

WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF SOFTWARE ENGINEERING

PROJECT ON

WEB BASED LAND MANAGEMENT SYSTEM FOR GUBRE CITY

BY

No	Name	Id
1	Amanuel Abera	245/10
2	Desalegn Banbeta	263/10
3	Desalegn Dana	262/10

PROJECT ADVISOR: TAGELE BERIHUN(MSC.)

June 10, 2021

WOLKITE UNIVERSITY, WOLKITE, ETHIOPIA



WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF SOFTWARE ENGINEERING

PROJECT ON

WEB BASED LAND MANAGEMENT SYSTEM FOR GUBRE CITY

SUBMITTED TO DEPARTMENT OF SOFTWARE ENGINEERING IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHLER OF SCIENCE IN SOFTWARE ENGINEERING

BY

No	Name	Id
1	Amanuel Abera	245/10
2	Desalegn Banbeta	263/10
3	Desalegn Dana	262/10

PROJECT ADVISOR: TAGELE BERIHUN(MSC.)

June 10, 2021

WOLKITE UNIVERSITY, WOLKITE, ETHIOPIA

DECLARATION

This is to declare that this project work which is done under the supervision of Mr. Tagele Berihun and having the title Gubre land management system is the sole contribution of Amanuel Abera, Desalegn Banbeta, and Desalegn Dana. No part of the project work has been reproduced illegally (copy and paste) which can be considered Plagiarism. All referenced parts have been used to argue the idea and have been cited properly. We will be responsible and liable for any consequence if a violation of this declaration is proven.

Date: _____

Group Member:

Full name	signature
Amanuel Abera	_____
Desalegn Banbeta	_____
Desalegn dana	_____

APPROVAL FORM

This is to confirm that the project report entitled “Web Based Land Management System for Gubre City” submitted to Wolkite University, College of Computing and Informatics Department of Software Engineering by: Amanuel Abera, Desalegn Banbeta and Desalegn Dana is approved for submission.

Name of Advisor	Signature	Date
……	……	……..

Department Head Name	Signature	Date
…….....	……	…….

Examiner 1 Name	Signature	Date
……	……..	…….....

Examiner 2 Name	Signature	Date
……..	……	…….

Examiner 3 Name	Signature	Date
……..	……..…..	……

ACKNOWLEDGMENT

Firstly, we are greatly indebted to the following for their contribution to our project: The Almighty God, who gave us the insight and perseverance to accomplish this project. Secondly, we wish to express our sincere gratitude to the Faculty of Computing, for providing us an opportunity to perform our project work on “Land Management System for Gubre City “and also Our Advisor Instructor Mr. Tagele Berihun who is sounding advice helped us guide this project in the right direction, in which he enforces us to have a massive knowledge about the project to easily familiarize with the concept of project development. Finally, we would like to thank the Gubre land management officer and the department manager for their partial willingness to interview, patience in answering our numerous questions, giving existing activity materials, and reading materials that help us to precede our project.

TABLE OF CONTENT

DECLARATION i
APPROVAL FORM ii
ACKNOWLEDGMENT iii
LIST OF FIGURES vii
LIST OF TABLE viii
LIST OF ABBREVIATIONS ix
ABSTRACT x
CHAPTER ONE 1
1. INTRODUCTION 1
1.1. Background of the study and organization 2
1.2. Statement of the Problem 2
1.3. The objective of the Project 4
1.3.1. General Objective 4
1.3.2. Specific Objective 4
1.4. Feasibility Analysis 4
1.4.1. Operational Feasibility 4
1.4.2. Technical Feasibility 5
1.4.3. Economic Feasibility 5
1.5. Scope and Limitation of the Project 6
1.5.1. Scope of project 6
1.5.2. Limitation of the Project 6
1.6. Significance of the Project 6
1.7. The beneficiary of the project 7
1.8. Methodology of the Project 7
1.8.1. Data Collection Tools/Techniques 8
1.8.2. System Analysis and Design 8
1.8.3. System Development Model 9
1.8.4. System Testing Model 9
1.8.5. Development Tools and Technologies 10
1.9. Document Organization 12
CHAPTER TWO 14
2. DESCRIPTION OF THE EXISTING SYSTEM 14

2.1. Introduction of the existing system 14
2.2. Users of Existing System 14
2.3. Major Functions of the Existing System 15
2.4. Drawbacks of the Existing System 16
2.5. Business Rules of the Existing System 16
CHAPTER THREE 18
3. PROPOSED SYSTEM 18
3.1. Introduction 18
3.2. Functional requirements 18
3.3. Non-functional requirements 19
3.3.1. User interface 19
3.3.2. Security 19
3.3.3. Performance 20
3.3.4. Fault tolerance 20
3.3.5. Accessibility 20
3.3.6. Reliability 20
3.3.7. Reusability 20
3.3.8. Portability 20
3.3.9. Scalability 21
3.3.10. Integrity 21
3.3.11. Resource issues 21
3.3.12. Documentation 21
CHAPTER FOUR 22
4. SYSTEM ANALYSIS 22
4.1. System Model 22
4.1.1. Use Case Model 22
4.2. Object Model 31
4.2.1. Class Diagram 32
4.3. Dynamic Model 35
4.3.1. Sequence Diagram 35
4.3.2. Activity Diagram 37
4.3.3. State chart diagram 40
CHAPTER FIVE 42

5. SYSTEM DESIGN 42
5.1. Design Goals 42
5.1.1. User interface 42
5.1.2. Security 42
5.1.3. Performance 42
5.1.4. Fault tolerance 42
5.1.5. Accessibility 43
5.2. Proposed System Architecture 43
5.2.1. Subsystem Decomposition and Description 44
5.2.2. Hardware/Software Mapping 47
5.2.3. Detailed Class Diagram 47
5.2.4. Persistent Data Management 49
5.2.5. Access Control and Security 50
5.3. Packages 52
5.4. Algorithm Design 52
5.5. User Interface Design 55
REFERENCES 58

LIST OF FIGURES

Figure4. 1 Use case diagram 24
Figure4. 2 class diagram 32
Figure4.3 sequence diagram for creating an account 36
Figure4.4 sequence diagram for login 36
Figure4.5 Activity diagram for creating an account 37
Figure4.6 Activity diagram login system 38
Figure4.7 Activity diagram for register Land 39
Figure4.8 state chart diagram for login 40
Figure4.9 state chart diagram for register land 41
Figure5.1 System architecture 44
Figure5.2 Subsystem Decomposition and Description 46
Figure5.3 Hardware/Software Mapping 47
Figure5.4 Detailed Class Diagram 48
Figure5.5 Persistent Data Management 50
Figure5.6 Packages 52
Figure5.7 Login interface 55
Figure5.8 Create account 56
Figure5.9 Land owner registration form 57

LIST OF TABLE

Table1. 1 Front end technology 10
Table1. 2 Backend Technology 11
Table1. 3 Software Tool 11
Table1. 4 Deployment Environment 12
Table4. 1 Use case description for Login 24
Table4. 2 Description for create account 25
Table4. 3 Description for change password 26
Table4. 4 Description for giving certificate 27
Table4. 5 Description for registered land 28
Table4. 6 Description for view land information 28
Table4. 7 Description for user registration 29
Table4. 8 Description for maintaining land transfer 30
Table4.9 Data Dictionary for the landowner 32
Table4.10 Data Dictionary for Employee 33
Table4.11 Data Dictionary for users 33
Table4.12 Data Dictionary for Role 33
Table4.13 Data Dictionary for land 34
Table4.14 Data Dictionary for land boundary 34
Table4.15 Data Dictionary for certeficate 34
Table4.16 Data Dictionary for permission 35
Table5.1 Access control and table 50

LIST OF ABBREVIATIONS

No	Abbreviations	Descriptions
1	GLMS	Gubre land management system
2	GPS	Global positioning system
3	GUI	Graphical user interface
4	HTML	Hypertext markup language
5	ID	Identification
6	LA	Land administration
7	LAN	Local area network
8	LMS	Land management system
9	LO	Land owner
10	LRO	Land registeral officer
11	LS	Land survayor
12	MYSQL	Michael wideniu's daughter structured query language
13	OOP	Object oriented programing
14	OOSAD	Object-oriented system analysis design
15	PHP	Hypertext preprocessor
16	SA	System administrator
17	SDLC	Software development life cycle
18	SIMS	Students information management system
19	UC	Use case
20	UML	Unified modeling language
21	XAMPP	Cross-platform approach Maria DB PHP and Perl

ABSTRACT

The land management system is concerned with the social, legal, economic, and technical framework within which land managers and administrators must operate. Gubre city's land management system uses a manual system to manage lands. This leads to losses of data, data redundancy, needs more manpower and time, conflict among a society, unauthorized man access to user's data. Due to this reason, we will develop a web-based land management system (LMS) that facilitates various activities taking place at the Gubre City land management office. This project is a web-based land Management system designed for Gubre city land Management. To develop a proposed system, we use object-oriented system analysis and design methodology and it has seven main actors involving in the new system. These are Landowner, Land registrar office, Surveyor, System administrator, Land administration office, Building permit officer, and Mayor. Each Actor has a login page and can access the system by filling in a Username and password. But actors have the privilege to access the system determine by admin Therefore the proposed system is secured, more user-friendly, and minimizes a problem of the existing system. In designing, we use PHP-which has been engaged as a development language, and xammp as a backend database with CSS implemented for the interface. Our project includes land Registration, free land Allocation, to register the stakeholder (landowner) It is necessary to register land and property, provide ownership certificate, inform private and public right regarding land, provide amendment information, provide transfer of land in form or legacy, Generate and view the report, give privilege for creating an account, delete and update.

CHAPTER ONE

1. INTRODUCTION

Land is a scarce resource in Ethiopia. Because Land is an essential natural resource, both for the survival and prosperity of humanity, and for the maintenance of all terrestrial ecosystems. It is a key natural resource that requires proper consideration in rural and urban development. In Ethiopia, the land has been used as a policy instrument to attract domestic and foreign direct investment. To incentivize investment, land in urban, rural, and semi-urban areas has been transferred to private and public companies. Precisely, Ethiopia's urban land lease policy, according to the 2011 urban land lease proclamation, is formulated to attract both domestic and foreign direct investment. Investment attraction is chiefly to stimulate economic growth and reduce poverty. Gubre in the region of Southern Nations, Nationalities, and People's Region is a city in Ethiopia - some 86 mi or (139 km) South-West of Addis Ababa, the country's capital. The city is developing at a higher rate. And the use of land by society is also increased even if the land is scarce. The ever-changing global factors like sustainable development, environmental sustainability, globalization, rapid urbanization, economic reform, and technological improvements necessitate the adjustment of various strategies and models of land administration. The land management system is the process of use and development of the land resource. To enhance the practice of good governance in society successfully achieving good land administration is the prior one. However, there are many problems in emerging cities related to good governance in urban land administration that hinder sustainable urban development. Urban land management is a sustainable solution to those problems which requires immediate decisions to be made before the problems are beyond the control of the local government Gubre city has a manual-based land management system with many problems such as Data storage and retrieval problem, the investment allocation is inefficient and time-consuming, let delivery of owner's land registration, Information storage risk, and the like. Due to the above reasons, we are motivated to seek a solution to address the problems raised. This project is a web Based land management system for Gubre city. It reduces the problem in the existing system Since the existing system is manual (fully paperwork). So, it needs more manpower and consumes time to act within a prescribed time. But, the proposed system is more user-friendly and the users interact with it simply every time in everywhere

with a LAN and it is more secured. Because we try to authenticate the legal user by user name and password.

1.1. Background of the study and organization

The land is a key natural resource that requires proper consideration in rural and urban development. The Issue of land everywhere in Gubre is economically, politically, and socially the most sensitive and significant aspect of human beings

Currently, Gubre city land management offices process data manually, this manual system process data using papers and set files on the shelf or any other catalog. The completed project solves the problems that had faced on Gubre land management offices. The land management office performs different activities. such as land registration, give land to an investor for building industry, and give lands to an individual and group of people for normal residence and business activities.in addition to these activities store data in a manual manner. The land management office transfer lands based on the status of land use for:

- Business: -There are different types of business activities implementing on lands. Such as the building of a shop, the building of a hotel.
- Industry:-Is an institution that constructs in individual or organizational lands for a specific purpose
- Residents: -The uses of land for building homes to live
- Mixed use:-It implies land user for residence and an industry Much of the problems of the manual system avoids through developing a good system and data management practices.

1.2. Statement of the Problem

Because the existing system of the land administration is fully manual (data storage and search, file about the owner, and document identification method), it leads to losing very sensitive data. Another problem of an existing system is a security issue where anybody who enters the record office sees the owners' profile and can update it as he/she wants. Due to an unorganized database about the lands the town losses much money and time every year. This led to time delay to assign the land to customers. On the other hand, there is the case where more than two

individuals own one plot that raises a conflict between communities. Generally, the problem of the services delivery at Gubre town administration office put as follows.

- Performance Related problem
  - ✓ Poor performance in terms of time.
  - ✓ The work procedure is prone to error.
- Information related problem
  - ✓ Mishandling of information occurs.
  - ✓ Data is difficult to correct and maintain.
  - ✓ Data is stored redundantly in multiple files.
  - ✓ Data is not easily accessible.
  - ✓ Data is not well organized.
  - ✓ Data is not secure from damage.
- Efficiency related problem
  - ✓ Since the work is performed manually the efficiency of the working system is poor.
  - ✓ As the amount of the customer increase, the waiting time of the customer will be increase.
  - ✓ As the amount of the customer increase, the resources are wasted.
  - ✓ Since the data is stored redundantly, the information generated also is redundant.
- Control related problem
  - ✓ Problem encounters due to the exhaustion of the staff's maters the organization.
  - ✓ Security is another big issue concerning the manual-based system.
- Economic related problem
  - ✓ The office needs many stores to place the files.
  - ✓ Manual handling of data is expansive.
  - ✓ Cost in terms of time is high.
- Service-related problem
  - ✓ Customer doesn't get better service and they don't satisfy with service.

1.3. The objective of the Project

1.3.1. General Objective

The general objective of the project is to design and develop a web-based land management system for Gubre city.

1.3.2. Specific Objective

- Exploring and studying the existing Gubre land management manual-based system.
- Implementing user information management component.
- Implementing a Land registration component to register every land of the city and owners.
- Implementing component for registering land cadastral.
- Developing and maintaining information of each land and landowner lease rights.
- Designing and developing the database to store face templates, and recorded land.
- Developing components for generating and printing different certificates like land certificates and land ownership certificates.
- Developing land transfer component to the individual based on the rule and specification of the land rule and regulation on the ground.
- Developing a report generator component that generates reports for the recorded lands.

1.4. Feasibility Analysis

1.4.1. Operational Feasibility

The system to be developed is going to provide accurate, active, and secure service that fits with the existing system in terms of operation. The system is operationally feasible as it very easy for the end-users to operate it. It only needs basic information about the Windows platform. The proposed system offers effective controls to protect the data and information accuracy and security of the organization against any faulty.

1.4.2. Technical Feasibility

The system is going to be developed by following the Object-Oriented System Development technique. And the team can develop this system without any difficulty since the team has studied the required methodologies and tools. So the system will be technically feasible. The project members also know PHP, JavaScript, CSS, and MYSQL databases. Besides, the group members have enough capability to develop the project. So the system is technically feasible.

1.4.3. Economic Feasibility

The proposed system provides a choice of services with an efficient and reliable automated system because officers manage accurate landowner information in a minimum amount of time. Also proposed system reduces workload officers, such as preparation of land owner's and non-land owner's reports, modifying land owner certificate annually in a little amount of time. The landowner can view posted notes and get legalized ownership certificates anywhere at any time using the network in the minimum amount of time. Our proposed system compared to the existing, new system does not waste cost in terms of time. So, the proposed system is economically feasible**.** The cost of the new system is less as compared to the current paper-based manual system in which it requires a small number of employees and it reduces costs for buying of paper. As part of this, the costs and benefits associated with the proposed system are compared and the project is economically feasible only if tangible or intangible benefits outweigh costs.

Tangible benefits: Includes **e**rror reduction, cost reduction, increased service

- Reduce the number of employers and reduce the wastage of time.
- Reduces unnecessary wastage of materials like paper, pen, and human power.
- Reduced time consumption.
- Reduce economic consumption.

Intangible benefits**:** those benefits that are difficult or impossible to quantify or cannot easily be measured by money. The following are intangible benefits: improved employer's morale, motivate staff, **l**ittle job burden to employers of Gubre Land management office, better service to Gubre Land Management office, better decision making, speed of activities and services and workflow is efficient.

1.5. Scope and Limitation of the Project

Scope and limitation important to define specific boundaries of your project in terms of what the project does and what the project doesn't.

1.5.1. Scope of project

This project will develop a web-based land management system for gubre.

The land management system will perform the following activities: -

- Design a system that applies to Gubre City land management office.
- The system is web-based and mainly used for Land registration.
- Registering the basic information of Lands by registering Land Boundary Registration.
- Transferring land to the individual based on the rule and specifications of the land rule and regulations on the ground.
- Maintaining Personal (Land Owner) Information.
- Preparing a report for the recorded land information.
- Generating different certificates like land certificates and land ownership certificates.

1.5.2. Limitation of the Project

In this section, you need to define specific boundaries of your project in terms of what the project doesn't.

- The system uses the only English language. Hence it does not support the local language.
- The system does not include GPS

1.6. Significance of the Project

The societal and technological importance of our project and who benefit from the system. discuss in this part-

This project aims to develop a web-based system for the Gubre City Land Management office. Land is a scarce resource that should manage carefully and gives a crucial role play in the economy. So, efficient and Careful managing is required to make the best of it. Since the

existing filing system is manual (data storage and search, file about the owner, and document identification method) it leads to losing very sensitive data.

- The system not support license for the investor .it is simply give information whether or not the land is vacant
- Emphasizes automating databases that store and retrieve documents about the owner of the land.
- Helps Gubre City land management to avoid delay in service delivery, enhancement activities.
- Ensures Security issues by developing a password-protected database.
- The Staff workers to registering a land within LAN
- The Staff workers easily show vacant lands with a minimum amount of time.
- It reduces paper consumption cost, reduces human labor, and the office more manageable.

The system provides and stores correct, accurate information, got rid of time-consuming and errors, and registers of land or land holder's related activities. It provides an easy-to-use and timely delivery of information.

1.7. The beneficiary of the project

Different bodies are benefited from this system. The main beneficiaries of this system include: The Gubre Land Management Office: - To Gubre land management system to minimizing the workload of the employees and reduces wastage of time, increase to the work efficiency of the office and all the customer record information's are kept properly in the well-organized database.

The Gubre Landowner: - It reduces the time, cost, and energy of landholders by providing full access in a short time.

1.8. Methodology of the Project

To accomplish this project on time and within the cost, we follow different procedures which are described below.

1.8.1. Data Collection Tools/Techniques

Data collection methodology is the way of gathering relevant data/information to study the problems of the current system.

Interview: -We use this technique to interview employees and managers for an overall virtual impact of a system environment. This takes into the study details relating to land or landholders and the land municipality office, number of staff, critical and important information about the overall view of the current system such kind of information experienced, and type of services the office provides. For that purpose, interviews were employed to get relevant information for conducting this project from the principal persons of Land management.

Document analysis: - to conduct this project document analysis is essential to analyze different documents that are found in the office which contains information related to the office, problems found in the manual system, the services that the office provides to the users, and the number of employees.

Observation: -witness the actual events which happen here in Gubre city. In this method, we observe all the activities that are performed in Gubre Land Administration. It helps us to get real information and how the organization performs its function. So, this helps to strengthen the data gathered through interviews and document analysis.

1.8.2. System Analysis and Design

Our project includes object-oriented system analysis and design (OOSAD).

Object-Oriented Analysis (OOA)

During this phase, the team uses to model the function of the system (use case modeling), find and identify the business objects, organize the objects and identify the relationship between them and finally model the behavior of the objects in detail.

Object-Oriented Design (OOD)

During this phase, our team uses Enterprise Architecture software to refine the use case model and for designing the sequence, class diagram, statechart diagram, and activity diagrams and to model object interactions and behavior that support the use case scenario. This system analysis and design have many benefits:

- Increase Extensibility.
- Increase Reusability.
- Improve Quality.
- Minimize maintenance costs and burden.

1.8.3. System Development Model

Specify how the activities of the development process are organized in the total system development effort. We use iterative system development methodology because of its extensibility which means through the process of developing the system error is occur we back to the previous phase and correct the problem.

We use the Iterative model for system development. The reasons that we are selecting the Iterative SDLC Model are: -

- The ability of the system to allow changes
- The ability of the system to support component addition
- Easily incarnate or backward to any phase
- Results are obtained early and periodically.
- Less costly to change the scope/requirements.

1.8.4. System Testing Model

Unit Testing: -We test each unit or component of the system to validate if each unit performs what it is expected to do.

Integration testing: -We test individual software modules are combined and function as a group.

We conduct integration testing to evaluate the compliance of a system or component with specified functional requirements.

System testing: -We conduct system testing to validate the complete and fully integrated web application or no malfunctioned portion on the system.

1.8.5. Development Tools and Technologies

In this section and subsections, we mention programming languages and their editor, database technologies, documentation tools, unified modeling language (UML) design tools, hardware tools for deployment purposes.

Software: - are application programs that help us to

- Design interface
- Create database
- Write and run our code efficiently and effectively

1.8.5.1. Frontend Technologies

The user interface is developing using HTML, JavaScript, and CSS**.** Since it is easily designing the front end and connected to the database easily.

Table1. 1 Front end technology

Technologies	**Description**
HTML 5	All browsers support HTML HTML is free Most development tools support HTML Include new attribute, elements, and abilities Client-side coding
CSS	Browser compatible Save time Easy to maintain
JavaScript	Interoperability Client-side scripting Speed Simplicity

1.8.5.2. Backend Technologies

MYSQL software of the database system and PHP language is used in developing and managing the back end of the system

Table1. 2 Backend Technology

Technologies	**Description**
PHP 7.1	Cross-platform, speed, stable Server-side scripting
MySQL	Is the most secure and reliable Database server
Apache server	Is a fast, reliable, and secure web server

1.8.5.3. Documentation and Modeling Tools

Table1. 3 Software Tool

Software	**purpose**
Visual Studio Code	For editing PHP, CSS, and HTML codes
Enterprise architecture	To design and construct a software system For modeling business processes For modeling industry-based domains
Microsoft word 2016	To edit the documentation of the project
Snipping tool	For sniping (cutting or shape) image

1.8.5.4. Deployment Environment

We will deploy our system on localhost for training purposes, and we illustrated the hardware tools that we will use for deploying our system for training purposes.

Table1. 4 Deployment Environment

Hardware Type	Specification	Description
Computer Brand	Dell, Toshiba, and HP	We will deploy our system, for training purposes, on the computer brand on our hands.
Processor	Core(i3) or above with 2.5 GHz or above	The greater the speed of the processor, the faster the execution speed of our system.
RAM	4.00 GB or above	The larger the RAM, the faster the execution speed of our system.
System Type	64-bit Operating system	We will use a 64-bit processor to deploy our system. Why a 64-bit processor? It is capable of storing more computational values, including memory addresses, which means it can access over four billion times the physical memory of a 32-bit processor. It has a higher performance than a 32-bit processor.

1.9. Document Organization

Chapter Two: - This chapter is mainly concerned with studying the existing system by elaborating its existing system, business rules, main functions, forms and other documents, and bottlenecks.

Chapter Three: - This chapter deals with the proposed system by deeply explaining its functional and non-functional requirements.

Chapter Four: - This chapter deals with the system model, object model, and dynamic model of the proposed system.

Chapter Five: - This chapter deals with the design goals of the proposed system, the proposed system's architecture (it includes subsystem decomposition, hardware, and software mapping), algorithm design, and user interface design of the proposed system.

CHAPTER TWO

2. DESCRIPTION OF THE EXISTING SYSTEM

2.1. Introduction of the existing system

Gubre land management starts in 1999 E.C. Various tasks are performed by this Gubre land management. Land Management, whether formal or informal, comprises an extensive range of systems and processes to manage. The processes of land Management include the transfer of rights in land from one party to another through sale, lease, gift, and inheritance, the regulating of land and property development; the use and conservation of the land; and the resolving of conflicts concerning the ownership and the use of land. The organization uses a land registration book manuscript to register the holder's land, update when the holder the land in the form of bequeathing to his/ her children, donation or transfer the holder's land in the form of rent to other land users. The land tenure manuscript has many pages and large in height, width so that if the land Manager wants the profile of the specific landholders it is too difficult and time-consuming and also if they want to do any work for one's landholder such as update, search, delete the landholder profile when death, or other specific work. For conducting this project we have identified limitations or factors from the existing systems are the unavailabilities of file and misplacement(regarding places) creates difficulty in searching, loss of landholding files during manual file transferring in a different location, If the holding book is lost from the people update and retrieve are very difficult and complex in manual system and lack of file security, immediate information storage, data integrity that leads to information redundancy, unavailability of accurate and up-to-date information that results from the wrong decision in the management It requires more labor to manage the data contained in every area of the process, Physical damage of file due to dust, moisture, and other cause, low accuracy, large storage space and time consumption for searching and retrieving landholders record manually. Software modernization problem is transmitted from one sector to another by flash.

2.2. Users of Existing System

The users of the existing system are the landowner, Land registrar office, Surveyor, Land administration office, Building permit officer. All of them are users in Gubre land management.

- The main role of the landowner:
  - ✓ Apply for lease right
  - ✓ View report
  - ✓ View land information
- The main role of the Land registrar office:
  - ✓ The land registrar's office can manage personal information.
  - ✓ The land registrar's office can register land information.
  - ✓ The land registrar's office can view the report.
- The main role of Surveyor
  - ✓ Surveyors can view information.
  - ✓ Surveyors can register land boundaries.
- The main role of the Land administration office
  - ✓ The land administration office can prepare a landowner certificate.
  - ✓ The land administration office can maintain land transfer.
  - ✓ The land administration office can view the report
- The main role of the Building permit officer
  - ✓ The building permit officers can approve the building permit.
- The main role of the mayor
  - ✓ approve the land ownership certificate.

2.3. Major Functions of the Existing System

These are major functionalities of Existing Land Management.

- Solves land conflict
- Differentiate between those who paid tax and those who did not.
- To register the stakeholder (landowner).
- Provide ownership certificate
- Change the old possession to lease system for cities that register as the city administration
- Registration land.
- Makes a name transfer.
- Generate and view the report.

- It transmits in the form of a lease.
- Provide amendment information.
- Provide transfer of land in form or legacy.
- Assigning free lands to Applicant.

2.4. Drawbacks of the Existing System

Since the existing system of Gubre city management faces many drawbacks. The project team analyses the existing system drawbacks as: -

- Lack of information exchange.
- The system has problems related to security.
- It takes a long time to search for data.
- Lack of efficiency, effectiveness,
- lack of organization of data.
- Lack of accurate data.
- Since it operable manually it uses a large amount of space/staff to store data
- Lack of portability's

2.5. Business Rules of the Existing System

In every organization or institution, there are rules and policy, which used to govern all activities. Gubre land management also has its own business rule then the business rules of the existing Gubre land management is identified below:

- BR1: Transfer Land
  - ✓ The owner should have to be a legal landowner
  - ✓ The customer has to pay tax within the specified time and amount.
  - ✓ The reason why to transfer land should have to be clearly stated.
  - ✓ The land to be transferred should have to be free from Third-party allegation.
  - ✓ The land to be transfer owned should be free from bank and insurance issues
- BR2: Land owner
  - ✓ the owner should be at least age of 18
  - ✓ The owner should have an ID and a Valid license
  - ✓ an owner should be residences of the city administration

- BR3: Land registration
  - ✓ Identify lands on the kebele of the city administration.
  - ✓ Register land to land bank or land information system

CHAPTER THREE

3. PROPOSED SYSTEM

3.1. Introduction

The proposed system is designed to eliminate all the drawbacks of the existing land management system. The system to be developed automating distributed admin, staff workers, the landowner in Gubre city. The proposed system will have a small number of computer intelligent employees who can perform any web-based related works. The system is a user-friendly, easy, fast, and secured database to store all information of every landholder/owner. The proposed system can perform every activity using web-based to provide fast, efficient, and effective services to the users and it saves the cost, time, and human labor of the office and the users. It can search, update, delete and land holder's information using a unique identifier and it can register new information that contains land and holder's details. The proposed system has the following advantages including ensuring data accuracy, security of data, system admin controls the entire system, greater efficiency, user friendly and interactive, and minimum time required.

3.2. Functional requirements

- User Management:
  - ✓ Create account.
  - ✓ Update user profile.
  - ✓ Assign access privilege for users (grant access privilege).
  - ✓ Create user groups (Roles).
  - ✓ Control User account (Edit, Active/Inactive)
- The system allows viewing the reports about the land records in different categories as described below.
  - ✓ Free land report.
  - ✓ Reserved land report.
  - ✓ Total land report.
- Maintain land transfer: the system can able to transfer land to the individual based on the rule and specifications of the land rule and regulations on the ground.

- This system aimed to reduce the time of the finding of information from the set of huge data.
- Maintain Personal (Land Owner) Information: The landowner who wants to be registered as a member of the annuity can be registered into the system.
- Land registration: The system registers every land of the city and owners. To know the available land, the users of the system can search out from land entities.
- Land Boundary Registration: the system should register land cadastral.
- Approve Building Permit: the system should maintain detailed information about the building permits like commercial, residence, industrial or other.
- Apply Land Lease Rights. The System should apply information on each land and landowner lease rights.
- Maintain certificate: The system should generate and print different certificates like land certificates and land ownership certificates.
- Maintain administration and backup: - the system can allow backup of the data if something may crash.
- View land information: - the system that allows us to view the land-related information.
- Approve land owner certificate: - the system can allow approving the landowner certificate.

3.3. Non-functional requirements

In this section, we have discussed the quality attributes in which our system should have to perform the functional requirements.

3.3.1. User interface

The system intends to provide a user-friendly and clean GUI using a consistent theme throughout all to avoid user confusion, with intuitively understandable buttons, tooltips, and wizards. Moreover, the GUI will consider different browsers and monitor the resolutions of the user workstation.

3.3.2. Security

The system should be secured, i.e., not allow unauthorized users to access the database system, and also the system uses a hash function algorithm. In our system, no one can access the system

without getting a privilege by the system. The system will not allow unauthorized users to add or delete packages in the system.

3.3.3. Performance

The system will have good performance as much as possible this will be attained via easily loadable interface components and optimal algorithms and database performance tuning, which make searching, updating, deleting, inserting, and saving easy and fast.

3.3.4. Fault tolerance

Whenever the user commits errors intentionally or unintentionally the system should continue performing its task without interruption by displaying the user-friendly error message.

3.3.5. Accessibility

our project is web-based and accessed through web browsers using computers and mobile phones all land management system users can access the system very easily.Accessibility is the ability of the system to be accessible to users. since our project is web-based and accessed through web browsers using computers and mobile phones all land management system users can access the system very easily.

3.3.6. Reliability

The software should be a system users can trust or rely on due to the accuracy of the information or data it provides. When end users want to perform their task the system should provide the appropriate data without frustrating users with ambiguous or unreliable data.

3.3.7. Reusability

The system, which is divided into subsystems, partitions, layers, and modules, will attain massive reusability. The modules will be repeatedly used in the different subsystems. And the subsystems and layers are arranged to be used by other systems outside CMIS. This makes software development to be smooth, speedy, and modifiable.

3.3.8. Portability

The system development uses PHP and PHP is programming once, run anywhere; platform independence. Hence the system is readily installable and operable on any kind of software

like operating systems, browsers, and hardware.

3.3.9. Scalability

Since the system is modular it is assumed that the system will be scalable for futureuse.

3.3.10. Integrity

Data integrity ensured that data is of high quality, correct, consistent, and accessible, important to follow rules governing data integrity. Hence the developed system will be tested for data and database integrity. The system should easily integrate with any applications.

3.3.11. Resource issues

To use the system users should have to install any browsers on their computer.

3.3.12. Documentation

The system document is well defined to use and maintain easily if the system is failing. The system will have contained short and precise file like readme file that will guide(help) the system user how will use the system easily without spending their time on training and in addition, the system document will have included the same video about the system to indicate how should it do and the video support the system users by showing the process of how they will have to use the system.

CHAPTER FOUR

4. SYSTEM ANALYSIS

4.1. System Model

To produce a model of the system which is correct, complete, and consistent we need to construct the analysis model which focuses on structuring and formalizing the requirements of our system. The analysis model contains three models: functional model this model deals with the functionality of the system, what is expected from the system, the services that are given by the system. The functional model is generally described by the use case diagram. Object model this model describes the structure of the system. The object model includes attribute, object, and association. Dynamic models deal with the internal behavior of the system and also how one-use case executes, series of activities that one-use case executes as well as the state. The dynamic model is described by the sequence, statechart, and activity diagrams

4.1.1. Use Case Model

The Use Case Model describes the functionality of the land management system. It represents a discrete unit of the interaction of a user (actors of the system) with the land management system. Use Case has a description that describes the functionality that will be built in the proposed system. Use Cases are typically related to 'actors', that is a human or machine entity that interacts with the system to perform meaningful work.

4.1.1.1. Use case Diagram

To create relevant use cases for the system, we used the following actors for the system that have been identified with description. A use case defines a goal-oriented set of interactions between external users and the system under consideration or development. To create relevant use cases for the system, we used the following actors for the system that have been identified with description.

- Landowner
  - ✓ Landowner Can log in to the system
  - ✓ Landowner Can apply for lease right
  - ✓ Landowner Can view land info

- Land registrar office
  - ✓ Land registrar office can log in to the system
  - ✓ The land registrar's office can manage personal information.
  - ✓ The land registrar's office can register land information.
  - ✓ The land registrar's office can view the report.
- Surveyor
  - ✓ Surveyor can log in to the system
  - ✓ Surveyors can view information.
  - ✓ Surveyors can register land boundaries.
- System administrator.
  - ✓ The system administrator can log in to the system
  - ✓ The system administrator can manage the account.
  - ✓ The system administrator can maintain administration and backup data
- Land administration office
  - ✓ The land administration office can log in to the system.
  - ✓ The land administration office can prepare a landowner certificate.
  - ✓ The land administration office can maintain land transfer.
  - ✓ The land administration office can view the report.
- Building permit officer
  - ✓ The building permit officer can log in to the system.
  - ✓ The building permit officers can approve the building permit.
  - ✓ The building permit officer can view the report.
- Mayor
  - ✓ Mayor can log in to the system.
  - ✓ Mayor can approve the land ownership certificate.
  - ✓ Mayor can view the report

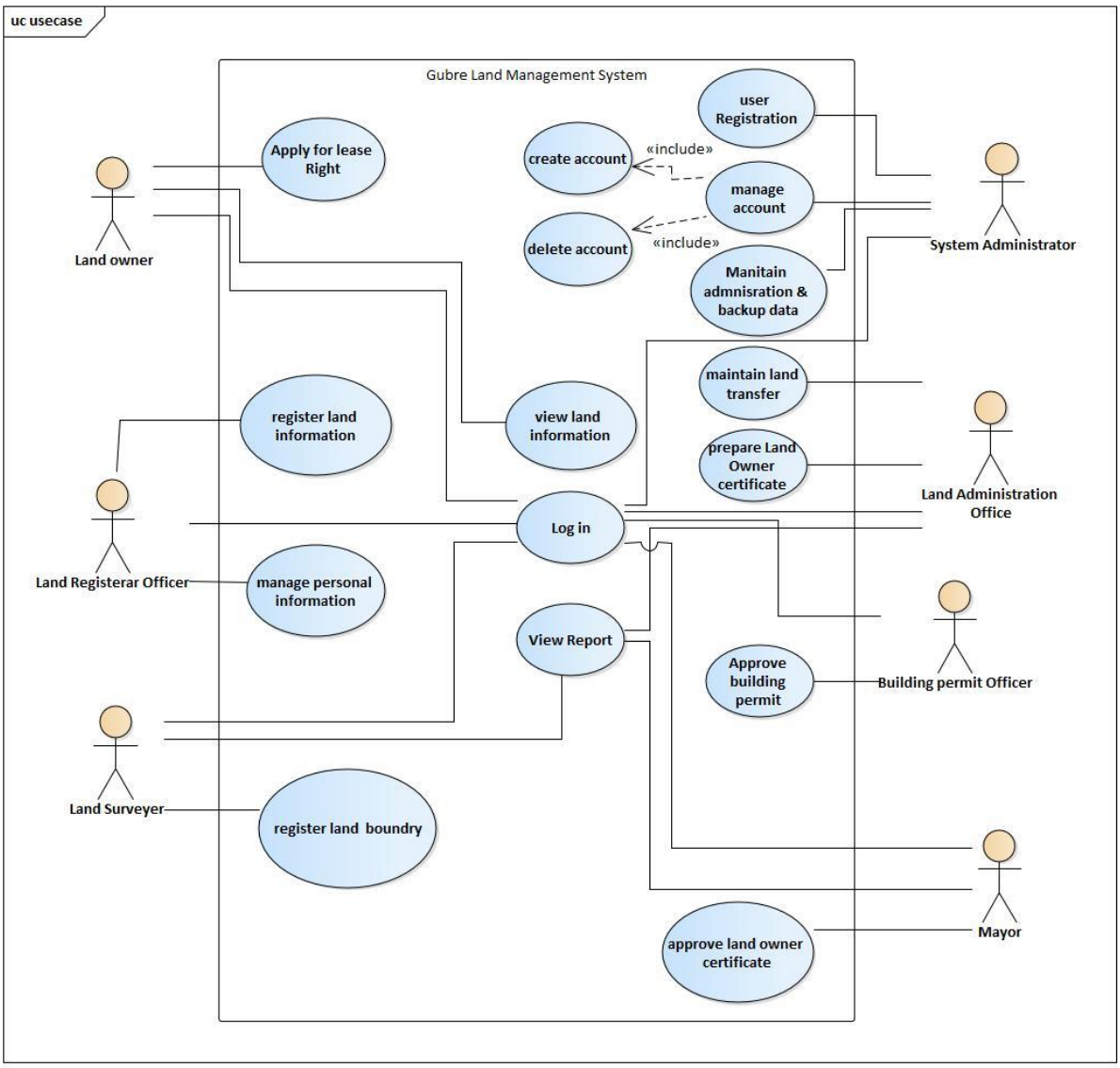


Figure4. 1 Use case diagram

4.1.1.2. Use Case Description

Table4. 1 Use case description for Login

Use case name	Login

Use case id	Uc-1	
Primary actor	All users	
Description	The user who wants to access the system can be login to the system	
Pre-condition	The user must have an account	
Basic course of action	Actor action	System response
	1.The user opens the home page 3. The user fill in username and Password 4 The user clicks on the login button 7.Use case end	2.The system displays the login page 5. the system cheek username and password 6.The appropriate page will be display
Alternative course of action	3.1. The system generating message incorrect username or password Go back to action no 3 3.2. Use case end	
Postcondition	Users access the system	

Table4. 2 Description for create account

Use case Name	Create account
Use case id	Uc-2
Actor	System Administrator
Description	The admin create an account for the user by giving the privilege to access the system

Pre-condition	The user must be registered within the database	
Basic course of action	Actor action	System response
	1. System Administrator must log in. 2. System Administrator click on create account link 4. The system admin fill the form and click on create button 6. Use case end	3.The system display create account form 5 the system display a success message
Alternative course of action	4.1.If the admin fill incorrect data in the form, the system displays an incorrect message and Go back to step 4 4.2.Use case end	
Postcondition	The account is created successfully	

Table4. 3 Description for change password

Use case name	Change password	
Use case id	Uc-3	
Primary actor	All users he/she has the privilege to log in	
Description	Change password	
Pre-condition	They must be login	
Basic course of action	Actor action	System response
	1.The user enter username and password	3.The system checks whether the entered username and

	2. The user clicks on the login button 4.Click on the change password link 5. Enter the old password, new password and confirm the new password 6. Save the change. 8.Logout when they finish the actual work 9.Use case end	password is correctly fill 7.The system displays the password is changed successfully
Alternative course of action	5.1. The system generating message incorrect username or password". Go back to action no 5. 5.2. Use case end	
Post-condition	Users updated password if he/she is correct	

Table4. 4 Description for giving certificate

Use case Name	Give certificate
Use case id	Uc-4
Actor	Land Administrator office
Description	Provide owner ship certification
Pre-condition	The user should be the legal landowner
Basic course of action	1.The officer must be login to the system 2.search owner who has to be

	certified 3.View information about the user while searching 4.Give a certificate for selected owners
Alternative course of action	Back to the basic course of action 2
Post-condition	Legal landowner get the certificate

Table4. 5 Description for registered land

Use case Name	Register land	
Use case id	Uc-5	
Actor	Land registrar officer	
Description	Register land information in the system	
Basic course of action	Actor Action	System response
	1.The officer must be login to the system 2.click the register land link 4.the officer fill the land information and click the registration button	3.system display the form 5. the system displays a success message 6. use case end
Alternative course of action	4.1. If the officer fill in the wrong information to the system display an error message and back to the system basic course of action 3	
Post-condition	The land is registered in the database	

Table4. 6 Description for view land information

Use case Name	View land information	
Use case id	Uc-6	
Actor	Land owner	
Pre-condition	The land information is registered in a database	
Description	View information about land	
Basic course of action	Actor Action	System response
	1.The system admin must be login to the system 2.click the user register link 4.the officer fill the land information and click the registration button	3.system display the form 5. the system displays a success message 6. use case end
The alternative course of action		
Post-condition	The land information display as the owner wants	

Table4. 7 Description for user registration

Use case Name	User Registration	
Use case id	Uc-7	
Actor	System Administrator	
Description	Register users	
Basic course of action	Actor Action	System response
	1.The system admin must be login to the system	3.system display the form

	2.click the user registration link 4. the admin fill in the user information and click the registration button	5. the system displays a success message 6. use case end
Alternative course of action	4.1. If the admin fill in wrong information to the system displays an error message and back to the system basic course of action step 3	
Post-condition	The user is registered in the database	

Table4. 8 Description for maintaining land transfer

Use case Name	Maintain land transfer	
Use case id	Uc-8	
Actor	Land administration officer	
Description	To transfer the land	
Pre-condition	The owner should have to be a legal landowner	
Basic course of action	Actor Action	System response
	1.The Land administration officer must be login to the system 2.click maintain land transfer link 4. the Land admin fill in the transfer information and click the button 6. Land admin check the legality of the landowner and click the button.	3.system display transfer form 5. the system display checking legality form 7. the system displays a success message 8. use case end
Alternative course of action	4.1. If the Land admin fill in wrong information to the system display error message and back to the	

	system basic course of action step 3 6.1. If the landowner can have an illegal issue (like-bank and insurance issue) the system can display you should avoid those illegal issue message and back to the system basic course of action step 3
Post condition	Maintain the land transfer successfully

4.2. Object Model

An object model is a description of an object-oriented architecture, including the details of the object structure, interfaces between objects, and other object-oriented features and functions.. In this section, we discuss the object model class diagram and data dictionary. In the class diagram, we have described diagrammatically the conceptual relationship among objects and class of the proposed system (land management system). Metadata that describes the database of land management systemt hat includes data ownership, data relationships to other objects, and other data; thus, metadata are discussed under data dictionary.

4.2.1. Class Diagram

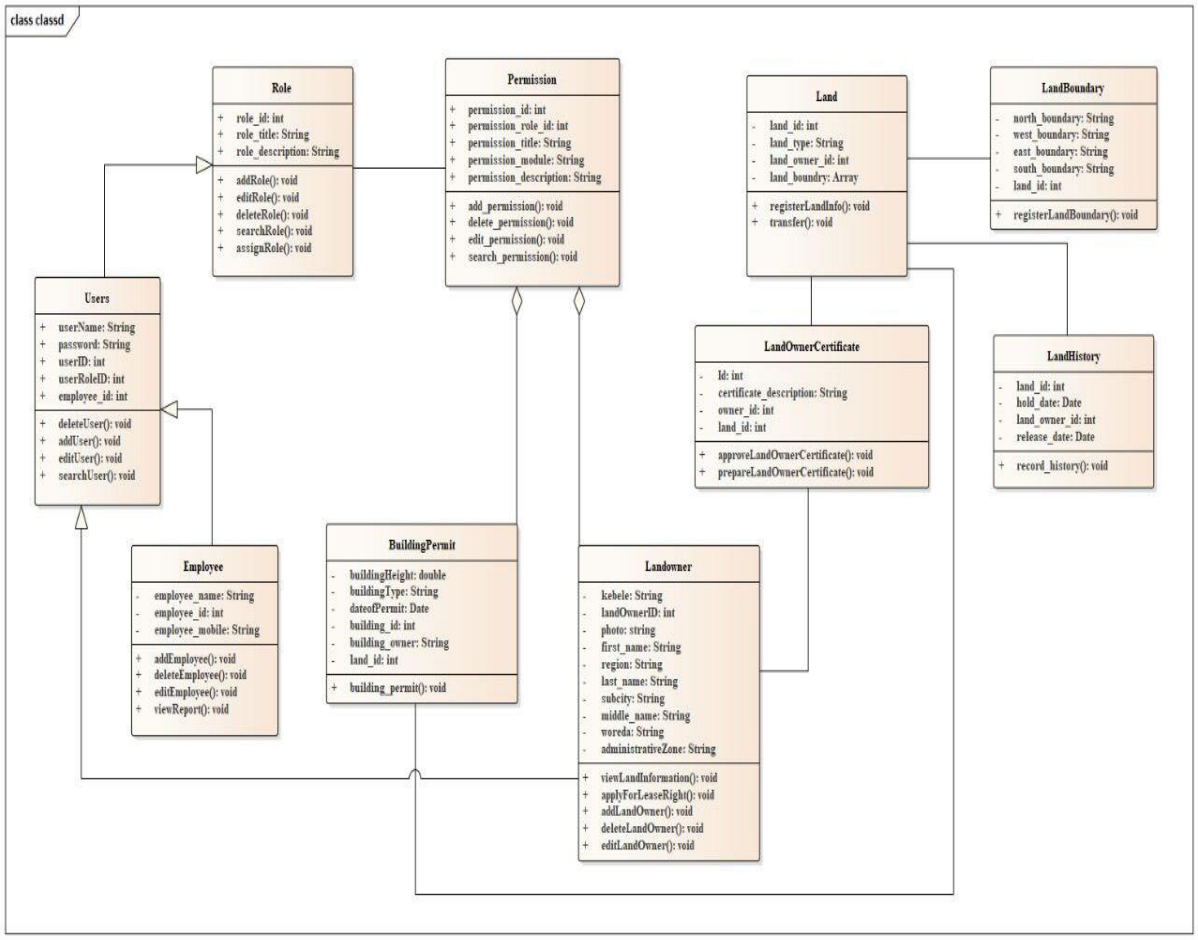


Figure4. 2 class diagram

Table4.9 Data Dictionary for the landowner

Attribute	Data type	Size	Format	Constraint
Land owner ID	int	50	1234	primary key
First name	varchar	60	Abc	not null
Middle name	varchar	60	Belay	not null
Last name	varchar	60	Alemu	not null

Region	varchar	100	SNNP	not null
Kebele	varchar	80	01	not null
woreda	varchar	60	cheha	not null
Administrative zone	varchar	60	Guragie	not null
Subcity	varchar	70	Gubriye	not null
Photo	varchar	200	--	not null

Table4.10 Data Dictionary for Employee

Attribute	Data type	Size	Format	Constraint
Employee ID	int	50	1234	primary key
Employee name	varchar	60	Abc	not null
Employee mobile	varchar	13	+251 912324422	not null

Table4.11 Data Dictionary for users

Attribute	Data type	Size	Format	Constraint
User ID	int	50	1234	primary key
User name	varchar	60	Abc	not null
password	varchar	60	*********	not null
Employee ID	int	60	1234	Foreign key
User role ID	int	60	1234	Foreign key

Table4.12 Data Dictionary for Role

Attribute	Data type	Size	Format	Constraint

Role ID	int	50	1234	primary key
Role title	varchar	60	Abc	not null
Role description	varchar	200	Role is …	not null

Table4.13 Data Dictionary for land

Attribute	Data type	Size	Format	Constraint
Land tipe	varchar	60	Abc	not null
Land owner ID	Int	50	1234	Foreign key
Land boundery	float	50	{20,20,…}	Nott null

Table4.14 Data Dictionary for land boundary

Attribute	Data type	Size	Format	Constraint
North boundary	varchar	50	13’n	Not nul
East boundary	varchar	50	12’e	not null
West boundary	varchar	50	12’w	not null
South boundary	varchar	50	34’s	Not null
Land id	Int	50	1234	Foreign key

Table4.15 Data Dictionary for certeficate

Attribute	Data type	Size	Format	Constraint
Id	int	50	1234	Primary key
Certificate description	varchar	250	12’e	not null
Owner id	int	50	1234	Foraign key

Land id	Int	50	1345	Foreign key

Table4.16 Data Dictionary for permission

Attribute	Data type	Size	Format	Constraint
Permission ID	int	50	1234	primary key
Permission type	varchar	60	Abc	not null
Permission description	varchar	200	Role is …	not null
Permission role id	Int	50	1234	Foreign key
Permission module	varchar	50	Module...	Nott null

4.3. Dynamic Model

Under this model we have discussed three of these dynamic model diagrams, thus are sequence diagram,activity diagram, and state chart. The dynamic model shows the behavior and collaboration between objects found in the land management system.Under this model we have discussed two of these dynamic model diagrams, thus are sequence diagram and activity diagram and state chart diagram.

4.3.1. Sequence Diagram

A sequence diagram is emphasizing the time ordering of messages. We have tried to show the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario in the land management system.

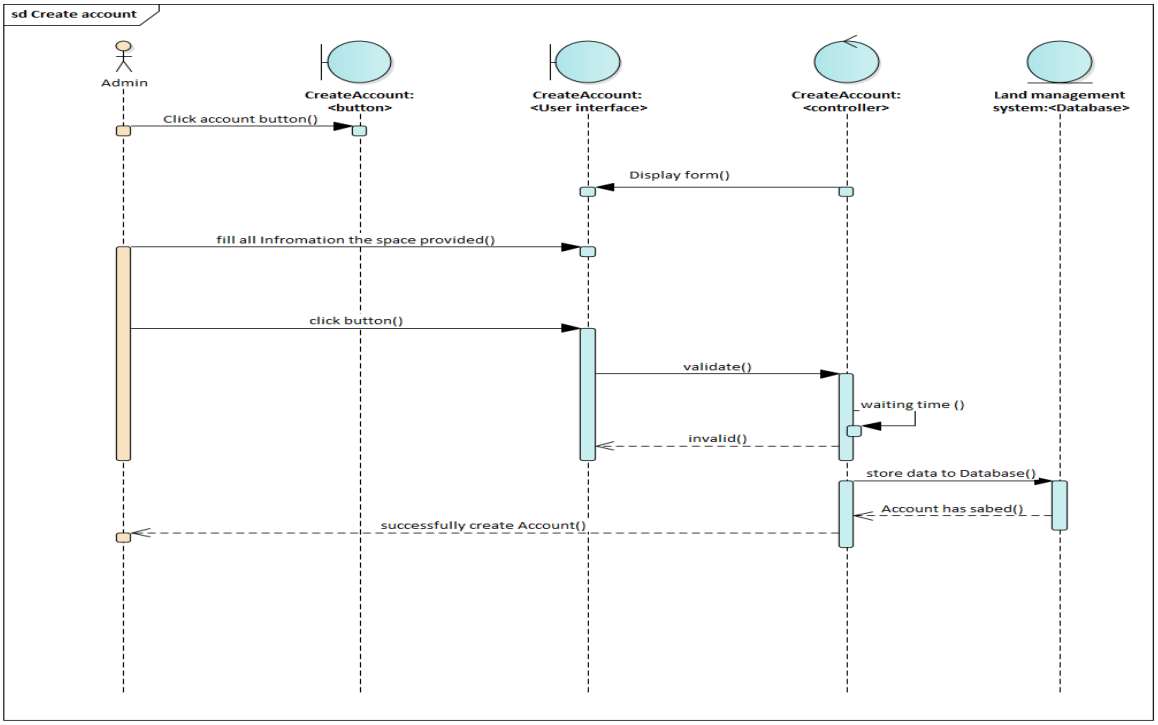


Figure4.3 sequence diagram for creating an account

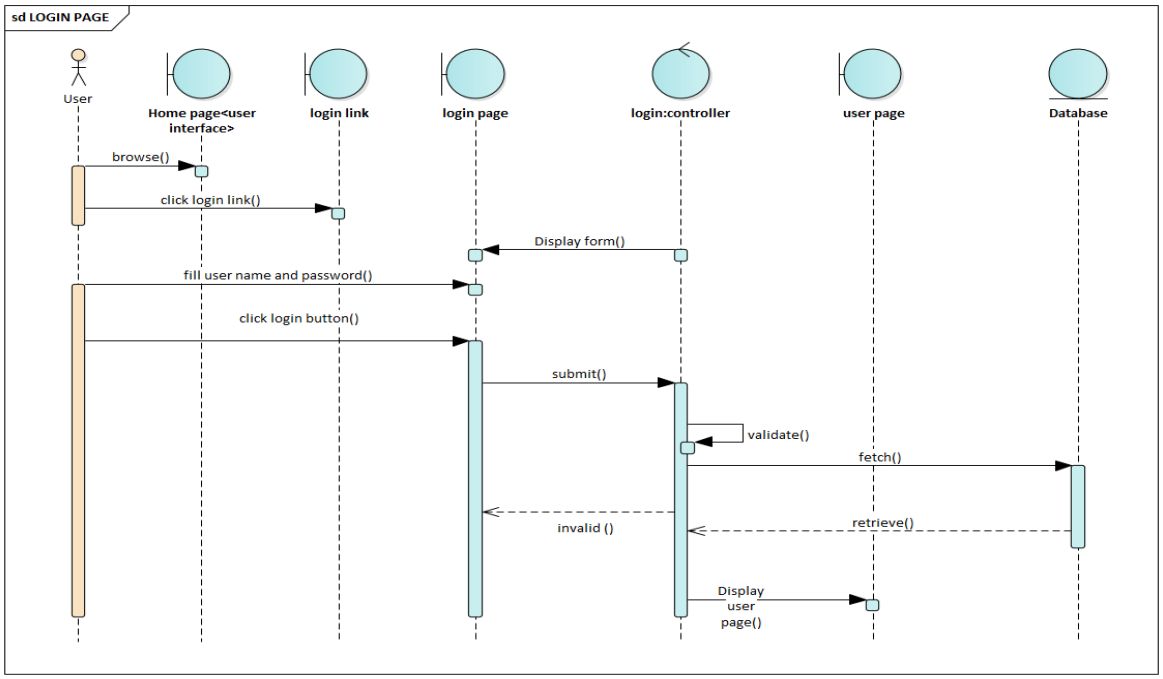


Figure4.4 sequence diagram for login

4.3.2. Activity Diagram

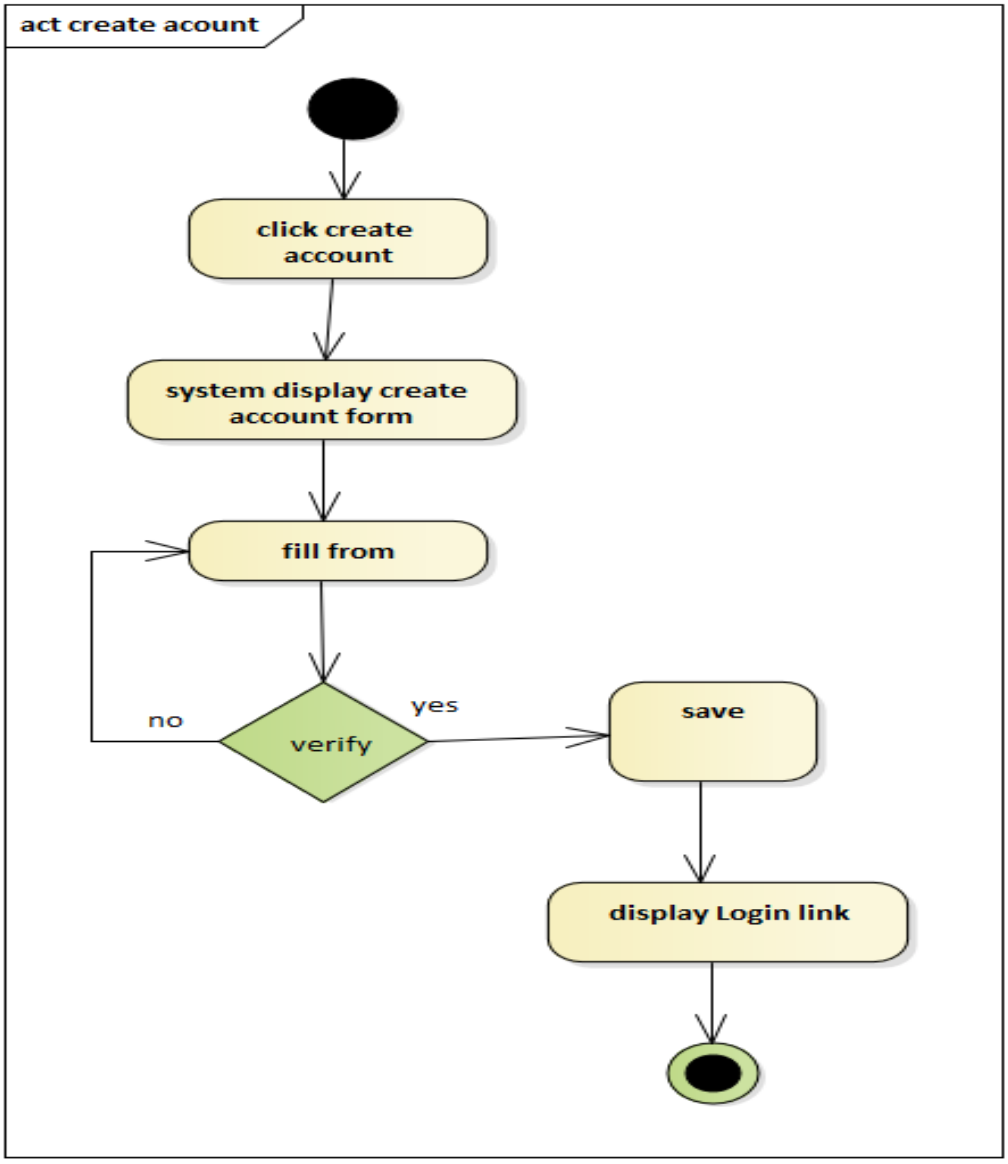


Figure4.5 Activity diagram for creating an account

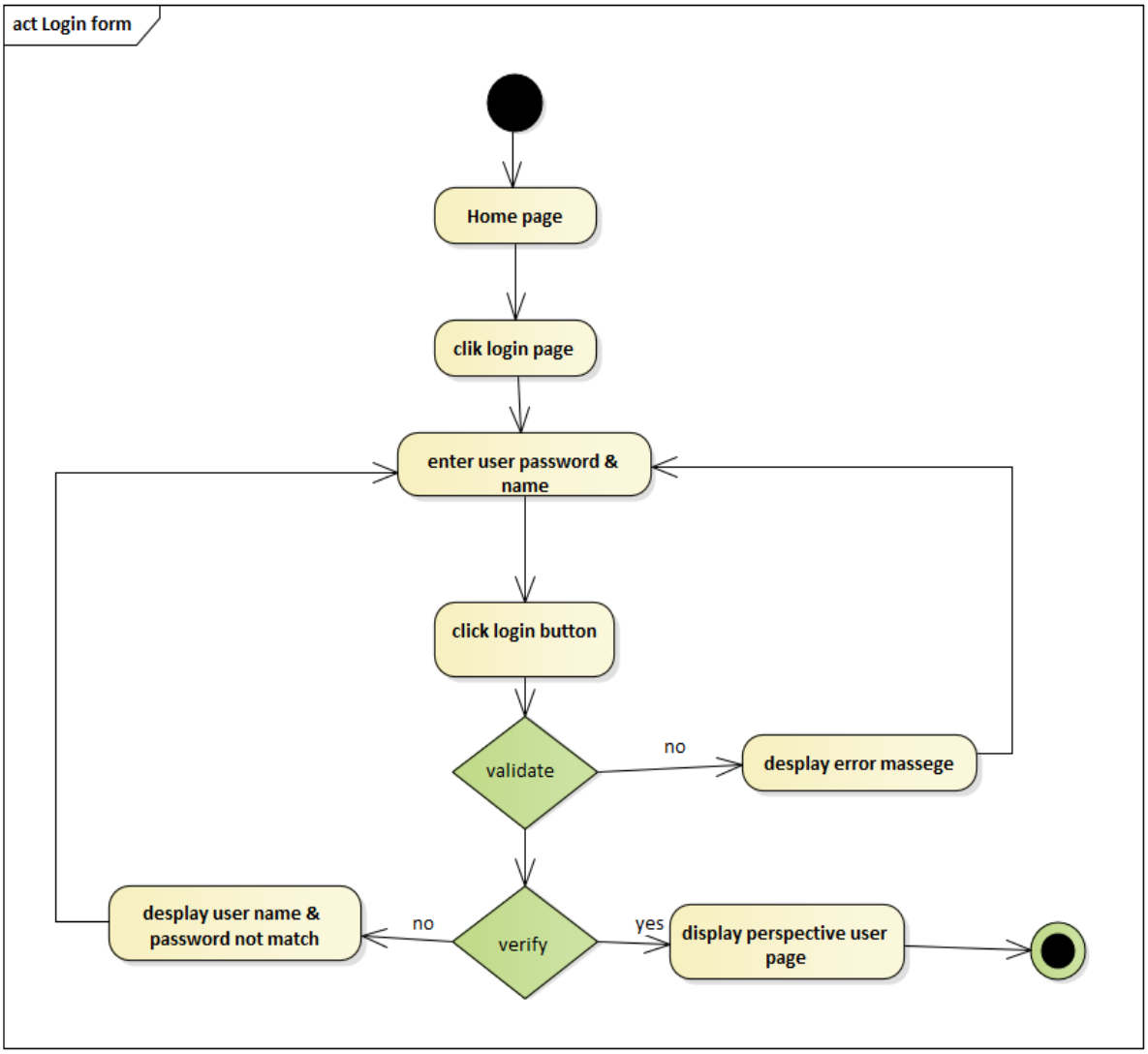


Figure4.6 Activity diagram login system

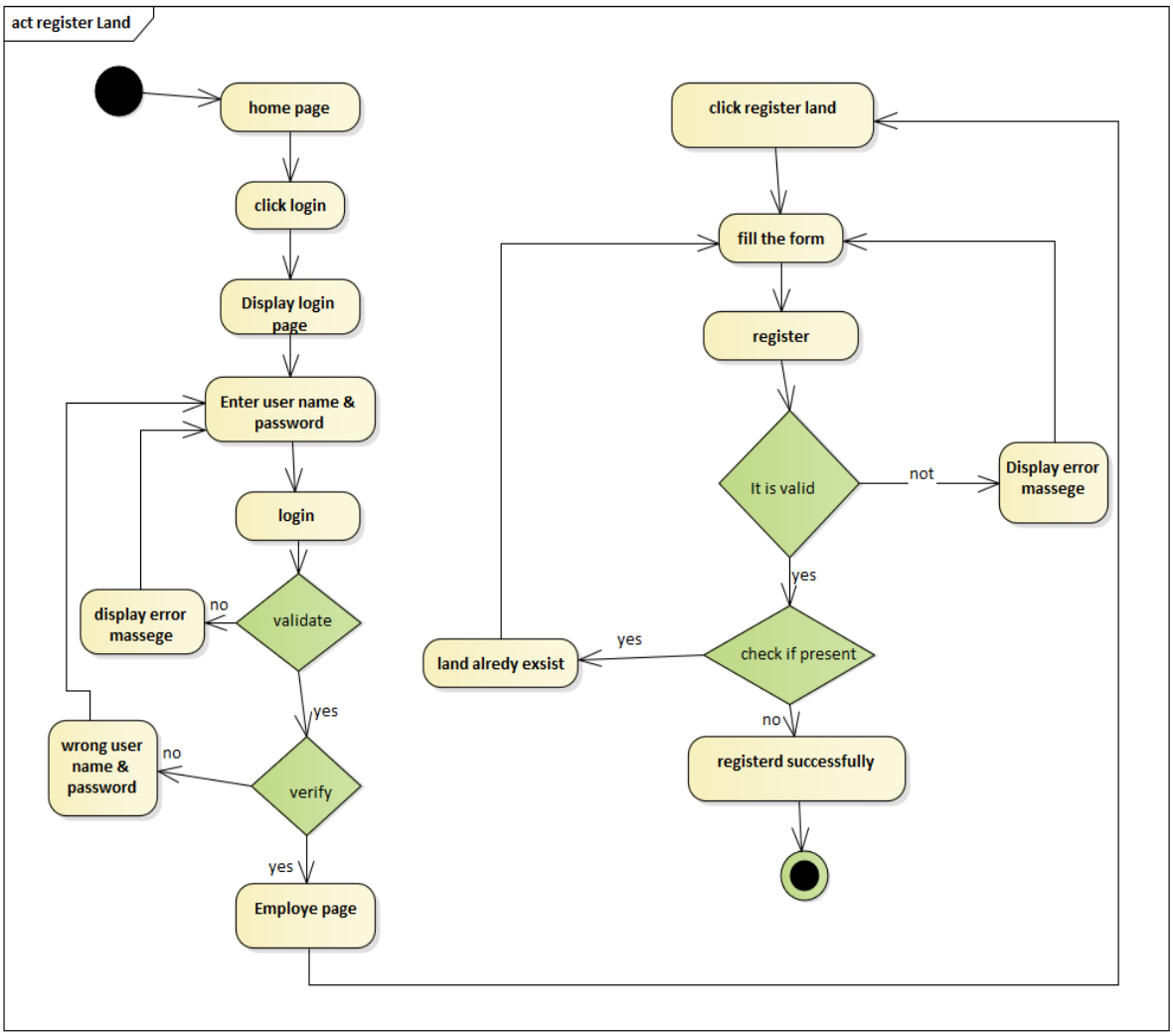


Figure4.7 Activity diagram for register Land

4.3.3. State chart diagram

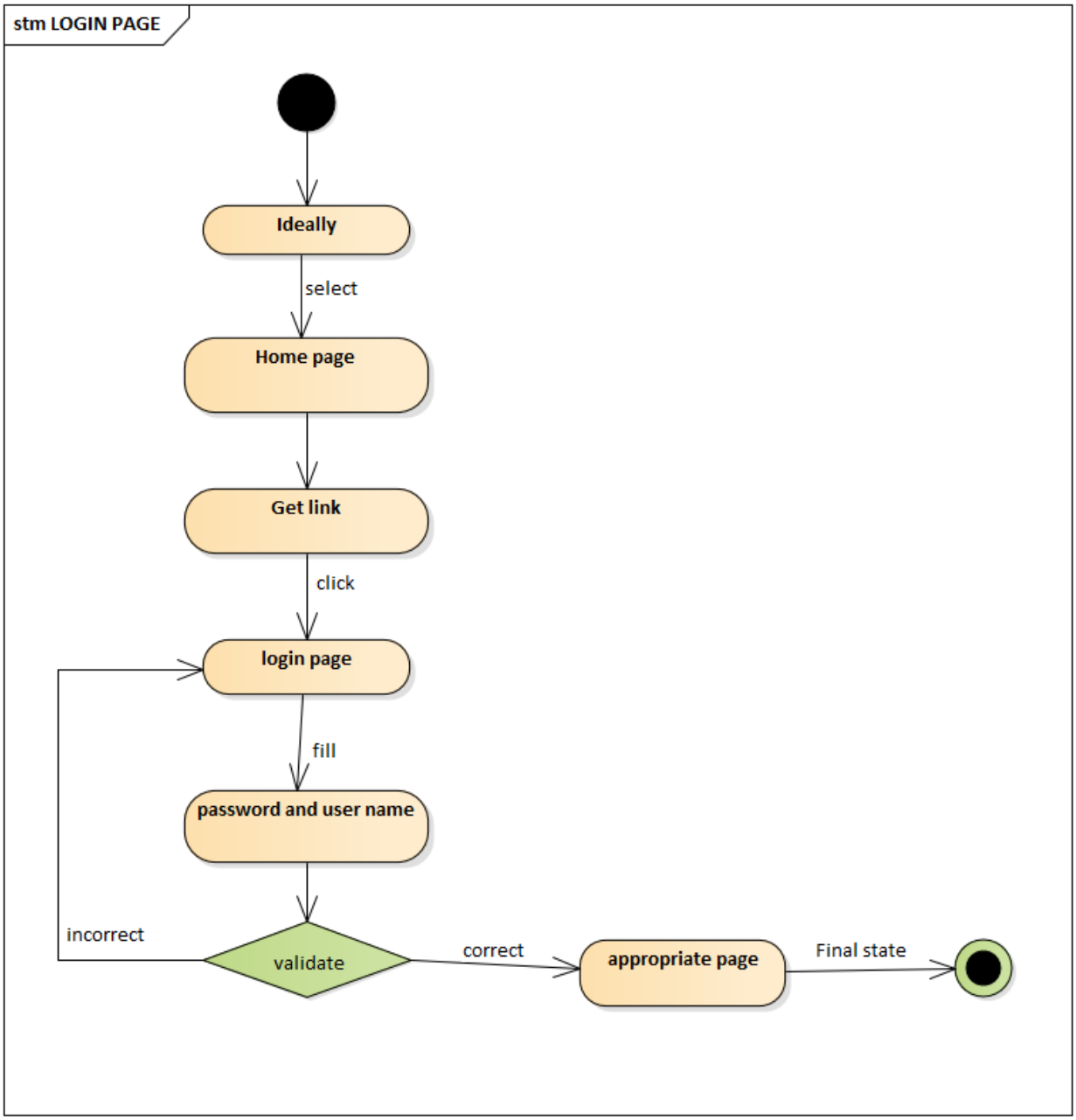


Figure4.8 state chart diagram for login

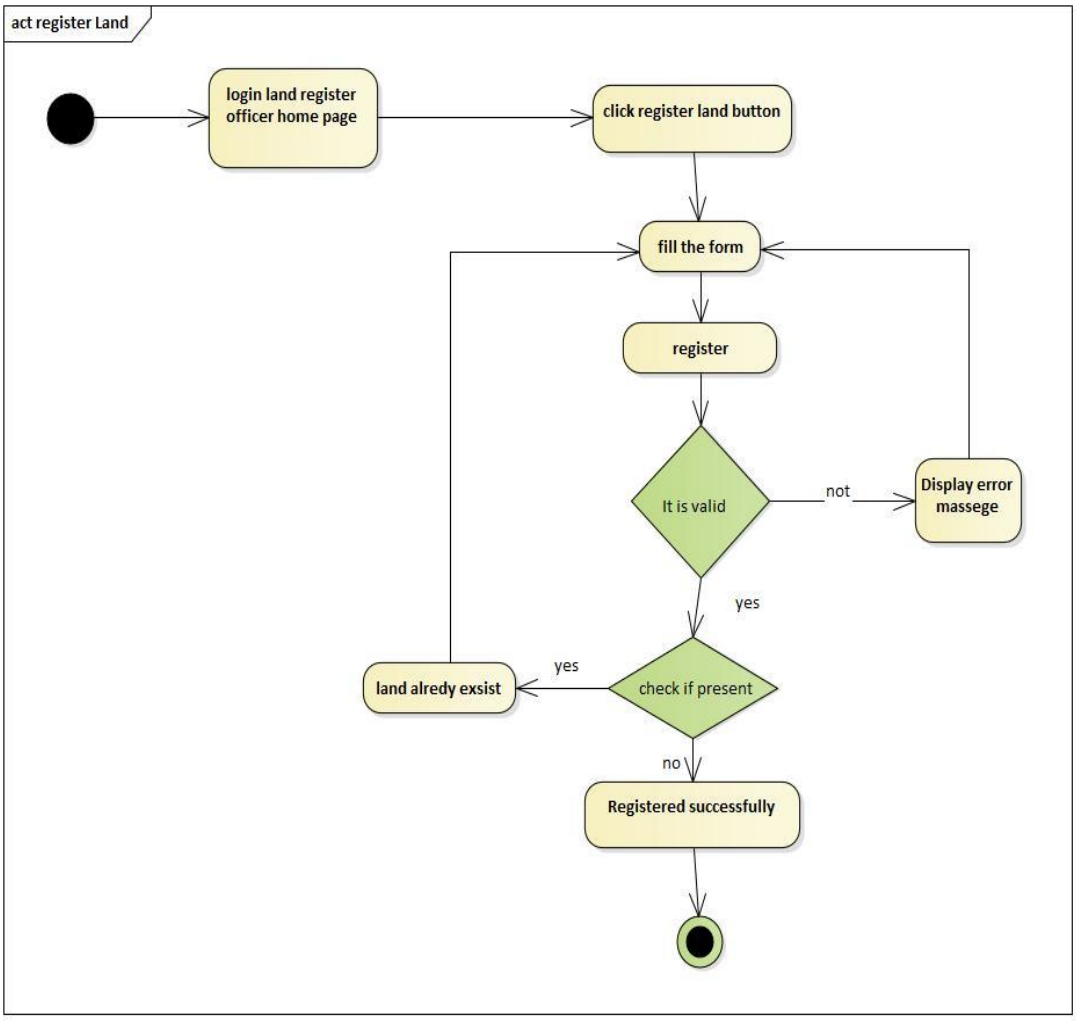


Figure4.9 state chart diagram for register land

CHAPTER FIVE

5. SYSTEM DESIGN

This chapter provides a brief overview of the design goals, current and proposed software architecture, Hardware/software mapping, Persistent data management, and Access control and security.

5.1. Design Goals

The design goals are derived from the non-functional requirements of the system, which describes the land management systemt hat specifies the qualities of the system that should be achieved and addressed during the design of the system.

5.1.1. User interface

The system intends to provide a user-friendly and clean GUI using a consistent theme throughout all to avoid user confusion, with intuitively understandable buttons, tooltips, and wizards. Moreover, the GUI will consider different browsers and monitor the resolutions of the user workstation.

5.1.2. Security

The system should be secured, i.e., not allow unauthorized users to access the database system, and also the system uses a hash function algorithm. In our system, no one can access the system without getting a privilege by the system. The system will not allow unauthorized users to add or delete packages in the system.

5.1.3. Performance

The system will have good performance as much as possible this will be attained via easily loadable interface components and optimal algorithms and database performance tuning, which make searching, updating, deleting, inserting, and saving easy and fast.

5.1.4. Fault tolerance

Whenever the user commits errors intentionally or unintentionally the system should continue performing its task without interruption by displaying the user-friendly error message.

5.1.5. Accessibility

Accessibility is the ability of the system to be accessible to users. since our project is web-based and accessed through web browsers using computers and mobile phones all land management system users can access the system very easily.

5.2. Proposed System Architecture

In this project, the team uses a three-tier architecture that has three layers. These three layers are the Application or Presentation layer, the business layer, and the data access layer.

- The application or presentation layer is the form that provides the user interface to either programmer or end-user.
- The business layer is the class that the team uses to write the function which works as a mediator to transfer data from the application layer or presentation layer to the data layer. This layer also has a property layer which is a class where variables are declared corresponding to the fields of the database which can be required for the application and make the properties so that the team can get or set the data using these properties into the variables.
- The data access layer is also a class to get or set data to the database queries back and forth. This layer only interacts with the database. The database queries or stored procedures will be written here to access the data from the database or to perform any operation to the database.

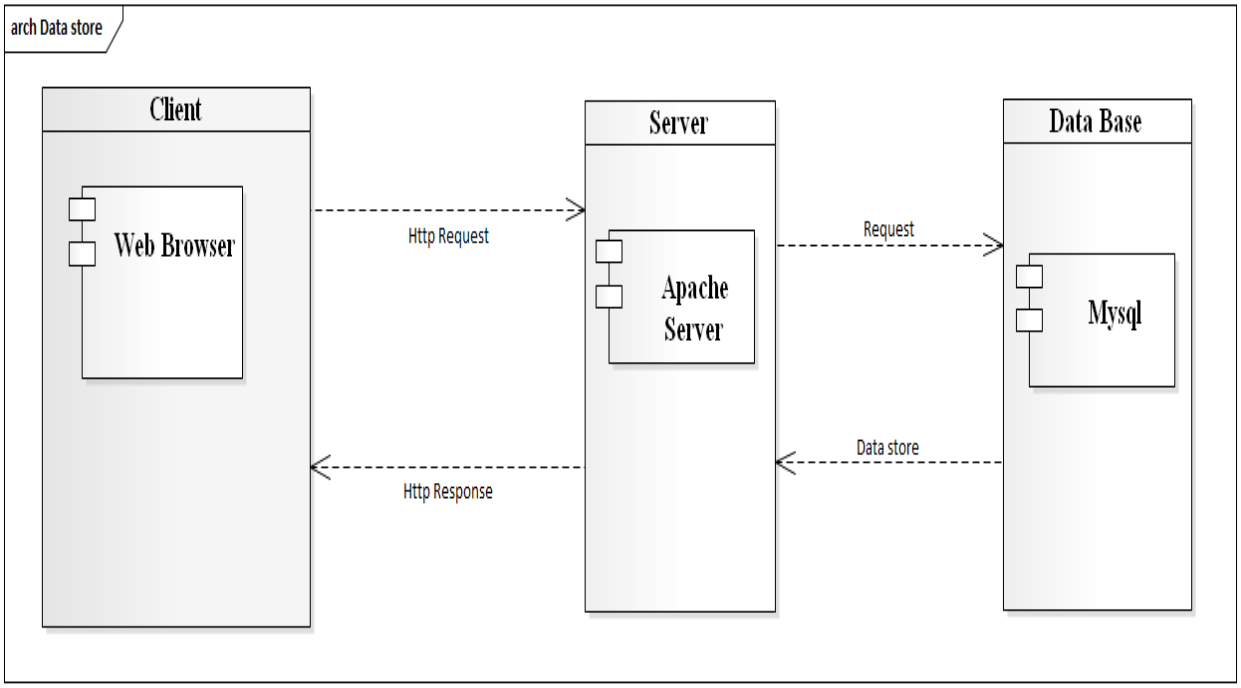


Figure5.1 System architecture

5.2.1. Subsystem Decomposition and Description

Subsystem decomposition is the process of dividing the land management system into manageable subsystems from the analysis model of the system. Subsystem decomposition of the proposed system is all about dividing the system into manageable subsystems from the analysis model of the system. Our goal in the subsystem decomposition is to reduce the complexity of the design model of a land management system and to distribute the class of the system into large-scale and cohesive components. The different components of our system are likely to interact while the system is in operation to provide the expected services. In our system the following subsystems are available.

In our system the following subsystems are available.

- Land Owner Subsystem
  - ✓ View land information
  - ✓ Apply for lease right
- Employee Subsystem
  - ✓ Add employee

- ✓ Delete employee
- ✓ Edit employee
- ✓ View employee

- Building Permit Subsystem
  - ✓ View report
  - ✓ Approve building permit
- User Subsystem
  - ✓ Delete user
  - ✓ Edit user
  - ✓ Add user
  - ✓ Search user
- Land Information Management Subsystem
  - ✓ Register land information
  - ✓ Update land information
- Permission Subsystem
  - ✓ Delete permission
  - ✓ Add permission
  - ✓ View permission
  - ✓ Search permission

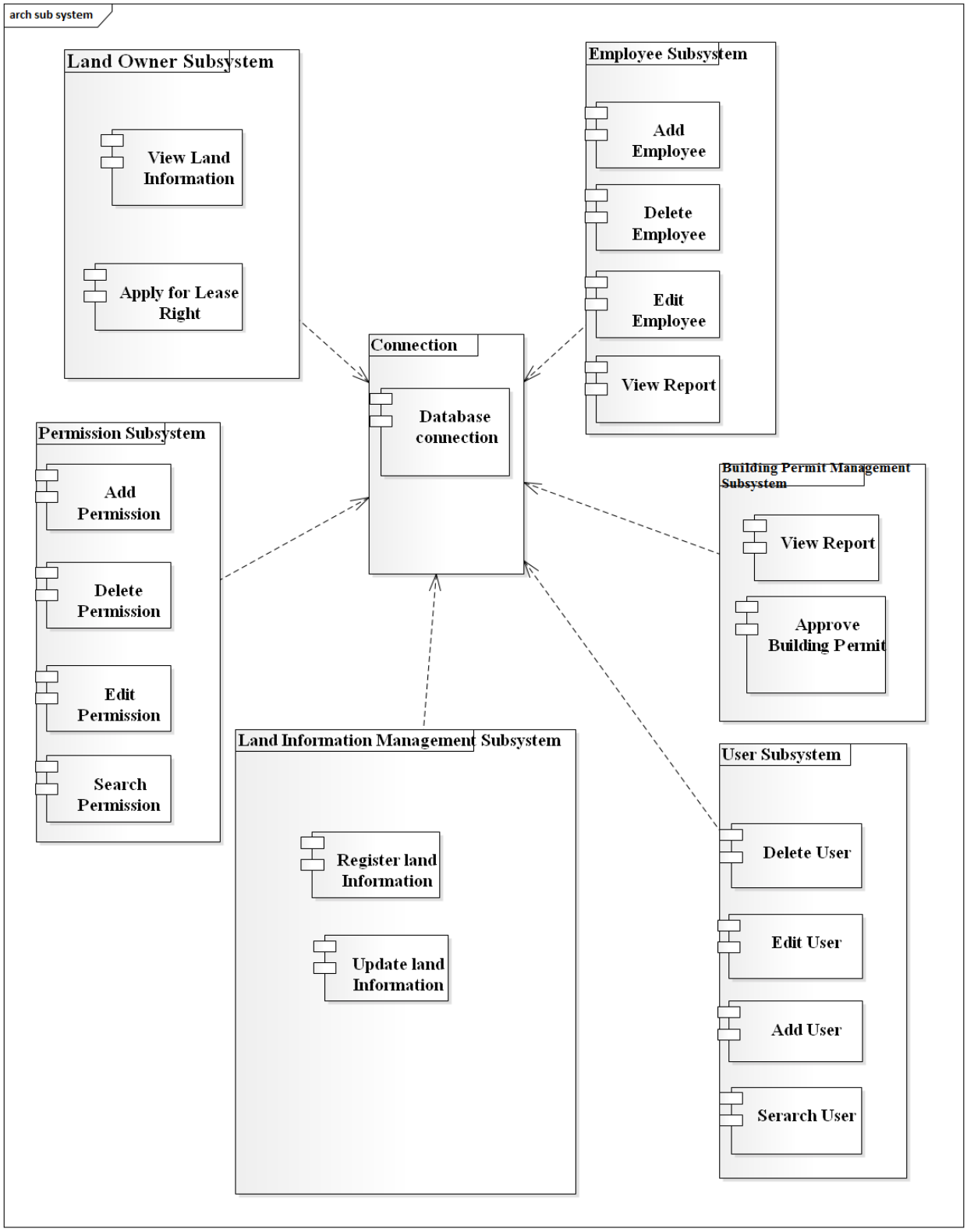


Figure5.2 Subsystem Decomposition and Description

5.2.2. Hardware/Software Mapping

The hardware/software map shows how land management systema nd the hardware components are work together. Their interaction can be shown by using a deployment diagram which shows a static view of the run-time configuration of processing nodes and the components that run on those nodes

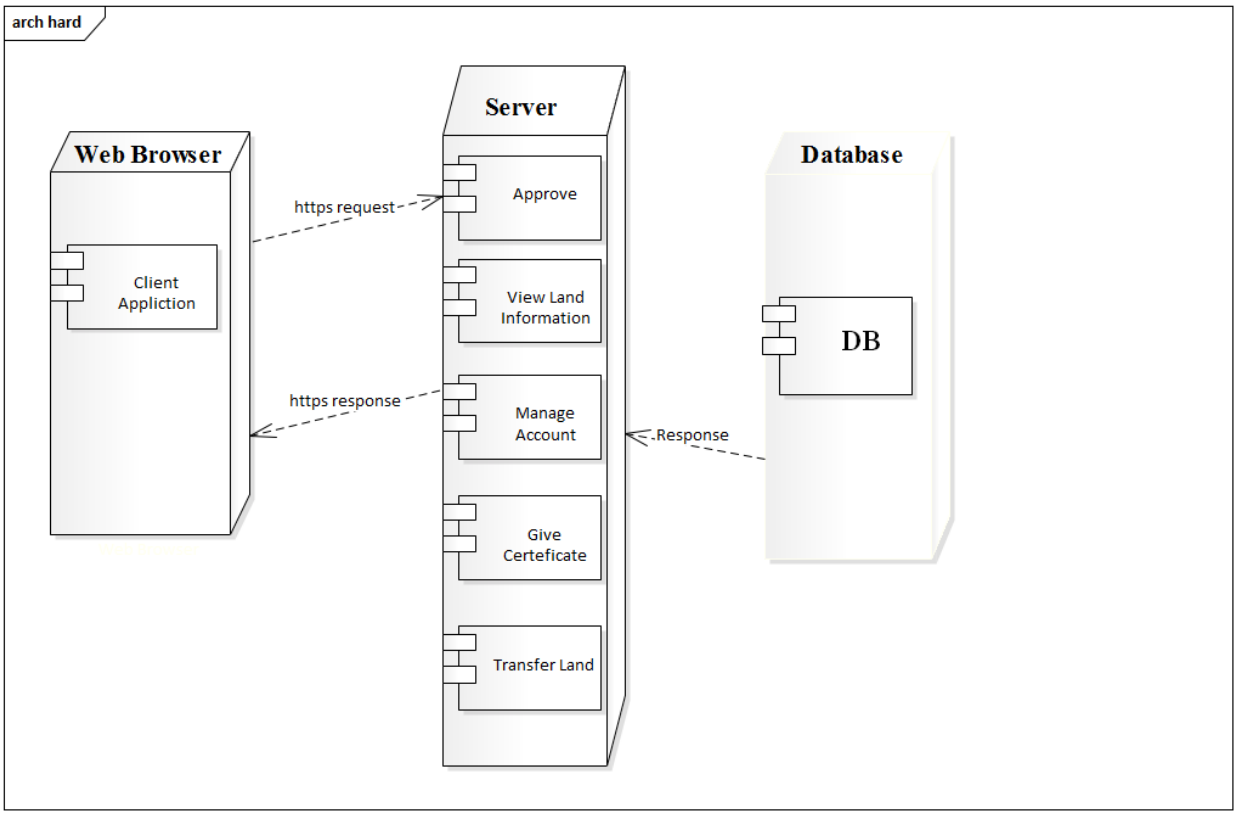


Figure5.3 Hardware/Software Mapping

5.2.3. Detailed Class Diagram

This section shows, classes, attributes, methods, attribute data types, visibility ((Public (+), Private (), Protected (#)) of attributes and methods), inheritance, association, aggregation, composition, dependency, and municipality (cardinality and optimality). Here, uses the UML class diagram to specify attributes and operations with their visibility information.

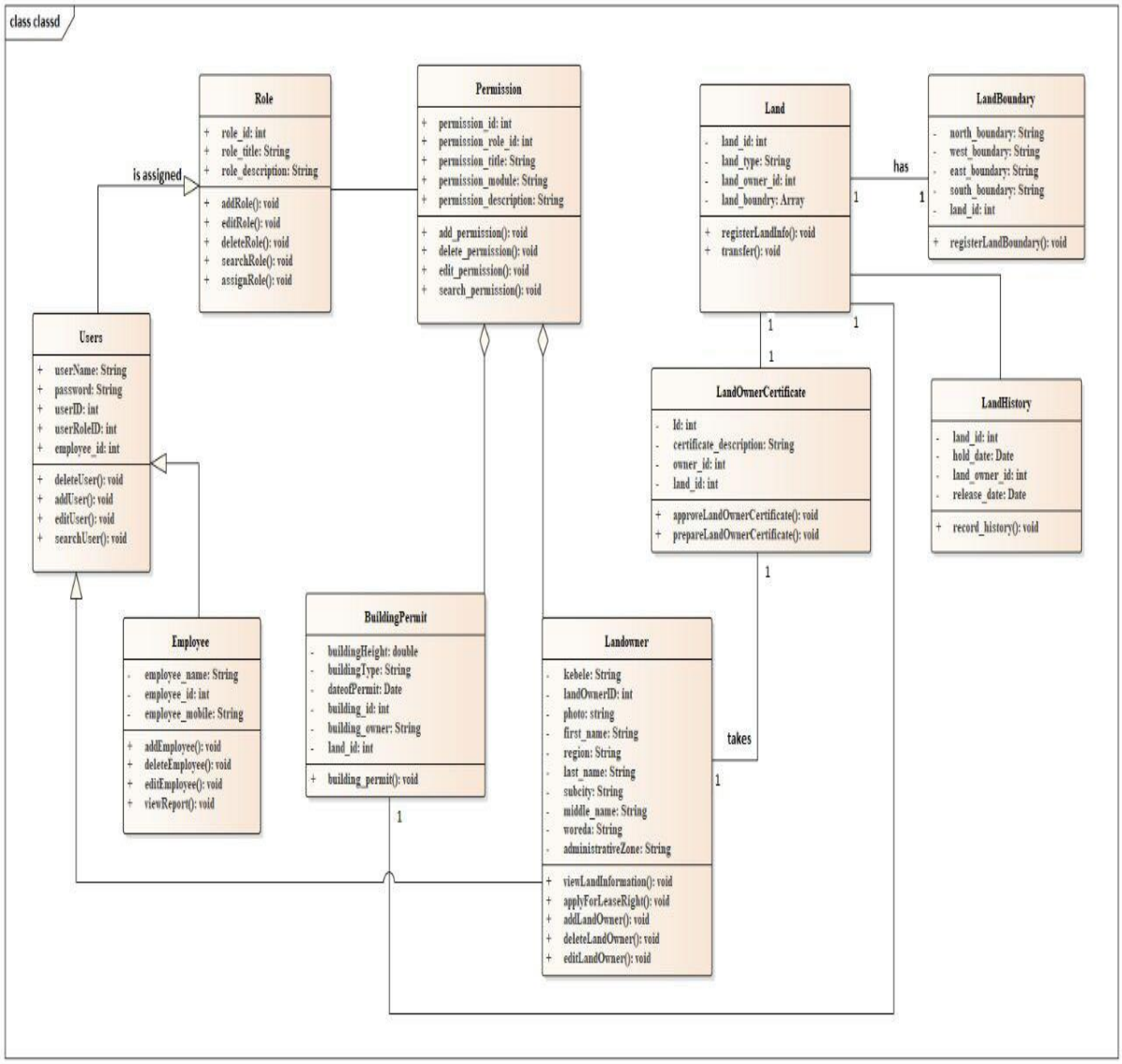
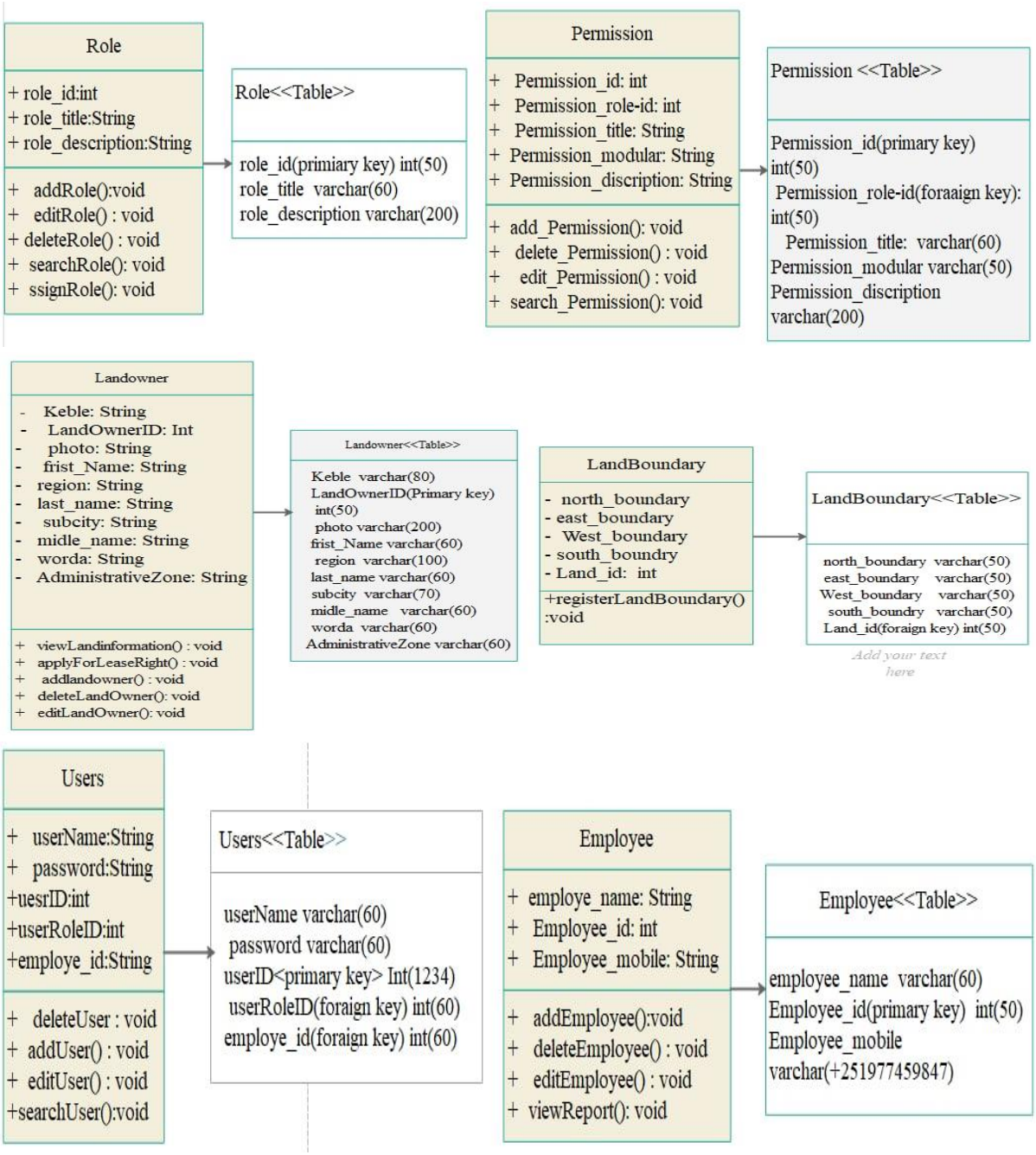


Figure5.4 Detailed Class Diagram

5.2.4. Persistent Data Management



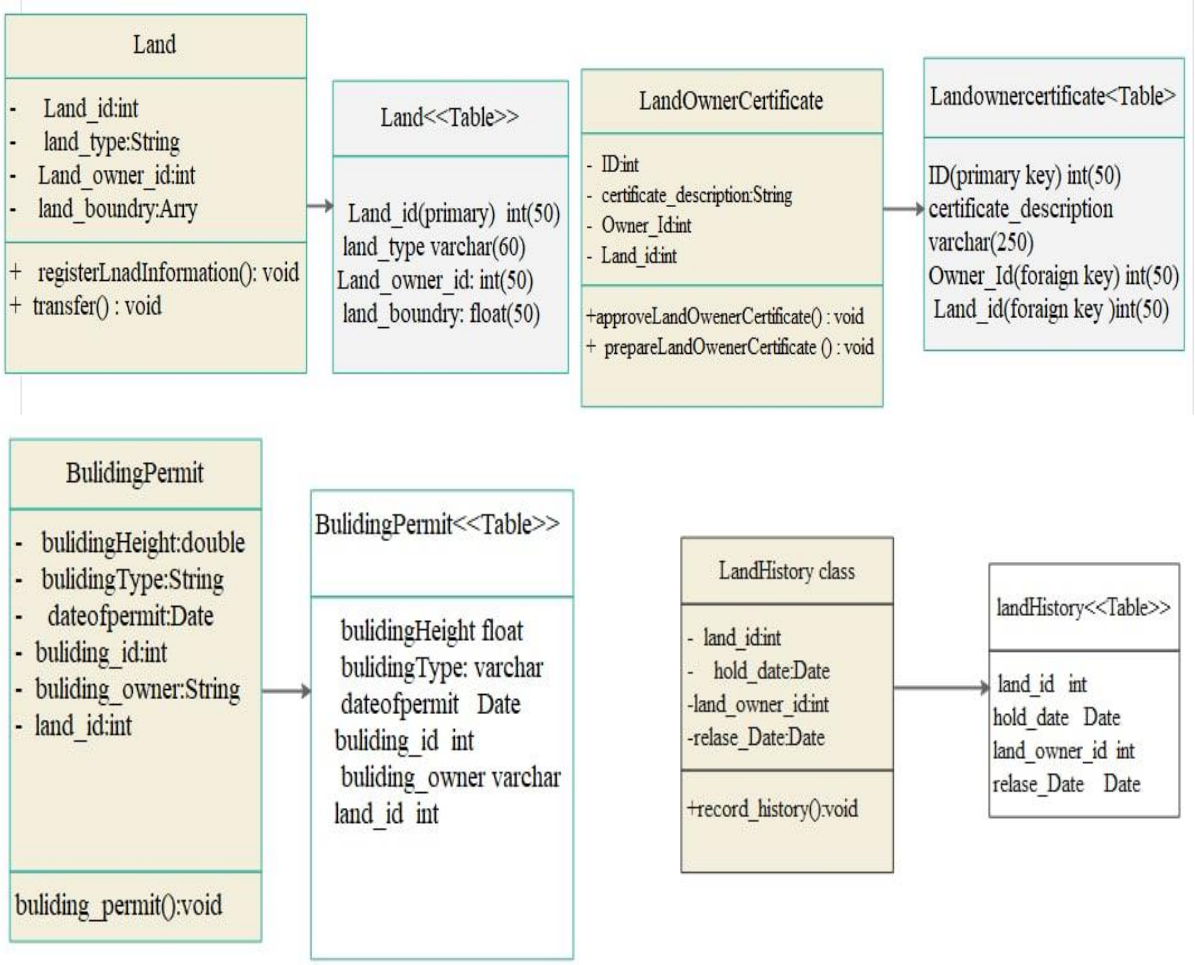


Figure5.5 Persistent Data Management

5.2.5. Access Control and Security

Access control is one of the security techniques that we designed to use. By access control, we mean that a technique to regulate who can or who cannot use the computing environment or the system. For this system, we have designed this table to show who is able and who is not able to do so.

Table5.1 Access control and table

Subsystem	Operations	Actor					
		LO	SA	LA	Mayer	LS	LRO

Land Owner Subsystem	View land information()	✓	✓	✓	✓	✓	✓
	Apply for lease right()	✓					
Employee Subsystem	Add employee()			✓			✓
	Delete employee()			✓			✓
	Edit employee()			✓			✓
	Search employee()			✓			✓
Building Permit Subsystem	View report()			✓	✓	✓	✓
	Approve building permit()				✓		
User Subsystem	Delete user()		✓	✓			✓
	Edit user()		✓	✓			✓
	Add user()		✓	✓			✓
	Search user()		✓	✓			✓
Land Information Management Subsystem	Register land information()		✓	✓			✓
	Update land information()		✓	✓			✓
Permission Subsystem	Delete permission()				✓		
	Add permission()				✓		
	View permission()				✓		

	Search permission()				✓		

5.3. Packages

Package diagrams of land management systemr eflect the organization of packages and their elements and provide a visualization of their corresponding namespaces. The following figure represents the general package diagram of the land management system.

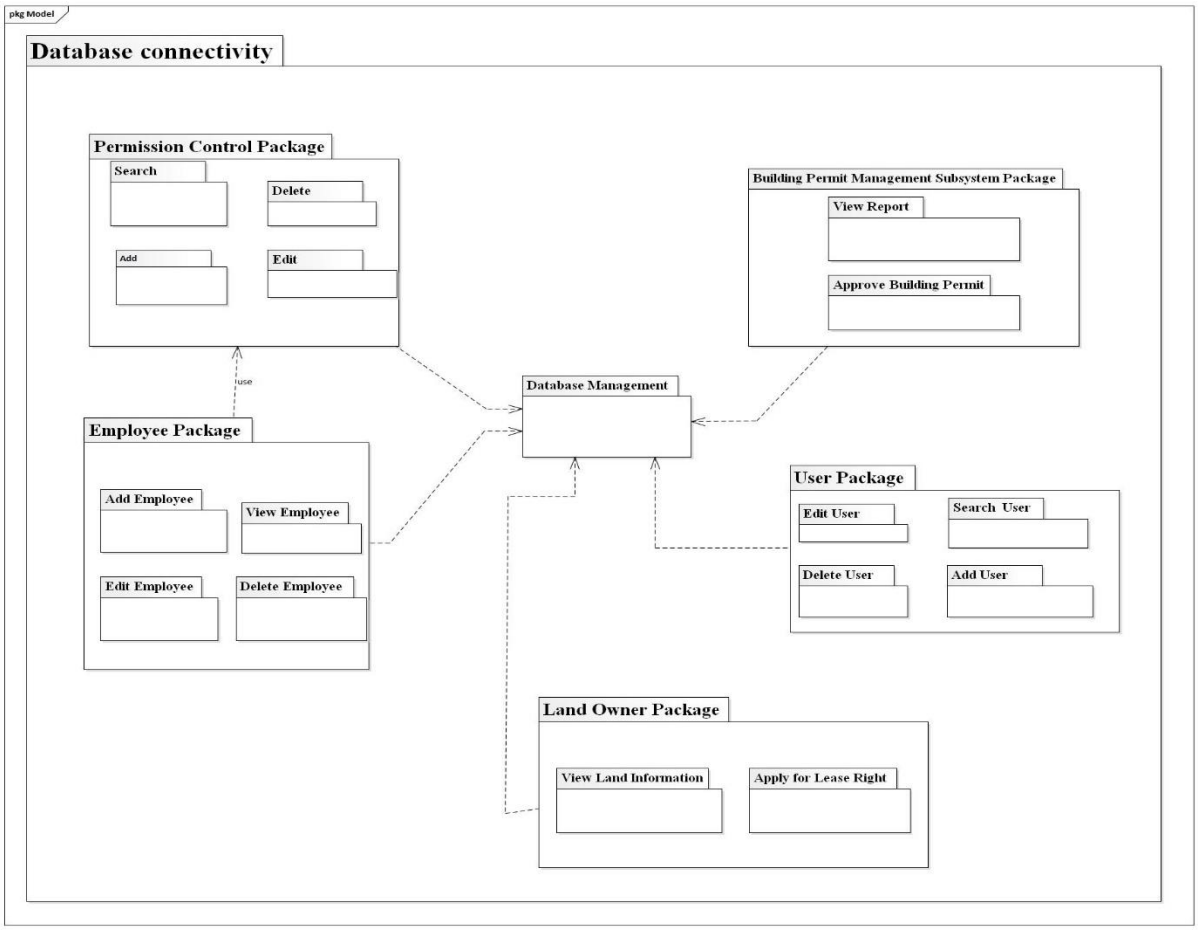


Figure5.6 Packages

5.4. Algorithm Design

One common problem-solving technique includes analyzing a problem, outlining the problem requirements and modeling steps called a logic model. Algorithms are used to solve the

problem. We also write a simple algorithm for some basic methods using human language (pseudo-code) in the following way.

---------For Login……..

Login ()

{

Insert username and password on the login form

Check the matching username and password

If (match)

Display “the Perspective user page”

Else

Display “error message, and try again”;

}

----------For Registering Land----------

Register_Land()

{

Insert Land Id, Kebele, Owner, Land Use, Relatives, Area, Address, Coordinates on Land Registration Form

Check if the form filled correctly

If (correct)

If (Correct)

{

Check for user existence

If (Exist)

Display User Information

Then deactivate user}

Display “Not found, try again”

}

----------Forgiving Approval----------

Approve ()

{

Login ()

If (View Request! =Null)

Then the manager will approve the request

Else

Nothing is done

}

----------For Inheritance Transfer----------

Inheritance Transfer ()

{

Login ()

Insert Owner ID, receiver (Kebele ID, name, age, gender, phone number, and relation), Kebele approval, court approval on transfer form, and then submit. If (correct)

Display “Successfully transferred”

Else

Display “Error, try again”

}

----------For Edit Account----------

Edit Account ()

{

Insert username, old password, and new password on edit account form If (correct)

Display “successfully edited”

Else

Display “error, try again”}

5.5. User Interface Design

Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed.

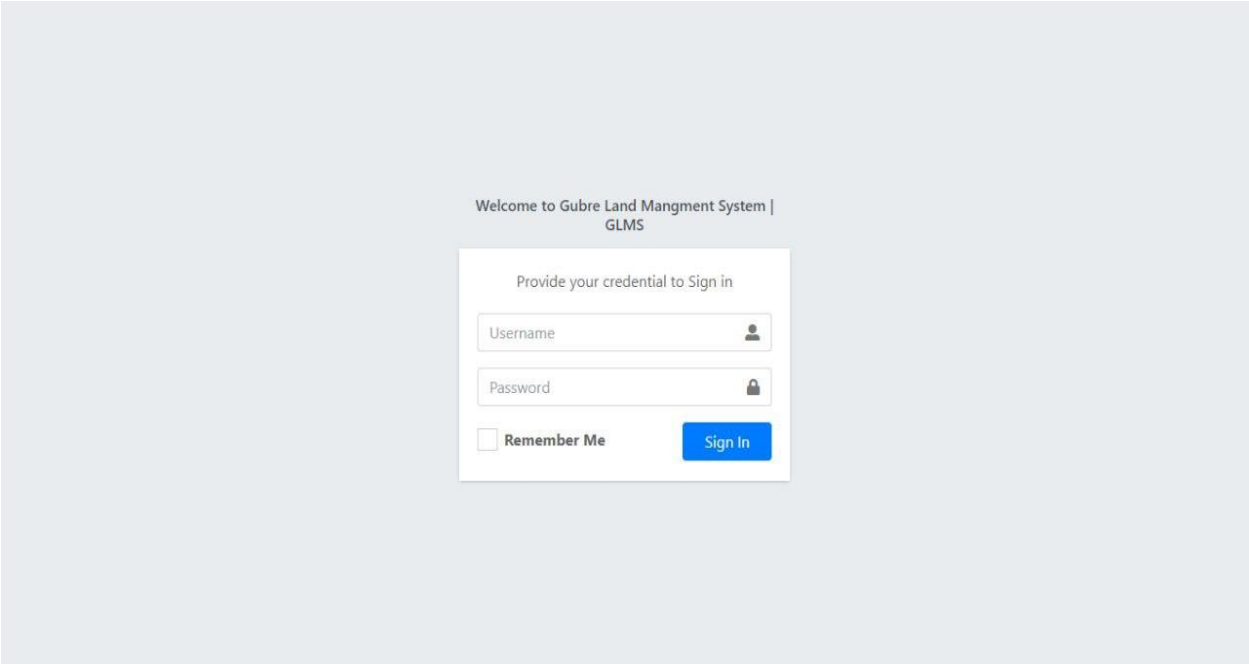


Figure5.7 Login interface

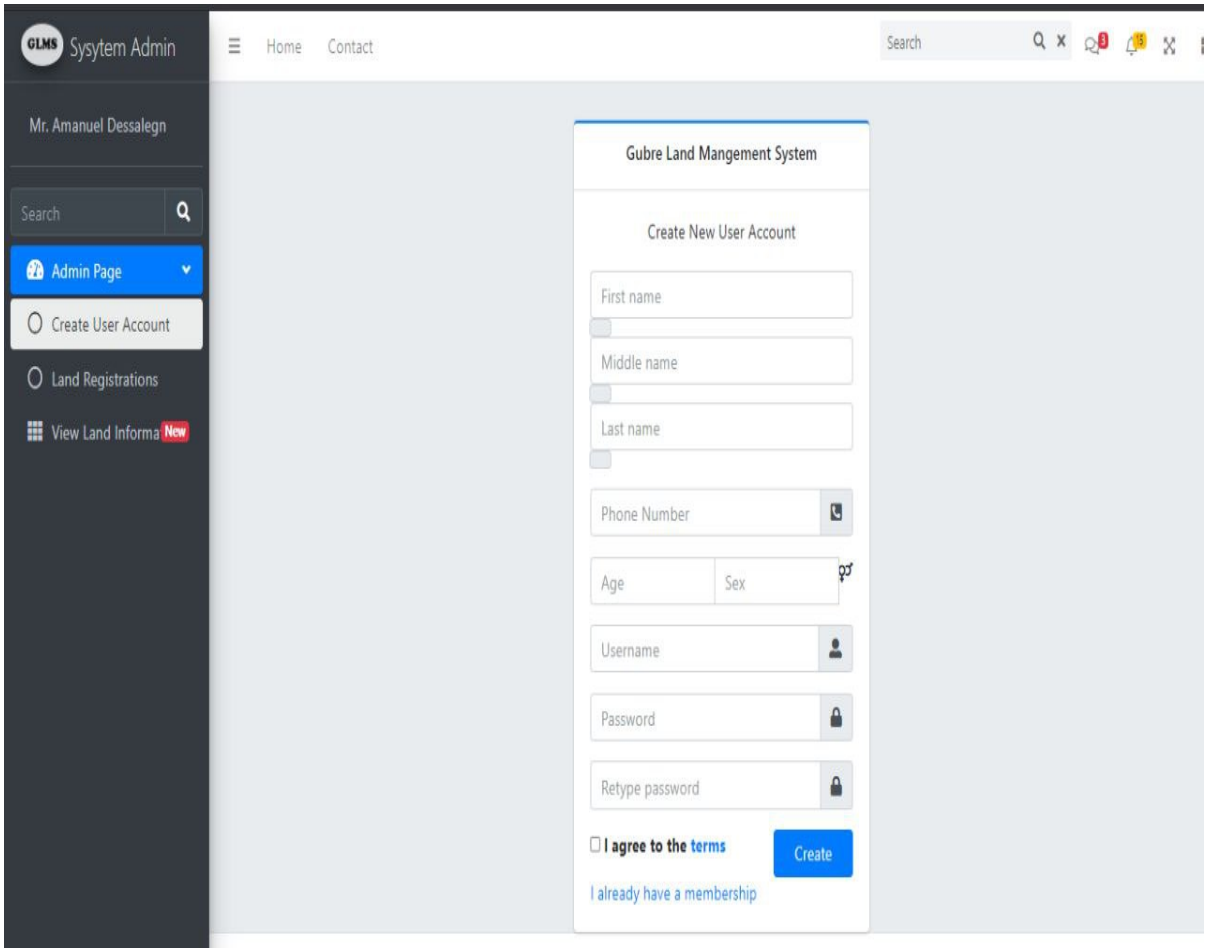


Figure5.8 Create account

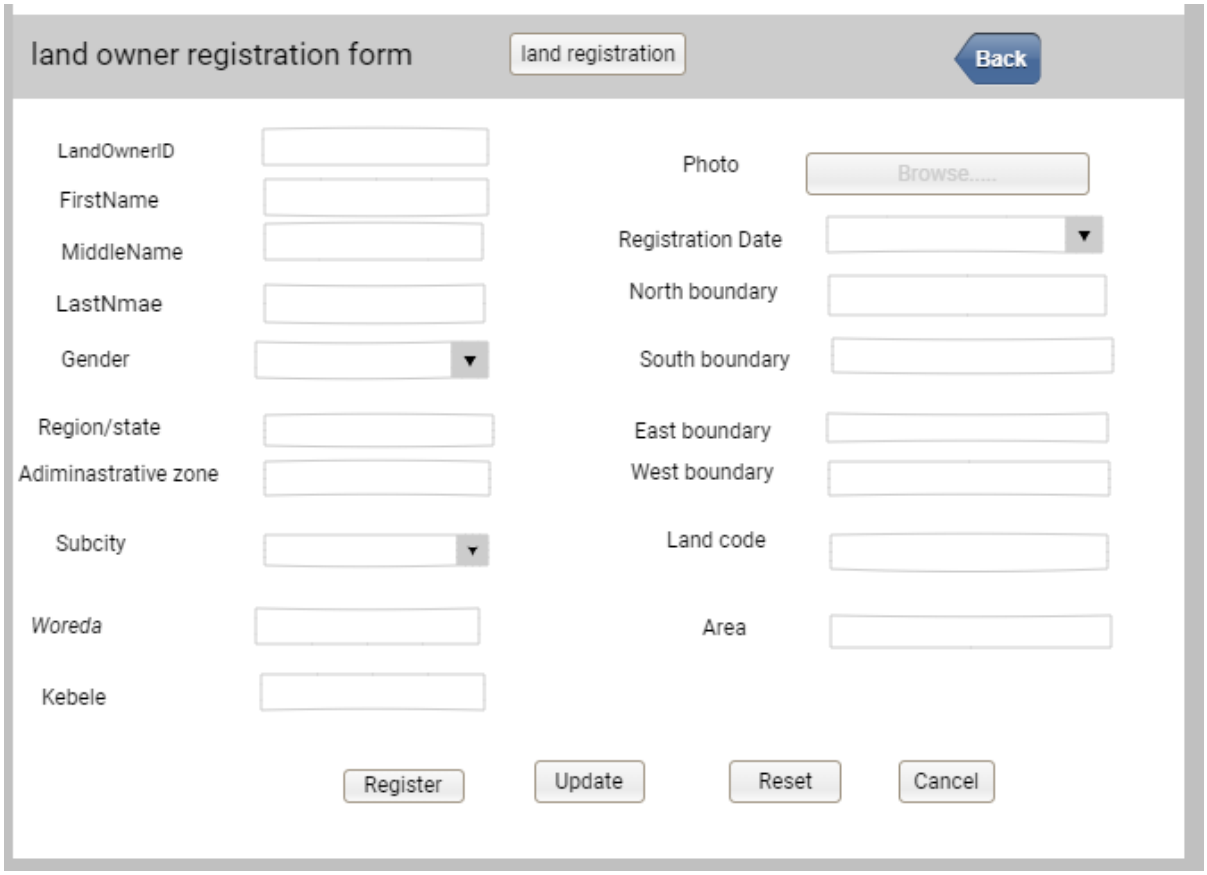


Figure5.9 Land owner registration form

REFERENCES

Banhazi, T. M. (2012, 05 08). Retrieved from https://www.researchgate.net/profile/Thomas-Banhazi

Gray, J. (2016, 12 13). Retrieved from https://www.researchgate.net/figure/Hardware-software-mapping-of-the-system-shown-using-a-UML-deployment-diagram_fig2_311634500

Ross, D. (1985, 05 12). Retrieved from https://www.computer.org/csdl/magazine/co/1975/05/01649428/13rRUILLkyg

Semmlow, J. (2002, 10 05). Retrieved from https://www.sciencedirect.com/topics/psychology/systems-model

Wazlawick, R. S. (2014, 06 23). Retrieved from https://www.sciencedirect.com/topics/computer-science/functional-requirement