used to build the software.

Finally,

Chapter five will conclude our project as well as recommendations for future development.

CHAPTER 2

ANALYSIS OF THE EXISTING SYSTEM

Introduction

In this chapter, we will take a look on the existing system which regards on everything concerning smart crime scene evidences analysis system how it works, and the problems it causes and also their effects in daily life and how the system in implementation will answer to them. The analysis of the existing system consists of learning the existing system operations, the functional activities and non-functional activities of it. Often, the purpose of designing a new system is to replace the manual system with an automated one. Before the end of this chapter, proposed solution to solve the mentioned problems will be highlighted.

This project emphasizes on evidences analysis and suspect reporting system in Rwanda Investigation Bureau. Therefore, we give an overview of the evidences analysis and reporting system of suspects in Rwanda.

Description of Current System Environment

While making the analysis of the existing system, we need first to know all about the organization that we are working with which is RIB. It is in charge of crime controls. It is very necessary first take a look at the existing environment so that we know the history, vision and mission of the organization with the existing system then know the problems faced by it and the proposed solution.

Organization Chart

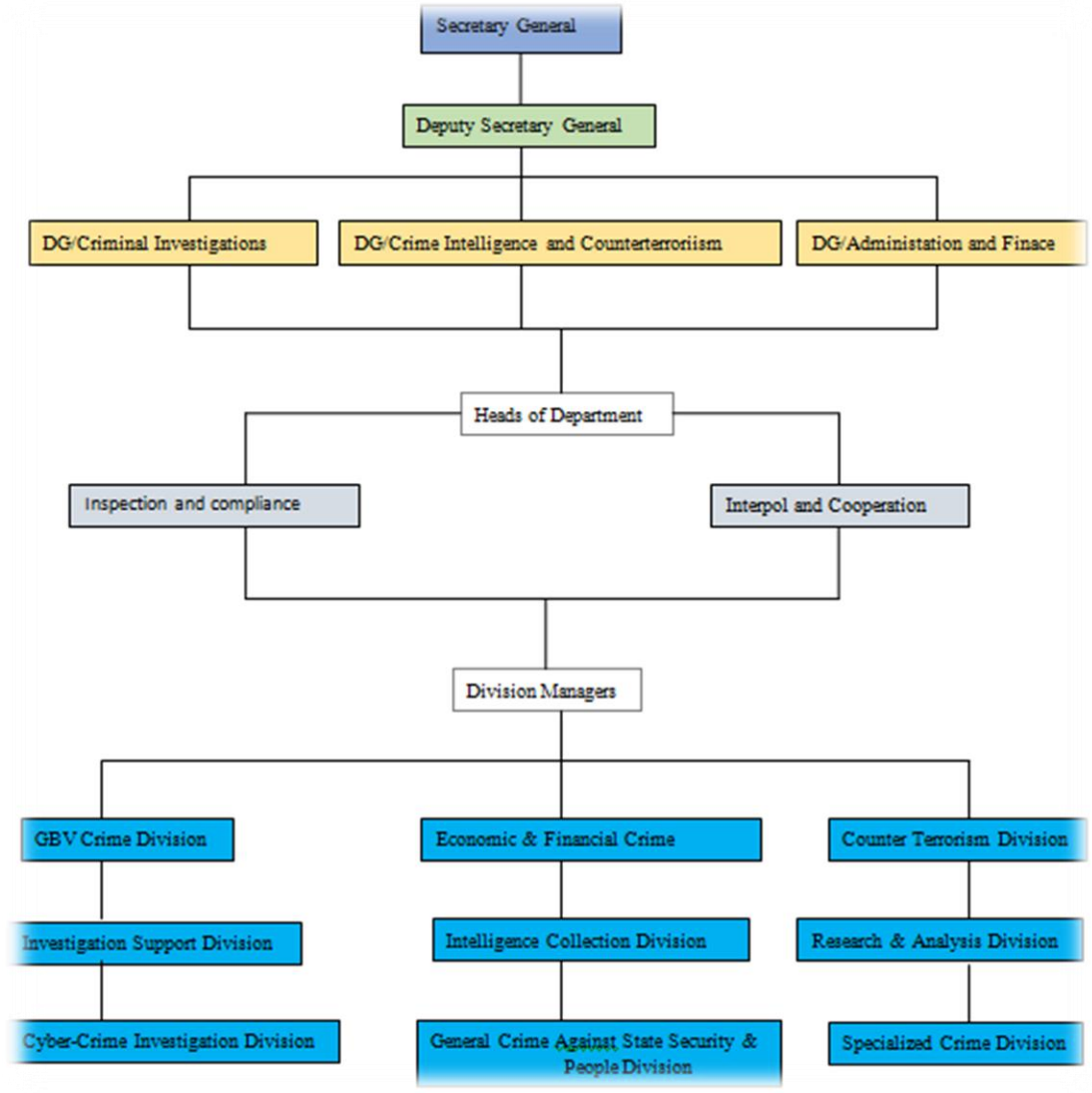


Figure 1: Organisation Structure

Who is RIB?

The Rwanda Investigation Bureau (RIB) is an autonomous specialized body established by the law N°12/2017 of 07/04/2017 and responsible for performing career investigative functions, gathers evidence, and assists local law enforcement agencies in ensuring law and order.

The law provided a traditional period of one year for RIB to assume criminal investigation responsibilities from 20th April 2017. (RIB, 2021)

Historical background

On 09th April 2018, His Excellency Paul KAGAME, the President of the Republic of Rwanda, appointed the Secretary General and the Deputy Secretary General of the Bureau. This was followed by the swearing in of the Secretary General and the Deputy Secretary General in function presided over by the Head of State on 10th April 2018. (RIB, 2021)

On 11th April 2018 the cabinet meeting chaired by His Excellency Paul KAGAME, the President of the Republic of Rwanda approved the following Orders:

- Presidential Order's Order establishing salaries and fringe benefits for Rwanda Investigation Bureau staff;
- Prime Minister's Order determining organizational structure of Rwanda Investigation Bureau
- Prime Minister's Order determining modalities of transfer of 463 personnel and assets related to criminal investigation department from Rwanda National Police (RNP) to Rwanda Investigation Bureau.

On 18th April 2018, Rwanda National Police formally handed over criminal investigation responsibilities to RIB in an event that was presided over by the Minister of Justice/Attorney General.

RIB's insignias, including the logo, a website and contact telephone lines were also unveiled.

On the deadline of the transition provided by the law, April 20, 2018, RIB started operating countrywide and the Bureau Leadership began with field visits across the country, starting with City of Kigali to assess the preparedness and readiness of RIB station Bureau and staff.

Mission

Rwanda Investigation Bureau's mission is to ensure professionalism, transparency and open-ended independence in criminal investigations with a broad focus on high impact crimes including but not limited to;

- Counter terrorism
- Cybercrime
- Gender based violence
- Public fund embezzlement and corruption

- Human trafficking
- Drug trafficking

Vision

To ensure a crime-free society

Description of the Current System

The present system is a manual system. In case of present manual system all the records are kept in documented form and stored in various types of registers. The manual system has no proper system for record storage and storing the information about crimes and criminals. The present system of RIB is not automated. The documentation of criminal records is done manually. Case files are dumped in heaps while some are kept on shelves. These important documents quickly get dusty and because of lack of proper storage.

Analysis of the Current System

Modelling Current System

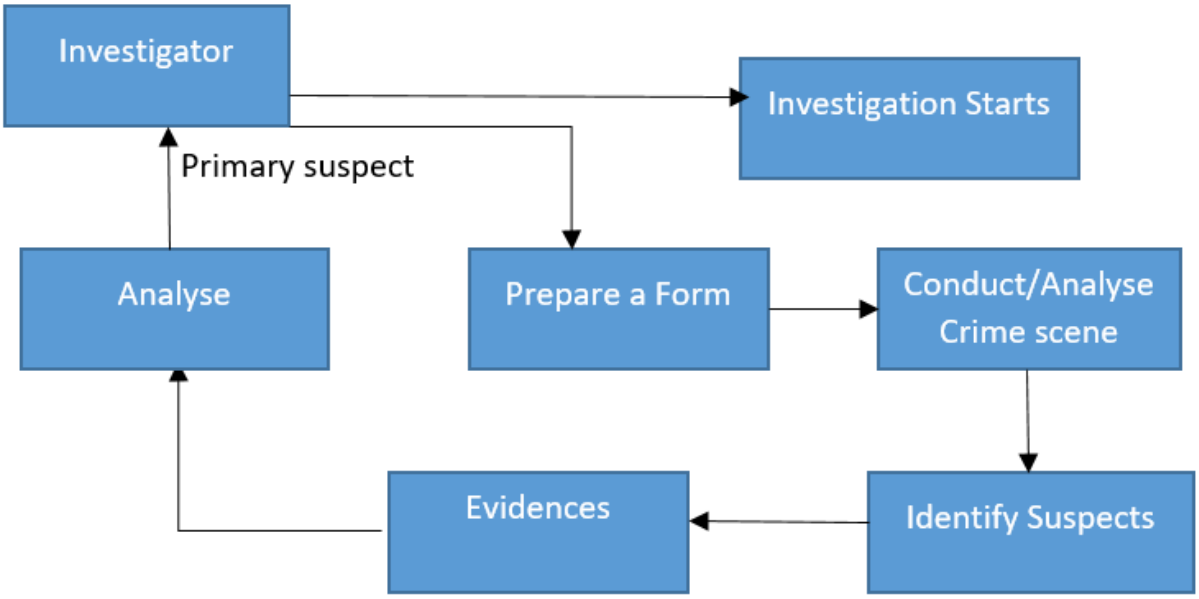


Figure 2: Modelling Current System

Problems of the existing system

Fewer of systems that will allow the investigation bureaus to easily identify suspected person who committed crimes and other security problems, lack of system which will link the current evidences and past criminal records of the suspect for providing automation so that investigators

will allocate easily identify the suspected person among the other suspects and be able to provide to them fast emergence judgement to the innocent and arrest criminal before spending time being prisoned without guilty, and also make easier follow up to criminals records. On the suspects side there is a need to get judgement at real time.

Nowadays as technology is being improved so that it required system that will help RIB against mishap by fast crime scene evidences analysis. In addition, if you compare the time spent during the act of analysing the evidences between two systems (manual and automatic systems) you found out that there is a big difference and more sophisticated improvements of the automatic system than using manual one.

Proposed Solutions

Smart crime scene evidences analysis system will allow the RIB to investigate and analyse crime scene evidences for easy crimes follow up on criminals, it will also help the investigators in general to monitor criminal records and crime evidences at real time.

Functional Requirements

- ✓ The system must only allow a user to login by using a valid username and password.
- ✓ The user should be able to generate a report in pdf formats.
- ✓ The system should be able to assignee a new password once is needed.
- ✓ The system must be able to retrieve the information from the database.
- ✓ They system must be able to validate while entering the information.
- ✓ The system must be able to keep user logs.
- ✓ Primary suspected person must be presented within 48hours.
- ✓ Generate reports about crimes, cases and primary suspected person.

Non-functional requirements

- ✓ The system must be able to hide the user's information
- ✓ RIB station can generate its case reports
- ✓ Only RIB Headquarter can view the RIB stations that exist in Rwanda
- ✓ The System will block you once you click 5 times on login steps
- ✓ The system should be able to run on windows OS
- ✓ The system shall be able to protect the user's privacy
- ✓ The system shall has high availability
- ✓ The system shall not have unexpected downtime
- ✓ The system shall have downtime at most 3 hours/month
- ✓ The system shall provide an easy-to-use graphical interface
- ✓ The web interface should be intuitive and easily navigable Users should be able to understand the menu and options provided by crime reporting system.
- ✓ Any notification or error messages generated by crime reporting system shall be clear and polite.

CHAPTER 3

REQUIREMENTS ANALYSIS AND DESIGN OF THE NEW SYSTEM

Introduction

A strong foundation will always lead to a strong output; the same applies to the concept of system. Whenever you have a strong analysis and a clear design this is when you can be able to build a perfect and useful system. Actually, the main goal of a new system is to satisfy the needs of its users by solving problems they face with the existing system. Deep analysis of users' needs will most of the time lead to a useful software development as a system might give perfect result.

System development involves mainly two major components which are:

- System Analysis
- System Design

The system analysis is the process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. System analysis is conducted for the purpose of studying a system or its parts in order to identify its objectives. It is a problem-solving technique that improves the system and ensures that all components of the system work efficiently to accomplish their purpose. Analysis specifies what the system should do.

System design is a process of planning a new business system or replacing an existing system by defining its components or modules to satisfy the specific requirements. Before planning, you need to understand the old system thoroughly and determine how computers can best be used in order to operate efficiently and solve a specific issue as to meet type goal. System Design focuses on how to accomplish the objective of the system. System analysis and design mainly focuses on: Systems, Processes and Technology. (tutorialspoint, 2021)

In order to really analyse and design our new system, we judged very essential to understand the flow of data through that system.

Analysis and Design Methodology

Object Oriented Methodology (OOM)

There are many methodologies for the development of information systems such as Data Structure-Oriented, Object-Oriented, Prototyping, and so on. We shall be concerned here with Object Oriented.

Object Oriented Methodology (OOM) is a system development approach encouraging and facilitating re-use of software components. With this methodology, a computer system can be developed on a component basis which enables the effective re-use of existing components and facilitates the sharing of its components by other systems.**Invalid source specified.**

They are two main basic building blocks are useful in development of the system by using Object Oriented Methodology those are: Classes and Objects.

Object: refers to a particular instance of a class where the object can be a combination of variables, functions, and data structures. An Object can be a thing, a concept, or an event.

Class: is a description of a collection of objects with common attributes and behaviours. A class is divided in three parts as shown below:

Class Name
Attributes
Operations ()

- The upper part holds the name of the class.
- The middle part contains the attribute of the class.
- The last part gives the method or operation the class can take or undertake.

An attribute is a named property of a class that describes a range of values that instances of the property may hold. A method is the implementation of a service that can be requested from any object to the class to affect behaviour.

Unified Modelling Language (UML)

The Unified Modelling Language is a general-purpose, developmental, modelling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system. The language provides us with the notations to produce models and, is explicitly designed to be implemented by computer-assisted software engineering (CASE) tools.

Naturally, some aspects of methodology are implied by the elements that comprise a UML model, but UML itself just provides a visual syntax that we can use to construct models. It does not give us any kind of modelling methodology.

UML Models

A **model** a thing used as an example to follow or imitate. It is easier to refer on a model than to refer on the reality because the model represents just essential aspects of the reality and ignores the useless aspects.

The UML provides many different models for a system such as use case diagram, class diagram, and sequence diagram.

Actually, it is not required to make all the models for a system; instead, it is advised to model the ones that are to visualize sufficiently the system.

Analysis of the new system

The analysis phase answers the questions of who will use the system, what the system will do, and where and when it will be used. During this phase, the project inventor investigates any current system(s) which have been previously done in the second chapter, identifies improvement opportunities, and develops a concept for the new system.

Requirement Analysis

Requirements analysis, also called "*requirements* engineering", is the process of determining user expectations for a new or modified product. These features, called *requirements*, must be quantifiable, relevant and detailed. In software engineering, such *requirements* are often called functional specifications.

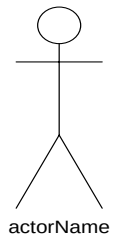
Design of the New System

Use Case Diagram

Use case diagram describes the functionality provided by a system in terms of actors, their goals represented as use cases, and relationships between actors and use cases. The followings are components of a use case model:

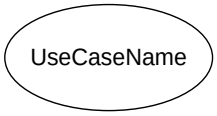
- **Actor**

An actor specifies a role that some external entity adopts when interacting with a system directly. It may represent a user role or a role played by another system that touches a system.



- **Use Case**

A use case is a list of actions or event steps, typically defining the interactions between a role (known in the Unified Modelling Language as an actor) and a system, to achieve a goal and is represented as follows:



- **Relationship**

Meaningful relationships between actors and use cases which is a UML association symbol.

- **System boundary**

It is a box drawn around the use case to denote the edge or boundary of the system being modelled.

The figure below describes the operations of new system and the stakeholders through the use case

Figure 3: Use Case Diagram

Use case description

Use Case description details what a use case do, and what it requests in order to be well executed. Each use case looks like this:

- **Name:** a name of a use case
- **Description:** what a system intends to do
- **Actor:** the actor involved in the use case
- **Pre-condition:** the system state before the use case can begin
- **Post-condition:** the system state when the use case is over
- **Normal flow**: the actual steps of the use case
- **Alternative flow:** steps which may happen in case a normal flow fails.

1. **Use Case description for Create Account**

Name: Create User Account
Actor: RIB Headquarter
Description: This use case describes how the actor creates account for helping them to log in into the system.
Pre-condition: None
Post-condition: If the use case is successful the user account is created and he/she logs into the system, if not the system state does not change.
Normal Flow: 1. The use case asks the actor to create account if it's for the first time to use the system and if not, log in into the system with his/her username and password 2. The system checks the information provided if true the account is created. Then the actor is able to fill the username and password.

3. System validates names and password, and if finds correct allow the actor to log into the system.
Alternative flow: • 3. a. if the information is not valid the system shows the error message. • 3. b. The actor can decide to go back to the beginning of the main flow or cancel the login, at that point the use case ends.

Table 1 : Create Account

2. **Use Case description for Add Case**

Name: Add Case
Actor: RIB Station
Description: This use case describes how the actors will add all information regarding new cases this includes saving them
Pre-condition: Actors must be logged in
Post-condition: If the use case is successful, the case is saved
Normal Flow: 1. Actors requests a form to add a case. 2. System provides the form to fill the case information. 3. The Actor saves case.
Alternative flow: • If the case information is not valid the system shows the error where is not correct.

Table 2 : Add Case

3. **Use Case description for Add suspect**

Name: Add suspect
Actor: Station Officer
Description: This use case describes how the actors will add all information regarding new suspect this includes saving them
Pre-condition: Actors must be logged in
Post-condition: If the use case is successful, the suspect is saved

Normal flow: 1. Actors requests a form to add a suspect. 2. System provides the form to fill the suspect information. 3. The Actor saves a suspect.
Alternative flow: • If the case information is not valid the system shows the error where is not correct.

Table 3 : Add Suspect

4. **Use Case description for Add evidences**

Name: Add evidences
Actor: Station Officer
Description: This use case describes how the actors will add all information regarding evidences this includes saving them
Pre-condition: Actors must be logged in
Post-condition: If the use case is successful, the evidences are saved
Normal flow: 1. Actors requests a form to add new evidences. 2. System provides the form to fill the evidences information. 3. The Actor saves evidences.
Alternative flow: • If the information is not valid the system shows the error where is not correct.

Table 4 : Add evidence

5. **Use Case description for View suspect**

Name: View suspect
Actor: RIB Station, Station Officer, RIB Headquarter
Description: This use case describes how the actors will view all information regarding all case suspect
Pre-condition: Actors must be logged in
Post-condition: you should have case registered
Normal flow: 1. Actors requests a form to view the suspects of a certain case. 2. System asks the user to select a certain case to view its suspects. 3. The Actor can view the list of suspect.
Alternative flow: • If there is no case exists the system shows an error.

Table 5: View Suspect

6. **Use Case description for Manage RIB Stations**

Name: Manage RIB Stations
Actor: RIB Headquarter
Description: This use case describes how the RIB Headquarter will manage all RIB Stations. This includes adding, updating, and deleting from the system.
Pre-condition: RIB Headquarter must be logged in

Post-condition: If the use case is successful, the RIB Station is added, updated or deleted from the system.

Normal Flow:

1. This use case starts when admin wishes to add, update and/or delete the RIB Station from the system.
2. The system requests the RIB Headquarter to specify the function he/she would like to perform i.e. Add, Update, and Delete RIB Station. Once the RIB Headquarter provides the required information, one of the sub-flows is executed:

– Add RIB Station: The system requests the RIB Headquarter to enter the station information, once RIB Headquarter provides the requested information, the system saves the information and the RIB Station is added.

– Update RIB Station: The system requests the RIB Headquarter to select from the list the RIB Stations to be updated. RIB Headquarter selects RIB Station, the system retrieves and displays the RIB station information, RIB Headquarter makes the changes, and record is updated.

– Delete RIB Station: Select RIB Station, alter this, system retrieves and displays RIB Station, system prompts the RIB Headquarter to confirm the deletion, RIB Headquarter verifies the deletion, if confirmed the deletion is executed.

Alternative flow:

1. RIB Station not found, if in any sub-flow, station information not found, error message is displayed. The RIB Headquarter may enter a different name or cancel the case ends here.
2. Update Cancelled, if in the update station details sub-flow, the RIB Headquarter decides not to update the station information, the update is cancelled and the basic flow is restarted at the beginning.
3. Delete Cancellation, if in the delete station information sub-flow, the RIB Headquarter decides not to delete the station information, the delete is cancelled and the basic flow is restarted at the beginning.

Table 6: Manage RIB Station

7. **Use Case description for View criminal record**

Name: view criminal record
Actor: RIB Station, Station Officer
Description: This use case describes how the RIB Station will manage to know the past criminal records of the suspect. This includes adding, updating his/her records from the system.
Pre-condition: Actors must be logged in
Post-condition: If the use case is successful, the criminal record information is added, updated from the system.
Normal Flow: 1. This use case starts when RIB Station wishes to add, update the suspect criminal records from the system. 2. The system requests the RIB Station to specify the function he/she would like to perform i.e. Add, Update. Once the RIB Station provides the required information, one of the sub-flows is executed:

Table 7: View criminal record

8. **Use Case description for View Primary suspect**

Use case: View primary suspect
Actor: Station User, RIB Station, RIB Headquarter
Description: it allows the Actors to view the primary suspect among other suspects
Precondition(s): Actors should have logged on system.
Post condition(s): you should have a valid crime case, suspect and evidences

Normal flow: • Actors requests a form to view the primary suspected person among other suspects. • System will analyses evidences and based on past criminal records of the suspects, it will identify the guilty suspect • The Actor can view the full information of suspect and his/her criminal record.

Table 8: View Primary suspect

9. Use Case description for Add Questions/Answers

Name: Add Questions/Answers
Actor: RIB Headquarter
Description: allows RIB Headquarter to create questions to be asked suspects and reporters and their predictable answers to the RIB Station when investigating
Pre-condition: RIB Headquarter user must be logged in
Post-condition: RIB Headquarter should get a message "Question/Answer has been sent"
Normal flow: 1. RIB Headquarter requests a form to add question/Answer. 2. System provides the question/Answer form. 3. RIB Headquarter user fills and sends the question/answer to RIB Stations. 4. The RIB Station receives the questions/Answer.
Alternative flow: • If the information is not valid the system shows the error where is not correct.

Table 9: Add Question/Answer

10. Use Case description for View RIB Station reports

Use case: View RIB Stations reports
Actor: RIB Headquarter
Description: it allows the RIB Headquarter to view reports from all RIB stations
Precondition(s): RIB Headquarter should have logged on system.
Post condition(s): Station's Reports are viewed
Normal flow: • RIB Headquarter office must log into the system • RIB Headquarter can views the list of all reports in all RIB stations. • The RIB Headquarter saves all station's reports.

Table 10: View RIB Station Reports

11. Case description for Generate Reports

Name: Generate Reports
Actor: RIB Station
Description: This use case describes how the RIB Station will generate Station Report.
Pre-condition: RIB Station must be logged in.
Post-condition: If the use case is successful, the reports will be generated.
Normal Flow: 1. The RIB Station is the one who has the privilege of downloading the reports. So after the viewing primary suspected person details are being resulted the RIB Station will choose the option of generating reports. 2. Choose the type of Report to be downloaded. 3. Download the Report in pdf.

Alternative flow: If information is not well insert it will be impossible of generating the reports and the warning message will be displayed.

Table 11: Generate Reports

Class diagram

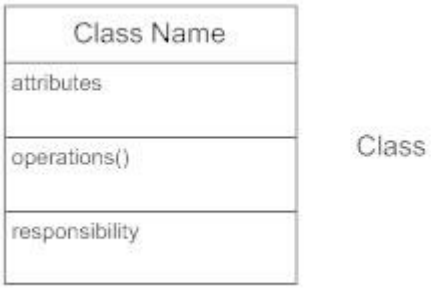
A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods) and the relationships among objects. Class diagram shows a set of classes, interfaces and collaborations, and their relationships; addresses static design view of a system. (smartdraw, 2021)

The followings are components of a class diagram:

Classes

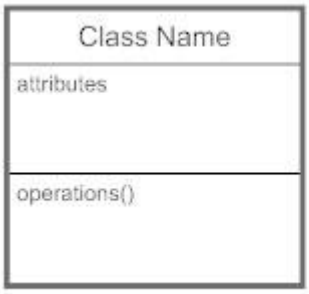
Classes represent an abstraction of entities with common characteristics. Associations represent the relationships between classes.

Illustrate classes with rectangles divided into compartments. Place the name of the class in the first partition (centred, bolded, and capitalized), list the attributes in the second partition (left-aligned, not bolded, and lowercase), and write operations into the third.



Active Classes

Active classes initiate and control the flow of activity, while passive classes store data and serve other classes. Illustrate active classes with a thicker border.



Active class

Visibility

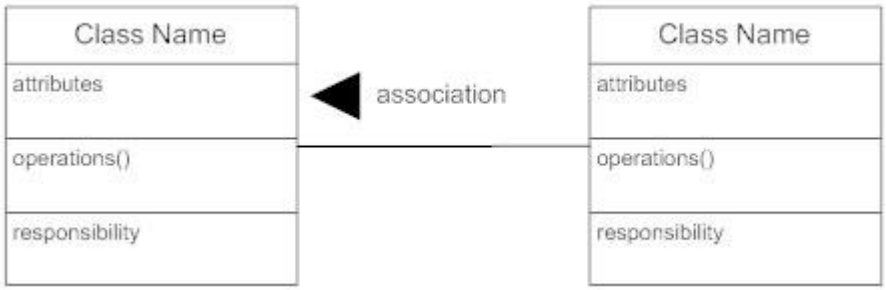
Use visibility markers to signify who can access the information contained within a class. Private visibility, denoted with a - sign, hides information from anything outside the class partition. Public visibility, denoted with a + sign, allows all other classes to view the marked information. Protected visibility, denoted with a # sign, allows child classes to access information they inherited from a parent class.

Visibility

Marker	Visibility
+	public
-	private
#	protected
~	package

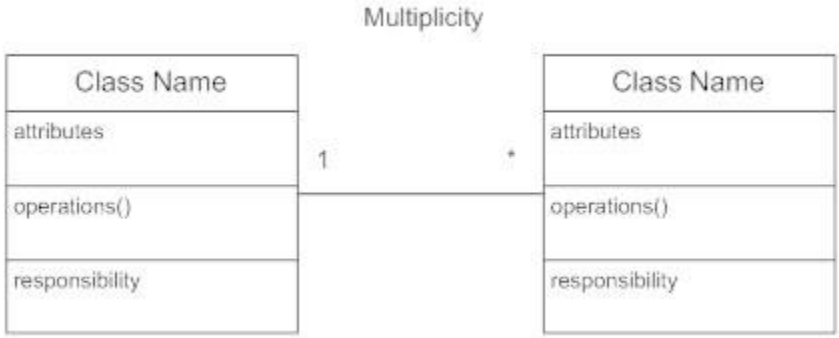
Associations

Associations represent static relationships between classes. Place association names above, on, or below the association line. Use a filled arrow to indicate the direction of the relationship. Place roles near the end of an association. Roles represent the way the two classes see each other.



Multiplicity (Cardinality)

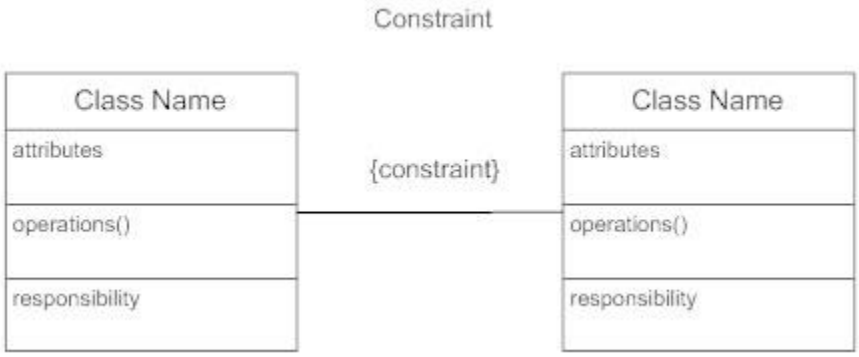
Place multiplicity notations near the ends of an association. These symbols indicate the number of instances of one class linked to one instance of the other class.



Indicator	Meaning
0..1	Zero or one
1	One only
0..*	0 or more
1..* *	1 or more
n	Only n (where n > 1)
0..n	Zero to n (where n >1)
1..n	One to n (where n > 1)

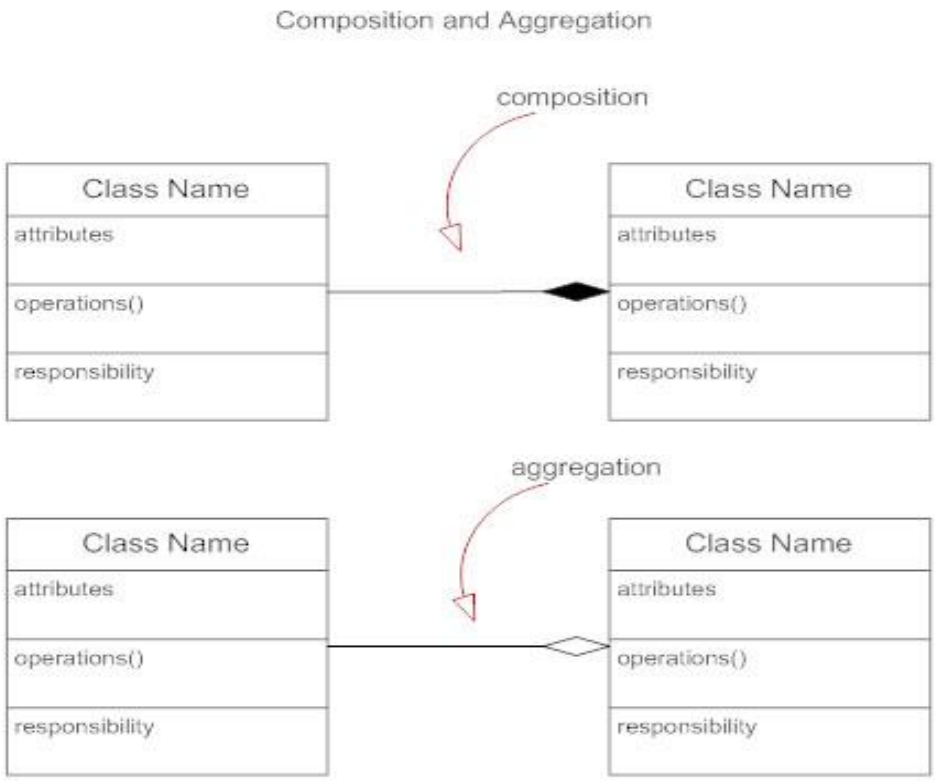
Constraint

Place constraints inside curly braces {}.



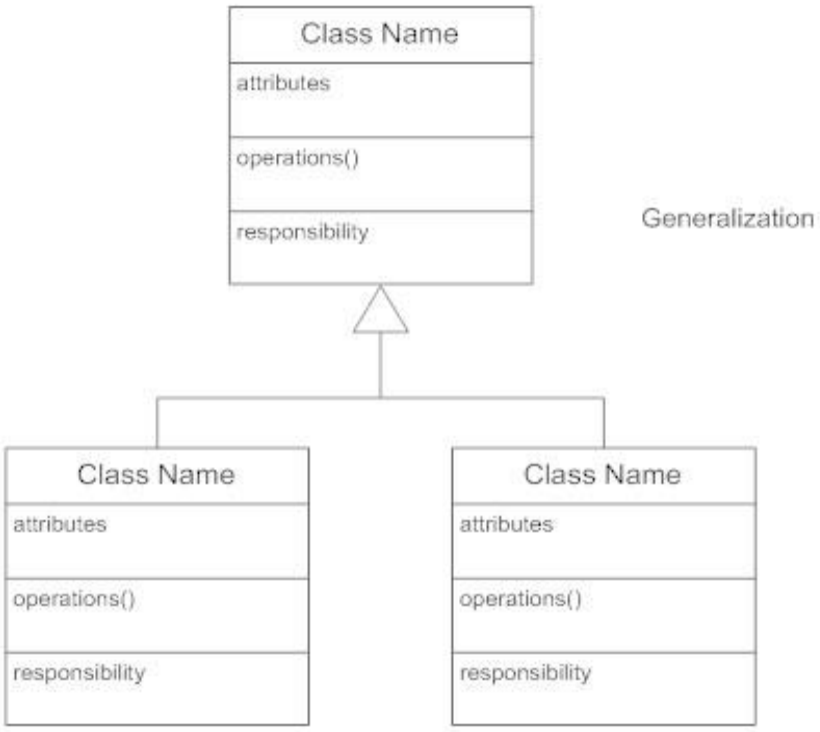
Composition and Aggregation

Composition is a special type of aggregation that denotes a strong ownership between Class A, the whole, and Class B, its part. Illustrate composition with a filled diamond. Use a hollow diamond to represent a simple aggregation relationship, in which the "whole" class plays a more important role than the "part" class, but the two classes are not dependent on each other. The diamond ends in both composition and aggregation relationships point toward the "whole" class (i.e., the aggregation).



Generalization

Generalization is another name for inheritance or an "is a" relationship. It refers to a relationship between two classes where one class is a specialized version of another.



The diagram below shows the class diagram of the new system

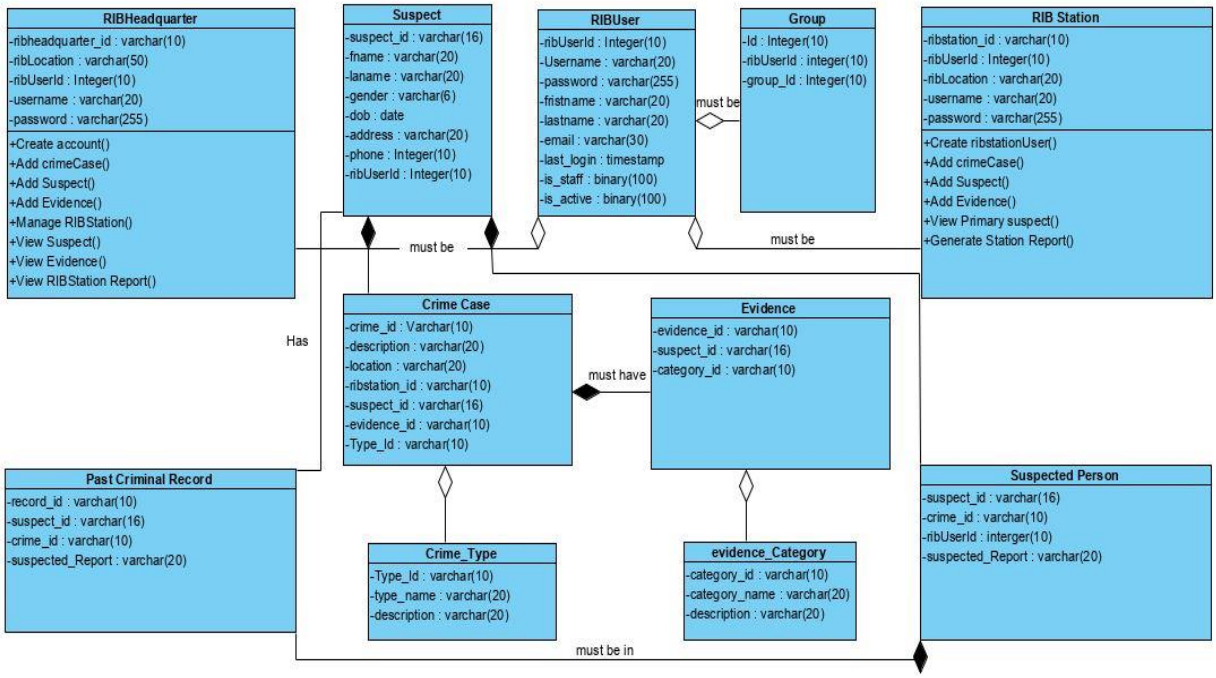


Figure 4: Class Diagram

Sequence diagram

Sequence Diagrams is an interaction diagram that details how operations are carried out: what messages are sent and when sequence diagrams are organized according to time. The time progresses as you go down the page (Wesley A, 2004)

The objects involved in the operation are listed from left to right according to when they take part in the message sequence. Each vertical dotted line is lifeline, representing the time that an object exists. Each arrow is a message on the receiver's lifeline. The activation bar represents the duration of execution of the message. Sequence diagram is the most common kind of interaction diagram, which focuses on the message interchange between a numbers of lifelines.

Sequence diagram describes an interaction by focusing on the sequence of the message that is exchanged, along with their corresponding occurrence specification on the lifelines.

Term and definition	**Symbol**
An actor: ✓ It can be a person or system that derives benefit from and is external to the system. ✓ It participates in a sequence by sending or receiving messages. ✓ It is placed across the top of the diagram.	
An object lifeline: ✓ It participates in a sequence by sending or receiving messages. ✓ It is placed across the top of the diagram.	
An activation: ✓ It is a long narrow rectangle placed on top of a lifeline. ✓ It denotes when an object is sending or receiving messages	

A message: ✓ It conveys information from one object to another one. ✓ An operation call is labeled with the message being sent and a solid arrow, whereas a return is labeled with the value being returned and shown as a dashed arrow.	: Message send :asynchronous : Message return Return Message : Message call

Diagram for Sequence Diagrams

Login request

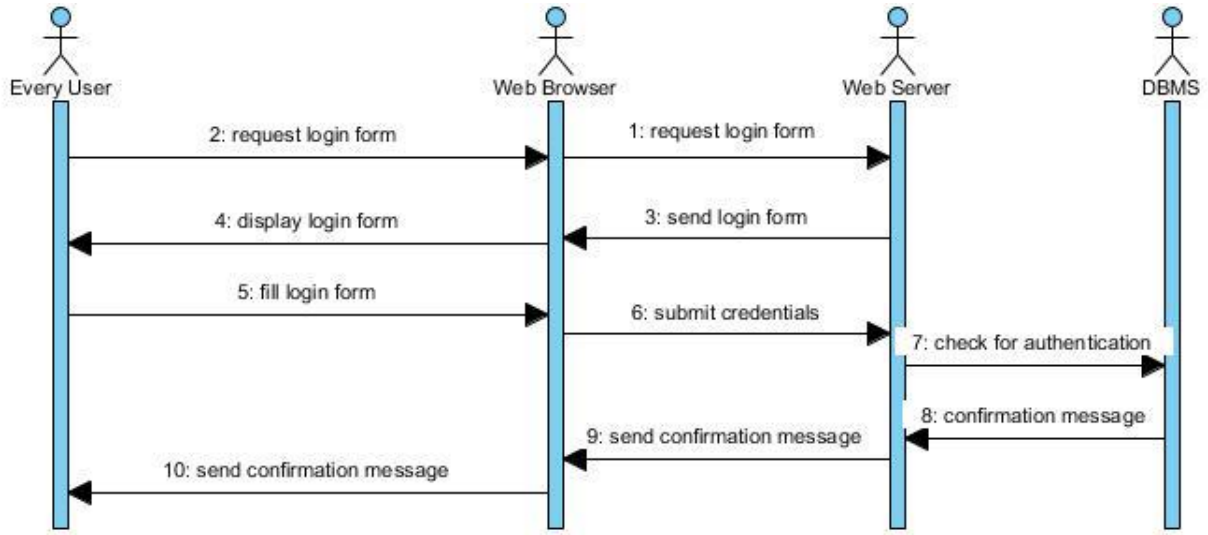


Figure 5: Sequence diagram for login request

Create user Account

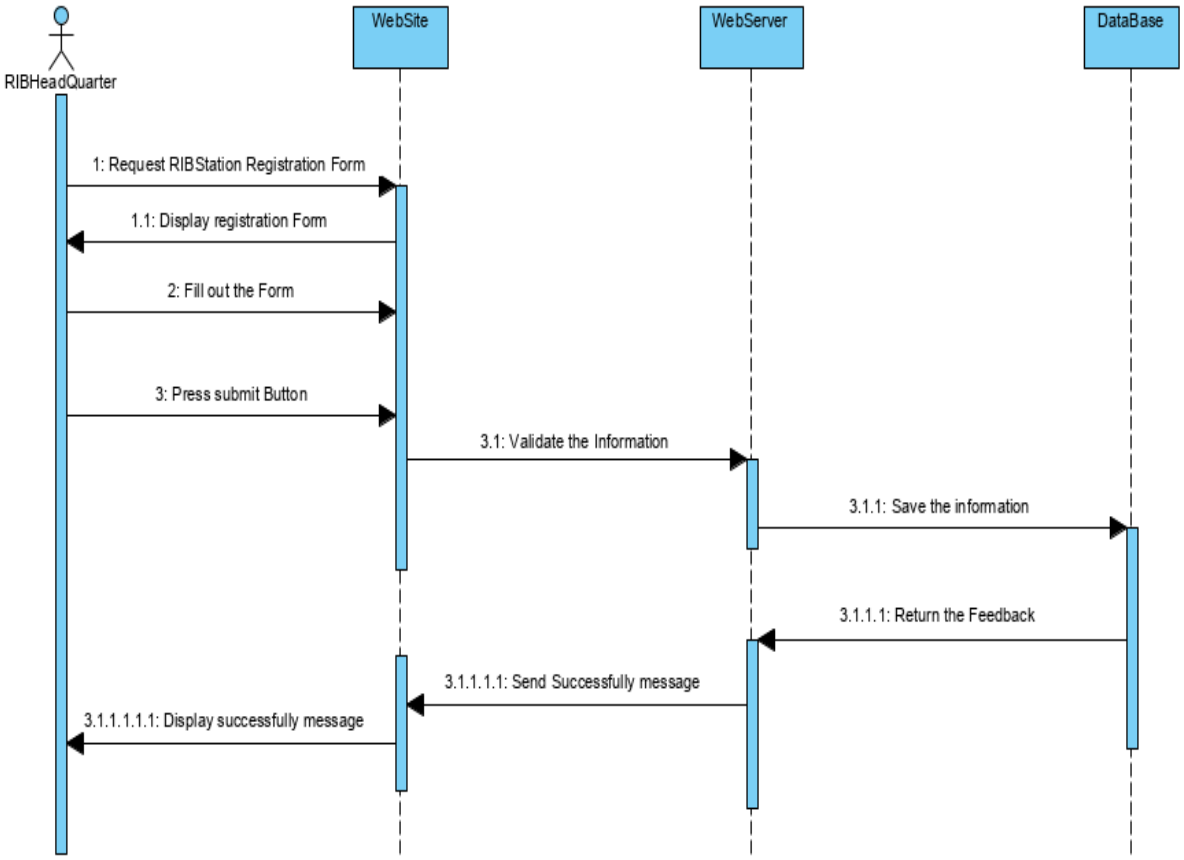


Figure 6 : Sequence Diagram to Create user Account

Add case

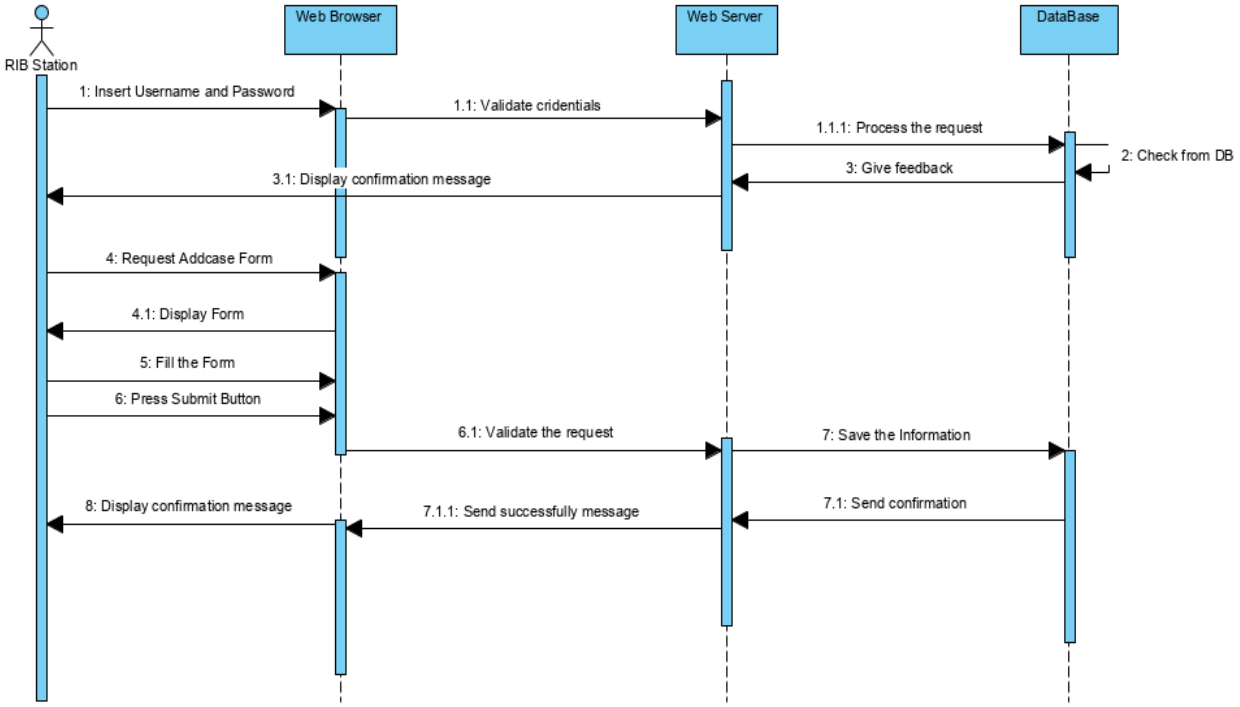


Figure 7 : Sequence Diagram to Add case

Add Suspect

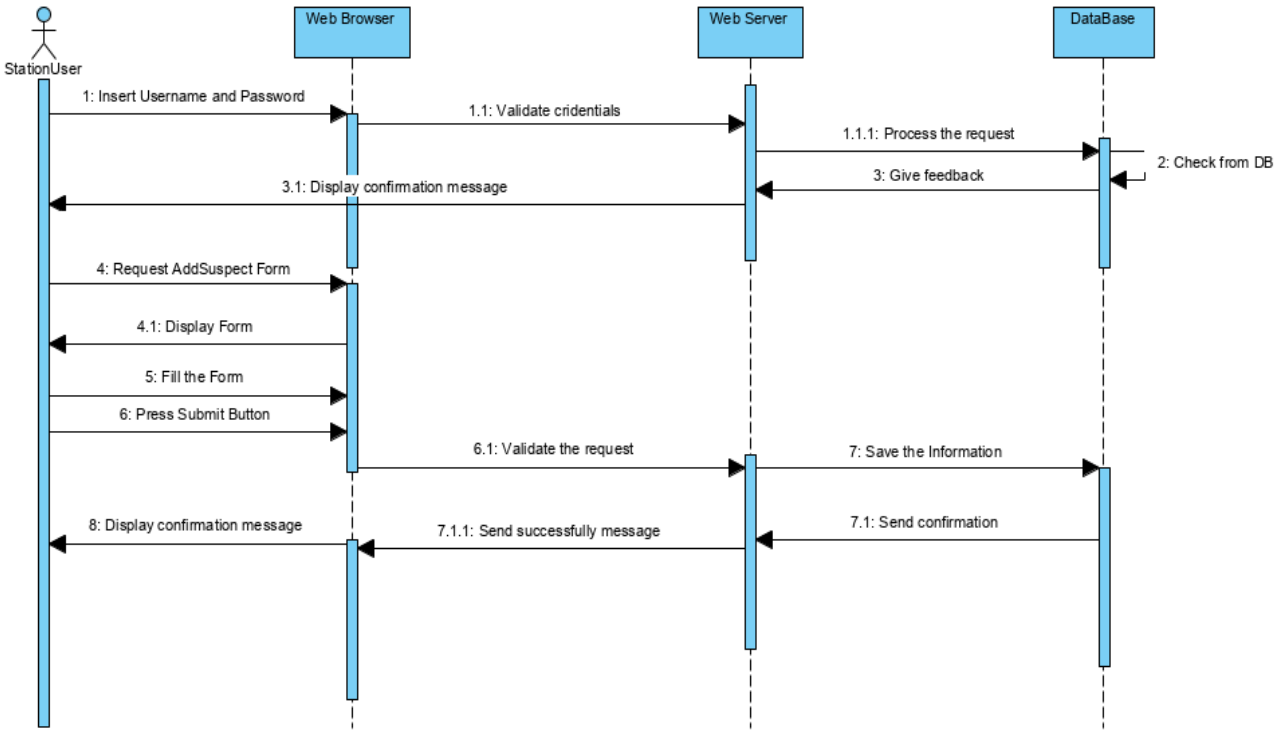


Figure 8 : Sequence Diagram to Add suspect

Add evidence

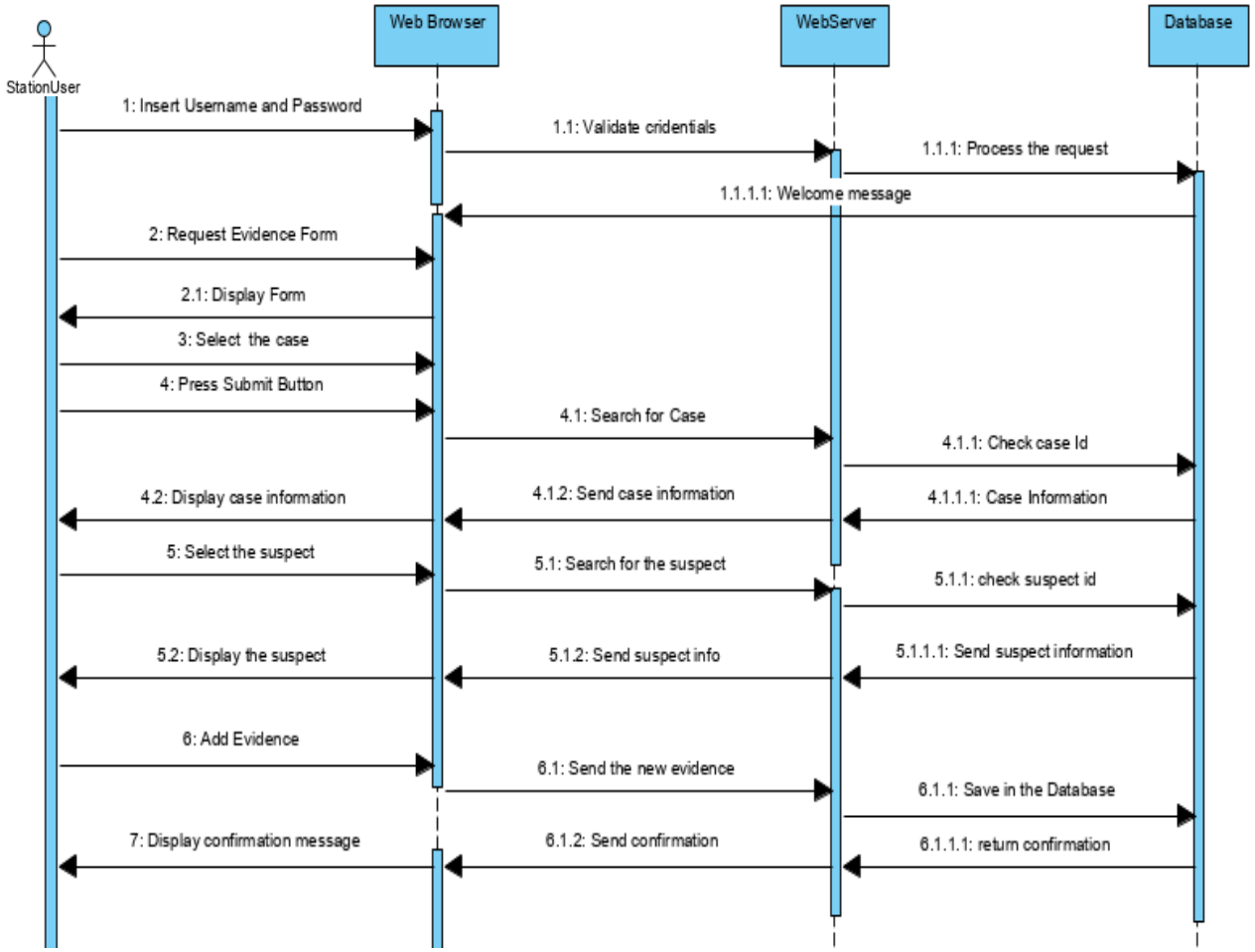


Figure 9: Sequence Diagram to Add evidence

View Suspect

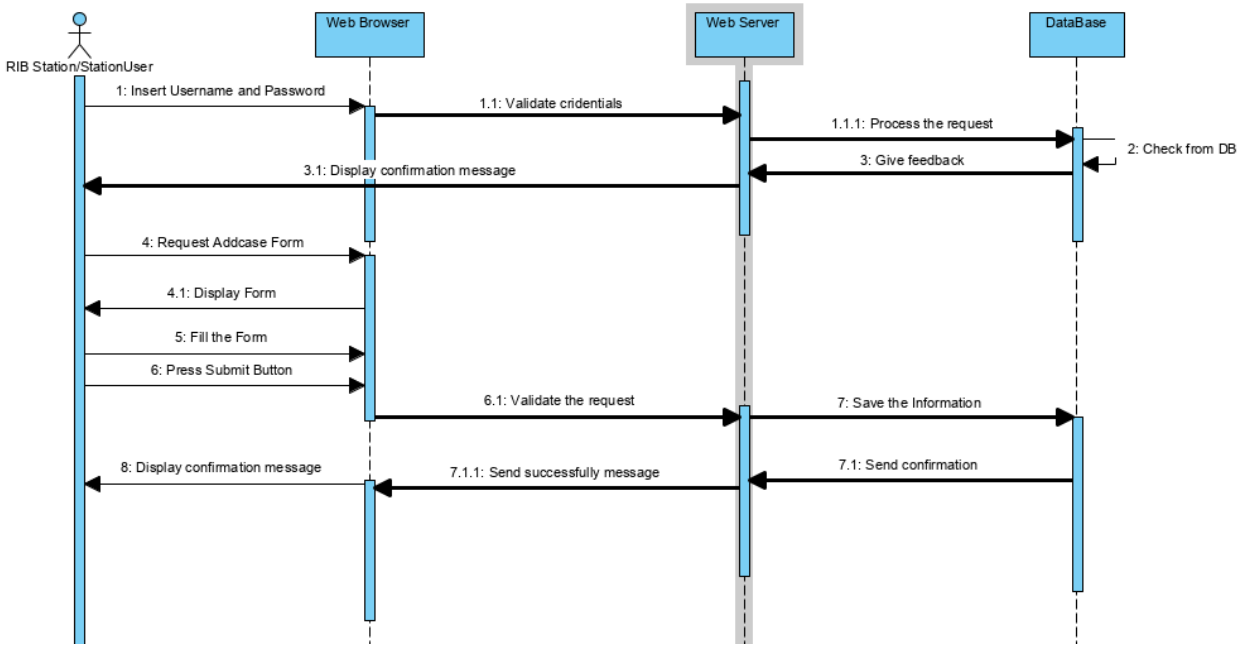


Figure 10 : Sequence Diagram to View Suspect

Manage RIB Station

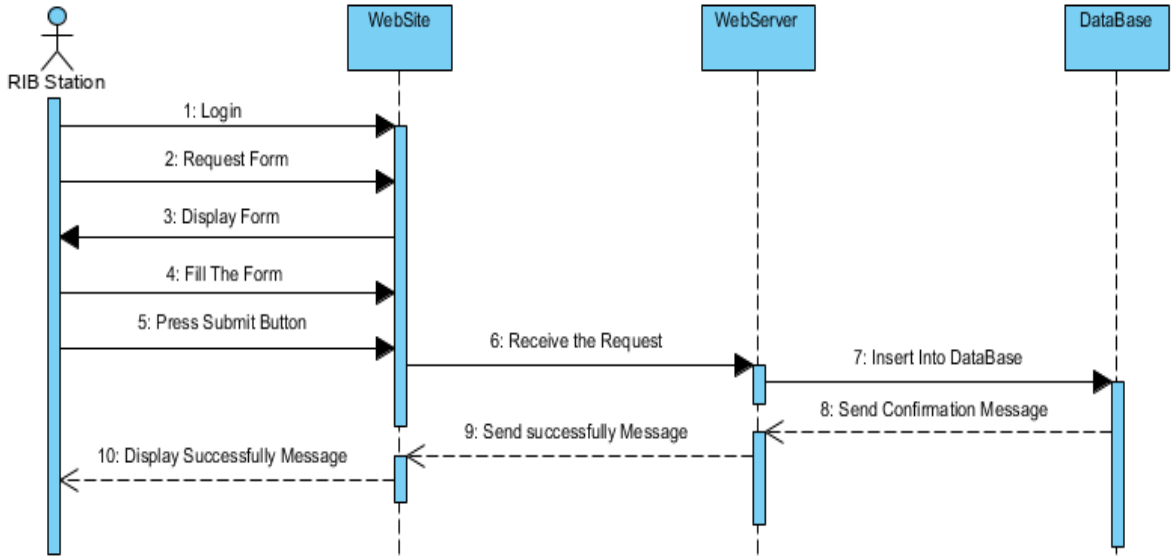


Figure 11 : Sequence Diagram to Manage RIB Station

View primary suspect

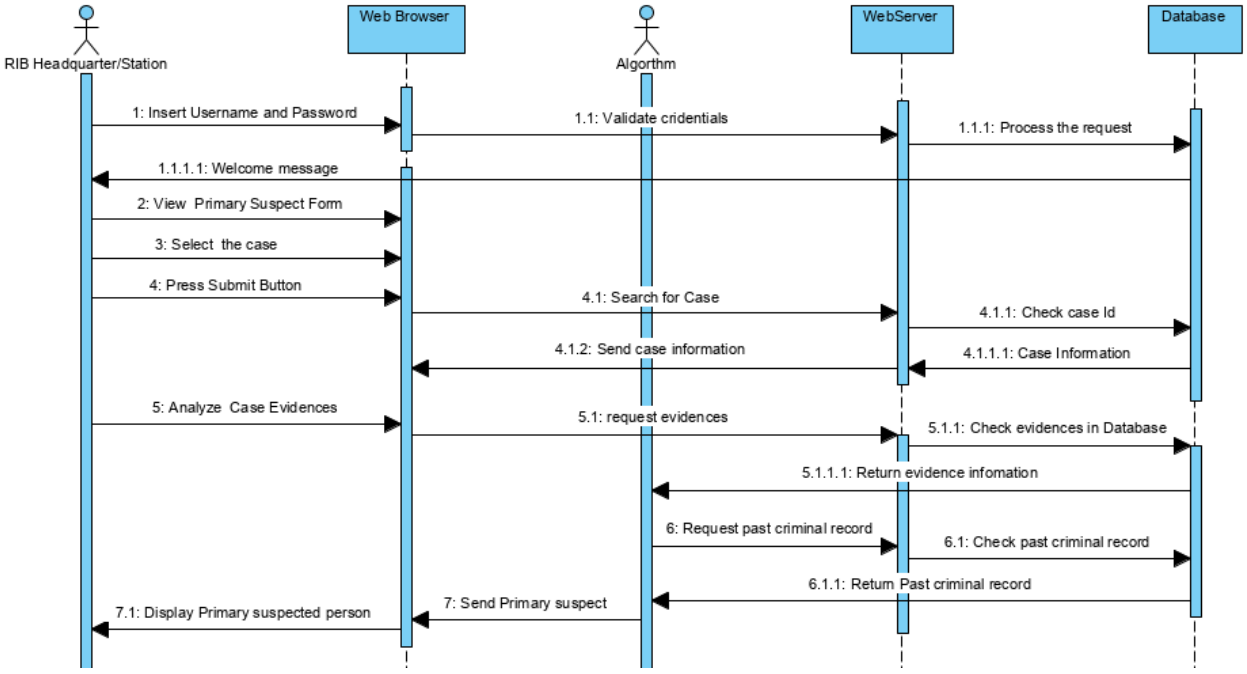


Figure 12 : Sequence Diagram to View primary suspect

RIB Station Reports

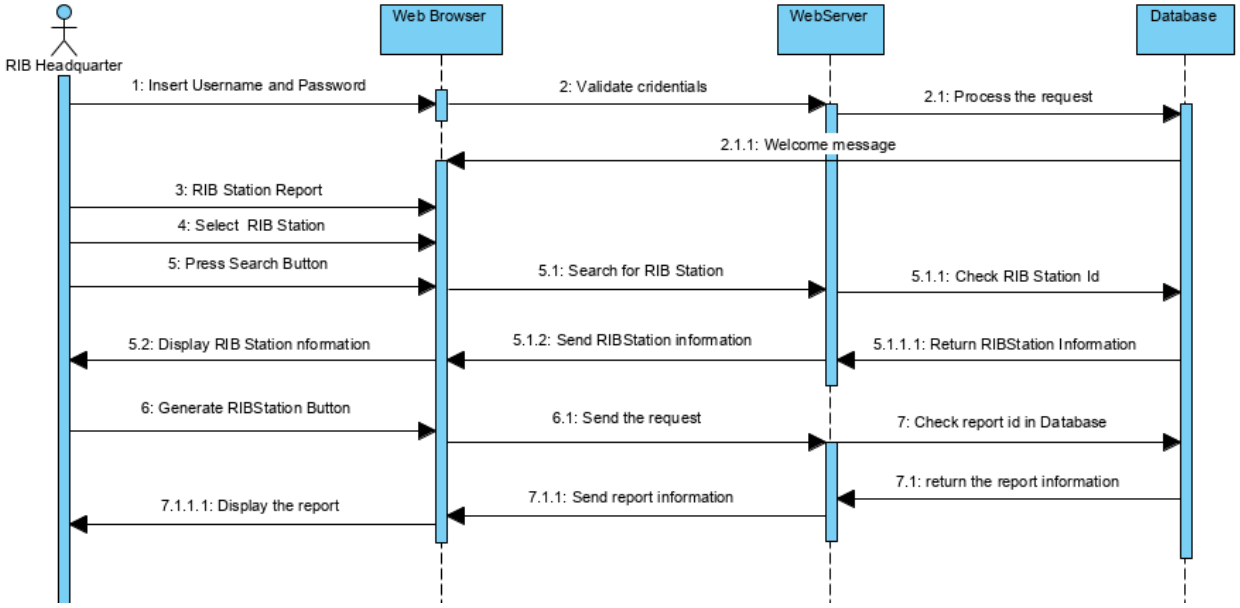


Figure 13 : Sequence Diagram to RIB Station Reports

Generate Reports

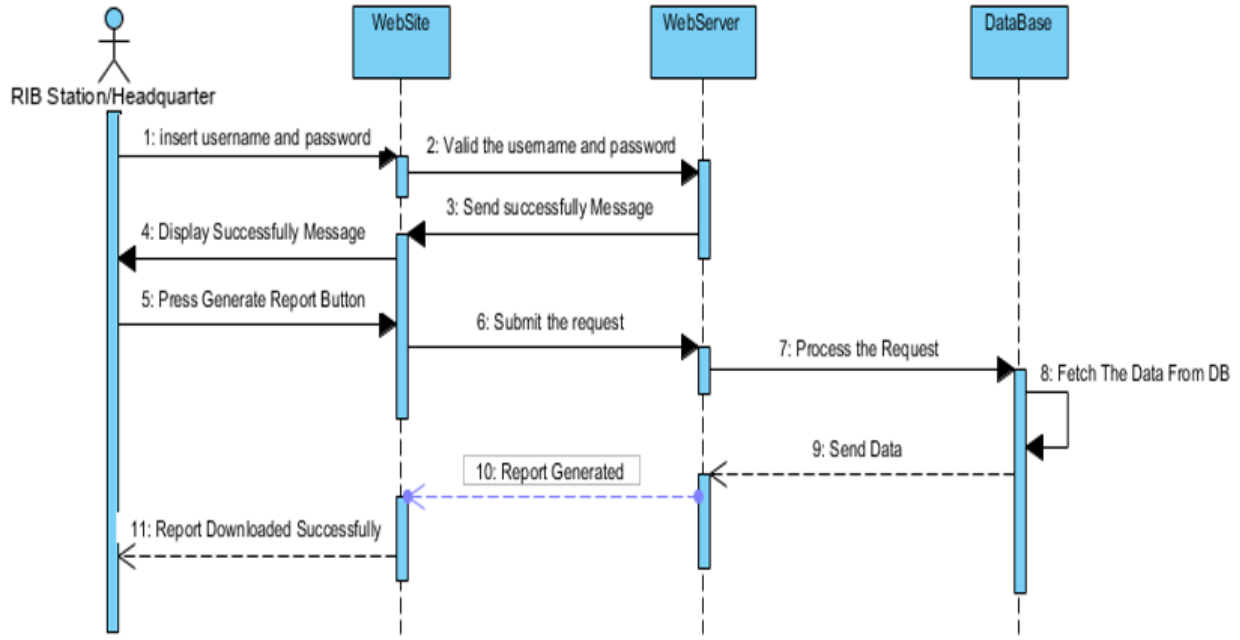


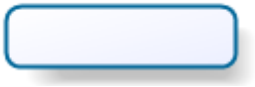
Figure 14 : Sequence Diagram to Generate Report

Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows). Activity diagrams show the overall flow of control.

Activity diagrams are constructed from a limited number of shapes, connected with arrows.

The most important shape types:



A task is an atomic activity that is included within the process

Exclusive gate ways (decisions) are locations within a business process where the sequence can take two or more alternative paths

Bars represent the start (split) or end (join) of concurrent activities

The start event indicates where a particular process will start.

The End event indicates where a particular process will End.

Activity diagram for sending request and getting response

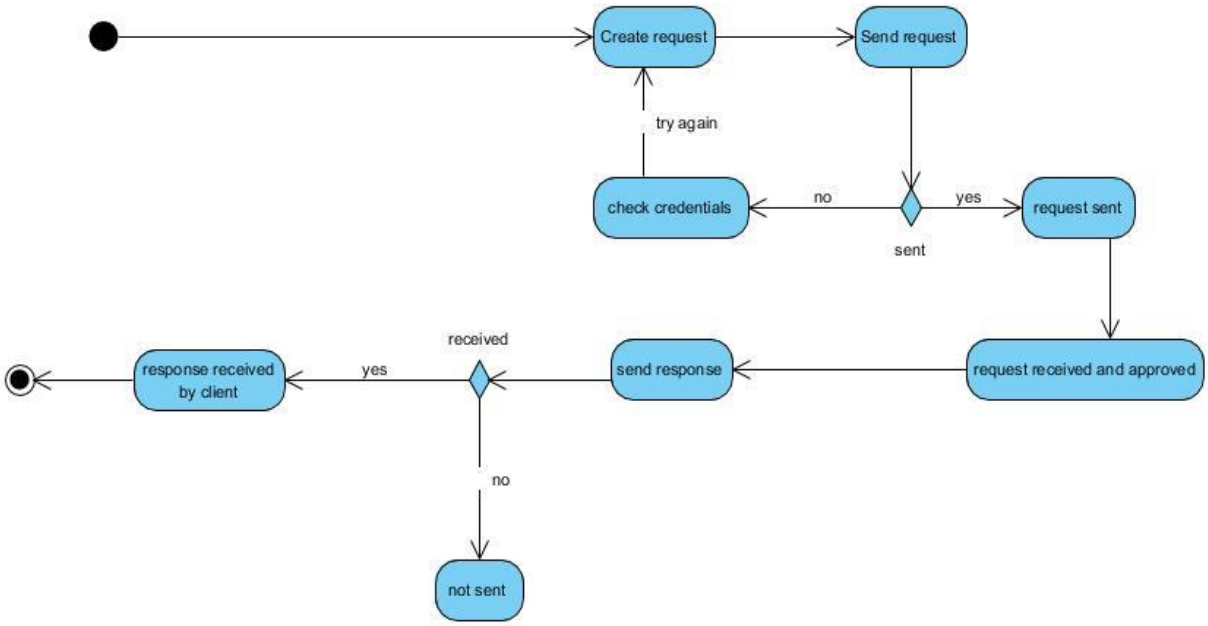


Figure 15 : Send request and getting response Activity Diagram

Activity diagram for login

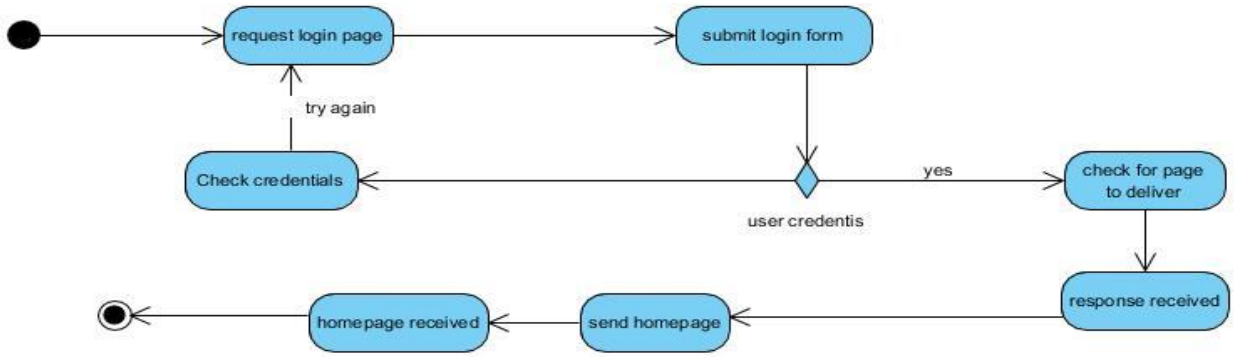


Figure 16 : Login activity diagram

Activity diagram for RIB Headquarter, RIB Station user Registration

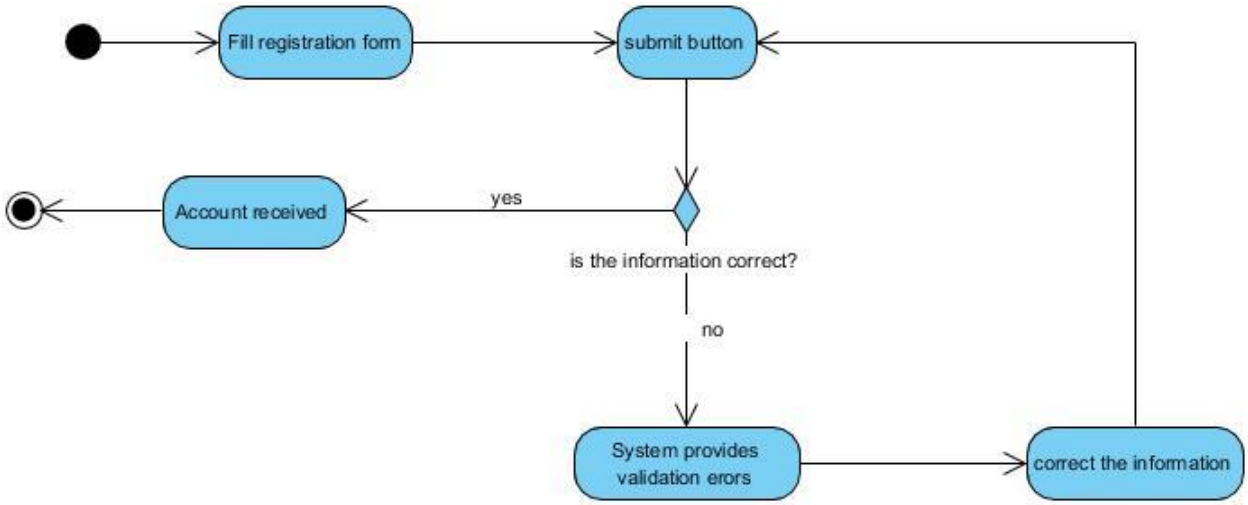


Figure 17 : User registration activity diagram

Database schema diagram

Database schema diagram is a set of collection of information that is organized so that it can easily be accessed, managed, and updated. Data base Management System (DBMS) are referred to as database software tools which are primarily used for storing, modifying, extracting, and searching for information within database.

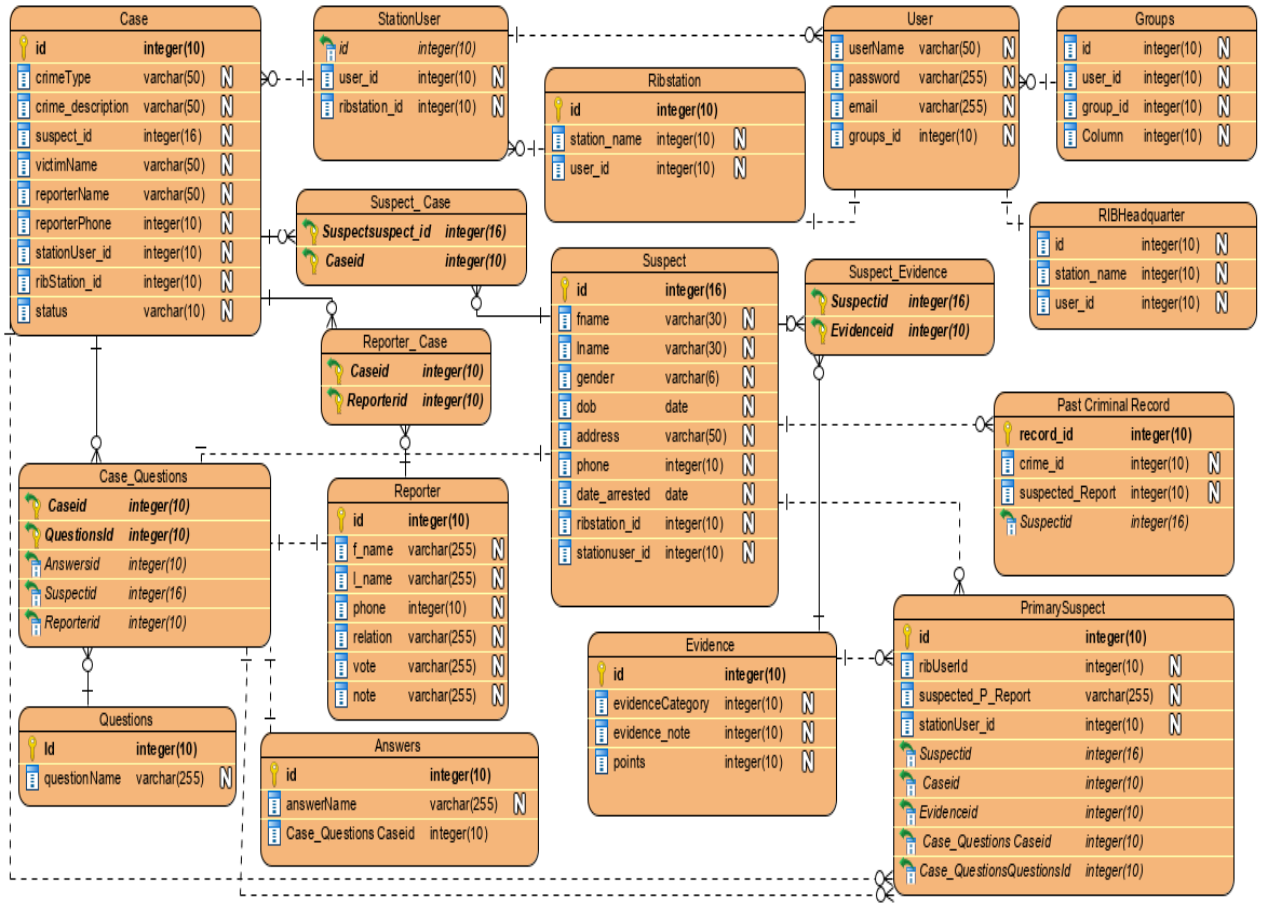


Figure 18 : Database Schema

Information design

This may be as simple as designing a set of hyper-linked Web page, this include a database with user table, district table, sector table, school table etc. the set of tables is created using relational database for the identifies the entities. The uniqueness of the data fields is these tables are established using primary keys, while the relationships are maintained using foreign keys (Elmasri R, 1994).

User table

Name	Data type	Length/Precision	Scale	Not NULL?	Primary k
id	integer			Yes	Yes
password	character varying	128		Yes	No
last_login	timestamp with time zone			No	No
is_superuser	boolean			Yes	No
username	character varying	150		Yes	No
first_name	character varying	150		Yes	No
last_name	character varying	150		Yes	No
email	character varying	254		Yes	No
is_staff	boolean			Yes	No
is_active	boolean			Yes	No

Table 12 : Group table

Group table

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer			Yes	Yes
user_id	integer			Yes	No
group_id	integer			Yes	No

Table 13 : Group table

RIB Headquarter Table

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
		id	integer			Yes	Yes
		station_name	character varying	30		Yes	No
		province_id	integer			Yes	No
		user_id	integer			No	No

Table 14 : RIB Headquarter table

RIB Station Table

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
		id	integer			Yes	Yes
		station_name	character varying	30		Yes	No
		province_id	integer			Yes	No
		user_id	integer			No	No

Table 15 : RIB Station table

Station User Table

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer			Yes	Yes
officerNationalId	character varying	16		Yes	No
f_name	character varying	30		Yes	No
l_name	character varying	30		Yes	No
gender	character varying	1		Yes	No
phone	character varying	10		Yes	No
email	character varying	50		Yes	No
rank	character varying	30		Yes	No
username	character varying	50		Yes	No
password	character varying	5		Yes	No
station_id	integer			Yes	No
user_id	integer			No	No

Table 16 : Station user table

Officer Table

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer			Yes	Yes
OfficerNationalId	character varying	16		Yes	No
f_name	character varying	30		Yes	No
l_name	character varying	30		Yes	No
gender	character varying	1		Yes	No
phone	character varying	10		Yes	No
email	character varying	50		Yes	No
officerimage	character varying	154		No	No
rank	character varying	30		Yes	No
recruit_year	integer			Yes	No
user_id	integer			No	No

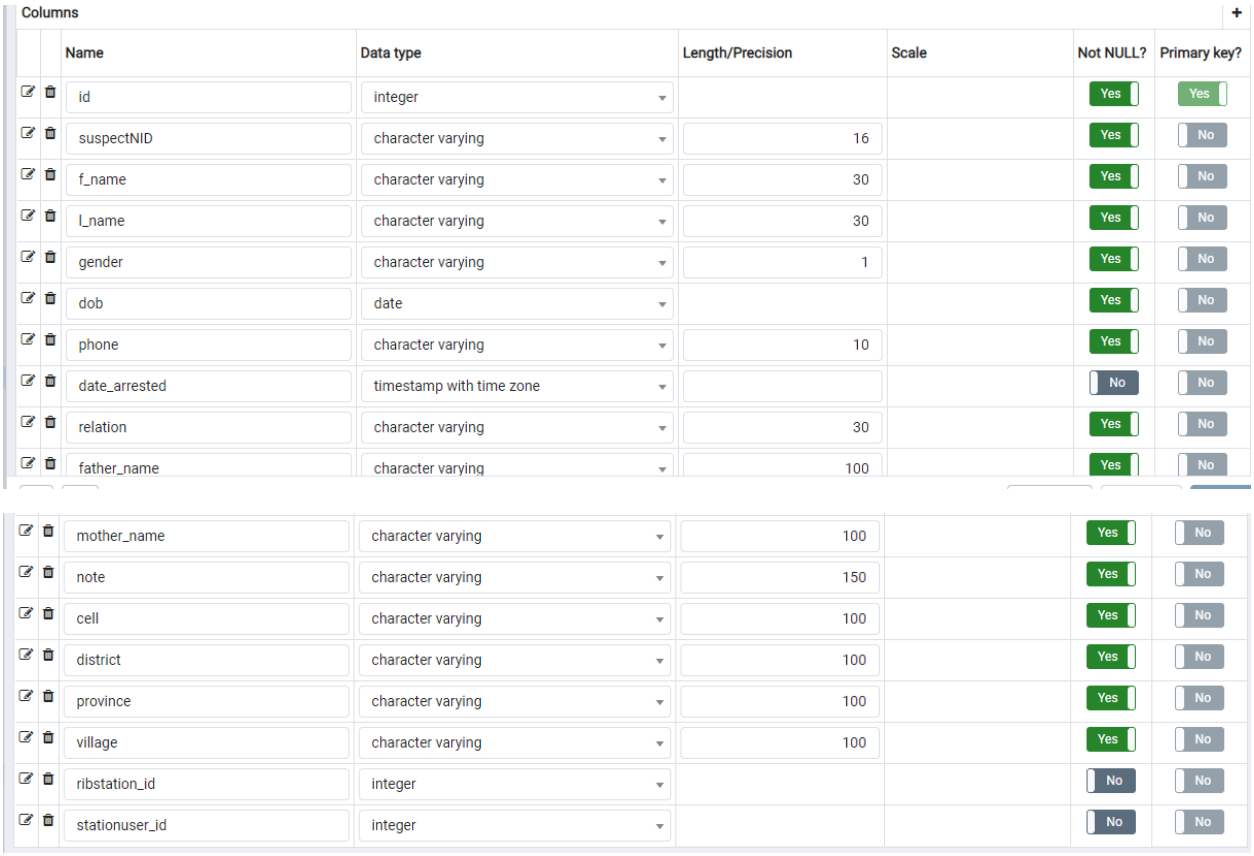
Table 17 : Officer Table

Case Table

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer			Yes	Yes
case_name	character varying	7		Yes	No
crimeType	character varying	15		Yes	No
case_desc	text			Yes	No
status	character varying	15		Yes	No
reporter_name	character varying	90		No	No
reporter_phone	character varying	10		No	No
victim_address	character varying	90		No	No
victim_name	character varying	90		No	No
ribstation_id	integer			No	No
stationuser_id	integer			No	No

Table 18 : Case table

Suspect Table



Columns

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
✎	🗑	id	integer			Yes	Yes
✎	🗑	suspectNID	character varying	16		Yes	No
✎	🗑	f_name	character varying	30		Yes	No
✎	🗑	l_name	character varying	30		Yes	No
✎	🗑	gender	character varying	1		Yes	No
✎	🗑	dob	date			Yes	No
✎	🗑	phone	character varying	10		Yes	No
✎	🗑	date_arrested	timestamp with time zone			No	No
✎	🗑	relation	character varying	30		Yes	No
✎	🗑	father_name	character varying	100		Yes	No
✎	🗑	mother_name	character varying	100		Yes	No
✎	🗑	note	character varying	150		Yes	No
✎	🗑	cell	character varying	100		Yes	No
✎	🗑	district	character varying	100		Yes	No
✎	🗑	province	character varying	100		Yes	No
✎	🗑	village	character varying	100		Yes	No
✎	🗑	ribstation_id	integer			No	No
✎	🗑	stationuser_id	integer			No	No

Table 19 : Suspect table

Evidence Table

Columns +

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
✎	🗑	id	integer			Yes	Yes
✎	🗑	evidenceCategory	character varying	15		Yes	No
✎	🗑	evidence_note	character varying	255		Yes	No
✎	🗑	points	character varying	10		Yes	No
✎	🗑	officerimage	character varying	154		No	No

Table 20 : Evidence table

User Table

Columns +

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
	id	integer			Yes	Yes
	password	character varying	128		Yes	No
	last_login	timestamp with time zone			No	No
	is_superuser	boolean			Yes	No
	username	character varying	150		Yes	No
	first_name	character varying	150		Yes	No
	last_name	character varying	150		Yes	No
	email	character varying	254		Yes	No
	is_staff	boolean			Yes	No
	is_active	boolean			Yes	No
	date_joined	timestamp with time zone			Yes	No

Table 21 : User table

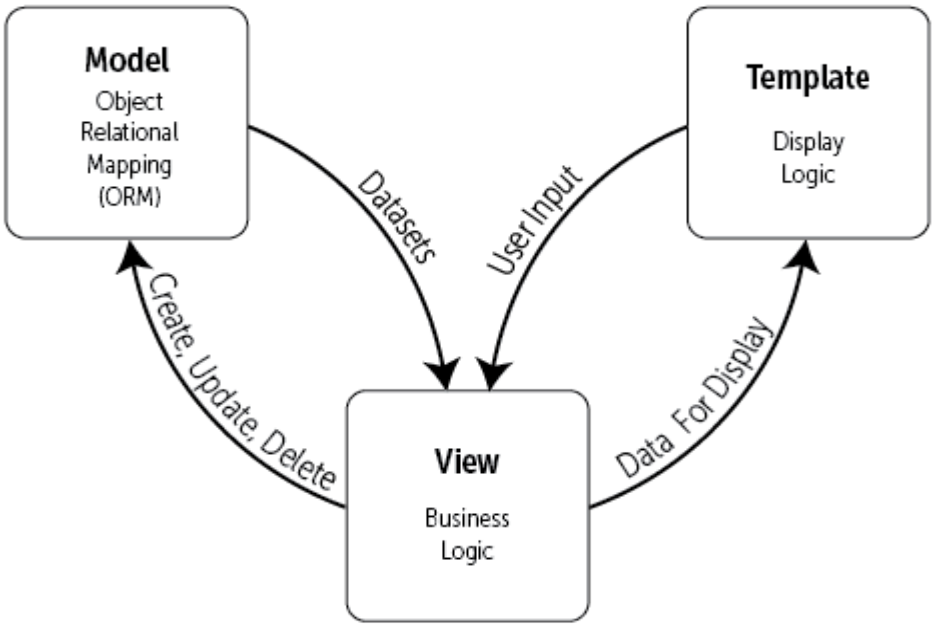
System Architecture Design

System architecture or systems architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.

Objectives of architecture design

a. Describe the fundamental components of an information system.
b. Describe server-based, client-based, and client–server architectures.
c. Describe newer architectural options, such as cloud computing.
d. Explain how operational, performance, security, cultural, and political requirements affect the architecture design.
e. Create an architectural design.

The following picture represents the architecture design of the system



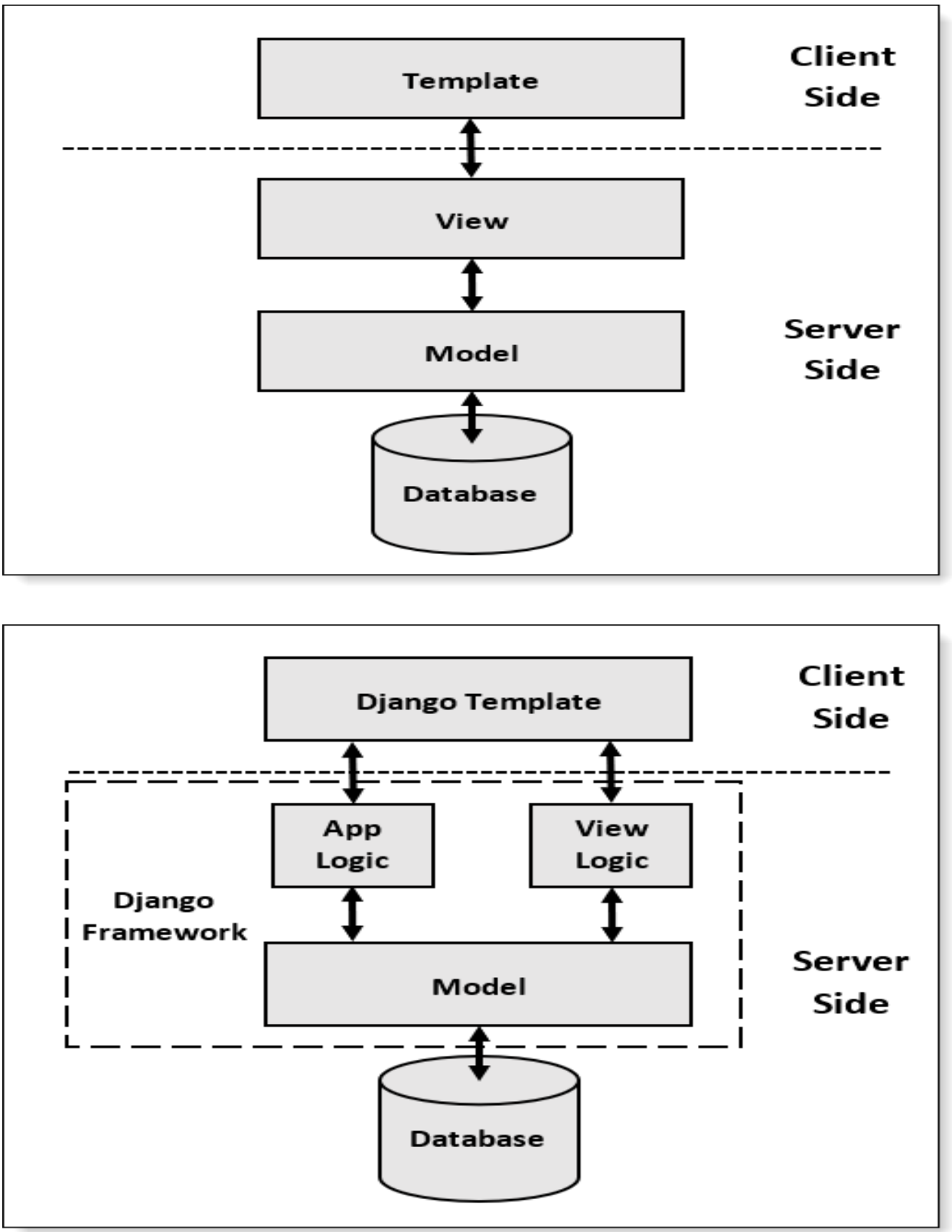


Figure 19: Architecture design of the system

CHAPTER 4

IMPLEMENTATION AND PRESENTATION OF THE NEW SYSTEM

Introduction

This chapter describes the development of "**SMART CRIME SCENE EVIDENCES ANALYSIS**", it encloses a brief overview of the technologies used to make this application, operation, tests that have been applied. Last but not least, software and hardware compatibility requirements.

Technologies used

To develop this application, I used different technologies namely:

Technologies used

Technology Stack Software	Product
Database	PostgreSQL
Integrated development environment	Visual Studio Code
programming languages	Python, Django

Presentation of the New System

Index Page

Figure 19 : Index Page

Login Page

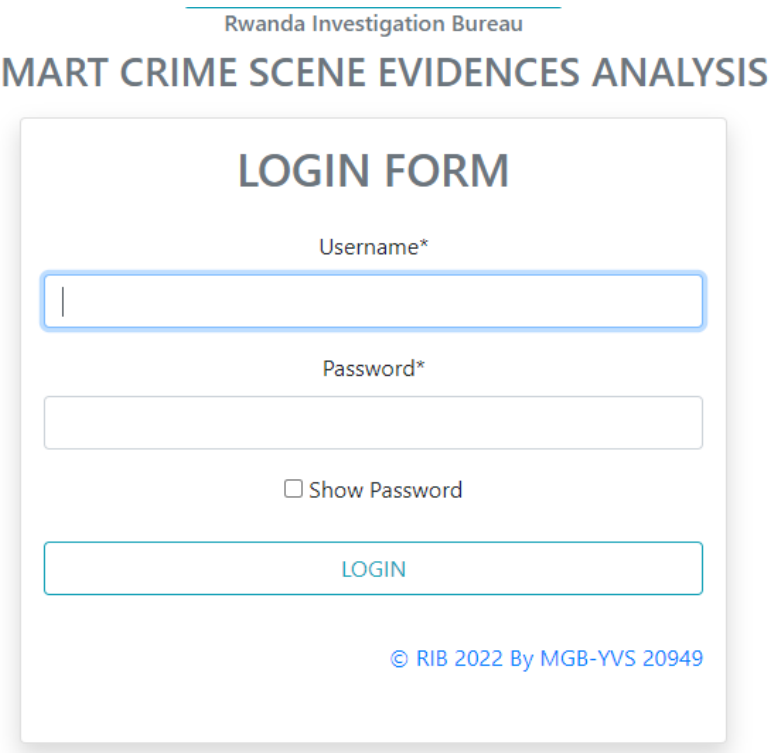


Figure 20 : Login Form

Add Case

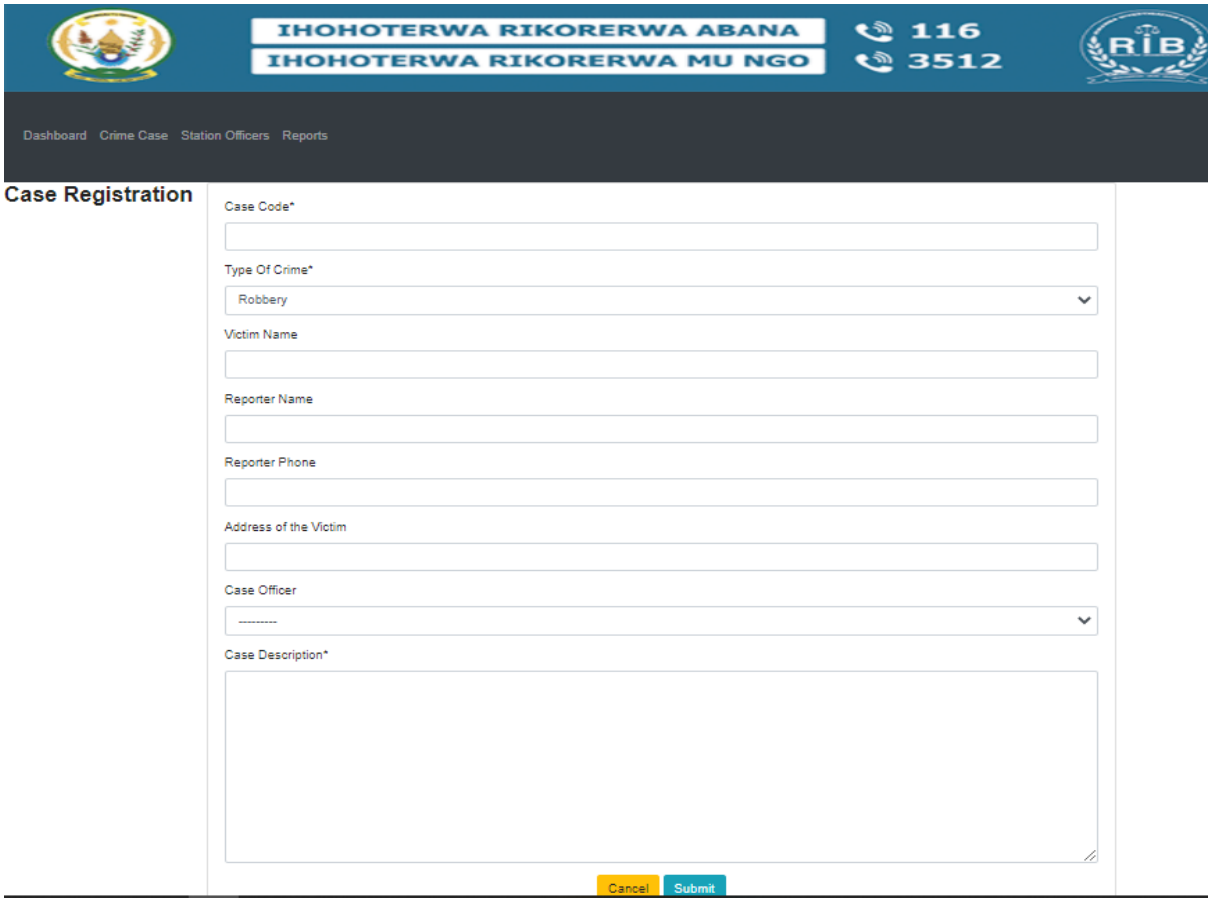


Figure 21 : Add Case

Suspect

Figure 22: Add Suspect

Headquarter site

Dashboard

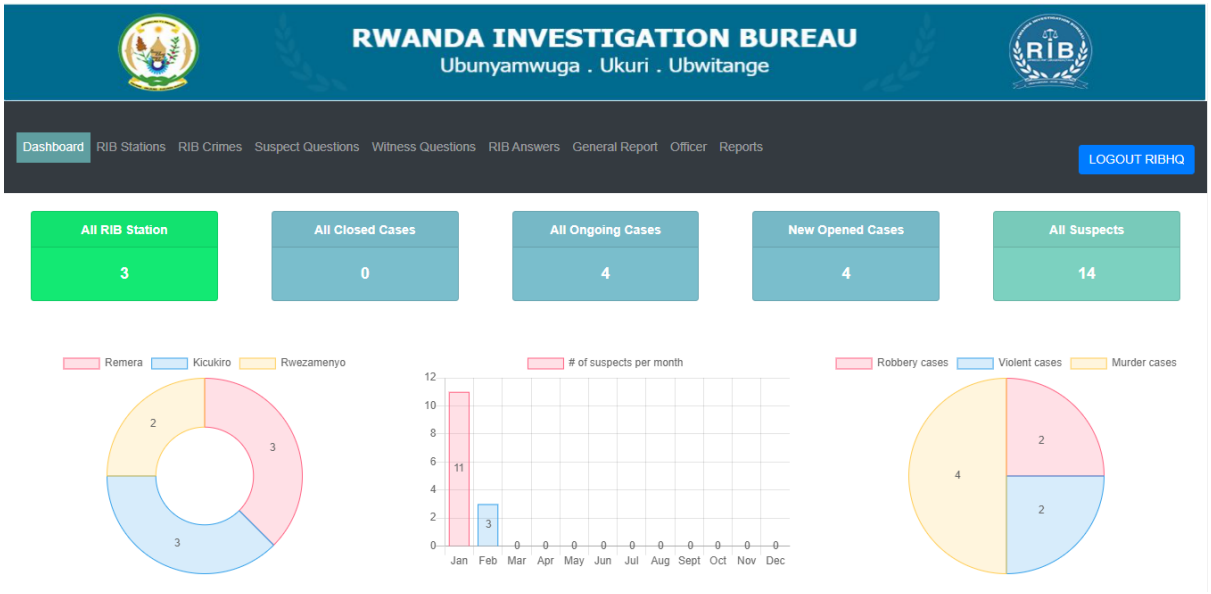


Figure 23: Head quarter Dashboard

RIB Station Dashboard

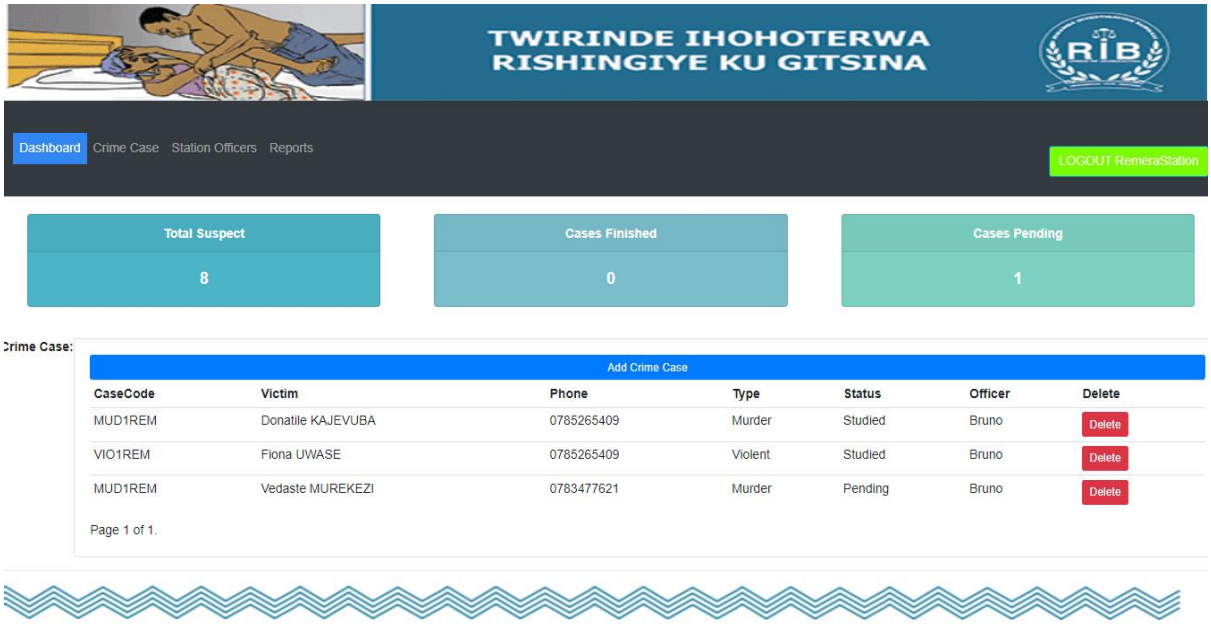


Figure 24 : RIB Station Dashboard

Station user Dashboard

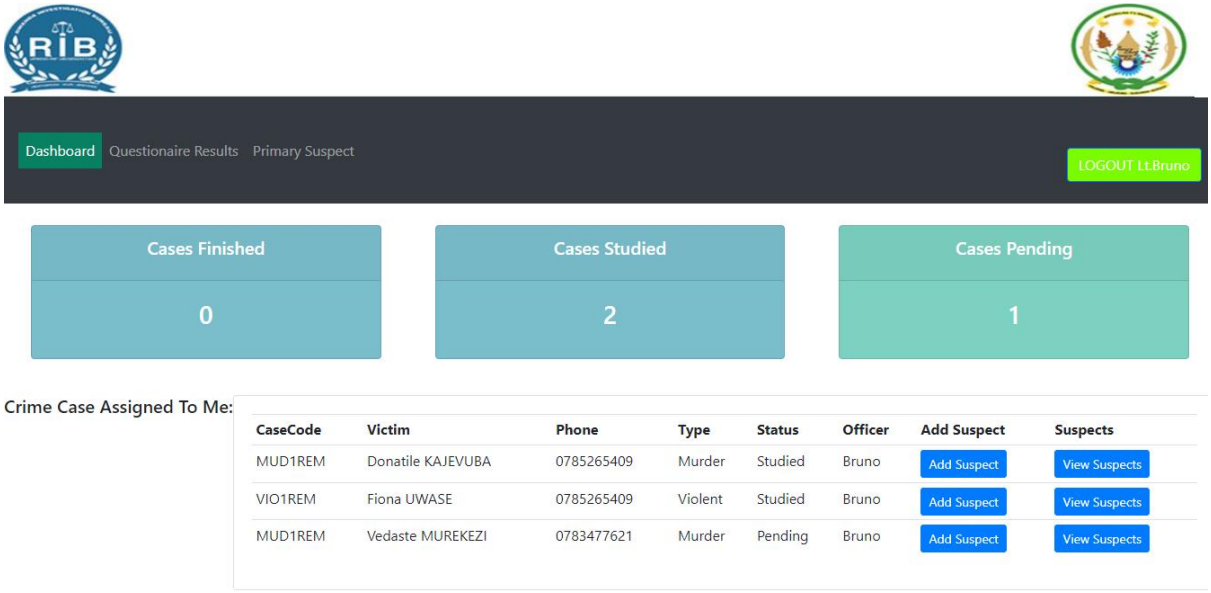


Figure 25 : Station User Dashboard

Generate Report

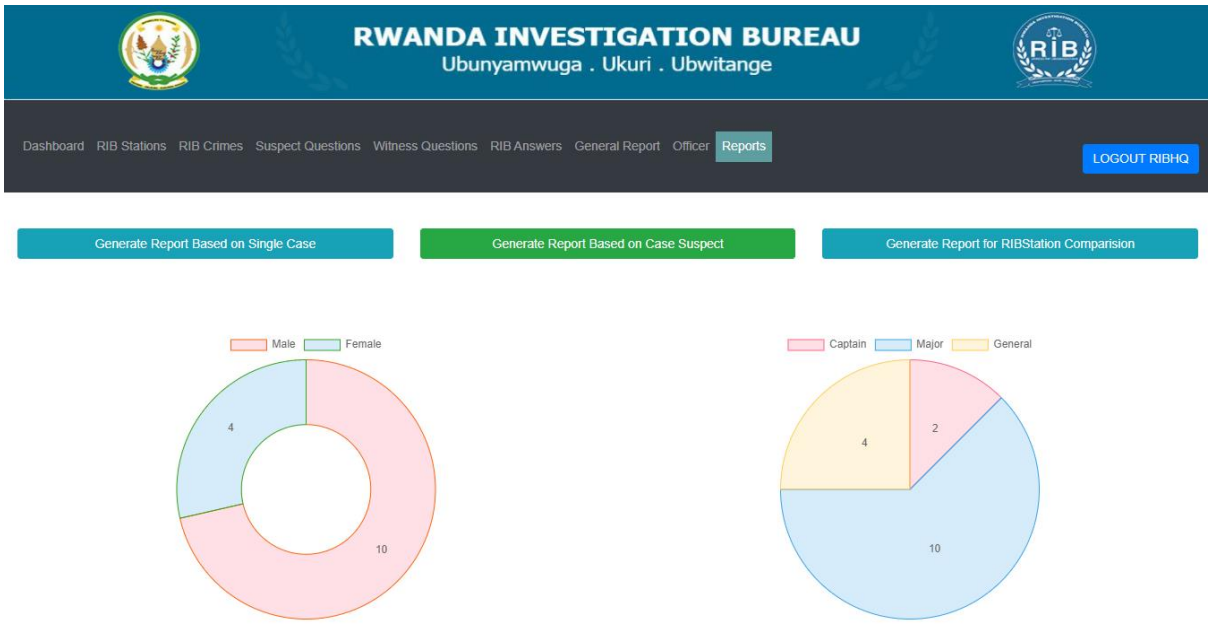


Figure 26 : Generate Report

RIBStation comparison based on cases

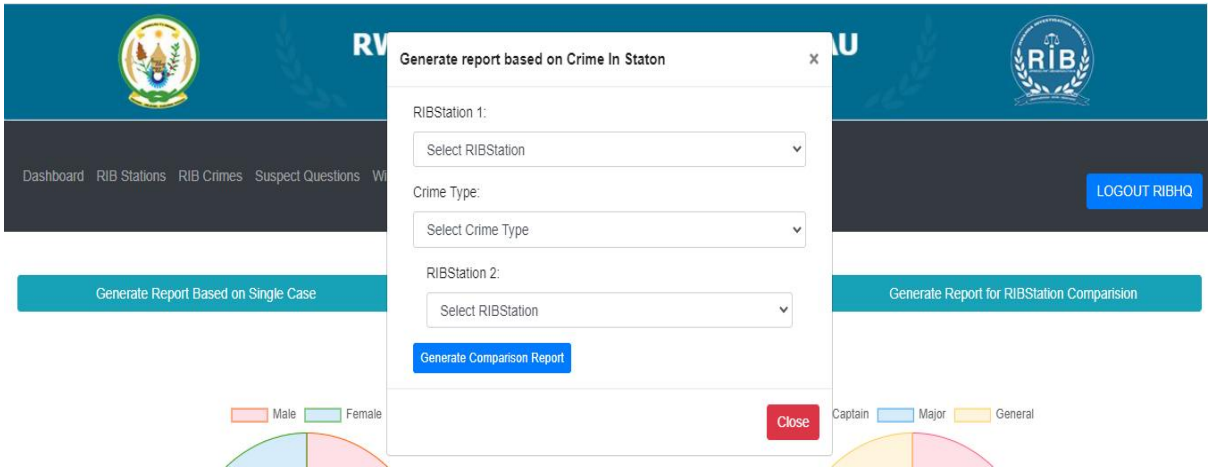


Figure 27 : RIB Report comparison

RIBStation comparison based on cases

RIB Headquarter Report

RIBStation: **RIBHQ**

Location: **Kigali City**

"RIB Remera station" Statistical Report on: Violent Crime

Total Case On this Crime:	Total Suspects:	Total Evidences	Total Witness
1	3	4	4

"RIB Kicukiro station" Statistical Report on:"Violent Crime"

Total Case On this Crime:	Total Suspects:	Total Evidences	Total Witness
1	0	0	0

Figure 28: Pdf Report

Software Testing

Software tests play an important role in the software designing. They help to verify the effectiveness of the software to see if it actually does what it was supposed to solve.

Listed are key aspects to take into consideration in software testing

- Does the application meet the requirements that guided its design and development?
- Does the application works as expected?
- Can the application be implemented with the same characteristics and satisfies the needs of the stakeholders?

The following are some software testing

Unit Test

Unit testing is a process to ensure the proper functioning of particular software or a portion of a program. It is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine if they are fit for use. In other words every small component that can be compiled with the goal to know that every unit matches to its specifications, and to know if there are logical mistakes. Indeed, the unit test is an efficient means that permits to detect the maximum possible mistakes. The application has been checked with the unit test at each piece of the code written.

Integration test

Integration testis the phase in software testing in which individual software modules are combined and tested as a group. This test is useful to check the assembly of the different part of the software. It is also a progression of tests, in which the software and hardware components are collected and tested until the entire system is tested. The application modules have been successively tested until completion to ensure that the whole constituted by the assembled software components answers to the required functional and technical specifications.

Validation test

The last test phase has the role of validating the software in its external environment. The product has been put in final situation in order to verify if it perfectly answers to the needs expressed in the first phase. The validation test is important, since it is necessary to verify if the setting up of the application corresponds to the expressed needs. The application has been tested in its entirety, and it is in this way that we noticed that the progress of operations done corresponds to the functional specifications.

Alpha testing

This is a type of validation testing. It is type of acceptance testing which is done before the product is released. It is typically done by the quality assurance people.

Performance Testing

This is designed to test the run-time performance of software within the context of an integrated system. It is used to test speed and effectiveness of a program.

Regression Testing

This type of testing makes sure that whole component works properly even after adding components to the complete program.

Software and Hardware compatibility requirements

Client-side requirements:

- A web browser (Mozilla Firefox, Google Chrome, Torch, etc.)
- Operating system (Windows xp, vista, 7, 8 10; Linux, iOS).
- A RAM of 2Gigabyte (minimum)
- A hard disk of 8 Gigabyte of free space

Server-side requirements:

- Java SE Development Kit;
- Windows server 2008
- A Web server which support Java and JSP (TOMCAT, GLASSFISH, etc.);
- MYSQL SERVER 5.5;
- Network cark: 1GB/Second;
- RAM: 1GB minimum;
- 2GB or Freer hard disk space.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

Conclusion

The main objective of this project was to design and implement SMART CRIME SCENE EVIDENCE ANALYSIS web project which incorporates different services needed by suspects and very intuitive to them so that the system will support Rwandan Investigation Board to get access to easy way of analysing the crime scene evidences.

During this research project different methods and tools have been used. The observation, documentation and interviews have been used as tools for a good understanding of the existing system. The conception of this information system indeed contributed to the improvement of evidence analysis.

After getting the problems in our scope, analysis was done, using UML in order to find the adequate solution by developing the new system using django framework for python programming language.

With this system being developed, I hope to reduce time wasting, avoid misunderstandings, and provide easy data flow and less hard work. I believe that the pertinent departments, such as RIB stations, Rwanda National Police will find this system highly convenient and reliable in order to accomplish their goals

Our hope is that this software SMART CRIME SCENE EVIDENCE ANALYSIS SYSTEM to improve their performance in activity of primary suspect delivery based on the evidences, and we hope that the use of this software will enhance the Service of evidence analysis.

Finally, it is remained available to receive suggestions and to meet the demands that could be sent to me to participate to the perfection of this work.

Recommendations

I would like to recommend to Rwanda Investigation Bureau to train their officers to use the new system on how it works and what it does, so that the SCSEA will be managed with the use of ICT which is faster, easily and in an effective manner.

In the future the application would include many other functions such as the communication of the system content in different languages like Kinyarwanda and French, to increase the number of users.

In closing this work, I would like to suggest any interested person to add other functions to improve my work in order to improve the crime scene evidences control and analysis in Rwanda.

REFERENCES

Books

- ✓ ARIADNE TRAINING CORPORATE, (2021). UML Applied-Object oriented analysis and design using the UML, Somerset UK.
- ✓ Douglas Bell.(2021).Software Engeneering for Students: A Programming Approach (fourth edition).Addison Wesley.
- ✓ Steve Levine, Maria Rodriguez, Geoffrey Thomas. (2021). Web Programming in Python with Django.
- ✓ Ramez & et al, 2004. Fundamentals of Database Systems. Addison-Wesley

Websites

1. RIB. (2021, May 05). *History*. Retrieved from rib.gov.rw: www.rib.gov.rw/index.php?id=371
2. Django Software Foundation. (2020). *Django Documentation.* Django Software Foundation 15th June 2021. < https://docs.djangoproject.com/en/3.1/
3. System Architecture. (2021). In Wikipedia, the free encyclopedia. Retrieved June 12, 2021, from https://en.wikipedia.org/wiki/Systems_architecture
4. smartdraw. (2021, june 02). *class diagram*. Retrieved from smartdraw.com: https://www.smartdraw.com/class-diagram/#classDiagramSymbols
5. DataFlair. (2021). Django Architecture. DataFlair. 12th August 2021. < https://data-flair.training/blogs/django-architecture/>
6. technopedia. (2021, June 02). *data collection*. Retrieved from technopedia.com: https://www.technopedia.com/definiion/30318/data-collection
7. tutorialspoint. (2021, July 07). *System Analysis and Design Overview*. Retrieved from tutorialspoint.com: https://www.tutorialspoint.com/system_analysis_and_design/system_analysis_and_design_overview
8. Real python (2021). Get started with Django part 2: Django user management. Retrieved August 1, 2021, from Real Python: https://realpython.com/django-user-management/

APPENDICES

CURRICULUM VITAE

Name: Yves MUGABO

Address: Gasabo, Kigali, Rwanda

Nationality: Rwanda

Tel: 0784960500

Email: mugaboyves1@gmail.com

Personal Statement

I am young, energetic and self-motivated person, ready to work and attain results. I strive to achieve and value in everything I put my hands on. Honesty and respect at the work place, is what I put forward, team work and knowledge sharing is what I value.

Educational background

Period	Level	Institution	Awards and Relevant courses
2017-Now	Undergraduate	Adventist University of Central Africa	Bachelor's degree in Information Technology-Software Engineering
2014-2016	Advanced Level(A1)	College ACEJ/KARAMA	High school Diploma in Computer science
2010-2013	Ordinary Level(O' level)	Lycee de KICUKIRO(APADE)	O' level certificate

Relevant Experience

Year	Type of service	Place
(2017-2020)	**IT Support in MAINTENANCE Department**	**OMEGASIR Ltd**
(2018-2020	**Information Technology IT**	**ONCONSULTING GROUP**
(2019-Now)	**Assistant IS Auditor**	**ONTHECH Ltd**

Awards & Trainings

January – May 2017: Trained in computer maintenance and MS-Office professional use and library configuration and troubleshooting problems at TNT Academy.

May – August 2017: Trained in Leadership style at New life Ministries

January– March 2020: Trained in IS Auditing at ONTECH Ltd

Skills

Language skills	Able to speak/write English, Kirundi and French.
Technical skills	In the computer, I am able to use PostgreSQL in database, and to code using Python, php, Java language and JavaScript in designing software.
Professional skills	I am able to lead different people and negotiate with them.
Interpersonal skills	I am sociable, team player.

Hobbies/Interests

-Reading books.

-Using different new application on my computer.

Declaration

I declare that the information provided above is authentic and correct to the best of my knowledge and belief.

References

-Olivier NTAWUYIRUSHINTEGE

Tel:0788303061

-John AFRICA

Tel:0788845687