Safe Kerala

Project Report Submitted By

REENU JOSEPH

Reg. No: AJC20MCA-2059

In Partial fulfillment for the Award of the Degree Of

MASTER OF COMPUTER APPLICATIONS(2 Year)
(MCA)
APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

AMAL JYOTHI COLLEGE OF ENGINEERING
KANJIRAPPALLY

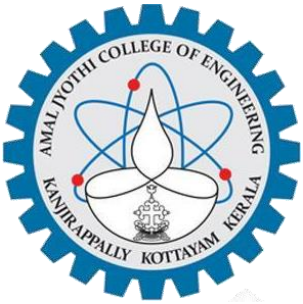
[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala – 686518]

2020-2022

DEPARTMENT OF COMPUTER APPLICATIONS

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY



CERTIFICATE

This is to certify that the Project report, "**Safe Kerala"** is the bonafide work of **REENU JOSEPH (Reg.No:AJC20MCA-2059)** in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications under APJ Abdul Kalam Technological University during the year 2021-2022

Ms.Nimmy Francis
Internal Guide

Ms.Grace Joseph
Coordinator

Rev.Fr.Dr.Rubin Thottupurathu Jose
Head of the Department

External Examiner

(ADD PLAGIARISM CERTIFICATE)

DECLARATION

I hereby declare that the project report **"Safe Kerala"** is a bonafided work done at Amal Jyothi College of Engineering, towards the partial fulfilment of the requirements for the award of the Degree of Master of Computer Applications (MCA) from APJ Abdul Kalam Technological University, during the academic year 2021-2022.

Date: 21/07/2022 **REENU JOSEPH**

KANJIRAPPALLY **Reg. No: AJC20MCA-2059**

ACKNOWLEDGEMENT

First and foremost, I thank God almighty for his eternal love and protection throughout the project. I take this opportunity to express my gratitude to all who helped me in completing this project successfully. It has been said that gratitude is the memory of the heart. I wish to express my sincere gratitude to our manager **Rev. Fr. Dr. Mathew Paikatt** and Principal **Dr. Lillykutty Jacob** for providing good faculty for guidance.

I owe a great depth of gratitude towards our Head of the Department **Rev.Fr.Dr. Rubin Thottupurathu Jose** for helping us. I extend my whole hearted thanks to the project coordinators **Rev.Fr.Dr. Rubin Thottupurathu Jose** and **Ms**.**Grace Joseph** for their valuable suggestions and for overwhelming concern and guidance from the beginning to the end of the project. I would also like to express sincere gratitude to my guide**, Ms.Nimmy Francis** for her inspiration and helping hand.

I thank our beloved teachers for their cooperation and suggestions that helped me throughout the project. I express my thanks to all my friends and classmates for their interest, dedication, and encouragement shown towards the project. I convey my hearty thanks to my family for the moral support, suggestions, and encouragement to make this venture a success.

REENU JOSEPH

ABSTRACT

.At present it is very difficult to find authorized workers. Workers Portal is a solution for this difficulty. It is an interactive web enabled application. After identity verification from police workers details can be added to this site. So the users can easily search authorized workers with their needs. Police can block the workers from site if they are in criminal list. The main website feature is admin can monitor all functions and provide authorized workers for users.

.

CONTENT

List of Abbreviation

IDE	-	Integrated Development Environment
HTML	-	Hyper Text Markup Language.
CSS	-	Cascading Style Sheet
SQL	-	Structured Query Language
UML	-	Unified Modeling Language

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

At present it is very difficult to find authorized workers.Workers Portal is a solution for this difficulty. It is an interactive web enabled application. After identity verification from police workers details can be added to this site. So the users can easily search authorized workers with their needs. Police can block the workers from site if they are in criminal list.. The main website feature is admin can monitor all functions and provide authorized workers for users.

1.2PROJECT SPECIFICATION

The system includes 4 modules. They are:

Main Users are,

1. Admin
2. Contractor
3. Police
4. Users

Functions of Administrator:

Admin can do the following functions:

- Add Category
- Contractor Approval
- View Labours

Functions of Contractor:

Contractor can do the following functions:

- Add labours
- Rate per day
- View Request
- View payment details

Functions of Police:

Police can do the following functions:

- Labour verification
- Block Labours
- View Labours

Functions of Users:

Users can do the following functions:

- Search Labours
- Send Message
- Payment

CHAPTER 2

SYSTEM STUDY

2.1 INTRODUCTION

System analysis is the process of acquiring and evaluating facts, detecting issues, and recommending system changes. It is a problem-solving activity that necessitates close collaboration between system users and system developers. A critical part of any system development process is system analysis or research. The system is meticulously investigated and assessed. The system analyst assumes the role of interrogator and delves deeply into the current system's operation. The system is seen as a whole, and the system's input is recognised. The numerous procedures are linked to the outputs of the organisations. The goal of system analysis is to identify the problem, identify the important and decisional variables, analyse and synthesise the numerous components, and determine an optimal or at least adequate solution or plan of action.

A thorough examination of the process must be conducted using various approaches such as interviews, questionnaires, and so on. To reach a conclusion, the data gathered by various sources must be evaluated. The end result is a grasp of how the system works. This is known as the existing system. The present system is now being closely examined, and problem areas are being discovered. The designer now acts as a problem solver, attempting to resolve the issues that the company is experiencing. The solutions are presented as suggestions. The suggestion is then analytically compared against the existing system, and the best one is chosen. The suggestion is submitted to the user for his or her approval. On user request, the proposal is assessed and appropriate revisions are implemented. This is a loop that finishes when the user is pleased with the proposal.

Preliminary research is the process of acquiring and analysing facts in preparation for subsequent research on the system. Preliminary research is a problem-solving activity that necessitates frequent contact between system users and system developers. It does a variety of feasibility studies. These investigations can provide an approximate estimate of the system activities, which can be used to make decisions regarding the tactics to be used for successful system research and analysis.

2.2 EXISTING SYSTEM

For the successful designing of the management system it is necessary to study and analyze the present working mode of the existing system. Here it has been attained through the interview of the staffs using the existing system as well as the related officials. The existing system contains all the work done manually. All the details regarding the resource allocation are stored in separate register. The provisions for the citizens include registering a complaint. Different files are maintained for registering complaint, with drawing complaint, and also citizen list and to acquire other details. If a police reports a crime the details regarding the crime are kept in a file. Citizen details have to be maintained using separate files .The address of the citizen along with his all details is to be recorded. The retrieval of these data at times of emergencies is very tough task. Maintenance of proper records and other files are necessary in this site.

2.3 DRAWBACKS OF EXISTING SYSTEM

- There is no adequate online system administration.
- Human effort is required
- More manual hours are required to create the relevant reports.

2.4 PROPOSED SYSTEM

In the proposed system we proposed to computerize the above mentioned activities. In the existing system, all data processing is done manually. All the files and record books are replaced by the system software. When there are numerous problems, such as with information retrieval and storage, reporting, etc., keeping track of them becomes a difficult effort. The constraint of the current method will be lessened by the use of a computerised system. Efficiency and precision can be greatly improved while manpower needs can be decreased significantly. The proposed solution can also significantly reduce time consumption if it is put into practise.

1.5 ADVANTAGES OF PROPOSED SYSTEM

The system is very simple in design and to implement. The system requires very low system resources, and the system will work in almost all configurations. It has got following features:

- **Better security: -**

  For data to remain secure measures must be taken to prevent unauthorized access. Security means that data are protected from various forms of destruction. The system security problem can be divided into four related issues: security, integrity, privacy and confidentiality. Username and password requirement to sign in ensures security. It will also provide data security as we are using the secured databases for maintaining the documents.

- **Ensure data accuracy: -**

  The proposed system eliminates the manual errors while entering the details of the users during the registration.

- **Better service: -**

  The product will avoid the burden of hard copy storage. We can also conserve the time and human resources for doing the same task. The data can be maintained for longer period with no loss of data.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

A feasibility study is conducted to determine whether the project, once completed, will serve the organization's goal for the amount of labour, effort, and time invested in it.A feasibility study allows the developer to forecast the project's future and usefulness. A feasibility study of a system plan is based on its workability, which is the organization's influence, ability to satisfy user demands, and effective use of resources. . As a result, when a new application is proposed, it is usually subjected to a feasibility assessment before being allowed for development.

The paper describes the project's viability and contains numerous aspects that were carefully addressed throughout the project's feasibility assessment, such as technical, economic, and operational feasibilities. It has the following characteristics: -

3.1.1 Economical Feasibility

The expense and value of constructing the system must be justified. Criteria to ensuring that effort is focused on projects that will provide the best results and return the earliest. The cost of developing a new system is one of the elements that influence its development.

During the preliminary inquiry, the following serious financial issues were raised:

- The expenses of doing a thorough system inquiry.
- The cost of hardware and software.
- The advantages in terms of cost savings or less costly errors.

The suggested system is being created as part of a project, thus there will be no manual costs associated with it. Also, because all of the resources are already accessible, it indicates that the system is economically feasible for development.

The cost of the SAFE KERALA project was divided according to the system utilised, the cost of development, and the cost of hosting the project. The project was produced at a modest cost, according to all estimations. Because it is entirely built with open source software.

3.1.2 Technical Feasibility

The system must first be analysed from a technological standpoint. This feasibility study must be based on an outline design of the system requirements in terms of input, output, programmes, and processes. Once an outline system has been discovered, the study must proceed to recommend the kind of equipment, needed technique of constructing the system, and manner of running the system once it has been developed.

Technical issues raised during the investigation are:

- Is the present technology adequate for the proposed one?
- Can the system be expanded once it is developed?

The project should be designed in such a way that the required functionalities and performance are met within the limits. The project necessitates the use of a high-resolution scanning instrument as well as cryptographic procedures. Although the technology may become obsolete after a certain length of time, the system may still be utilised since newer versions of the same programme support older ones. As a result, there are little limits in this project. The system was created with PHP in the front end and MySQL in the back end; the project is technically possible for development. The system was created with PHP in the front end and MySQL in the back end; the project is technically possible for development. The system employed was likewise of high processor performance.

3.1.3 Behavioral Feasibility

The following questions are included in the proposed system:

- Is there adequate help for users?
- Will the suggested system be harmful?

When designed and implemented, the project would be advantageous since it meets the objectives. All behavioural factors are thoroughly studied, and the project is determined to be behaviorally viable.

3.2 SYSTEM SPECIFICATION

3.2.1 Hardware Specification

Processor - Intel core i3

RAM - 4 GB

Hard disk -1 TB

3.2.2 Software Specification

Front End - HTML, CSS

Backend - MySQL

Client on PC - Windows 7 and above.

Technologies used - JS, HTML5, AJAX, J Query, PHP, CSS

3.3 SOFTWARE DESCRIPTION

3.3.1 PHP

PHP is a server-side scripting language that is used for web development but may also be used for general-purpose programming. PHP is presently on over 244 million websites and 2.1 million web servers. The PHP group presently produces the reference implementation of PHP, which was founded by Rasmus Ledorf in 1995. While PHP initially stood for personal home page, it is now a recursive abbreviation for PHP:HypertextPreprocessor. A web server with a PHP processor module interprets PHP code and creates the resultant web page. Instead of contacting an external file to handle data, PHP instructions can be inserted directly into an HTML source document. It has also grown to include a command-line interface and may be used in standalone mode that is incompatible with.

3.3.2 MySQL

Oracle Corporation develops, distributes, and supports MySQL, the most popular Open Source SQL database management system. The MySQL Web site contains up-to-date information regarding MySQL software.

- **MySQL is a database management system.**

A database is a well-organized collection of data. It might be anything from a basic grocery list to a photo gallery or the massive volumes of data in a business network. A database management system, such as MySQL Server, is required to add, access, and process data contained in a computer database. Database management systems play an important role in computing, either as independent utilities or as components of other programmes, because computers are exceptionally adept at processing enormous volumes of data.

- **MySQL databases are relational.**

A relational database stores data in distinct tables rather than in a single large storehouse. The database structures are grouped into physical files that are optimised for performance. The logical model, which includes objects like databases, tables, views, rows, and columns, provides a versatile programming environment. You define the connections between distinct data fields, such as one-to-one, one-to-many, unique, necessary, or optional, as well as "points" between different tables. The database enforces these constraints, ensuring that your application never encounters inconsistent, duplicate, orphan, out-of-date, or missing data. SQL is an abbreviation for "Structured Query Language."

SQL is the most widely used standardised language for accessing databases. Depending on your programming environment, you may input SQL directly (for example, to produce reports), embed SQL statements in other language code, or utilise a language-specific API that hides the SQL syntax. The ANSI/ISO SQL Standard defines SQL. Since 1986, the SQL standard has been changing, and various variants exist. In this document, "SQL92" refers to the 1992 standard, "SQL: 1999" refers to the 1999 standard, and "SQL: 2003" refers to the current version of the standard. We refer to "the SQL standard" as the current version of the SQL Standard at any given moment.

- **MySQL software is Open Source.**

Anyone may use and change the programme because it is open source. Anyone may use the MySQL software for free by downloading it from the Internet. You are free to examine the source code and modify it to suit your needs. The GPL (GNU General Public License) is used by MySQL to describe what you may and cannot do with the

programme in certain scenarios. If you are unhappy with the GPL or require MySQL code to be embedded in a business application, you may purchase a commercially licenced version from us. For further information, see the MySQL Licensing Overview.

- **The MySQL Database Server is very fast, reliable, scalable, and easy to use.**

If that's what you're after, you should give it a shot. MySQL Server may operate easily with your other apps, web servers, and so on on a desktop or laptop, requiring little or no attention. If you dedicate a whole system to MySQL, you may configure it to take advantage of all available memory, CPU power, and I/O capacit

- **MySQL Server works in client/server or embedded systems.**

MySQL Database Software is a client/server system that includes a multi-threaded SQL server that supports several backends, a variety of client applications and libraries, administration tools, and a diverse set of application programming interfaces (APIs). MySQL Server is also available as an embedded multi-threaded library that you can integrate into your programme to create a smaller, quicker, and easier-to-manage standalone offering.

CHAPTER 4

SYSTEM DESIGN

4.1 INTRODUCTION

The initial stage in the development process of any designed product or system is design. Design is an artistic process. A good design is essential for an efficient system. The term "design" is described as "the act of using numerous approaches and concepts to specify a process or a system in sufficient detail to allow its physical implementation." It may be described as the process of employing multiple methodologies and concepts to specify a device, process, or system in sufficient depth to allow physical implementation. Software design is the technical core of the software engineering process and is employed independently of development methodology. The architectural detail necessary is developed by the system design.

4.2 UML DIAGRAM

UML is a standard language for describing, visualising, building, and documenting software system artefacts. The Object Management Group (OMG) produced UML, and the UML 1.0 standard draught was submitted to the OMG in January 1997.

UML stands for Unified Modeling Language. UML is distinct from other popular programming languages such as C++, Java, COBOL, and so on. To design software blueprints, a visual language known as UML is employed. UML is a general-purpose visual modelling language used for software system visualisation, specification, development, and documentation. Despite the fact that this is its most prevalent use, UML is used for more than merely representing software systems. It is also used to simulate non-software systems. For example, consider the process flow in a manufacturing unit. UML is not a programming language, but UML diagrams may be used to create code in a variety of languages. UML is inextricably linked to object-oriented analysis and design. UML has gained popularity after some standardisation.

- Class diagram
- Object diagram
- Use case diagram
- Sequence diagram
- Activity diagram
- Statechart diagram
- Deployment diagram

4.2.1 USE CASE DIAGRAM

A use case diagram is a graphical representation of the interactions between system parts. A use case is a strategy for identifying, clarifying, and organising system requirements in system analysis. The term "system" in this sense refers to something that is being constructed or operated, such as a mail-order goods sales and service Web site. UML (Unified Modeling Language), a standard language for modelling real-world objects and systems, employs use case diagrams.Planning general requirements, verifying a hardware design, testing and debugging a software product in development, producing an online help reference, or completing a consumer-service-oriented activity are all examples of system objectives. In a product sales scenario, for example, use cases might involve item ordering, catalogue updating, payment processing, and customer interactions. A use case diagram is made up of four parts.

- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.
- The use cases, which are the specific roles are played by the actors within and around the system.
- The relationships between and among the actors and the use cases.

Use case diagrams are used to capture a system's functional needs. After identifying the things listed above, we must follow the principles below to create an effective use case diagram.

- The name of a use case is very important. The name should be chosen in such a way so that it can identify the functionalities performed.
- Give a suitable name for actors.
- Show relationships and dependencies clearly in the diagram.
- Do not try to include all types of relationships, as the main purpose of the diagram is to identify the requirements.
- Use notes whenever required to clarify some important points

Figure 4.2.1-Usecase Diagram

4.2.2 SEQUENCE DIAGRAM

A sequence diagram simply illustrates object interactions in a sequential order, i.e. the order in which these interactions occur. A sequence diagram can also be referred to using the words event diagrams or event scenarios. Sequence diagrams show how and in what sequence objects in a system work. Businesspeople and software engineers frequently use these diagrams to describe and understand requirements for new and current systems.

Sequence Diagram Notations –

i. **Actors** – In a UML diagram, an actor represents a sort of role that interacts with the system and its objects. It is vital to note that an actor is always beyond the scope of the system that we are attempting to depict using the UML diagram. Actors play a variety of roles, including human users and other external subjects. A stick person notation is used to represent an actor in a UML diagram. A sequence diagram can have several actors.

ii. **Lifelines –** A lifeline is a named element in a sequence diagram that represents an individual participant. So, in a sequence diagram, each incident is represented by a lifeline. A sequence diagram's lifeline elements are at the top.

iii. **Messages –** Messages are used to illustrate communication between items. The messages show on the lifeline in chronological sequence. Messages are represented by arrows. A sequence diagram is built around lifelines and messages.

iv. Messages can be divided into the following categories:

v. Synchronous messages

- Asynchronous Messages
- Create message
- Delete Message
- Self-Message
- Reply Message

vi. **Guards** – In UML, we utilise guards to model circumstances. They are used to restrict the flow of messages under the guise of a condition being satisfied. Guards play a significant function in informing software developers about the limits associated with a system or a certain procedure.

vii. Uses of sequence diagrams –

- Used to model and visualize the logic behind a sophisticated function, operation or procedure.
- They are also used to show details of UML use case diagrams.
- Used to understand the detailed functionality of current or future systems.
- Visualize how messages and tasks move between objects or components in a system.

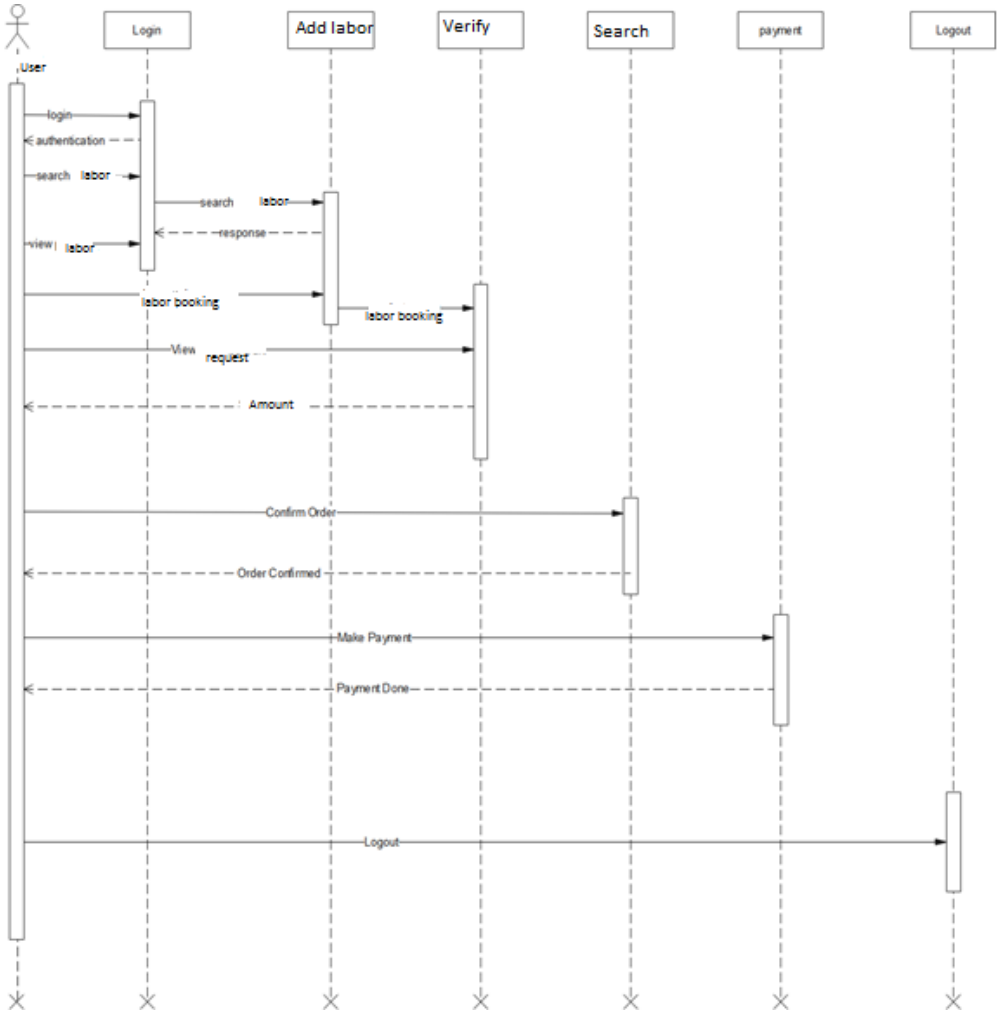


Figure 4.2.2-Sequence Diagram

4.2.3 ACTIVITY DIAGRAM.

Another essential diagram in UML for describing the dynamic characteristics of the system is the activity diagram.An activity diagram is essentially a flowchart that depicts the flow from one activity to another. The action can be defined as a system operationThe control flow is directed from one operation to the next. This flow might be sequential, branching, or concurrent in nature. Activity diagrams deal with all types of flow control by employing various features such as fork, join, and so on.

Purpose of Activity Diagrams

The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.

Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

It does not show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not. It shows different flows such as parallel, branched, concurrent, and single.

The purpose of an activity diagram can be described as −

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

How to Draw an Activity Diagram?

Activity diagrams are mainly used as a flowchart that consists of activities performed by the system. Activity diagrams are not exactly flowcharts as they have some additional capabilities. These additional capabilities include branching, parallel flow, swimlane, etc.Before drawing an activity diagram, we must have a clear understanding about the elements used in activity diagram. The main element of an activity diagram is the activity itself. An activity is a function performed by the system. After identifying the activities, we need to understand how they are

associated with constraints and conditions.Before drawing an activity diagram, we should identify the following elements −

- Activities
- Association

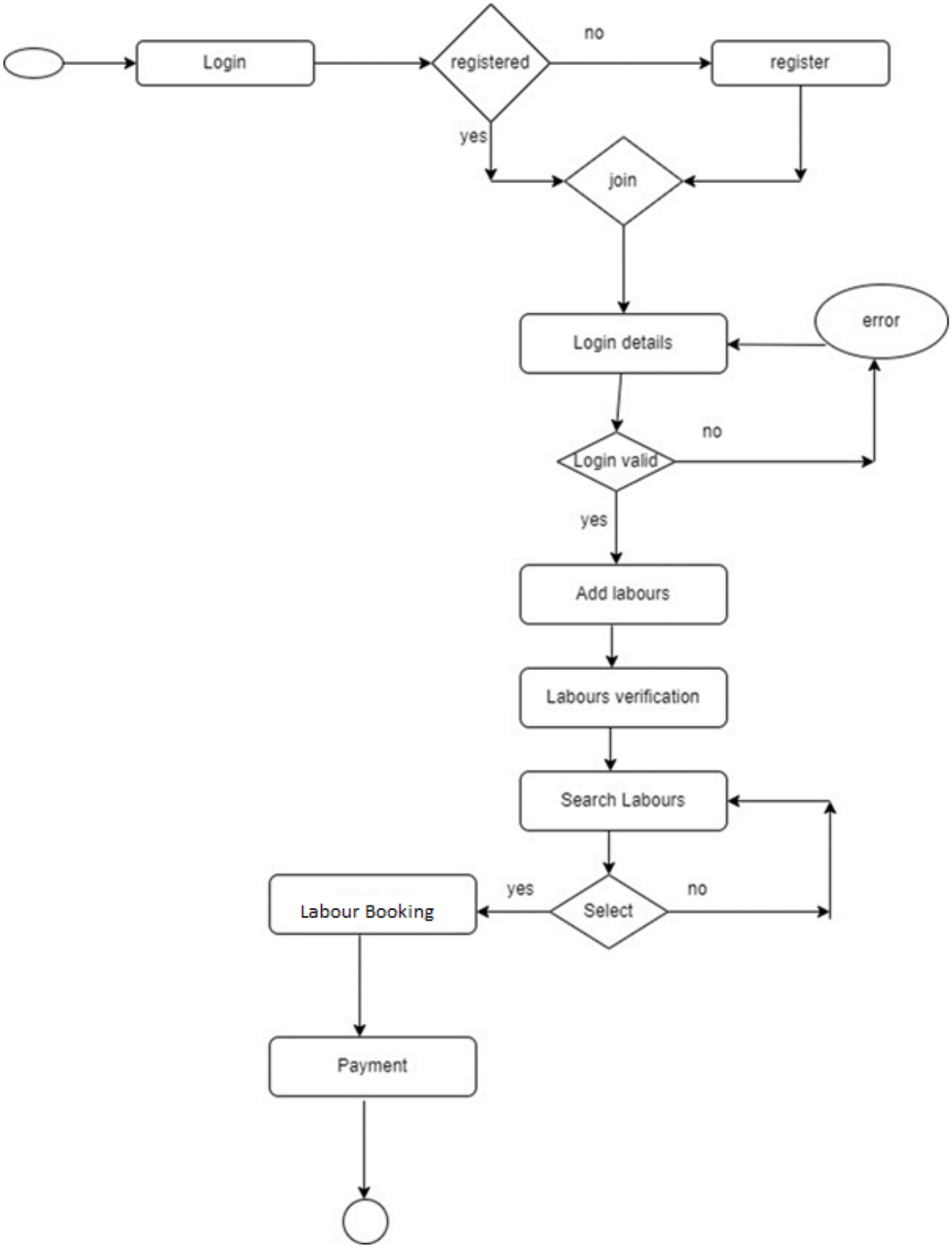


Figure 4.2.3-Activity Diagram

4.2.4 DEPLOYMENT DIAGRAM

A deployment diagram is a sort of UML diagram that depicts a system's execution architecture, containing nodes such as hardware or software execution environments and the middleware that connects them.Deployment diagrams are commonly used to depict a system's physical hardware and software. It allows you to see how the system will be physically installed on the hardware.When compared to other UML diagram types that largely detail the logical components of a system, deployment diagrams help depict the hardware topology of a system.

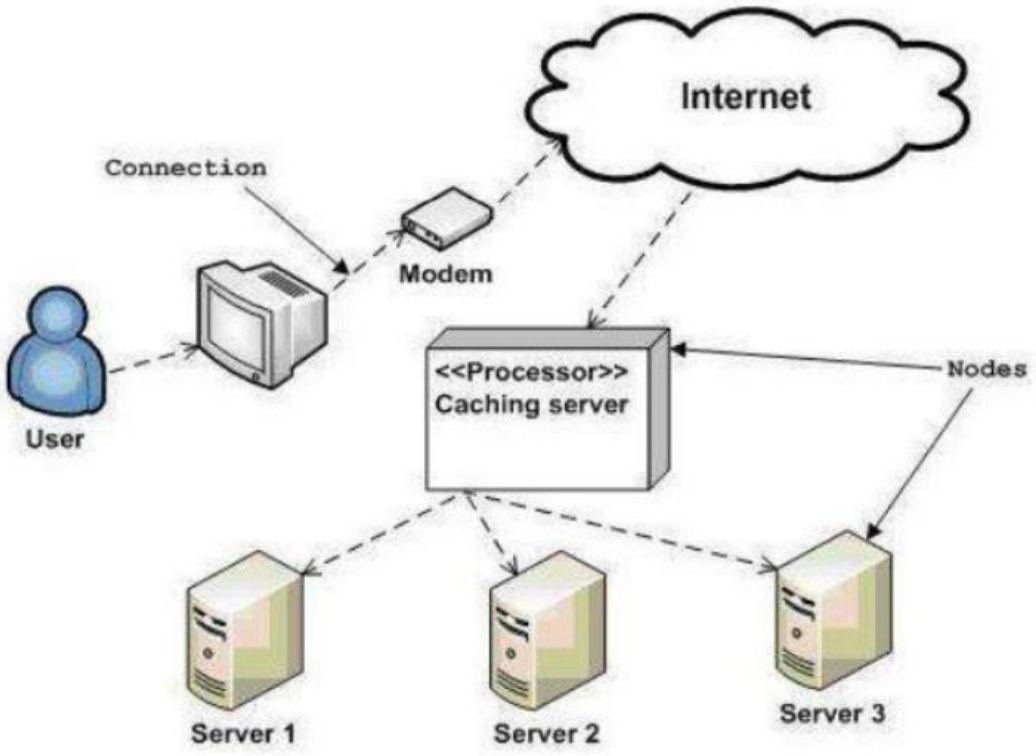


Figure 4.2.4-Deployment Diagram

4.2.5 STATE DIAGRAM

The name of the graphic itself indicates its purpose and other details. It describes the various states of a system component. The states are unique to a system component or entity.A state machine is depicted by a Statechart diagram. A state machine is a machine that specifies multiple states of an entity and controls these states through external or internal events.An activity diagram, as defined in the next chapter, is a subtype of a Statechart diagram. Because a statechart diagram describes the states, it is used to model an object's lifecycle.

Purpose of Statechart Diagrams

One of the five UML diagrams used to depict the dynamic aspect of a system is the statechart diagram. They define multiple states of an entity during its existence, which are modified by events. To model reactive systems, statechart diagrams are important. A reactive system is one that reacts to either external or internal events

The movement of control from one state to another is depicted by a statechart diagram. A state is defined as a condition in which an item exists and changes when an event occurs. The primary goal of a Statechart diagram is to model an object's lifespan from creation to termination.

Statechart diagrams are also used for forward and reverse engineering of a system. However, the main purpose is to model the reactive system.

Following are the main purposes of using Statechart diagrams −

- To model the dynamic aspect of a system.
- To model the life time of a reactive system.
- To describe different states of an object during its life time.
- Define a state machine to model the states of an object.

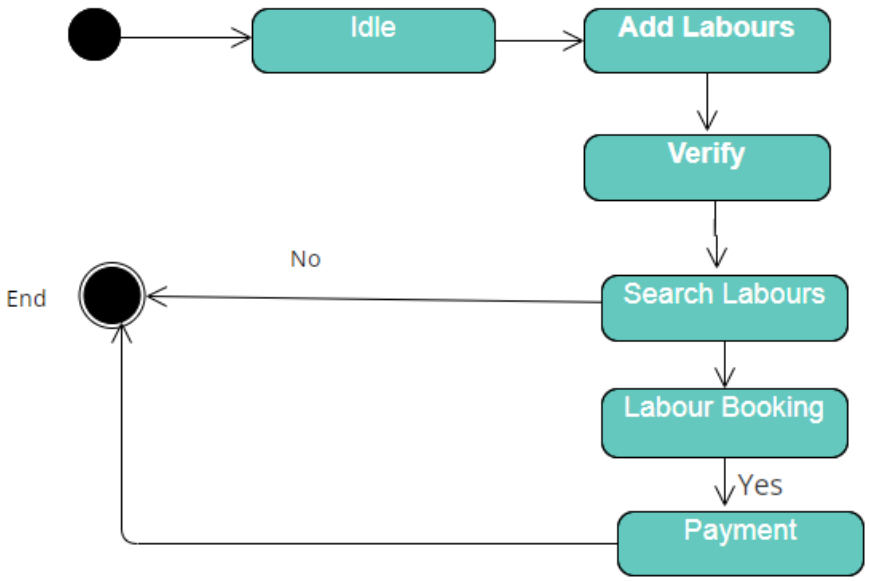


Figure 4.2.5-State Diagram

4.2.6 Class Diagram

A static diagram is a class diagram. It depicts an application's static view. A class diagram is used not only for visualising, describing, and documenting many parts of a system, but also for building executable code for a software programme. A class diagram illustrates a class's properties and actions, as well as the limitations imposed on the system. Because they are the only UML diagrams that can be translated directly to object-oriented languages, class diagrams are frequently utilised in the modelling of object-oriented systems. A class diagram is a visual representation of a collection of classes, interfaces, affiliations, collaborations, and constraints. A structural diagram is another name for it.

Purpose of Class Diagrams

The goal of a class diagram is to represent the static perspective of an application. Class diagrams are the only diagrams that can be directly transferred to object-oriented languages and are thus commonly utilised throughout creation.UML diagrams such as activity diagrams and sequence diagrams can only show the application's sequence flow, however class diagrams are a little different. It is the most widely used UML diagram in the coding world.

The purpose of the class diagram can be summarized as −

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

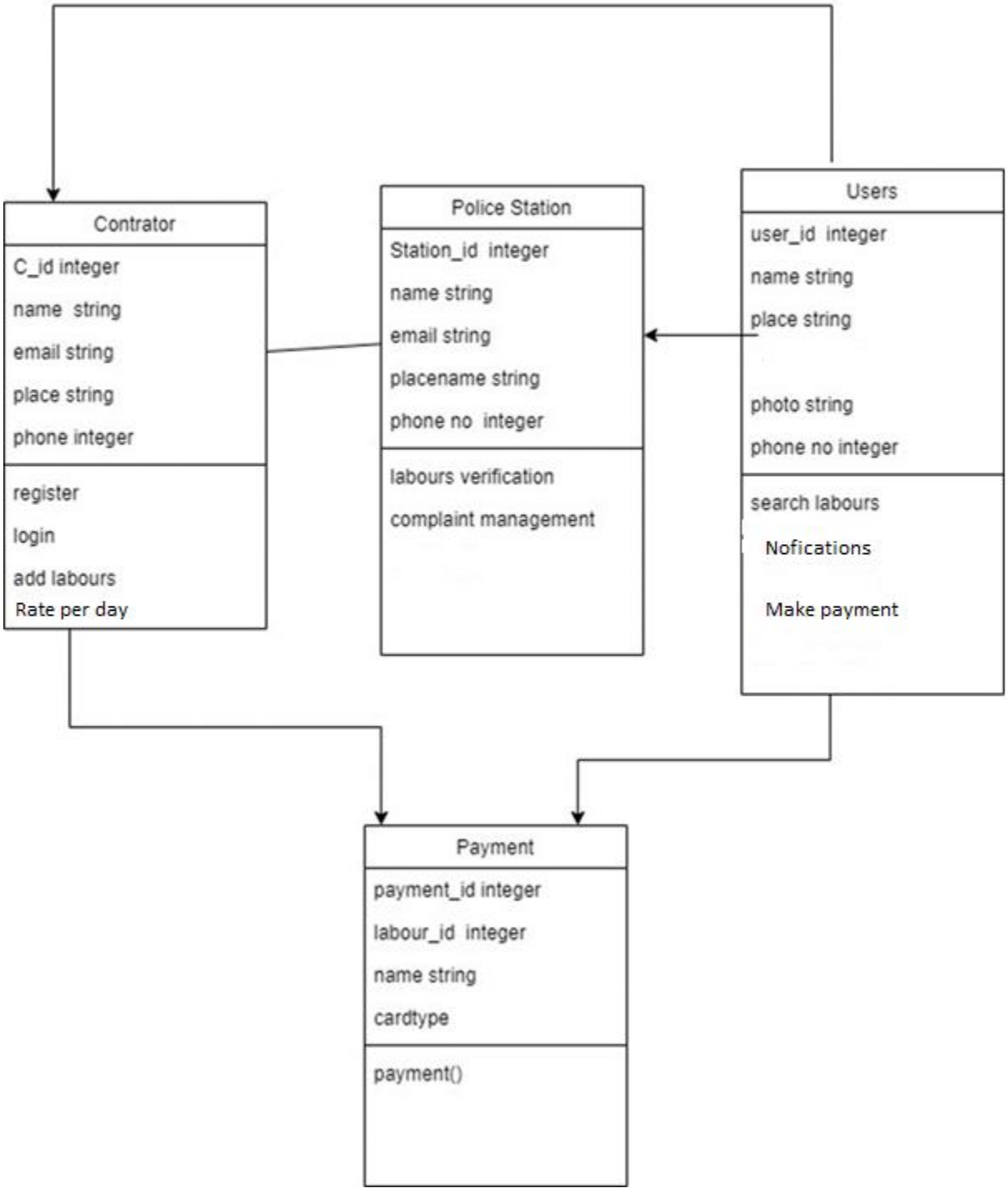
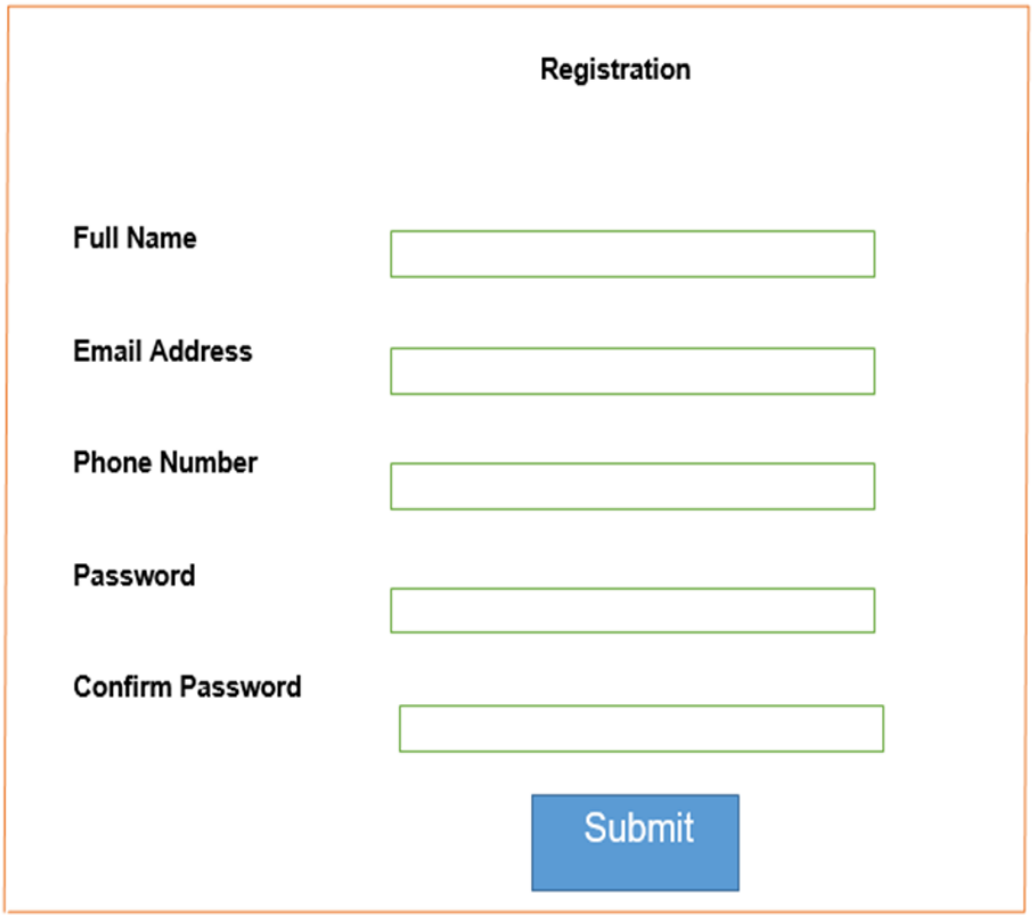


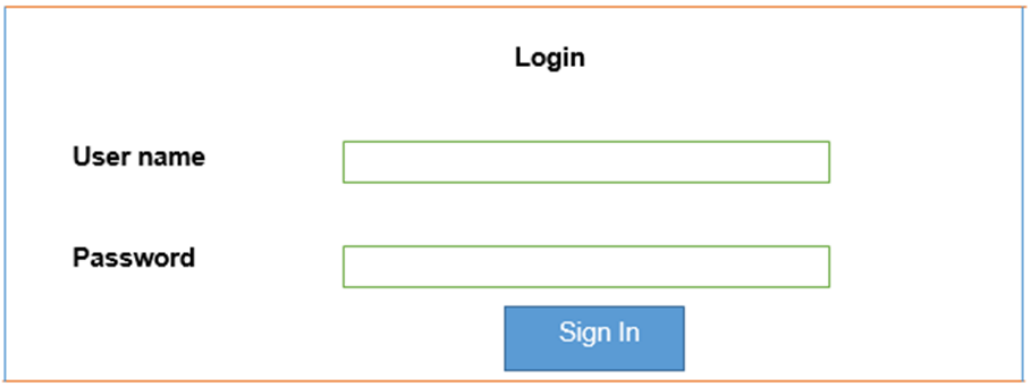
Figure 4.2.6-Class Diagram

4.3 USER INTERFACE DESIGN

4.3.1-INPUT DESIGN

Form Name : User Registration

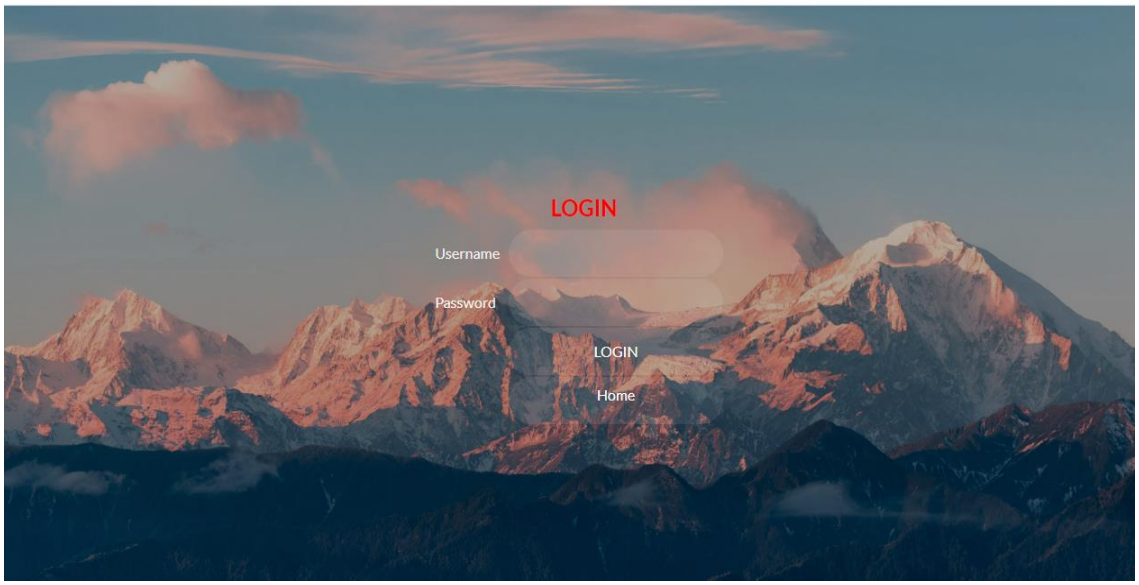




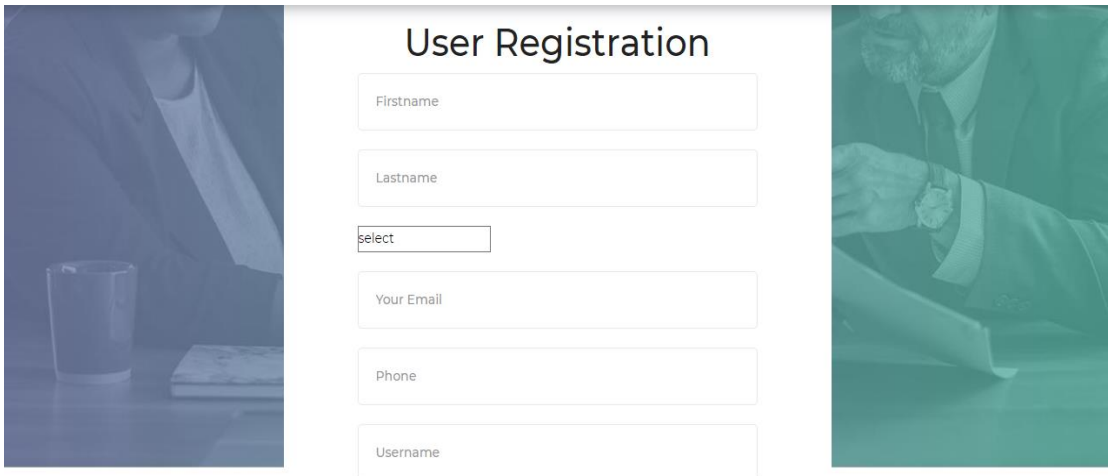
4.3.1.1: User Login

4.3.2 OUTPUT DESIGN

4.3.2.1-User Login



4.3.2.1.1 User Registration



4.6. DATABASE DESIGN

A database is an organised technique for keeping information that allows a user to retrieve stored information in an effective and efficient manner. The objective of any database is to store data, which must be kept secure. The database design process is divided into two stages. The initial stage is to collect user needs and create a database that meets these criteria as clearly as feasible. This is known as Information Level Design, and it is performed independently of any specific DBMS

The second stage is to translate this information level design into a design for the specific DBMS that will be used to construct the system in issue. This process is known as Physical Level Design, and it is concerned with the properties of the DBMS that will be utilised. A database design runs concurrently with a system design. The database's data arrangement aims to meet the two key goals listed below.

- Data Integrity
- Data independence

4.6.1 Relational Database Management System (RDBMS)

A relational model depicts a database as a set of relationships. Each relationship is similar to a table of values or a file of records. A row is referred to as a tuple in formal relational model language, a column header is referred to as an attribute, and the table is referred to as a relation. A relational database is made up of a collection of tables, each with its own name. A row in a story indicates a collection of connected values.

A table is a type of relation. Tuples are the rows of a table. A tuple is a set of n elements that is ordered. Columns are known as characteristics. Every table in the database has relationships set up. This protects the integrity of both the referential and entity relationships. D domains are collections of atomic values. A typical way to establish a domain is to declare a data type from which the domain's data values are pulled. It is also important to give the domain a name to aid in comprehending its values.

Every value in a relation is atomic, which means it cannot be decomposed.

Relationships

- Table relationships are established using Key. The two main keys of prime importance are Primary Key & Foreign Key.
- Entity Integrity and Referential Integrity Relationships can be established with these keys.
- Entity Integrity enforces that no Primary Key can have null values.
- Referential Integrity enforces that no Primary Key can have null values.
- Referential Integrity for each distinct Foreign Key value, there must exist a matching Primary Key value in the same domain. Other key are Super Key and Candidate Keys.

4.6.2 Normalization

Data are put together in the simplest way possible so that future modifications have the least influence on data structures. Normalization is the formal process of arranging data structures in ways that reduce duplication and improve integrity. Normalization is the process of separating duplicate fields and dividing a huge table into smaller ones. It is also used to prevent insertion, deletion, and update errors. In normal form, two notions are used in data modelling: keys and relationships. A key is used to uniquely identify a row in a table. Primary keys and foreign keys are the two sorts of keys. A primary key is a table element or combination of components that is used to identify records from the same table. A foreign key is a key that comes from another country.

As the name indicates, it refers to returning things to their original state. The application developer attempts to produce a reasonable structure of data into correct tables and columns, with names that can be easily connected to the data by the user, through normalisation. Normalization removes recurring data groupings, avoiding data redundancy, which is a significant drain on computer resources. These are some examples:

- ✓ Normalize the data.
- ✓ Choose proper names for the tables and columns.
- ✓ Choose the proper name for the data.

First Normal Form

According to the First Normal Form, an attribute's domain must include only atomic values, and the value of every attribute in a tuple must be a single value from that attribute's domain. To put it another way, 1NF forbids "relations inside relations" or "relations as attribute values within tuples." 1NF only allows single atomic or indivisible values for attribute values. The initial step is to convert the data to First Normal Form. This may be accomplished by separating data into different tables with comparable data types in each table. Each table is assigned a Primary Key or a Foreign Key based on the project's needs. In this case, we create new relations for each non-atomic attribute or nested attribute.

Second Normal Form

According to Second Normal Form, no non-key attribute should be functionally reliant on a component of the main key in relations when the primary key has several attributes. We deconstruct and create a new relation for each partial key and its dependent attributes in this step. Maintain a relationship with the original primary key and any properties that are completely functionally dependent on it. This phase assists in extracting data that is solely dependent on a portion of the key. A connection is said to be in second normal form if and only if it meets all of the first normal form criteria for the main key and all of the relation's non-primary key qualities are entirely reliant on its primary key alone.

Third Normal Form

Relationships should not have a non-key attribute that is functionally determined by another non-key property or a group of non-key attributes, according to Third Normal Form. That is, no transitive reliance on the main key should exist. We deconstruct and establish a relationship that contains non-key qualities that functionally determine other non-key attributes. This step is used to remove everything that is not completely dependant on the Primary Key. A relation is said to be in third normal form if it is solely in second normal form and the non-key qualities of the relation do not depend on each other.

4.6.3 TABLE DESIGN

Tablename:Tbl_Login

Primary Key:L_id

SL.No	Fieldname	Type	Description
1	L_id	Integer(30)	Login id
2	Username	Varchar(40)	Username
3	Password	Varchar(50)	Password
4	Status	Varchar(50)	Status

Tablename:Tbl_Police station

Primary Key:Station_id

SL.No	Fieldname	Type	Description
1	Station_id	Interger(50)	Station id
2	Stationname	Varchar(50)	Stationame
3	Place	Varchar(50)	Place
4	Email	Varchar(50)	Email
5	Phone_no	Varchar(50)	Phone number

-

Tablename:Tbl_Contractor

Primary Key:Contractor_id

Sl.No	Fieldname	Type	Description
1	Contractor_id	Integer(50)	Contractor id
2	Name	Varchar(50)	Contractor name
3	Exp	Varchar(50)	Experience Certificate
4	District	Varchar(50)	District
5	Email	Varchar(50)	Email
6	Phone_no	Varchar(50)	Phone number

Tablename:Tbl_Users

Primary Key:User_id

Sl.No	Fieldname	Type	Description
1	Firstname	Varchar(50)	Firstname
2	Lastname	Varchar(50)	Lastname
3	Place	Varchar(50)	Place
4	Email	Varchar(50)	Email
5	Phone	Varchar(50)	Phone number
6	User_id	Integer(50)	User id

Tablename: Tbl_labours

Primary key:Labour_id

SL.No	Fieldname	Type	Description
1	Labour_id	Integer(50)	Labour id
2	Name	Varchar(50)	Name
3	Place	Varchar(50)	Place
4	Adhar_no	Integer(50)	Adharcard
5	dob	Integer(50)	Date of birth
6	Gender	Varchar(50)	Gender
7	Photo	Varcar(500)	Photo
8	Contractorname	Varchar(50)	Contractorname
9	Stationname	Varchar(50)	Stationname
10	Category	Varchar(50)	Category
11	Experience	Integer(50)	Experience

Tablename:Tbl_labourbooking

Sl.No	Fieldname	Type	Description
1	Book_date	Date(50)	Labour booking date
2	Username	Varchar(50)	Username
3	Category	Varchar(50)	Category
4	Required_date	Date(50)	Requried date
5	Work_details	Varchar(50)	Work details

CHAPTER 5

SYSTEM TESTING

5.1 INTRODUCTION

Software testing is the controlled execution of programme in order to answer the question, "Does the software behave as specified?" The words software testing and verification and validation are sometimes used interchangeably. The inspection or testing of goods, including software, for conformity and consistency with an accompanying specification is known as validation. Software testing is only one type of verification; other methods include reviews, analyses, inspections, and walkthroughs. Validation is the process of ensuring that what has been specified is indeed what the user desired.

Static and dynamic analysis are two more processes that are frequently connected with software testing. Static analysis examines the source code of software for issues and metrics without actually running the code.

Testing is a set of activities that can be planned ahead of time and carried out in a systematic manner. Testing begins with modules and progresses to the integration of the entire computer-based system. Nothing is complete without testing, since testing objectives are critical to the success of the system. There are various guidelines that might serve as testing objectives. They are as follows:

Testing is the process of running a software in order to detect errors.

- A good test case is one that has high possibility of finding an undiscovered error.
- A successful test is one that uncovers an undiscovered error.

If the testing is carried out correctly in accordance with the objectives given above, it will reveal faults in the programme. Testing also shows that the software functions appear to be operating as expected and that the performance requirements appear to have been satisfied.

There are three ways to test program.

- For correctness
- For implementation efficiency
- For computational complexity

Correctness tests are meant to ensure that a software accomplishes precisely what it was planned to do. This is far more complex than it appears, especially for large projects.

5.2 TEST PLAN

A test plan denotes a sequence of intended actions to be taken in order to complete various testing methodologies. The Test Plan serves as a blueprint for the activity to be taken. Software engineers build computer programmes, documentation, and data structures. The software developers are always in charge of evaluating the different parts of the programmes to ensure that they accomplish the purpose for which they were developed. An independent test group (ITG) exists to eliminate the inherent issues involved with allowing the builder to test the finished product. The testing goals should be specified in quantifiable ways. As a result, the mean time to failure, the cost of finding and fixing the flaws, the residual defect density or frequency of occurrence, and test work-hours per regression test should all be specified in the test plan.

The levels of testing include:

- Unit testing
- Integration Testing
- Data validation Testing
- Output Testing

5.2.1 Unit Testing

The smallest unit of software design - the software component or module - is the subject of unit testing. Important control pathways are checked within the module's border using the component level design description as a reference. Unit testing's relative difficulty and uncovered scope are established. Unit testing is white-box in nature, and numerous components may be tested in parallel. The modular interface is evaluated to ensure that data flows correctly into and out of the software unit under test. The local data structure is inspected to guarantee that data stored temporarily retains its integrity across all phases in the execution of an algorithm. To ensure that all statements in a module have been executed at least once, boundary conditions are tested. Finally, there is error handling.

Before starting any additional tests, data flow over a module interface must be tested. All other tests are rendered ineffective if data does not enter and depart appropriately. During the unit test, it is critical to do selective testing of execution pathways. Error circumstances should be expected and error handling pathways should be put up to redirect or cleanly end operations when an error occurs. The final duty of the unit testing process is boundary testing. Software frequently fails at its limits.

In the Sell-Soft System, unit testing was performed by considering each module as a separate entity and testing each one with a broad range of test inputs. Some weaknesses in the underlying logic of the modules were discovered and fixed. Following coding, each module is tested and run independently. All extraneous code was deleted, and all modules were tested to ensure that they functioned properly and produced the intended results.

5.2.2 Integration Testing

Integration testing is a methodical methodology for building the programme structure while also running tests to detect interface issues. The goal is to use unit-tested components and develop a programme structure governed by design. The entire software is tested in its entirety. Correction is difficult since isolating reasons is confounded by the program's enormous scope. Once these flaws are addressed, new ones develop, and the process appears to loop indefinitely. Following unit testing in the system, all modules were merged to test for interface inconsistencies. Furthermore, discrepancies in programme architectures were eliminated, and a single programme structure formed.

5.2.3 Validation Testing or System Testing

This is the last stage of testing. The complete system was tested in its entirety, including all forms, code, modules, and class modules. This type of testing is sometimes referred to as Black Box testing or System testing.

The Black Box testing approach focuses on the software's functional requirements. In other words, Black Box testing allows a software engineer to generate sets of input circumstances that completely exercise all functional requirements for a programme.

Black Box testing looks for defects in the following categories: erroneous or missing functions, interface errors, data structure or external data access errors, performance errors, initialization mistakes, and termination faults.

5.2.4 Output Testing or User Acceptance Testing

The system under consideration has been tested for user approval; it should meet the needs of the company. The software should maintain contact with the prospective system and user while developing and making modifications as needed. This is done in relation to the following points:

- Input Screen Designs,
- Output Screen Designs,

The aforesaid testing is carried out using several types of test data. The preparation of test data is critical in system testing. The system under investigation is tested using the test data that has been prepared. While testing the system, test data mistakes are discovered and fixed using the aforementioned testing processes, and the fixes are also recorded for future use.

5.2.5 Automation Testing

Software and other computer goods are tested automatically to make sure they abide by tight guidelines. In essence, it's a test to ensure that the hardware or software performs exactly as intended. It checks for errors, flaws, and any other problems that might occur throughout the creation of the product. Any time of day can be used to do automation testing. It looks at the software using scripted sequences. It then summarizes what was discovered, and this data can be compared to results from earlier test runs.

Benefits of automated testing

- Capabilities for detailed reporting - Automated testing uses well created test cases for diverse scenarios. These planned sequences can cover a lot of ground and produce in-depth reports that are simply impossible for a human to produce.
- Improved bug detection - Finding bugs and other flaws in a product is one of the key goals of testing it. This procedure can be made simpler with automation testing. Additionally, it can examine a greater test coverage than perhaps people can.
- Simplifies testing - Most SaaS and IT organizations routinely include testing in their daily operations. The key is to keep things as basic as you can. Automation has a lot of advantages. The test scripts can be reused when automating test tools.
- Accelerates the testing procedure - Machines and automated technology operate more quickly than people. This is why we employ them, along with increased accuracy. Your software development cycles are subsequently shortened by this.
- Lessens the requirement for human supervision - Tests may be conducted at any time of day, including overnight. Additionally, this can lessen the possibility of human error when it is carried out mechanically.

5.2.6 Selenium Testing

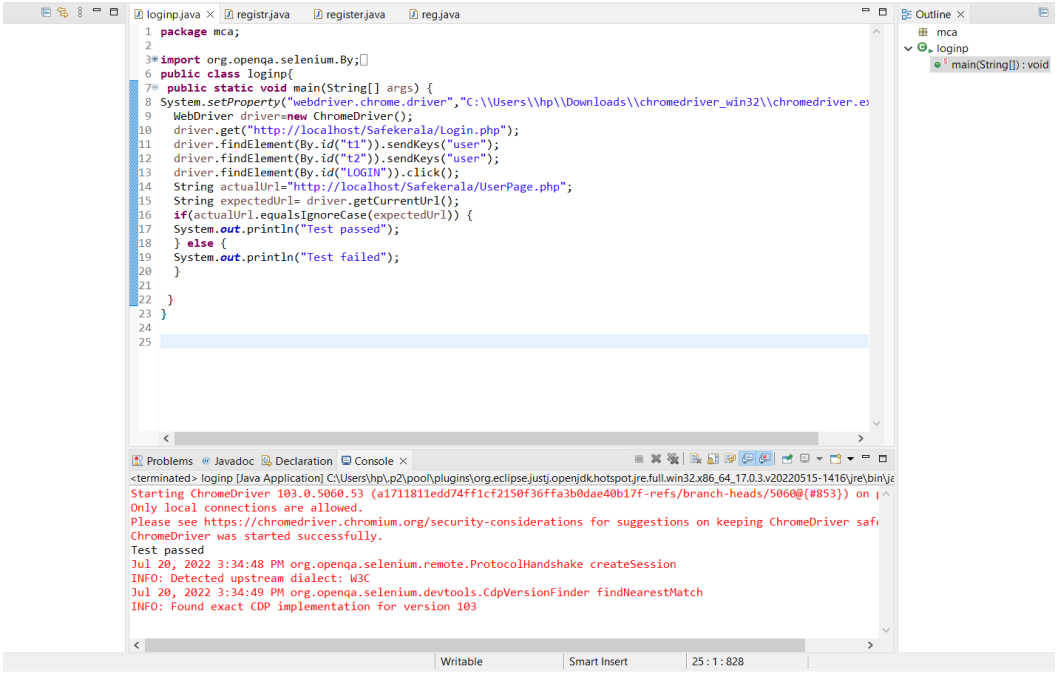
Selenium is a free and open-source technology for automating web browsers. It provides a unified interface for writing test scripts in a variety of computer languages, including Ruby, Java, NodeJS, PHP, Perl, Python, and C#. The Selenium testing tool is used to automate web application tests across browsers. It is used to assure high-quality online applications, whether responsive, progressive, or consistent. Selenium is a free and open source utility.

Test cases for a Login Page

Project Name:Safekerala					
Login Test Case					
Test Case ID: Fun_1			**Test Designed By: Reenu Joseph**		
Test Priority (Low/Medium/High): High			**Test Designed Date:**20-07-2022		
Module Name: Login Screen			**Test Executed By: Ms.Nimmy Francis**		
Test Title: Verify login with valid username and password			**Test Execution Date:** 20-07-2022		
Description: Test the Login Page					
Pre-Condition: User has valid username and password					
Step	**Test Step**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/Fail)**
1	Navigation to Login Page		Login Page should be display ed	Login page displayed	Pass
2	Provide Valid username	Username : user	User should d be able to Login	User Logged in and navigated to User Dashboard	Pass
3	Provide Valid Password	Password:user			
4	Click on Sign In button				
5	Provide Invalid username orpassword	Username:user Password: user	User should not be able to Login	Message for enter valid email id or password displayed	Pass
6	Provide Null username or Password	Username : null Password: null			
7	Click on Sign In button				

Code package

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
public class loginp{
 public static void main(String[] args) {
System.setProperty("webdriver.chrome.driver","C:\\Users\\hp\\Downloads\\chromedrive
r_win32\\chromedriver.exe" );
  WebDriver driver=new ChromeDriver();
  driver.get("http://localhost/Safekerala/Login.php");
  driver.findElement(By.id("t1")).sendKeys("user");
  driver.findElement(By.id("t2")).sendKeys("user");
  driver.findElement(By.id("LOGIN")).click();
  String actualUrl="http://localhost/Safekerala/UserPage.php";
  String expectedUrl= driver.getCurrentUrl();
  if(actualUrl.equalsIgnoreCase(expectedUrl)) {
  System.out.println("Test passed");
  } else {
  System.out.println("Test failed");
  }

 }
}
```

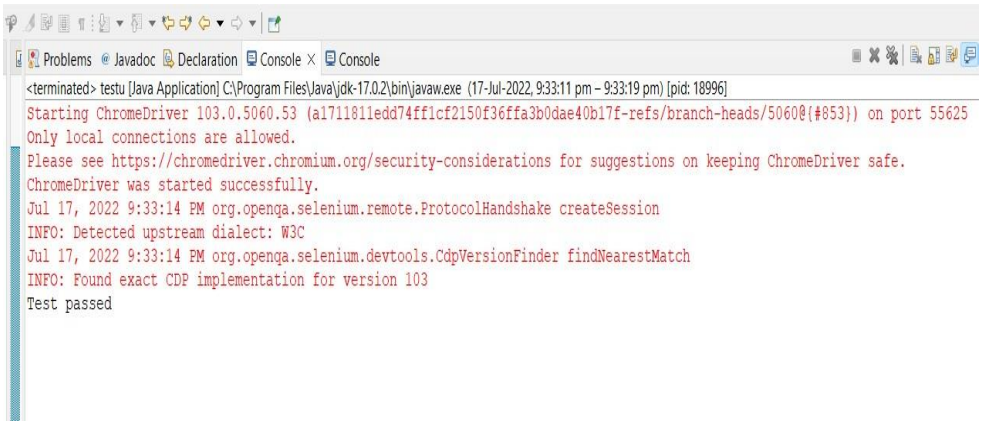


Test cases for User Registration

Project Name: Safekerala					
			Updation Test Case		
Test Case ID: registration			**Test Designed By: Reenu Joseph**		
Test Priority (Low/Medium/High): High			**Test Designed Date:** 20-07-2022		
Module Name: Login Screen			**Test Executed By: Ms. Nimmy Francis**		
Test Title: User Registration Details			**Test Execution Date:** 20-07-2022		
Description: Register to system and Registration is completed then login , if someerror occurs, testwill fail					
Pre-Condition: User has valid user name and password					
Step	**Test Step**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/Fail)**
1	Navigation to Register Page		Register Page shouldbe	Registrtion page displayed	Pass
2	Provide Valid Registration details	User Name: manu	User should be ableto Register	User registrion Completed after go to the login page	Pass
3					
4	Click on Login button				
5	Provide profile details	Input profile details	User will be redirected to Login page	Use will be redirected to Login page	Pass
7	Click on register button				
8	Provide invalid information	Input invalid profile details.	User will be stay in register page	User will be stay on that page showing error message	Pass
9	Click on register button				

Code package

```
package test;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;public class testu {
public static void main(String[] args) {

System.setProperty("webdriver.chrome.driver","C:\\Users\\nibin\\Downloads\\chrome
driver_win32\\chromedriver.exe" );
 WebDriver driver=new ChromeDriver();
 driver.get("http://localhost/hostel/login/register.php");
 driver.findElement(By.id("name")).sendKeys("manu");
 driver.findElement(By.id("email")).sendKeys("manujames96@gmail.com");
 driver.findElement(By.id("address")).sendKeys("pala");
 driver.findElement(By.id("phone")).sendKeys("8089388381");
 driver.findElement(By.id("dob")).sendKeys("27-05-1996");
 driver.findElement(By.id("username")).sendKeys("manu");
 driver.findElement(By.id("pass")).sendKeys("manu");
 driver.findElement(By.id("re_pass")).sendKeys("manu");
 driver.findElement(By.id("signup")).click();
 String actualUrl="http://localhost/hostel/login/login.php?e=1";String expectedUrl=
 driver.getCurrentUrl(); if(actualUrl.equalsIgnoreCase(expectedUrl)) {
 System.out.println("Test passed");
 } else {
 System.out.println("Test failed");
 }

 }
}
```



CHAPTER 6

IMPLEMENTATION

6.1 INTRODUCTION

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

At this stage the main work load, the greatest upheaval and the major impact on the existing system shifts to the user department. If the implementation is not carefully planned or controlled, it can create chaos and confusion.

Implementation includes all those activities that take place to convert from the existing system to the new system. The new system may be a totally new, replacing an existing manual or automated system or it may be a modification to an existing system. Proper implementation is essential to provide a reliable system to meet organization requirements. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to the new system. The system can be implemented only after through testing is done and if it is found to be working according to the specifications. The system personnel check the feasibility of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required to implement the three main aspects: education and training, system testing and changeover.

The implementation state involves the following tasks:

- Careful planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.

6.2 IMPLEMENTATION PROCEDURES

The final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system, is referred to as software implementation.Many organisations will commission the software development project by someone who will not be operating it. Before proceeding to see the system, the user should be aware that in order to view the results, the server software must be operating on the server. The real procedure will not take place if the server object is not up and running on the server.

6.2.1 User Training

User training is intended to prepare the user for system testing and conversion. To accomplish the goals and advantages anticipated from a computer-based system, the people who will be participating must be confident in their position in the new system. The demand for training grows as the system becomes more sophisticated. User training teaches the user how to enter data, reply to error signals, query the database, and invoke routines that generate reports and do other important operations.

6.2.2.Training on the Application Software

After receiving the essential fundamental computer awareness training, the user will need to be instructed on the new application software. This will provide the fundamental philosophy of the new system's usage, such as the screen flow, screen design, kind of help on the screen, type of mistakes while entering data, the associated validation check at each input, and methods to rectify the date entered. It should then cover the knowledge required by the specific user/group to operate the system or a portion of the system while imparting programme training on the application. This training may change amongst user groups and at different levels of organisation.

6.2.3.System Maintenance

Maintenance is the enigma of system development. When a software product is in the maintenance phase of its life cycle, it is performing useful work. After a system has been effectively established, it must be properly maintained. System upkeep is an essential part of the software development life cycle. System maintenance's goal is to make the system more adaptable to changes in the system environment. All in all,

software maintenance requires far more than "Finding Mistakes." Maintenance is the enigma of system development. When a software product is in the maintenance phase of its life cycle, it is performing useful work. After a system has been effectively established, it must be properly maintained. System upkeep is an essential part of the software development life cycle. System maintenance's goal is to make the system more adaptable to changes in the system environment. All in all, software maintenance requires far more than "Finding Mistakes."

6.2.4 Hosting

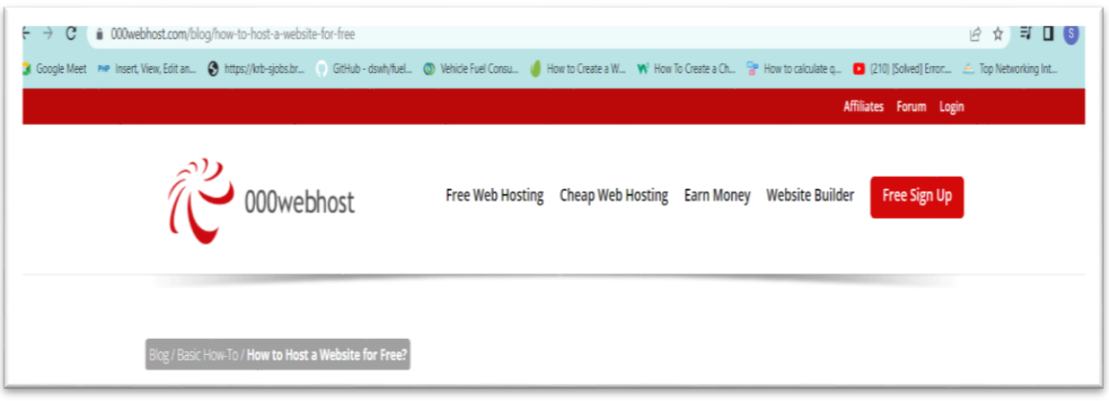
A web hosting service is necessary if you wish to publish your website online. A web hosting service enables users to publish a website or web pages online. Servers are specialised computers that house or save website data. The user must enter the website address (Domain Address) in the address bar of the browser in order to access any website. After that, the browser will establish a connection between the user and the server and display the webpage.

000Webhost

The capabilities and features of free web hosting are frequently so constrained that customers must pay to acquire what they require. A website builder, WordPress support, and the absence of adverts are just a few of the useful features offered by 000webhost, a free website hosting service. Users can switch to a paid plan to gain access to even more services and assistance, but according to our reviews, 000webhost is the finest free web hosting option for people who are really trying to stretch their money.

Steps for hosting website in 000Webhost

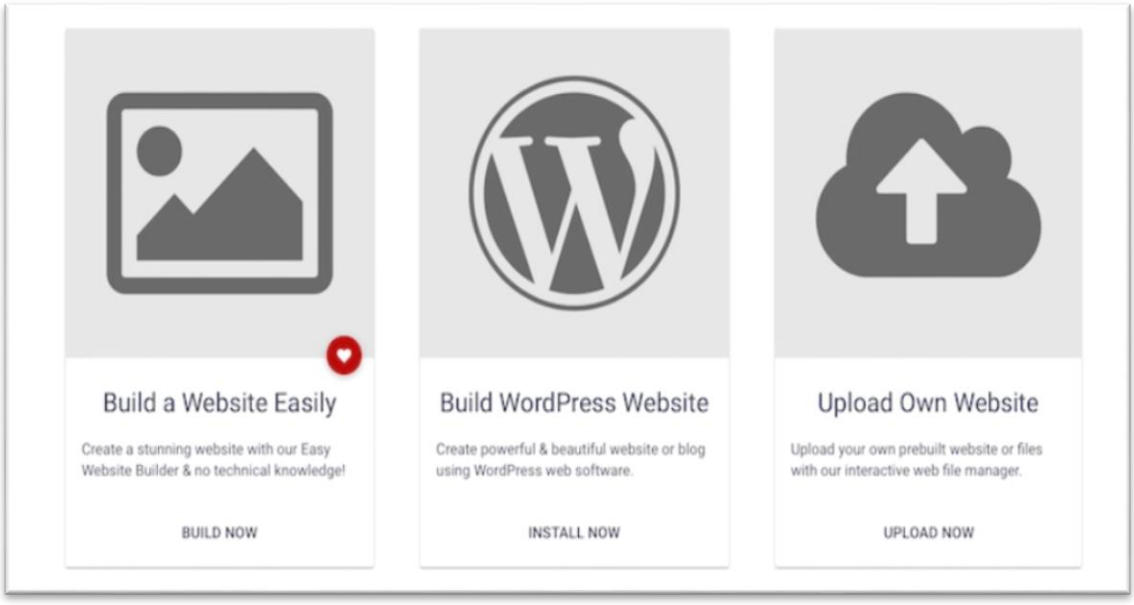
Step 1 : Visit 000webhost.com.



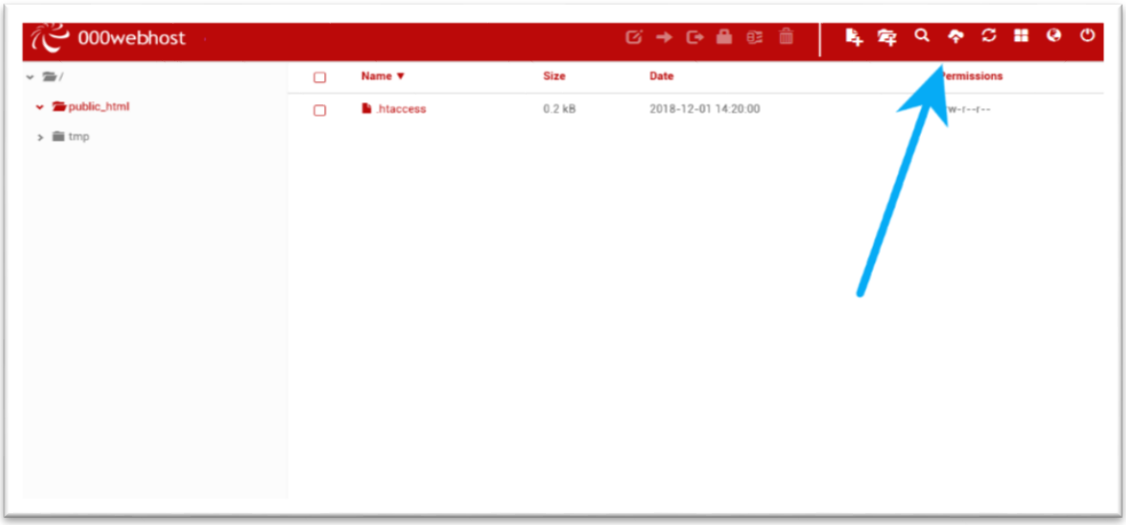
Step 2 : Complete all required fields to create an account.

Step 3 :To verify your hosting account, open your email inbox and click the verification link. You can access your hosting account after confirming your email address.

Step 4 : You recieve a form to complete with your website username and password. The website will now be added to your list of 000webhost sites. Then choose the Upload Now button fixed beneath the Own Website button.

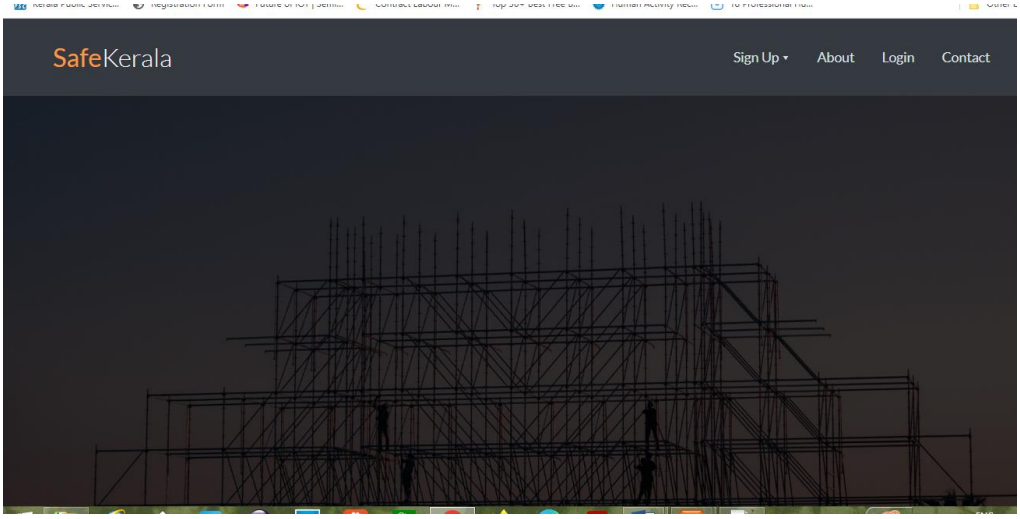


Step 5 : To upload an HTML or PHP file from your device, click the upload button in the top right corner.



Step 6 :Click the Database Manager in the left sidebar after that, then upload the database.

Step 7 : The connection file should be updated with the specified DB Name, DB User, and Password. Finally, hosting of your website was completed.



CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

The existing system's operating technology is outdated, and there is no use of regularly utilised technologies such as the internet or digital money.The suggested system allows users to book labours online and access all details. First of all contractor adds labour details and after police verification verified labours can be viewed by users. Finally user can search for labours and book them. An advance payment is also done.

7.2 FUTURE SCOPE

This application can be easily implemented and there is scope for future development of this project. We can add new features as and when we require. Enhancements can be done in an efficient manner. With subsequent modification in an efficient manner. With subsequent modification establishment an minimumchange, we can even update the same system. As a result, the project is adaptable and can always be upgraded with newer features.

We may be able to hire workers from outside Kerala in the future. By tracking their location we can find to which station they belongs to.

CHAPTER 8

BIBLIOGRAPHY

REFERENCES:

- Gary B. Shelly, Harry J. Rosenblatt, "*System Analysis and Design*", 2009.
- Roger S Pressman, "*Software Engineering*", 1994.
- PankajJalote, "So*ftware engineering*: a precise approach", 2006.
- James lee and Brent ware Addison, "Open source web development with LAMP", 2003
- IEEE Std 1016 Recommended Practice for Software Design Descriptions.

WEBSITES:

- www.w3schools.com
- www.jquery.com
- http://homepages.dcc.ufmg.br/~rodolfo/es-1-03/IEEE-Std-830-1998.pdf
- www.agilemodeling.com/artifacts/useCaseDiagram.html

CHAPTER 9

APPENDIX

9.1 SAMPLE CODE

Login.php

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>Untitled Document</title>
</head>

 <?php
$conn=mysqli_connect("localhost","root","","SafeKerala");
session_start();
 $invalid="";
 if(isset($_POST["LOGIN"]))
  {

    $uname=$_POST['t1'];
   $pword=$_POST['t2'];
   $user1=mysqli_query($conn,"select * from login where Username='$uname' and
Password='$pword'");
   $user=mysqli_fetch_array($user1);

   if(!empty($user[1]))
   {
            $L_id=$user[0];
     $Status=$user[3];
            $sn="";
            if ($Status=='S')
            {
                    $q=mysqli_query($conn,"select Stationname from PSreg where
Station_id='$L_id'");
                    $result=mysqli_fetch_array($q);
                    $sn=$result[0];

            }


     $_SESSION['uname']=$uname;
     $_SESSION['pword']=$pword;
            $_SESSION['L_id']=$L_id;
     $_SESSION['Status']=$Status;
            $_SESSION['sn']=$sn;


    if($Status=="A")
            {
             header('location:Adminpage.php');
            }
```

```
        else if($Status=="U")
         {
      $q=mysqli_query($conn,"select firstname from userreg where User_id='$L_id'");
                        $result=mysqli_fetch_array($q);
                        $sn=$result[0];
                        $_SESSION['sn']=$sn;
                header('location:UserPage.php');
               }
               else if( $Status=="NC")
        {
                 echo "<script>alert('Wait application under Process')</script>";
               }
               else if($Status=="R")
        {
                 echo"<script>alert('Your Application rejected')</script>";
               }
               else if($Status=="S")
        {
                 header('location:PSPage.php');
               }
               else if($Status=="C")
        {
      $q=mysqli_query($conn,"select Name from conreg where Contractor_id='$L_id'");
                        $result=mysqli_fetch_array($q);
                        $sn=$result[0];
                        echo $sn;
                        $_SESSION['coname']=$sn;
                header('location:ConPage.php');
               }
      }
      else
        {
               $invalid="Not Found..... ";
               ?>
               <script>alert('Invalid User')</script>
               <?php

       }
  }

  ?>



    <head>
     <title>Login</title>
      <meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

  <link href="https://fonts.googleapis.com/css?family=Lato:300,400,700&display=swap"
  rel="stylesheet">

  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-
  awesome.min.css">
```

```
    <link rel="stylesheet" href="css/style.css">

    </head>
    <body class="img js-fullheight" style="background-image: url(images/bg.jpg);">
    <section class="ftco-section">
            <div class="container">
                    <div class="row justify-content-center">
                            <div class="col-md-6 text-center mb-5">

                            </div>
                    </div>
                    <div class="row justify-content-center">
                            <div class="col-md-6 col-lg-4">
                                    <div class="login-wrap p-0">
<body onLoad="noBack()">
<form name="form1" method="post" action="">
<div style="background-image:url(img/login1.jpg); background-repeat:no-repeat;">
   <table width="100%"   border="0">
     <tr>
       <td colspan="6"> </td>
     </tr>
     <tr>
       <td colspan="6"><div align="center"><strong><font color="#FF0000" size="5">LOGIN
          </font></strong></div></td>
     </tr>

      <tr>
       <td> </td>
       <td>Username </td>
       <td><input name="t1" type="text" id="t1" class="form-control"></td>
       <td> </td>
       <td> </td>
       <td> </td>
     </tr>
     <tr>
       <td> </td>
       <td>Password</td>
       <td><input name="t2" type="password" id="t2" class="form-control"></td>
       <td> </td>
       <td> </td>
       <td> </td>
     </tr>

     <tr>
       <td> </td>
       <td> </td>
       <td><input name="LOGIN" type="submit" id="LOGIN" value="LOGIN" class="form-
control"></td>

    </table>
  </div>
</form>

</div>
 <td> </td>
```

```
        </div>
        </div>
        </div>
        </section>
  <script src="js/jquery.min.js"></script>
       <script src="js/popper.js"></script>
       <script src="js/bootstrap.min.js"></script>
          <script src="js/main.js"></script>

        </body>
      </html>
      </body>
      </html>
```

Registration.php

```
<!DOCTYPE html>
<html>
<head>
  <title> User Registration </title>
</head>
<?php
include "commonhead.php";

$conn=mysqli_connect("localhost","root","","SafeKerala");
$temp=1;
$errfirstname=$errlastname=$errPlace=$errEmail=$errPhone=$errUsername=$errpassword="
";
 if(isset($_POST["Submit"]))
{
  $mfirstname=$_POST['firstname'];
  $mlastname=$_POST['lastname'];
  $mPlace=$_POST['Place'];
  $mEmail=$_POST['Email'];
  $mPhone=$_POST['Phone'];
  $mUsername=$_POST['Username'];
  $mpassword=$_POST['password'];
  if(empty($mfirstname))
   {
    $temp=0;
    $errfirstname="Name is blank.....";
   }
  if(empty($mlastname))
   {
    $temp=0;
    $errlastname="Name is blank.....";
   }
   if(empty($mPlace))
   {
    $temp=0;
```

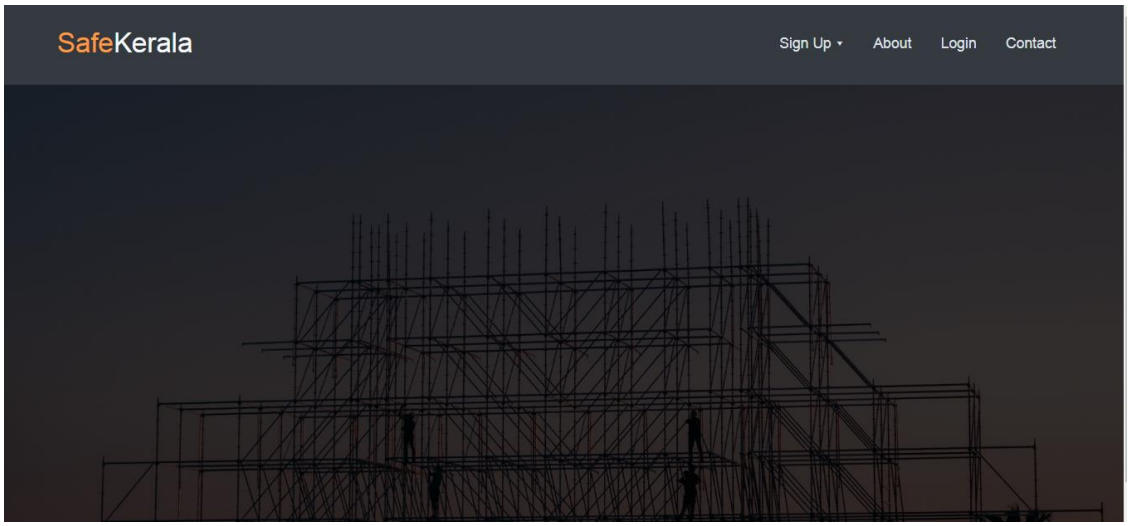
```
 $errPlace="Place is blank.....";
}
if(empty($mEmail))
{
 $temp=0;
 $errEmail="email  is blank.....";
}
if(empty($mPhone))
{
 $temp=0;
 $errUsername="Username is blank.....";
}
if(empty($mpassword))
{
 $temp=0;
 $errpassword="Password is blank.....";
}

if(!filter_var($mEmail,FILTER_VALIDATE_EMAIL))
{
 $temp=0;
 $errEmail="Not a valid email .....";
}
if(!preg_match('/^[0-9]{10}$/',$mPhone))
{
 $temp=0;
 $errPhone="Numbers only and must 10 digits.....";
}

 if ($temp==1)
{
$autonum=mysqli_query($conn,"select max(L_id)+1 from Login");
$res=mysqli_fetch_array($autonum);
if($res[0]==null)
        $mx=1;
else
        $mx=$res[0];

$res=mysqli_query($conn,"select * from Login where Username='$mUsername'");
$res1=mysqli_fetch_array($res);
if( !empty($res1[0]))
{
        ?>

        <script> alert("Duplication")</script>
        <?php

}
else
{
```

```
  $S1="insert into Login (L_id,Username,Password,Status)
values('$mx','$mUsername','$mpassword','U')";
  $n=mysqli_query($conn,$S1);
  $S2="insert into userreg(firstname,lastname,Place,Email,Phone,User_id)
values('$mfirstname','$mlastname','$mPlace','$mEmail','$mPhone','$mx')";
  $n=mysqli_query($conn,$S2);

  if ($n>0)
          ?>
          <script>alert("Registration success ")
  </script>
  <?php
  }

}
}
  ?>
          <head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>SafeKerala</title>
        <div class="container">
          <div class="signup-content">
          <form method="post">
          <center><h1>User Registration</h1></center>
          <table align="center" border="5"  bgcolor="">
                    <div class="form-group">
<input type="text" class="form-input" name="firstname" placeholder="Firstname"/>
          <?php echo $errfirstname;?>
          </div>
          <div class="form-group">
                <input type="text" class="form-input" name="lastname"
placeholder="Lastname"/>
                <?php echo $errlastname;?>
          </div>


          <div class="form-group">
                <select name="Place" class="select-dropdown">
              <option value="">select</option>

                <option value="Thiruvananthapuram">Thiruvananthapuram</option>
                <option value="Kozhikode">Kozhikode</option>
                <option value="Thissur">Thissur</option>
                      <option value="Kollam">Kollam</option>
                      <option value="Malappuram">Malappuram</option>
                      <option value="Eranakulam">Eranakulam</option>
```

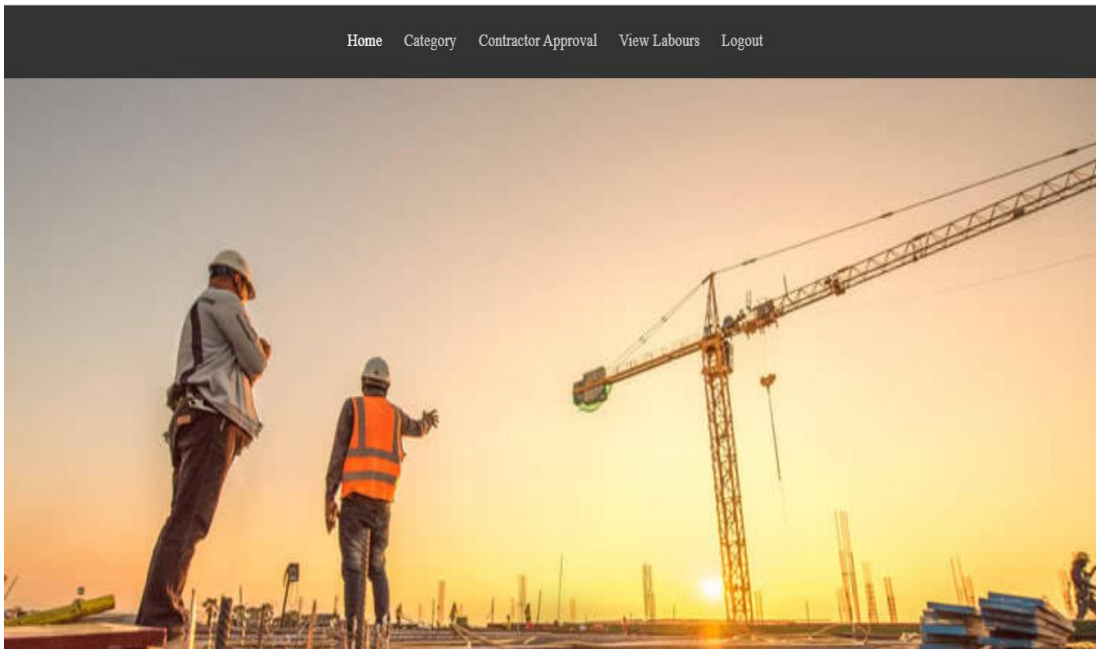
```
                                <option value="Kasaragod">Kasaragod</option>
                                <option value="Alappuzha">Alappuzha</option>
                                <option value="Palakkad">Palakkad</option>
                                <option value="Kottayam">Kottayam</option>
                                <option value="Pathanamthitta">Pathanamthitta</option>
                        <option  value="Idukki">Idukki</option>
                                <option  value="Wayanad">Wayanad</option>
                                 </select>
                        <?php echo $errPlace;?>
                </div>
                <div class="form-group">
                        <input type="Email" class="form-input" name="Email" id="Email"
    placeholder="Your Email"/>
                         <?php echo $errEmail;?>

                </div>
                <div class="form-group">
                        <input type="Phone" class="form-input" name="Phone"
    placeholder="Phone"/>
                        <?php echo $errPhone;?>
                </div>


                <div class="form-group">
                        <input type="text" class="form-input" name="Username"
    placeholder="Username"/>
                <?php echo $errUsername;?>
                </div>
                <div class="form-group">
                        <input type="Password" class="form-input" name="password"
    placeholder="Password"/>
                        <?php echo $errpassword;?>
                </div>
                <div class="form-group">
<input type="Submit" name="Submit" id="Submit" class="form-Submit" value="Sign up" />
                </div>
          </form>

           <script src="reg/vendor/jquery/jquery.min.js"></script>
           <script src="reg/js/main.js"></script>
    </body><!-- This templates was made by Colorlib (https://colorlib.com)
    </html>
                </table>
          </form>

    </html>
```

9.2 SCREENSHOTS

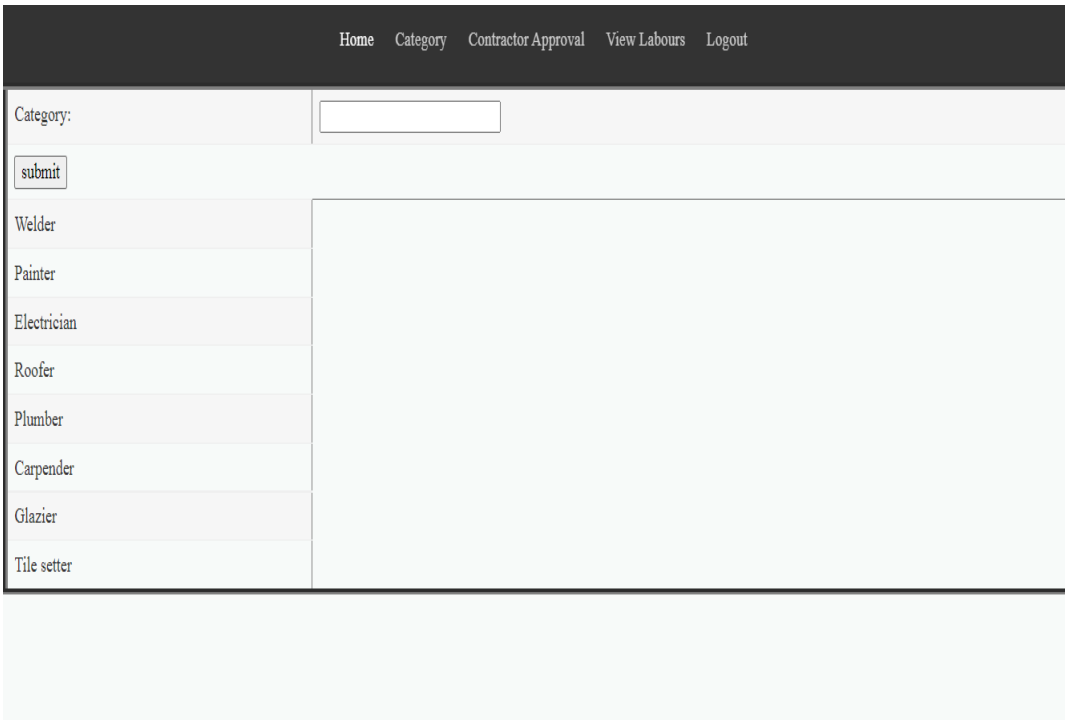
9.2.1- Home page



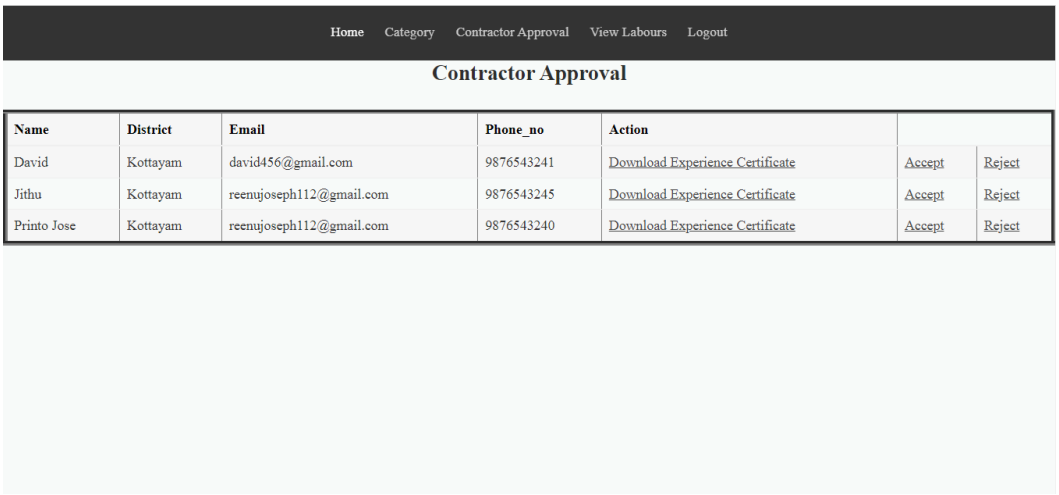
9.2.2- Admin homepage



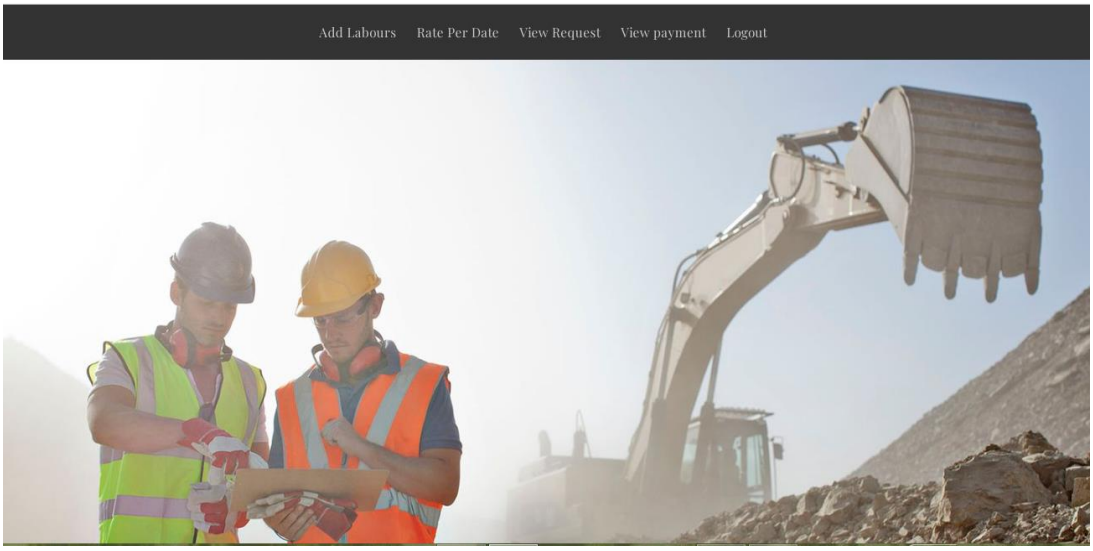
Admin:- Add Category



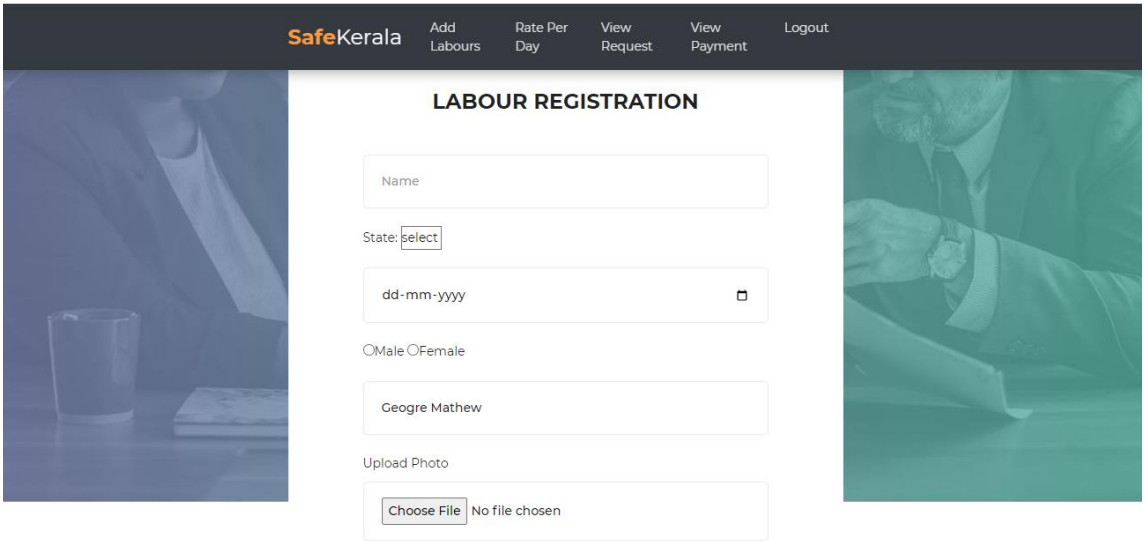
Admin: Contractor Approval



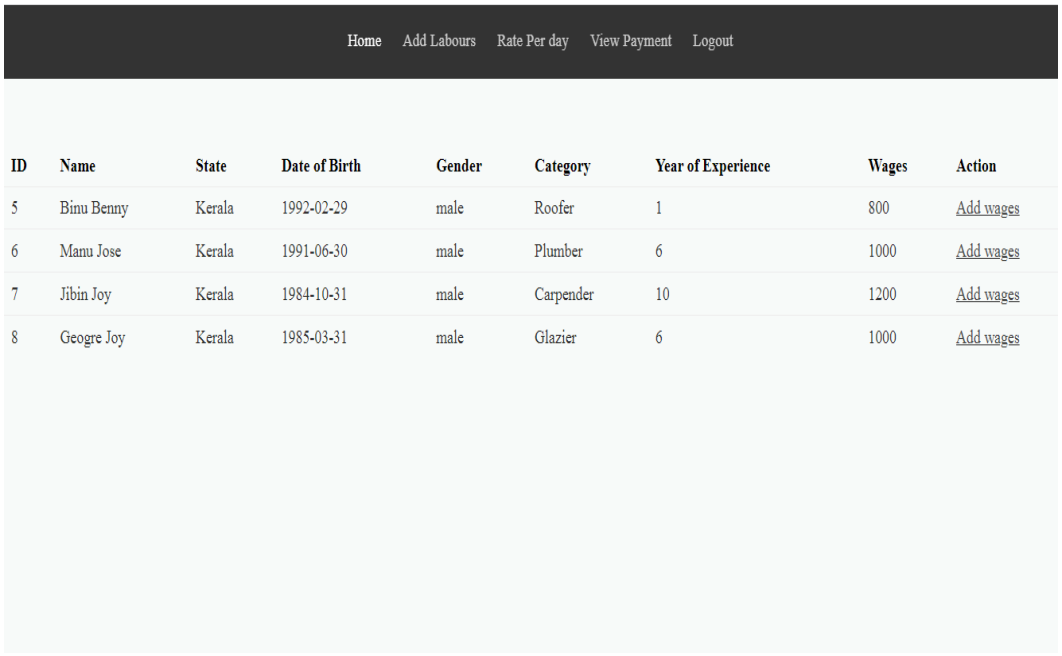
9.2.3-Contractor homepage



Contractor: Add Labours

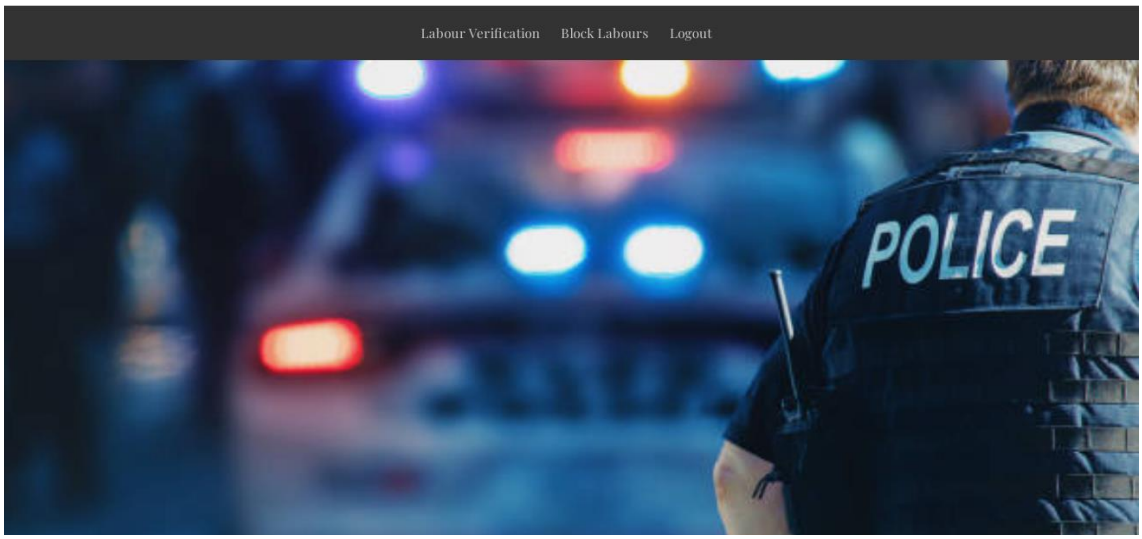


Contractor: Rate per day

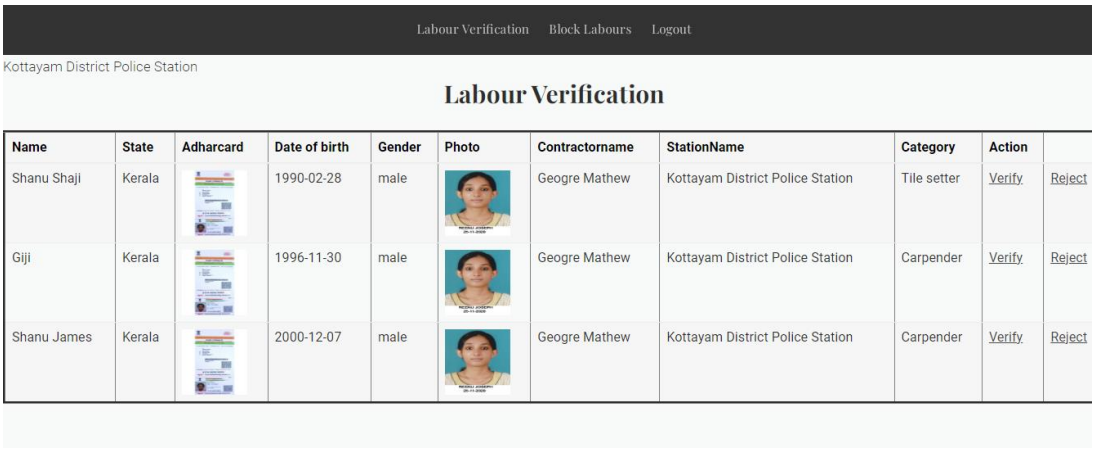


ID	Name	State	Date of Birth	Gender	Category	Year of Experience	Wages	Action
5	Binu Benny	Kerala	1992-02-29	male	Roofer	1	800	Add wages
6	Manu Jose	Kerala	1991-06-30	male	Plumber	6	1000	Add wages
7	Jibin Joy	Kerala	1984-10-31	male	Carpender	10	1200	Add wages
8	Geogre Joy	Kerala	1985-03-31	male	Glazier	6	1000	Add wages

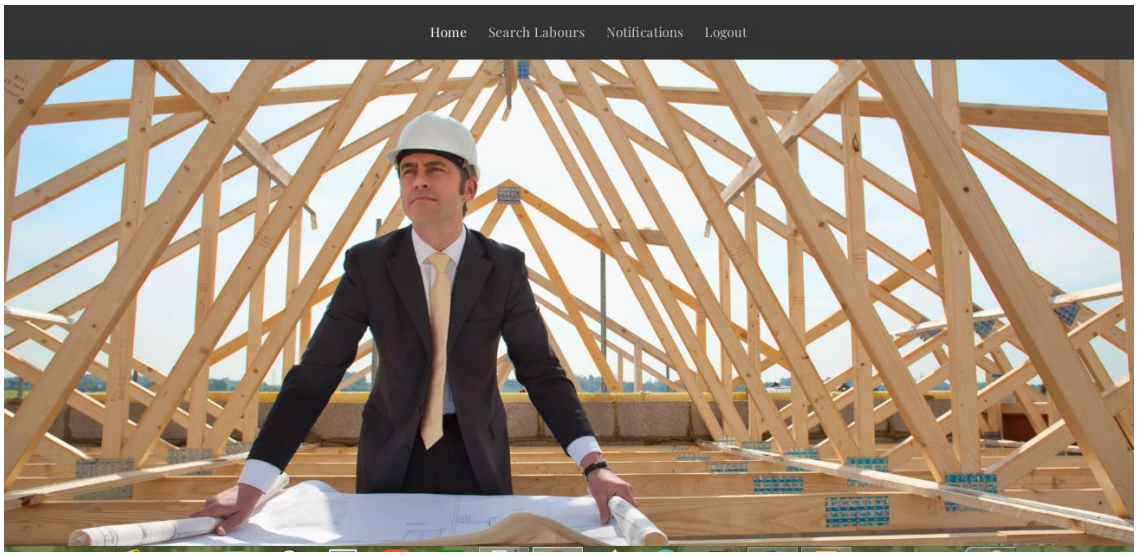
9.2.4-Police station homepage



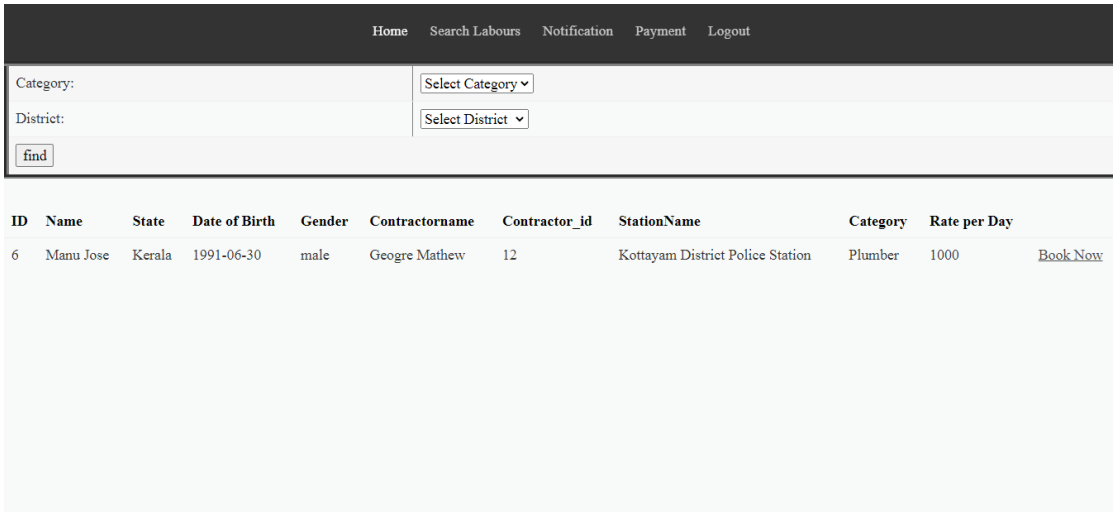
Police:Labour Verification



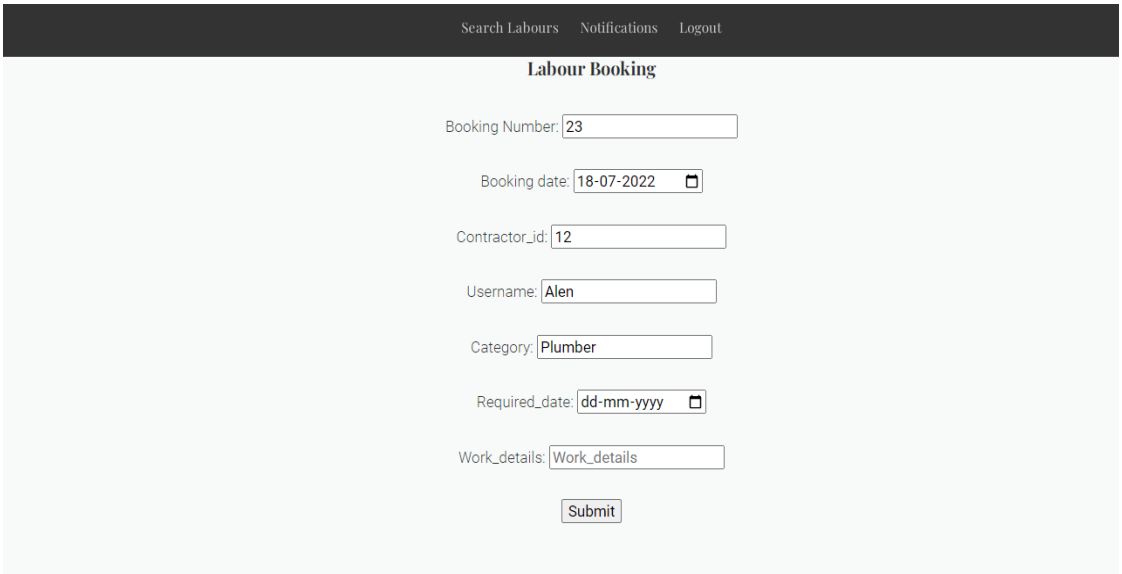
9.2.5- User homepage



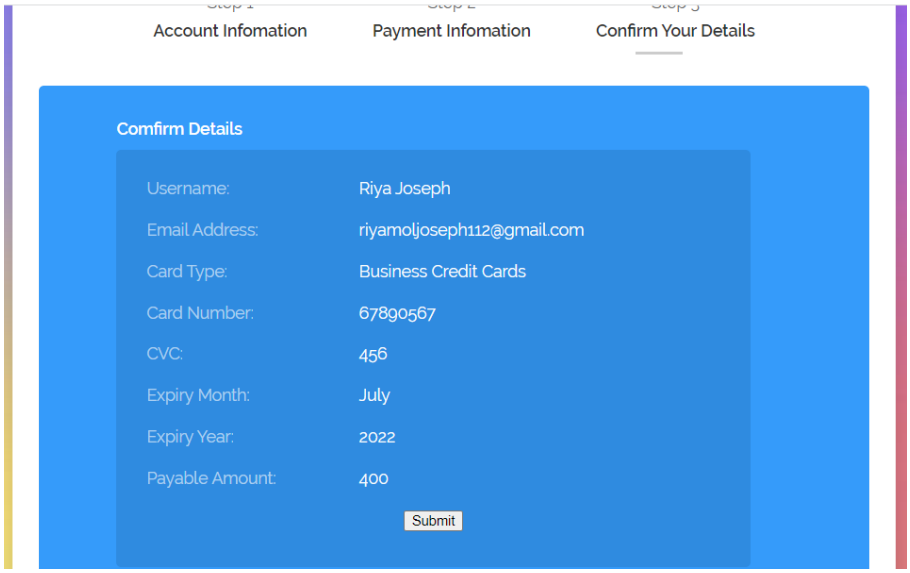
User: Search Labours



Labourbooking



Payment


Account Infomation
Payment Infomation
Confirm Your Details
Comfirm Details
Username: Riya Joseph
Email Address: riyamoljoseph112@gmail.com
Card Type: Business Credit Cards
Card Number: 67890567
CVC: 456
Expiry Month: July
Expiry Year: 2022
Payable Amount: 400
Submit

PAPER NAME

project document-1-52.pdf

AUTHOR

Reenu Joseph

WORD COUNT

9166 Words

CHARACTER COUNT

49912 Characters

PAGE COUNT

52 Pages

FILE SIZE

1.3MB

SUBMISSION DATE

Jul 22, 2022 2:20 PM GMT+5:30

REPORT DATE

Jul 22, 2022 2:21 PM GMT+5:30

13% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 13% Internet database
- 0% Publications database
- Crossref database

Excluded from Similarity Report

- Small Matches (Less then 50 words)

13% Overall Similarity

Top sources found in the following databases:

- 13% Internet database
- Crossref database
- 0% Publications database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	tutorialspoint.com Internet	5%
2	issuu.com Internet	4%
3	cse.anits.edu.in Internet	<1%
4	seminarprojects.com Internet	<1%
5	whatis.techtarget.com Internet	<1%
6	docshare.tips Internet	<1%
7	rxiv.org Internet	<1%
8	geeksforgeeks.org Internet	<1%