WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

TITLE: WEB BASED PRISONER INFORMATION MANAGEMENT SYSTEM FOR WOLKITE CITY

Project Team Members:

NO	NAME	ID
1.	EKRAM KEMER	CIR/041/11
2.	ABDELA NESRE	CIR/002/11
3.	ANWAR TIGISTU	CIR/013/11

Advisor Name:-Mr. Feidu Akmel

June 15, 2022

Wolkite, Ethiopia

WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

Web-Based Prisoner Information Management System for Wolkite City

Submitted to department of computer science in partial fulfillment of the requirement for the degree of Bachelor science in computer science.

By:

1. **Ekram Kemer CIR/041/11**
2. **Abdela Nesre CIR/002/11**
3. **Anwar Tigistu CIR/013/11**

June 15, 2022

Wolkite, Ethiopia

Declaration

This is to declare that this project work which is done under the supervision of Mr. Feidu Akmel. And having the title Web-Based Prisoner Information Management System for Wolkite City is the sole contribution of Ekram Kemer, Anwar Tigistu, and Abdela Nesre No part of the project work has been reproduced illegally (copy and paste) which can be considered as Plagiarism. All referenced parts have been used to argue the idea and have been cited properly.

Date: _____

Group Members:

Full Name	Signature
_____	_____
_____	_____
_____	_____

Approval of board of examiners

We, the members of the board of examiners of the final open defense by Ekram Kemer, Anwar Tigstu and Abdela Nesre have read and evaluated his/her final project entitled web based prisoner information management system for Wolkite city and examined the candidates. This is, therefore, to certify that the project has been accepted in partial fulfillment of the requirement of the Degree of Computer Science.

Advisor Name	Signature	Date
Department Head Name	Signature	Date
Examiner 1 Name	Signature	Date
Examiner 2 Name	Signature	Date
Examiner 3 Name	Signature	Date

Acknowledgment

We would like to praise the almighty God who gave us the patience and courage to finish our final project. This project would not have been possible without the support of lots of people. At first our unalloyed appreciation goes to our amiable, ever supportive and humble advisor **Mr. Feidu Akmel** for his great contribution guidance, constant support and gives us continuous initiation throughout the course of our work and we want to thank our department head, **Mr.Sisay** for his support and unlimited guidance to us. Also we would like to thank **Inspector Desta** for giving us detail information about existing system. At last we thank all our teacher and friends who have been always helping and encouraging us.

Table of Contents

Declaration i

Acknowledgment iii

List of Figures viii

List of Tables ..... **Error! Bookmark not defined.**

List of Abbreviations x

ABSTRACT xi

CHAPTER ONE 1

1. INTRODUCTION 1

1.1. Background of the organization 1

1.2. Statement of problem 2

1.3. Objective of the project 2

1.3.1. General Objective 2

1.3.2. Specific Objective 2

1.4. Feasibility study of the new system 3

1.4.1. Operational Feasibility 3

1.4.2. Economic Feasibility 3

1.4.2.1. Tangible benefit 3

1.4.2.2. Intangible benefit 3

1.4.3. Technical Feasibility 4

1.4.4. Legal Feasibility 4

1.5. Scope and limitation of the project 4

1.5.1. Scope Of project 4

1.5.2. Limitation of project 4

1.6. Significance of the project 5

1.7. Beneficiary of the project 5

1.8. Methodology of the project 5

1.8.1. Data Collection Tools/Techniques 5

1.8.2. System analysis and design 6

1.8.3. System Development Model 6

1.8.4. System Testing Methodology 7

1.8.4.1. Unit Testing 7

1.8.4.2. System Testing 7
1.8.4.3. Integration testing 7
1.8.4.4. Acceptance testing 7
1.8.5. Development tools and Technologies 7
1.8.5.1. Front-end technologies 7
1.8.5.2. Back-end technologies 7
1.8.5.3. Documentation and Modeling Tool 7
1.8.5.4. Deployment Environment 8
1.9 .Document Organization 8
CHAPTER TWO 9
2. DESCRIPTION OF THE EXISTING SYSTEM 9
2.1. Introduction of the Existing System 9
2.2. Users of the Existing System 9
2.3. Major Functions of the Existing System 10
2.4. Forms and Other Documents of the Existing System 11
2.5. Drawbacks of the Existing System 11
2.6. Business Rules of the Existing System 11
CHAPTER THREE 13
3. PROPOSED SYSTEM 13
3.1. Functional Requirements 13
3.2. Non-functional Requirements 15
3.2.1. User Interface and Human Factors 15
3.2.2. Hardware Consideration 16
3.2.3. Security Issues 16
3.2.4. Performance Consideration 16
3.2.5. Error Handling and Validation 16
3.2.6. Quality Issues 16
3.2.7. Backup and Recovery 16
3.2.8. Physical Environment 17
3.2.9. Resource Issues 17
3.2.10. Documentation 17
CHAPTER FOUR 18
4. SYSTEM ANALYSIS 18
4.1. System Model 18

4.2. Use Case Model 18
4.2.1. Use Case Diagram 18
4.2.2. Use case description 19
4.2.3. Use case Scenario 25
4.3. Object Model 26
4.3.1. Class Diagram 26
4.3.2. Data Dictionary 28
4.4. Dynamic Model 32
4.4.1. Sequence Diagram 32
4.4.2. Activity Diagram 35
4.4.3. State Chart Diagram 39
CHAPTER FIVE 42
5. SYSTEM DESIGN 42
5.1. Design Goals 42
5.2. Current System Architecture 44
5.2. Proposed System Architecture 44
5.2.1. Subsystem Decomposition and Description 45
5.2.2. Hardware/Software Mapping 47
5.2.3. Detailed Class Diagram 49
5.2.4. Persistent Data Management 50
5.2.5. Access Control and Security 51
5.3. Packages 51
5.4. Algorithm Design 52
5.5. User Interface Design 54
CHAPTER SIX 56
6. IMPLEMENTATION AND TESTING 56
6.1 Implementation of the database 56
6.2. Implementation of the Class Diagram 57
6.4. Configuration of application security 58
6.5. Implementation of user interface 59
6.6. Testing 60
6.6.1. Test Case 60
6.6.2. Unit Testing 61
6.6.3. Integration Testing 61

6.6.4. System Testing 61
6.6.5. Acceptance Testing 62
CHAPTER SEVEN 63
7. CONCLUSION AND RECOMMENDATION 63
7.1. Conclusion 63
7.2. Recommendation 63

List of Figures

Figure 4. 1 use case diagram of the system 19
Figure 4. 2 Class diagram of the system.....27
Figure 4. 3 Sequence diagram for login page 33
Figure 4. 4 Sequence diagram for create account 34
Figure 4. 5 Sequence diagram for register 34
Figure 4. 6 Activity diagram for Login page 36
Figure 4. 7 Activity diagram for prisoner registration.....37
Figure 4. 8 Activity diagram for create account 38
Figure 4. 9 State chart diagram for login 40
Figure 4. 10 State chart diagram for create account 41

Figure 5. 3 Login user interface 54
Figure 5. 4 Register prisoner user interface.....55
Figure 5. 5 New prisoner registration form.....65
Figure 5. 6 Released prisoner registration.....66

Figure 6. 1 Sample database implementation 57
Figure 6. 2 XAMPP application server.....59

List of Tables

Table 4. 1 Login Use case description 20
Table 4. 2 Login Use case description 20
Table 4. 3 Register prisoner use case description 21
Table 4. 4 Assign lawyer use case description 21
Table 4. 5 Generate Report use case description 22
Table 4. 6 Update prisoners' information use case description 22
Table 4. 7 View request use case description 23
Table 4. 8 Send request use case description 24
Table 4. 9 User data dictionary 28
Table 4. 10 Police data dictionary 28

Table 5. 1 1Access privilege 51

Table 6. 1 Test case to create account 60

List of Abbreviations

Abbreviation	Description
WBPIMS	Web Based Prisoner Information management System
CPU	Central Processing Unit
RAM	Random Access Memory
OOD	Object Oriented Design
OOP	Object Oriented Programming
PC	Personal Computer
PHP	Personal Home Page
HTML	Hyper Text Markup Language
CSS	Cascading Style Sheet
DBMS	Data Base Management System
MYSQL	My Structured Query Language
GUI	Graphical User Interface
UI	User Interface

ABSTRACT

Currently, prisoner's records are maintained in a very rudimentary way in the form of both manual and semi-automated in wolkite prison administration. This current manual and semi-automated system cannot handle the records well. This method of data management often results in human error, delay to retrieve information effectively and efficiently etc. There are repetitive and impairment activities occur in the organization those are difficult to integrate data from different individual about prisoner records. The general objective of the project is to develop web-based prisoner information management system for Wolkite City. Our project made use of data collected method from Wolkite prison administration, different materials, journals, and from various authors. Proposed system would developed by PHP as back-end technology and HTML, CSS, and JAVASCRIPT as front-end technology and all of the data need for the application is stored in the form of tables in phpmyadmin. Our proposed system solves the problems of the existing system, because we would develop automated system which replace activities of existing system in Wolkite prison administration. Generally this project mainly focuses on the prisoner which is already judged by the court, So Web based Prisoner Information Management System for Wolkite City concerns after having judged prisoner with its back ground detailed.

CHAPTER ONE

1. INTRODUCTION

As today's world there are many organizations and these organizations should have facilitated their working area in a computerized way to be preferable and competitive. In Ethiopia context, since manual and semi-automated prisoner information management system is a routine work, the processes takes more time and human power to enter data. Web based prisoner information management system for Wolkite city is the title which we are interested and work on it.

Wolkite "maremiya bet" uses traditional way of performing activities related to manage prisoner information. This makes the system very complex, boring and time wasting activity. Because of this reason, it is difficult to register, search and update a single data of many prisoner's information, loss of data which lead to security problem of the organization, use much space and human power. And activities like giving parole for prisoner made manually. By considering these, our group aimed to propose automated system to solve such type of tedious work flow and to fast the working structure in the organization. The ultimate goal of the prison satiation can be achieved through the newly developed system.

The goal of this project is to develop an automated system which can be easily accessed by the user. Generally, this web-based prisoner information management system would help the prison stuff to perform their tasks efficiently and timely.

1.1. Background of the organization

Wolkite prison which known as "maremiya bet" found in wolkite city, Gurage zone, southern region. The prison was established in 1980 E.C. Its main objective was preventing different type of crime and providing other security issue for the people of the Wolkite city. Prisoners' can be punished by heavy and simple crimes such as murder, abduction, steal, terrorism and corruption. Currently the number of prisoner in Wolkite prison administration is 467. From this the number of male prisoners are 409 and female prisoners are 58.There is one child from 407 male prisoners.The current system organization uses both manual and semi-automated system but there are many drawback like difficult to access prisoner information, register trainee of prisoner, generate report and many other problem which decrease the work flow of the prison. Due to this system employee in the organization loss their time, power and money to manage prisoner information.

1.2. Statement of problem

Prison management is a tedious task and especially when the numbers are too high to handle, resources are minimal to manage and there is lack of trained manpower to manage pressures and crisis situations. “Wolkite maremiya bet” applies both manual and semi-automated way of recording prisoners’ information. Among that semi computerized activity prisoners’ registration is the most common system activity, but other activities like generate report, visitor registration, trainee registration, evaluation of parole point has done in manual and there is no well-organized database to store and manage prisoners’ information. Because of this many problems occur. There is data sharing problem which mean there is recorded prisoner information in a single computer but no networked that enables employees want to do their works on another computer. The system is not secured because easy access prisoners’ information by unauthorized user. Document which generated from different staff take much paper. Takes much time and human power when employee manages prisoner information like register, generate report and give certificate and others. There is poor flow of critical information between departments in the organization. Such types of manual system requires much human power to accomplish the task. There is improper management, creation, preservation and removal of record of prisoner. The existing system does not create smooth work flow between staffs of prison. Also the system coasts high, which mean the system takes a lot of hard copies and other instruments.

1.3. Objective of the project

1.3.1. General Objective

The main objective of this project is to develop Web based Prisoner information management system for Wolkite prison administration which can be easily accessed by the user.

1.3.2. Specific Objective

To achieve our general objective, the following specific objective needed:-

- To understand the existing system.
- To analyze the problem to specifying the user and system requirements.
- To design the System and produce design document
- To implement the system
- To test the system

1.4. Feasibility study of the new system

Feasibility study is essential to evaluate the cost and benefits of the new system. The main objective of feasibility study is an analysis of how successfully a project can be completed and to test the technical, operational, economical and legal feasibility. On the basis of the feasibility study decision is taken on whether to proceed or to cancel the proposed project.

1.4.1. Operational Feasibility

The system to be developed will provide accurate, active, secured service and decreases labor of workers. Since the proposed system is web based application, it is plat form independent i.e.it runs in all operating system and supports all browser. It determines how the system satisfies organizations needs or solves the organizations problem and system has good user interfaces. It is simple for users to use the system after developed, so the system operationally feasible.

1.4.2. Economic Feasibility

Both hardware and software required for developing this system are highly available and can be owned with small cost. The proposed system to develop will decrease a lot of cost that was expensive to buy the hard document material such as paper, pencil, rubber, and printer and so on. Also reduce the loss of data that means if it reduces the loss of data it also reduces the cost of replacing the lost data. We divide the economic feasibility in to two parts. These are tangible and intangible benefits.

1.4.2.1. Tangible benefit

- Reduces cost for printing and distributing registration papers.
- Reduce cost for transporting registration papers from one place to another.
- Reduce number of employees to manage the data of prisoner manually.
- Decrease a lot of money to buy the hard document material such as paper, pencil, rubber, and printer and so on.

1.4.2.2. Intangible benefit

- Reduce errors
- Increase consistency and correctness
- Improve the moral of the employee’s
- Enable to make fast decision
- To get well organized information in short period of time
- Work flow would efficient

1.4.3. Technical Feasibility

In technical feasibility study, one has to test whether the proposed system can be developed using existing technology or not. This system is technically big enough to be applied easily to the problem identified in the existing system. In addition; both hardware and software for this system are highly available and can be owned with small cost. Also we the group members have the capability to use these technologies to develop the system. Therefore, it can be concluded that the system is technically feasible.

1.4.4. Legal Feasibility

Legally the proposed system is fully feasible because it never contradicts with legal and criminal Laws of the country. If the system has contradicted with the governmental rules regulations automatically rejected. But this system has not contradicted with the governmental rule and regulations. Therefore the system is legally feasible.

1.5. Scope and limitation of the project

1.5.1. Scope Of project

Scope defines the coverage areas of the project, activities, and operations done by the system. The scope of our project particularly focus on automating over all prisoner related information management only for Wolkite City. The new system keeps track of the prisoner information management system activities such as such as requesting, training, transfer, generation of certificate, generation of overall report. The main module that are going to be developed are:-

- Register detail information of the prisoner: all information of the prisoner registered and stored on the system.
- Generation of certificate for the prisoners attending the training: by selecting the prisoner who finished the training
- Approve or reject the prisoners request based on certain evaluation point
- Identify the prisoner per-charge means checking whether the prison record or not.
- Upload the schedule for police
- Post a notice to announce different type of information
- Generating timely reports to the concerned body

1.5.2. Limitation of project

The existing prisoner information management system performs various activities.

However, our system not address the following activities.

- The system does not support blind user
- The system does not allow multiple language

1.6. Significance of the project

The main purpose of our project is to change the manual and semi-automated system into automated one. Generally the system have the following purpose:

- Reduce time consuming activities like register, search, and update prisoners' and visitors' information easily in the organization.
- Avoid data redundancy, which means extended data stored at integrated format for easiness of searching, sorting.
- Increase efficiency of data integrity in the organization.
- The details information of the prisoner stored in the database.

1.7. Beneficiary of the project

The prison is beneficiary because:-

- It can perform different activities such as avoiding improper resource consumption like paper, pen.
- Decrease managing cost
- Decrease wastage of human power
- Create fast & efficient sharing of critical information within the Staff.

The Project also help team members

- The project initiates our team to get knowledge of how to develop the required system.
- Our team get a lot of experience of solving the problem while they are facing different type of difficulties.

1.8. Methodology of the project

When our group developing this project different methodologies and tools would be applied or used.

1.8.1. Data Collection Tools/Techniques

Data collection is the most important part of the project to find the main requirement of the system and to understand how the system does among the methods. We used different methods to collect data. Data for developing this system obtained from different sources.

Questionnaires: To get more information we select some workers and prepare questions for them about the system.

Interview: Was used to gather required data for the project by contacting different employees of the organization like police officer and inspector.

Direct Observation:-We observe the current system and identify the problems regarding to the prison in managing the prisoners information.so it help us as easy way to understand the system and to develop the project.

1.8.2. System analysis and design

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 the components of the system work efficiently to accomplish their purpose. In analysis stage the developers has done a survey to get more information about the software they are going to build. What the developer achieved at the end of this phase is that all the information needed including what the problem that the current software encounter, how to fix and enhance it and choose the best alternative solution for all these problems. We have selected the object-oriented system analysis and design method specifically UML (Unified Modeling Language) model because:

- ❖ It enables us to comprehensively model a system before we develop it
- ❖ Increased consistency among analysis, design and programming activities
- ❖ Improved communication among users, analysis, design, and programming

1.8.3. System Development Model

A system development methodology refers to the framework that is used to structure, plan, and control the process of developing the system. There are various software development life cycle models defined and designed which can be followed during the software development process. These models are also referred as “Software Development Process Models". Each process model follows a series of steps unique to its type to ensure success in the process of software development. But the proposed system follows iterative model. It work by repeating every step after every cycle of SDLC process.

- ❖ Software will be first developed on a very small scale.
- ❖ On every next iteration, more features and modules will be designed, coded, tested, and will be added to the software.

- Every cycle will produce a software, which is complete in itself and has more features and capabilities than that of the previous one

1.8.4. System Testing Methodology

Testing will be done while coding, in order for early discovery of errors and their remedy is the key to reliable software.

1.8.4.1. Unit Testing

Unit testing will be done at the source code level for language specific programming error such as bad syntax, logic errors, or to test particular function or code error.

1.8.4.2. System Testing

The goal of system testing is to detect fault that can only be exposed by testing the entire integrated system or some major part of it.

1.8.4.3. Integration testing

In this level of testing we will examine how the different procedure works together to achieve the goal of the sub system.

1.8.4.4. Acceptance testing

The purpose of acceptance testing is purely to ensure that the end user can achieve the goals set in the business requirements.

1.8.5. Development tools and Technologies

1.8.5.1. Front-end technologies

- HTML: to define the content of web pages
- CSS: to specify the layout of web pages
- JavaScript: to program the behavior of web pages
- Bootstrap: to develop responsive website

1.8.5.2. Back-end technologies

- Apache Server: to compile the sever side scripting language
- MYSQL DBMS: server to compile SQL queries and store data
- PHP: for server-side scripting language

1.8.5.3. Documentation and Modeling Tool

We use the following software tools:-

- Microsoft office word 2010:To write the documentation
- Microsoft PowerPoint 2010:For presentation at the end of the project

- Edraw max: to draw different UML diagrams such as use cases, sequence diagrams, activity diagrams, class diagram and so on.
- Browser: chrome, Firefox, opera, UC browser, and Internet explorer used to run the program.

1.8.5.4. Deployment Environment

Editor: Notepad++, notepad for writing programs or codes

1.9 .Document Organization

Chapter one discuss about the introduction of the system, and the statement of problems, the scope and limitation, the objective of the project, the methodology, and the beneficiaries of the project after results.

Chapter two discuss about the existing system, user of existing system major functions and the business rules.

Chapter Three is about Overview of the proposed system which includes functional And non-functional requirements.

Chapter four discuss about system analysis using system model, object model and Dynamic models that are defined in the functional.

Chapter five deals with system design. Which includes the overview of the system, Design consideration, design goal, design trade-offs, the architecture of the system, Subsystem decomposition, persistent data management and class interface.

CHAPTER TWO

2. DESCRIPTION OF THE EXISTING SYSTEM

2.1. Introduction of the Existing System

In this chapter we deal about the existing system by interview and document analysis method. During the time of requirement gathering of existing system our team members have direct contact with the concerned prison to know about the major functionality of the existing system, players in the existing system, working flow and also any kind of business rule that prison follow.

The existing system of prisoner information management system uses semi-automated way of performing activities, however there are some activities that done manually. Many processes such as search for prisoner information, transfer to another prison, registration for prisoners, report generation, visitors request and many other activates is been done manually.

There is no well-organized database to store and manage prisoners' information. This leads to a very inefficient and improper way of storing records and greatly slow down the flow of critical information as well as makes looking up of information time-consuming.

When we look for transferring the prisoner to other prison the prisoner should apply request to the register officer then the register officer will analyze and report it to the administrator to confirm the request if the request is accepted then the prison will contact to another prison that they need to transfer a prisoner due to some specific case, then after all these process the prisoner will be transferred which is very time wasting process.

2.2. Users of the Existing System

Users are actors that interact, use and participate with in the system. They are players of the existing system who are involved in the system directly or indirectly. Users of existing system are listed below with their given responsibilities.

- **Registrar officer**: responsible to register the prisoner detail information and generates different types of reports.
- **Security manager**: responsible for controlling an activity that needs security and assigns police for a different purpose.
- **Police:** police responsible to keep the prisoners for example when they are moving to clinics, courts and keep the prison.

- **Prisoners:** prisoner is the one who arrested in the prison or a person who did a crime and that are registered in the system.
- **Visitors:** visitor is the one came to visit the arrested person in the prison.
- **Training department**: responsible to register the prisoner that needs training and give an academic or vocational training.
- **Discipline officer**: responsible to manage the prisoner discipline and imposed disciplinary penalties also providing counseling service for the prisoners.
- **Inspector:** responsible to manage all over the system and approve situation when approvement of anything needed.

2.3. Major Functions of the Existing System

The existing Wolkite prisoner information management system performs the following major function:

- Register prisoner's information: if new prisoner who judged by court came to prison his/her detail information is registered on manual form or semi-automated system.
- Give an academic training for the prisoners: the prison administration give academic training for prisoner.
- Generate report: in the existing system different types of reports generated for different reason
- Food service for prisoner: in the existing food services given for prisoner every day.
- Controlling prisoner's discipline: day to day discipline of prisoner checked in the existing system
- Register visitor: the detail information of visitor registered on the system for different purpose such as for security issue.
- Give certificate for prisoner: in the existing system prisoner who finished the training given by training officer, he/she take certificate.
- Give parole for prisoner: in the existing system prisoner who fulfilled the evaluation criteria for parole they can get the parole.
- Schedule for police: in the existing system the prison administration upload schedule for police.

2.4. Forms and Other Documents of the Existing System

There are a number of forms and documents which are used by the existing system. The uses of these forms are to assure the prisoner information management system process. In addition to this, these forms are used to increase easy management of activities that are done in the system. Form which listed below (Figure 5.7) is form which used in existing system to register new prisoner. Figure 5.8 tells about list of prisoner who released from the prison. (Figure 5.7) is prisoner registration form and (Figure 5.8) show list of released prisoner.

2.5. Drawbacks of the Existing System

- **Registration related problem:** The visitor, trainee, and prisoner registration is done manually. This will lead to poor techniques of handling the data because the data can be loss or redundantly registered.
- **Information Problem:** Due to manual collecting of data, there are a redundant record and inconsistency of data problems. The inaccuracy of data and information may produce poor flow of information between staff and lack of a well-organized database system.
- **Data storage problem:** Data are not easily accessible due to its integration which is placed in different locations because of this loss of data may occur.
- **Efficiency Problem:** The efficiency of the existing system is not optimal, because storing, locating data of prisoners' takes much more time.
- **Security and control problem:** The existing system can be accessed by an unauthorized person since it doesn't have any authentication and authorization system.

2.6. Business Rules of the Existing System

Existing System business rule is a statement that describes a business policy or procedure that must be fulfilled and obligated in order for the existing system will function properly and effectively. Prisoner information management system has its own businesses rule to facilitate the service and the type of services which are functional used by them. So the business rules of existing system are the following:

Name: TO REGISTER PRISONER

BR1: The age of prisoner to be registered must be greater than 15

Name: TO GIVE PAROLE FOR PRISONER

BR2: Only the prisoner who full-fill the evaluation method, such as good behavior will get parole.

Name: VISITOR TO VISIT PRISONER

BR3:Visitors whose age less than 9 are not registered in the system.

CHAPTER THREE

3. PROPOSED SYSTEM

3.1. Functional Requirements

This part of the document defines the function of the proposed system. The functional requirement is functioning or features that the system must include to satisfy the system need and to be acceptable by the user. Our proposed system is web based prisoner information management system and developed to solve the problems of the existing system. Because of this we are proposed to solve the problem of the existing prisoner information management system in wolkite city by developing an automated system. That means our proposed system will eliminates problems of the existing system. The functional requirements for the new system that will replace the existing system include:

- The system should register prisoner, visitor, trainee and user to the system.
- The system enables to create, update and delete accounts.
- The system must allow search information
- System allows parole for prisoners
- The system generate report from the database
- The system should identify the prisoner per-charge,which mean system check weather prisoner have record or not.
- The system allow generation of certificate for prisoner
- The system allow to send request when transfer prisoner

Functional requirement with Actors

System administrator:-The proposed system allow System administrator to do the following functionalities:

- Manage Account
- Restore database

Discipline officer:-The proposed system allow discipline officer to do the following functionalities:

- View request that come from users
- Evaluate parole point
- View prisoner information
- Send evaluated parole point

Register officer:-The proposed system allow register officer to do the following functionalities:

- Register prisoner information
- View evaluated parole point
- View new transfer
- Register lawyer
- Assign lawyer
- Generate Report

Security manager:-The proposed system allow security manager to do the following functionalities:

- View request
- View prisoner information
- Register Visitor

Police:-The proposed system allow Police to do the following functionalities:

- View assign position and time

Inspector:-The proposed system allow Inspector to do the following functionalities:

- Post notice
- Approve Request
- View Report
- View request
- View prisoner information

Training officer:-The proposed system allow training officer to do the following functionalities:

- Generate Report
- View prisoner information
- Register trainer
- Give certificate
- View applicant

Prisoner lawyer:-The proposed system allow Prisoner lawyer to do the following functionalities:

- View prisoner information
- Send request

- View response

3.2. Non-functional Requirements

Non-functional requirements, as the name suggests, are requirements that are not directly concerned with the specific services delivered by the system to its users. On-functional requirements specifies how the system should behave. It describe user-visible aspects of the system that are not directly related with the functional behavior of the system. A non-functional requirement relates to the technical aspects of system must fulfill, such as performance-related, security and reliability issues and others. Some of the nonfunctional requirements are the following:

3.2.1. User Interface and Human Factors

The User Interface (UI) is the space where interactions between humans and computers occur; it consists of information output from the machine, as well as a set of control elements for the user to perform certain actions. These interactions enable an efficient system where machines aid the user's decision-making process and, in turn, the user can effectively operate it. When building the UI, the goal is to make it self-explanatory and user-friendly so that users can quickly achieve the desired results. The proposed system provides web based application user interfaces that are compatible with any platforms. The interface is simple to understand; easy to use and user-friendly interface and users of the system easily use and perform their task. Everything stems from knowing our users, including understanding their goals, skills, preferences, and tendencies. The interface will be user friendly and can properly guide the user how to use the system and perform operations easily since every link name in the system similar to the task performed. When we design our system user interface the following should be considered:

- Keep the interface simple: the best interfaces are almost invisible to the user. They avoid unnecessary elements and are clear in the language they use on labels and in messaging.
- Strategically use color and texture: You can direct attention toward or redirect attention away from items using color, light, contrast, and texture to your advantage.
- By carefully thinking about and anticipating the goals people bring to our site; we create defaults in order to reduce the burden on the user.

3.2.2. Hardware Consideration

Our system does not need any other hardware requirements rather than the existing standard desktop computer and laptops. The system will be portable that can be run on any type of computer and it supports any type of browsers.

3.2.3. Security Issues

The objective of system security is the protection of information and property from theft, corruption and other types of damage. Our system is secure that meant any users who haven't an authentication can't access the system. Any user who wants to login to the system must have a user name and password. This protects the system from any unauthorized access.

3.2.4. Performance Consideration

Performance of the system is depending on the time required to perform a given task and memory required for that task so factors like avoiding redundancy of codes increases the time of operation and memory space of the system. This make our system perform user task efficiently.

3.2.5. Error Handling and Validation

The proposed system consists the way to handle error and check whether the information entered valid or not. The system displays error messages if the user enters invalid input. This means that we will implement this system by validating at the front end and back ends.

3.2.6. Quality Issues

The proposed system should always be available for users to access. This makes the system to be available. The system also reliable because it ignore unauthorized user to use the system. With this the proposed systems quality increase.

3.2.7. Backup and Recovery

Backup and recovery describe the process of creating and storing copies of data that can be used to protect organizations against data loss. The proposed system can be damaged or fail if there is virus attack and continuous power disconnection from the source station and the data can be lost at that time. The system should be holding a backup of the data because database backup technology is available. The proposed system consists backup and recovery technique to protect data failure because of some system problem. The system administrator can back up the data when the system needs a backup.

In generally, the purpose of backup and recovery strategy is to protect the database against data loss and reconstruct database after data loss. Typically, backup administration tasks include the following:

- Planning and testing responses to different kinds of failure
- Configuring the database environment for backup and recovery
- Setting up the backup schedule
- Monitoring the backup and recovery environment
- Troubleshooting backup problems
- Recovering from data loss if need arises

3.2.8. Physical Environment

The system needs resources in order to do the requirements one is a server computer and client computer. It deploys in server computer which support window operating system and the client computer access it from the server and can use it. In the physical environmental factors, to protect the server from overheat.

3.2.9. Resource Issues

The system needs different hardware and software resources to deploy the system on the prison like servers, computers and other software setups.

3.2.10. Documentation

For programmer reliable documentation is always a must. Successful documentation will make information easily accessible, provide a limited number of user entry points, help new users learn quickly.

Documentation exists to explain product functionality, unify project-related information, and allow for discussing all significant questions arising between stakeholders and developers. The proposed system should have a good document in order to guide user how to use the system with detail and more precise way.

CHAPTER FOUR

4. SYSTEM ANALYSIS

The project development team used an object-oriented system development methodology. Because the Object system development approach gives an easier and natural way to break down problems into simple, small and manageable components so that it reduces the vague appearance of the big problem. Moreover, it is a predominately used and popular method in the present software development trend. The major activities described in this chapter are constructing a use case model, documenting the use case course of events, constructing sequence diagram, activity, state chart diagram, data dictionary and class diagram.[1]

4.1. System Model

To produce a model of system which is correct, complete and consistent we need to construct the analysis model which focuses on structuring and formalizing the requirements of the system analysis model contains three models: functional, object and dynamic models. The functional model can be described by use case diagrams. Class diagram describe the object model. Dynamic model can also be described in term of sequence, activity diagram and state charts.

4.2. Use Case Model

Actors: An actor is a person, organization, or external system that plays a role in one or more interactions with your system. It is about who use, affected and affects the system. Our system consists the following actors which listed below:

- System administrator
- Discipline officer
- Registrar officer
- Security manager
- Police
- Inspector
- Training officer
- Prisoner lawyer

4.2.1. Use Case Diagram

A use case: is an interaction between users and a system. It captures the goal of the users and the responsibility the system to its users. We are using use case model because of use

case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. [2]

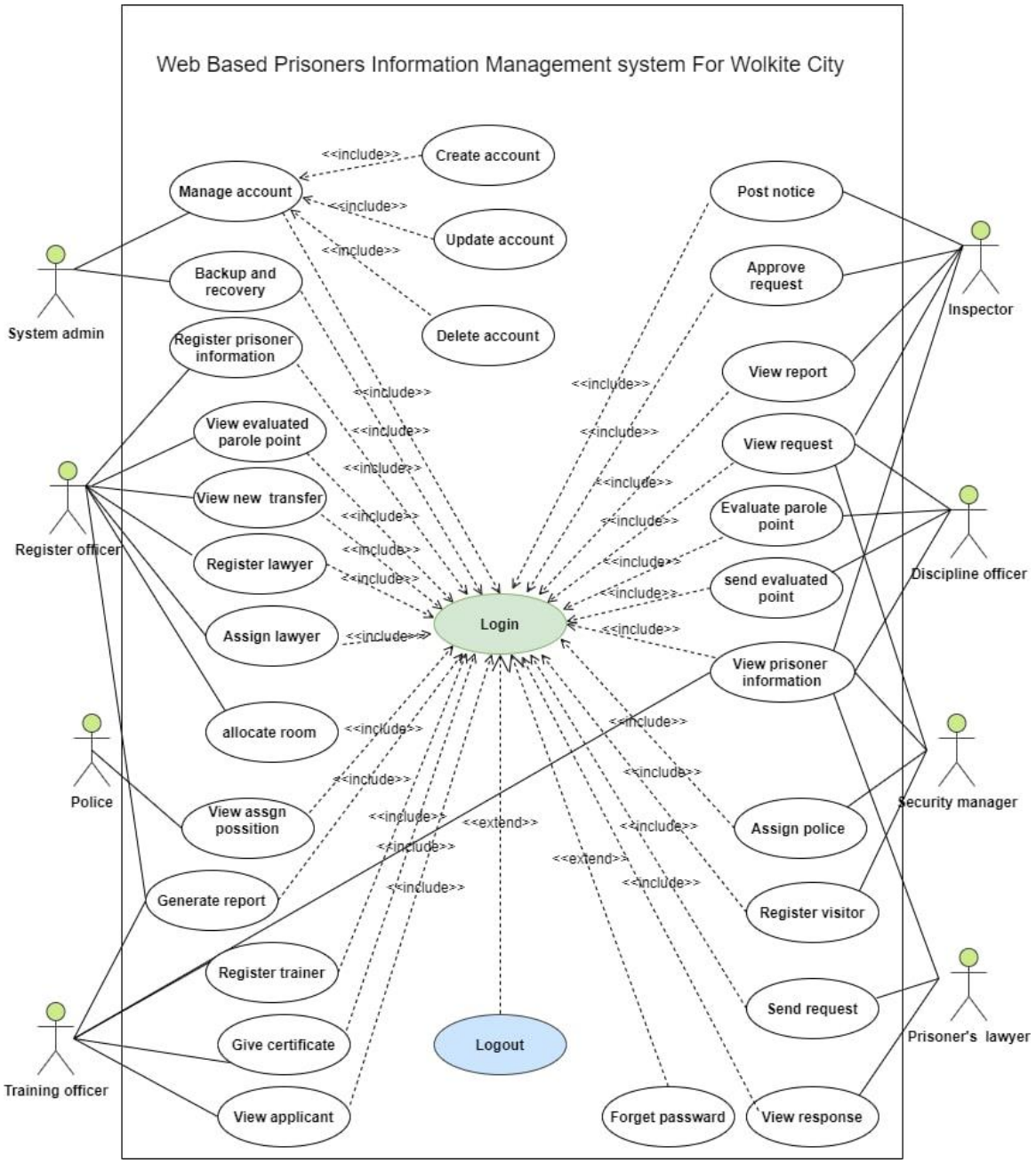


Figure 4. 1 use case diagram of the system

4.2.2. Use case description

System use cases specify the requirements of a system to be developed. They are the starting point for further analysis models and design activities. It is the functionality of the system or the service provided by the system.

Table 4. 1 Login Use case description

Use case Name	Login	
Actor	Inspector, System administer, Security manager, police, Training officer, Registrar officer, prisoner lawyer, and Discipline Officer	
Description:	Indicate that users are authenticated and taken to their own user interface based on the given privilege	
Pre-conditions	Users must have a valid user name and password.	
Basic course of action	User action	System response
	1. First user opens the home page. 2. Then user clicks the login menu. 4. The user fills the login form. 5. The user clicks on the login button.	3. The system will display the login form. 6. The system will check for the validity of the username and password from the database. 7. The system will display the homepage. 8.End-use case
Alternative course of action	A1.If user name or password is invalid. The System display Error Message and return back the user to login form page.	
Post condition	The user is authenticated and taken to their own user interface.	

Table 4. 2 Login Use case description

Use case Name	Create account	
Actor	System administer	
Description:	The administrator creates an account for all actors found in a system	
Pre-conditions	The admin must be an authorized to perform the operation	
Basic course of action	User action	System response
	1. The system admin click on the create account link.	2.The system will display the user Account creation form.

	3. The system admin fills the form.	5. The system will check the filled form. 6. The system will create a user account.
	4. The system admin clicks on create account button.	7. The system will display a successful message. 8. End-use case.
Alternative course of action	A1.If a user misses the required field or entered wrong data, the system notifies "the missed field or wrong data" and the process continues from step 3.	
Post condition	Account Created Successfully	

Table 4. 3 Register prisoner use case description

Use case Name	Register prisoner	
Actor	Register officer	
Description	Registers prisoners	
Pre-conditions	The Register office must login into the system	
Basic course of action	User action	System response
	1.The user clicks on the register prisoner menu 3. The user fills the registration form. 4.The user clicks on the register button	2.The system will display a registration Form 5. The system will check the filled form. 6. The system will display successfully registered message. 7.End use case
Alternative course of action	A1.If invalid data register. Return to step 3 of the flow of events	
Post condition	Prisoners registered to the system	

Table 4. 4 Assign lawyer use case description

Use case Name	Assign lawyer to the prisoner	
Actor	Register officer	
Description	The Register officer assign lawyer for prisoners that needs a lawyer	
Pre-conditions	The Register office must login into the system.	
Basic course of action	User action	System response

	1. The Register officer clicks the assign prisoner link. 3. The Register officer selects the lawyer id. 4. The register officer clicks the assign button. 6. The register officer selects the prisoner who want the lawyer and clicks assign.	2. The system display assign page. 5. The system will display a prisoner select page. 7. The system will display a successful message. 8. End use case
Alternative course of action	A1. If a user misses the required field or entered wrong data, then click the button the system notifies “not assigned successfully”.	
Post condition	lawyer assigned to a prisoner	

Table 4. 5 Generate Report use case description

Use case Name	Generate Report	
Actor	Register officer	
Description	Generate a report specified.	
Pre-conditions	General report.	
Basic course of action	User action	System response
	1. User must be login to the system. 2.user click on generate report button in their pages 4.user fill the provided form	3.The system displays generate report form 5. The system save and store the report. 6. The reports generated successfully.
Alternative course of action	A1.The system displays error message If the generate report form are not fulfill correctly back to step 4	
Post condition	Generate reports from the database	

Table 4. 6 Update prisoners’ information use case description

Use case Name	Update prisoners’ information
Actor	Register officer

Description	Update prisoner information whenever needed.	
Pre-conditions	Prisoner's information record must exist.	
Basic course of action	User action	System response
	1. The user clicks on the edit menu. 3. The user edits any information in the form. 4. The user clicks on the update button.	2.The system will display an update Form. 5. The system will check the filled result. 6. The system will update prisoner information. 7. The system will display a successfully updated message. 8. End-use case
Alternative course of action	A1. If a user misses the required field or entered wrong data, the system notifies "the missed field or wrong data" and the process continues from step 3	
Post condition	The prisoner information updated.	

Table 4. 7 View request use case description

Use case Name	View request	
Actor	Inspector, security manager, training officer and discipline officer	
Description	View any request that comes from prisoner lawyer page	
Pre-conditions	The actor must be login into the system.	
Post-condition	The actor accept or reject request	
Basic course of	User action	System response

action	1. The user clicks on the view request menu. 3. The user clicks on the inbox menu. 5. The user will select from the selection box. 7.User click on OK button	2.The system will display a view request 4. The system will display all Inbox requests. 6. The system will display a selection button contains. ❖ Approve request ❖ Reject request 8. The system will send the response. 9. The system will display a successful message. 10. End use case.

Table 4. 8 Send request use case description

Use case Name	Send request	
Actor	prisoner Lawyer	
Description	Prisoner Lawyer can send any request about prisoner based on different reason like for transfer or for appointment	
Pre-conditions	Prisoner Lawyer Should have an account or registered and assign for a prisoner.	
Basic course of action	User action	System response
	1. The prisoner lawyer clicks on send Request menu.	2.The system will display the link ❖ Transfer
	3. The prisoner lawyer clicks on type of request link. 5. The prisoner lawyer fills the form. 6. The prisoner lawyer clicks on send button.	❖ Appointment ❖ Parole 4. The system display form. 7. The system will check the filled form. 8. The system will display a successful message. 9.End use case
Alternative course of action	A1. If the form is not filled with correct data the system will loop back to step 5 and messaged the prisoner what type of error has been occurred.	
Post condition	System transfer request to the inspector	

4.2.3. Use case Scenario

A scenario is an instance of a use case describing a concrete set of actions. It describes any relevant aspects of the context in which the user is working with the system, including what information the user has on hand when beginning to use the System. It also describes for the user when and where they are working, why they are using the system, and what they need the system to do for them. The following describes scenario of how the user use the systems to perform operations.

1. Scenario name: login

Participant actor: All User

Flow of events for login scenario are listed below:

If users want to login to the system the first step will be enter to the system Home Page then he/she should have login Password and user name, if he/she has click to the login link on the home page then the system display login form. Then the user must fill the username and password. If he/she is an authenticated user, he/she did some operation they wants to do. If he/she is not an authenticated user, he/she must enter a valid user name and password.

2. Scenario name: Register prisoner's information.

Participant actor: Register officer

Flow of events for Register prisoner's information scenario are listed below:

If the Register officer wants to register prisoners first run the system, then the system displays the system home page and then register officer must login first by filling the username and password then click the login button.

If the username and password are correct, the home page displayed and then register officer click the prisoner registration button.

Then the Register officer registers the prisoner information and other operation, then after the operation completed the register officer must be logout from the system.

If a username and password are not correct must reenter the correct username and password.

3. Scenario name: send a request.

Participant actor: prisoner lawyer

Flow of events for send a request scenario are listed below:

If Prisoner lawyer wants to send request first he/she must run the system and fill the username and password, then click the login button.

If the Username and password are correct the home page displays then click prisoner send request button.

Then system display transfer, appointment, and parole link, then prisoner lawyer click on type of request he/she want to perform, then click the send button.

If a username and password are not correct must reenter the correct username and password.

4.3. Object Model

Object Model refers to a visual representation of software or systems' objects, attributes, actions, and relationships. The basic factors of an object model are classes and objects. An object model helps describe or define a software/system in terms of objects and classes. It is a visual representation of a system's objects, actions, and associated attributes.

4.3.1. Class Diagram

Class diagram in the Unified Modeling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects. Class diagrams are the blueprints of your system or subsystem. You can use class diagrams to model the objects that make up the system, to display the relationships between the objects, and to describe what those objects do and the services that they provide. Class diagrams are useful in many stages of system design. You can use class diagrams to visualize, specify, and document structural features in your models.

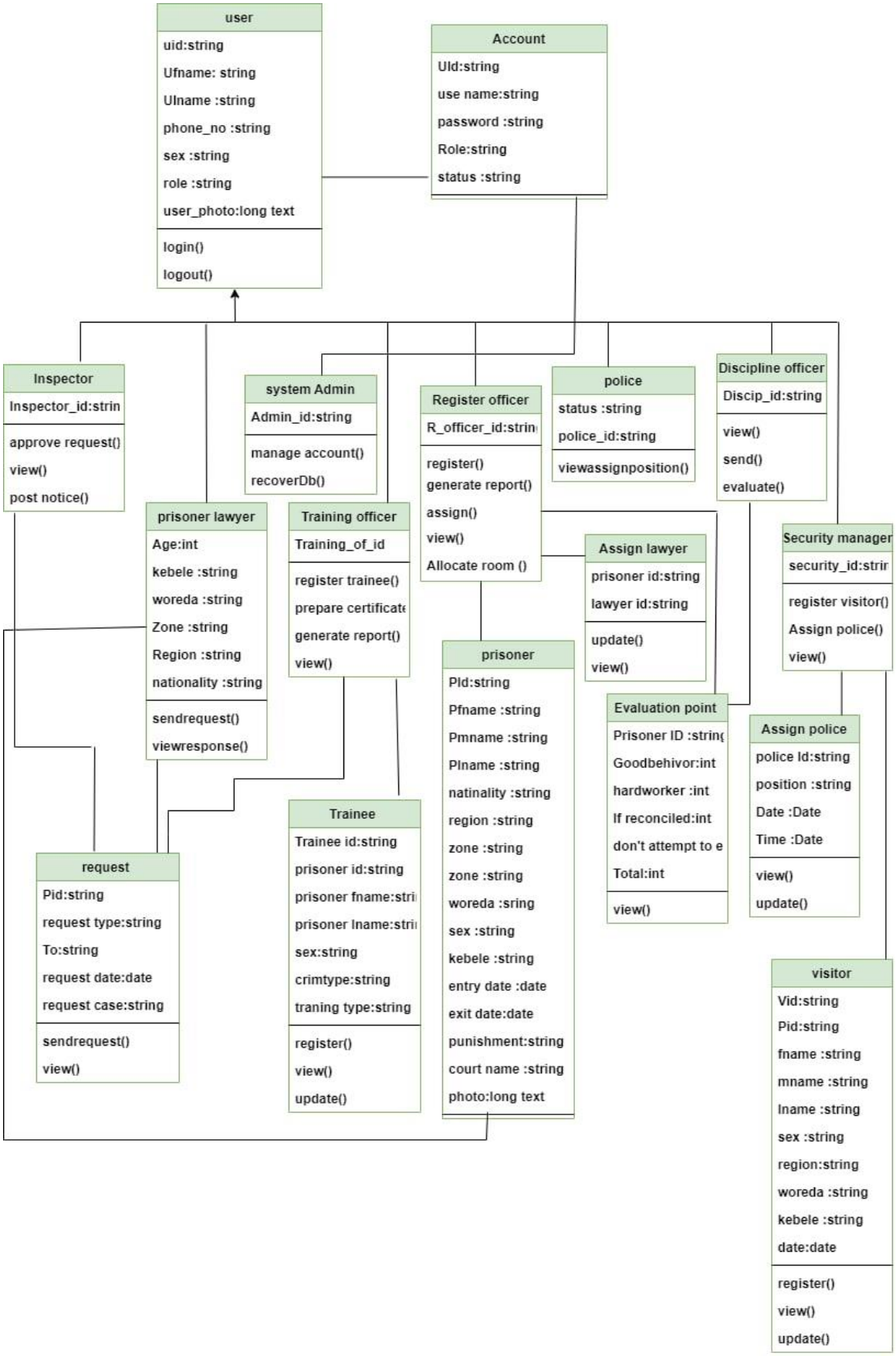


Figure 4. 2 Class diagram of the system

4.3.2. Data Dictionary

Data dictionaries are used to provide detailed information about the contents of a data-set or database, such as the names of measured variables, their data types or formats, and text descriptions. We are going to mention it by using tabular form.

Table 4. 9 User data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
User_id	String	15	Primary key	Not Null
User_Fname	String	20		Not Null
User_Fname	String	20		Not Null
Phone number	String	13		Not Null
Sex	String	10		Not Null
Role	String	20		Not Null
Photo	Long text	Max		Not Null

Table 4. 10 Police data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
Police _id	String	15	Primary key	Not Null
Police_fname	String	20		Not Null
Police_lname	String	20		Not Null
Phone number	String	13		Not Null
Sex	String	10		Not Null
Role	String	20		Not Null
Photo	Long text	Max		Not Null
Police status	String	20		Not Null

Table 4. 11 lawyer data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
Lawyer _id	String	15	Primary key	Not Null
Lawyer_fname	String	20		Not Null
Lawyer_lname	String	20		Not Null
Age	int	20		Not Null
Kebele	String	20		Not Null
Woreda	String	20		Not Null
Zone	String	20		Not Null
Region	String	20		Not Null
Nationality	String	20		Not Null
Phone number	String	13		Not Null
Sex	String	10		Not Null
Role	String	20		Not Null
Photo	Long Text	Max		Not Null

Table 4. 12 Account data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
Account id	String	20	Primary key	Not Null
User name	String	30		Not Null
Password	String	20		Not Null
Role	String	20		Not Null
Status	String	20		Not Null

Table 4. 13 prisoner data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
Prisoner id	String	15	Primary key	Not Null
Prisoner_Fname	String	15		Not Null
Prisoner_Mname	String	15		Not Null
Prisoner_Lname	String	15		Not Null
Nationality	String	20		Not Null
Region	String	30		Not Null
Woreda	String	20		Not Null
Kebele	String	20		Not Null
Age	Int	15		Not Null
sex	String	15		Not Null
Entry date	Date	20		Not Null
Exit date	Date	20		Not Null
Year of punishment	String	20		Not Null
Court name	String	20		Not Null

Crime type	String	20		Not Null
Photo	Long text	Max		Not Null
status	String	20		Not Null

Table 4. 14 Request data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
Prisoner id	String	15	Primary key	Not Null
Request type	String	20		Not Null
To	String	20		Not Null
Request date	Date	20		Not Null
Request case	String	30		Not Null

Table 4. 15 Assign police data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
Police id	String	15	primary key	Not Null
Assign position	String	25		Not Null
Date	Date	20		Not Null
Time	Date	10		Not Null

Table 4. 16 Evaluation data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
Prisoner id	String	15	Primary key	Not Null
Good behavior	int	20		Not Null
Hard worker	int	20		Not Null
If Reconciled	int	20		Not Null
Don’t attempt to escape	int	20		Not Null
Total	int	20		Not Null

Table 4. 17 Trainee data dictionary

Attributes	Data type	Data size	Key constraint	Constraint
Trainee id	String	15	Primary key	Not Null
Prisoner id	String	15	Foreign key	Not Null
F_name	String	15		Not Null
L_name	String	15		Not Null
sex	String	15		Not Null
Training type	String	20		Not Null
Crime type	String	10		Not Null

4.4. Dynamic Model

4.4.1. Sequence Diagram

Sequence diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when. A sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development.

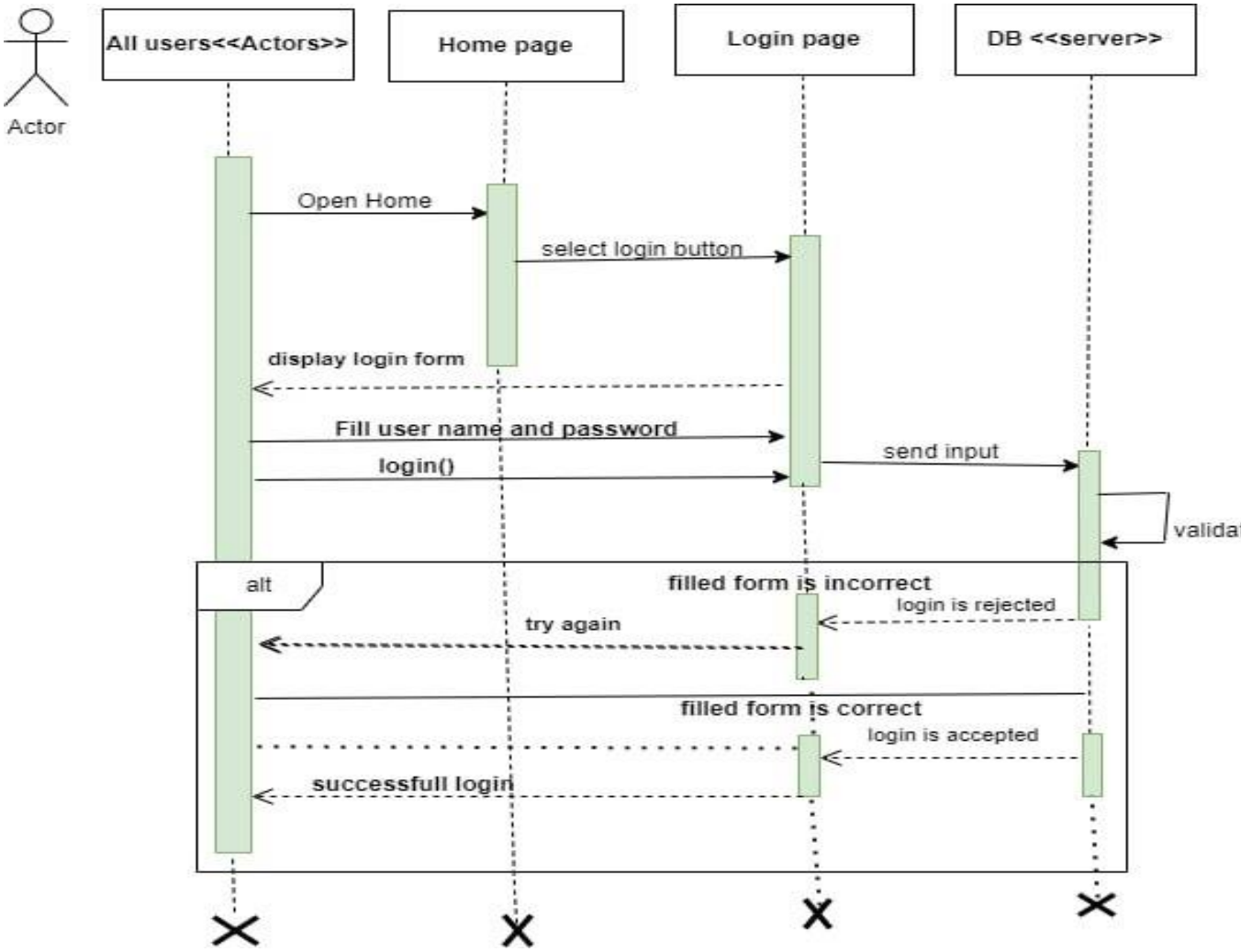


Figure 4. 3 Sequence diagram for login page

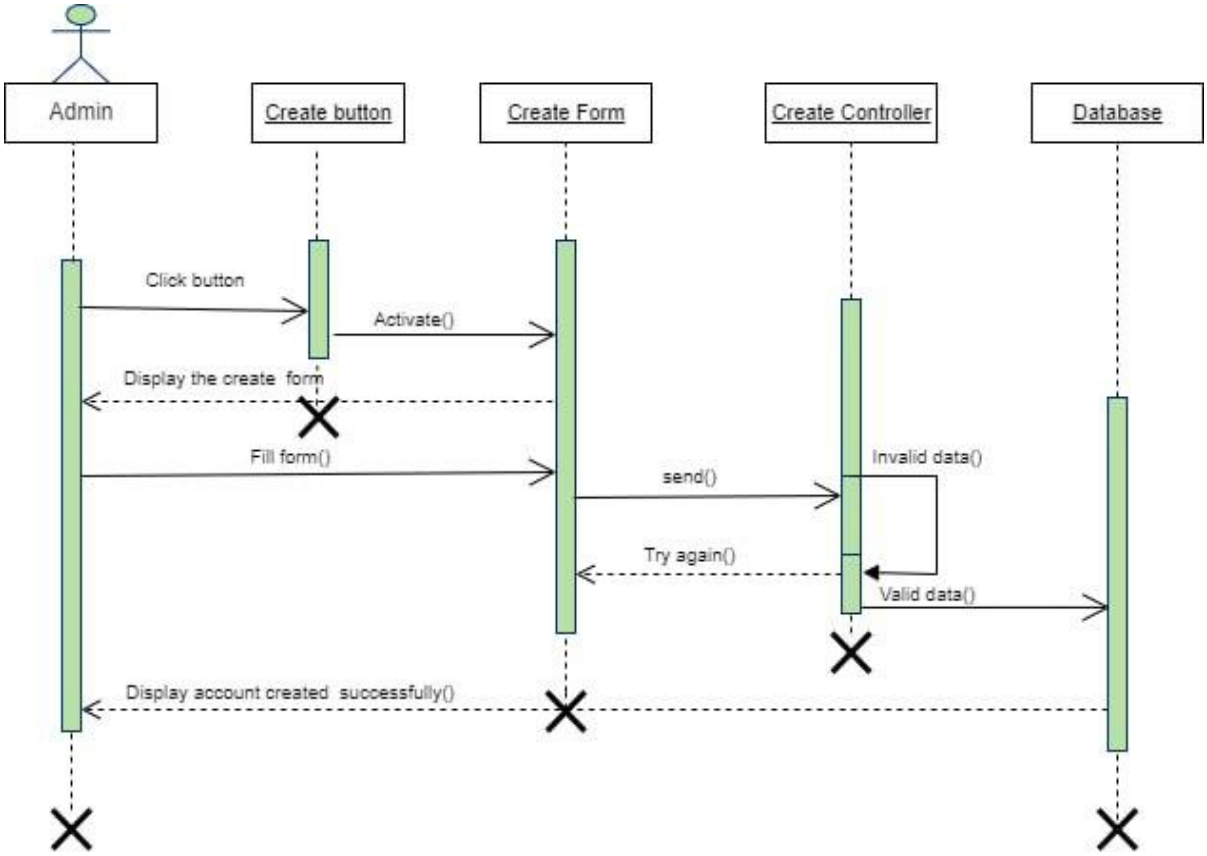


Figure 4. 4 Sequence diagram for create account

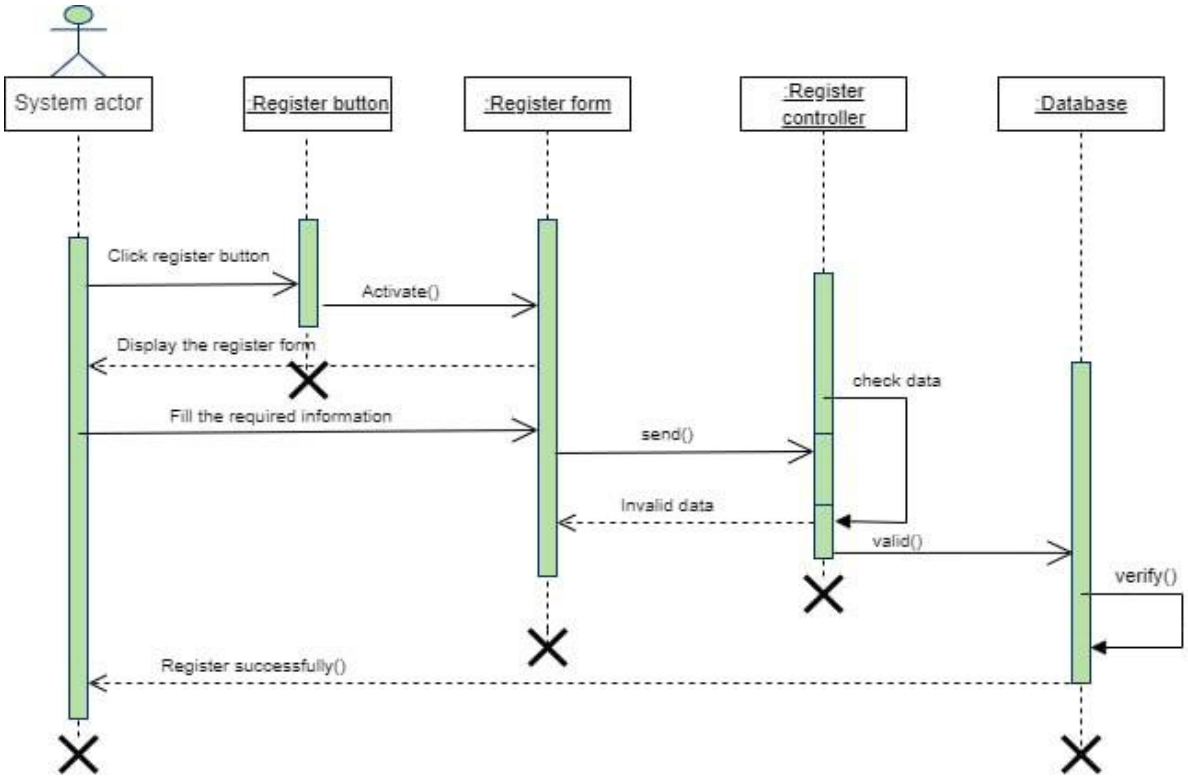


Figure 4. 5 Sequence diagram for register

4.4.2. Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. Activity diagram describes the flow of control in a system. So it consists of activities and links. The flow can be sequential, concurrent or branched. Activity diagrams are used to visualize the flow of controls in a system. This is prepared to have an idea of how the system will work when executed. Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system. [3]

So the purposes can be described as:

- Draw the activity flow of a system.
- Describe the sequence from one activity to another.
- Describe the parallel, branched and concurrent flow of the system.

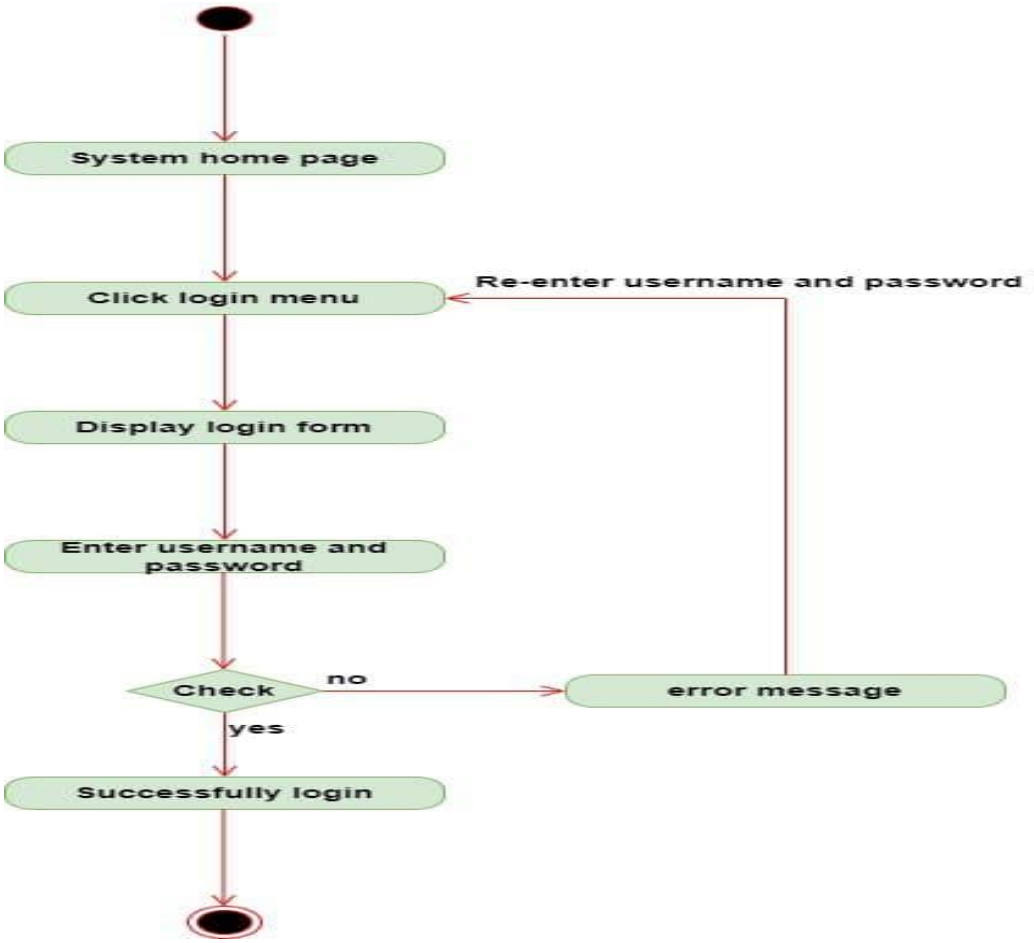


Figure 4. 6 Activity diagram for Login page

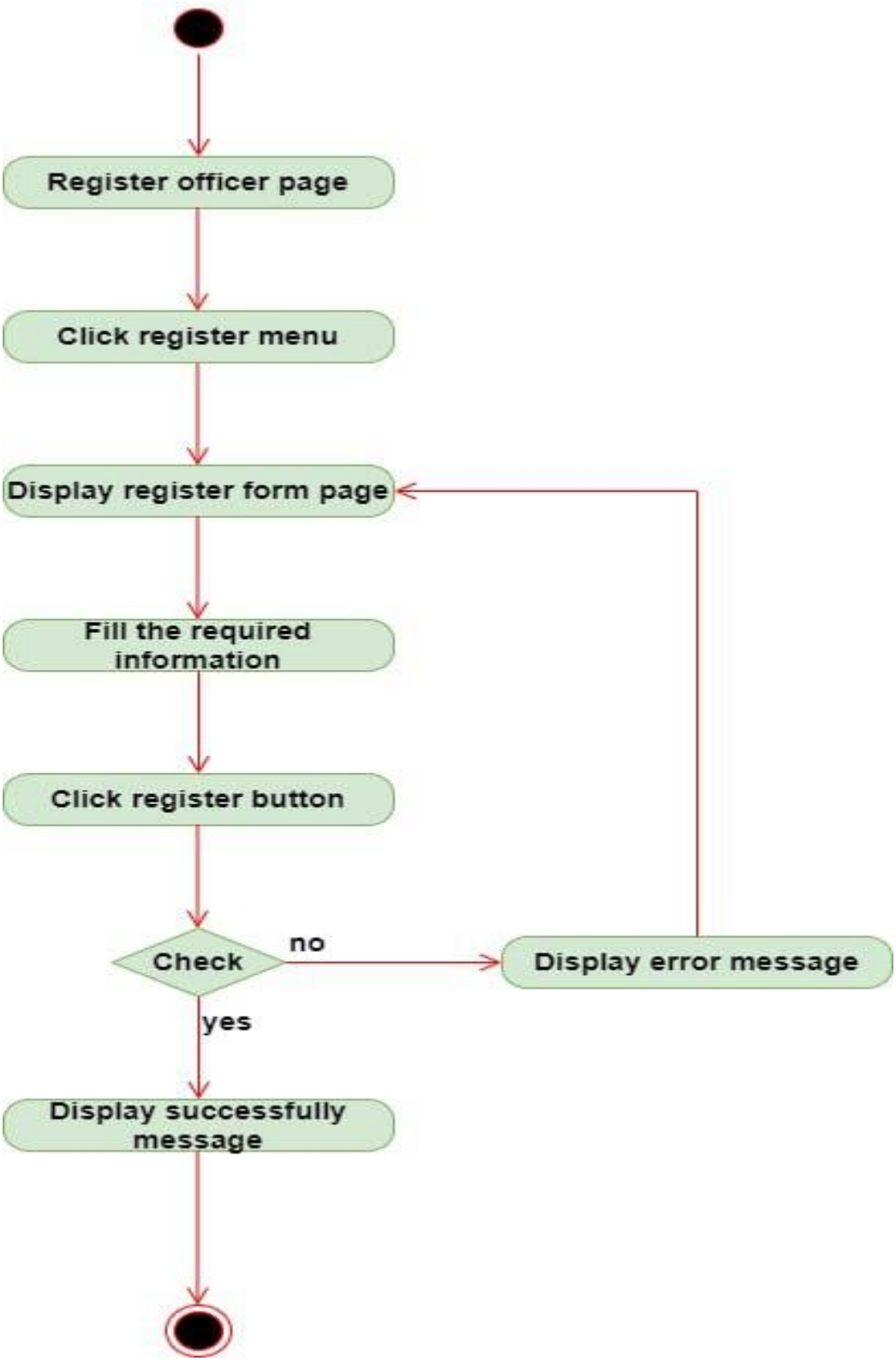


Figure 4. 7 Activity diagram for prisoner registration

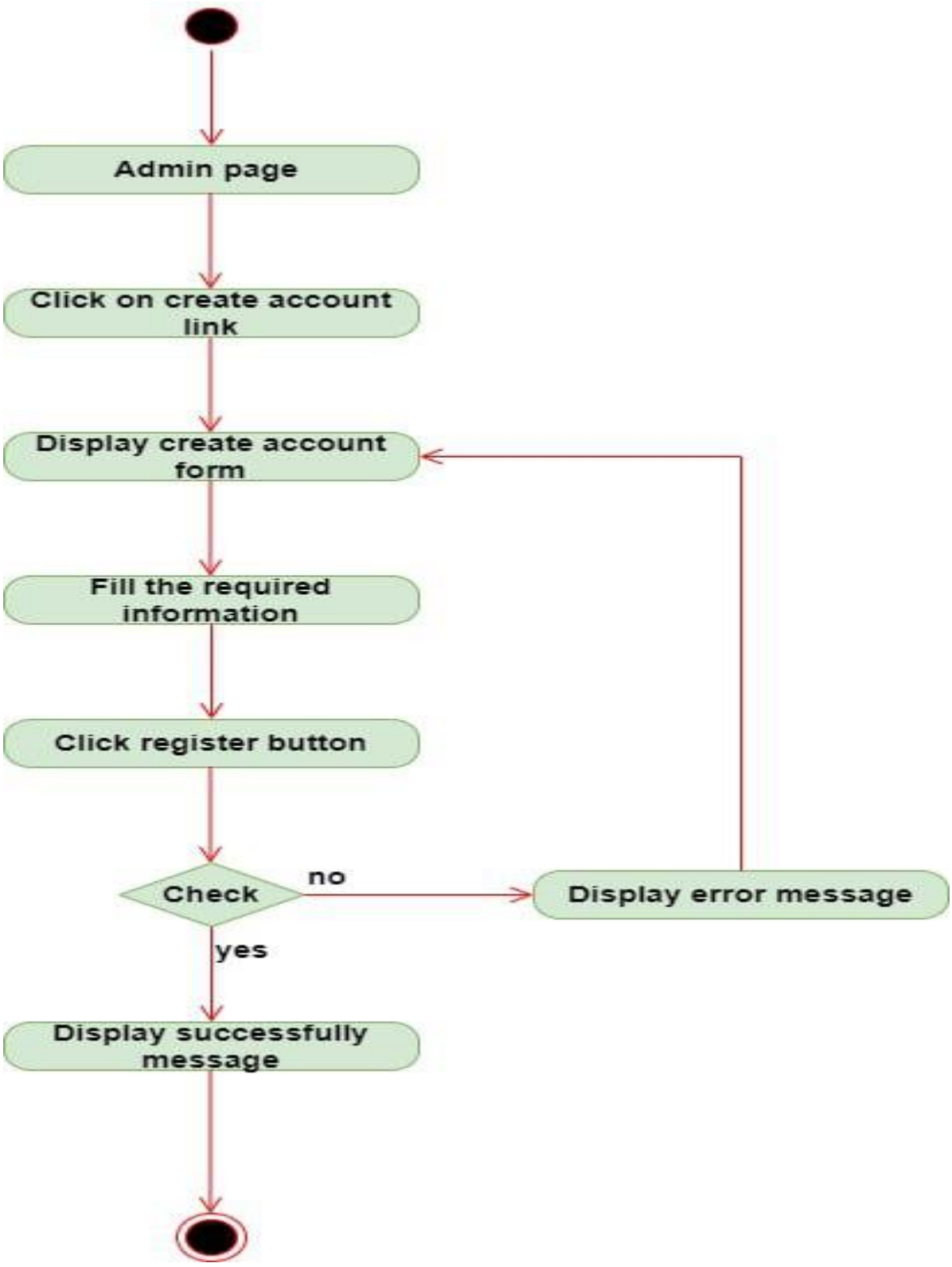


Figure 4. 8 Activity diagram for create account

4.4.3. State Chart Diagram

A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It's a behavioral diagram and it represents the behavior using finite state transitions. State diagrams are also referred to as state machines and state-chart diagrams. These terms are often used interchangeably. So simply, a state diagram is used to model the dynamic behavior of a class in response to time and changing external stimuli. State chart diagrams show the various states that an object goes through, as well as the events that cause a transition from one state to another. The common model elements that state chart diagrams contain are states, start, end states and transitions.

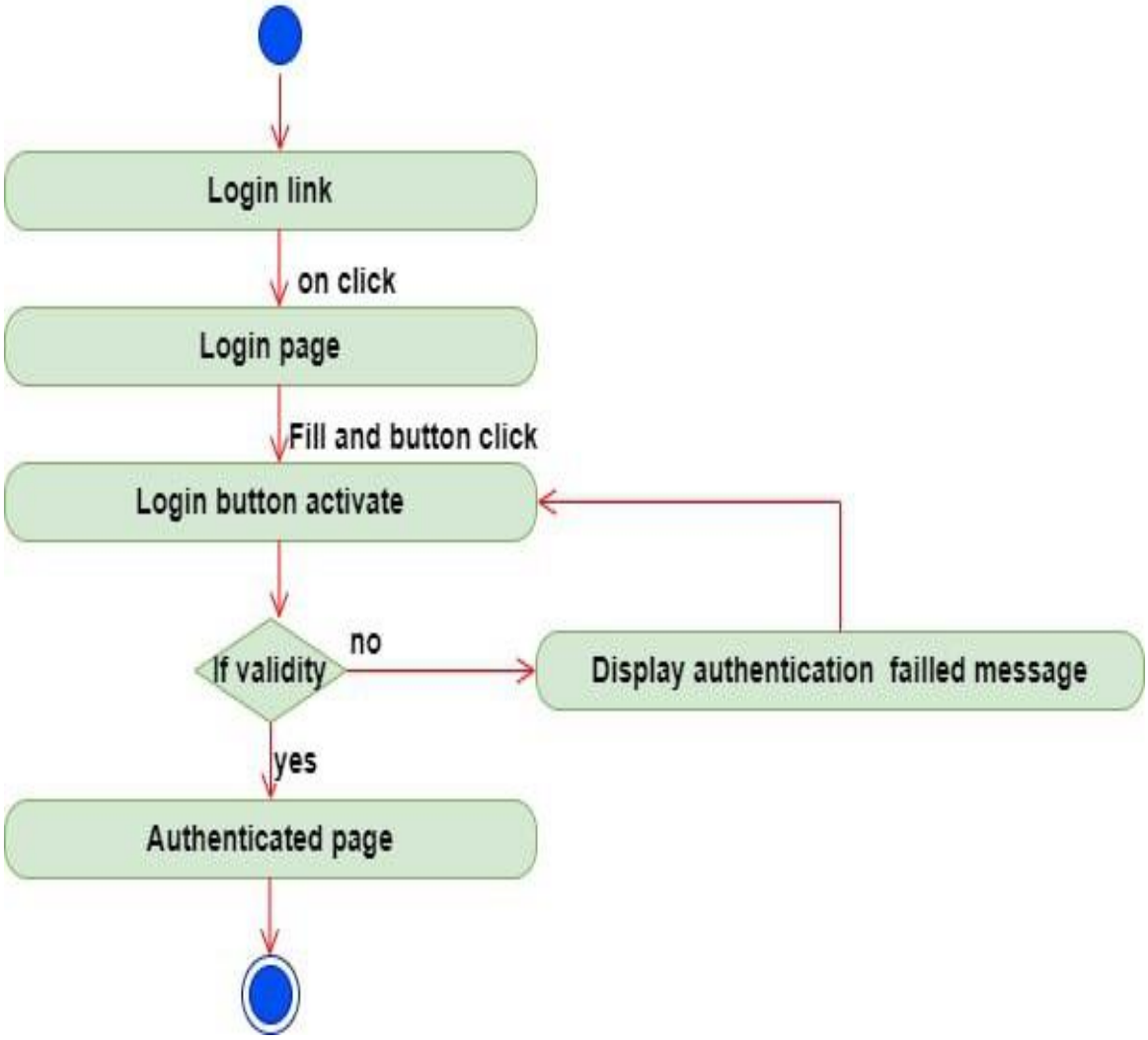


Figure 4. 9 State chart diagram for login

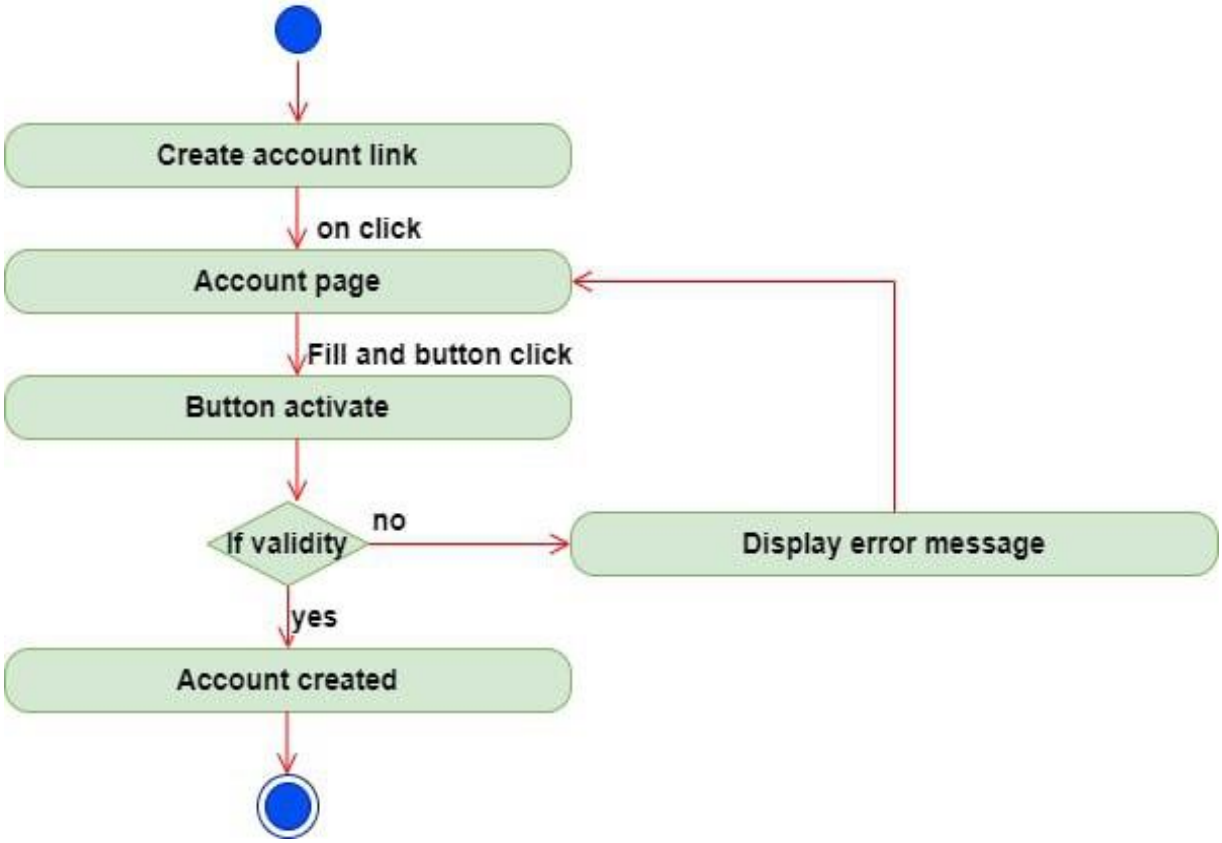


Figure 4. 10 State chart diagram for create account

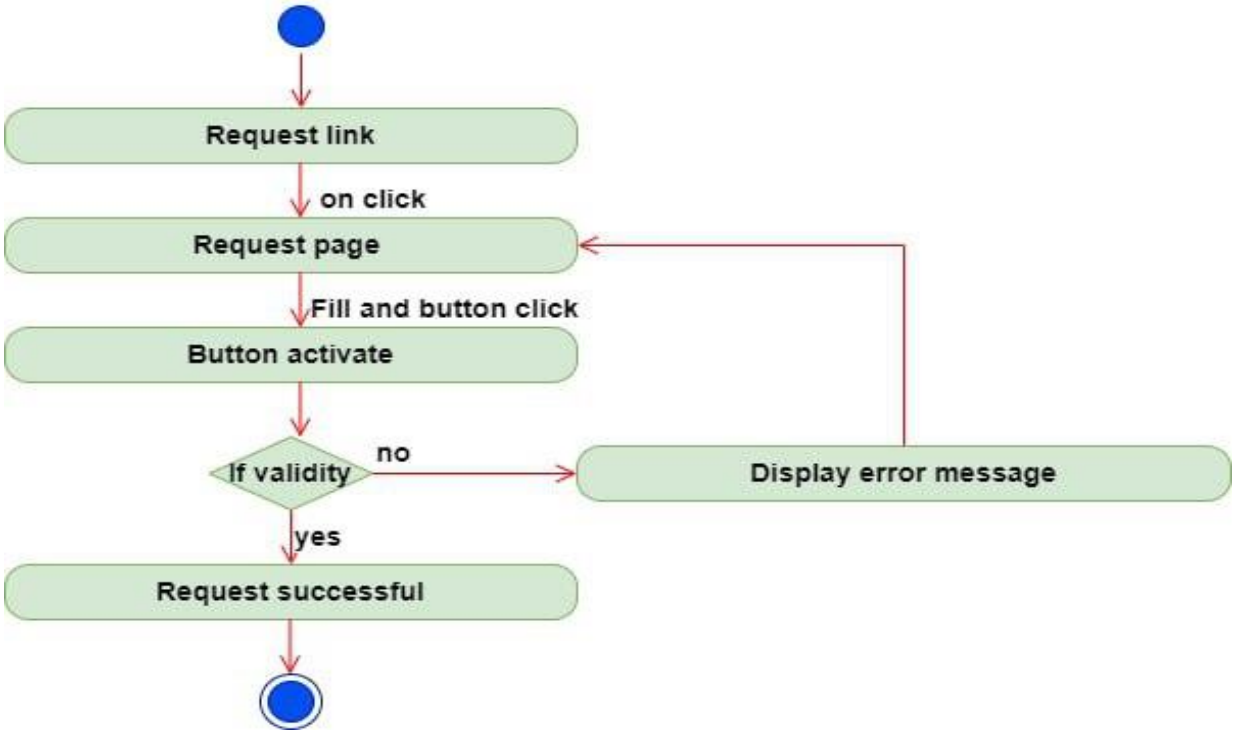


Figure 4.11state chart diagram for send request

CHAPTER FIVE

5. SYSTEM DESIGN

Systems design is the process of defining the architecture, product, modules, interfaces, and data for a system to satisfy specified requirements. During system design we focus on the processes, data structures, and software and hardware components necessary to implement it. The challenge of system design is that many conflicting criteria and constraints need to be met when decomposing the system. It is the process of defining and developing systems to satisfy specified requirements of the user. The basic study of system design is the understanding of component parts and their subsequent interaction with one another. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization. So, the proposed system design lead to transformation of the analysis model into a system design model.

5.1. Design Goals

Describes the important system qualities and defines the values against which options are evaluated (i.e.it describes the qualities of the system that developers should optimize). The objectives of design are to model the system with high quality. Implementing a high-quality system depends on the nature of design created by the designer. If one wants to changes the system after it has been put in-to operation depends on the quality of the system design. So if the system is designed effectively, it will be easy to make changes to it. The design goals of our system are the following:

User interface: The interface of the proposed system is simple to understand, easy to use and user-friendly interface and users of the system easily use and perform their task. We point up the following considerations when designing our interface:

- **Interface simplicity**: when our system users use our systems, they will not get difficulty when using it in terms of interface.
- **Conventionality**: we design our system by considering universal naming, alignment, symbols and icons which enable our users to use our system without requiring any learning time.
- **Using attractive and mining full colors**: in order to make our users to be stick to our system when he or she uses our system without frustration and stress.
- Any user who have basic computer skill will use our proposed system.

Hardware Consideration: The system should run in any desktop and personal computer with any flat form because we will use PHP server-side scripting language to implement the system that support platform independent. And needs server to start the system and contain database. Hardware requirement include:

- Computer with 4GB RAM and above
- Server at list one

Security Issues: Our system keeps security by using effectively different mechanisms like:

Database security: this is done to protect unauthorized user from access data and information by giving appropriate privilege to each user according their role.

- Since the system hold important information (data), the system requires strong security features to protect that valuable information, i.e. not allow other users or unauthorized users to access data that has no right to access it.
- To prevent brute force, attack the system control a maximum of 5 login attempts.

Performance Consideration: the performance of the system is depending on the time required to perform a given task and memory required for that task. The factors such as avoiding redundancy of codes increases the time of operation and memory space of the proposed system. Proposed system of web based prisoner information management system should respond fast with high throughput, i.e.it should perform the task quickly because we use efficient algorithm to develop it.

Error Handling and Validation: error handling is the most important thing which need to be considered when developing a system. Errors could rise from users and the system. Errors that occurred from the wrong doing of users will be handled by appropriate exception handling mechanisms. Generally, if an error occurs, the system will identify the error and notify the user so that he/she can take the appropriate corrections rather than terminating the system.

Quality Issues: Since the system is used for the prisoner information management process it should have qualities such as accuracy, robustness and reliability.

Requirement for reliability includes:

User requirements: the system should be work as the requirements of the users.

System portability: the ability to work on a different personal computer and mobile phone

Backup and Recovery: backup and recovery refers to the process of backing up data in case of a loss and setting up systems that allow that data recovery due to data loss. Backing up data requires copying and archiving computer data, so that it is accessible in case of data deletion or corruption. Data from an earlier time may only be recovered if it has been backed up. In proposed system the system administrator can back up the data when the system needs a backup.

Documentation: The System has well-defined document which helps to easily maintain the system and we will also prepare a short and precise help file on how to use the system for the system users. It will have a helping page to guide the user of the system and to show the process how they will have to use it.

5.2. Current System Architecture

The existing system of prisoner information management system is not web based and hence there is no current software architecture that will be considered. As a result, we only describe the software architecture of the proposed system.

5.2. Proposed System Architecture

The proposed system of Web Based Prisoner Information Management System expected to replace the existing manual and semi-automated system by an automated system in many aspects. The architecture used for the Web Based Prisoner Information Management System is a three tier client/server architecture. A three-tier client/server architecture is a type of multi-tier computing architecture in which an entire application is distributed across three different computing layers or tiers. It divides the presentation, application logic and data processing layers across client and server devices.

Presentation layer: is client layer (User Interface) and topmost layer of the application. This is the layer we see when we use the proposed system. The main functionality of this layer is to communicate with the application layer. This layer passes the information which given in terms of keyboard action and mouse click to the application layer. For example when user want to login into WBPIMS first see two text box and login button to enter user-name and password and click on login.

Application (Middle) layer: it is application layer which interacts with data layer and sends information retrieved from database toward to presentation layer. It acts as the mediator between presentation layer and data layer. From above example once, the user

clicks on login button application layer interact with the database and send information towards to presentation layer.

Database (Data) layer: This is used to store data entered by the user. Generally client of our system use browser to access the system using the internet. In this case when the user enters input and takes certain action application server process client request to interacting with the database server. Below figure is proposed system architecture.

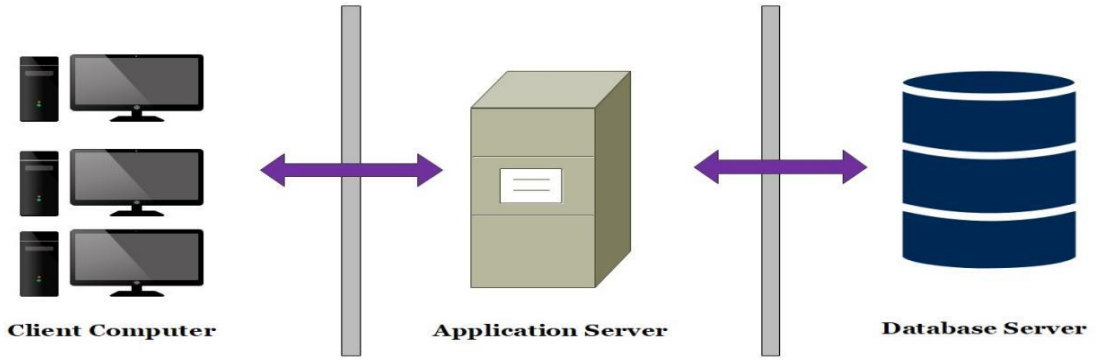


Figure 5. 11 Proposed system architecture

5.2.1. Subsystem Decomposition and Description

A subsystem decomposition is the activity of identifying subsystems, their services, and their relationships with each other. Subsystem decomposition will help to reduce the complexity of the system. The correct decomposition is critical. A wrong decomposition means wrong architecture, which in turn inflicts a horrendous pain in the future, often leading to a complete rewrite of the system. Subsystem decomposition for proposed system have significant advantage to make the system clear and understandable by dividing the whole system into subsystems and components. Subsystem decomposition of proposed system refers to the process by which a complex system is broken down into parts that are easier to understand, program, and maintain. The major subsystems identified as following:

Registration subsystem

- Register prisoner
- Register prisoner lawyer
- Register trainers
- Register user
- Register visitor

Assign subsystem

- Assign lawyer
- Assign police
- View assign position and time

View subsystem

- View prisoner's information
- View new transfer
- View user information

Request Subsystem

- View request
- Send request
- View response
- Approve request

Account subsystem

- Create an account
- Update account
- Delete account

Report Subsystem

- View report
- Generate report

Parole point Subsystem

- View evaluated parole point
- Evaluate parole point
- Send evaluated parole point

Certificate Subsystem

- Give certificate
- View certificate

Database Connection Subsystem

- Used for established connection between class and database management system

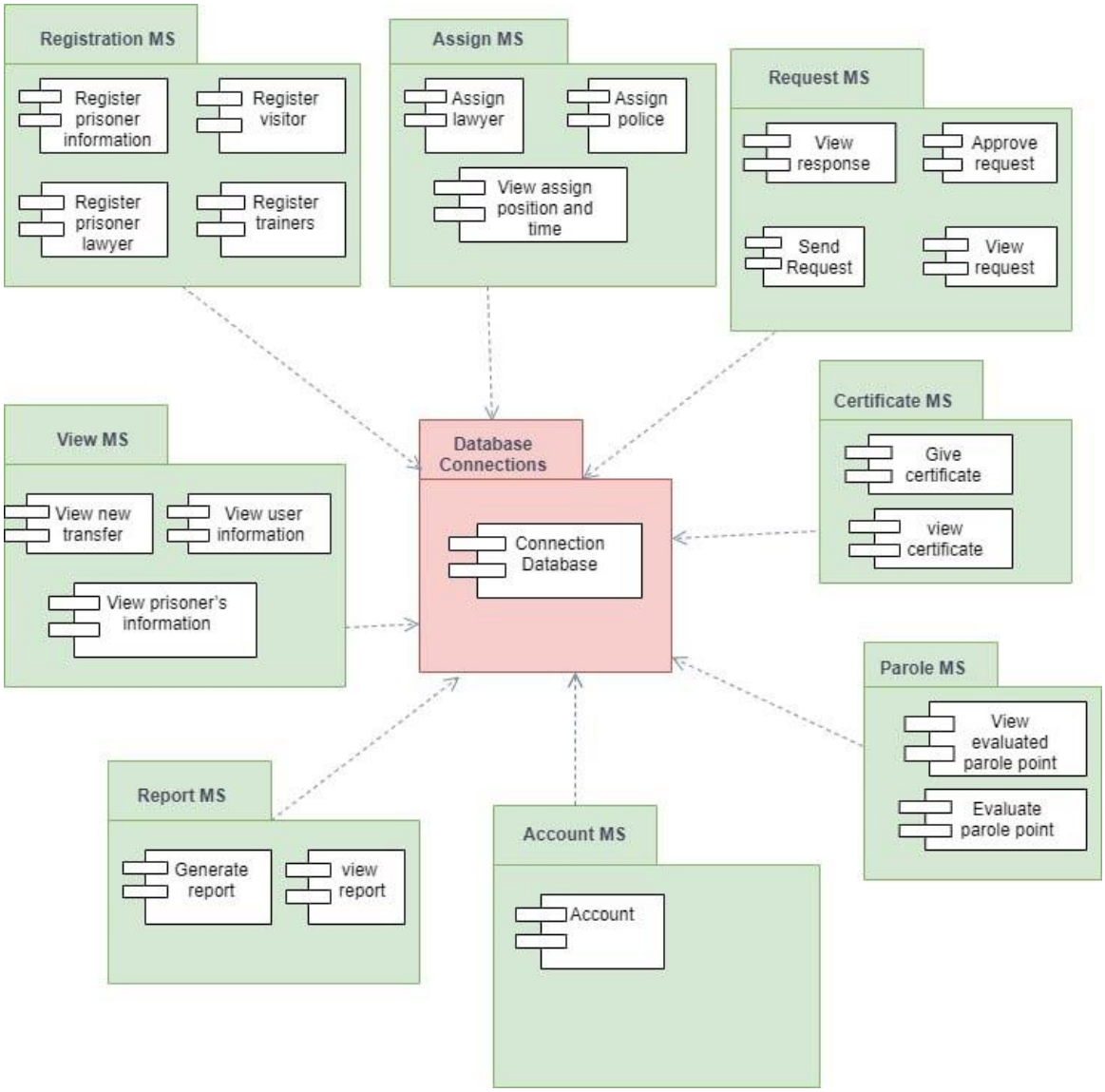


Figure 5.2 component diagram

5.2.2. Hardware/Software Mapping

Deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. This UML diagrams help model the hardware topology of a system compared to other UML diagram types which mostly outline the logical components of a system. UML deployment diagram of proposed system is an implementation diagram that shows the structure of a run-time system. From it, we can learn about the physical relationships among software and hardware components and the distribution of components to processing nodes.

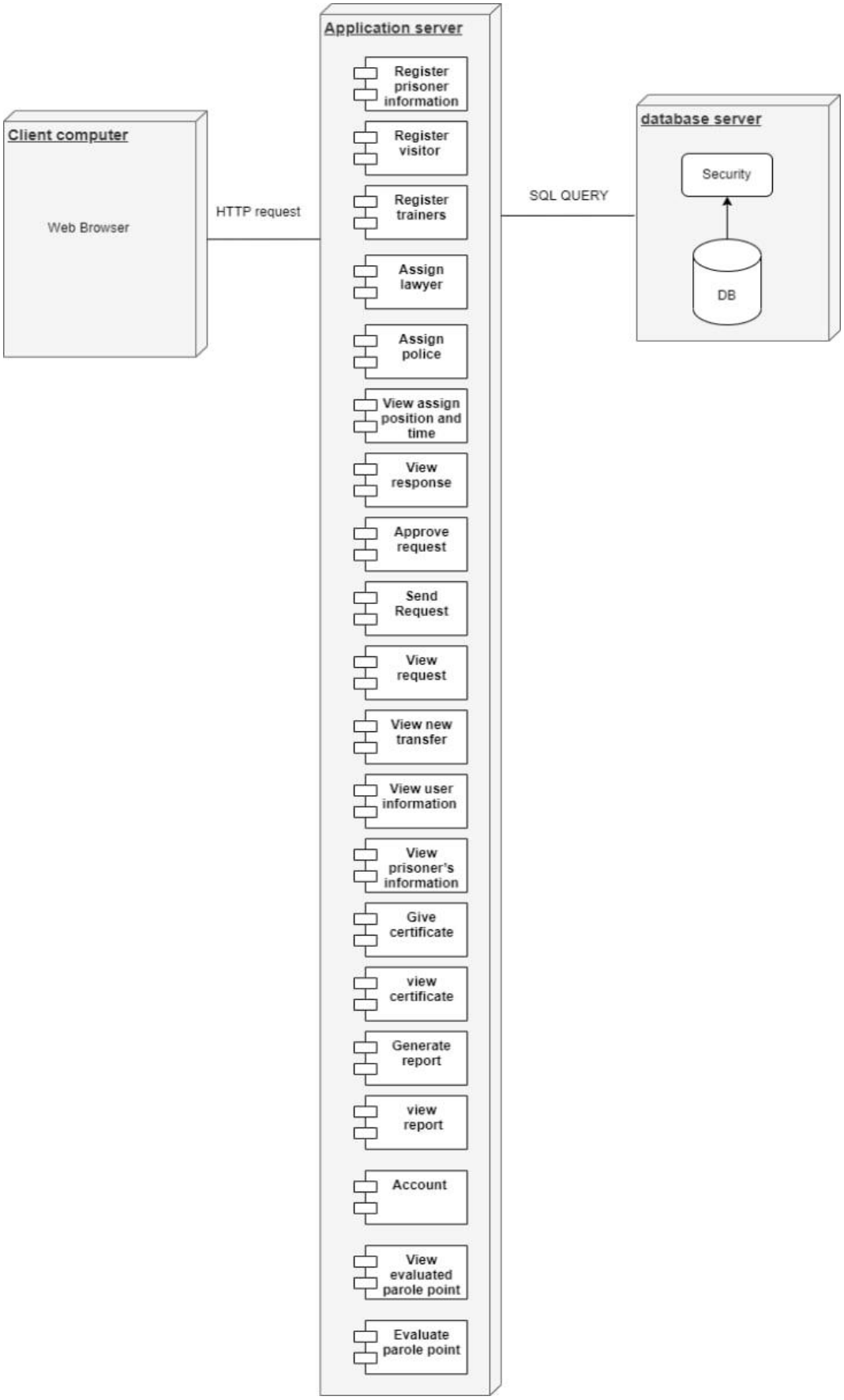


Figure 5. 3 Deployment diagram

5.2.3. Detailed Class Diagram

A detailed class diagram in the Unified Modeling Language (UML) 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.

The detailed class diagram of proposed system look like this:

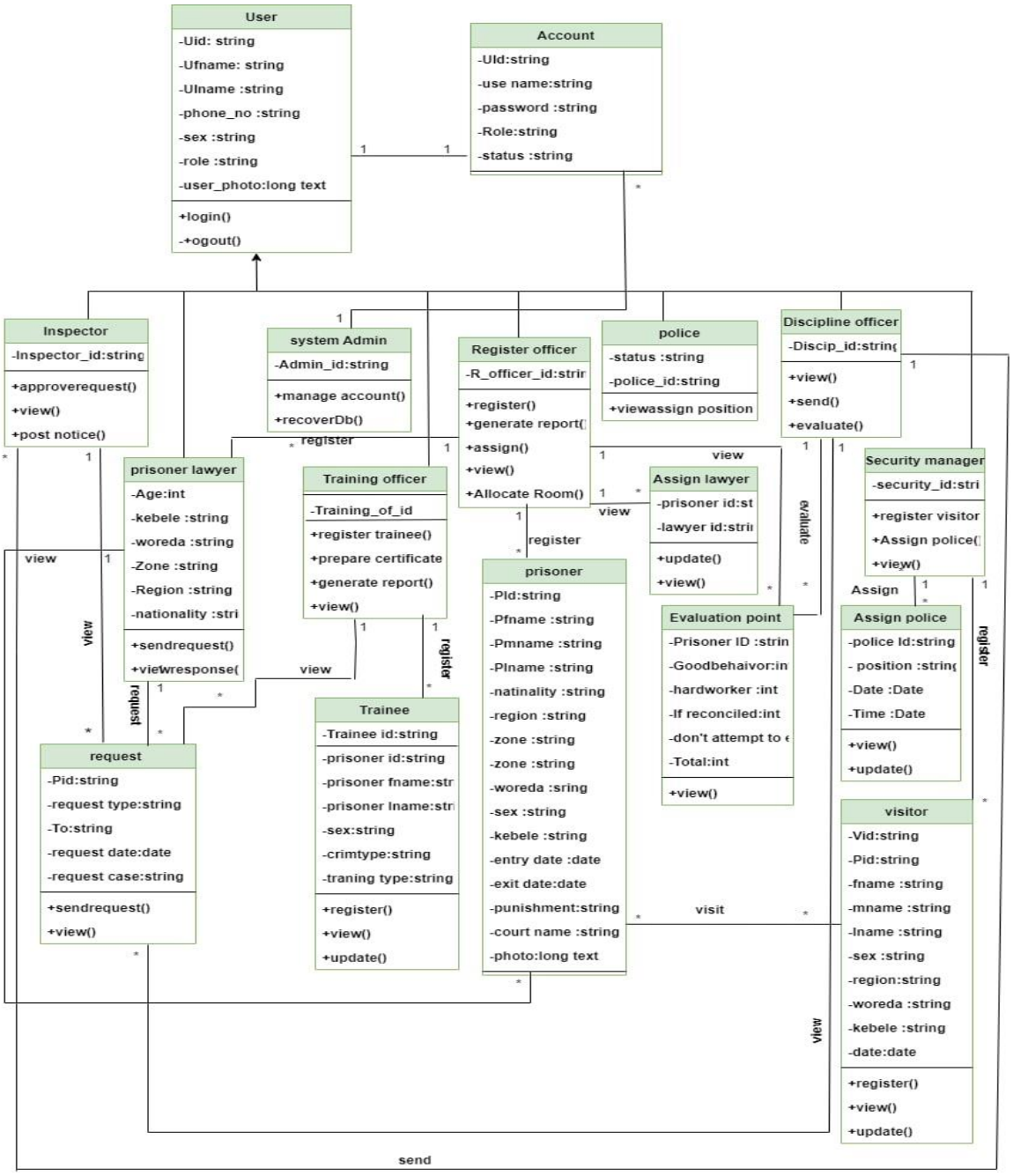


Figure 5. 4. Detailed Class Diagram

5.2.4. Persistent Data Management

Persistent modeling deals with how the persistent data that is the files with a database are stored and managed. The overall information of the prisoner information management system depend on persistent data. So it must be stored on a secure database system. The persistence classes are used to store the most important and permanent information of the proposed system. The following tables indicate the persistence of data management that are used in the proposed system. [4]

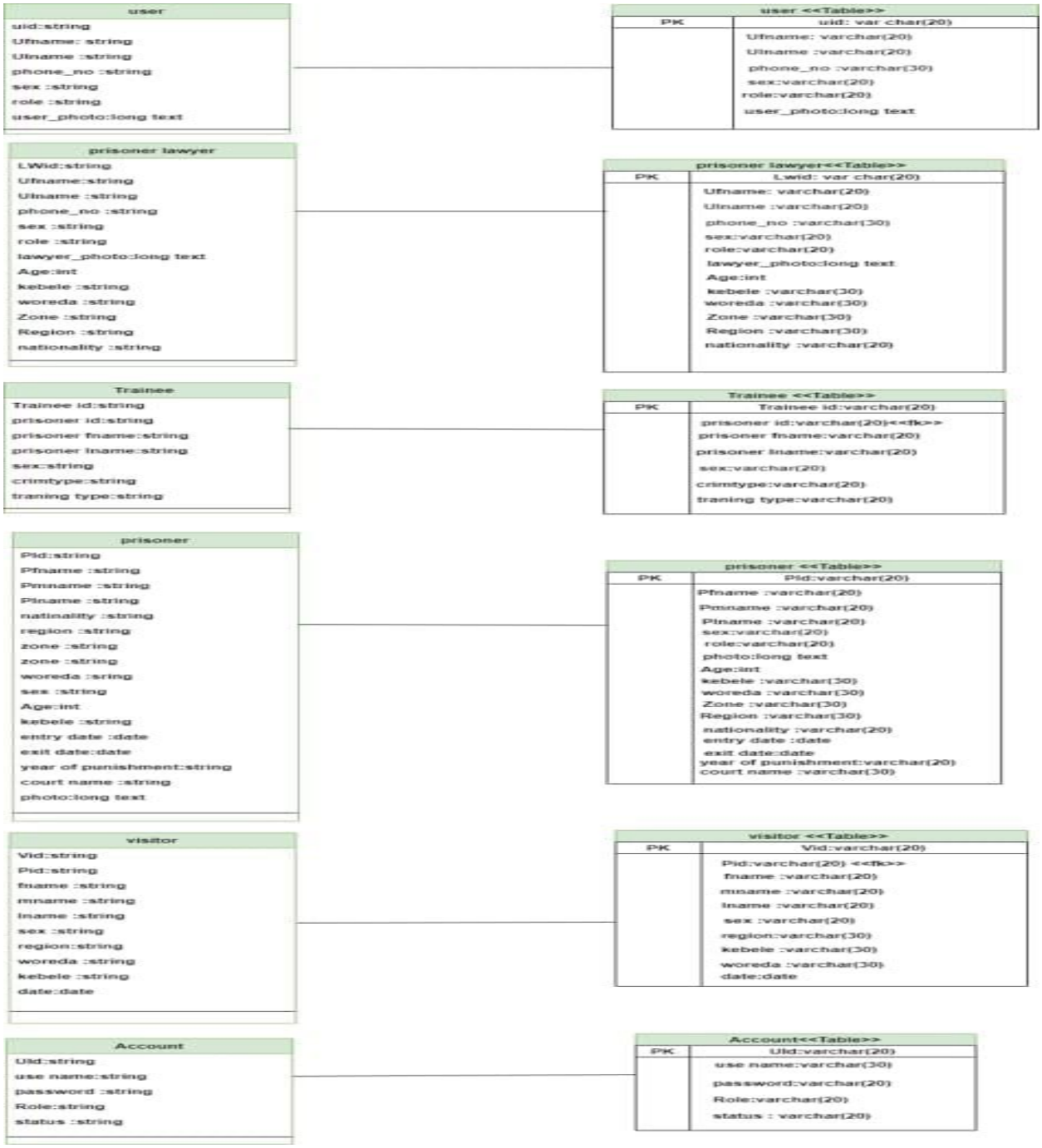


Figure 5.5 Persistent Data Management

5.2.5. Access Control and Security

In our system, different actors have access to different information and data. Access control and security specify which users are granted access and certain privileges to systems, resources or information. This access control is verified by a username and password. The system authenticated the user. The proposed system follows a multi-user system. In a multi-user system, different actors have access to different functions and data. The prisoner information management system has the following:

Table 5. 1 1Access privilege

Actors	**Operation**								
	Manage account	Register	Assign	Request	Report	Parole	approve	Certificate	Backup and recovery
Admin	Manage	……..	……..	……..	……..	……..	……..	……..	Recover database
Register officer	……..	RU….	Assign	……..	Generate	VIEW	……..	……..	……..
Inspector	……..	…….	……..	VIEW	VIEW	……..	AR……	……..	……..
Training officer	……..	RP….	……..	VIEW	Generate	……..	……..	Generate	……..
Prisoner lawyer	……..	…….	……..	Send	……..	……..	……..	……..	……..
Discipline officer	……..	…….	……..	VIEW	……..	Evaluate	…….	……..	……..
Police	……..		VIEW	……..	……..	……..	……..	……..	……..
Security manager	……..	RV….	Assign	VIEW	……..	……..	……..	……..	……..

5.3. Packages

A package diagram in the UML Language depicts the dependencies between the packages that make up a model. It describes the decomposition of subsystems into packages and the file organization of the code. In previous section we were try to define the subsystems of our system, in this part we are going to describe the dependency of each subsystems to other sub system which are found in the same package.

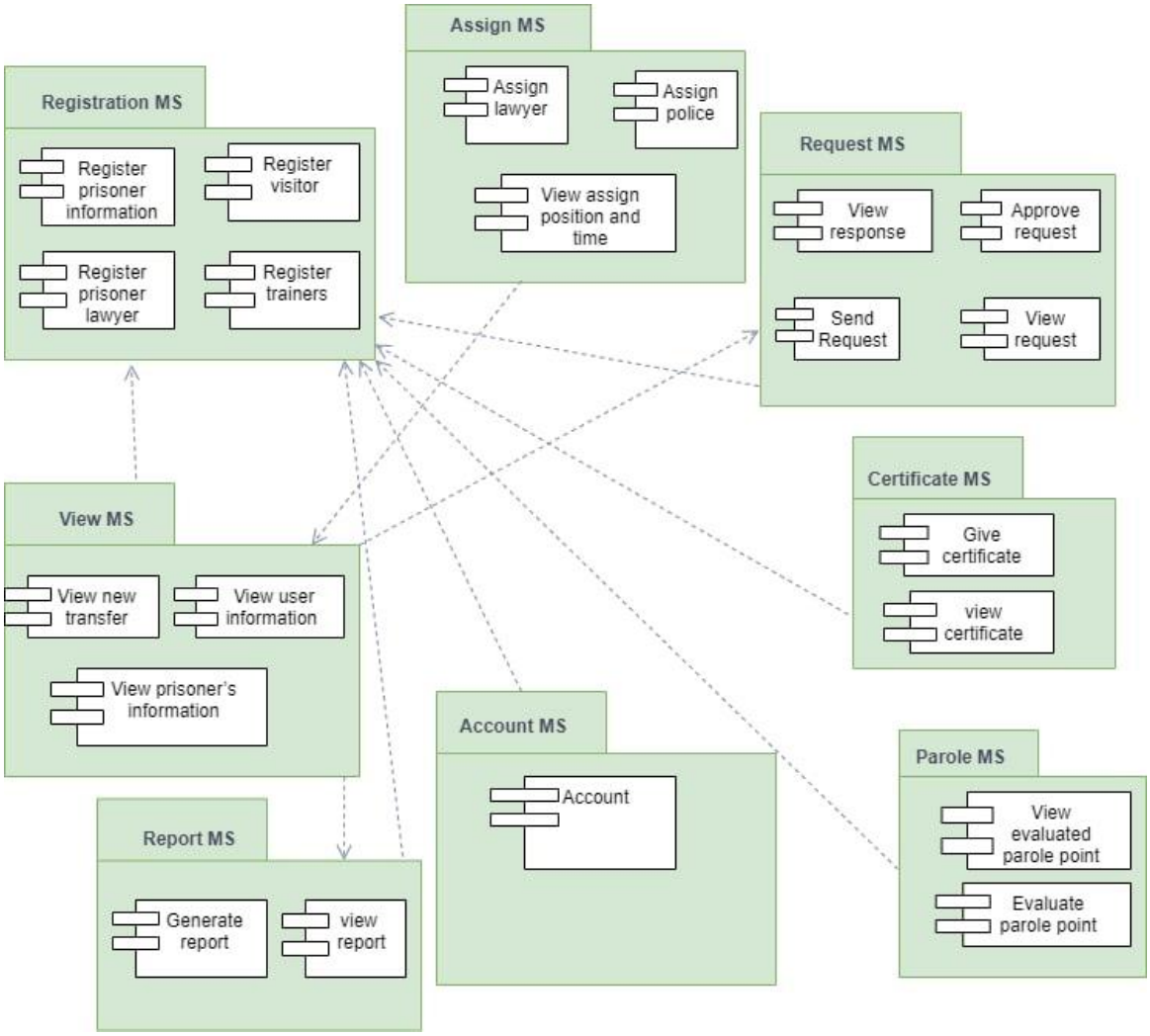


Figure 5.6 ***Package diagram***

5.4. Algorithm Design

Algorithm design is identified and incorporated into many solution theories of operation research, such as dynamic programming techniques for designing and implementation. This design part is important in the coding part of implementation. The purpose of using pseudo code is that it is easier for humans to understand than conventional programming language code and that it is a compact and environment independent description of the key principles of an algorithm. Some of the algorithms are listed below:

Algorithm for login

```
Begin
If user click login button
        The login form
        Fill form
        If(valid)
                Generate SQL select queries
                Connect to database
                Check query within database
                IF information in database
                        Display page
                Else
                        Display error message
        Else
                Display error message (Ask user to refill the form)
        End if
        End if
 End if
 End
```

Algorithm for prisoner registration

```
1  Begin
2  If register officer click register button
3         Display prisoner register form page
4         Then Register officer fill the form
5         If(valid)
6                 Send information into the database (DB)
7                 If information exist on DB
8                         Display the prisoner already registered message
9                 Else
10                        Prisoner information successfully registered
11         Else
12                Error message
13        End If
14        End If
15 End if
16 End
```

Activate Windows

5.5. User Interface Design

User interface design is the overall process of designing how a user will be able to interact with a system. Its goal is to make the user's interaction as simple and efficient as possible. The following interface design pictures describe the logical characteristics of some interfaces between the system and the users. So the sample interfaces of proposed system are shown as follows:

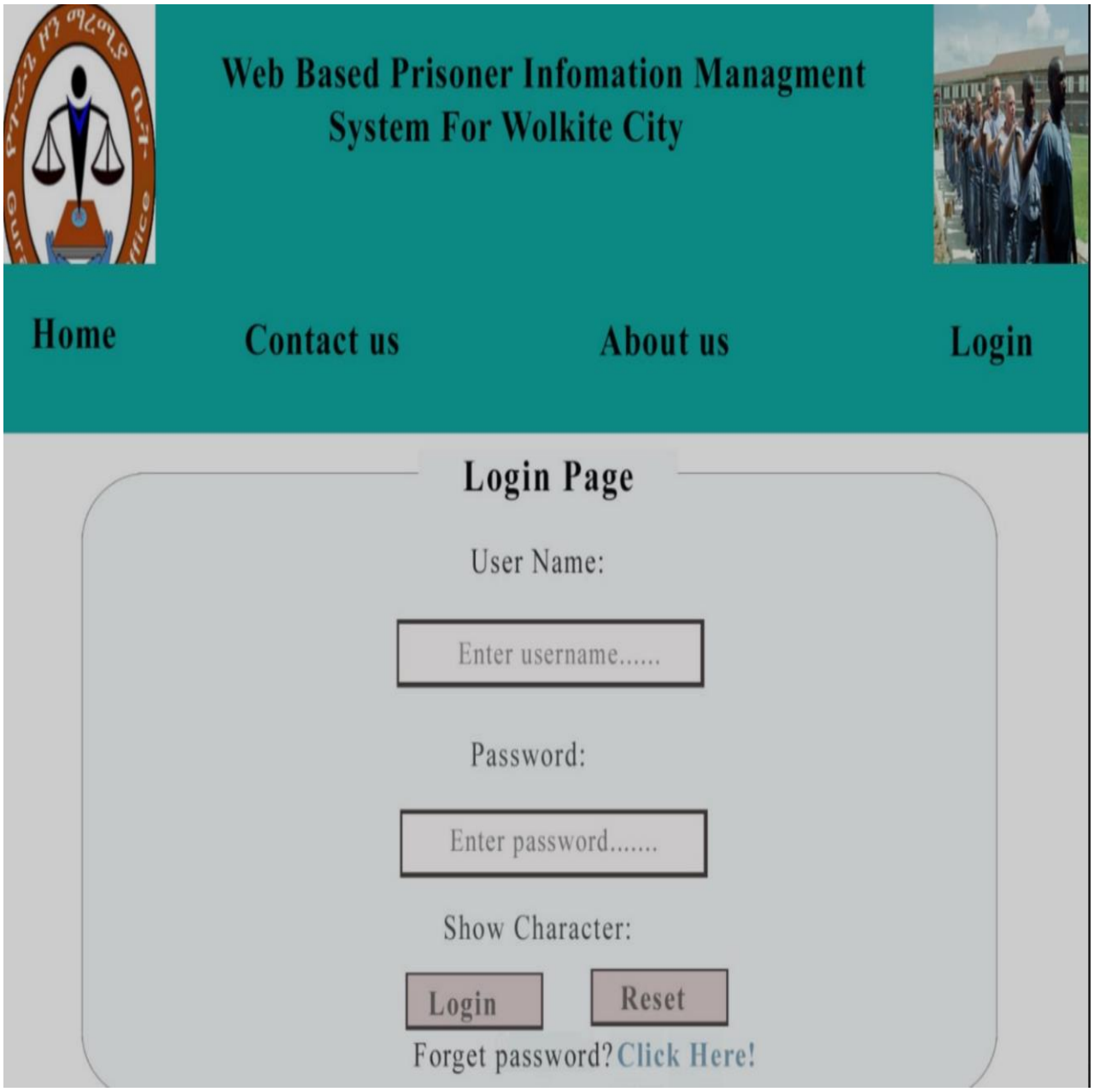


Figure 5. 1 Login user interface

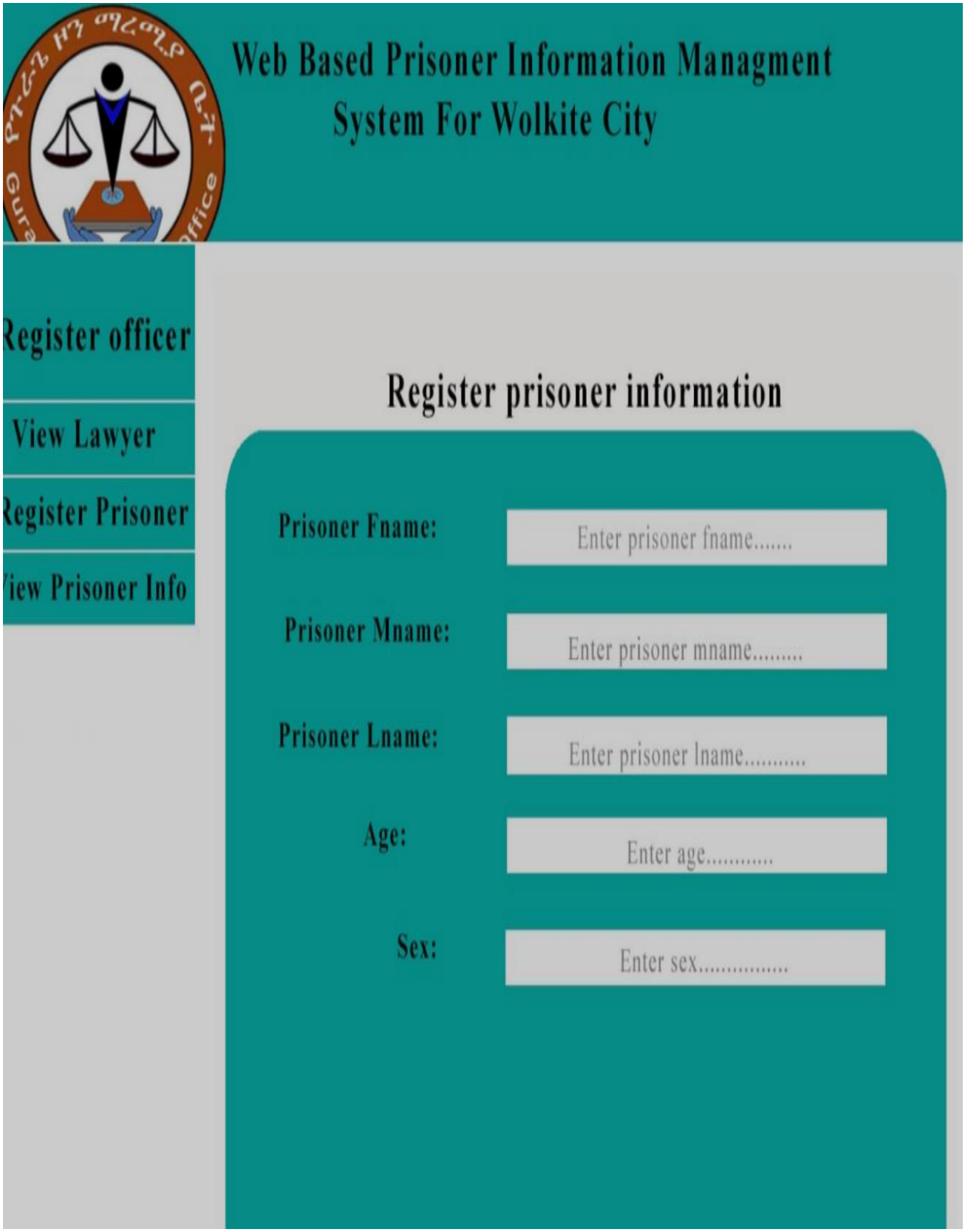


Figure 5. 2 Register prisoner user interface

CHAPTER SIX

6. IMPLEMENTATION AND TESTING

This chapter basically highlights the issues deal with the implementation phases. Implementation is the phase where objectives of physical operations of the system turned into reality i.e. real working model. In this phase the coding convention has made it possible as it's the real phase of objectivity to reality. Then the code is tested until most of the errors have been detected and corrected. The goal of implementation is to introduce our system for the users in real sense that how they use this new system which is developed for their intended objectives.

The implementation phase is the most crucial phase in which it transforms the design and analysis of the system into a tangible system by writing the code to be developed and make it operational and applicable by testing and debugging the functionalities that are done.

6.1 Implementation of the database

We have used MYSQL database because that can run any operating or browser application, installation is streamlined, security features are better, enhanced performance, it's important to maintain an environment that's standardized. MYSQL is easy fast and can used for any type of database weather it is relational or simple database, large or small database. So we use MYSQL server for implementation of our system or to store our data.And we have created a data base which is called pims and it have 18 different tables.



Figure 6. 1 Sample database implementation

6.2. Implementation of the Class Diagram

In this class diagram implementation we have implement the class diagram structure. That implements the view of an application, visualizing, describing and documenting different aspect of the system. And also implements the attribute and operation of the class.

```
CREATE TABLE `prisoner` (
  `pid` varchar(13) NOT NULL,
  `LID` varchar(34) NOT NULL,
  `firstname` varchar(20) DEFAULT NULL,
  `middlename` varchar(20) DEFAULT NULL,
  `lastname` varchar(20) DEFAULT NULL,
  `sex` varchar(8) DEFAULT NULL,
  `age` int(11) DEFAULT NULL,
  `nationalty` varchar(23) DEFAULT NULL,
  `region` varchar(11) DEFAULT NULL,
  `zone` varchar(16) NOT NULL,
  `worda` varchar(21) DEFAULT NULL,
```

```
  `kebele` varchar(21) DEFAULT NULL,
  `entrydate` date DEFAULT NULL,
  `exitydate` date DEFAULT NULL,
  `appdate` date NOT NULL,
  `app` varchar(43) NOT NULL,
  `crimetype` varchar(19) DEFAULT NULL,
  `courtname` varchar(18) DEFAULT NULL,
  `Photo` longtext NOT NULL,
  `status` varchar(54) NOT NULL,
  `appstatus` varchar(23) NOT NULL,
  `new` varchar(12) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6.3. Configuration of the application server

We have been used XAMPP as application server because it is a simple, lightweight Apache distribution that is extremely easy for us to create a local web server for testing and deployment purposes. Since the basic job of all web servers is to accept requests from clients (visitor of web browser) and then send the response to that request (the components of the page that a visitor wants to see). And a web server is an essential part of any website, since our developed system is web-based system, we had used Apache web Server which is open-source software.

- In the configuration process its installation is straightforward and simple then set upped and run the configuration to start, stop, and configure the services by opening the dashboard.
- It is integrated with most popular operating systems. We use XAMPP for windows.
- It is automatically installed on our computer as we install it.
- It is free web server and supports many operating systems.
- It is used to easily run and test websites and web applications locally.

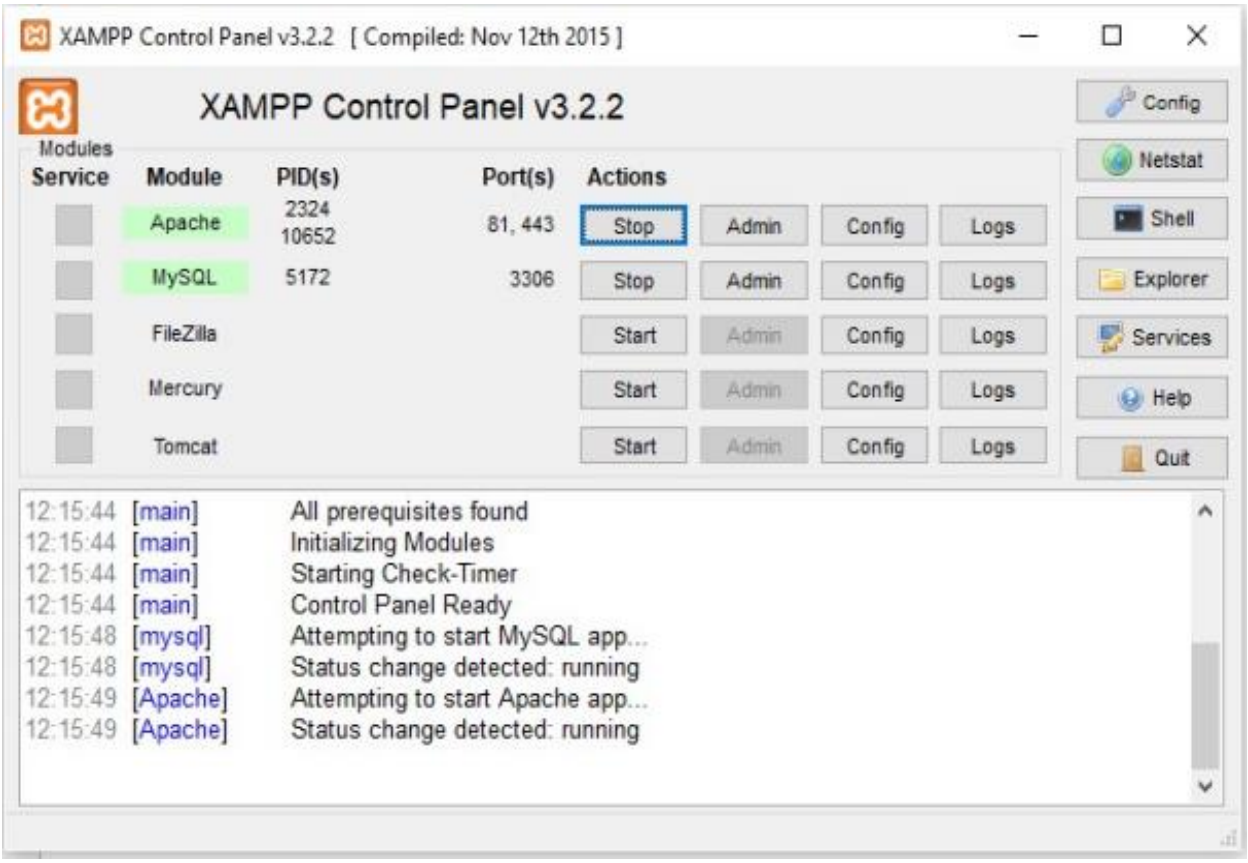


Figure 6. 2 XAMPP application server

6.4. Configuration of application security

Our system validates all the inputs by returning error message and suggesting to try again when invalid input occurred. We implement encryption for user password by MD5 when the system admin creates a user account or the user changes their password the system encrypt the password. The system has its security principles that control unauthorized authentication.

All input validations are implemented properly.

- Encryption and decryption algorithms are implemented properly for sensitive data.
- User's roles are handled properly for accessing the system.
- User's accounts are assigned with necessary access privileges.
- Functional requirements are described well and checked.
- Session and cookies are also implemented in this system properly

6.5. Implementation of user interface

The user interface of this system is implemented based on the design. It is clear and interactive; as much as possible complexity is handled in this user interface.

- It reduces memory loads of the users.
- It is consistent and interactive.
- It is easy and clear to use.

6.6. Testing

To know the fault of our developing system we use the testing method to solve the error in progressively. Testing is one of the developing stages of software development phase. This is done manually from system requirement phase system design phase and implementation phase. According to our system we make error checking for each phase. To solve error problem our group member uses different testing mechanism according to our team error checking method. These testing methods are unit testing, integration testing, system testing and acceptance testing. Testing evaluates a software product to ensure that it satisfies its planned purpose. A test that is modified to and consistent with development methodologies provides an observable and structured approach to verifying requirements and quantifiable performance.

6.6.1. Test Case

Test Case is a set of actions executed to verify a particular feature or functionality of your software applications.

Test Case 1: Check result on entering Valid User ID & Password

Test Case 2: Check result on entering Invalid User ID& Password

Test Case 3: Check response when a User ID is empty & Login Button is pressed, and many more. A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly.

Sample Test Case

Registration Form having following fields:-

- User Id
- User Name
- Password
- Role

Table 6. 1 Test case to create account

Test Case1: Register user account
Test case objective : To create user account

Test case description: Admin enters basic information of the user, then presses register button. Client program contacts with server, then server contacts with the database, and database checks for registration and sends message to the administrator.	
Requirements Verified: Yes	
Test Environment: Apache MySQL server must be in running state, Database Should contain appropriate table and link must be established between server and client program.	
Actions	Expected Results
The administrator should login and press the register menu.	Displays registration page
Administrator should enter user information	Displays success message of registration.
If some fields are not filled the system display message to fill the fields again.	

6.6.2. Unit Testing

Unit test is a way of testing each of the system functionality independently. The entire system will only be able to work well if the individual part are working well. Accordingly, the team has test each one of the major activities and the rest accompanying activities independently using different user input, different login mechanisms and any technique of fault finding so that an incorrect functioning of the activities was corrected at the right time. The main objective of unit testing to ensure that each individual part is working well and as it's supposed to work.

6.6.3. Integration Testing

As the name of the test indicates that the test performs to ensure the integration of the
User interface module that interact the user with the system and the database module that store the information. Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. Integration testing is conducted to evaluate the compliance of a system or component with specified functional requirements. It occurs after unit testing and before system testing.

6.6.4. System Testing

System testing is testing conducted on a complete integrated system to evaluate the system's compliance with its specified requirements. System testing takes, as its input, all of the integrated components that have passed integration testing. In this we tested all the functionalities in the System. All errors in the forms, functions, modules have been tested.

Finally, System testing ensures that the entire integrated software system meets the desired requirements. It tests a configuration to ensure known and predictable results. We apply System testing to insure the functional and non-functional requirements correctness, performance, accuracy, function validations, and security have been met correctly.

6.6.5. Acceptance Testing

Acceptance testing is a quality assurance (QA) process that determines to what degree an application meets end users' approval. Acceptance testing is a level of the software testing where a system is tested for acceptability. It is formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system. Similar to a unit test, an acceptance test generally has a binary result, pass or fail.

CHAPTER SEVEN

7. CONCLUSION AND RECOMMENDATION

7.1. Conclusion

Implementing the analyzed and designed web based prisoner information management system will be the best solution to the current manual based prisoner management system problem. The project is aimed at improving the management and service in prison of wolkite city. The new system might look and feel with greater comfort and met the desired objectivity. In design phase, we extended work analysis with more models. The class type architecture, class diagrams in analysis were extended in design to step closer to the implementation. We have applied the best efforts towards the achievement of our desired objectives of the proposed system. The system will play a crucial role for prison administrator by avoiding wastage of time. It will also be benefited to management stuffs by making their work effective to control and manage the recording system and also create job satisfaction to the stuff and reduces resource wastage by avoiding paper and other instruments cost.

Generally: This web based prison management system for Wolkite prison office performs or cover some limitation of manual based system like prisoner registration, certificate generation and other like searching prisoner data and updating data are previously performed manually.

7.2. Recommendation

According to scope of our project the team develops web based application. Because of the time constraint we may have limitation which should be consideration in, but in the feature the team believes that this system should be fully operationally by adding some functionality that are not included in the system. We also want to recommend this project can be expanded and more automated with additional functionalities by integrating with many new technologies .We recommend in future the following functionality:

- The system should have to support different languages
- The system should have to support consists of all employees of the prison not only about prisoner.

References

[1] "Java T point,," https://www.javatpoint.com, [Online]. Available: uml-use-case-diagram. [Accessed 4 2 2013].

[2] "w3shool,," [Online]. Available: https://www.w3schools.in/sdlc-tutorial/activity diagram. [Accessed 20 5 2014].

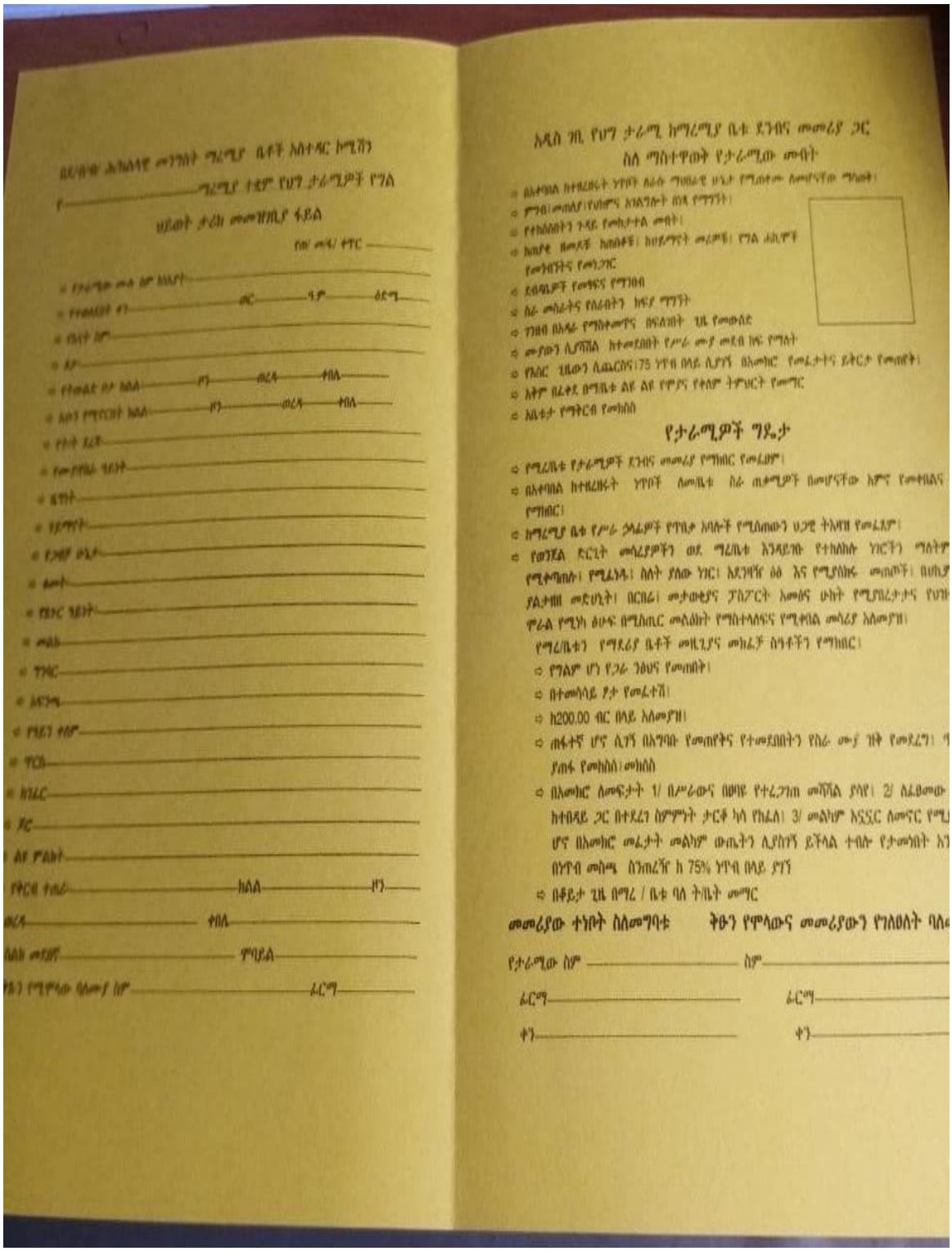
[3] "G. Notton," 2018. [Online]. Available: Persistence Model. [Accessed 2018].

[4] "Java T point,Online," [Online]. Available: https://www.javatpoint.com/uml-use-case-diagram.. [Accessed 4 2 2013].

[5] " I. Rafaqat,," 2016. [Online]. Available: UML diagram.

Appendix

Appendix A1:



አዲስ ገቢ የህግ ታራሚ ከማረሚያ ቤቱ ደንብና መመሪያ ጋር
ስለ ማስተዋወቅ የታራሚው ሙቡት

የታራሚዎች ግዴታ

መመሪያው ተነቦት ስለመግባቱ ቅፁን የሞላውና መመሪያውን የገለፀለት ባለ

Figure 5. 3 New prisoner registration form

Appendix A2:

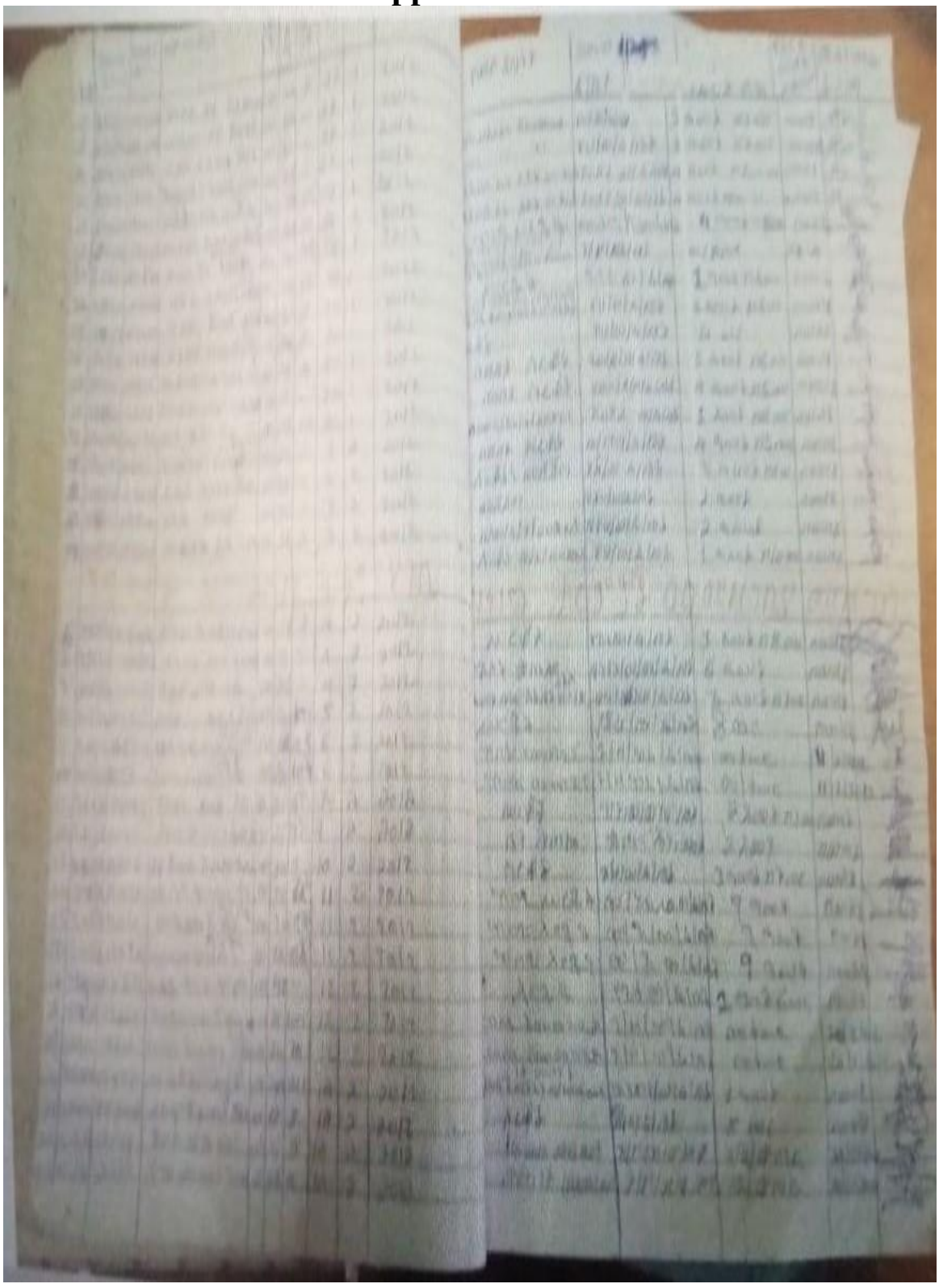


Figure 5. 4 Released prisoner registration