GeoLand

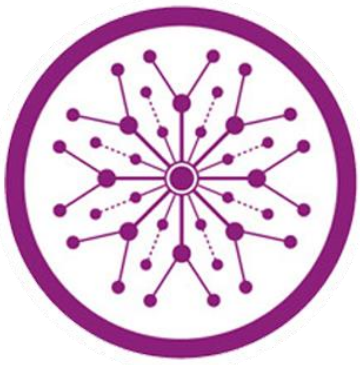
(Blockchain based Land Registry system)

Final Year Project

Session 2019-2022

A project submitted in partial fulfillment of the degree of

BS in Software Engineering



Department of Software Engineering

Faculty of Computer Science & Information Technology

The Superior University, Lahore

Spring 2022

Type (Nature of project)	[✓] **D**evelopment [ ] **R**esearch [ ] **R&D**
Area of specialization	
FYP ID	**BSSE-FYP-S22-008**

Project Group Members

Sr.#	Reg. #	Student Name	Email ID	*Signature
(i)	BSEM-S19-009	Abdullah Asif	bsem-s19-009@superior.edu.pk	
(ii)	BSEM-S19-024	Abdullah Mohsin	bsem-s19-024@superior.edu.pk	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

Plagiarism Free Certificate

This is to certify that, I **Abdullah Asif** Son of **Muhammad Asif**, group leader of FYP under registration no BSEM-S19-009 at Software Engineering Department, The Superior University, Lahore. I declare that my FYP report is checked by my supervisor.

Date: 20 Jun, 2022 Name of Group Leader: Abdullah Asif Signature: _____

Name of Supervisor: Muhammad Ahmad Aqib

Designation: Lecturer

Signature: _____

Co-Supervisor: Muhammad

Designation: Lecturer

Signature: _____

HoD: Dr. Tehreem Masood

Signature: _____

GeoLand

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
	1.0		<Original Draft>	
			<Changes Based on Feedback from Supervisor>	
			<Changes Based on Feedback From Faculty>	
			<Added Project Plan>	
			<Changes Based on Feedback from Supervisor>	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

_____

Name:_____

Date:_____ Signature:_____

PROJECT MANAGER

Comments: _____

_____

Date:_____ Signature:_____

HEAD OF THE DEPARTMENT

Comments: _____

_____

Date:_____ Signature:_____

Dedication

This study is whole-heartedly dedicated to our beloved parents and teachers, who have been our source of inspiration and gave us strength when we think of giving up,

who continually provide their moral, spiritual, emotional, and financial reports. I dedicate this project to God Almighty, my creator, my strong pillar, my source of inspiration, wisdom, knowledge, and understanding. He has been the source of my strength throughout this program and only have I soared in his wings. I also dedicate this work to Teacher Mr. Muhammad Ahmed and all the professors of superior who has encouraged me all the way and whose encouragement have made sure that I give it all it takes to finish that which I have started. I also dedicate this to my fellows who supported me throughout the process.

Acknowledgements

I would like to thank Professor Muhammad Ahmed for his expert advice and encouragement on this difficult project as well as Professor Muhammad Ahmed and all superior professors and teachers for his brilliance in the class and he taught me

how to write a report on a relevant topic. This enhanced my skills in writing the report and further research papers and articles etc. I would like to thank those people who wrote reports on this topic in the past and I got many concepts and ways of writing the report from those reports and I am very thankful to Google too. This project is impossible without networks and search engines.

Executive Summary

Blockchain technology is a peer-to-peer protocol that can be leveraged to keep track of transactions over the internet. Publicized for its use in the bitcoin revolution, the technology provides transparency and traceability that can be used in the management of land rights. When it comes to the formalization of land rights, blockchain technology promises to authenticate owners and other users of land, and

provides a fixed ledger of land use rights transactions. At present, blockchain technology is being explored as a proof of concept in several countries to track land titles. We extend the idea to capture the granting of land use rights (individual to individual) making use of the decentralization, peer-to-peer nature of blockchain technology. This technology can offer an effective means to manage land transactions, provide digital documentation to actors in the informal land rental market and reduce inefficiency in land systems. However, the uptake of the technology in land administration is limited by human related factors. These limitations include, but are not limited to, the accuracy of data being entered into the system, the ability of the system to facilitate data preservation, pre-existing institutional and legal pillars, and the digital divide across communities. Part of overcoming these barriers requires the political will of governments to invest in digital technologies and develop institutional capacities to overcome current limitations to bring land management into the industry.

Table of Contents

Table of Figures

Table of Tables

Chapter 1

Introduction

Chapter 1: Introduction

The system of land registration in any use of a is thought to be a multistep system, because it involves the engagement of all stakeholders who can have a right away or oblique stake within the registration. The presently used land file identify garage gadget increases fundamental problems approximately records fraud, the safety of fantastically touchy records, and the threat of gadget failure because of herbal disasters, along with the server used for records garage going down. Compared to the contemporary processes and strategies for land identify control and records garage, block-chain is a modern era and database that has the capacity to absolutely cope with the troubles that plague contemporary systems. The fundamental and maximum critical element of block-chain era is that it's far a decentralized community wherein all records provided via way of means of a unmarried node are showed via way of means of all different to be had nodes, and best after a consensus is made, can then the shared records be stored to the block-chain.

1.1. Background

A land registry is a way for government agencies to register and record ownership and rights to land. These records verify land ownership, promote land transactions and avoid corruption. Existing land registration models have slowed valid land transactions and proof of ownership. The Land Register is a system by which government agencies document ownership and land ownership changes in accordance with existing laws and regulations to protect landowner rights and facilitate land management. Currently, land registration methods are generally established in Western countries. Usually, it is classified into three types: contract registration, right registration, and Torrens registration. Zhang Wei compares and links land registration systems in France, Germany, the United Kingdom and Australia.

1.2. Motivations and Challenges

Pakistan does not have such a framework for implementing a block-chain-based land registration system that can used for the actual implementation of decentralized land registration. A systematic contribution is to provide an important and certified conceptual framework for block-chain-based land registration systems that can

ensure transparency, security, and rights without the need for a trusted third party. Centralized servers for land registration data face many problems. Data loss due to natural disasters biggest and most realistic problem is that the land registrar that handles the data can also tamper with the data, and many stakeholders who are the actual owners of the property have real rights. Is that you may not be able to find. Our motivation and goal is to find a solution to the shortcomings of centralized land management systems using decentralized and decentralized block-chain technology.

1.3. Goals and Objectives

- Users register to the platform
- Sellers upload the property specifications on the platform
- Buyers request access to the listed property
- Sellers approve the transfer request and land inspector gets the notification
- Land Inspector verifies the transaction and initiates the transfer
- Land Registry Document Validation and Authenticity

1.4. Literature Review/Existing Solutions

- We gathered requirements of Pakistani peoples / landowners and provide solution block-chain based Land Registry name Geo Land.
- There is a need of this kind of platform for Land Registry.
- We also need to spread awareness through social platforms. This Geo Land web app help the people to get their rights.

1.5. Proposed Solution

This use case of blockchain goes beyond just a database to take advantage of the ability to create permanent, unbreakable evidence of ownership of land or real estate. The simplest implementation of blockchain-based land registration allows you to capture an ownership document and associate it with the owner's user account. If the property sold, all relevant documents can transferred to the new owner. All transactions are traceable, time stamped and indisputable. In this way, the blockchain

can provide a very secure record of ownership that cannot tampered with. Paper-based systems cannot provide this flexibility, resilience, and durability.

1.6. Project Plan

Project plan includes Work Breakdown Structure (WBS) and Gantt Chart. WBS describes complete project decomposition by breaking down its structure. And Gantt chart will describe the timeline and time strategies about the whole project deliverables.

1.6.1. Work Breakdown Structure

- **Requirements**
  - Requirement Gathering
  - Literature Review
  - Informal Requirements
- **Analysis**
  - Analysis Process
  - Formal Documentation
  - Software Requirement Specifications (SRS)
- **Design**
  - Software Architecture
    - UML Diagram
    - Class Diagram
    - Deployment Diagram
    - Activity Diagram
    - Use case Diagram
    - ER-Diagram
    - Sequence Diagram
    - Package Diagram
  - Network Diagram
    - System Architecture (Service Oriented)
  - User Interface View
    - Web Application Interface

        - Android Application Interface

- **Implementation**
    - Web Front End
        - HTML
        - CSS
    - Web Back End
        - Laravel
    - Database Management System
        - My-SQL
        - SQL Server
- **Testing**
    - Module testing
    - System testing
- **Deployment**
    - Web Front End
        - React
        - HTML
        - CSS
    - Web Back End
        - Node
    - Database Management System
        - My-SQL
        - SQL Server

1.6.2. Roles & Responsibility Matrix

Table 1: WBS

WBS #	WBS Deliverable	Activity #	Activity to Complete the Deliverable	Duration (# of Days)
1	Literature Review	1.1.1	-	2
2	Informal Requirements	1.1.2	1.1.1	3
3	Analysis Process	1.2.1	1.1.2	2 (1st Week)
4	Formal Documentation	1.2.2	1.2.1	3

5	SRS	1.3	1.2.2	4 (2nd Week)
6	UML Diagram	2.1.1	1.3	3
7	Class Diagram	2.1.2	1.3	2
8	Deployment Diagram	2.1.3	1.3	3 (3rd Week)
9	Activity Diagram	2.1.4	1.3	3
10	Use case Diagram	2.1.5	1.3	3 (4th Week)
11	ER-Diagram	2.1.6	1.3	3
12	Sequence Diagram	2.1.7	1.3	3
13	Package Diagram	2.1.8	1.3	2 (5th Week)
14	System Architecture	2.3.1	1.3	3
15	Web UI Interface	2.2.1	1.3	3 (6th Week)
17	Web App front end	3.1	1.3 , 2.1 , 2.2 , 2.3.1	14 (8th Week)
18	Web App back end	3.2	1.3 , 2.1 , 2.2 , 2.3.1	28 (20th Week)
21	Database Management	3.5	1.3 , 2.1.6	21 (23rd Week)
22	Module testing	4.1	3.1 , 3.2 , 3.3 , 3.4	3
23	System testing	4.2	4.1	7 (25th Week)
24	Live web on server	5.1	4.2	2 (26th Week)

1.6.3. Gantt Chart

Table 2: Milestone-1

Milestone 1		
Name	**Start Date**	**End Date**
Requirement Gathering	**01/02/22**	**05/02/22**
1. Literature Review	01/02	02/02
2. Informal Requirements	03/02	05/02

Table 3: Milestone-2

Milestone 2		
Name	**Start Date**	**End Date**
Analysis	**08/02/22**	**018/02/22**
1. Analysis Process	08/02	09/02
2. Formal Documentation	12/02	12/02
3. SRS	15/02	18/02

Table 4: Milestone-3

Milestone 3		
Name	**Start Date**	**End Date**
Software Architecture	**19/02/22**	**20/03/22**
1. UML	19/02	23/02
2. Class Diagram	24/02	26/02
3. Deployment Diagram	26/02	29/02
4. Activity Diagram	1/03	05/03
5. Use Cases	06/03	08/03
6. ER Diagram	09/03	13/03
7. Sequence Diagram	14/03	16/03
8. Package Diagram	19/03	20/03

Table 5: Milestone-4

Milestone 4		
Name	**Start Date**	**End Date**
System Architecture	**21/03/22**	**23/03/22**
1. Network Diagram	21/03	23/03

Table 6: Milestone-5

Milestone 5		
Name	**Start Date**	**End Date**
User Interface	**26/03/22**	**05/04/22**
1. Web UI	26/03	05/04

Table 7: Milestone-6

Milestone 6		
Name	**Start Date**	**End Date**
Implementation	**03/05/22**	**14/09/22**
1. Web Frontend	03/05	30/06
2. Web backend	01/07	14/09

Table 8: Milestone-7

Milestone 7		
Name	**Start Date**	**End Date**
Database Management System	**15/09/22**	**12/10/22**
1. MySQL/SQL Server	15/09	12/10

Table 9: Milestone-8

Milestone 8		
Name	**Start Date**	**End Date**
Testing	**13/10/22**	**26/10/22**
1. Module Testing	13/10	17/10
2. System Testing	18/10	26/10

Table 10: Milestone-9

Milestone 9		
Name	**Start Date**	**End Date**
Deployment	**27/10/22**	**31/10/22**
1. Live web in server	27/10	28/10
2. Launch Android app on server	29/10	31/10

1.7. Empathy Map

An empathy map is a widely used visualization tool within the field of UX and HCI practice. In relation to empathetic design, the primary purpose of an empathy map is to bridge the understanding of the end user.

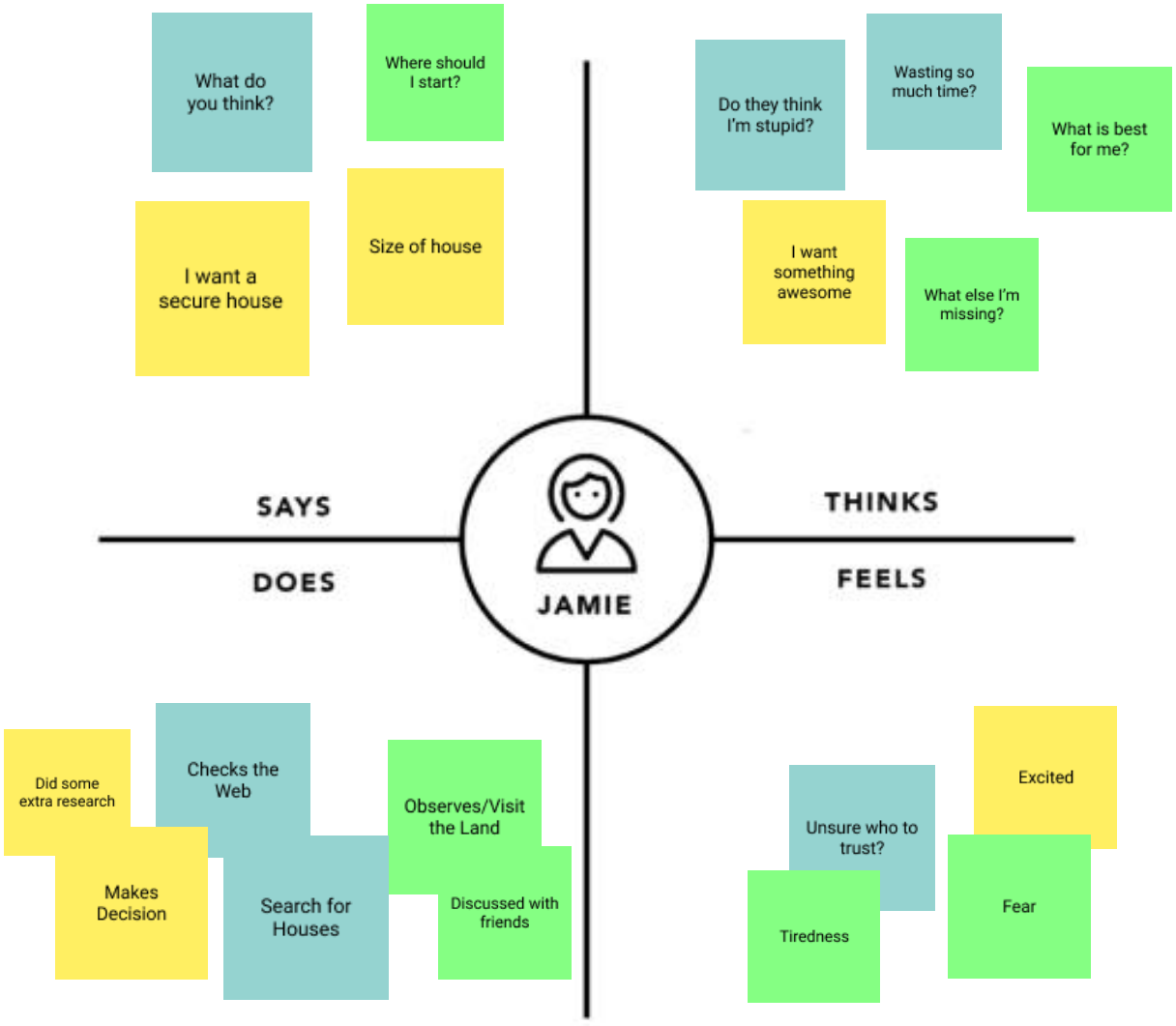


Figure 1: Empathy Map

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1. Introduction

2.1.1. Purpose

The purpose of using Blockchain in Land Registry system is because Blockchain provides a potential solution for many of the challenges of land registration. The simplest implementation of a blockchain-based land registry could enable the ownership documents to be recorded and assigned to the owner's user account. Every transaction and property details is traceable, timestamped, and indisputable. Used in this way, blockchain could provide a highly secure record of ownership that cannot be manipulated.

2.1.2. Document Conventions

Table 11 : Document Conventions

Name	Meaning
DDB	Distributed Database
ER	Entity Relationship
DB	Database
API	Application Programming Interface, which is a software intermediary that allows two applications to talk to each other.
Web Services	A web service is a piece of code that is available on through internet from a specific server and work as a standardized XML messaging system. The XML is used to encode all communication to web services.
Blockchain	Blockchain technology is a peer-to-peer protocol that can be leveraged to keep track of transactions over the internet. Publicized for its use in the bitcoin revolution, the technology provides transparency and traceability that can be used in the management of land rights.

2.1.3. Intended Audience and Reading Suggestions

The project has been implemented under the guidance of university and professors and Entrepreneurship professors. This project is useful for the people who are facing several issues in their property. This document is also readable for Stakeholders and Investors this Document and project are not only for FYP it is also readable for Several Companies and Housing Societies. This document also helps out the developer and project manager to implement the future requirements and future upgradations.

2.1.4. Product Scope

The currently used land record title storage system raises major issues about data fraud, the security of highly sensitive data, and the risk of system failure due to natural disasters, such as the server used for data storage going down. Compared to the current approaches and procedures for land title management and data storage, block-chain is a cutting-edge technology and database that has the ability to completely address the problems that plague current systems. The basic and most important aspect of block-chain technology is that it is a decentralized network in which all data supplied by a single node are confirmed by all other available nodes, and only after a consensus is made, can then the shared data be saved to the block-chain.

2.1.5. References

- **System Requirement Specification**

  This project is based on a web application that can connect the libraries and some external services the other feature like Requirement, Safety, and Security Requirements, and the quality attributes of this project are described below.

- **Source**

  IEEE Format for Software Requirement Specification

- **Document help Reference**

  IEEE Version 3.0

2.2. Overall Description

2.2.1. Product Perspective

2.2.1.1: Users register to the platform

- create the profile on the platform with details like name, government-issued ID proofs and designation
- A hash for the identity information submitted by the users gets stored on the blockchain.

2.2.1.2: Sellers upload the property specifications on the platform

- Sellers can upload properties' images and documents on the platform and pin the land's location on the map.
- Then record is stored in the Blockchain.
- Once the property's details are uploaded to the platform, it is made available to all users who have signed up as a buyer.

2.2.1.3: Buyers request access to the listed property

- A buyer interested in any specific property can send a request to access its specification to the seller.
- Sellers receive notification for property access requests. They can either deny or accept it by looking at the buyer's profile.
- Buyers can view the previous ownership records of the property and send a request to purchase it and initiate the transfer.
- Transactions corresponding to the requests made by both sellers and buyers are recorded on the blockchain to ensure authenticity and traceability.

2.2.1.4: Sellers approve the transfer request and land inspector gets the notification

- If the seller approves the land ownership transfer request, the land inspector gets the notification to initiate the transfer of property.
- Smart contracts trigger to provide land documents' access to the land inspector.

- After the land inspector verifies the documents, they schedule the meeting for ownership transfer with buyer and seller.
- The meeting record is also added to the blockchain to solve property related disputes if occur in the future.

2.2.1.5: Land Inspector verifies the transaction and initiates the transfer

- Land inspector verifies the documents submitted by buyers and sellers and adds the authenticated records to the blockchain land registry platform.
- Sellers and buyers sign the property ownership transfer document in front of the land inspector on the land registry platform.
- The signed document gets saved in the database and transaction corresponding to it is recorded on the blockchain.
- The transfer is initiated and smart contracts trigger to send funds to the seller and title's ownership to a new buyer.

2.2.1.6: Land Registry Document Validation and Authenticity

- In case of any disputes, any authorized party can upload the signed land registry document on the platform to check its authenticity and validate it.
- If hash generated after uploading the document is the same as that of the hash created at the time of signing the document, then the document is authenticated and no modifications have been made to the document.

2.2.2. User Classes and Characteristics

This system allows the user to enter his/her personal data. They can update and delete it too. There will be three users involved in the Blockchain Land Registry System:

- **Buyer:** A person who buys the land and uses the platform to search the property, request access and interact with the seller and get the land title ownership.
- **Seller:** A person who sells the land and uses the platform to manage properties and transfer land title to buyers.

- **Land inspector:** A person who uses the platform to manage property requests, view reports, confirm and initiate the transfer.

2.2.3. Operating Environment

- Web Services database
- Operating system

  Windows
- Database

  Firebase database
- Platform

  Blockchain System

2.2.4. Design and Implementation Constraints

- There is also a limitation of online network is available to mobile so that is why the application runs properly.
- The application also has the limitation of Constraints are design limitations. An obvious example is a budget. Money affords labor, tools, and to some extent, time
- Limitation for other users they only see the details after login.
- Corporate or regulatory policies

2.2.5. Assumptions and Dependencies

- A Request for the same identification is not given by the system No Same User exist. It generates System Error so the system should avoid this.
- A Request for the same identification is not given by the system no same mobile number exists. It generates System Error so the system should avoid this.
- A Request for the same identification is not given by the system no same CNIC exist. It generates System Error so that system should avoid from this.

2.3. External Interface Requirements

2.3.1. User Interfaces

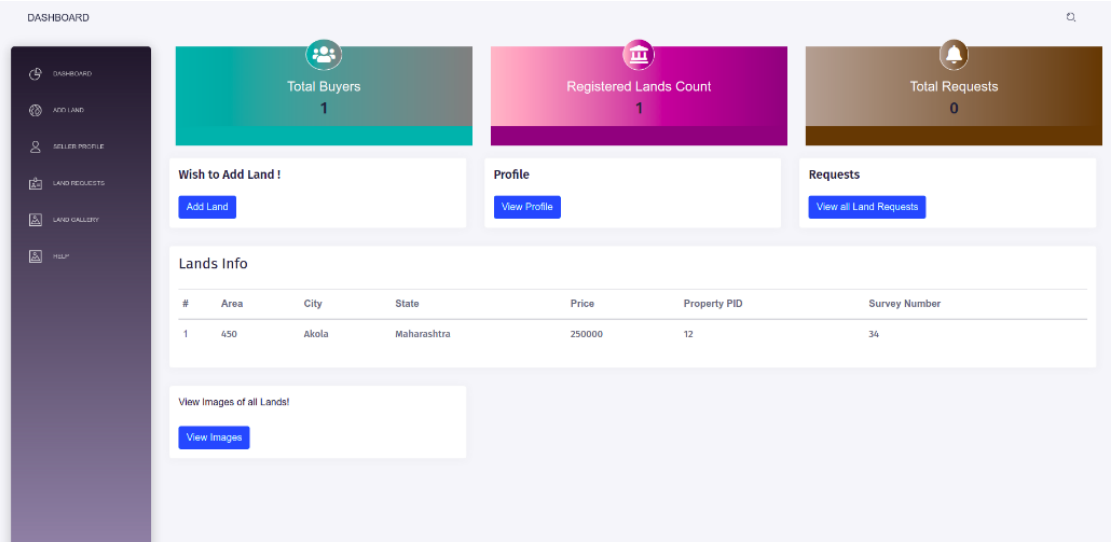


Figure 2: User Interface

2.3.2. Hardware Interfaces

- As this system is online so it only require a web browser and can run with any system specifications.

2.3.3. Software Interfaces

Table 12: Software Interfaces

Software used	Description
Operating system	We have chosen Windows for this platform
Database	To save the data of users and their properties.
XML	Used to create front end like login, Register, Homepage, Map Page. And other pages
Activities	Activity shows the front more effective and speedily working.
fragment	The fragment is used to work and transact between activities.
ReactJS	To implement the project, we use react with JavaScript
Firebase Database	Firebase database used to store images and location and used data
Authentication	Authentication Used for Google, Email, Facebook, etc.

Fire Storage	Fire storage used storing images
Google Map	Google map used to get location and get the actual GPS tracking system
Remote Config	Remote Config is used on an admin side
Notification Manager	To notify the user
Firebase Real-time	Firebase real-time used to store Uri and string, int, float data
Web Services	Web services used in it to add statistics in an application and connect with NGOs

2.3.4. Communications Interfaces

- Communication Standards

  HTTP

  FTP

- Google map uses HTTP protocol and secure protocol.
- API uses HTTP protocol to connect the user.
- Firebase Server communication needs an internet connection on any browser.
- Frontend*:*

  Javascript, React Framework, CSS and Metamask Chrome Extension.

- Backend:

  Ethereum Blockchain (Truffle Suite), Solidity and Ganache.

2.4. System Features

This system provides multiple features such as:

- User registration
- Uploading property pictures
- Buyer Request
- Role of Land Inspector
- Land Registry Document Validation and Authenticity

2.4.1. User Registration

2.4.1.1. Description and Priority

Users who either want to sell or buy properties register to the blockchain land registry platform. They can create the profile on the platform with details like name, government-issued ID proofs and designation. A hash for the identity information submitted by the users gets stored on the blockchain.

2.4.1.2. Functional Requirements

- REQ-SF1-1: <User should enter CNIC, Email and Phone No to verify his identification>
- REQ-SF1-3: <The system will return a hash or ID to the user>
- REQ-SF1-4: <User data should encrypt>
- REQ-SF1-5: <User data should Report any time into system>
- REQ-SF1-6: <User data should access any time the application>
- REQ-SF1-7: <User also check his updates data >
- REQ-SF1-8: <User connect with other user by chat>
- REQ-SF1-9: <User add the Wishlist to the user>

2.4.2. Uploading Property Pictures

2.4.2.1. Description and Priority

Sellers can upload properties' images and documents on the platform and pin the land's location on the map. The transaction corresponding to the seller's action of listing the property details is recorded on the blockchain. Once the property's details are uploaded to the platform, it is made available to all users who have signed up as a buyer.

2.4.2.2. Functional Requirements

- REQ-SF1-1: <Seller can upload pictures>
- REQ-SF1-2: <Seller can pin the land's location on the map>
- REQ-SF1-3: <Add details to the property detail section>
- REQ-SF1-4: <Seller can then make it public where other buyers can view the property>

2.4.3. Buyer Request

2.4.3.1. Description and Priority

A buyer interested in any specific property can send a request to access its specification to the seller. Sellers receive notification for property access requests. They can either deny or accept it by looking at the buyer's profile. Buyers can view the previous ownership records of the property and send a request to purchase it and initiate the transfer. Transactions corresponding to the requests made by both sellers and buyers are recorded on the blockchain to ensure authenticity and traceability.

2.4.3.2. Functional Requirements

- REQ-SF1-1: < buyer interested in any specific property can send a request to access its specification to the seller>
- REQ-SF1-2: <Seller can either deny or accept it>
- REQ-SF1-3: < Buyers can view the previous ownership records of the property and send a request to purchase it >
- REQ-SF1-4: <All the transaction and the data are recorded on the blockchain>

2.4.4. Role of Land Inspector

2.4.4.1. Description and Priority

If the seller approves the land ownership transfer request, the land inspector gets the notification to initiate the transfer of property. Smart contracts trigger to provide land documents' access to the land inspector. After the land inspector verifies the documents, they schedule the meeting for ownership transfer with buyer and seller. The meeting record is also added to the blockchain to solve property related disputes if occur in the future.

2.4.4.2. Functional Requirements

- REQ-SF1-1: <Seller approves the land ownership>
- REQ-SF1-1: <Land Inspector will receive the notification>
- REQ-SF1-1: <Land Inspector then verifies the document>
- REQ-SF1-1: <Then land inspector schedule meeting with buyer and seller>
- REQ-SF1-1: <Land Owner and the Buyer sign that paper>

- REQ-SF1-1: <The paper is then saved to the blockchain system>

2.5. Nonfunctional Requirements

2.5.1. Performance Requirements

- The application should load and be usable within 3 seconds.
- The database should be normalized to prevent redundant data and improve performance.
- The retrieving power of the application should be good.
- The database should be distributed to prevent outages.
- The data saving efficiently in the database.
- Remove duplicate data.
- When moving from one to second activity system performance not disturb.

2.5.2. Safety Requirements

- Databases should use sharing to be redundant to prevent loss of data.
- Backups of the databases should be done hourly and be kept for one week.
- Data should be store in the arrangement of hash and key.
- Key where the data store must be encrypted.
- Images stored in the database also in the backend of hash keys.
- Authentication of a user when he/she tries to log into the system.
- System shutdown in the case of a cyber-attack.
- The verification email is sent to the user whenever he/she registers for the first time on some software system.
- In any draw, the back system should tell
- Authentication used when the email enter by the user is safe.
- Authentication used by Google is safe

2.5.3. Security Requirements

In order to evaluate the security requirement of our application. We evaluate our application with the security point of view. The following are the different security aspects Security has three aspects Confidentiality, Integrity and Availability.

- **Confidentiality:**

  Confidentiality means that the message between the two users through a WEB Socket will be readable to the other user. All the chat between the users will be encrypted stored into the database. A message is encrypted by cryptographic technology and can be easily readable to another user.

- **Integrity:**

  This ensure that the messages chat between the two users will not be edited or change during the transfer of it. We will use hashing function which will map an encrypted message to fixed size length integer. A has function is a one-way function if somebody has an output of has it cannot be reversed. A sender message encrypts and generates a hash out of the encrypted message. The recipient use the same hash function to generate hash out of received encrypted message The recipient compares two hashes if both of the hashes are same then it's an original message.

- **Availability:**

  100% availability 24/7 because our web server is running 24/7

2.5.4. Software Quality Attributes

- Because this application is critical to business communication, we will have a goal of four nines (99.99%) availability.
- The interface should be easy to learn without a tutorial and allow users to accomplish their goals without errors.
- The interface can use by anyone.
- The interface should be converted into Urdu after some time in future

2.5.5. Business Rules

- Application is efficient in use and user should be satisfied.

- Working on data is efficient.
- Working on web services data is sufficient.
- Working on application loading is efficient.
- The application should never allow anyone to read messages or discussions not intended for that person.
- Emails should be sent with a latency of no greater than 12 hours.
- Each request should be processed within 10 seconds.
- The application should load in 3 seconds when the number of simultaneous users is> 10000

2.6. Other Requirements

- The application should use continuous integration so that features and bug fixes can be deployed quickly without downtime.
- Function – function must be reliable for the user.
- Environment – use conditions and profile details.
- Duration – duration must be not enough to boor the user.
- Probability – The likelihood of successfully functioning over the duration.

Chapter 3

Use Case Analysis

Chapter 3: Use Case Analysis

3.1. Use Case Model

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

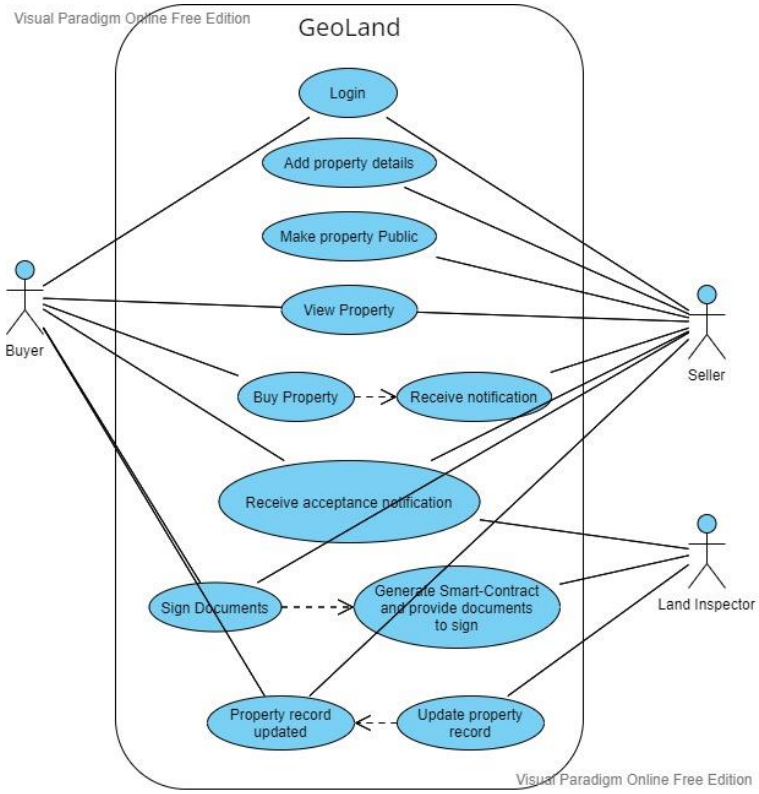


Figure 3: Use Case Analysis

3.2. Use Cases Description

- On Land Registry, buyers can sign up and use the platform to view the land, request access and get the land title ownership.
- To sign up to the application, buyers need to enter details like personal, financial and professional details to complete the verification process.
- Buyers can request access to view the properties details on the mobile.
- Sellers can view the request and grant access to the buyer. Also, the seller can either deny or allow the requests.

- As soon as the seller accepts the request, the buyer can view the property details.
- Moreover, the buyer can also view the ownership history and documents of the property.
- Seller sends the request to the Land inspector to register the property. The land inspector can verify the documents from the blockchain network and add a new land record after complete verification.
- The land inspector uses the land registry platform to add a new land record. They can search for the properties that need to be registered. They can find properties either by entering the details or pinning the location on the map.
- The land inspector can view the ownership details of the properties and initiate the property transfer.
- Buyer and seller both enter their private key to verify the identities. After verification, they digitally sign the agreement from the land inspector's portal.
- Land inspector captures the pictures of the seller and buyer to ensure authenticity. After getting the property document signed by the seller and purchaser, the land inspector can add the legal land record to the system.

Chapter 4

System Design

Chapter 4: System Design

4.1. Architecture Diagram

An architectural diagram is a visual representation that maps out the physical implementation for components of a software system.
Then:

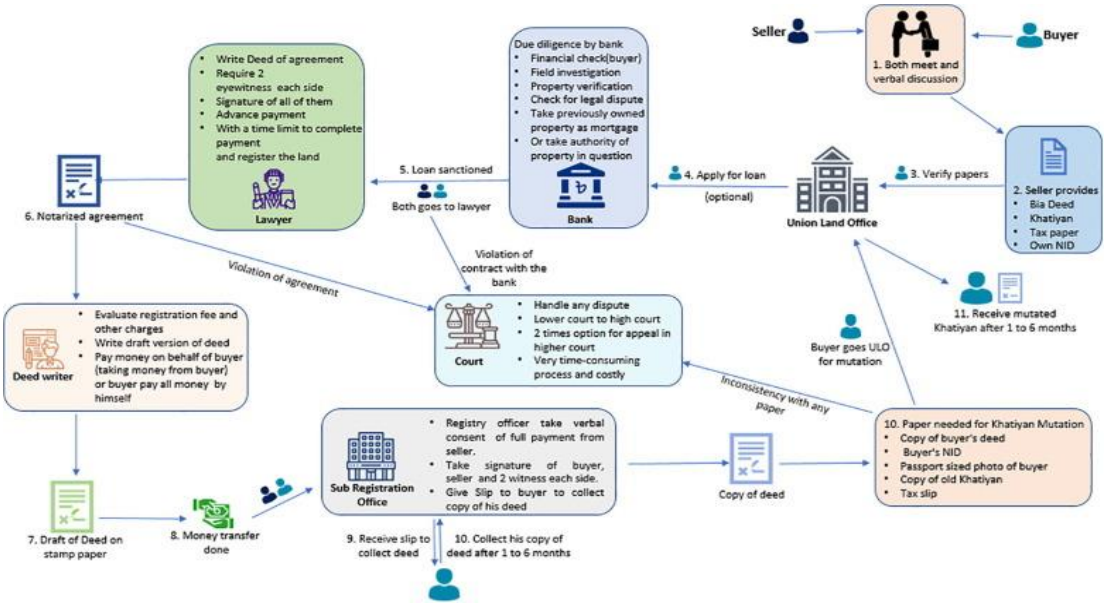


Figure 4: Architecture Diagram

Now:

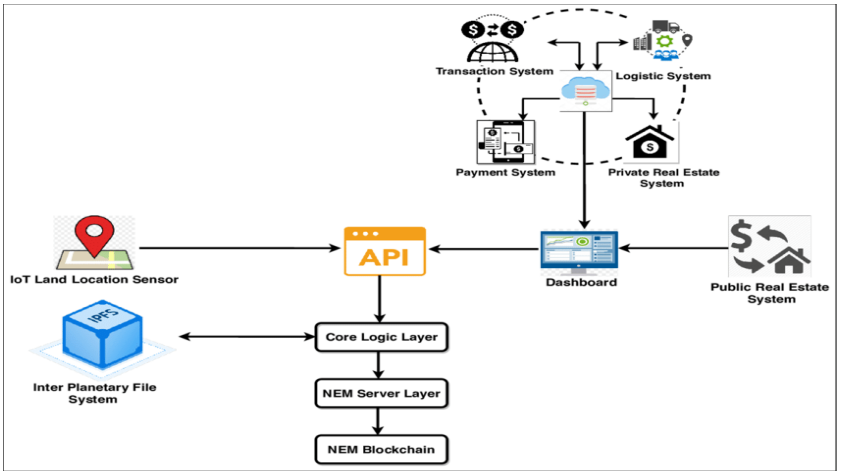


Figure 5: Architecture Diagram (Now)

4.2. Domain Model

In software engineering, a domain model is a conceptual model of the domain that incorporates both behavior and data. In ontology engineering, a domain model is a formal representation of a knowledge domain with concepts, roles, datatypes, individuals, and rules, typically grounded in a description logic.

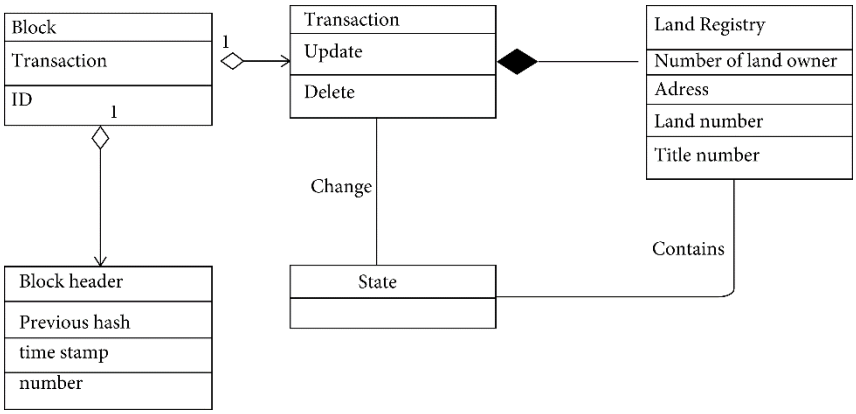


Figure 6: Domain Model

4.3. Entity Relationship Diagram with data dictionary

ER Diagram is a graphical representation of a data model using entities, their attributes and relationships between those entities. It has a form of a diagram.

Data Dictionary is a list of data elements (entity/table and attribute/column) with their attributes and descriptions. It has a form of a set of tables.

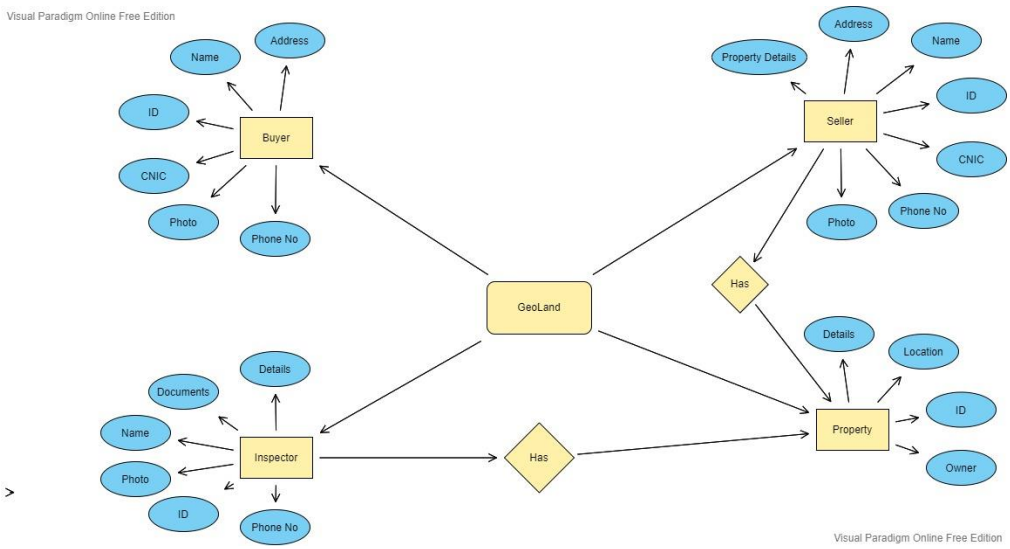


Figure 7: ERD

4.4. Class Diagram

In software engineering, a 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.

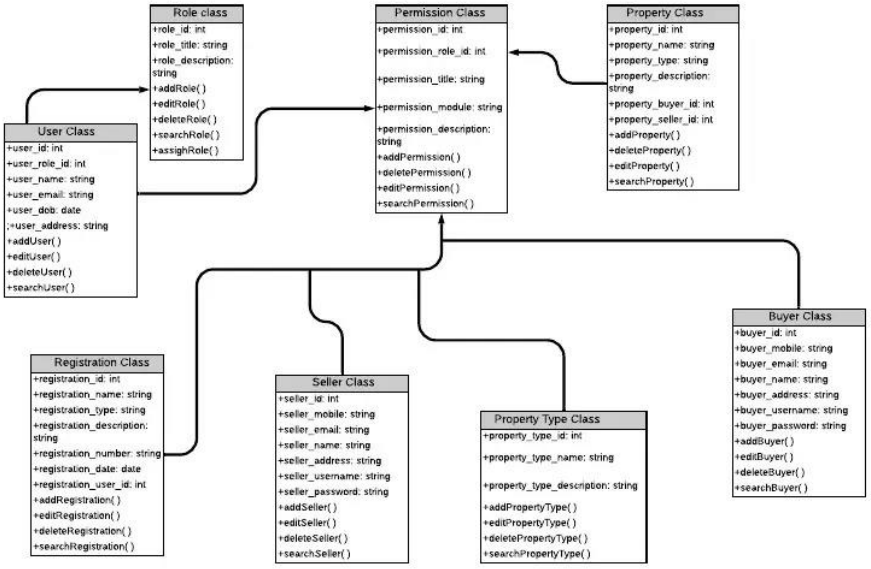


Figure 8: Class Diagram

4.5. Sequence / Collaboration Diagram

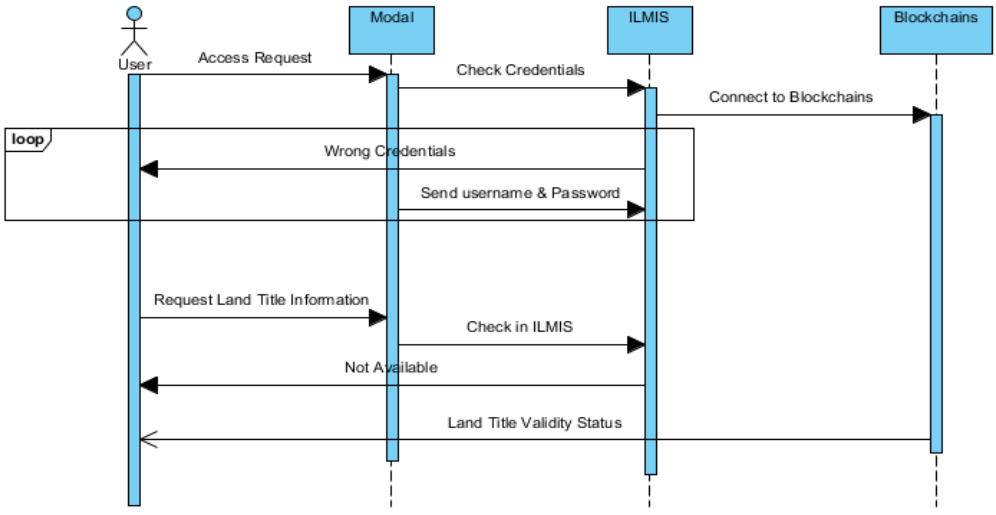


Figure 9: Sequence Diagram

4.6. Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes, as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

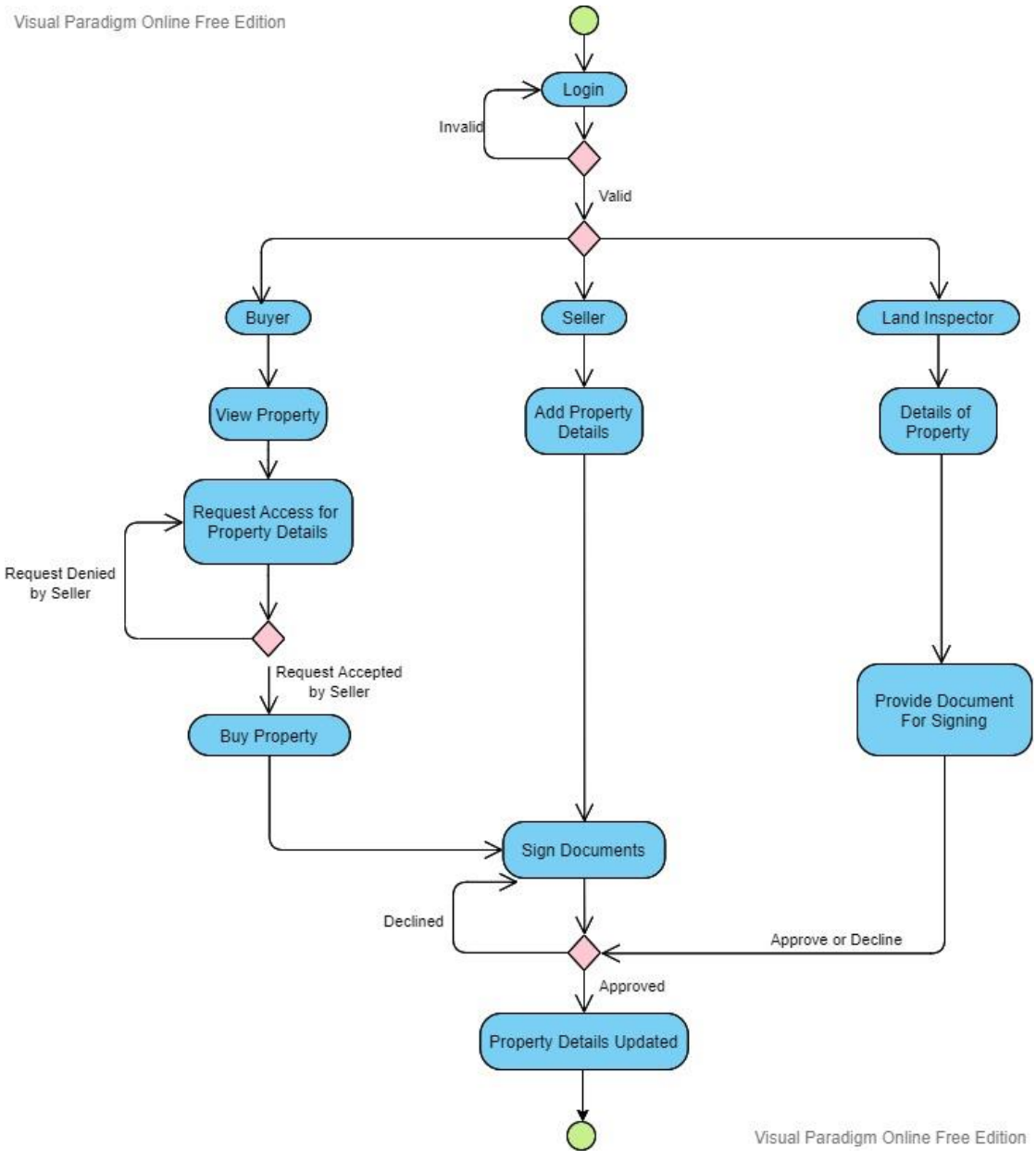


Figure 10: Activity Diagram

4.7. Deployment Diagram

A UML deployment diagram is a diagram that shows the configuration of run time processing nodes and the components that live on them. Deployment diagrams is a kind of structure diagram used in modeling the physical aspects of an object-oriented system. They are often be used to model the static deployment view of a system (topology of the hardware).

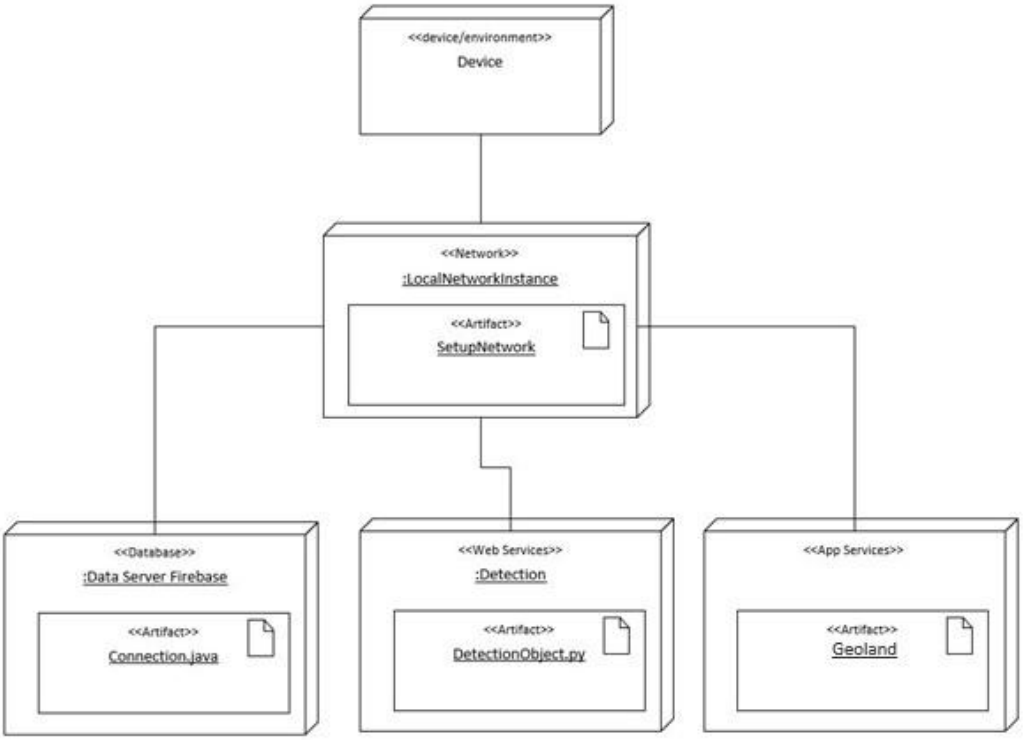


Figure 11: Deployment Diagram

Chapter 5

Implementation

Chapter 5: Implementation

5.1. Important Flow Control/Pseudo codes

- In application, all code is written as per line.
- In application all code is capitalize initial keyword.
- The hierarchy and structures are well defined
- End multi line Structure supported.
- A logical plan is used to programming in this application.
- Stand raised code and easy to understand and easy to develop again
- Security is used in development also.
- Structured should be implementable in this app
- Accessing the data in the standard form.
- Name or fields and variable name is easy to identify.
- Contains no space in the code
- Additional work is with capital form.
- Consistent use of names.

5.2. Components, Libraries, Web Services and stubs

Table 13: Components, Libraries, Web Services and stubs

Firebase Database	Firebase database used to store images and location and used data
Authentication	Using authentication
Fire Storage	Fire storage used storing images
Google Map	Google map used to get location of the property and get the actual GPS tracking system
Notification Manager	To notify the user
Firebase Real-time	Firebase real-time used to store Uri and string, int, float data

5.3. Deployment Environment

- Windows Operating System

5.4. Tools and Techniques

- ReactJS
- Javascript
- Metamask Chrome Extension
- Ethereum Blockchain
- Solidity
- Ganache
- API

5.5. Best Practices / Coding Standards

A coding standard's purpose is to restrict use of problematic areas of the programming language. Therefore, using coding standards prevents undefined or unspecified behavior. In addition, it limits the use of error-prone constructs, such as "goto" And using coding standards improves the code's readability, maintainability, and portability.

- Maintainability
- Dependability
- Efficiency
- Usability
-

5.6. Version Control

- Google map use http protocol and secure protocol.
- API use http protocol to connect the user.
- Firebase Server communication needs internet connection on any browser.
- GitHub version control

Chapter 6
Testing and Evaluation

Testing Evaluation

Testing & Evaluation is the system analysis in which system components are compared to requirements and specification through testing. We used this process to test our system components. The results are evaluated to assess the progress of design, performance, supportability etc. By this approach we discover defects/ Bugs in the system.

6.1 Testing Use Case

Test Case 1: Login

Table 14: Login

Test Case ID	C001
Test Case Name	Login
Test Case Description	To verify the user logged in with valid credentials or not
Purpose	The purpose to this test case is to verify that the user easily logged in into the system or not
Prerequisites	Must have a Wi-Fi connection. Must have a valid credentials for logged in
Test Procedure	Remotely
Expected Result	Users can log in into the system with valid credentials
Actual Result	Users can enter into the system with valid credentials
Status	Pass.

Test Case 2: Dashboard

Table 15: Dashboard

Test Case ID	C002
Test Case Name	Dashboard
Test Case Description	To verify the user logged in application to see dashboard
Purpose	The purpose to this test case is to verify that the user sees the dashboard or not.

Prerequisites	Must have a Wi-Fi connection. Must have a Valid credentials for logged in
Test Procedure	Remotely
Expected Result	Users can log in into the system
Actual Result	Users enter into the system
Status	Pass.

Test Case 3: Check Location

Table 16: Check Location

Test Case ID	C003
Test Case Name	Check Location
Test Case Description	To verify the user are able to check the accurate location of lands
Purpose	The purpose to this test case is to verify that the user can checks the location of lands for his ease of use.
Prerequisites	Must have Login in the System. Must have click on the get location button
Test Procedure	Remotely
Expected Result	Users can easily check the location.
Actual Result	Users get the accurate location
Status	Pass.

Test Case 4: Check Records

Table 17: Check Records

Test Case ID	C004
Test Case Name	Check Records
Test Case Description	To verify the users can checks the records.
Purpose	The purpose to this test case is to verify that the user can checks and get the accurate details of lands

Prerequisites	Must have a Wi-Fi connection. Must logged in into the system.
Test Procedure	Remotely
Expected Result	Users can easily see the details on the screen
Actual Result	Users check the history
Status	Pass.

Test Case 5: Update Data

Table 18: Update Data

Test Case ID	C005
Test Case Name	Update Data
Test Case Description	To verify that the seller can update data with correct/new information
Purpose	The purpose to this test case is to verify that the seller can update the data into the records by adding correct/new information
Prerequisites	Must have a Wi-Fi connection. Must logged in into the system Must have an idea which information wants to update
Test Procedure	Remotely
Expected Result	Data Updated successfully
Actual Result	Data Updated successfully
Status	Pass.

Test Case 6: Check Land Paper

Table 19: Check Land Papers

Test Case ID	C006
Test Case Name	Check Land Papers
Test Case Description	To verify that the user gets Land Papers or no
Purpose	The purpose to this test case is to verify that the seller and buyer both have received the papers

Prerequisites	Must have a Wi-Fi connection. Must logged in into the system Must have bought any land
Test Procedure	Remotely
Expected Result	Date Checks Successfully
Actual Result	Date Checks Successfully
Status	Passed

Test Case 7: Logout

Table 20: Logout

Test Case ID	C007
Test Case Name	Logout
Test Case Description	To verify that the user session is saved after clicking on logout
Purpose	The purpose to this test case is to verify the session are saved when user clicked on logout button
Prerequisites	Must have a Wi-Fi connection. Must logged in into the system Must Clicked on Logout Button
Test Procedure	Remotely
Expected Result	User Logout
Actual Result	Users Logout
Status	Passed

Test Case 8: Registration

Table 21: Registration

Test Case ID	C009
Test Case Name	Registration
Test Case Description	To verify that user be able to register into the system.
Purpose	The purpose to this test case is to verify the registration verified when user clicked on register button.

Prerequisites	Must have a Wi-Fi connection. Must have valid data that accept the system. Must be registered in system
Test Procedure	Remotely
Expected Result	Registered in System
Actual Result	Registered in System
Status	Passed

6.2 Performance Testing

Performance testing is the type of testing to test the system speed, throughput, responsiveness, and workload. Stability, Reliability, Scalability and other resources that are used in the system. The performance of the software is determined by these factors

- Speed
- Scalability
- Stability

This testing tool test our web application performance. It checks the speed that how much loading speed when web application runs on the browser. It also checks that the application is able to handle the work load on the system and application is stable and error free or not. The record system is working fine and its performance according to the expectation and all the modules are working properly. We tested the Volume testing to test the database.

6.3 Unit Testing

This testing is to test the individual components of the application. We divided all the components into the modules and test the modules and testing each module.

6.4 Equivalence Partitioning

Using this Equivalence Partitioning method test cases are divided into the three categories or set of input data is classes. Each test cases represent the class so we divide our test case into the three equivalence classes that are used to test the test case with valid and invalid inputs.

Login:

User can be logged by enter valid username and key/password to enter into the system. When user entered the username and key/password system checks that the entered data is already exist in database or not. If it exists it redirect to the dashboard and if it's not existed in the database the system saves the data into the database and redirect to dashboard.

6.5 Boundary Value Analysis

Boundary Value analysis is to analyze and measure the boundary value in the range by the system. The system set the value range for the system.

- Only Authorize person can logged in into the system
- 0 – 100 users can enter into the system

6.6 Stress Testing

Stress testing is the type of testing which test how system handles the extreme workload and high traffic or data processing. We used this testing to check that our system handles lots of incoming messages from the server or not. We used Meter tool for the stress testing of our application. This testing mainly determines the system on its robustness and handle extremely heavy load conditions. This ensure that our application works fine under the incredibly hefty burden conditions

6.7 Data Flow Testing

Data flow testing is used to analyze the flow of data into the system. It's the process to analyze that the how-to data flow into the system. Data Flow testing centers focuses on what factors get values and the points at these values are used in which we characterize numerous flows of our application additionally depict how the system flowing in our system.

User 1 Data Flow

The user first turns on his/her Wi-Fi to use the application to check data with someone. Once application is opened user enters the valid username and key/password into the fields. When user entered the username and key/password system checks that the entered data is already exist in database or not. If it exists it redirect to the dashboard and if it's not existed in the database the system saves the data into the database and redirect to dashboard.

Chapter 7

Summary, Conclusion & Future Enhancements

7.1 Project Summary

This document is the report about our FYP project; it will give us a detailed view of the data flowand the roles and responsibilities of the actors and the system. There are seven Chapters in this project and the content in them is mentioned below Table.

Table 22: Project Summary

#	*Chapters*	Document Usage
1.	***Introduction***	This section covers the • Summary and background of this project. • Project scope, goals, and objectives are also mentioned • Project and work breakdown structure
2.	***Software Requirement Specifications***	This section covers the • Description of the product • Functional and non-functional requirements • Product interfaces and business rules
3.	***Use Case analysis***	This section covers • Use cases that are intended for developers, testers, and other technical readers *"Use case is a list of actions or event steps typicallydefining the interactions between a role and a system to achieve a goal"*
4.	***System Design***	• This Section is solely intended for the developersand the project managers • Detailed diagrams are the part of this section describing the system architect and the data flow
5.	***Implementation***	This section describes • The tools and techniques used for the development • Development environment components and best practices are also mentioned.
6.	***Testing and Evaluation***	This section describes • This section describes the types of testing through which the System has passed
7.	***Summary, Conclus***	This section describes • The overall summary of the chapters • The conclusions made from this Project

	ion,and Future Enhancements	The future enhancements that are to be made in it
8.	***Appendix***	This section includes any additional or supplementary information on the book's topic

7.2 Achievements and Improvements

By working on this project, we got hands-on practical experience on blockchain and web development knowledge that we gained during our 4 years of degree this project cleared our basic to a high level of concepts or any confusion that we had. We now have a thorough understanding of how does the blockchain and databse work. By doing this project we not only technologically but as a team learned many things. We learned to do teamwork and how to manage and divide work and save time by doing so. Teamwork is the collaborative effort of a group to achieve a common goal or to complete a task most effectively and efficiently. This concept is seen within the greater framework of a team, which is a group of interdependent individuals who work together towards a common goal. Contribute to Open-Source Projects, Teach Others What You Know, Challenge Yourself with a New Skill, Schedule Regular Downtime.

Our Improvement Involves increase in technical abilities these skills include working withmodern-day technologies like

- Frontend & Backend **skills**.
- Responsive design **skills**.
- Blockchain **skills**.
- Testing and debugging **skills**.

Amplifying the need for good communication skills is the fact that coding is something that is very hard to grasp for non-developers. You must be able to make other people understand technical problems, which is something that

can be tough. As a way of improving Communication skills, a great idea is to have regular talks with people you know outside of the office and try to make them understand what you are working on.

7.3 Critical Review

Blockchain technology is a peer-to-peer protocol that can be leveraged to keep track of transactions over the internet. Publicized for its use in the bitcoin revolution, the technology provides transparency and traceability that can be used in the management of land rights. When it comes to the formalization of land rights, blockchain technology promises to authenticate owners and other users of land, and provides a fixed ledger of land use rights transactions. At present, blockchain technology is being explored as a proof of concept in several countries to track land titles.

7.4 Lessons Learnt

This section describes the lesson learned in the process of making this software working which is mentioned below

7.4.1 Applying Knowledge

The project provided us the opportunity to apply the knowledge that we learned during our 4 years of the degree, and to self-explore and learn new technologies that would prepare us for over future.

7.4.2 Collaboration

It’s incredibly important to have close collaboration with the full project team at the very early stages of the project before the budget is set. Do not make key decisions without your project team already on board.

Collaborating early in the project will save you both time and heartache.

7.4.3 Be very patient

Having trust in each other and on Allah being patient is one of the most important things that we learned from this project do not give up, keep working hard you will achieve yourgoal.

7.5 Future Enhancements/Recommendations

This software was a very basic one, saving multiple issue and time, and demonstrates the idea we had. For better and more accurate results, preferable Recognize reorganization system that a user giving instruction through view to save time and life. We can conclude thatthis project of ours has a vast scope in future application.

Reference and Bibliography

Reference and Bibliography

- Blockchain Website: https://www.blockchain.com/
- Investopedia: What is blockchain and why we use it. https://www.investopedia.com/terms/b/blockchain.asp
- Punjab Land Authority: https://www.punjab-zameen.gov.pk/
- Land Record Information Management System: https://proceedings.esri.com/library/userconf/proc05/papers/pap1279.pdf
- Blockchain in Land record System: https://iopscience.iop.org/article/10.1088/1755-1315/614/1/012055/pdf#:~:text=Blockchain%20in%20the%20land%20registry,buyer%20through%20an%20application%20form