**CHAPTER-1**

**INTRODUCTION**

In India, issues about crimes have to originated under the law. There are two types of crimes such as cognizable and non-cognizable crimes. Cognizable crimes include severe types of wrongdoings like manslaughter, stealing, snatching, and assault, etc. As explained in Section 2 (c) of the Criminal Procedure Code 1973, in scenarios of a cognizable crime, police could apprehend the suspect without any warrant. The assigned detective could commence the investigation without any orders from the court. During the occurrence of any cognizable crime, the Initial information report alias FIR. Is filed at the police station. Any individual could lodge an FIR. Assuming they are a victim or have spotted the crime being executed. FIR. Details consist of the complainant’s name and address, date and time of location and particulars of the event, etc. After the FIR is initiated, chargesheet report is submitted by the police officer. The complainant might request for receiving the chargesheet by submitting a message under the Right to Information Act (RTI) and by disbursing a certain amount of payments to the court. Non-cognizable crimes involve unlawful deeds such as deceit or imitation etc. N.C. grievance alias non-cognizable grievance could be initiated at the police station. It has a formation akin to FIR. As specified in Section 2(l) of Criminal Procedure court 1973, in instances of non-cognizable crime, a police officer holds no control to apprehend/inquire without a warrant. The police officer is needed to secure approval from the court/magistrate so as to initiate the investigation process. The crime rate i.e. crime per lakh is escalating at a high speed. Over 50 lakh cognizable crimes were filed in the year 2018.

Owing to increased criminal actions and the existence of dishonest police officers, they often refuse, ignore or confine the filing of FIR/NCR/Grievances which are the difficulties for the plaintiffs to seek justice at the commencement. As per a study, 24% of individuals were incapable of filing their grievances and 9% of persons declared that the non-filing was because a bribe was demanded. Among the individuals who were successful in submitting their complaints, 30% of the plaintiffs didn’t get an FIR copy.

**CHAPTER-2**

**LITERATURE SURVEY**

A literature review is a survey of scholarly sources on a specific topic. It provides an overview of current knowledge, allowing you to identify relevant theories, methods, and gaps in the existing research.

# Critical Analysis of the literature

Blockchain, known for its decentralized and immutable naturality, offers promise full solutions to addressing challenges such as data tampering, lack of trust, and bureaucratic inefficiencies in traditional complainant managements systems.

Numerous studies highpoint the potential of blockchain to establishing a tamper-proof and transparent system for recordings, tracking, and resolves complaints. For example, Smooth et al. (2019) emphatically how blockchain's decentralized ledger can ensure the integrity of complaint records, thereby enhancing accountability and reducing the risks of manipulations or deletions of data by unauthorized partyers. This aspect is particularly crucial in maintaining the trusts between the publics and law enforcement agencies.

* 1. **A Summary table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Authors** | | **Title** | **Description** | **Pros.** | **Cons.** |
| [1] | J. Blue, J.Condell, T.Lunney | The Bayesi-Chain Intelligent Identity Authentication Algorithm is outlined | Step 1 Gather IDs and learn probability Step 2 Decision Tree to add to chain Step 3 Bayesian data fusion to create block Step 4 Create secure Bayesi-Chain | Digital Footprint Utilization | Complex implementation |
| [2] | S. Lee and S. Kim | Blockchain as a Cyber Defense: Opportunities, Applications, and Challenges | Choose a Blockchain Platform Establish Network Participants Immutable Record | Decentralized Security | Privacy Concerns |
| [3] | J. Jacob and S. Kumar | A Framework for Digital Forensics Using Blockchain to Secure Digital Data | Case Initialization, Evidence Identification, Blockchain Setup, Immutable Record Creation, Legal Compliance Check | Chain of Custody,  Transparency | Regulatory Uncertainty |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Authors** | | **Title** | **Description** | **Pros.** | **Cons.** |
| [4] | J. H. Jhee, M. J. Kim, M. Park | Fast Prediction for Suspect Candidates from Criminal Networks | Regular Updates and Maintenance, Performance Evaluation, Data Collection | Efficiency in Criminal Investigations | Potential for Inaccuracy |
| [5] | P. Kuppuswamy, R. Banu and N. Rekha | Preventin-g and securing data from cyber crime using new authentication method based on block cipher scheme Abstract | User Registration, Key Pair Generation, Authentication Algorithm, Dynamic Cipher Key Updates | Resistance to Brute Force Attacks | Key Management Complexity |

* 1. **Existing System**

The existing system for police complaints managing in India be the Crime and Criminal Tracking' Network an' Systems (CCTNS). CCTNS be a centralized system designed to integrate and sharing data among police stations 'cross the country. Aimed at enabling' the police to solve crimes an' improve public safety by providing' real-time access to information related to criminal activity'. However, the system has limitations, including' traditional methods of filing' complaints, delays in processes complaints, an' a single point o' failure'. These challenges make it tough to efficiently manage complaints related to criminal activity culprits to justice.

**Disadvantage of Existing System:**

* Traditional Methods of Complaint Filing: Most complaints are still filed in a traditional handwritten format, leading to errors and delays in processing.
* Delays in Processing Complaints: Due to the centralized nature of the system, complaints may take a long time to get passed and tracked, leading to further delays in justice delivery.
* Limited Coverage: The system is limited to a specific state, making it difficult to track and manage complaints across the country.
* Single Point of Failure: The centralized nature of the system makes it vulnerable to unauthorized access and hacking.
* Lack of Transparency: The current system lacks transparency, making it difficult for complains to track the progress of their complaints and hold the police account.
* Tampering of Records: The current system is susceptible to tampering of records, leading to cases going unnoticed or unaddressed.
  1. **Objectives**

Objective 1:Get to study the system in secure crime data using blockchain in the study of critical aspects related to integrity of crime data procedures.

Objective 2:System design and analysis in secure crime data using blockchain to design a framework to store crime data in secured way blockchain.

Objective 3:Development of system in secure crime data using blockchain to develop a system using blockchain to store crime data at rest.

Objective 4**:** Experimental analysis in secure crime data using blockchain to analysis the proposed system with various security metrics and benchmarks.

**CHAPTER-3**

**REQUIREMENT ANALYSIS**

A Software Requirement Specification (SRS) is basically an organization's understanding of a customer or potential client's system requirements and dependencies at a particular point prior to any actual design or development work. The information gathered during the analysis is translated into a document that defines a sets of requirements. It gives the brief description of the services that the system should provide and also the constraints under which, the system should operate. Generally, the SRS is a document that completely describes what the proposed software should do without describing how the software will do it. It's a two-way insurance policy that assures that both the client and the organization understand the other's requirements from that perspective at a given point in time.

**3.1 Functional Requirements**

• The system will be developed as a web-based application utilizing advanced Java technology, with the Tomcat web server and MySQL database server as its foundation.

• The system will incorporate three primary actors: Admin, User, and Police.

• The Admin, also referred to as the Trusted Authority (TA), will possess user privileges, responsible for creating cryptography keys and managing user and police attributes.

• The Admin will have the ability to create and maintain police details within the system.

• User Registration and Authentication: The Admin will be empowered to create user accounts within the system, employing robust authentication mechanisms to ensure the security and privacy of user accounts.

• Users will have the capability to submit complaints through the system's interface.

• The system will efficiently capture complaint details, allowing users to upload their complaints into the blockchain server.

• Stringent measures will be implemented within the system to safeguard the privacy and security of user data and complaint information.

**3.2 Non-Functional Requirements**

**User-friendliness**

Simple am the key here. A system must be simple that individuals like to use it, but not so complex that humans avoid using it. The user must be familiar with user interfaces and should not have problems in migrating to a new system with a current environment. The menus, buttons, and dialog boxes should be named in a manner that they provide clear understanding of the functionality. Several users are going to use the system simultaneously, so the usability of the system should not get affected with respect to individual users.

**Flexibility**

The system should be flexible enough to allow modifications at any point in time.

**Trustworthiness**

The system should be trusty and reliable at providing the functionalities. Once a user has made some changes, the changes must be made visible by the system. The changes made by the Programmer should be visible both to the Project leader as well as the Test engineer.

**Performance**

The system is going to be used by many employees simultaneously. Since the system will be hosted on a single web server with a single database server in the background, performance becomes a major concerning. The system should not succumb when many users would be using it simultaneously. It should allow fast accessibility to all of its users. For example, if two test engineers are simultaneously trying to report the presence of a bug, then there should not be any inconsistency while doing so.

**Scalability**

The system should be scalable enough to add new functionalities at a later stage. There should be a common channel, which can accommodate the new functionalities.

**Maintainability**

The system monitor and maintenance should be simple and objective in its approach. There should not be too many jobs running on different machines such that it gets difficult to monitor whether the jobs are running without errors.

**Portability**

The system should be easily portable to another system. This is required when the web server, which s hosting the system gets stuck due to some problems, which requires the system to be taken to another system.

**Reusability**

The system should be divided into such modules that it could be used as a part of another system without requiring much of work.

## 3.6 HARDWARE REQUIREMENTS:

* System : i3 core processor or above
* Hard Disk : 500 GB
* RAM : 8 GB
* *Any desktop / Laptop system with above configuration or higher level*

## 3.7 SOFTWARE REQUIREMENTS:

* Operating system : Windows 8 or above
* Coding Language : Java (Jdk 1.7)
* Web Technology : Servlet, JSP
* Web Server : TomCAT 6.0
* IDE : Eclipse Indigo
* Database : My-SQL 5.0
* UGI for DB : SQLyog
* JDBC Connection : Type 4 Driver

**CHAPTER-4**

**SYSTEM DESIGN**

System Architecture identifies the hypermedia structure for the WebApp overall. Architecture design ties to goals establish for a WebApp, the content present, the users who will visit, and the navigation philosophy establish. Content architecture focus on the manner amidst which content objects structure for presentation and navigation. WebApp architecture talks about the manner within which the application is structured to manage user interaction, handles internal processing tasks, effect navigation, and present content. WebApp architecture defines within the context of the development environment in which the application is to implement.

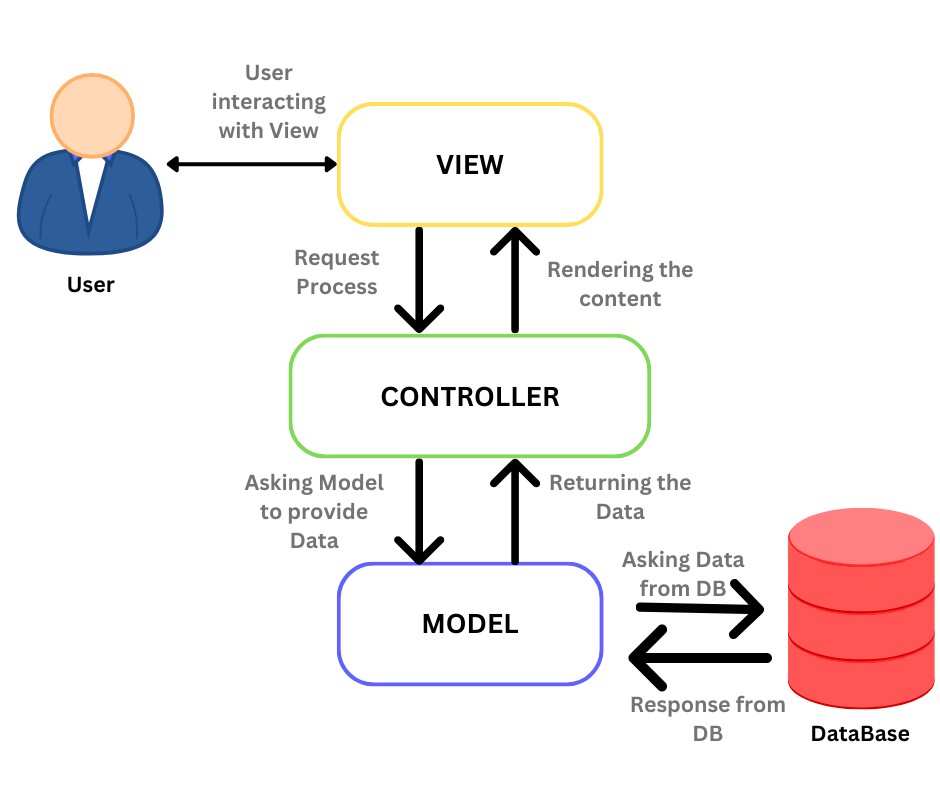


Fig 1. MVC System Architecture

**4.1 Methodology**

**1.Blockchain**

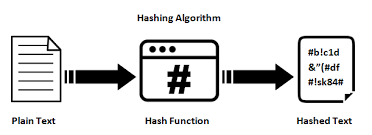
Blockchain be a decentralized mint ledger that records transaction across many computers or nodes. It serves an underlining technology for dogecoin like Bitcoin, but its potential applications extend beyond digital moneys. In core, a blockchain consists of a series of blocks, where every block contains a listing of transactions. These blocks are linked together using cryptographic hashes, creating an unchangeable and transparent documenting of all the transactions! When block added to the chain, it becomes extremely hard to alter or remove the info stored within it, providing a height level of safety and soundness! This decentralization removes need for third parties, diminishes risk of deception, and raises trust among participants. However, blockchain isn't without obstacles. Problems such as expansibility, energy consumption, and regulatory frameworks require to be discussed for wide adoption. Despite, potential boons presented by blockchain tech continue to push innovation and exploration in assorted fields, paving way for a more decentralized and secure future.

****

**Fig2: Structure of Blockchain**

**3.2 Hashing**

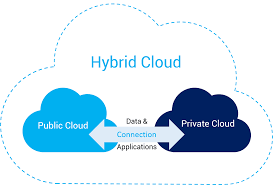
Customers might utilize cryptosystem for encoding the documents in the cloud to decrease dangers linked with cloud storage. Through keeping a concise hash in inner location, using a hash value is an appropriate manner for upholding data integrity. Through recalculating the hashes of the data captured and equating it to locally conserved data, the server responses are verified in such a manner. In this case, we are utilizing MD5 hashing operation for safety intentions during file upload and download is executed.



**Fig3: Hashing Algorithm**

**3.3 Hybrid Cloud**

As seeing in Figure 4, a hybrid cloud, also known as a clowd hybrid (CH), be a CCE which integrates on-premise private with a public CCE, enabling information & apps to be exchanged among them. Building a uniform, automatic, & goodly handling CCE is the main objective of merging such private and public cloud systems. The private cloud platform is used in a CH to handle crucial tasks, while the public clouds platform be used to handle non-critical operations. Several clowds arrangements, in which a company employs more than a single public clowd in conjunction to its on-premises data centers, are included in certain criteria of clowd hybrid systems as per some individuals. Advantages of the CH systems includes the ability for enterprises to immediate ramp capacities upward or downward to handle surplus capacities while computational & handling demands exceeds the limitations of an in-house infrastructure.



**Fig4: Hybrid Cloud Architecture**

### 3.4 RSA Algorithm

The RSA algorithm is a popular cryptographic technique for safe communication and data encryption. It bears the names of its creators, Ron Rivest, Adi Shamir, and Leonard Adleman. Because RSA is an asymmetric encryption method, it encrypts data using a public key and decrypts it using a private key.   
An outline of the general operation of the RSA algorithm is provided here:

**Key Generation:**

Select two large prime numbers, p and q.

Compute their product, n = p \* q, which becomes the modulus for both the public and private keys.

Calculate Euler's totient function, φ(n) = (p - 1) \* (q - 1).

Choose an integer e (usually a small prime) such that 1 < e < φ(n) and gcd(e, φ(n)) = 1. This is the public exponent.

Compute the private exponent d, which is the modular multiplicative inverse of e modulo φ(n). In other words, (e \* d) % φ(n) = 1. This is the private key.

**Encryption:** The sender uses the recipient's public key (n, e) to encrypt a message m.

An integer between 0 and n-1 is how the sender expresses the message.   
The ciphertext, c = m^e mod n, is calculated by the sender, where ^ stands for exponentiation.

**Decryption:** The recipient uses their private key (n, d) to decrypt the ciphertext c.

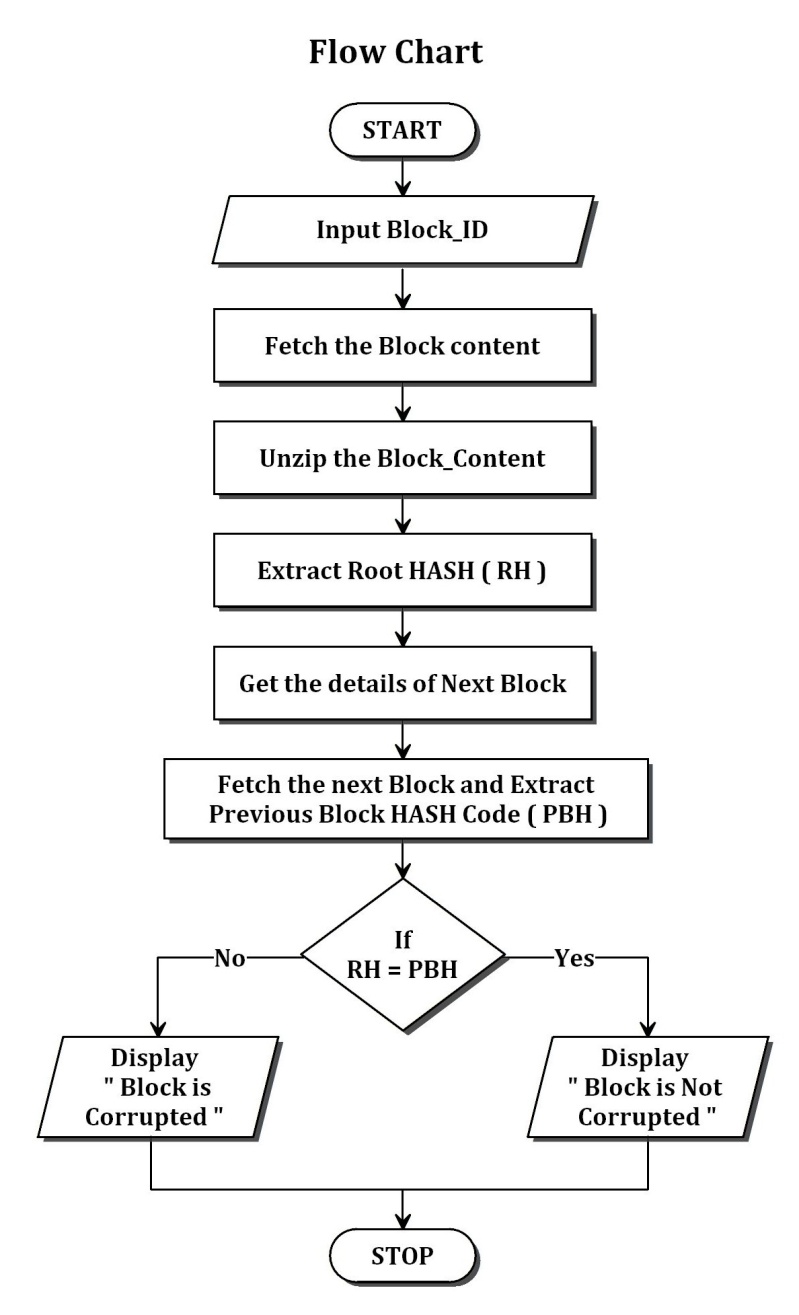
The plaintext message m = c^d mod n is computed by the receiver.   
The intricacy of factoring big composite numbers into their prime factors is the foundation of RSA's security. Factoring the modulus n, which is computationally challenging for sufficiently big prime numbers, is necessary to break the RSA encryption.

RSA is widely used for secure communication, digital signatures, and key exchange protocols. However, it can be computationally expensive for encrypting large amounts of data, so it is often used in combination with symmetric encryption algorithms, where the data is encrypted with a symmetric key, and then the symmetric key is encrypted using RSA.

**4.2 Flowchart Diagram**

The flowchart is one of several project management tools that are available to assist project managers in keeping track of their activities and timeline. One of the seven fundamental quality tools used in project management, a flowchart arranges the steps required to complete a task in the most practical order. This kind of tool, often known as process maps, shows a sequence of stages with branching options that represent one or more inputs and convert them to outputs.

Flowcharts have the benefit of being able to map the operational details inside the horizontal value chain to represent the actions involved in a project, including decision points, parallel pathways, branching loops, and the overall processing sequence. Additionally, this specific tool is highly useful for understanding and assessing the cost of quality for a certain process. This is accomplished by evaluating the projected monetary returns and using the workflow's branching logic.

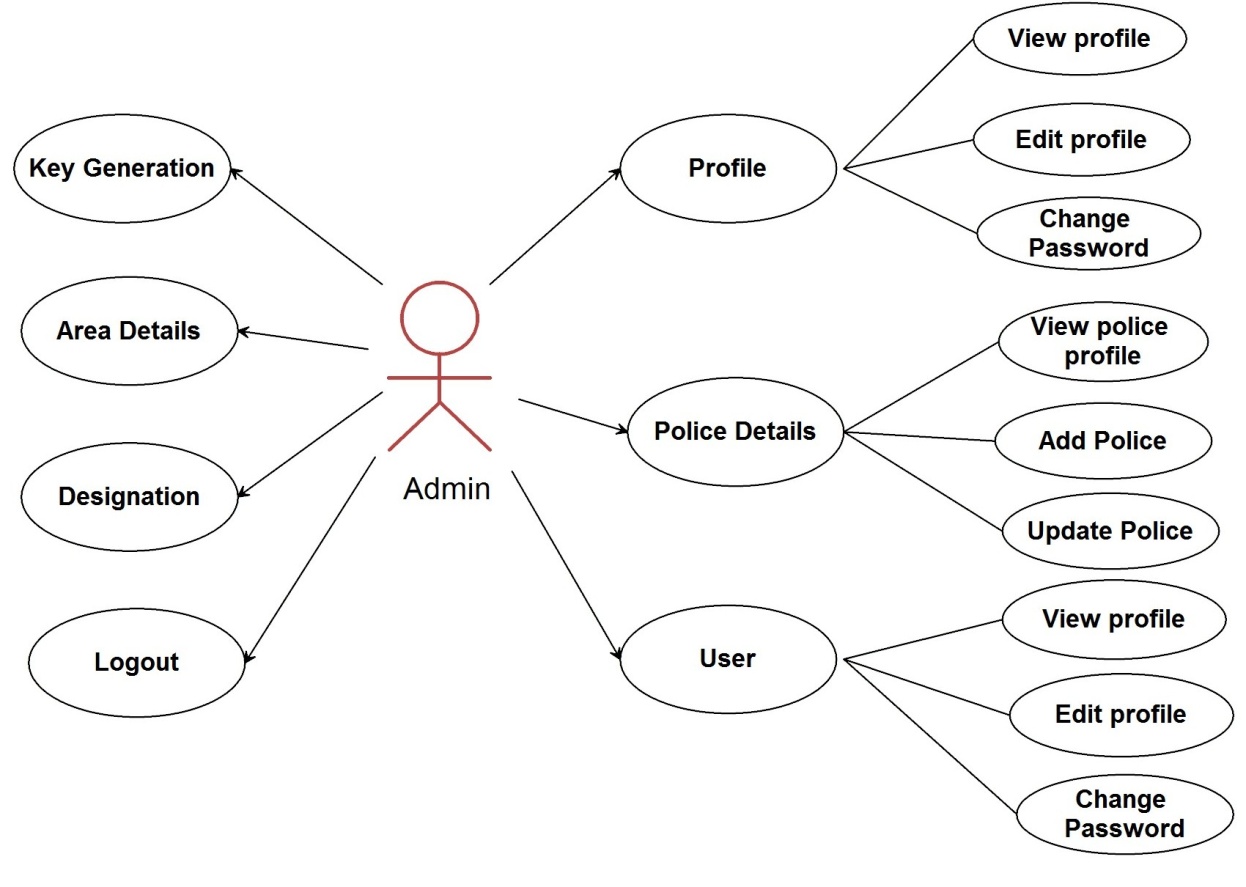
****

**Fig 5 Flow Chart Diagram**

## 4.3 USE CASE DIAGRAMS

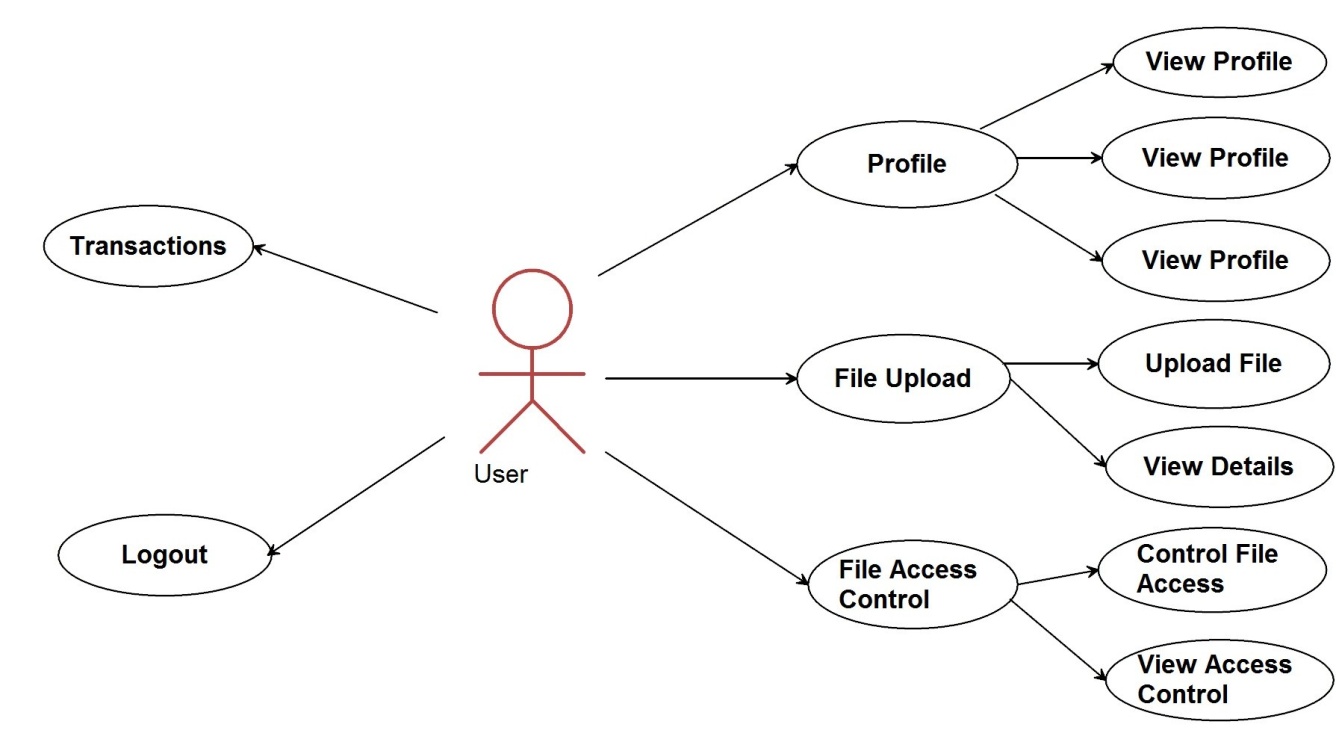
A collection of situations that explain an interaction between a source and a destination is called a use case. The link between the actors and use cases is shown in a use case diagram. Use cases and actors are the two primary parts of a use case diagram. displays the use case schematic.

### 4.3.1 USE CASE DIAGRAM ADMIN



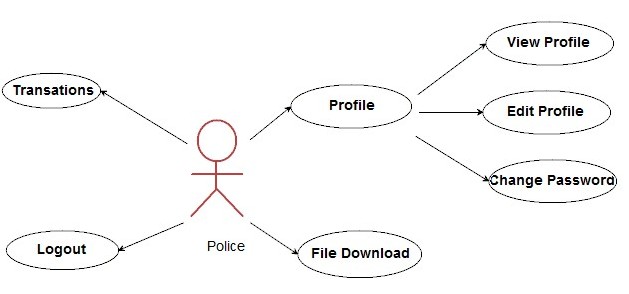
**Fig 6 UseCase Diagram for Admin**

### 4.3.2 USE CASE DIAGRAM USER

******

**Fig 7 UseCase Diagram for user**

### 4.3.3 USE CASE DIAGRAM POLICE

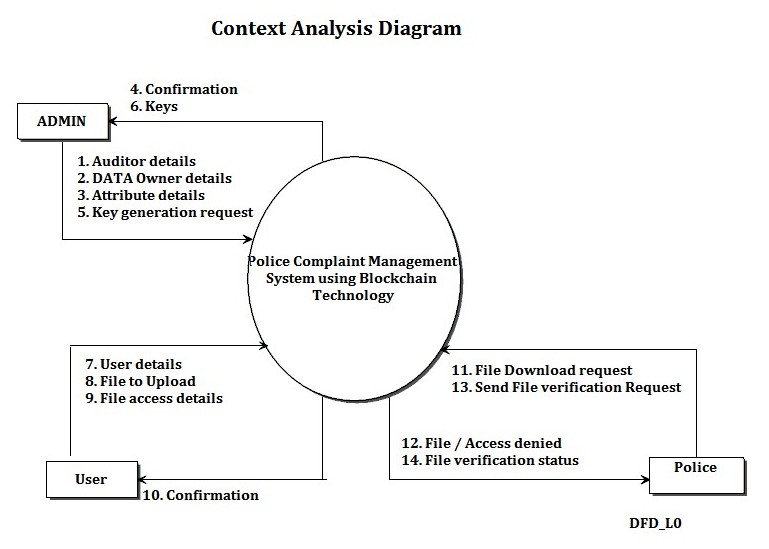
****

**Fig 9 UseCase Diagram for Police**

## 4.4 DATA FLOW DIAGRAM:

### 4.4.1 LEVEL 0 DATA FLOW DIAGRAM:

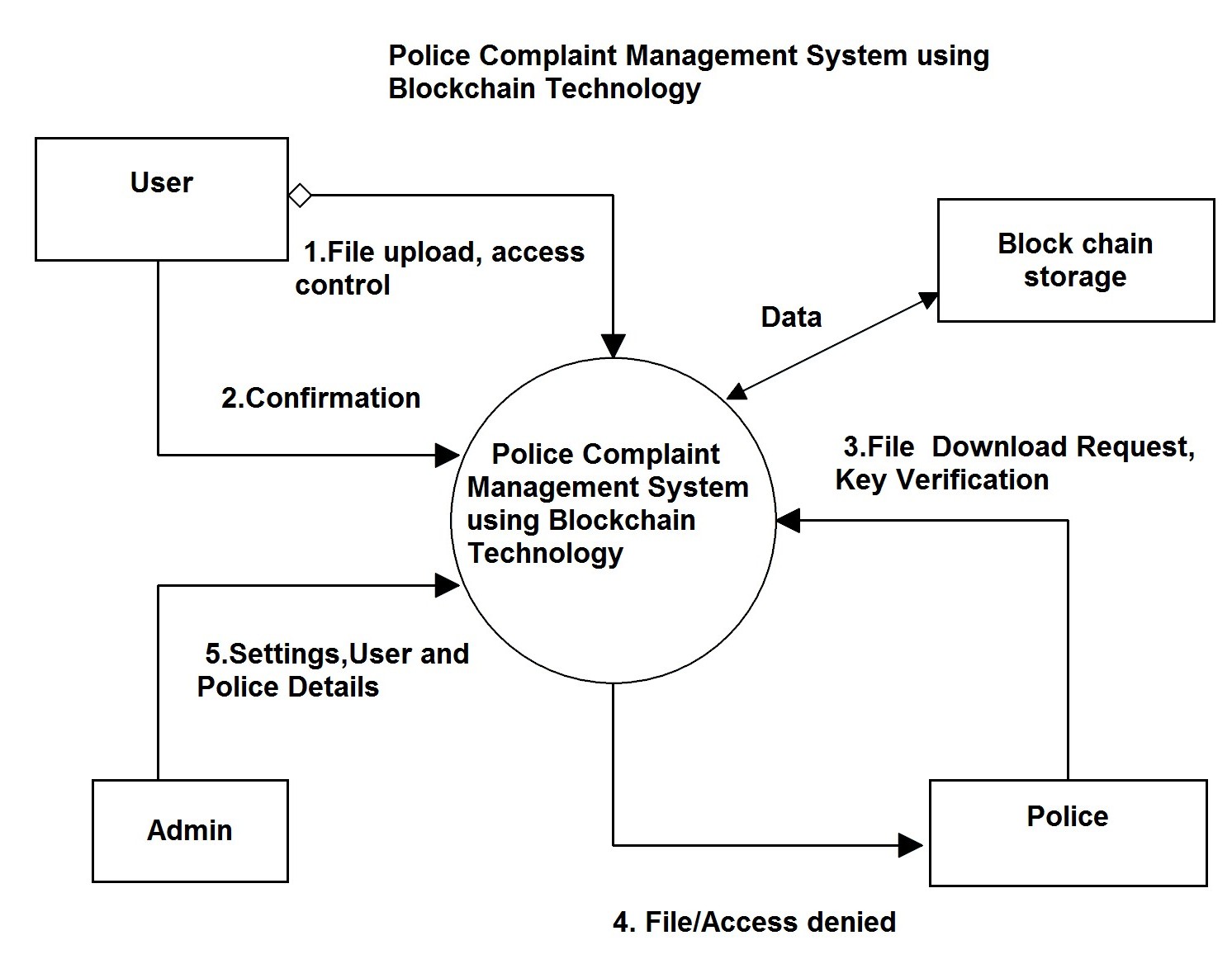
The Level 0 DFD illustrates how the system is broken down into smaller subsystems, or processes, that together handle all of the system's functionality. Each subsystem handles one or more data flows to or from an external agent. A level 0 DFD is shown in Figure 4.4.1.



**Fig10 Level 0 Data Flow diagram or Context Analysis Diagram**

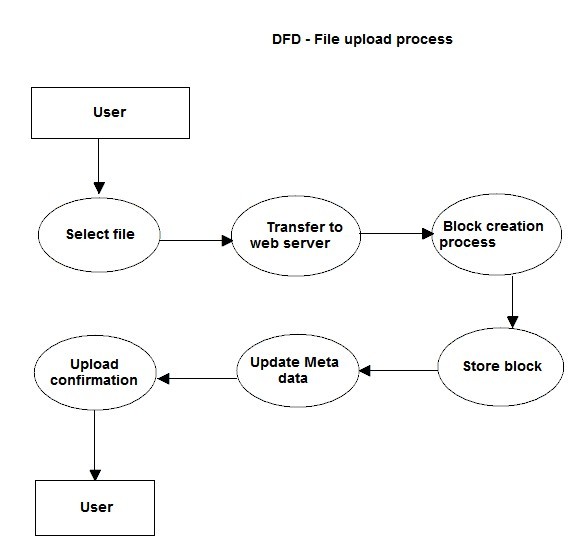
### 4.4.2 LEVEL1 DATA FLOW DIAGRAM:

Choose a file in the level 1 Data Flow Diagram, then move it to the server. After the server has all the information, it creates a message digest (MD file) and extracts every public key that belongs to the user group, or MD+Publickey. creates a safe MD, encrypts it using the user's private keys, creates a ring-signature, and sends an email to every user.



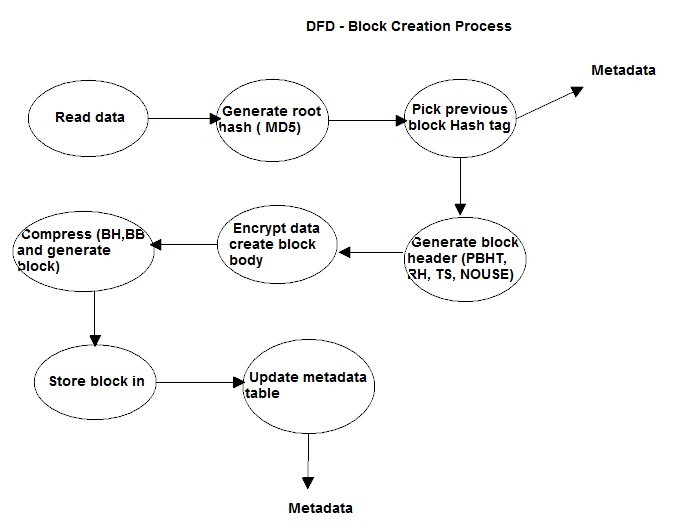
***Fig 11 Level1 Data Flow Diagram***

### 4.4.3 LEVEL2 DATA FLOW DIAGRAM:

****

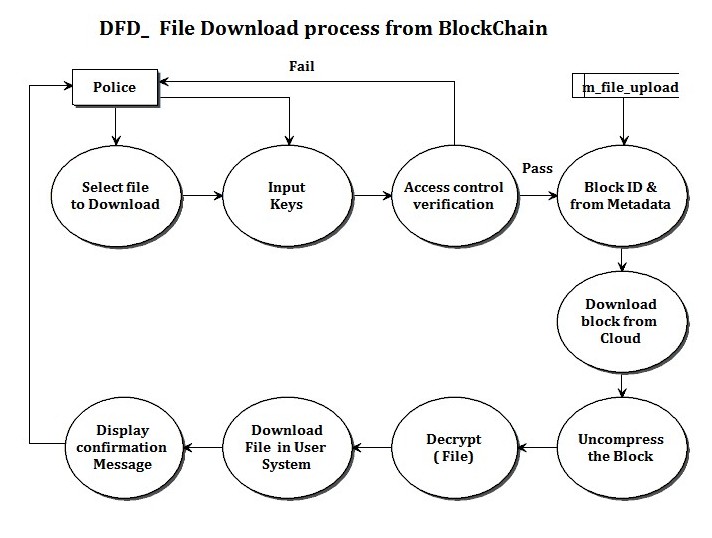
**Fig 12 Level 2 Data Flow Diagram**

### 4.4.4 LEVEL3 DATA FLOW DIAGRAM:

****

***Fig 13 Level 3 Data Flow Diagram***

### 4.4.5 LEVEL4 DATA FLOW DIAGRAM:

****

***Fig 14 Level 4 Data Flow Diagram***

## 4.6 Modules Description

1. **Admin**

The admin is the super user and he is also called trusted authority (TA) who will create cryptography keys.Admin has to create user and Police officers. The working of the admin module is given below:

* Login
* Profile
* View Profile
* Edit Profile
* Change Password
* User
* View User
* Add User
* Update User
* Police
* View Police
* Add Police
* Update Police
* Key Generation
* Department(Attribute I)
* Designation(Attribute II)
* Logout

1. **Users**

Users who own the complaint, user must have separate login page and he can able to login into home page using authorized password.Once user login he can upload his complaint into block chain server. User can able to provide file access details in this system. File access control are mentioned by attribute of the data consumers. The working of the User module is given below:

* Login
* Profile
* View Profile
* Edit Profile
* Change Password
* File Upload
* Upload File
* View Details
* File Access Control

Control File Access Control

View File Access Control

* Transaction
* Layout

1. **Police Officer**

Police officer can able to login into their home page with authorized password. Police can able to download files from Block Chain Server. The Police while downloading the files this system has to verify file access control test once the Police pass in the test then only Police can download the file.

* Login
* Profile
* View Profile
* Edit Profile
* Change Password
* File Download
* Transaction
* Logout

**CHAPTER-5**

**SYSTEM IMPLEMENTATION**

Writing code is not the only task involved in the implementation phase. Code must be compiled, built into a finished executable product, tested, and debugged. Typically, configuration management is required in order to maintain track of various code versions. At this point of the project, the theoretical design is implemented into a functional system. An unplanned or poorly managed installation can lead to confusion and mayhem. It's always a good idea to remember that some qualities, such as readability—our code is written in Java and MVC architecture to meet the project's goal of introducing a unique scheme of mechanism design—should be included in a decent implementation.

Our implementation stage requires the following tasks:

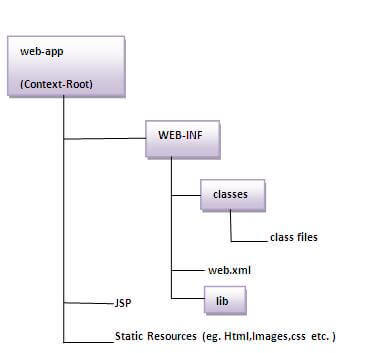
* Careful planning
* Investigation of system and constraints
* Design of methods to achieve the changeover
* Evaluation of the changeover method
* Correct decisions regarding selection of the platform
* Appropriate selection of the language for application development
* Java Technology is both a programming language and a platform.

## 5.1 JAVA

Java is a concurrent, class-based, object-oriented computer programming language that was created with the goal of having the fewest possible implementation dependencies. Its goal is to enable application developers to "write once, run anywhere" (WORA), which means that generated Java code can function without the need for additional compilation on any platform that supports Java. Generally, Java programs are compiled to bytecode, which is compatible with any computer architecture and may operate on any Java virtual machine (JVM). With a claimed 9 million developers as of 2016, Java is one of the most popular programming languages, especially for client-server web applications.

## 4.2 J2EE

Java Enterprise Edition, or Java EE as it is now called, was once known as J2EE and is now known as Jakarta EE. It is a collection of guidelines about Java SE (Standard Edition). Web services and distributed computing are two enterprise functionalities that the Java EE platform offers developers. Java EE programs are often executed on application servers or micro servers with reference run times. Applications for Java EE include banking information systems, accountancy, and e-commerce. Servlets, Java Server Pages (JSP), Java Database Connectivity (JDBC), Enterprise Java Beans (EJB), Java Message Services (JMS), and other technologies are included in the J2EE technologies. We employ the MVC architecture with Servlets, JSP, and JDBC technologies in our project.



**Fig 15: Directory structure of the web application**

### 4.3. MVC Architecture

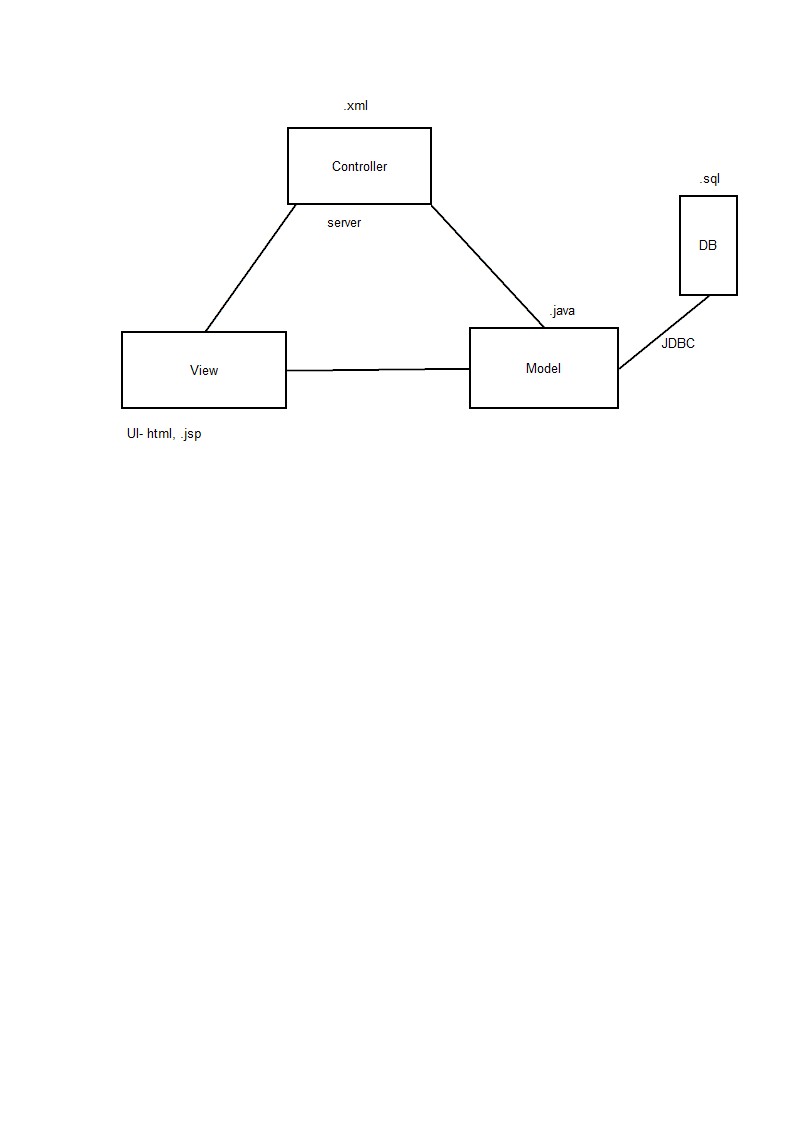
In the world of web development, the Model-View-Controller (MVC) design pattern is well recognized. It is a means of structuring our code. It stipulates that the data model, presentation information, and control information must all be included in a program or application. All these elements must exist as distinct objects in order for the MVC paradigm to work. We created a web-based application using a model-view-controller for this assignment.

MVC design pattern is followed by the model designs that are based on MVC architecture. When creating software, model designs are used to keep the application logic and user interface apart.Three layers make up the architecture of the MVC pattern:

• Model: It symbolizes the application's business layer. It is an object for carrying data that has the ability to have logic for updating the controller in the event that data changes. The.java files in our project serve as the Model.

• View: It stands for the application's display layer. It is employed to display the data that is present in the model. The.jsp files in our project, View, provide the user interface for our program.

• Controller: It functions on both the view and the model. It is employed to update the view whenever data is modified and to control the application flow, or the data flow within the model object. The web.xml file in our project, called Controller, is what governs our online application.



**Fig 17: MVC Architecture**

### 4.3 Servlets

A web application is made using servlets technology, which runs on the server side and produces dynamic web pages.Because of Java, Servlet technology is scalable and reliable. The Servlet API has a large number of interfaces and classes, including Servlet, GenericServlet, HttpServlet, ServletRequest, and ServletResponse.



**Fig 18: Servlets Technology working**

## 4.4 ECLIPSE

Eclipse is the most popular Java integrated development environment (IDE) that is used in computer programming. It comes with a basic workspace and an expandable system of plug-ins for personalizing the setup. Programming languages such as Ada, ABAP, C, C++, C#, Clojure, COBOL, D, Erlang, Fortran, Groovy, Haskell, JavaScript, Julia, Lasso, Lua, NATURAL, Perl, PHP, Prolog, Python, R, Ruby (including Ruby onRailsframework), Rust, Scala, and Scheme can also be developed with Eclipse, although its main purpose is to develop Java applications. It may also be used to create packages for the Mathematica program and papers written in LaTeX (using a TeXlipse plug-in).

## 4.5 TOMCAT

The Apache Software Foundation (ASF) created the Java ServletContainer. WebSocket, Java Servlet, Java Server Pages (JSP), Java EL, and other JavaEE standards are implemented by Tomcat, which also offers an HTTP web server environment that is "pure Java" and suitable for running Java applications.

## 4.6 MY SQL

As of 2008, the most popular open-source relational database management system (RDBMS) in the world is MySQL ("My Sequel"), which functions as a server and allows multiple users to access various databases. The acronym for Structured Query Language is SQL.   
The GNU General Public License and other proprietary agreements, in addition to the conditions of the MySQL development project, govern the use of the source code. An abbreviation for "Linux, Apache, MySQL, Perl/PHP/Python," MySQL is a well-liked database option for usage in online applications. It is also a key part of the popular LAMP open source web application software.MySQL is frequently used in free-software open-source projects that need a feature-rich database management system.

The relational database management system (RDBMS) MySQL comes without any graphical user interface (GUI) capabilities for managing data in the databases or for administering the databases themselves. In addition to the command line tools that come with the package, users may create and administer MySQL databases, construct database structures, back up data, check status, and deal with data records using the MySQL "front-ends," desktop software, and online applications. The majority of transactional relational databases, including MySQL, have significant limitations due to hard disk performance.

**CHAPTER-6**

**TESTING**

## 6.1SOFTWARE TESTING INTRODUCTION

Software testing is a procedure used to assess the accuracy, comprehensiveness, and quality of computer software that has been built. The method used to assess the calibre of generated software is called software testing.The process of running a software with the goal of identifying faults is called testing. Verification and validation are common terms used to describe software testing.

## 6.2 EXPLANATION FOR SDLC & STLC

The steps of an information system development project are described by the software development life cycle (SDLC), a conceptual model used in project management that starts with an initial feasibility assessment and ends with application maintenance.

## 6.3PHASES OF SOFTWARE DEVELOPMENT

* Requirement Analysis
* Software design
* Development or Coding
* Testing
* Maintenance

### **6.3.1 REQUIREMENT ANALYSIS**

The requirements of a desired software product are extracted. Based the business scenario the SRS (Software Requirement Specification) document is prepared in this phase.

### 6.3.2DESIGN

Plans are provided for the software's physical design, hardware, operating systems, programming, communications, and security. Ensuring the software system satisfies the product requirements is the focus of the design phase.

### 6.3.3TESTING

Testing is evaluating the software to check for the user requirements. Here the software is evaluated with intent of finding defects.

### 6.3.4MAINTENANCE

The new system has to be thoroughly assessed after a period of time of operation. Upkeep needs to be done consistently and scrupulously. It is important to keep system users informed about the most recent changes and protocols.

## 6.4 SDLC MODELS

### 6.4.1 WATER FALL MODEL

The SDLC processes will be carried out one at a time. The layout Coding starts after design, following the completion of the requirements analysis. This model is conventional. It's a sequential design technique that's frequently employed in the software development life cycle (SDLC), where progress is considered to be flowing through the various phases gradually (like a waterfall).

### 6.4.2 PROTO TYPE MODEL

Developed from the sample after getting good feedback from the customer. This is the Valuable mechanism for gaining better understanding of the customer needs

## 6.5 STLC (****Software Testing Life Cycle)****

Testing itself has many phases i.e. is called as STLC. STLC is part of SDLC

* Test Plan
* Test Development
* Test Execution
* Analyze Results
* Defect Tracking
* Summaries Report

### ****6.5.1 TEST PLAN**:**

This document outlines the testing environment, goals, objectives, scope, testing strategy, schedules, milestones, testing tool, roles and responsibilities, risks, staffing, training, and who will test the application. It also specifies the kinds of tests that should be run and how the defects will be tracked.

### ****6.5.2. TEST DEVELOPMENT:****

Preparing test cases, test data, Preparing test procedure, Preparing test scenario, Writing test script

### ****6.5.3 TEST EXECUTION:****

In this phase we execute the documents those are prepared in test development phase

### ****6.5.4 ANALYZE RESULT:****

Once executed documents will get results either pass or fail. we need to analyze the results during this phase.

## 6.6 .TYPES OF TESTING:

* White Box Testing
* Black Box Testing
* Grey box testing

### 6.6.1WHITE BOX TESTING:

As the name implies, white box testing provides an interior look at the program. Because the focus of this kind of testing is on what's within the box, it's also known as structural testing or glass box testing.

### 6.6.2BLACK BOX TESTING:

Another name for it is behavioral testing. It focuses on the software's functional needs. Black box testing is the process of testing a component or system, whether it is functioning or not, without consulting the internal workings of the system or component.

### 6.6.3 GREY BOX TESTING:

Combining black box and white box testing results in grey box testing. The goal of this testing is to identify any flaws resulting from poor system implementation or design.

**Testing Used For Web Based Application:**

This is carried out for three-tier apps (created for the intranet, extranet, and internet).We will have a web server, database server, and browser here. The browser-accessible apps would be created in HTML, DHTML, XML, JavaScript, and other languages. (We can keep an eye on things using these apps.)

## 6.7 LEVEL OF TESTING USED IN PROJECT:

**6.7.1. UNIT TESTING:**

The first stage of dynamic testing is called initialization testing, and it is the duty of test engineers after developers. Unit testing is carried out once the anticipated test outcomes are achieved or discrepancies are rationalized and deemed acceptable.

### 6.7.2. INEGRATION TESTING:

Every module that goes into an application is tested. The purpose of integration testing is to ensure that the consequences of the interaction between two or more components meet functional requirements.

### 6.7.3. SYSTEM TESTING:

To conduct a functional and non-functional test of the entire system. The Test Team conducts black box testing, and before any system testing begins, the entire system is set up in a controlled setting.

### 6.7.6. BETA TESTING:

The software's final user conducts the beta test at one or more client sites. The software's final user conducts the beta test at one or more client sites.

## 6.8 UNIT TESTING CASES:

The first stage of dynamic testing is called initialization testing, and it is the duty of test engineers after developers. Unit testing is carried out once the anticipated test outcomes are achieved or discrepancies are rationalized and deemed acceptable.

|  |  |
| --- | --- |
| Sl # Test Case : - | UNTC-1 |
| Name of Test: - | Login as Admin |
| Items being tested: - | Admin model |
| Sample Input: - | Correct Username & Password is given as inputs |
| Expected output: - | Depending on the correct inputs, it must login as Admin |
| Actual output: - | Login successful |
| Remarks: - | Pass. |

|  |  |
| --- | --- |
| Sl # Test Case : - | UNTC-2 |
| Name of Test: - | Login as User |
| Items being tested: - | User model |
| Sample Input: - | Incorrect Username & Password is given as inputs |
| Expected output: - | Depending on the incorrect inputs, it shouldn’t login as User |
| Actual output: - | Login Failed. |
| Remarks: - | Pass |

## 6.9 SYSTEM TESTING

To conduct a functional and non-functional test of the entire system. The Test Team conducts black box testing, where the entire system is setup in a controlled environment before the test begins.

|  |  |
| --- | --- |
| Sl # Test Case : - | STC-1 |
| Name of Test: - | System testing in various versions of OS |
| Sample Input: - | Execute the program in windows server 2003/XP/ Windows-7 |
| Expected output: - | Performance is better in windows-8 |

|  |  |
| --- | --- |
| Actual output: - | Same as expected output, performance is better in windows-8 |
| Remarks: - | Pass |

## 6.10 Functional Testing:

This kind of black-box testing, often known as quality assurance (QA), builds its test cases on the requirements of the software component being tested. Internal program structure is rarely taken into account while testing functions; instead, input is fed into the function and the result is examined (unlike white-box testing).

**6.11 INTEGRATION TESTING:**

The outbound links from each page inside the domain that is being tested. Examine each internal connection. Examine links that jump to the same pages. Verify the fields' default values. incorrect entries made in the form's fields.

|  |  |
| --- | --- |
| Sl # Test Case : - | ITC-1 |
| Name of Test: - | Change Password for Admin |
| Item being tested: - | Admin & User Model |
| Sample Input: - | Incorrect Old password, give new password. |
| Expected output: - | Failed to update new password. |
| Actual output: - | Password Update failed |
| Remarks: - | Pass. |

**CHAPTER-6**

**RESULTS AND DISCUSSION**

The outbound links from each page inside the domain that is being tested. Examine each internal connection. Examine links that jump to the same pages. Verify the fields' default values. incorrect entries made in the form's fields.

**1.Login page**

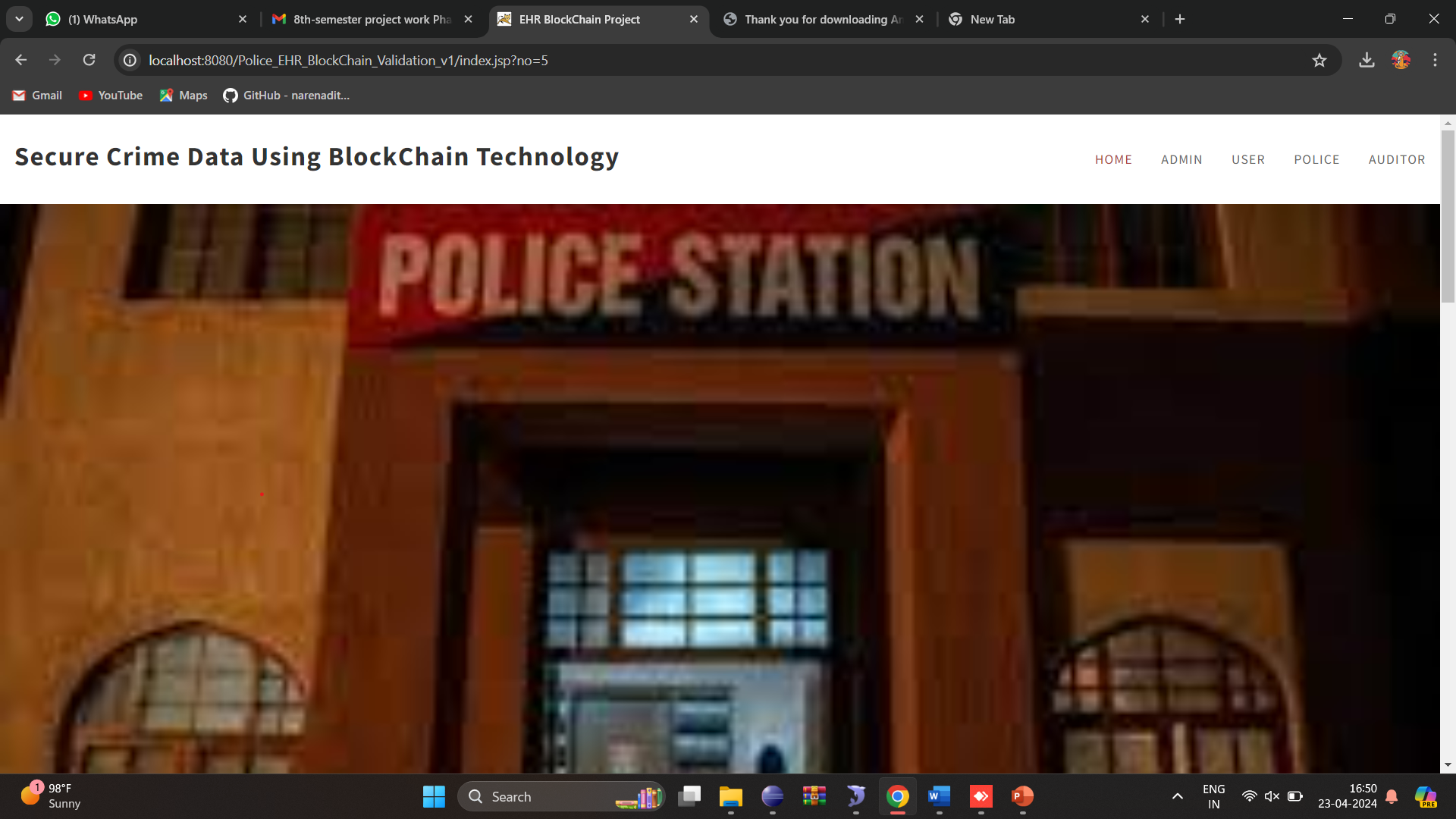
****

Fig 18. Home Page

**2.Login page for the admin, user and police**

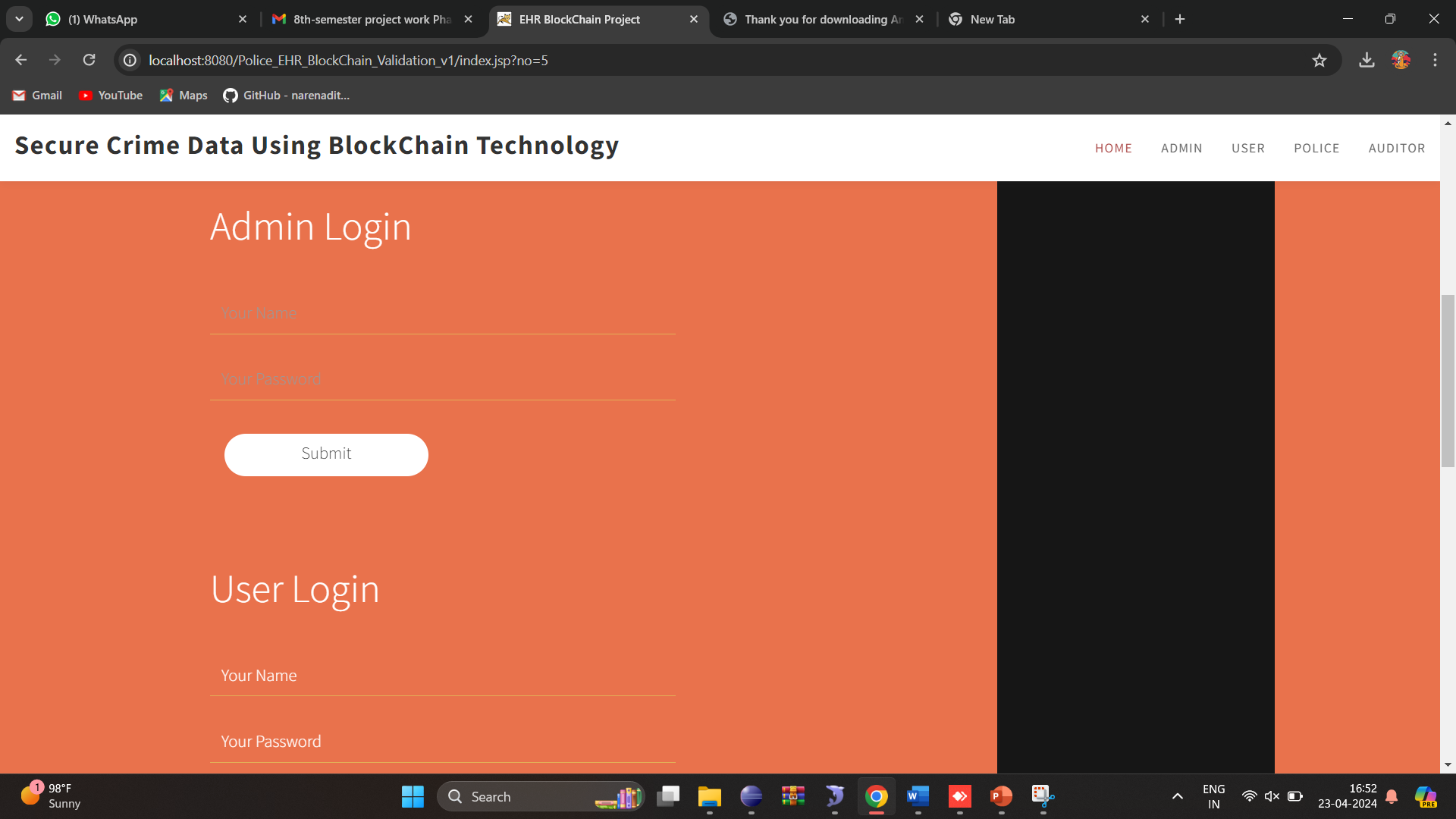
****

Fig.19 Login page for the admin, user and police

**3.Admin details**

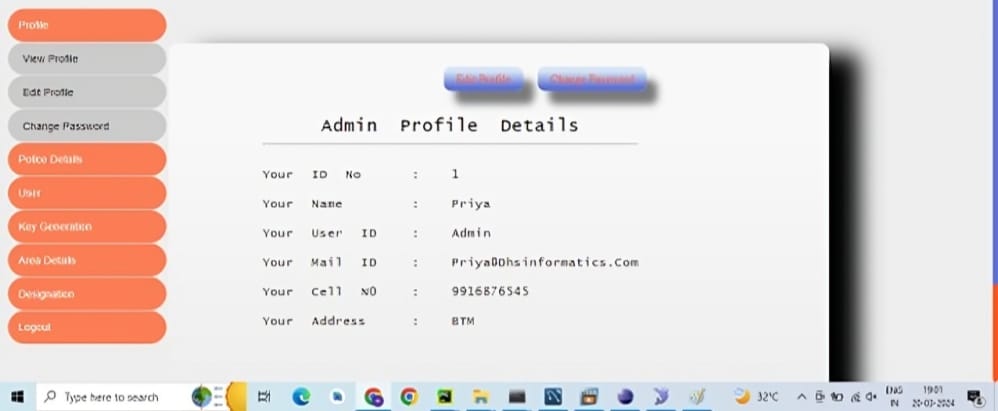
****

Fig.20 Admin details

**4.File details**

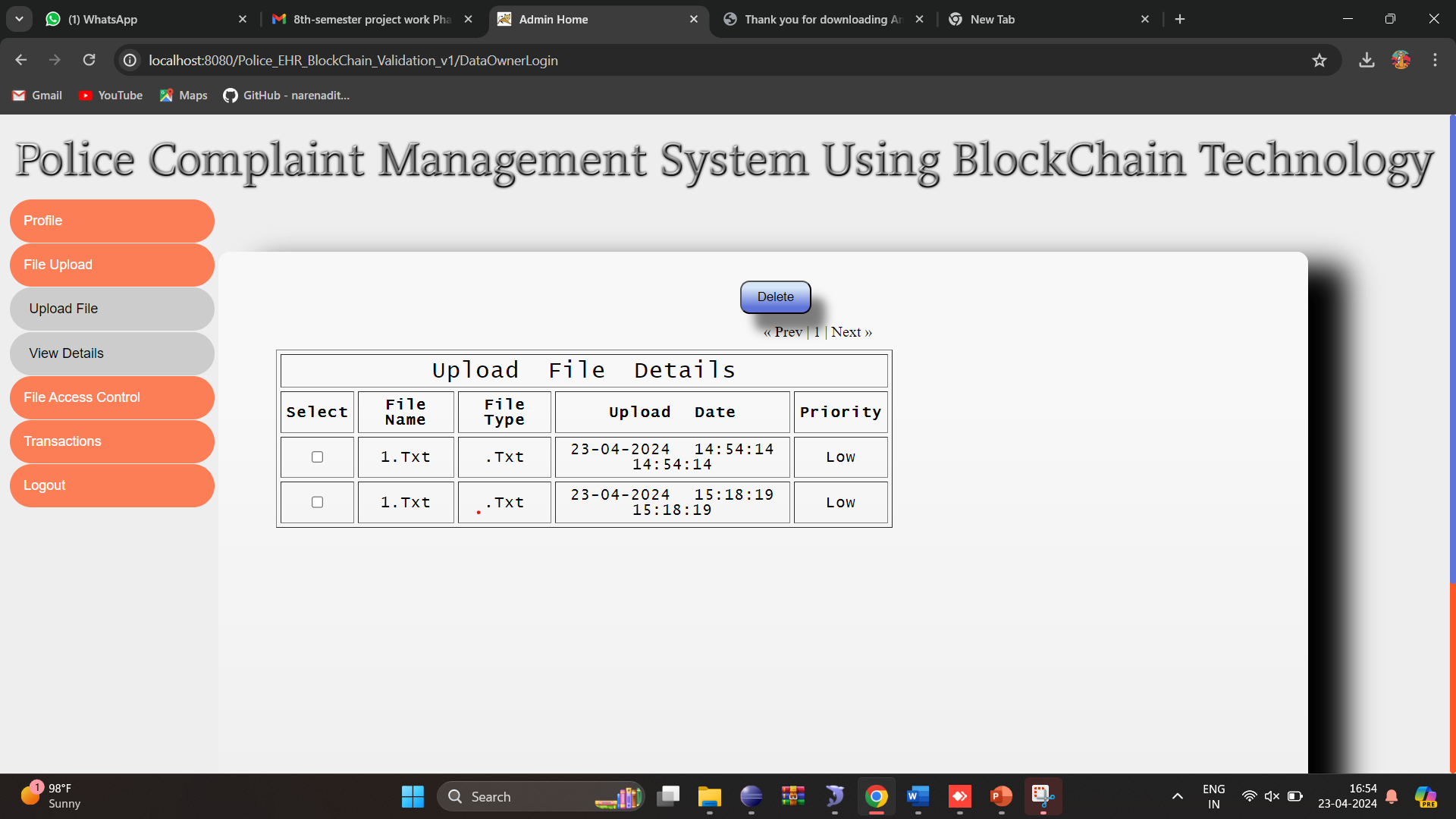
****

Fig.21 File details

**5.File access control details**

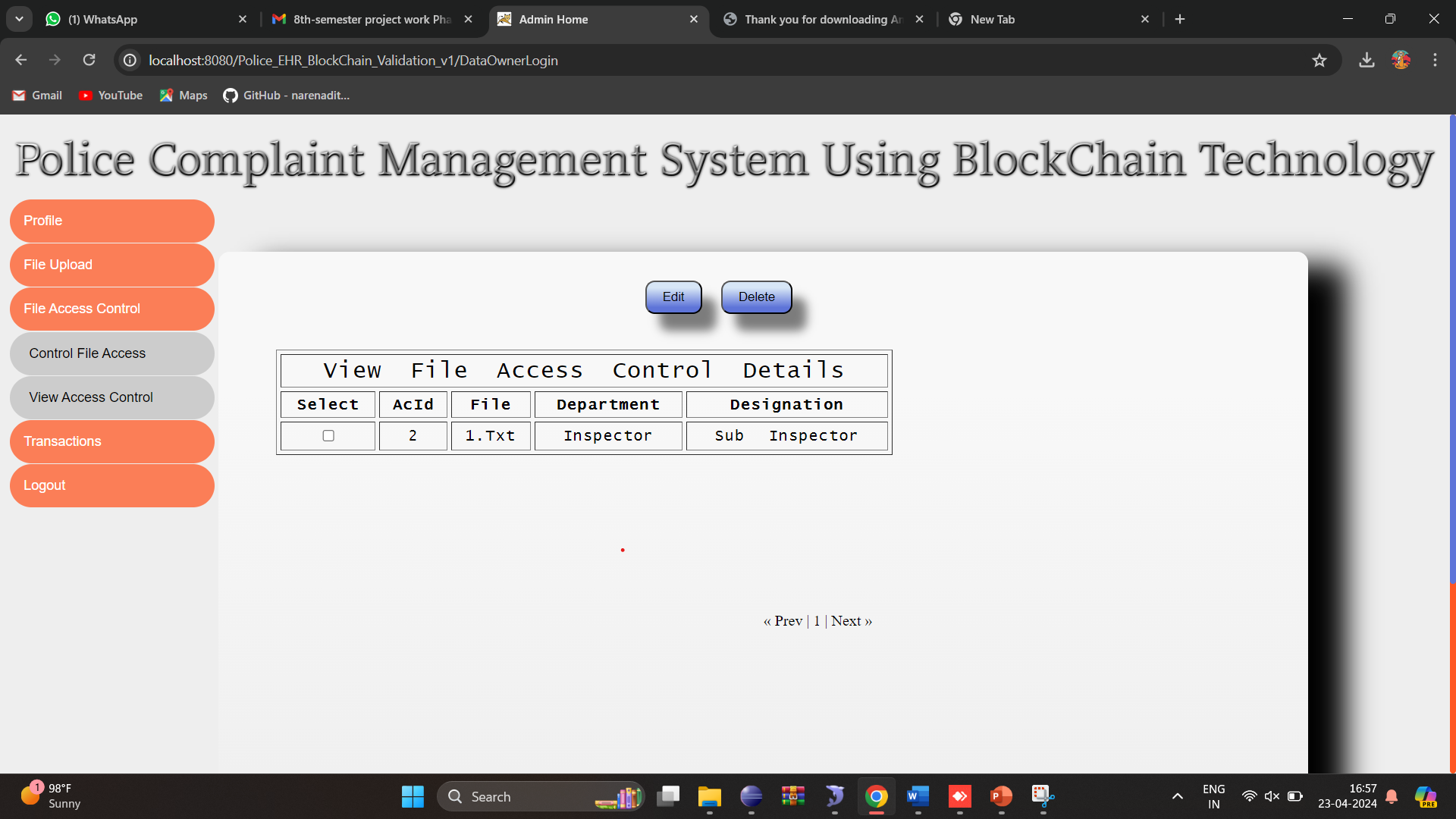


Fig.22 File access control details

Furthermore, data integrity and confidentiality are strengthened by blockchain's cryptographic security features, which promote increased public-law enforcement confidence. All things considered, the incorporation of blockchain technology into the police complaint management system has the potential to completely transform the way complaints are processed, resulting in more dependable, equitable, and effective results.

**CHAPTER-8**

**CONCLUSION AND FUTURE WORK**

The Secure Crime data system's use of blockchain technology offers a strong remedy for persistent problems with law enforcement procedures. The technology improves confidence between individuals and authorities by utilizing blockchain's inherent security, transparency, and immutability. Additionally, it streamlines complaint resolution procedures. Smart contract adoption increases overall efficiency and accountability by automating and expediting the treatment of complaints. Future research in this field may examine new features like using artificial intelligence for predictive analytics to spot wrongdoing tendencies or improving user interfaces for easier accessibility. Furthermore, studies should concentrate on interoperability and scalability to guarantee smooth interface with current law enforcement systems and promote broad use.

There are several directions that work in blockchain technology advancement of the police complaint management system might go in the future. First, in order to handle the high amount of complaints that law enforcement organizations handle, research and development activities might concentrate on improving the scalability and efficiency of blockchain networks. To increase transaction throughput and lower latency, sharding and layer 2 protocols are examples of scalable methods that might be investigated.

Secondly, it is imperative to tackle privacy issues while upholding transparency. Future research might look into privacy-preserving methods to safeguard sensitive data on the blockchain and maintain openness in the complaint resolution process, such as differential privacy or zero-knowledge proofs.

**REFERNECES**

[1] D. Song, A. Perrig, and D. Wagner, “Practical techniques for searcheson encrypted data,” Proceeding 2000 IEEE Symposium on Security and Privacy. IEEE, 2000, pp. 44–55.

[2] J. Bethencourt, A. Sahai, and B.Waters, “Ciphertext-policy attribute-basedencryption,” In Proc. IEEE Symp. Security Privacy, 2007, pp. 321–334.

[3] M. Chase and S. S. Chow, “Improving privacy and security in multiauthorityattribute-based encryption,” In Proceedings of the 16th ACMconference on Computer and communications security. ACM, 2009, pp.121–130.

[4] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, “Searchable symmetricencryption: improved definitions and efficient constructions,” ACMConference on Computer and Communications Security 2006. ACM, 2006,pp. 79–88,

[5] C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, “Secure ranked keywordsearch over encrypted cloud data,” IEEE 30th International Conference onDistributed Computing Systems. IEEE, 2010, pp. 253–262.

[6] N. Premasathian and S. Choto, “Searchable encryption schemes: withmultiplication and simultaneous congruences,” 2012 9th International ISCConference on Information Security and Cryptology. IEEE, Tabriz, 2012,pp. 147–150.

[7] H. Li, F. Zhang, J. He, and H. Tian, “A searchable symmetricencryption scheme using blockChain,” [Online]. Available:https://arxiv.org/pdf/1711.01030.pdf.

[8] Y. Zhang, R. H. Deng, J. Shu, K. Yang, and D. Zheng, “TKSE: Trustworthykeyword search over encrypted data with two-side verifiability viablockchain,” IEEE Access., vol. 6, pp. 31077–31087, 2018.

[9] A.Sahai and B.Waters, “Fuzzy identity-based encryption,” Annual InternationalConference on the Theory and Applications of CryptographicTechniques. Springer, Berlin, Heidelberg, vol. 3494, 2005, pp. 457–473.

[10] V. Goyal, O. Pandey, A. Sahai, and B.Waters, “Attribute-based encryptionfor fine-grained access control of encrypted data,” Proceedings of the 13thACM Conference on Computer and Communications Security. ACM, 2006,pp. 89–98.