

# **REGIONAL TRANSPORT AUTHORITY SYSTEM**

*Project Report Submitted By*

**TEENA ROSE MATHEW**

**Reg. No.: AJC20MCA-2080**

*In Partial fulfillment for the Award of the Degree Of*

**REGULAR MASTER OF COMPUTER APPLICATIONS  
(RMCA)**

**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]

**2021-2022**

**DEPARTMENT OF COMPUTER APPLICATIONS**  
**AMAL JYOTHI COLLEGE OF ENGINEERING**  
**KANJIRAPPALLY**



**CERTIFICATE**

This is to certify that the Project report, “**REGIONAL TRANSPORT AUTHORITY SYSTEM**” is the bonafide work of **TEENA ROSE MATHEW(Reg.No:AJC20MCA-2080)** in partial fulfillment of the requirements for the award of the Degree of Integrated Master of Computer Applications under APJ Abdul Kalam Technological University during the year 2021-22.

**Ms. Paulin Paul**  
**Internal Guide**

**Ms. Grace Joseph**  
**Coordinator**

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

## **DECLARATION**

I hereby declare that the project report “**REGIONAL TRANSPORT AUTHORITY SYSTEM**” 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 Regular Master of Computer Applications (RMCA) from APJ Abdul Kalam Technological University, during the academic year 2021-2022.

**Date:**

**KANJIRAPPALLY**

**TEENA ROSE MATHEW**

**Reg. No: AJC20MCA-2080**

## 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 coordinator **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. Paulin Paul** 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.

TEENA ROSE MATHEW

## **ABSTRACT**

**Regional Transport Authority** (RTA) Information System (RTA) is an online information source developed for Road Transport Authority to facilitate the users in applying for various licenses and registrations. This tool has been designed to facilitate the flow of information within the organization. RTA provides the facility of applying licenses online, issuance of permanent license, new vehicle registration, vehicle ownership transfer, duplication of license, mcq questions, apply RC book, renewal of license and receiving payments. User can apply to these services and the subofficer can view the registration of services that is applied by the user. He has the power to approve or reject the services and the approved registration list is sent to the RTO for final verification. RTO will approve these services and finally, the user can download the certificates from the application.

In the Previous System It is not efficient in performing office work in RTO services, It includes much manual process, expensive and time consuming, It is not user friendly, Maintains local data base. It is not Generating Accurate Reports.

The existing system is not giving accurate results while doing transactions. It doesn't provide security, anyone enter into the system and can do their own transactions. It is not flexible in generating reports and many manual processes are made computerized.

To overcome problems in the existing System a new RTO services "Road Transport Authority System" is proposed after study of system. The objectives of proposed system are: Ensure data integrity and security, less manpower, Generate accurate reports, Accurate handling in multiple details.

An android application is also used in this project for scanning the vehicle number to view the details.

# CONTENT

Sl. No	Topic	Page No
1	INTRODUCTION	1
1.1	PROJECT OVERVIEW	2
1.2	PROJECT SPECIFICATION	3
2	SYSTEM STUDY	7
2.1	INTRODUCTION	8
2.2	EXISTING SYSTEM	9
2.3	DRAWBACKS OF EXISTING SYSTEM	9
2.4	PROPOSED SYSTEM	9
2.5	ADVANTAGES OF PROPOSED SYSTEM	10
3	REQUIREMENT ANALYSIS	12
3.1	FEASIBILITY STUDY	13
3.1.1	ECONOMICAL FEASIBILITY	13
3.1.2	TECHNICAL FEASIBILITY	14
3.1.3	BEHAVIORAL FEASIBILITY	14
3.2	SYSTEM SPECIFICATION	15
3.2.1	HARDWARE SPECIFICATION	15
3.2.2	SOFTWARE SPECIFICATION	15
3.3	SOFTWARE DESCRIPTION	15
3.3.1	PHP	15
3.3.2	MYSQL	16
4	SYSTEM DESIGN	18
4.1	INTRODUCTION	19
4.2	UML DIAGRAM	19
4.2.1	USE CASE DIAGRAM	20
4.2.2	SEQUENCE DIAGRAM	23
4.5	USER INTERFACE DESIGN	26
4.6	DATA BASE DESIGN	32
5	SYSTEM TESTING	43
5.1	INTRODUCTION	44
5.2	TEST PLAN	45

<b>5.2.1</b>	<b>UNIT TESTING</b>	<b>45</b>
<b>5.2.2</b>	<b>INTEGRATION TESTING</b>	<b>46</b>
<b>5.2.3</b>	<b>VALIDATION TESTING</b>	<b>46</b>
<b>5.2.4</b>	<b>USER ACCEPTANCE TESTING</b>	<b>47</b>
<b>6</b>	<b>IMPLEMENTATION</b>	<b>48</b>
<b>6.1</b>	<b>INTRODUCTION</b>	<b>49</b>
<b>6.2</b>	<b>IMPLEMENTATION PROCEDURE</b>	<b>50</b>
<b>6.2.1</b>	<b>USER TRAINING</b>	<b>50</b>
<b>6.2.2</b>	<b>TRAINING ON APPLICATION SOFTWARE</b>	<b>50</b>
<b>6.2.3</b>	<b>SYSTEM MAINTENANCE</b>	<b>51</b>
<b>7</b>	<b>CONCLUSION &amp; FUTURE SCOPE</b>	<b>52</b>
<b>7.1</b>	<b>CONCLUSION</b>	<b>53</b>
<b>7.2</b>	<b>FUTURE SCOPE</b>	<b>53</b>
<b>8</b>	<b>BIBLIOGRAPHY</b>	<b>54</b>
<b>9</b>	<b>APPENDIX</b>	<b>56</b>
<b>9.1</b>	<b>SAMPLE CODE</b>	<b>57</b>
<b>9.2</b>	<b>SCREEN SHOTS</b>	<b>72</b>

## **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

**“REGIONAL TRANSPORT AUTHORITY SYSTEM”** is a web application developed for Road Transport Authority to facilitate the users in applying for various licenses and registrations. This tool has been designed to facilitate the flow of information within the organization. RTA provides the facility of applying licenses online, issuance of permanent license, new vehicle registration, vehicle ownership transfer, duplication of license, mcq questions, apply RC book, renewal of license and receiving payments. User can apply to these services and the subofficer can view the registration of services that is applied by the user. He has the power to approve or reject the services and the approved registration list is sent to the RTO for final verification. RTO will approve these services and finally, the user can download the certificates from the application.

In the Previous System It is not efficient in performing office work in RTO services, It includes much manual process, expensive and time consuming, It is not user friendly, Maintains local data base. It is not Generating Accurate Reports.

The existing system is not giving accurate results while doing transactions. It doesn't provide security, anyone enter into the system and can do their own transactions. It is not flexible in generating reports and many manual processes are made computerized.

To overcome problems in the existing System a new RTO services “Road Transport Authority System” is proposed after study of system. The objectives of proposed system are: Ensure data integrity and security, less manpower, Generate accurate reports, Accurate handling in multiple details.

The user can also reduce the time and effort by using this system. There are thirteen tables in this main project but only login, register, driving license, duplicate license, renewal license, rc book register, ownership transfer and vehicle registration tables are used here.

In our mini project, we are making only the two modules namely user and subofficer. In main project, we are making the two extra modules such as institution and rto.

## **1.2 PROJECT SPECIFICATION**

The proposed system is made to the existing system is not giving accurate results while doing transactions. It doesn't provide security, anyone enter into the system and can do their own transactions. It is not flexible in generating reports. and many manual processes are made computerized. To overcome problems in the existing System a new RTO services "Regional Transport Authority System" is proposed after study of system. The objectives of proposed system are: Ensure data integrity and security, less manpower, Generate accurate reports, Accurate handling in multiple details. The System keeps track of the transactions in the RTO office. It maintains Renewal of learner's License, Renewal of permanent license, Issue of learner's license, Online LLR Form, Registration, Form, Issue of permanent license, payment against challan and finally it produce printouts to payment of customers.

The system includes 4 modules. They are:

### **1.RTO Module**

#### **Login**

The main activities in the application are the user login page for user. The other modules are followed by this login page. This module records only user and password of the user. RTO can approve or reject user services. RTO can view only the filtered list of registered services . He also can able to add Institutions and add questions for the license.

#### **New LLR**

LLR means learner license registration. it means for granding a person to drive him/her independently in road.

#### **New license**

A driving license is an official document issued by the government of India, permitting an individual to drive a vehicle on Indian roads. This document certifies that the

individual is eligible to drive or ride motorized vehicles such as a car, truck, bike, bus, etc., without any supervision. Commonly referred to as DL, a driving license specifies that you have undergone the required training and tests and are well-aware of all the traffic rules and regulations.

### **New vehicle registration**

The Regional Transport Office (RTO) is responsible for registering and keeping track of every vehicle that runs on the Indian roads. Each state has many RTOs spread across their length and breadth and these offices offer citizens a wide range of services related to their vehicles. One of the most important services RTOs offer are vehicle registration.

### **Vehicle Ownership transfer**

If you sell your car, it is mandatory to get the ownership of vehicle transferred to the name of the buyer. Vehicle ownership transfer which is generally called the RC transfer is important because it ensures that the vehicle and all legal liabilities related to it are transferred to the buyer.

### **Renewal License**

Application for renewal of driving licence can be made within 30 days from the date of expiry of validity. ... The driving licence will be renewed for the period of five years in respect of Non-Transport Vehicle and three years for Transport vehicles.

### **Update status and logout**

An admin will be update a user's status like LLR, license, renewal license, vehicle registration etc. and then terminate the connection.

## **2.User Module**

### **User Login**

The main activities in the application are the user login page for user. The other modules are followed by this login page. This module records only user and password of the user.

### **User Registration**

Another main function of our proposed system is registration, in order to register with the unique application details such as name; password and email are required.

### **Institution Registration**

Another main function of our proposed system is registration, in order to register with the unique application . User can also register for the institution they wanted

### **Apply LLR and License**

A user who want to LLR and License can be apply from apply license and LLR module. After, get the LLR copy also.

### **Apply vehicle registration**

A user who want to register the vehicle can be apply from vehicle registration module.

### **Apply Ownership Transfer**

A user who want to transfer the RC can be apply from ownership module. Vehicle ownership transfer which is generally called the RC transfer is important because it ensures that the vehicle and all legal liabilities related to it are transferred to the buyer.

### **Renewal License**

The license renewal can also be managed through this application.

### **Apply RC book**

The RC is essentially a document verifying that your vehicle is registered with any Regional Transport Office within the country. The certificate is initially valid for a period of 15 years, following which, it must be renewed periodically every 5 years.

**View status**

A user can view their LLR, license and RC book status.

**My profile and Logout**

A user can view and modify their profile and maintain their profile details. And then, they can terminate their account.

**3. Subofficer Module**

They can see all the requests that are applied by the users. So that, the subofficer can study the details of the user. Subofficer can approve or reject the requests that are made by the users. He can also change the password.

**4. Institution Module**

Institution can register in this site and they can approve or reject the request made by the users. He can also view the learner's license list. So that, the user can send request to those institutions nearby them.

## **CHAPTER 2**

### **SYSTEM STUDY**

## 2.1 INTRODUCTION

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minute's detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies, a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.



## **2.2 EXISTING SYSTEM**

The existing system is not giving accurate results while doing transactions. It doesn't provide security, anyone enter into the system and can do their own transactions. It is not flexible in generating reports and many manual processes are made computerized. Existing RTO Office work is very complex, waste of time & much more Real-life problem for example if a person wants to make driving license then he or she first goes to RTO office and then they give work to the agent and then agent complete their work by taking a lot of money. In this way when passing his or her vehicle number, insurance of that vehicle, etc. are taking a lot of time. And nowadays each and every person is in hurry so by analyzing and considering these problems we are developing one web application which overcomes this problem and get a solution in an efficient way.

## **2.3 DRAWBACKS OF EXISTING SYSTEM**

- It is not efficient in performing office work in RTO services.
- It includes much manual process and time consuming.
- It is not user friendly.
- Maintains local database.
- It is not Generating Accurate Reports.

## **2.4 PROPOSED SYSTEM**

Here, we are developing a web application for RTO so here we give a brief description of our project overview. To provide familiar environment means the needy user can access this site for their work purpose related to RTO. First user needs to fill the registration form so that we provide authentication to him and then user can choose option he wants means if he select to making a driving license then we provide driving license requirement details and give available date to him so that he come on that date direct give the test so that he can save his time as well as money. If user wants to pass his vehicle number then also it takes time in old system but here we provide facility that user he buy new vehicle he should have to first register on our site and fill all the required and importance details of vehicle and we gives this details to RTO office directly so

that this work will get complete within less time and the user get his number template easily.

The administrator is providing for authentication purpose as well as it handles all the database of RTO and manages all the process. He has authority to approved learning license number, permanent license number; pass the vehicle registration number, etc. Facilities are provided by the administrator

The proposed system is defined to meets all the disadvantages of the existing system. It is necessary to have a system that is more user friendly and user attractive. In our proposed system .Users of this proposed system are RTO, user, subofficer and institution. RTA provides the facility of applying licenses online, issuance of permanent license, new vehicle registration, vehicle ownership transfer, duplication of license, mcq questions, apply RC book, renewal of license and receiving payments. User can apply to these services and the subofficer can view the registration of services that is applied by the user .He has the power to approve or reject the services and the approved registration list is send to the RTO for final verification .RTO will approve these services and finally, the user can download the certificates from the application.

The aim of proposed system is to develop a system of improved facilities. The system provides proper security and reduces the manual work.

The proposed system provides consistency of data and reduces the paper work. . It provides security, anyone enter into the system and can do their own transactions. It is flexible in generating reports and many manual processes are made computerized.

## **2.5 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**

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic and Operational feasibilities. The following are its features: -

#### **3.1.1 Economical Feasibility**

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

The proposed system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

The cost of project, REGIONAL TRANSPORT AUTHORITY SYSTEM was divided according to the system used, its development cost and cost for hosting the project. According to all the calculations the project was developed in a low cost. As it is completely developed using open source software.

### 3.1.2 Technical Feasibility

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.

Technical issues raised during the investigation are:

- Does the existing technology sufficient for the suggested one?
- Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project requires High Resolution Scanning device and utilizes Cryptographic techniques. Through the technology may become obsolete after some period of time, due to the fact that newer version of same software supports older versions, the system may still be used. So there are minimal constraints involved with this project. The system has been developed using PHP in front end and MySQL in server in back end, the project is technically feasible for development. The system has been developed using PHP in front end and MySQL in server in back end, the project is technically feasible for development. The System used was also of good performance of Processor Intel i3 core; RAM 4GB and, Hard disk 1TB

### 3.1.3 Behavioral Feasibility

The proposed system includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

## **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, J Query, PHP, CSS

## **3.3 SOFTWARE DESCRIPTION**

### **3.3.1 PHP**

PHP is a server side scripting language designed for web development but also used as a general purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by Rasmus Ledorf in 1995, the reference implementation of PHP is now produced by the PHP group. While PHP originally stood for personal Home page ,it now stands for PHP:HypertextPreprocessor, a recursive acronym.PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page.PHP commands can be embedded directly into a HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone incompatible with the GNU General Public License (GPL) due to restrictions on the usage of the term PHP.PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

### 3.3.2 MySQL

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. The MySQL Web site provides the latest information about MySQL software.

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

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- **MySQL databases are relational.**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and “pointers” between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data. The SQL part of “MySQL” stands for “Structured Query Language”. SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax. SQL is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, “SQL92” refers to the standard released in 1992,

“SQL: 1999” refers to the standard released in 1999, and “SQL: 2003” refers to the current version of the standard. We use the phrase “the SQL standard” to mean the current version of the SQL Standard at any time.



- **MySQL software is Open Source.**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), to define what you may and may not do with the software in different situations. If you feel uncomfortable with the GPL or need to embed MySQL code into a commercial application, you can buy a commercially licensed version from us. See the MySQL Licensing Overview for more information.

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

If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available.

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

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs). We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

## **CHAPTER 4**

### **SYSTEM DESIGN**

## 4.1 INTRODUCTION

Design is the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system. The term “design” is defined as “the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization”. It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm that is used. The system design develops the architectural detail required to build a system or product. As in the case of any systematic approach, this software too has undergone the best possible design phase fine tuning all efficiency, performance and accuracy levels. The design phase is a transition from a user oriented document to a document to the programmers or database personnel. System design goes through two phases of development: Logical and Physical Design.

## 4.2 UML DIAGRAM

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML was created by the Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.

UML stands for **Unified Modeling Language**. UML is different from the other common programming languages such as C++, Java, COBOL, etc. UML is a pictorial language used to make software blueprints. UML can be described as a general purpose visual modeling language to visualize, specify, construct, and document software system. Although UML is generally used to model software systems, it is not limited within this boundary. It is also used to model non-software systems as well. For example, the process flow in a manufacturing unit, etc. UML is not a programming language but tools can be used to generate code in various languages using UML diagrams. UML has a direct relation with object oriented analysis and design. After some standardization, UML has become an OMG standard. All the elements, relationships are used to make a complete

UML diagram and the diagram represents a system. The visual effect of the UML diagram is the most important part of the entire process. All the other elements are used to make it complete. UML includes the following nine diagrams.

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

#### **4.2.1 USE CASE DIAGRAM**

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems.

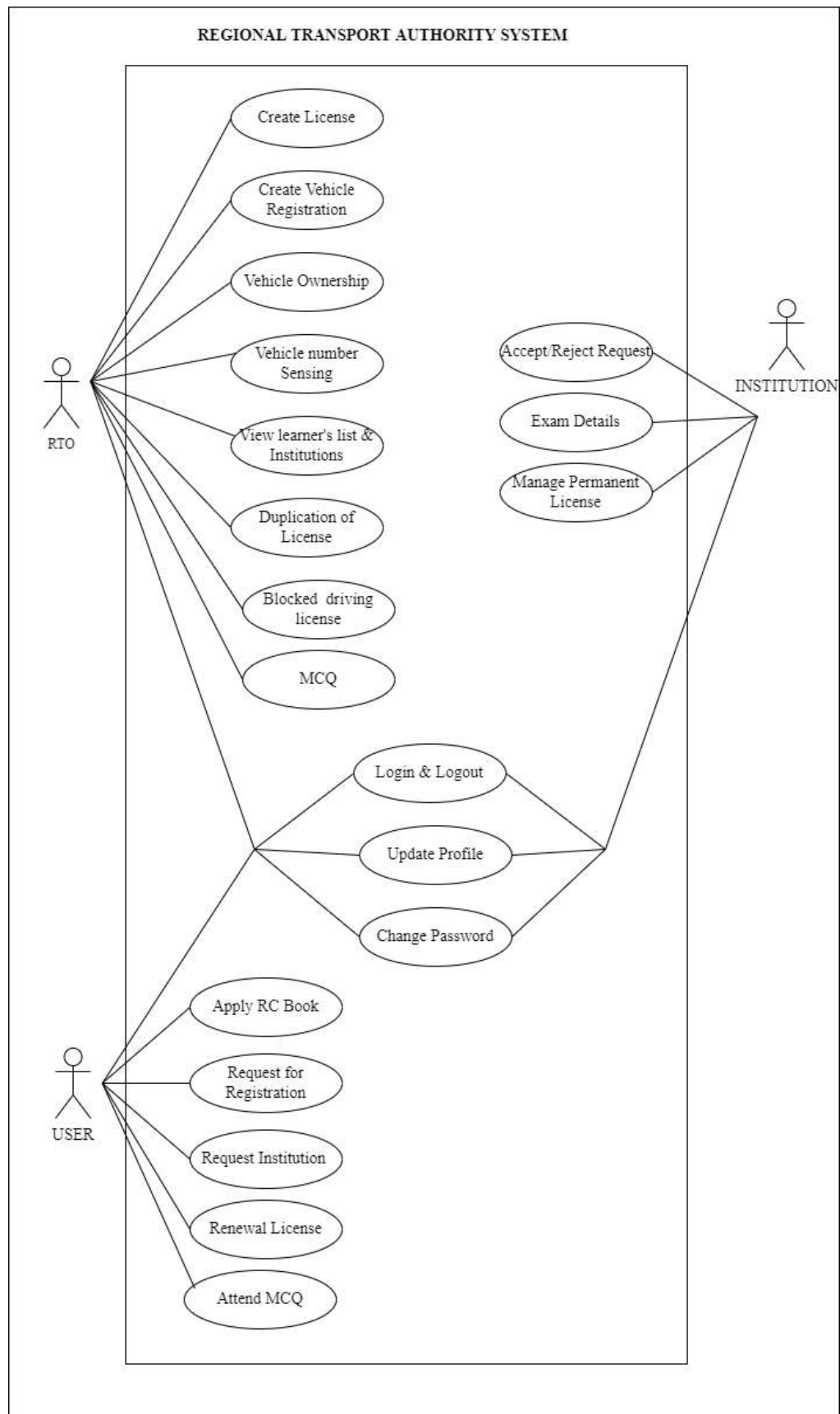
System objectives can include planning overall requirements, validating a hardware design, testing and debugging a software product under development, creating an online help reference, or performing a consumer-service-oriented task. For example, use cases in a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components.

- 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 drawn to capture the functional requirements of a system. After identifying the above items, we have to use the following guidelines to draw an efficient 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.

Fig 1 : Use case diagram for Regional Transport Authority System



### 4.2.2 SEQUENCE DIAGRAM

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

#### Sequence Diagram Notations –

- i. **Actors** – An actor in a UML diagram represents a type of role where it interacts with the system and its objects. It is important to note here that an actor is always outside the scope of the system we aim to model using the UML diagram. We use actors to depict various roles including human users and other external subjects. We represent an actor in a UML diagram using a stick person notation. We can have multiple actors in a sequence diagram.
- ii. **Lifelines** – A lifeline is a named element which depicts an individual participant in a sequence diagram. So basically each instance in a sequence diagram is represented by a lifeline. Lifeline elements are located at the top in a sequence diagram
- iii. **Messages** – Communication between objects is depicted using messages. The messages appear in a sequential order on the lifeline. We represent messages using arrows. Lifelines and messages form the core of a sequence diagram.

Messages can be broadly classified into the following categories:

- Synchronous messages
- Asynchronous Messages
- Create message
- Delete Message
- Self-Message
- Reply Message
- Found Message

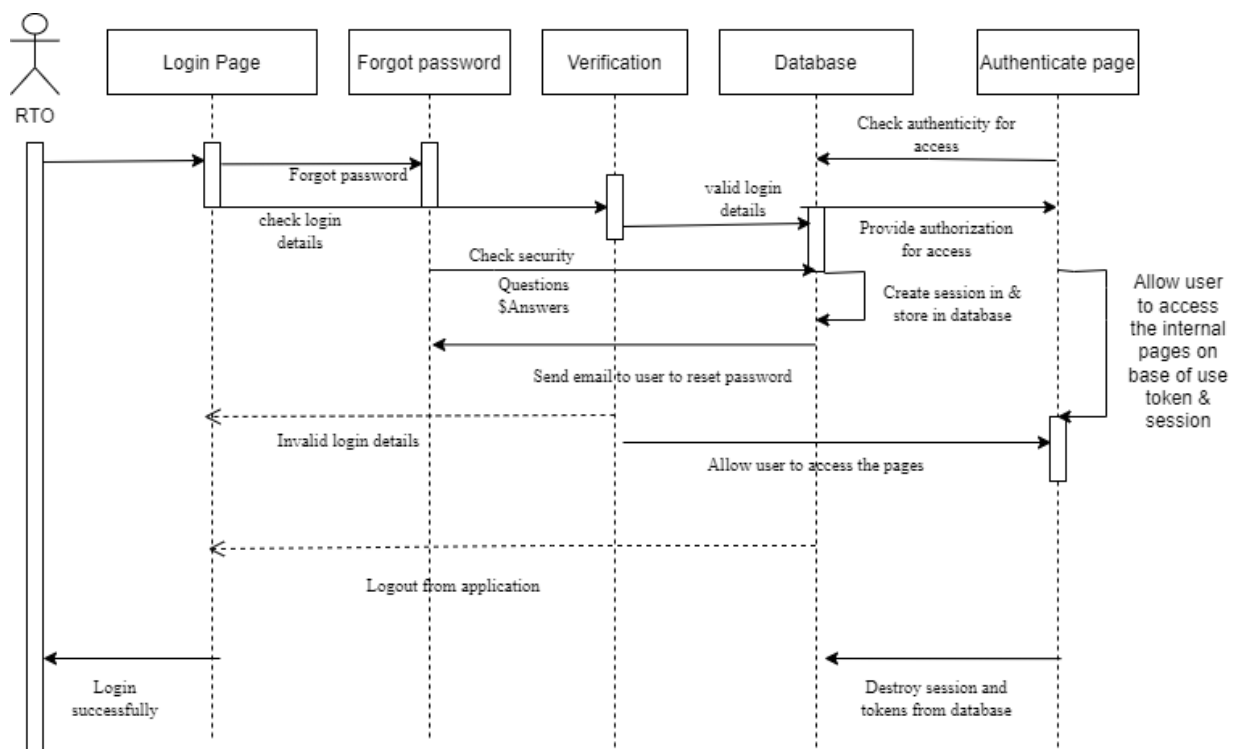
- Lost Message

**iv. Guards** – To model conditions we use guards in UML. They are used when we need to restrict the flow of messages on the pretext of a condition being met. Guards play an important role in letting software developers know the constraints attached to a system or a particular process.

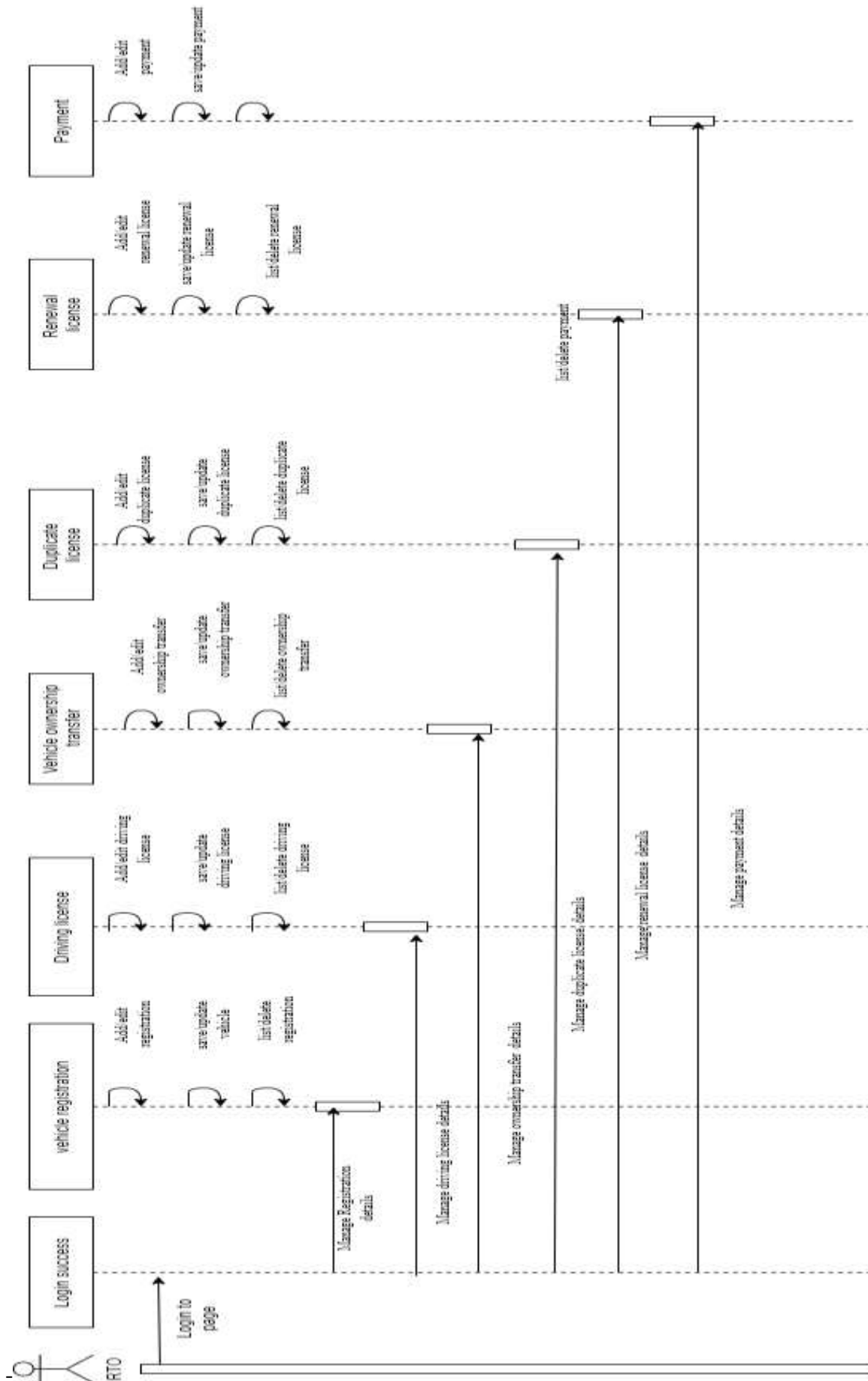
#### Uses of sequence diagrams –

- 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.

Fig 1 : Sequence diagram for Regional Transport Authority System



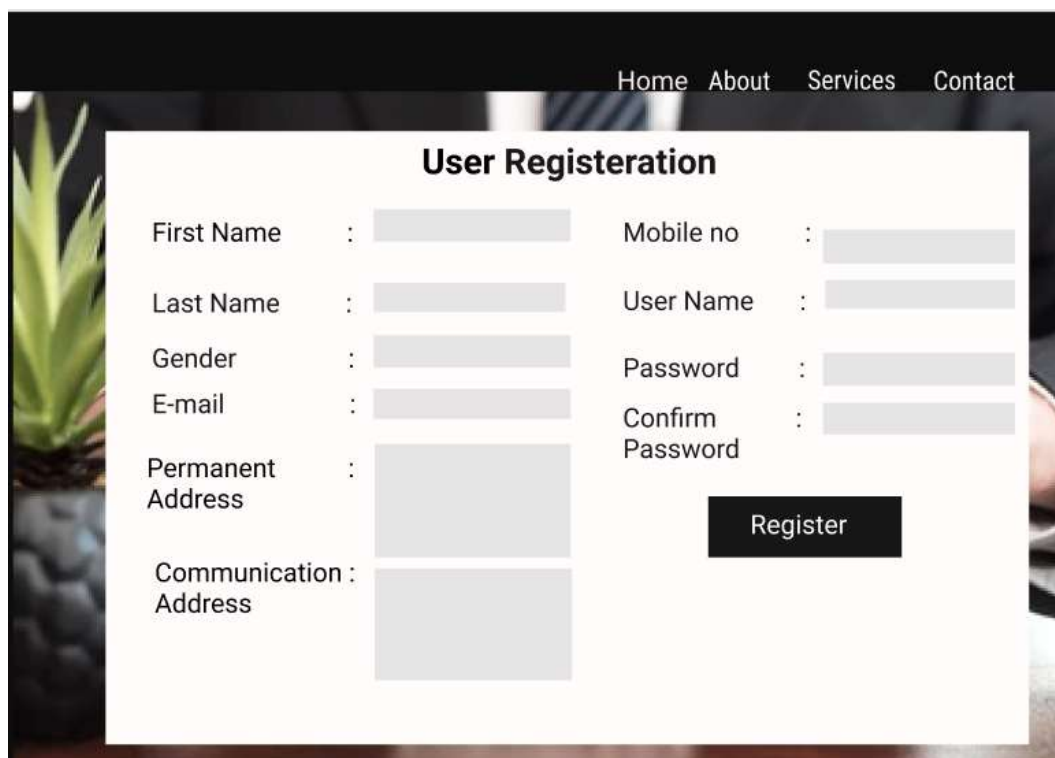




## 4.5 USER INTERFACE DESIGN

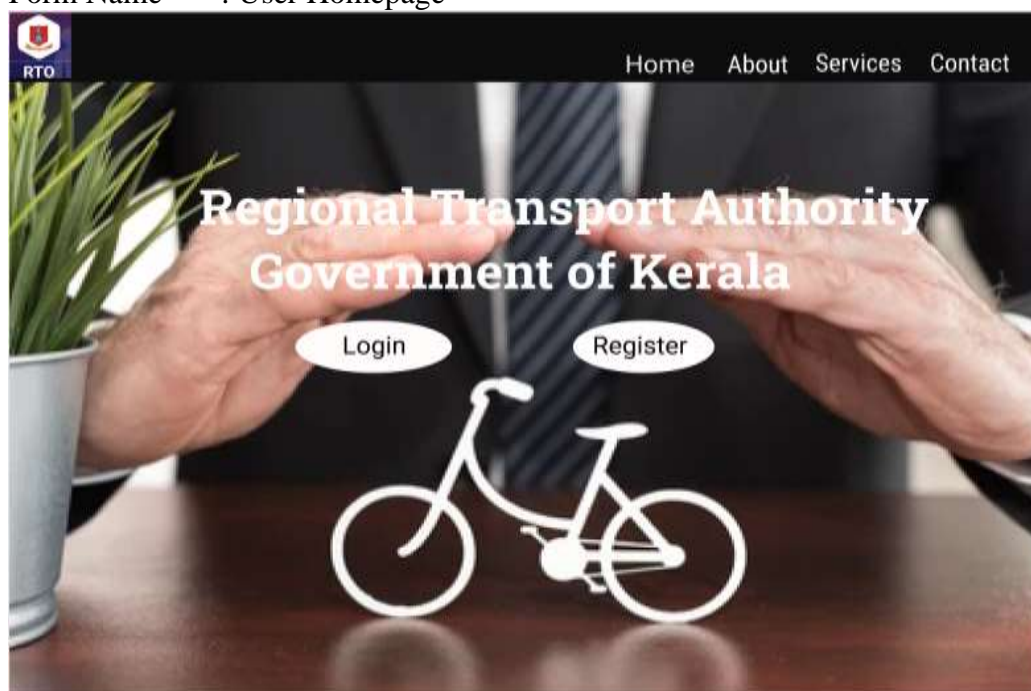
### 4.5.1-INPUT DESIGN

Form Name : User Registration

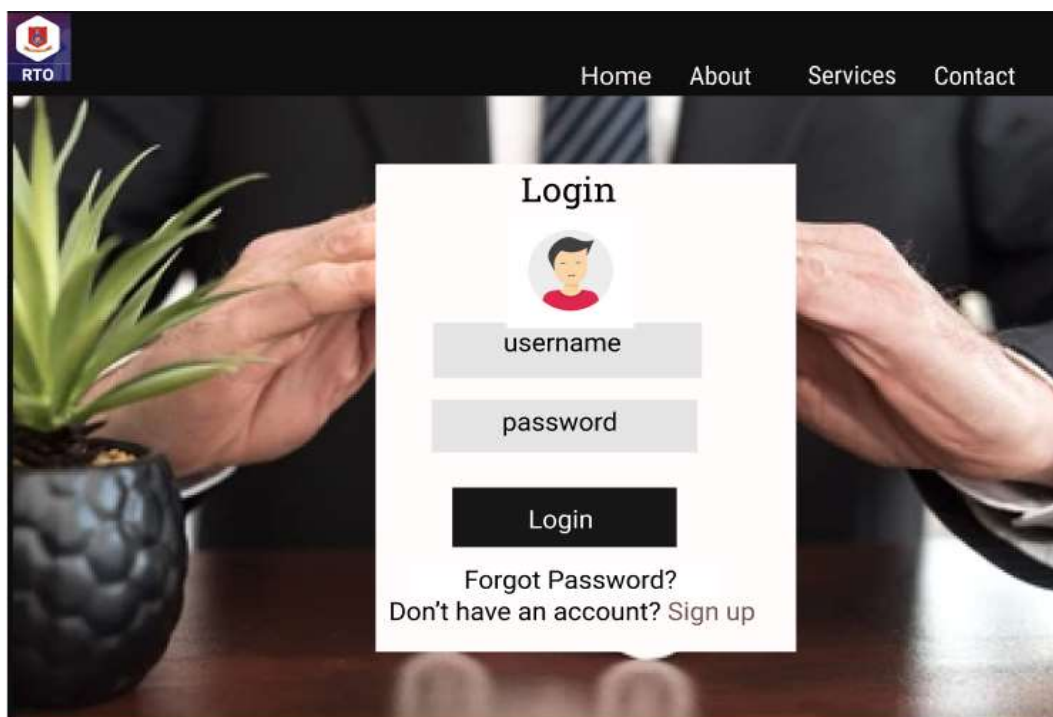


A screenshot of a web application's user registration form. The form is titled "User Registration" and is set against a background image of a person's hands holding a bicycle. The form fields are arranged in two columns. The left column contains: First Name, Last Name, Gender, E-mail, Permanent Address, and Communication Address. The right column contains: Mobile no, User Name, Password, and Confirm Password. Each field is represented by a grey rectangular input box. A "Register" button is located at the bottom right of the form. The website's navigation menu at the top includes "Home", "About", "Services", and "Contact".

Form Name : User Homepage

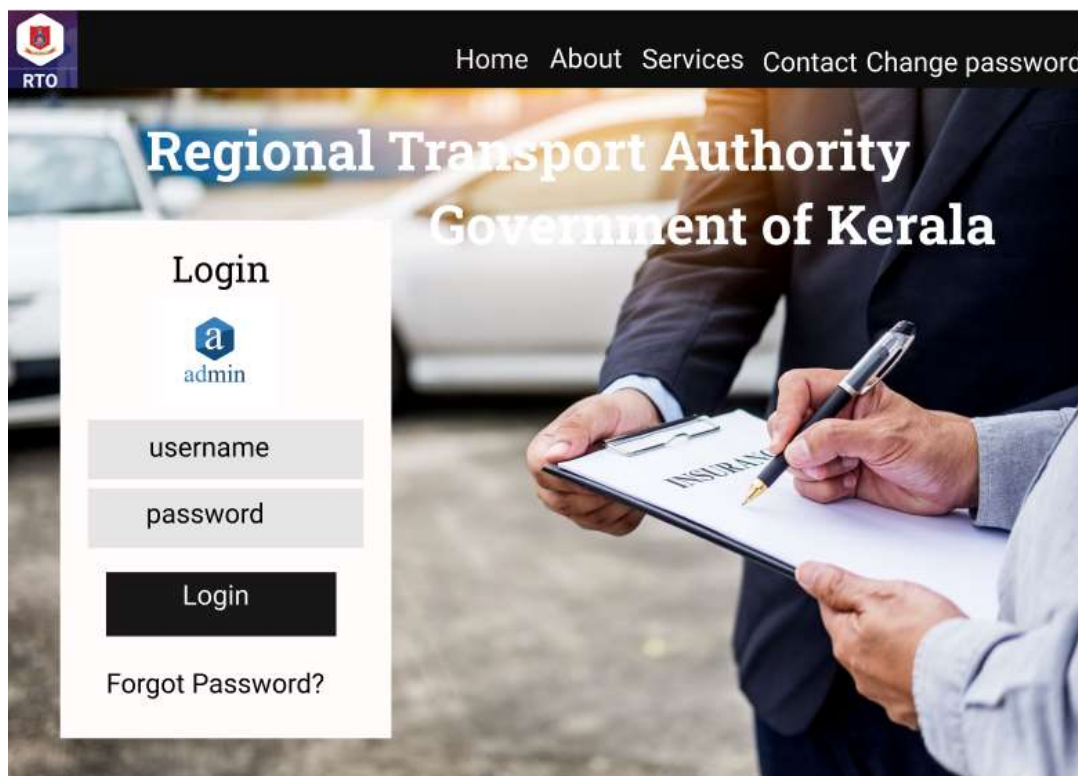


Form Name : User Login



The screenshot shows the 'User Login' form on the RTO website. The form is centered on a background image of a person's hands holding a small potted plant. The form has a white background and a black border. At the top, it says 'Login' with a user icon. Below that are two input fields: 'username' and 'password'. A black 'Login' button is positioned below the password field. At the bottom, there are two links: 'Forgot Password?' and 'Don't have an account? Sign up'.

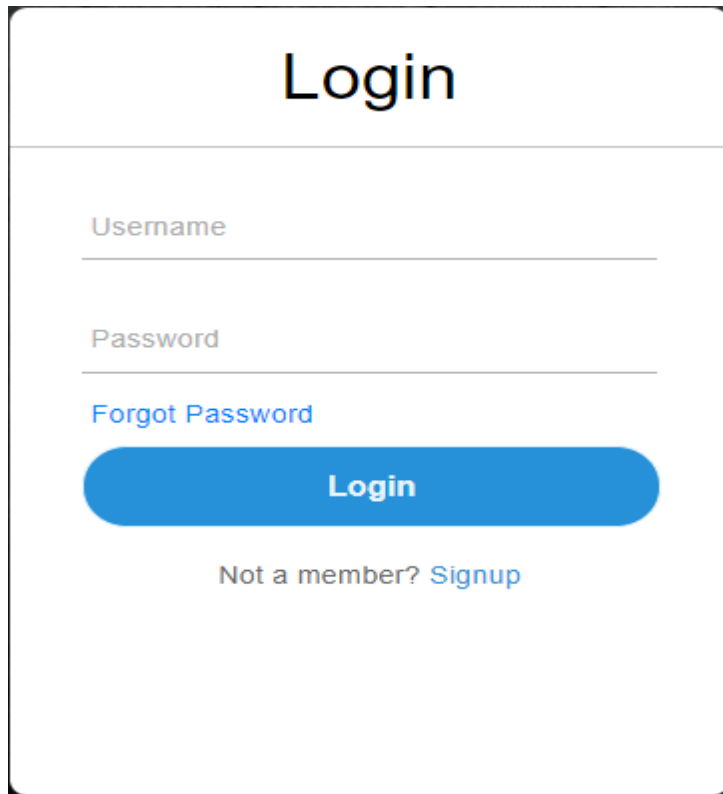
Form Name : Admin Homepage



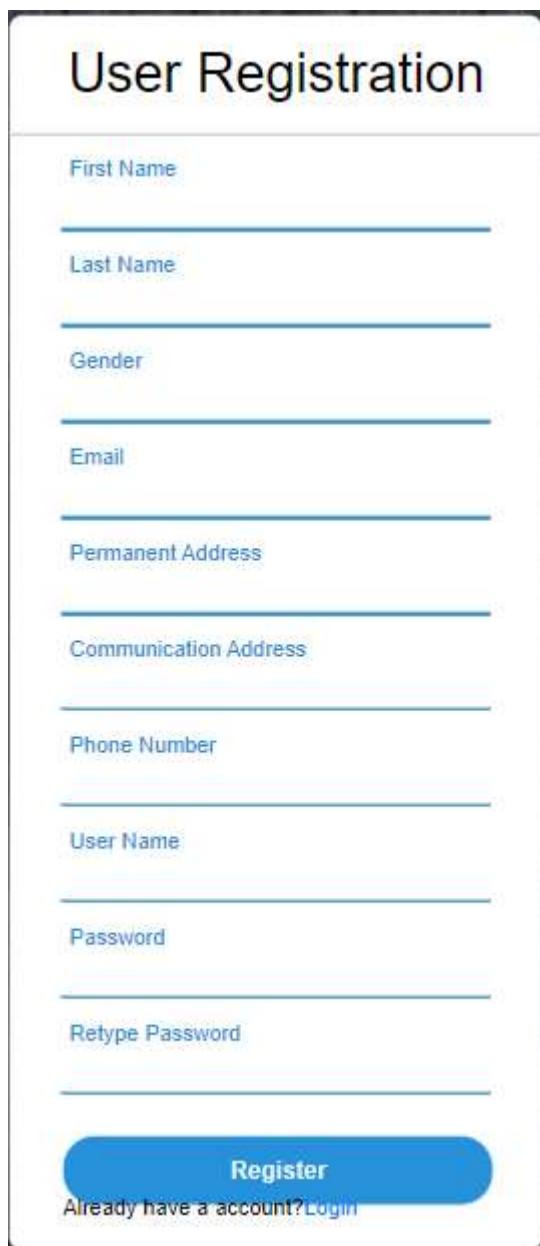
The screenshot shows the 'Admin Homepage' on the RTO website. The page has a dark header with the RTO logo and navigation links: 'Home', 'About', 'Services', 'Contact', and 'Change password'. The main content area features a large banner with the text 'Regional Transport Authority Government of Kerala'. Below the banner is a 'Login' form for the admin. The form includes a user icon with the letter 'a' and the text 'admin'. It has two input fields: 'username' and 'password'. A black 'Login' button is positioned below the password field. At the bottom, there is a link: 'Forgot Password?'.

## 4.5.2 OUTPUT DESIGN

**Form Name** : User Login



The image shows a user login form with a white background and a thin black border. At the top, the word "Login" is centered in a large, black, sans-serif font. Below this, there are two input fields: "Username" and "Password", each with a light gray label and a horizontal line for text entry. Under the "Password" field, there is a link "Forgot Password" in blue text. Below the links is a large, rounded blue button with the word "Login" in white, bold, sans-serif font. At the bottom, there is a link "Not a member? Signup" in blue text.

**Form Name : User Registration**

The image shows a user registration form titled "User Registration". It contains several input fields for personal and contact information, followed by a registration button and a login link.

**User Registration**

First Name

Last Name

Gender

Email

Permanent Address

Communication Address

Phone Number

User Name

Password

Retype Password

**Register**

Already have a account? [Login](#)

**Form Name: Institution Registration**

## Institution Registration

Ownership Name

Institution Name

Gender

Email

Permanent Address

Communication Address

Phone Number

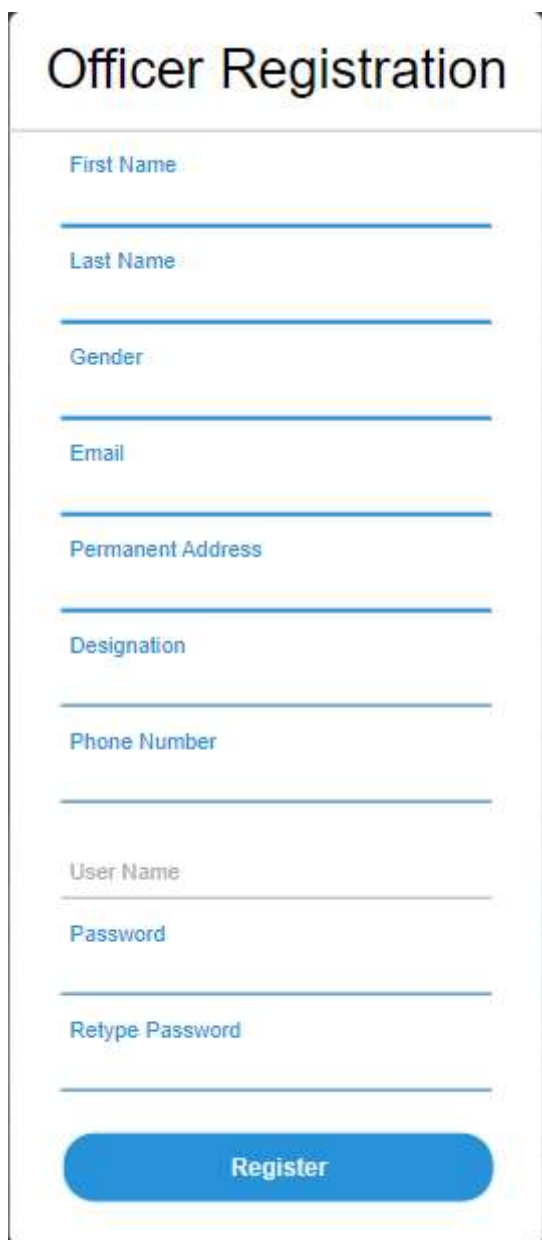
User Name

Password

Retype Password

Register

Form Name : Subofficer Registration



The image shows a web form titled "Officer Registration". It contains several input fields for personal and professional information, followed by a "Register" button. The fields are labeled as follows:

- First Name
- Last Name
- Gender
- Email
- Permanent Address
- Designation
- Phone Number
- User Name
- Password
- Retype Password

A blue "Register" button is located at the bottom of the form.

## 4.6. DATABASE DESIGN

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- Data Integrity
- Data independence

### 4.6.1 Relational Database Management System (RDBMS)

A relational model represents the database as a collection of relations. Each relation resembles a table of values or file of records. In formal relational model terminology, a row is called a tuple, a column header is called an attribute and the table is called a relation. A relational database consists of a collection of tables, each of which is assigned a unique name. A row in a table represents a set of related values.

#### Relations, Domains & Attributes

A table is a relation. The rows in a table are called tuples. A tuple is an ordered set of  $n$  elements. Columns are referred to as attributes. Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity. A domain  $D$  is a set of atomic values. A common method of specifying a domain is to specify a data type from which the data values forming the domain are drawn. It is also useful to specify a name for the domain to help in interpreting its values.

Every value in a relation is atomic, that is not decomposable.



### 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 grouped together in the simplest way so that later changes can be made with minimum impact on data structures. Normalization is formal process of data structures in manners that eliminates redundancy and promotes integrity. Normalization is a technique of separating redundant fields and breaking up a large table into a smaller one. It is also used to avoid insertion, deletion, and updating anomalies. Normal form in data modelling use two concepts, keys and relationships. A key uniquely identifies a row in a table. There are two types of keys, primary key and foreign key. A primary key is an element or a combination of elements in a table whose purpose is to identify records from the same table. A foreign key is a column in a table that uniquely identifies record from a different table. All the tables have been normalized up to the third normal form.

As the name implies, it denotes putting things in the normal form. The application developer via normalization tries to achieve a sensible organization of data into proper tables and columns and where names can be easily correlated to the data by the user. Normalization eliminates repeating groups at data and thereby avoids data redundancy which proves to be a great burden on the computer resources. These include:

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

**First Normal Form**

The First Normal Form states that the domain of an attribute must include only atomic values and that the value of any attribute in a tuple must be a single value from the domain of that attribute. In other words 1NF disallows “relations within relations” or “relations as attribute values within tuples”. The only attribute values permitted by 1NF are single atomic or indivisible values. The first step is to put the data into First Normal Form. This can be done by moving data into separate tables where the data is of similar type in each table. Each table is given a Primary Key or Foreign Key as per requirement of the project. In this we form new relations for each non-atomic attribute or nested relation. This eliminated repeating groups of data. A relation is said to be in first normal form if only if it satisfies the constraints that contain the primary key only.

**Second Normal Form**

According to Second Normal Form, for relations where primary key contains multiple attributes, no non-key attribute should be functionally dependent on a part of the primary key. In this we decompose and setup a new relation for each partial key with its dependent attributes. Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it. This step helps in taking out data that is only dependent on a part of the key. A relation is said to be in second normal form if and only if it satisfies all the first normal form conditions for the primary key and every non-primary key attributes of the relation is fully dependent on its primary key alone.

**Third Normal Form**

According to Third Normal Form, Relation should not have a non-key attribute functionally determined by another non-key attribute or by a set of non-key attributes. That is, there should be no transitive dependency on the primary key. In this we decompose and set up relation that includes the non-key attributes that functionally determines other non-key attributes. This step is taken to get rid of anything that does not depend entirely on the Primary Key. A relation is said to be in third normal form if only if it is in second normal form and more over the non key attributes of the relation should not be depend on other non-key attribute.

**TABLE DESIGN****Table No: 01**

Table Name: tbl\_register

Primary Key: reg\_id

Foreign Key: login\_id

Table Description : To store registration information of user , subofficer and institution

Fieldname	Datatype	Size	Description
reg_id	int	11	Registration id
fname	Varchar	50	First name of the user
lname	Varchar	50	Last name of the user
gender	Varchar	6	Gender of the user
email	Varchar	50	Email id
paddress	Varchar	100	Permanent address of the user
caddress	Varchar	100	Communication address of the user
designation	Varchar	50	Designation of the subofficer
phone_no	Varchar	10	phone number
ownership_name	Varchar	50	ownership name of the institution
Institution Name	Varchar	50	Name of the institution
username	Varchar	25	Username of the user
password	Varchar	25	Password of the user
repassword	Varchar	25	Retype the password

**Table No: 02**

Table Name: tbl\_drivinglicense

Primary Key: driving\_id

Foreign Key: reg\_id

Table Description : keeps information about the license owner

Fieldname	Datatype	Size	Description
driving_id	int	11	Driving license id
reg_id	int	11	Registration id
first_name	Varchar	25	First name of the user
last_name	Varchar	50	Last name of the user
age	Int	10	Age of the user
gender	Varchar	6	Gender of the user
parent_name	Varchar	50	Name of the parent
paddress	Varchar	100	Permanent address of the user
caddress	Varchar	100	Communication address of the user
email	Varchar	25	Email id
phone_no	Varchar	10	phone number
license_type	Varchar	50	Type of the license
date_of_issue	Date		license issue date
expiriry_date	Date		Expiriry date of the license
blood	Varchar	10	Blood group of the user
image	Varchar	255	Image of the user
is_approved	Varchar	20	status

**Table No: 03**

Table Name: tbl\_duplicatelicense

Primary Key: dup\_id

Foreign Key: driving\_id

Table Description : keeps information about duplication of license

Fieldname	Datatype	Size	Description
dup_id	int	11	Duplicate license id
driving_id	int	11	Driving license id
first_name	Varchar	25	First name of the user
last_name	Varchar	50	Last name of the user
age	Int	10	Age of the user
gender	Varchar	6	Gender of the user
parent_name	Varchar	50	Name of the parent
paddress	Varchar	100	Permanent address of the user
caddress	Varchar	100	Communication address of the user
email	Varchar	25	Email id
phone_no	Varchar	10	phone number
license_type	Varchar	50	Type of the license
date_of_issue	Date		license issue date
expiriry_date	Date		Expiriry date of the license
blood	Varchar	10	Blood group of the user
image	Varchar	255	Image of the user
is_approved	Varchar	20	status

**Table No: 04**

Table Name: tbl\_rcbook

Primary Key: rc\_id

Foreign Key: reg\_id

Table Description : keeps information about RC Book

Fieldname	Datatype	Size	Description
rc_id	int	11	RC Book id
reg_id	int	11	Registration id
first_name	Varchar	25	First name of the user
last_name	Varchar	50	Last name of the user
age	Int	10	Age of the user
paddress	Varchar	100	Permanent address of the user
caddress	Varchar	100	Communication address of the user
dealer_name	Varchar	25	Name of the Dealer
vehicle_name	Varchar	25	Name of the vehicle
vehicle_type	Varchar	25	Type of the vehicle
vehicle_number	Varchar	25	Number of the vehicle
fuel	varchar	25	Fuel type of the vehicle
weight	int	10	Weight of the vehicle
seating_capacity	int	10	Seating capacity of the vehicle
image	Varchar	255	Image of the vehicle
is_approved	Varchar	20	status

**Table No: 05**

Table Name: tbl\_ownership

Primary Key: owner\_id

Foreign Key: reg\_id

Table Description : keeps information about the ownership transfer

Fieldname	Datatype	Size	Description
owner_id	int	11	Ownership id
reg_id	int	11	Registration id
first_name	Varchar	25	First name of the user
last_name	Varchar	50	Last name of the user
age	Int	10	Age of the user
paddress	Varchar	100	Permanent address of the user
caddress	Varchar	100	Communication address of the user
email	Varchar	25	Email id
phone_no	Varchar	10	phone number
dealer_name	Varchar	25	Name of the Dealer
dealer_address	Varchar	50	Address of the dealer
recipient_name	Varchar	25	Name of the recipient
recipient_address	Varchar	50	Address of the recipient
vehicle_number	Varchar	25	Number of the vehicle
vehicle_name	Varchar	25	Name of the vehicle
fuel	varchar	25	Fuel type of the vehicle
weight	int	10	Weight of the vehicle
seating_capacity	int	10	Seating capacity of the vehicle
is_approved	Varchar	20	status

**Table No: 06**

Table Name: tbl\_renewal

Primary Key: renewal\_id

Foreign Key: reg\_id

Table Description : keeps information about renewal of license

Fieldname	Datatype	Size	Description
renewal_id	int	11	Renewal id
reg_id	int	11	Registration id
first_name	Varchar	25	First name of the user
last_name	Varchar	50	Last name of the user
age	Int	10	Age of the user
gender	varchar	10	Gender of the user
parent_name	varchar	25	name of the parent of the user
paddress	Varchar	100	Permanent address of the user
caddress	Varchar	100	Communication address of the user
email	Varchar	25	Email id
phone_no	Varchar	10	phone number
license_type	Varchar	50	Type of the license
date_of_issue	Date		Issue of the date
expiriry_date	Date		Expiriry date of the license
image	Varchar	255	Image of the vehicle
is_approved	Varchar	20	status



**Table No: 07**

Table Name: tbl\_vreg

Primary Key: vreg\_id

Foreign Key: reg\_id

Table Description : keeps information about vehicle registration

Fieldname	Datatype	Size	Description
vreg_id	int	11	Vehicle registration id
reg_id	int	11	Registration id
first_name	Varchar	25	First name of the user
last_name	Varchar	50	Last name of the user
age	Int	10	Age of the user
paddress	Varchar	100	Permanent address of the user
caddress	Varchar	100	Communication address of the user
email	Varchar	25	Email id
phone_no	Varchar	10	phone number
dealer_name	Varchar	25	Name of the Dealer
dealer_address	Varchar	50	Address of the dealer
vehicle_name	Varchar	25	Name of the vehicle
vehicle_type	varchar	25	Type of the vehicle
fuel	varchar	25	Fuel type of the vehicle
weight	int	10	Weight of the vehicle
seating_capacity	int	10	Seating capacity of the vehicle
image	varchar	255	Image of the vehicle
is_approved	Varchar	20	status

**Table No: 08**

Table Name: tbl\_login

Primary Key: login\_id

Foreign Key:

Table Description : To store user Login information

Fieldname	Data type	Size	Description
login_id	Int	10	Primary key
username	Varchar	50	Username of the user
password	Varchar	50	Password of the user
usertype	Varchar	25	Usertype of the user
userstatus	Varchar	25	User status

## **CHAPTER 5**

### **SYSTEM TESTING**

## 5.1 INTRODUCTION

Software Testing is the process of executing software in a controlled manner, in order to answer the question - Does the software behave as specified? Software testing is often used in association with the terms verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, analysis, inspections, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

Other activities which are often associated with software testing are static analysis and dynamic analysis. Static analysis investigates the source code of software, looking for problems and gathering metrics without actually executing the code. Dynamic analysis looks at the behavior of software while it is executing, to provide information such as execution traces, timing profiles, and test coverage information.

Testing is a set of activity that can be planned in advanced and conducted systematically. Testing begins at the module level and work towards the integration of entire computers based system. Nothing is complete without testing, as it vital success of the system testing objectives, there are several rules that can serve as testing objectives. They are:

Testing is a process of executing a program with the intent of finding an error.

- 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 a testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrate that the software function appear to be working according to the specification, that performance requirement appear to have been met.

There are three ways to test program.

- For correctness
- For implementation efficiency
- For computational complexity

Test for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

## **5.2 TEST PLAN**

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The Test Plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers is always responsible for testing the individual units of the programs, ensuring that each performs the function for which it was designed. There is an independent test group (ITG) which is to remove the inherent problems associated with letting the builder to test the thing that has been built. The specific objectives of testing should be stated in measurable terms. So that the mean time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours per regression test all should be stated within the test plan.

The levels of testing include:

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

### **5.2.1 Unit Testing**

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, all error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur. Boundary testing is the last task of unit testing step. Software often fails at its boundaries.

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of the modules were found and were rectified. After coding each module is tested and run individually. All unnecessary code were removed and ensured that all modules are working, and gives the expected result.

### **5.2.2 Integration Testing**

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expanse of entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop. After performing unit testing in the System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

### **5.2.3 Validation Testing or System Testing**

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly known as Black Box testing or System tests.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

#### **5.2.4 Output Testing or User Acceptance Testing**

The system considered is tested for user acceptance; here it should satisfy the firm's need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points:

- Input Screen Designs,
- Output Screen Designs,

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

## **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

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system. In many organizations someone who will not be operating it, will commission the software development project. In the initial stage people doubt about the software but we have to ensure that the resistance does not build up, as one has to make sure that:

- The active user must be aware of the benefits of using the new system. □ Their confidence in the software is built up.
- Proper guidance is imparted to the user so that he is comfortable in using the application.

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not up running on the server, the actual process won't take place.

### 6.2.1 User Training

User training is designed to prepare the user for testing and converting the system. To achieve the objective and benefits expected from computer based system, it is essential for the people who will be involved to be confident of their role in the new system. As system becomes more complex, the need for training is more important. By user training the user comes to know how to enter data, respond to error messages, interrogate the database and call up routine that will produce reports and perform other necessary functions.

### 6.2.2 Training on the Application Software

After providing the necessary basic training on computer awareness the user will have to be trained on the new application software. This will give the underlying philosophy of the use of the new system such as the screen flow, screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the ways to correct the data entered. It should then cover information needed by the specific user/ group to use the system or part of the system while imparting the training of the program on the application. This training may be different across different user groups and across different levels of hierarchy

### **6.2.3 System Maintenance**

Maintenance is the enigma of system development. The maintenance phase of the software cycle is the time in which a software product performs useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for system maintenance is for it to make adaptable to the changes in the system environment. Software maintenance is of course, far more than "Finding Mistakes".

## **CHAPTER 7**

### **CONCLUSION AND FUTURE SCOPE**

## **7.1 CONCLUSION**

Regional Transport Authority System is a web application , which is useful for RTO works completion online .Here we are developing such types of modules which helps to reduce the RTO work manually and it helps to save the time of the user. Considerably reduce the corruption in the transport department. Keep the license document safely.

## **7.2 FUTURE SCOPE**

This system is also helpful for Traffic police also. The traffic police to be more effective in controlling repeat violators of traffic rules. Traffic Police have the database of registration numbers as well as the history of driving license holders. When a traffic policeman would enter the details of any vehicle caught violating traffic rules, it would give the complete details of that particular vehicle including the name and address of the owner and the make, model and other details of the vehicle. Not only this, the details of the driving license holder would also be available. Therefore enhanced penalties would be imposed for repetition of violation of traffic rules. Fake registration plates, if any, would be detected immediately.

## **CHAPTER 8**

## **BIBLIOGRAPHY**

**REFERENCES:**

- Gary B. Shelly, Harry J. Rosenblatt, “*System Analysis and Design*”, 2009.
- Roger S Pressman, “*Software Engineering*”, 1994.
- PankajJalote, “*Software 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](http://www.w3schools.com)
- [www.jquery.com](http://www.jquery.com)
- <http://homepages.dcc.ufmg.br/~rodolfo/es-1-03/IEEE-Std-830-1998.pdf>
- [www.agilemodeling.com/artifacts/useCaseDiagram.html](http://www.agilemodeling.com/artifacts/useCaseDiagram.html)

## **CHAPTER 9**

## **APPENDIX**



## 9.1 Sample Code

### Login.php

```
<?php
session_start();
include("dbconnection.php");
?>

<!DOCTYPE html>

<html lang="en" dir="ltr">
<head>
<meta charset="utf-8">
<title>RTO - Login</title>
<meta content="width=device-width, initial-scale=1.0" name="viewport">
<meta content="Free Website Template" name="keywords">
<meta content="Free Website Template" name="description">

<!-- Favicon -->
<link href="img/favicon.ico" rel="icon">

<!-- Google Font -->
<link
href="https://fonts.googleapis.com/css2?family=Barlow:wght@400;500;600;700;800;900&display=swap" rel="stylesheet">

<!-- CSS Libraries -->
<link href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
rel="stylesheet">
<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css"
rel="stylesheet">
<link href="lib/flaticon/font/flaticon.css" rel="stylesheet">
<link href="lib/animate/animate.min.css" rel="stylesheet">
<link href="lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">

<!-- Template Stylesheet -->
<link href="css/style1.css" rel="stylesheet">

</head>
<body background="img/log_bg.jpg" >

<div class="centerr">
<h1>Login</h1>
<form method="post">
<div class="txt_field">
<input type="text" name="username" required>
```

---

```

        <span></span>
        <label>Username</label>
    </div>
    <div class="txt_field">
        <input type="password" name="password" required>
        <span></span>
        <label>Password</label>
    </div>
    <div class="pass"><a href="changepassword.php">Forgot Password</a></div>
    <input type="submit" name="submit" value="Login">
    <div class="signup_link">
        Not a member? <a href="user_register.php">Signup</a>
    </div>

</form>
</div>

</body>
</html>
<?php
if(isset($_POST["submit"]))
{

$username=$_POST["username"];
$password=$_POST["password"];

$sql2="select * from tbl_login where username='$username' AND password='$password'";

$result=mysqli_query($con,$sql2);

if($result){
    if($row=mysqli_fetch_array($result)){
        if($row[3]=="rto"){
            ?>
            <script type="text/Javascript">
                window.location.href="rtopanel.php";
            </script>
            <?php
            }else if($row[3]=="user"){
                $_SESSION['user']=$row['login_id'];

                ?>
                <script type="text/Javascript">
                    window.location.href="userpanel.php";
                </script>

            <?php

        } else if($row[3]=="subofficer"){

```

---

```

?>
<script type="text/Javascript">
    window.location.href="subpanel.php";
</script>
<?php
} else if($row[3]=="institution"){
?>
<script type="text/Javascript">
    window.location.href="inspanel.php";
</script>
<?php
}
else
{
    echo "Invalid Username and Password";

}
}
}
?>

```

#### **User\_register.php**

```

<?php
include 'dbconnection.php';
if(isset($_POST['submit']))
{
    $fname=$_POST['fname'];
    $lname=$_POST['lname'];
    $gender=$_POST['gender'];
    $email=$_POST['email'];
    $paddress=$_POST['paddress'];
    $caddress=$_POST['caddress'];
    $phone=$_POST['phone'];
    $username=$_POST['username'];
    $pass=$_POST['pass'];
    $repass=$_POST['repass'];

    mysqli_query($con,"insert into
tbl_register(fname,lname,gender,email,paddress,caddress,phone_no,username,password,rep
assword)values('$fname','$lname','$gender','$email','$paddress','$caddress','$phone','$userna
me','$pass','$repass')");
    $last_id=mysqli_insert_id($con);
    mysqli_query($con,"insert into tbl_login values($last_id, '$username','$pass','user','1')") or
die(mysqli_error($con));

```

```
header('location:login.php');

}

?>
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <title>RTO - User Registration</title>
    <meta content="width=device-width, initial-scale=1.0" name="viewport">
    <meta content="Free Website Template" name="keywords">
    <meta content="Free Website Template" name="description">

    <!-- Favicon -->
    <link href="img/favicon.ico" rel="icon">

    <!-- Google Font -->
    <link
href="https://fonts.googleapis.com/css2?family=Barlow:wght@400;500;600;700;800;900&dis
play=swap" rel="stylesheet">

    <!-- CSS Libraries -->
    <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
rel="stylesheet">
    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css"
rel="stylesheet">
    <link href="lib/flaticon/font/flaticon.css" rel="stylesheet">
    <link href="lib/animate/animate.min.css" rel="stylesheet">
    <link href="lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">

    <!-- Template Stylesheet -->
    <link href="css/style1.css" rel="stylesheet">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.js"></script>

<script type="text/javascript">
$(document).ready(function() {
    $("form").submit(function() {

        var validation = $(this); // Select Form
```

```
//var log_type = $("#type");

if (validation.find("[name='fname']").val() == "") {
    alert('Enter First Name');
    return false;
}

if (validation.find("[name='lname']").val() == "") {
    alert('Enter Last Name');
    return false;
}

if (validation.find("[name='gender']").val() == "") {
    alert('Enter Gender');
    return false;
}

if (validation.find("[name='email']").val() == "") {
    alert('Enter a Valid email id');
    return false;
}

if (validation.find("[name='paddress']").val() == "") {
    alert('Enter a Valid Permanent Address');
    return false;
}

if (validation.find("[name='caddress']").val() == "") {
    alert('Enter a Valid Communication Address');
    return false;
}

if (validation.find("[name='phone']").val() == "") {
    alert('Enter a Valid Phone number');
    return false;
}

if (validation.find("[name='username']").val() == "") {
    alert('Enter a Username');
    return false;
}

if (validation.find("[name='pass']").val() == "") {
    alert('Enter a password');
    return false;
}
```

```
}

if (validation.find("[name='repass']").val() == "") {
    alert('Enter confirm password');
    return false;
}

alert('You registered sucessfully');

$("#myform")[1].reset();
});
});
</script>

</head>
<script>
function pageRedirect() {
    window.location.href = "userpanel.php";
}
</script>

<body >
<div class="center">
    <h1>User Registration </h1>
    <form method="post" action="#" onsubmit="return validation(this);">
        <div class="txt_field">
            <input type="text" name="fname" value="" maxlength="50" >
            <span></span>
            <label>First Name</label>
        </div>
        <div class="txt_field">
            <input type="text" name="lname" value="" maxlength="50" >
            <span></span>
            <label>Last Name</label>
        </div>
        <div class="txt_field">
            <input type="text" name="gender" value="" maxlength="50">
            <span></span>
            <label>Gender </label>
        </div>
        <div class="txt_field">
            <input type="text" name="email" value="" maxlength="30" >
            <span></span>
```

```
<label>Email</label>
</div>
<div class="txt_field">
  <input type="text" name="paddress" value="" maxlength="30" >
  <span></span>
  <label>Permanent Address</label>
</div>
<div class="txt_field">
  <input type="text" name="caddress" value="" maxlength="30" >
  <span></span>
  <label>Communication Address</label>
</div>
<div class="txt_field">
  <input type="text" name="phone" value="" maxlength="12">
  <span></span>
  <label>Phone Number</label>
</div>
<div class="txt_field">
  <input type="text" name="username" value="" maxlength="15">
  <span></span>
  <label>User Name</label>
</div>
<div class="txt_field">
  <input type="password" name="pass" value="" maxlength="8" >
  <span></span>
  <label>Password</label>
</div>
<div class="txt_field">
  <input type="password" name="repass" value="" maxlength="8" >
  <span></span>
  <label>Retype Password</label>
</div>
  <input type="submit" name="submit" value="Register">

  Already have a account?<a href="login.html" >Login</a>

</form>
</div>
```

```
</body>
```

```
</html>
<!-- end document-->
```

### Driving\_license.php

```
<?php
session_start();
include'dbconnection.php';
if(!isset($_SESSION['user']))
{
header('location:index.html');
}
$regid=$_SESSION['user'];

if(isset($_POST['verifyemail']))
{
$user=$_POST['email'];
$q=mysqli_query($con,"select * from tbl_register where email='$user'");
$count=mysqli_num_rows($q);
if($count>0)
{
$otp=rand(100000,999999);
$_SESSION['check']=$otp;
$to = $user;
$subject = 'Email Verification';
$message = 'For requesting your services, you have to enter this verification code when
prompted: '.$otp;
$headers = 'From: RTO <teenarose2403@gmail.com>' . "\r\n" .
'X-Mailer: PHP/' . phpversion();

$result=mail($to, $subject, $message, $headers);
if($result)
{
mysqli_query($con,"insert into tbl_drivinglicense(reg_id,email)values('$regid','$user')");

echo "<script>alert('Mail Send
Successfully');window.location.href='drivinglic_2.php';</script>";
}
else
{
echo "<script>alert('Something went wrong...');</script>";
}
}
else
{
echo "<script>alert('Email not found');</script>";
}
```



```

}
?>
<!DOCTYPE html>

<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>RTO - Driving License</title>
    <meta content="width=device-width, initial-scale=1.0" name="viewport">
    <meta content="Free Website Template" name="keywords">
    <meta content="Free Website Template" name="description">

    <!-- Favicon -->
    <link href="img/favicon.ico" rel="icon">

    <!-- Google Font -->
    <link
href="https://fonts.googleapis.com/css2?family=Barlow:wght@400;500;600;700;800;900&di
splay=swap" rel="stylesheet">

    <!-- CSS Libraries -->
    <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
rel="stylesheet">
    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css"
rel="stylesheet">
    <link href="lib/flaticon/font/flaticon.css" rel="stylesheet">
    <link href="lib/animate/animate.min.css" rel="stylesheet">
    <link href="lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">

    <!-- Template Stylesheet -->
    <link href="css/style2.css" rel="stylesheet">

  </head>
  <body background="img/log_bg.jpg">

  <div class="container">
    <div class="title"><b>Driving License Registration<b></div>
    <br />
    <div class="content">
      <form action="" method="post" >
        <div class="input-box">
          <span class="details">Email</span>
          <input type="text" name="email" >
          <input type="submit" name="verifyemail" value="Verify">
        </div>
      </form>
    </div>
  </div>

```

```

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

</body>
</html>

```

### Drivinglicense\_2.php

```

<?php
session_start();
require("dbconnection.php");
$chk=$_SESSION['check'];
if(isset($_POST['otpverify']))
{
    $enteredotp=$_POST['enteredotp'];
    if($enteredotp == $chk)
    {
        echo "Valid OTP verified";
        echo "<script>alert('email verified');window.location.href='drivinglic_verified.php';</script>";
    }
    else{
        echo "<script>alert('invalid otp');</script>";
    }
}

?>

<!DOCTYPE html>

<html lang="en" dir="ltr">
<head>
    <meta charset="utf-8">
    <title>RTO - Driving License</title>
    <meta content="width=device-width, initial-scale=1.0" name="viewport">
    <meta content="Free Website Template" name="keywords">
    <meta content="Free Website Template" name="description">

    <!-- Favicon -->
    <link href="img/favicon.ico" rel="icon">

    <!-- Google Font -->
    <link href="https://fonts.googleapis.com/css2?family=Barlow:wght@400;500;600;700;800;900&display=swap" rel="stylesheet">

```

```

<!-- CSS Libraries -->
    <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
rel="stylesheet">
    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css"
rel="stylesheet">
    <link href="lib/flaticon/font/flaticon.css" rel="stylesheet">
    <link href="lib/animate/animate.min.css" rel="stylesheet">
    <link href="lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">

<!-- Template Stylesheet -->
<link href="css/style2.css" rel="stylesheet">

</head>
<body background="img/log_bg.jpg">

<div class="container">
    <div class="title"><b>Driving License Registration<b></div>
    <br />
    <div class="content">
        <form action="" method="post" >
            <div class="input-box">
                <span class="details">Enter otp recieved</span>
                <input type="text" name="enteredotp" >
                <button type="submit" name="otpverify">Verify</button>
            </div>
        </form>
    </div>

</div>

</div>

</body>
</html>

```

### Drivinglic\_verified.php

```

<?php
session_start();
include'dbconnection.php';
if(!isset($_SESSION['user']))
{
header('location:index.html');
}

$regid=$_SESSION['user'];

if(isset($_POST['register']))

```

```

{
    $fname=$_POST['fname'];
    $lname=$_POST['lname'];
    $age=$_POST['age'];
    $gender=$_POST['gen'];
    $pname=$_POST['pname'];
    $email=$_POST['email'];
    $paddress=$_POST['paddr'];
    $caddress=$_POST['caddr'];
    $phone=$_POST['ph'];
    $ltype=$_POST['ltype'];
    $isdate=$_POST['isdate'];
    $exdate=$_POST['exdate'];
    $blood=$_POST['blood'];

    $filepath=pathinfo($_FILES['file']['name']) ;
    $extension=$filepath['extension'];

    $iname= date('H-i-s').'.$extension;
    $path='img/'.$iname;
    move_uploaded_file($_FILES['file']['tmp_name'],$path);

    $sql=mysqli_query($con,"INSERT INTO `tbl_drivinglicense` (`reg_id`, `first_name`,
`last_name`, `age`, `gender`, `parent_name`, `paddress`, `caddress`, `email`, `phone_no`,
`license_type`, `date_of_issue`, `expiriry_date`, `blood`,`image`,`is_approved`) VALUES
('$regid', '$fname', '$lname', '$age', '$gender','$pname', '$paddress', '$caddress', '$email',
'$phone', '$ltype', '$isdate', '$exdate', '$blood','$iname','NO')") or die(mysqli_error($con));
    if($sql)
        echo "<script>alert('Registered Successfully');</script>";
}

if(isset($_POST['emailverify']))
{
    $email=$_POST['email'];
    $_SESSION['verifyemail']= $email;
    header('location:drivinglic_otp.php');
}
?>
<!DOCTYPE html>

<html lang="en" dir="ltr">
<head>
    <meta charset="utf-8">
    <title>RTO - Driving License</title>
    <meta content="width=device-width, initial-scale=1.0" name="viewport">
    <meta content="Free Website Template" name="keywords">
    <meta content="Free Website Template" name="description">

```

```

<!-- Favicon -->
<link href="img/favicon.ico" rel="icon">

<!-- Google Font -->
<link
href="https://fonts.googleapis.com/css2?family=Barlow:wght@400;500;600;700;800;900&dis
play=swap" rel="stylesheet">

<!-- CSS Libraries -->
<link href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"
rel="stylesheet">
<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css"
rel="stylesheet">
<link href="lib/flaticon/font/flaticon.css" rel="stylesheet">
<link href="lib/animate/animate.min.css" rel="stylesheet">
<link href="lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">

<!-- Template Stylesheet -->
<link href="css/style2.css" rel="stylesheet">

<script>
var _validFileExtensions = [".jpg", ".jpeg", ".bmp", ".gif", ".png"];
function ValidateSingleInput(oInput) {
    if (oInput.type == "file") {
        var sFileName = oInput.value;
        if (sFileName.length > 0) {
            var blnValid = false;
            for (var j = 0; j < _validFileExtensions.length; j++) {
                var sCurExtension = _validFileExtensions[j];
                if (sFileName.substr(sFileName.length - sCurExtension.length,
sCurExtension.length).toLowerCase() == sCurExtension.toLowerCase()) {
                    blnValid = true;
                    break;
                }
            }

            if (!blnValid) {
                alert("Sorry, file is invalid, allowed extensions are: " + _validFileExtensions.join(",
"));
                oInput.value = "";
                return false;
            }
        }
        return true;
    }
}

function check()
{
    if (document.getElementById('age').value < 18 || document.getElementById('age').value >
50)

```

```

{
    document.getElementById('message').style.color = 'red';
    document.getElementById('message').innerHTML = 'Age should be between 18 and 50';
}
else
{
    document.getElementById('message').style.color = "";
    document.getElementById('message').innerHTML = "";
}
}
}

</script>
</head>
<body background="img/log_bg.jpg">

<div class="container">
    <div class="title"><b>Driving License Registration<b></div>
    <br />
    <div class="content">
        <form action="" method="post" enctype="multipart/form-data">
            <div class="user-details">
                <div class="input-box">
                    <span class="details">First Name</span>
                    <input type="text" name="fname" value="<?php echo isset($_POST['fname']) ?
htmlspecialchars($_POST['fname'],ENT_QUOTES): ";>">
                </div>
                <div class="input-box">
                    <span class="details">Last Name</span>
                    <input type="text" name="lname" value="<?php echo isset($_POST['lname']) ?
htmlspecialchars($_POST['lname'],ENT_QUOTES): ";>">
                </div>
                <div class="input-box">
                    <span class="details">Age</span>
                    <input id="age" type="text" name="age" onkeyup="check();" value="<?php echo
isset($_POST['age']) ? htmlspecialchars($_POST['age'],ENT_QUOTES): ";>">
                    <span id='message'></span>
                </div>
                <div class="input-box">
                    <span class="details">Gender</span>
                    <input type="text" name="gen" value="<?php echo isset($_POST['gen']) ?
htmlspecialchars($_POST['gen'],ENT_QUOTES): ";>">
                </div>
                <div class="input-box">
                    <span class="details">Parent Name</span>
                    <input type="text" name="pname" value="<?php echo isset($_POST['pname']) ?
htmlspecialchars($_POST['pname'],ENT_QUOTES): ";>">
                </div>
                <div class="input-box">
                    <span class="details">Permanent Address</span>
                    <input type="text" name="paddr" value="<?php echo isset($_POST['paddr']) ?

```

```

htmlspecialchars($_POST['paddr'],ENT_QUOTES): ";?>">
    </div>
    <div class="input-box">
        <span class="details">Communication Address</span>
        <input type="text" name="cddr" value="<?php echo isset($_POST['cddr']) ?
htmlspecialchars($_POST['cddr'],ENT_QUOTES): ";?>">
    </div>
    <div class="input-box">
        <span class="details">Email</span>
        <input type="text" name="email" value="<?php echo isset($_POST['email']) ?
htmlspecialchars($_POST['email'],ENT_QUOTES): ";?>">
        <span class="details">Verifed</span>
    </div>
    <div class="input-box">
        <span class="details">Phone Number</span>
        <input type="text" name="ph" value="<?php echo isset($_POST['ph']) ?
htmlspecialchars($_POST['ph'],ENT_QUOTES): ";?>">

    </div>
    <div class="input-box">
        <span class="details">License Type</span>
        <input type="text" name="ltype" value="<?php echo isset($_POST['ltype']) ?
htmlspecialchars($_POST['ltype'],ENT_QUOTES): ";?>">
    </div>
    <div class="input-box">
        <span class="details">Date of Issue</span>
        <input type="date" name="isdate" value="<?php echo isset($_POST['isdate']) ?
htmlspecialchars($_POST['isdate'],ENT_QUOTES): ";?>">
    </div>
    <div class="input-box">
        <span class="details">Expiriy Date</span>
        <input type="date" name="exdate" value="<?php echo isset($_POST['exdate']) ?
htmlspecialchars($_POST['exdate'],ENT_QUOTES): ";?>">
    </div>
    <div class="input-box">
        <span class="details">Blood</span>
        <input type="text" name="blood" value="<?php echo isset($_POST['blood']) ?
htmlspecialchars($_POST['blood'],ENT_QUOTES): ";?>">
    </div>
    <div class="input-box">
        <span class="details">Image</span>
        <input type="file" onchange="ValidateSingleInput(this);" name="file" value="<?php
echo isset($_POST['file']) ? htmlspecialchars($_POST['file'],ENT_QUOTES): ";?>">
    </div>

</div>

    <input type="submit" name="register" value="Register">

</form>
</div>

```

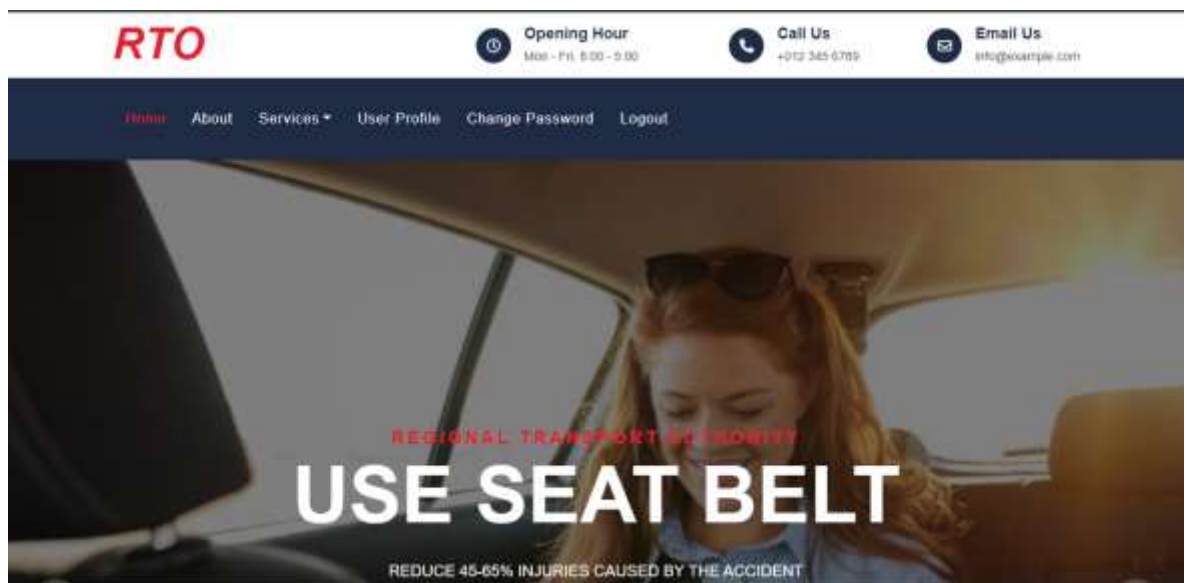
&lt;/div&gt;

&lt;/body&gt;

&lt;/html&gt;

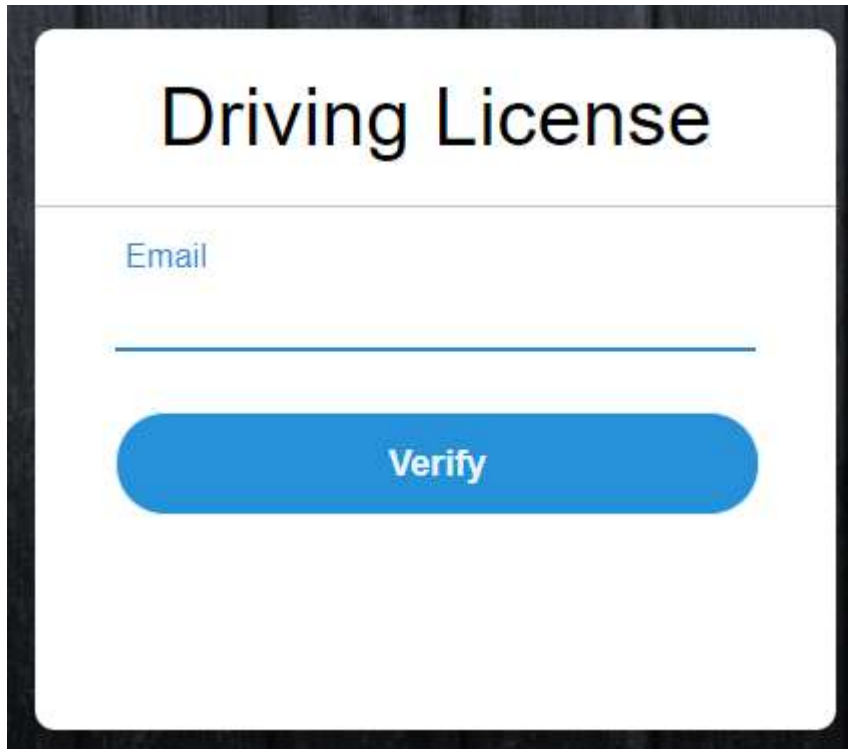
## 9.2 Screen Shots

### User Home page



### Driving License Registration page



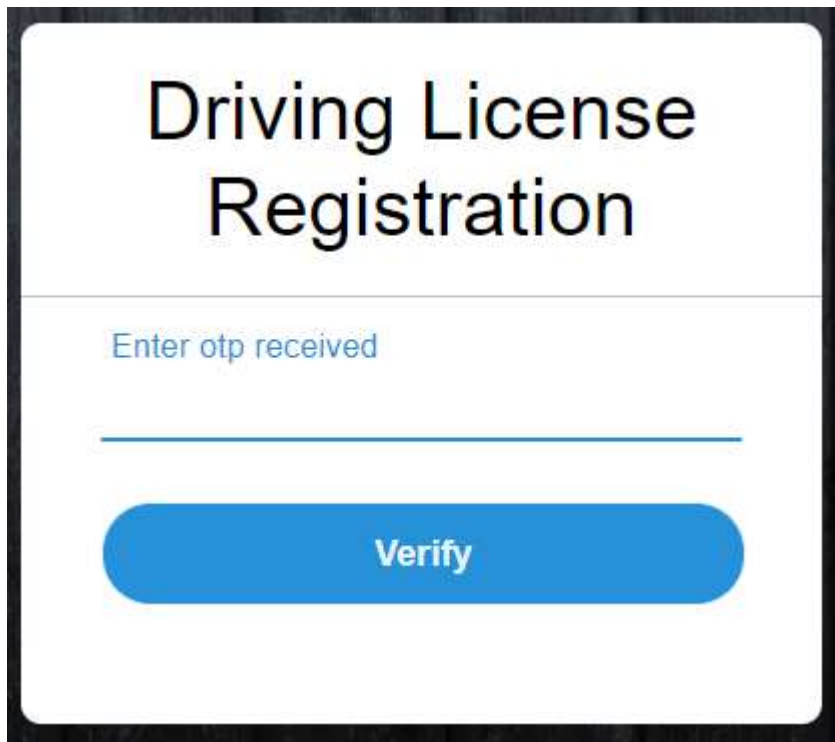


Driving License

Email

Verify

This is a mobile application screen for verifying a driving license. It features a white card with rounded corners on a dark background. The title 'Driving License' is at the top. Below it is a label 'Email' followed by a horizontal input line. At the bottom is a blue rounded button with the text 'Verify'.





Driving License  
Registration

Enter otp received

Verify

This is a mobile application screen for verifying driving license registration. It features a white card with rounded corners on a dark background. The title 'Driving License Registration' is at the top. Below it is a label 'Enter otp received' followed by a horizontal input line. At the bottom is a blue rounded button with the text 'Verify'.

### Driving License Registration

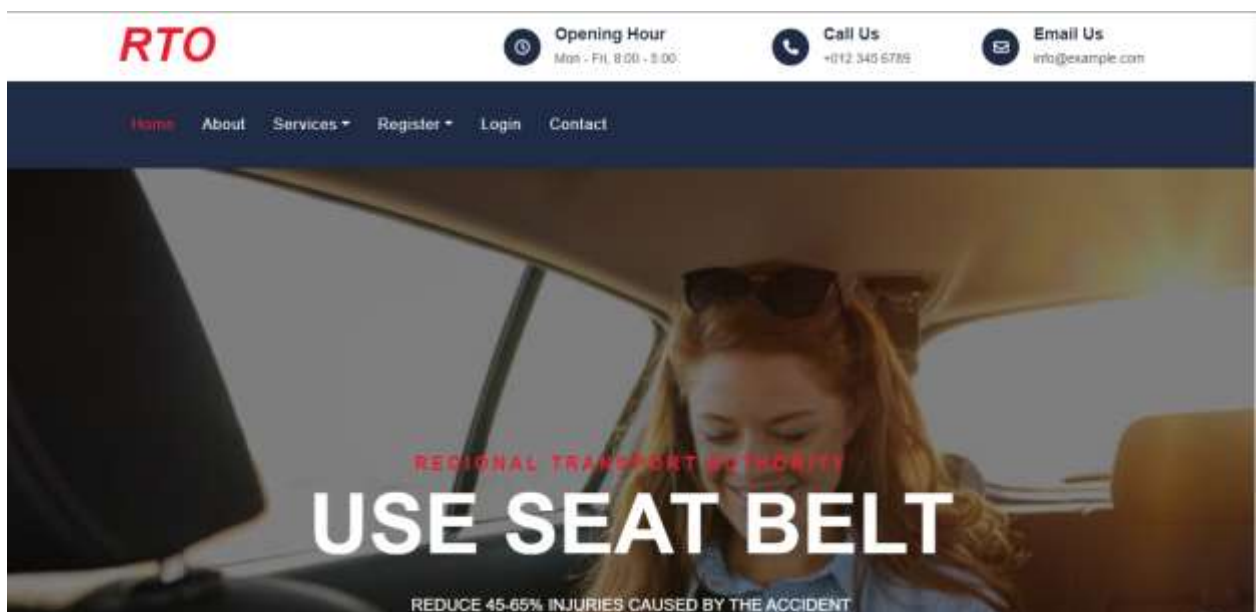
First Name	Last Name
<input type="text"/>	<input type="text"/>
Age	Gender
<input type="text"/>	<input type="text"/>
Parent Name	Permanent Address
<input type="text"/>	<input type="text"/>
Communication Address	Email
<input type="text"/>	<input type="text"/>
	Verified
Phone Number	License Type
<input type="text"/>	<input type="text"/>
Date of Issue	Expiriy Date
<input type="text" value="dd-mm-yyyy"/> 	<input type="text" value="dd-mm-yyyy"/> 
Blood	Image
<input type="text"/>	<input type="button" value="Choose File"/> No f...sen
<input type="button" value="Register"/>	

### View Profile of User

**USER-DETAILS**

First Name :	<input type="text" value="Teena"/>
Last Name :	<input type="text" value="Rose Mathew"/>
Gender :	<input type="text" value="Female"/>
Email :	<input type="text" value="teenarose890@gmail.com"/>
Permanent Address :	<input type="text" value="Malithara(H)"/>
Communication Address :	<input type="text" value="Malithara(H)"/>
Phone Number :	<input type="text" value="9495269697"/>

### User Home Page



## Subofficer Driving license View page

Driving License List

	Name	Age	Gender	Parent Name	Permanent Address	Communication Address	Email	Phone Number	License Type	Date of Issue	Expiry Date	Blood	Image	Action
	Samson Jose	22	Female	Mathew	Malithara(H)	Malithara(H)	ternarosemathew2710@gmail	7034695556	MC With Gear	2022-02-24	2037-02-24	O -ve	05-28-09.jpg	<a href="#">Approve</a> <a href="#">Reject</a>
	Aleena Mathew	22	Female	Mathew	ABC	ABC	trn4749@gmail.com	7034695556	M C with Gear	2022-02-08	2045-02-01	O -ve	18-06-15.jpg	<a href="#">Approve</a> <a href="#">Reject</a>
	Alen J Mathew	25	Male	Mathew	Malithara	Malithara	trn4749@gmail.com	7034695556	M C with Gear	2022-02-09	2029-02-09	A -ve	18-26-17.jpg	<a href="#">Approve</a> <a href="#">Reject</a>
	Ivan Joseph	20	Male	Joseph	ABC	ABC	trn4749@gmail.com	7034695556	M C with Gear	2022-02-18	2022-02-11	A -ve	11-23-12.jpg	<a href="#">Approve</a> <a href="#">Reject</a>
	Anymol Joseph	20	Female	Joseph	ABC	ABC	trn4749@gmail.com	6218290813	M C with Gear	2022-02-15	2037-02-15	O -ve	05-17-00.jpg	<a href="#">Approve</a> <a href="#">Reject</a>