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 Regular 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: 22/02/2022	**TEENA ROSE MATHEW**
KANJIRAPPALLY	**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

In order to make it easier for customers to apply for different permits and registrations, the Regional Transport Authority (RTA) Information System (RTA) was created for the Road Transport Authority. The purpose of this technology is to improve information flow inside the company. RTA offers online license applications, permanent license issuance, new vehicle registration, vehicle ownership transfer, license duplication, multiple-choice tests, license renewal, and payment processing. User can apply to these services and the sub-officer 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.

In the Former System It is not effective at handling office tasks for RTO services since it maintains a local data base and uses a lot of human work that is expensive and time-consuming. Inaccurate reports are not being generated.

When doing transactions, the current system does not provide reliable results. No security is provided; anyone can access the site and conduct their own transactions. The generation of reports is not flexible, and many manual operations have been mechanized.

After system analysis, a new RTO service called "Road Transport Authority System" is suggested to address issues in the current system. The system's goals are to ensure data security and integrity, use less work, Create precise reports and handle details with accuracy.

CONTENT

List of Abbreviation

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

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

A online application called "REGIONAL TRANSPORT AUTHORITY SYSTEM" was created for the Road Transport Authority to make it easier for users to register for different permits and registrations. The purpose of this technology is to improve information flow inside the company. RTA offers online license applications, permanent license issuance, new vehicle registration, vehicle ownership transfer, licenseduplication, MCQs, license renewal, and payment processing. The user can apply for certain services, and the sub-officer can see the user's registration for those services. He has the authority to accept or reject the services, and the RTO receives the authorized registration list for final verification. Following RTO approval, the user can ultimately download the certificates from the application.

The previous system is ineffective at handling office tasks for RTO services because it maintains a local data base and involves a lot of human labour that isexpensive and time-consuming. Inaccurate reports are not being generated. When doing transactions, the current system does not provide reliable results. No security is provided; anyone can access the site and conduct their own transactions. The generation of reports is not flexible, and many manual operations have been mechanized. Following system study, the "Road Transport Authority System" new RTO service is recommended to address problems with the existing system. The proposed system's goals are to: guarantee data security and integrity, use less staff, produce accurate reports, and accurately handle numerous details. Following a system analysis, a new RTO service dubbed "Road Transport Authority System" is recommended to alleviate problems with the current system. Goals of suggested system include ensuring data security and integrity, requiring less staff, creating accurate reports and handling accurate data in various aspects

By employing this system, the user can also save time and effort. This main project has twenty-one tables. Only the five modules user, RTO, institution, instructor, and sub-officer are being created for our main project.

1.2 PROJECT SPECIFICATION

The proposed system is a fix for the current one, which doesn't provide accurate results while conducting transactions. Anyone can use the website and handle their own transactions; there is no security offered. It is not adaptable while creating reports and other manual tasks are automated by computers. After system analysis, a new RTO service called "Regional Transport Authority System" is suggested to address issues inthe current system. The proposed system's goals are to guarantee data security andintegrity, use less staff, produce accurate reports, and accurately handle numerous details.The system keeps track of all transactions made at the RTO office. Printouts for client payments, records of learner license issuances, permanent license issuances, learner license renewals, online LLR forms, registration forms, permanent license issuances, and challan payments are some examples of final outputs.

The system includes five modules.

They are:

1. RTO Module

1.1 Login

The user login page is where the majority of the application's action takes place. This login page is followed by the other modules. Only the user's username and password are recorded by this module. User services may be accepted or rejected by RTO. Only the restricted list of registered services is shown to RTO. Additionally, he has the ability to add institutions and license-related inquiries.

1.2 New license

An individual can only operate a car on Indian highways with a driving license, which is an official document granted by the Indian government. This document attests that the person who is a member of the Regional Transport Authority System is qualified to operate motorized vehicles, such as cars, trucks, bikes, buses, etc., without the need for supervision. Applications that have only been authorized by the sub officer are accessible to the RTO.

1.3 New vehicle registration

Every vehicle that travels on Indian roadways must be registered with and tracked by the Regional Transport Office (RTO). Each state has a large number of RTOs dispersed throughout its length and breadth, and these offices provide residents with a variety of services pertaining to their automobiles. The registration of vehicles is one of the most significant services provided by RTOs.

1.4 Vehicle Ownership transfer

It is a requirement to transfer the ownership of the vehicle to the buyer when you sell your car. The RC transfer, also known as the transfer of vehicle ownership, is crucial because it guarantees that the buyer receives both the vehicle and its associated legal obligations.

1.5 Renewal License

A driving license renewal request may be submitted within 30 days of the expiration date. For non-transport cars, the driving license will be extended for an additional five years, and for transport vehicles, for an additional three years.

1.6 View Institution Vehicle Details

The RTO may determine whether a vehicle is safe to drive, as well as test for pollution and provide engine information.

1.7 Duplicate License

The user must submit an application for a duplicate license if their driver's license is misplaced. The RTO makes the final determination regarding duplicate licenses.

1.8 Add Exam, Remove Exam and Add Ranking

He is able to add the exam by zone. He can also rank the outcomes of the test. He can also discard the test.

1.9 Certificates

He can generate the certificates for the driving license, duplicate license, renewal license, ownership transfer and vehicle registration certificate.

1.10 Appointment

He has the authority to accept or deny the user's request for a renewal appointment for manual document verification.

1.11 Vehicle Number Plate using Image Processing

Here the vehicle number plate is fetched using image processing and also view the details about the vehicle as well as the accident.

1.12 View Generated List

He can view the generated list for the driving license, duplicate license, renewal license, ownership transfer and also vehicle registration.

1.13 Logout

The connection is finally cut off by an admin.

2. Institution Module

This website allows institutions to register. They have the authority to accept or reject requests submitted by users and instructors. The institution module also handles managing instructor leave. Additionally, this module offers traffic signs, fees, and videos about road safety.

2.1 Login

The user login page is where the majority of the application's action takes place. This login page is followed by the other modules. Only the user's username and password are recorded by this module.

2.2 Register

He can register to the site. The applicant is urged to get in touch with the local motor vehicle inspector to learn more about the specific plans that must be completed to set up adriving school and instruct its students in driving. Following receipt of the application, the licensing authority will contact the local motor vehicle inspector to undertake a thorough investigation. If a satisfactory report regarding the arrangements made for the school is obtained, the driving school license will be issued.

2.3 Add Instructors

Institutions can add the instructors to teach driving. Additionally, he has the authority to reject the instructors.

2.4 Add Users

Institutions can add the users. He also has the power to deny access to users.

2.5 Register the vehicle details to the RTA

Here the registration of vehicle is to ensure the vehicle is safe/not.

2.6 Provide Traffic Signs

Road signs or traffic signs are posted along roadways to alert drivers of important information. Instead of using words as symbols, pictorial signs are used. Traffic signs require the complete attention, respect, and appropriate response of drivers since they regulate traffic.

2.7 Road safety videos

He can provide online videos and teach the users how to use the vehicles.

2.8 Fees and user chargers

Institution can provide the fee charges for the basic knowledge.

2.9 Leave Approval

The leave requested by the instructor may also be approved or denied by the institution.

2.10 Logout

The connection is finally cut off by an instructor.

3. User Module

3.1 Services

He can register to all the services like driving license, duplicate license, renewal license, learner’s license, ownership transfer etc.

3.2 Institution registration

He can register to driving school for learning the driving. He can also view the status.

3.3 Attend Exam

He is able to take exams as zone wise.

3.4 Appointment

When a user needs to reschedule an appointment for manual document submission, it happens because they neglected to do so. He can also view the status.

3.5 Download Certificates

He can download the certificates that he applied after it is approved like driving license, duplicate license, renewal license etc. He can also download the affidavit for lost driving license. Then he can apply for the duplicate license with this affidavit.

3.6 User Profile

He can view the profile and update the profile also.

3.7 Applications

He can view the applications and update all the applications as well as the documents. The applications are driving license, duplicate license, renewal license, ownership transfer, vehicle registration and learner’s registration.

3.8 Logout

The connection is finally cut off by an user.

4. Sub officer Module

4.1 Driving License

An individual can only operate a car on Indian highways with a driving license, which is an official document granted by the Indian government. The sub officer has the authority to accept or reject the driver's license.

4.2 New LLR

LLR means learner license registration, it means for granting a person to drive him/her independently in road. The learner’s license may be approved or denied by the sub officer.

4.3 Duplicate License

If the user’s driving license is lost, then the user needs to apply for the duplicate license. The duplicate license may be approved or denied by the sub officer.

4.4 Renewal License

Within 30 days of the license's expiration date, an application for renewal may be submitted. For non-transport cars, the driving licence will be extended for an additionalfive years, and for transport vehicles, for an additional three years. The sub officer has the authority to accept or reject the renewal licence.

4.5 New vehicle registration

Every vehicle that travels on Indian roadways must be registered with and tracked by the Regional Transport Office (RTO). The sub officer has the authority to accept or reject the vehicle registration.

4.6 Vehicle Ownership transfer

It is a requirement to transfer the ownership of the vehicle to the buyer when you sell your car. The RC transfer, also known as the transfer of vehicle ownership, is crucial because it guarantees that the buyer receives both the vehicle and its associated legal obligations. The sub officer has the authority to accept or reject the transfer of vehicle ownership.

4.7 Logout

The connection is finally cut off by an institution.

5. Instructor Module

5.1 Login

He can login to the site.

5.2 Register to Institution

He can register to institution for teaching driving.

5.3 View institution details

He can view the driving school details.

5.4 Apply Institution

He can submit an application for the institution and check the registration status.

5.5 Fees and User Charges

He can also view the fees structure for the basic knowledge.

5.6 Apply Leave

He can apply for a leave of absence and track its progress.

5.7 Logout

He can terminate the connection.

CHAPTER 2

SYSTEM STUDY

2.1 INTRODUCTION

The process of gathering and evaluating data, identifying problems, and using the data to recommend system improvements is known as system analysis. During this problem-solving process, there must be substantial contact between the system developers and users. A system analysis or research should be the first step in any system development process. The system is carefully inspected and evaluated. The system analyst investigates the internal workings of the existing system in the capacity of an interrogator. The input into the system is acknowledged, and the system is seen as a whole. The organizational outcomes can be linked to the various processes. Understanding the issue, identifying the important and vital factors, evaluating and synthesizing the various components, and selecting the best or, at the very least, most acceptable course of action are all part of system analysis.

The procedure needs to be thoroughly investigated utilizing a range of approaches, such as surveys and interviews. The data acquired by various sources needs to be thoroughly evaluated in order to draw a conclusion. Understanding how the system works is the conclusion. The existing system is referred to as such. Now, problem areas have been identified after a detailed examination of the current system. Now the designer takes on the role of a problem-solver and attempts to fix the problems the company is experiencing. The solutions are replaced with proposals. The optimal solution is then picked after being analytically compared to the existing system. The user is given the chance to accept or reject the idea when it is presented to them. In response to user requests, the proposal is assessed and adjusted as needed. Once the user is content with the suggestion, the loop is over.

The process of collecting and analyzing data for use in later system studies is known as preliminary research. Since initial research involves problem-solving, it necessitates close coordination between system users and developers. It carries out several feasibility studies. The system activities are roughly estimated by these studies, which can be used to choose the methods to employ for effective system research and analysis.

2.2 EXISTING SYSTEM

When doing transactions, the current system does not provide reliable results. No security is provided; anyone can access the site and conduct their own transactions. The generation of reports is not flexible, and many manual operations have been mechanized. Current RTO Office job is extremely complicated, a time waster, and more. Taking a driving license as an example, a person must first visit the RTO office, where they assign the task to an agent, who then completes it while collecting a sizable fee. In this method, passing his or her car number, that vehicle's insurance, etc., takes a long time. Additionally, everyone is in a rush these days, therefore we developed a web application that addresses this issue and finds a solution quickly by assessing and taking into account these issues.

2.3 DRAWBACKS OF EXISTING SYSTEM

- It is ineffective at handling office tasks for RTO services.
- It is time-consuming and involves a lot of manual process.
- It is not intuitive.
- Keeps a local database current.
- It doesnot produce accurate reports.

2.4 PROPOSED SYSTEM

We will give a brief overview of our project below given that we are developing a web application for RTO. Allowing the less fortunate user to access this site for work-related RTO reasons will help to create a familiar environment. The user must first complete the registration form. He has access to all services, including those for duplicate licenses, renewals, learners' permits, and ownership transfers, among others. He can enroll in a driving school to learn how to drive. He can see the status as well. He can take tests according to zones. It occurs because a user omitted to arrange a time for manuallysubmitting a document and now needs to do so. He can see the status as well. After they have been authorized, he can download the certificates he applied for, such as a duplicate license, renewed license, and driving license. Additionally, he has access to the missing driver's license affidavit. With this affidavit, he can next submit an application for aduplicate license. He has the ability to access and edit the profile. He can view the applications and update all the applications as well as the documents. The applications are driving license, duplicate license, renewal license, ownership transfer, vehicle registration and learner’s registration. The connection is finally cut off by an user.

Additionally to managing the RTO database and all processes, the administrator also serves as a means of authentication. He has the power to approve the number for atemporary or permanent license, a car registration number, etc. The administrator provides facilities.

Every flaw in the current system is supposed to be fixed by the proposed system. It is necessary to have a system that is easier to use and appealing to people. The RTO, user, sub-officer, instructor, and institution are the system's intended users. RTA offers online license applications, permanent license issuance, new vehicle registration, vehicle ownership transfer, license duplication, MCQs, license renewal, and payment processing.

The user can apply for certain services, and the sub-officer can see the user's registration for those services. He has the authority to accept or reject the services, and the RTO receives the authorized registration list for final verification. Following RTO approval, the user can ultimately download the certificates from the application. RTO is able to control the exam, rankings, etc. He can generate the certificates for the driving license, duplicate license, renewal license, ownership transfer and vehicle registration certificate. He has the authority to accept or deny the user's request for a renewal appointment for manual document verification. He has access to the lists that were automatically generated for the car registration, ownership transfer, duplicate license, and renewal license. Image processing is used to retrieve the license plate number and to display information about the accident and the car.

Instructor can login to the site. He can register to institution for teaching driving. He can view the driving school details. He can submit an application for the institution and check the registration status. He can also view the fees structure for the basic knowledge. He can apply for a leave of absence and track its progress. He can terminate the connection.

Institution can login to the site. He can register to this site. Institutions can add theinstructors to teach driving. Additionally, he has the authority to reject the instructors. Institutions can add the users. He also has the power to deny access to users.

Here the registration of vehicle is to ensure the vehicle is safe/not. Road signs or traffic signs are posted along roadways to alert drivers of important information. Instead of using words as symbols, pictorial signs are used. Traffic signs require the complete attention, respect, and appropriate response of drivers since they regulate traffic. He can provide online videos and teach the users how to use the vehicles. Institution can provide the fee charges for the basic knowledge. The leave requested by the instructor may also beapproved or denied by the institution.

The proposed system aims to create a network of upgraded utilities. The solution minimizes manual work while providing enough security. The suggested technique eliminates paperwork while ensuring data consistency. It offers security, allowing everyone to log in and conduct their own transactions. It may generate reports in a variety of ways, and many laborious operations have been automated.

2.5 ADVANTAGES OF PROPOSED SYSTEM

The system is relatively easy to implement and design. The system works in practically all settings and uses very little system resources. It has got following features:

- **Better security: -**

Unauthorized access must be prevented in order for data to stay safe. Data protection means that they are shielded against different types of erasure. Security, integrity, privacy, and confidentiality are the four connected problems that make up the system security challenge. Security is maintained by requiring a username and password to sign in. As we use secured databases to maintain the papers, it will also ensure data security..

- **Ensure data accuracy: -**

The suggested solution reduces manual mistakes made when entering user information during registration.

- **Better service: -**

The product will avoid the burden of hard copy storage. For performing the same activity, we can also save time and resources. The data can be kept for a longer time without losing any information.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDY

To ascertain if the project will, after completion, achieve the goals of the organization in relation to the work, effort, and time put in it, a feasibility study is carried out. The developer can forecast the project's usefulness and potential future thanks to a feasibility study. The viability of the system concept, which includes its impact on the organization, capacity to satisfy user needs, and efficient use of resources, serves as the basis for a feasibility study. As a result, before a new application is given the go- ahead for development, a feasibility review is typically carried out.

The document describes the project's viability and includes a variety of elements, such as its technical, financial, and operational viabilities, that were carefully considered throughout this project's feasibility assessment. The following are its features: -

3.1.1 Economical Feasibility

Cost-benefit analyses are required to support the evolving system. must make sure that the project receiving the most attention is the one that will yield the best results the quickest. One of the variables is the cost associated with establishing a new system.

Some crucial financial inquiries made during the preliminary investigation include the following:

- The costs involved in conducting a complete system examination
- The cost of the software and hardware.
- The benefits in terms of lower costs or less costly errors.

The suggested solution was established as part of a project, thus there are no manual costs associated with it. Additionally, the availability of all necessary resources suggests that the system might be implemented at a reasonable cost..

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

A technical evaluation of the system is necessary first. An overview design of the system's requirements in terms of input, output, programmes, and procedures must serve as the foundation for the viability assessment. The investigation must then suggest thekind of equipment, the processes required to develop the system, and ways to operate the system once it has been designed after establishing an outline system. The followingtechnical difficulties came up throughout the investigation:

- ➢ Does the proposed technology function with existing technology?
- ➢ Can the system expand with improvements?

The project should be planned so that the necessary functionality and performance are met within the constraints. The project uses cryptographic methods and calls for a high resolution scanning device. The system may still be used even though the technology may become outdated after a while because a newer version of the same software still works with an earlier version. Therefore, this project only has a few limitations. The system was developed using PHP for the front end and a MySQL server for the back end, thus technically the project may be finished. Technically, the project can be completed because the system was created utilizing PHP for the front end and a MySQL server for the back end. 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 following inquiries are part of the suggested system:

- ➢ Is there enough assistance for the users?
- ➢ Will the suggested system harm anyone?

The project would be advantageous because, when created and implemented, it would achieve the goals. The project is deemed to be behaviorally feasible after carefully weighing all behavioural factors.

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 used for general-purpose programming and web development. PHP is now used by 2.1 million web servers and more than 244 million webpages. The reference version of PHP, which Rasmus Ledorf created in 1995, is now created by the PHP group. PHP:HypertextPreprocessor is the current meaning of the recursive acronym PHP, which previously stood for personal home page. The PHP processor module on a web server translates PHP code to create the finished web page. PHP commands can be directly inserted into an HTML source file to handle data rather than calling an external file. The GNU General Public License is incompatible with PHP since it has evolved to incorporate a command-line interface capability and can be used independently as a result of usage restrictions on the term PHP (GPL).The majority of web servers allow for the free deployment of PHP, which can also be used independently as a shell on almost all platforms and operating systems.

3.3.2 MySQL

Oracle Corporation created, distributed, and provided support for MySQL, the most well-known Open Source SQL database management system. On the MySQL website, you may find the most latest information about the MySQL programme.

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

  A planned collection of data is called a database. Anything might be it, including a straightforward shopping list, a photo gallery, or the enormous amount of data in a company network. The data stored in a computer database must be added to, accessed, and managed using a database management system, such as MySQL Server. Database management systems—whether used as standalone programmes or as a component of other applications—are essential to computing because computers are so adept at processing massive volumes of data.

- **MySQL databases are relational.**

  A relational database stores the data in separate tables rather than consolidating it into one enormous warehouse. The database structures are kept in physically fast-loading files. The logical model, which consists of objects like databases, tables, views, rows, and columns, provides a flexible programming environment. One-to-one, one-to-many, unique, required or optional, and "pointers" between other tables are just a few examples of the rules you may make to control the relationships between various data fields. Your application won't ever encounter inconsistent, duplicate, orphan, out-of-date, or missing data since a well-designed database enforces these limitations. The "Structured Query Language" prefix "SQL" in MySQL stands for this phrase. The most popular standard language for accessing databases is SQL. SQL is the most widely used standard language for database access. Depending on your programming environment, you might openly enter SQL (for example, to generate reports), embed SQL statements within other languages' code, or use a language-specific API that conceals the SQL syntax. The ANSI/ISO SQL Standard defines SQL. Since its inception in 1986, the SQL standard has seen a lot of modifications. The 1992 standard is referred to in this document as "SQL92," the 1999 standard is referred to as "SQL,"and the most recent version of the standard is referred to as "SQL: 2003."The SQL Standard as it exists at any one time is referred to as "the SQL

- **MySQL software is Open Source.**

  Anyone can use and modify the software because it is open source. TheMySQL software is available for free download and usage online by anyone. You are free to examine the source code and modify it as necessary. The GPL (GNU General Public License) is used by the MySQL software to specify whatyou are allowed to do and are not allowed to do with the software in certain circumstances. You can purchase a commercially licensed version from us if the GPL bothers you or if you need to include MySQL code into a product that will be sold for a profit. For further details, see the MySQL Licensing Overview. Fast, dependable, scalable, and user-friendly describe the MySQL Database Server.

  If it is what you're after, you should try it. MySQL Server may run easily on a desktop or laptop and needs little to no maintenance in addition to your other apps, web servers, and other software. If you dedicate an entire system to MySQL, you can change the settings to utilise all the RAM, CPU, and I/O power.

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

  The client/server system known as the MySQL Database Software includes a multi-threaded SQL server, a number of client programmers and libraries, administration tools, and a wide range of application programming interfaces (APIs). Additionally, we provide MySQL Server as a multi-threaded built-in library that you can link into your product to create a standalone service that is more manageable, speedy, and simple to use.

CHAPTER 4

SYSTEM DESIGN

4.1 INTRODUCTION

Any engineered system or product's development process begins with design. Design is a creative process. The secret to an efficient system is a decent design. Design is the process of completely describing a system or process so that it can be implemented physically utilizing a range of concepts and approaches. The process of specifying a tool, a procedure, or a system in enough detail to make it physically possible utilizing a variety of approaches and concepts. Software design serves as the technical cornerstone of the software engineering process, regardless of the development paradigm used. The architectural information needed to construct a system or product is produced by the system design. As with any rigorous methodology, our programme underwent the best design phase conceivable, fine-tuning all efficiency, performance, and accuracy levels. During the design stage, a user-focused document transitions into a document for programmers or database professionals. Logical design and physical design are the two phases of system design development.

4.2 UML DIAGRAM

The artefacts of the software system are specified, visualized, constructed, and documented using a standard language called UML. A draught of the UML 1.0 specification was provided to the Object Management Group (OMG), which was in charge of creating UML, in January 1997.

Unified Modeling Language is known as UML. Unlike other well-known programming languages like C++, Java, COBOL, etc., UML has certain unique features. A visual language called UML is used to create software blueprints. A general-purpose visual modelling language for software system visualization, specification, construction, and documentation is what UML is known as. UML is not just used to represent software systems, despite the fact that this is its most common application. It is also used to model systems that are not software-based. For instance, the manufacturing facility's process flow, etc. Although UML is not a programming language, there are tools that can convert UML diagrams into code in a variety of languages. The analysis and design of objects-oriented systems are intimately tied to UML.UML has been standardized to the point where it is now an OMG standard. A comprehensive UML a system is made up of all the elements and relationships. Themost crucial aspect of the entire procedure is the UML diagram's aesthetic impact. It is completed by using all the additional components. The following nine diagrams are part of UML.

- 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

An illustration of the interactions between system components is a use case diagram. An instrument for locating, classifying, and arranging system needs is the use case. The word "system" in this sense refers to something that is being built or operated, such as a website for online shopping and mail-order services. UML (Unified Modeling Language), a standard language for modelling actual products and systems, employs use case diagrams. Planning general requirements, validating hardware designs, testing and debugging software products while they are still in development, creating online help resources, and finishing customer support-focused tasks are a few examples of system objectives. For instance, customer support, item ordering, catalogue updating, and payment processing are examples of use cases in a setting of product sales. A use case diagram consists of four components.

- The boundary, which isolates the system of interest from its surroundings.
- The performers, who are typically system participants identified by the roles they play.
- The actors within and around the system play the roles specified by the use cases.
- The connections and interactions between the actors and use cases.

Use case diagrams are created to demonstrate a system's functional requirements. The principles listed below must be followed after the aforementioned components have been identified in order to create a successful use case diagram.

- A use case's naming plays a crucial role. The name should be selected so that it can specify the functionalities carried out.
- Give the actors names that fit them.
- Clearly depict links and dependencies in the diagram.
- Keep in mind that the diagram's primary function is to indicate the needs; do not attempt to include all possible relationships.
- When necessary, take notes to help you remember some crucial details.

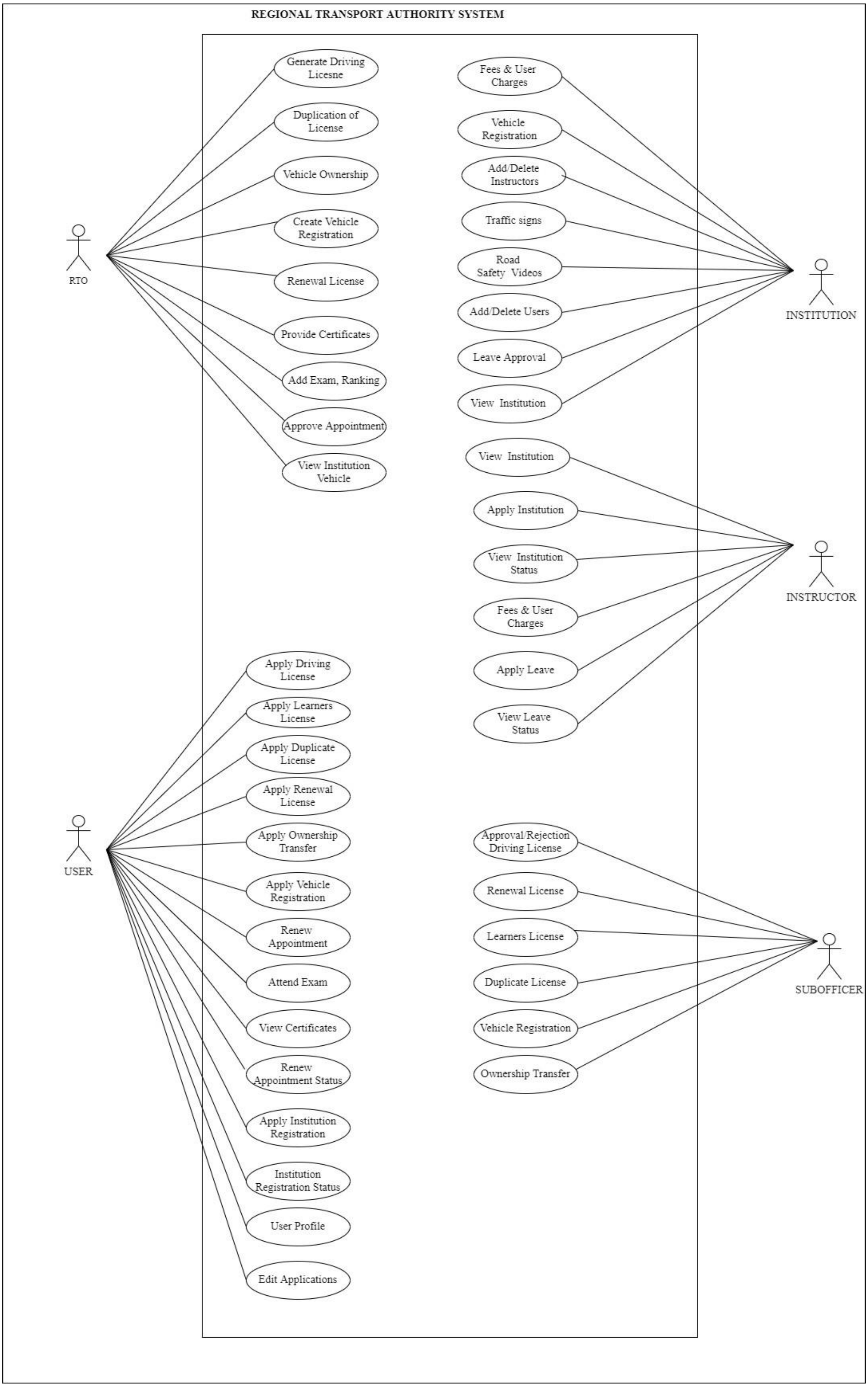


Fig 1: Use case diagram for Regional Transport Authority System

4.2.2 SEQUENCE DIAGRAM

A sequence diagram fundamentally depicts the order in which events occur or how they relate to one another. Event diagrams and event scenarios are other names for sequence diagrams. Sequence diagrams display the sequential actions performed by a system's parts. These diagrams are frequently used by businesspeople and software engineers to explain and record the requirements for both new and existing systems.

Sequence Diagram Notations –

i. **Actors –** In a UML diagram, an actor represents a particular kind of role that it plays in relation to other system parts. An actor is never included in the UML diagram's depiction of the system we want to explain. We use actors for a variety of roles, including those of human users and other external subjects. In a UML diagram, an actor is depicted using the stick person notation. Multiple actors could be present in a sequence diagram.

ii. **Lifelines –** A named component called a lifeline identifies a specific participant in a sequence diagram. A lifeline essentially stands in for each incident in a sequence diagram. In a sequence diagram, the lifeline components are at the top.

iii. **Messages –** It is shown how things can communicate with one another through messages. On the lifeline, the messages are presented in reverse chronological order. Arrows are how messages are represented. A sequence diagram's main components are lifelines and messages.

The following categories serve as general classifications for messages:

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

- Lost Message

iv. Guards – We use guards to model conditions in the UML. They are employed when it is necessary to restrict communication while giving the impression that a condition has been satisfied. Software engineers rely on guards to inform them of system or process restrictions.

Uses of sequence diagrams –

- Employed to model and illustrate the reasoning behind a complex function, process, or procedure.
- They are also employed to display the specifics of UML use case diagrams.
- Used to comprehend the precise operation of present or upcoming systems.
- Imagine the flow of information between different system elements or objects.

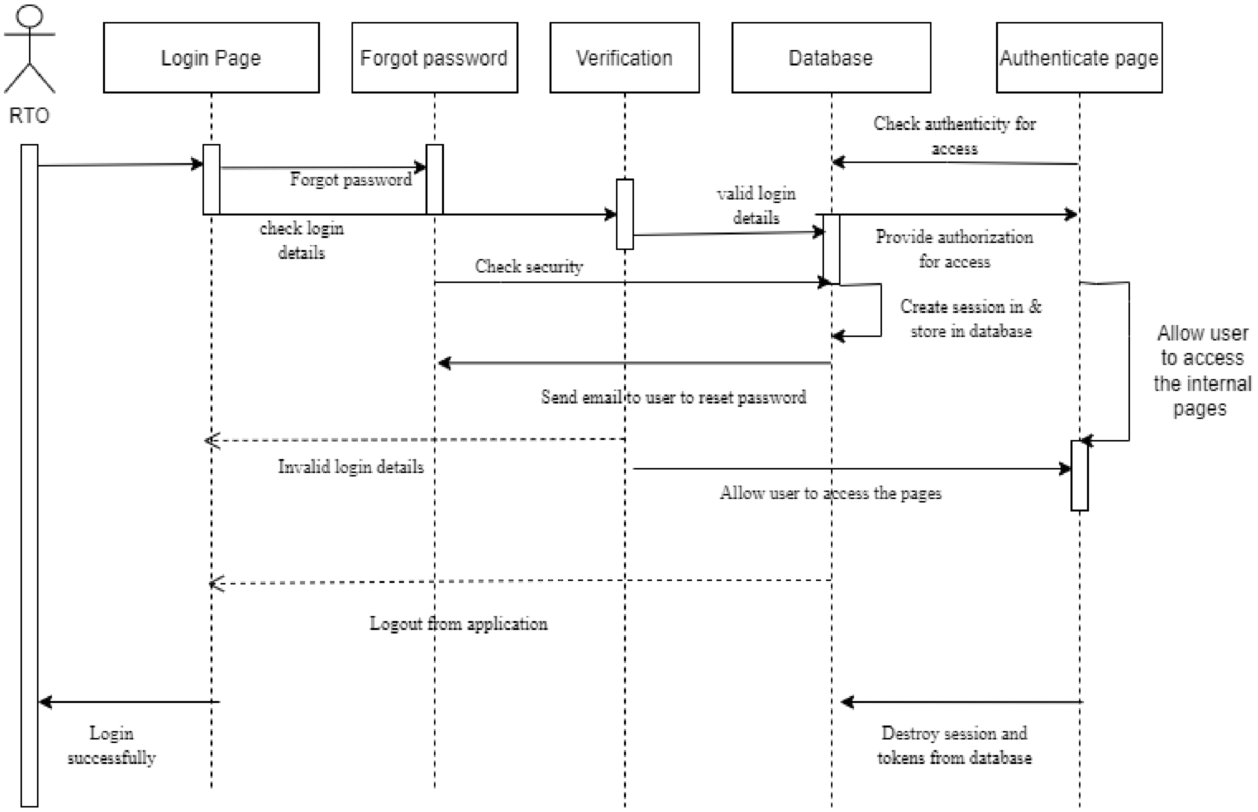

RTO
Login Page
Forgot password
Verification
Database
Authenticate page
Check authenticity for access
Forgot password
valid login details
check login details
Provide authorization for access
Check security
Create session in & store in database
Allow user to access the internal pages
Send email to user to reset password
Invalid login details
Allow user to access the pages
Logout from application
Login successfully
Destroy session and tokens from database

Fig 2: Sequence diagram for Regional Transport Authority System

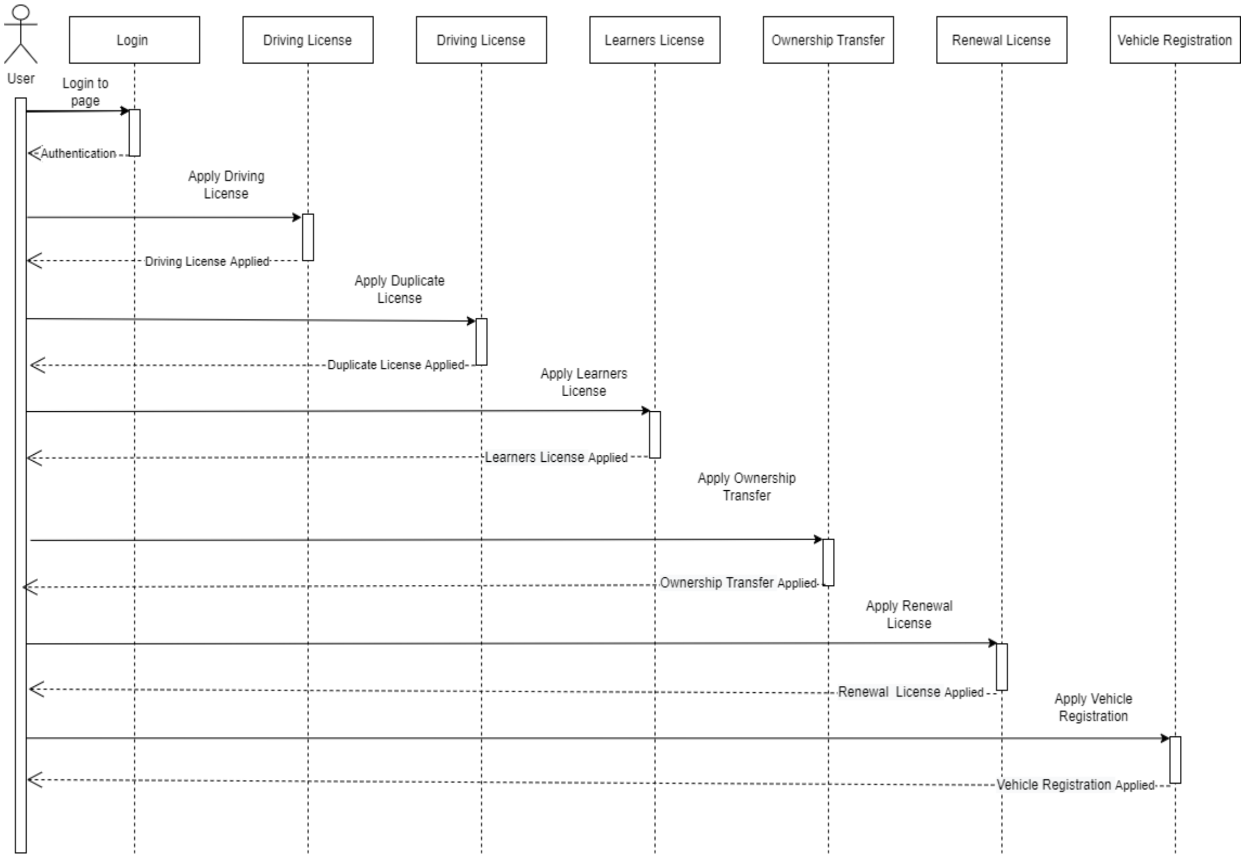


Fig 3(Continuation): Sequence diagram for User in Regional Transport Authority System

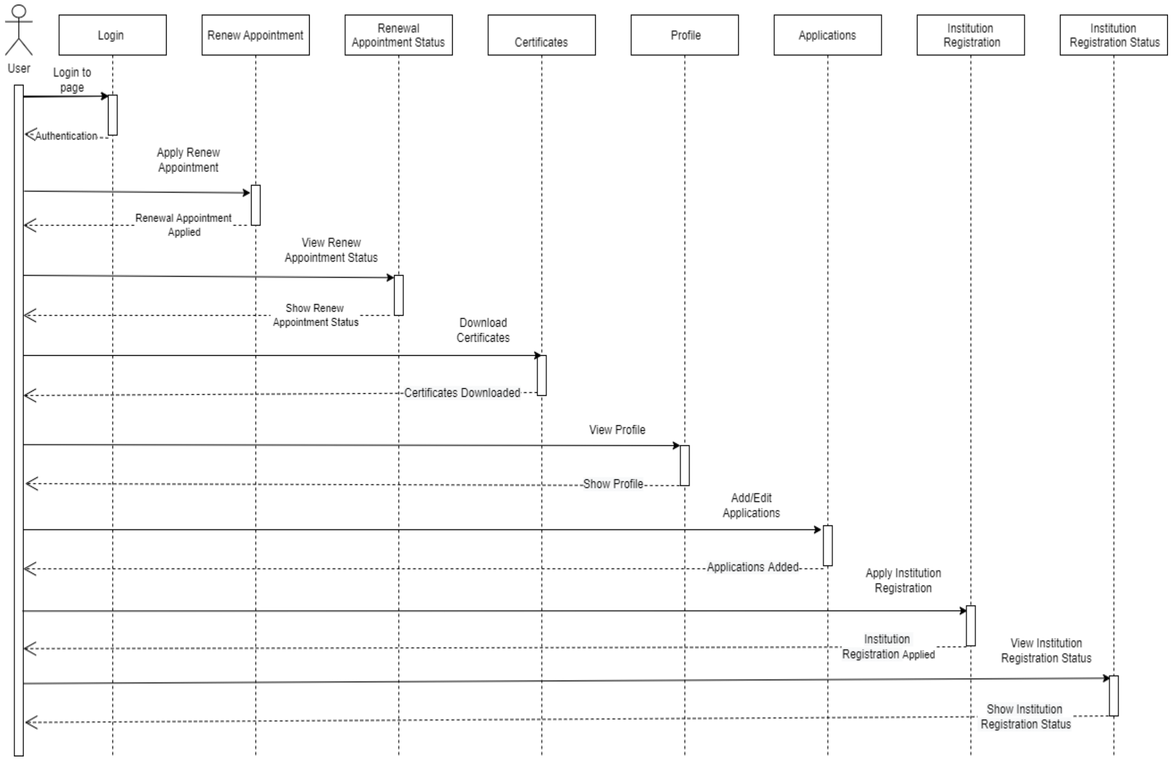

User
Login
Renew Appointment
Renewal Appointment Status
Certificates
Profile
Applications
Institution Registration
Institution Registration Status
Login to page
Authentication
Apply Renew Appointment
Renewal Appointment Applied
View Renew Appointment Status
Show Renew Appointment Status
Download Certificates
Certificates Downloaded
View Profile
Show Profile
Add/Edit Applications
Applications Added
Apply Institution Registration
Institution Registration Applied
View Institution Registration Status
Show Institution Registration Status

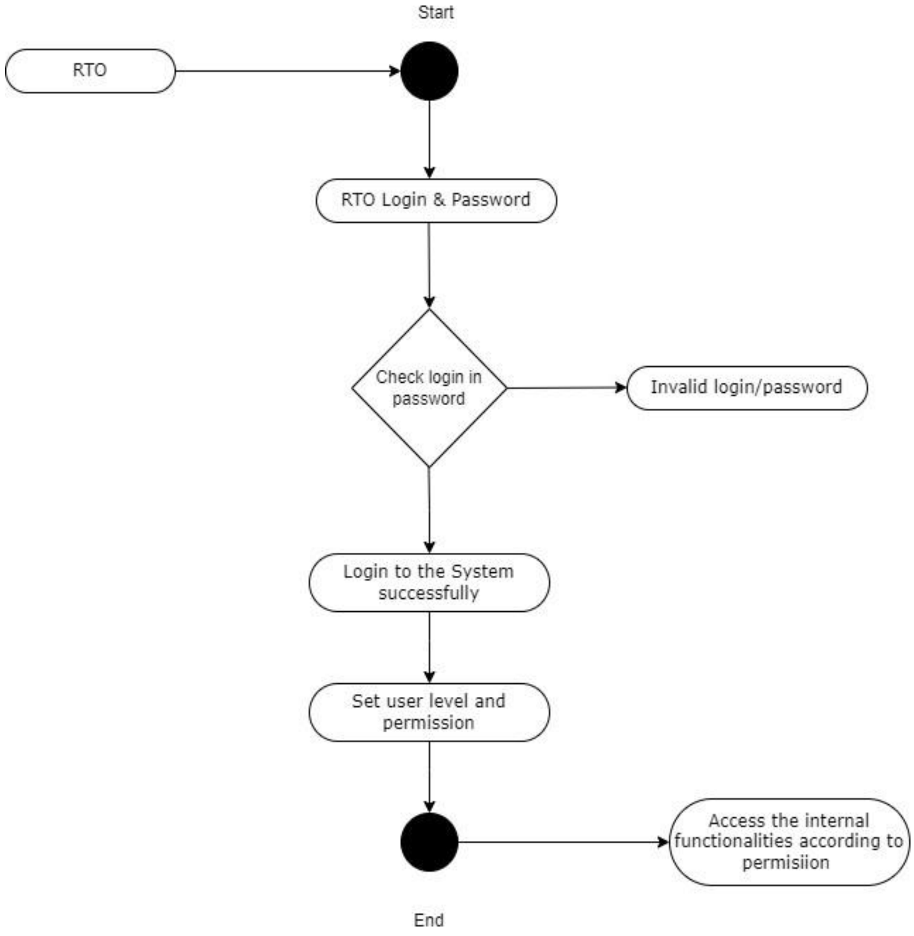
Fig 4(Continuation): Sequence diagram for User in Regional Transport Authority System

4.2.3 ACTIVITY DIAGRAM

The activity diagram describes the dynamic properties of the system. A flowchart that demonstrates how one activity leads to another is called an activity diagram. A system operation might be used to describe the action. One action follows another in the control flow. It is possible for this flow to be concurrent, parallel, or branched. Fork, join, and other elements are frequently used in activity diagrams to manage different sorts of flow control.

Activity diagrams are used to illustrate a system's dynamic nature as well as to perform forward and reverse engineering on the executable system. The message section is the sole element that the activity diagram is missing. No evidence exists to support the idea that data is sent from one action to another. On occasion, a flowchart will be replaced by an activity diagram. The diagrams are not flowcharts, despite their appearance.

Fig 5: Activity diagram for Login in Regional Transport Authority System


Start
RTO
RTO Login & Password
Check login in password
Invalid login/password
Login to the System successfully
Set user level and permission
Access the internal functionalities according to permisiion
End

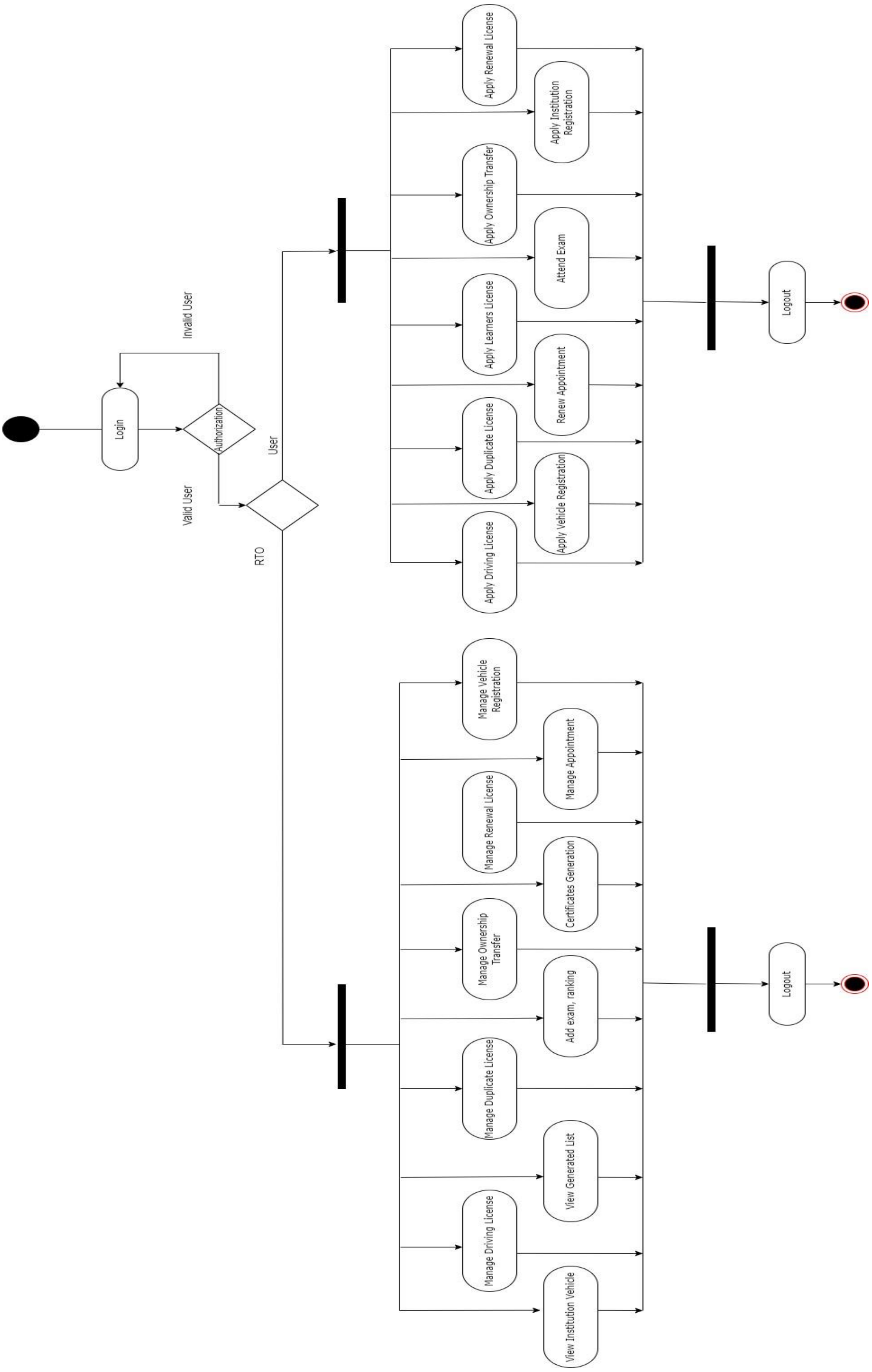


Fig 6(Continuation): Activity diagram for RTO and User in Regional Transport Authority System

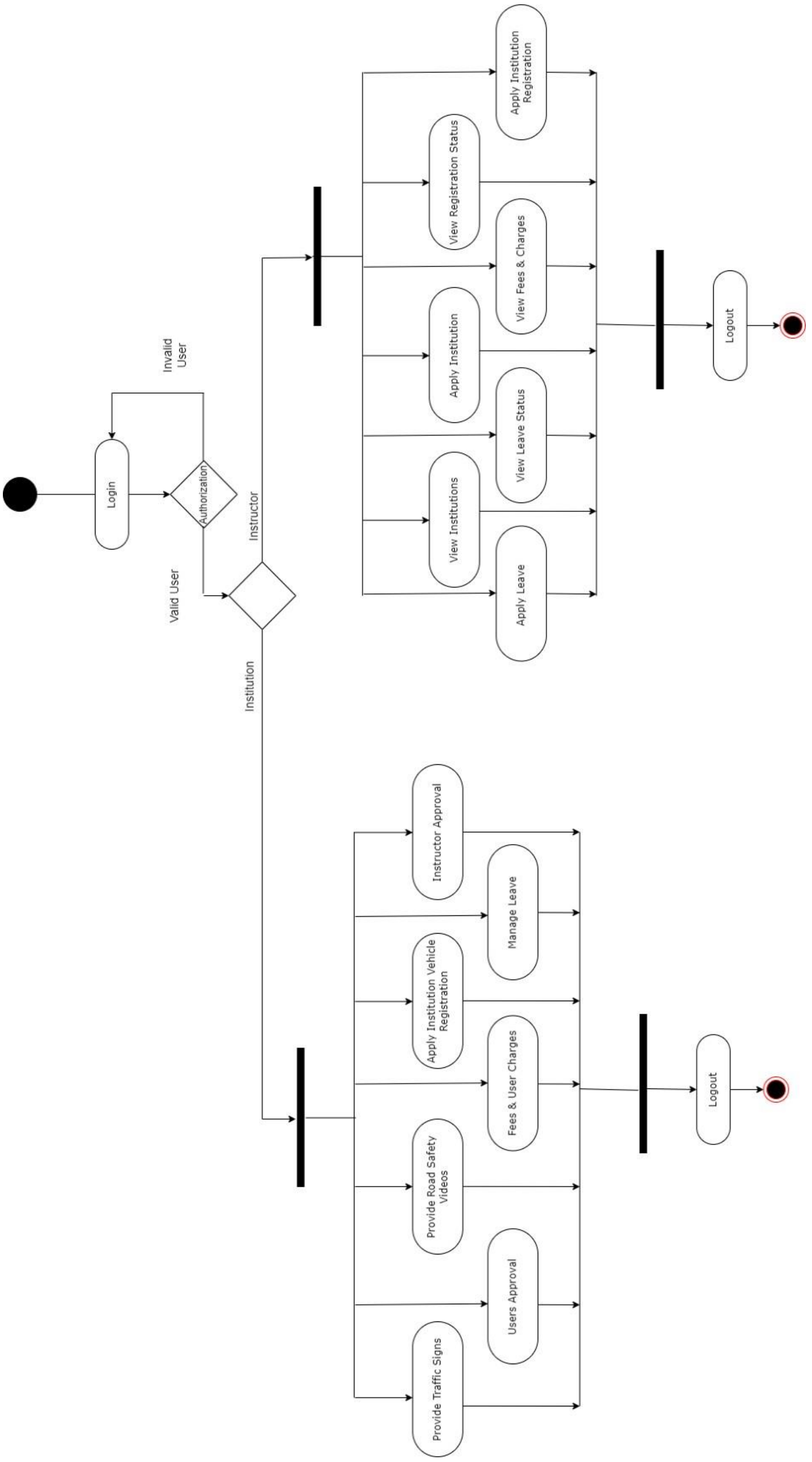


Fig 7(Continuation): Activity diagram for Institution and Instructor in Regional Transport Authority System

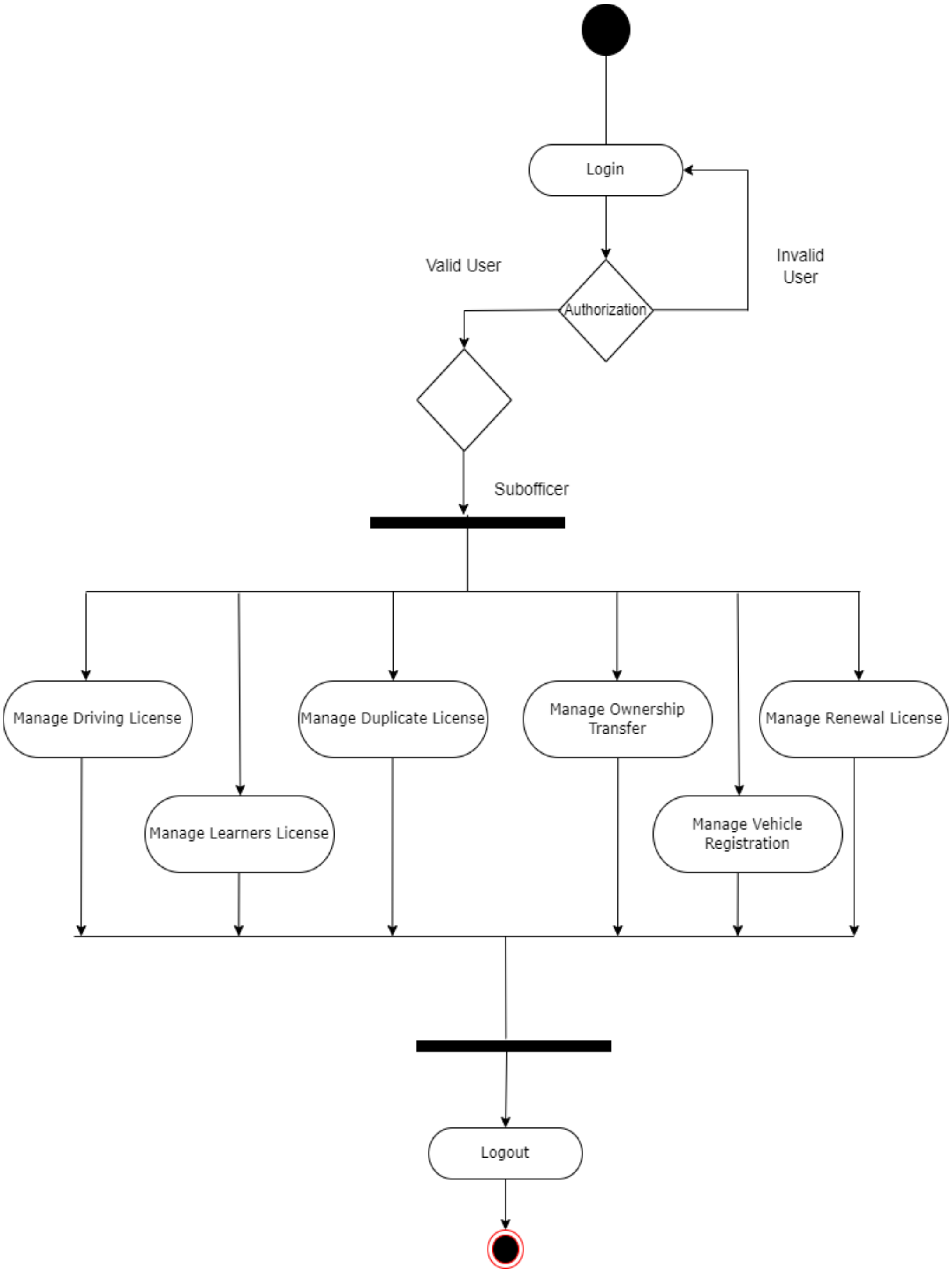


Fig 8(Continuation): Activity diagram for Sub-officer in Regional Transport Authority System

4.2.4 CLASS DIAGRAM

Class diagrams are a type of static diagram. It represents the application's static view. Class diagrams are employed in the creation of executable code for software programs as well as the visualization, clarification, and documentation of various system components. A class diagram depicts the qualities and functions of a class along with the limitations imposed on the system. Class diagrams are often used in the modelling of object-oriented systems since they are the only UML diagrams that can be directly mapped with object-oriented languages. A class diagram shows several classes, interfaces, connections, partnerships, and limitations. It is also known as a "structural diagram."

The objective of the class diagram can be summed up as follows:

- Analysis and design of an application's static view
- Describe a system's obligations.
- Component and deployment diagrams' starting point.
- Forward and reverse engineering.

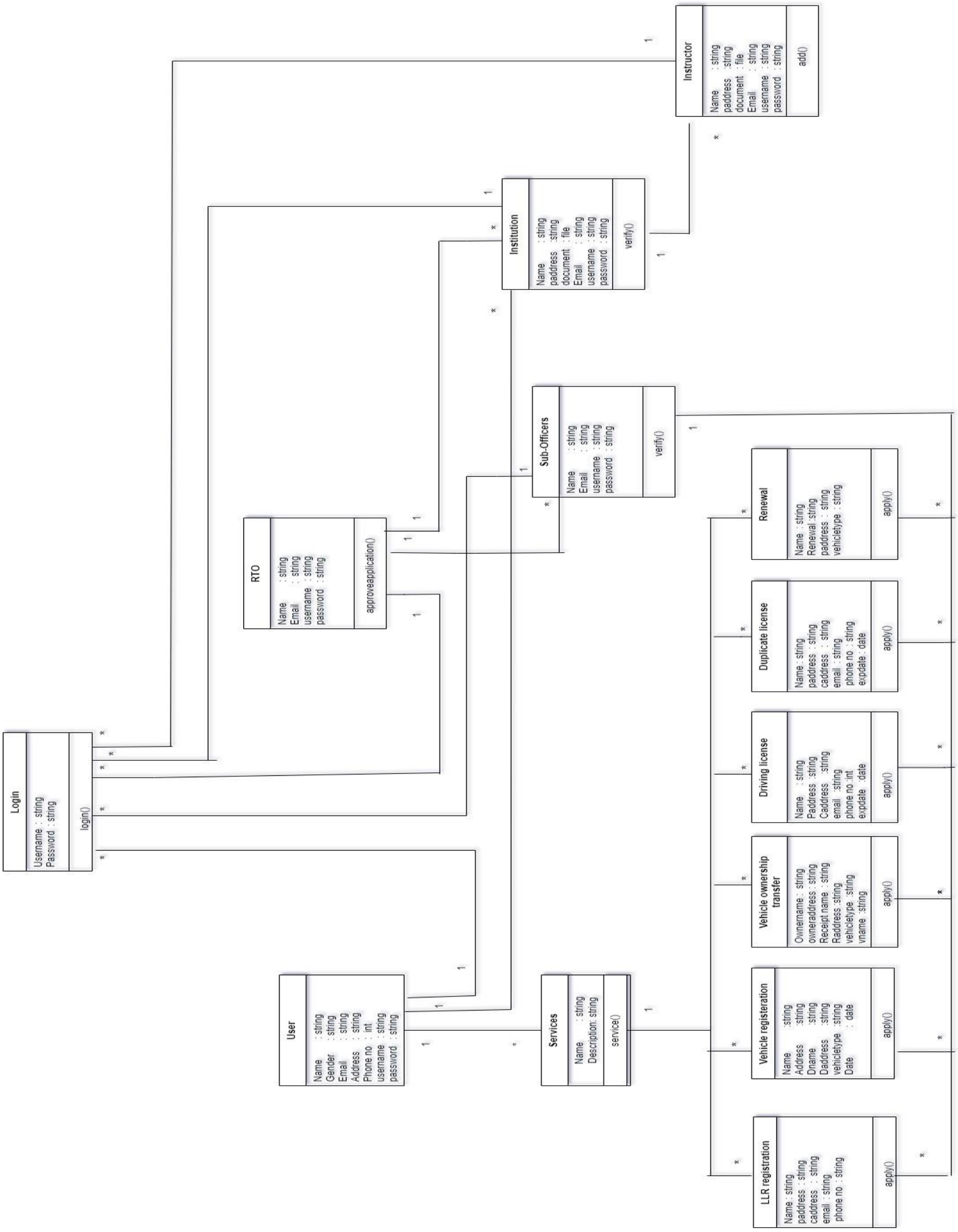


Fig 9: Class diagram of Regional Transport Authority System

4.2.5 COMPONENT DIAGRAM

A specific type of UML diagram is a component diagram. In addition, the goal is distinct from that of the earlier diagrams. Although it doesn't explain the system's functionality, it does describe the parts that go into creating that functionality. Therefore, from that perspective, component diagrams are utilized to represent the actual physical componentsof a system. These elements consist of files, libraries, and packages, among others. Another way to think of component diagrams is as a static implementation perspective of a system. Ina static implementation, the configuration of the components is displayed at a particular time. A collection of diagrams are used because one component diagram cannot adequately represent the overall system.

The component diagram's goal can be summed up as –

- Identify the parts of a system visually.
- Utilize both forward and reverse engineering to build an executable.
- Describe how the elements are arranged and connected.

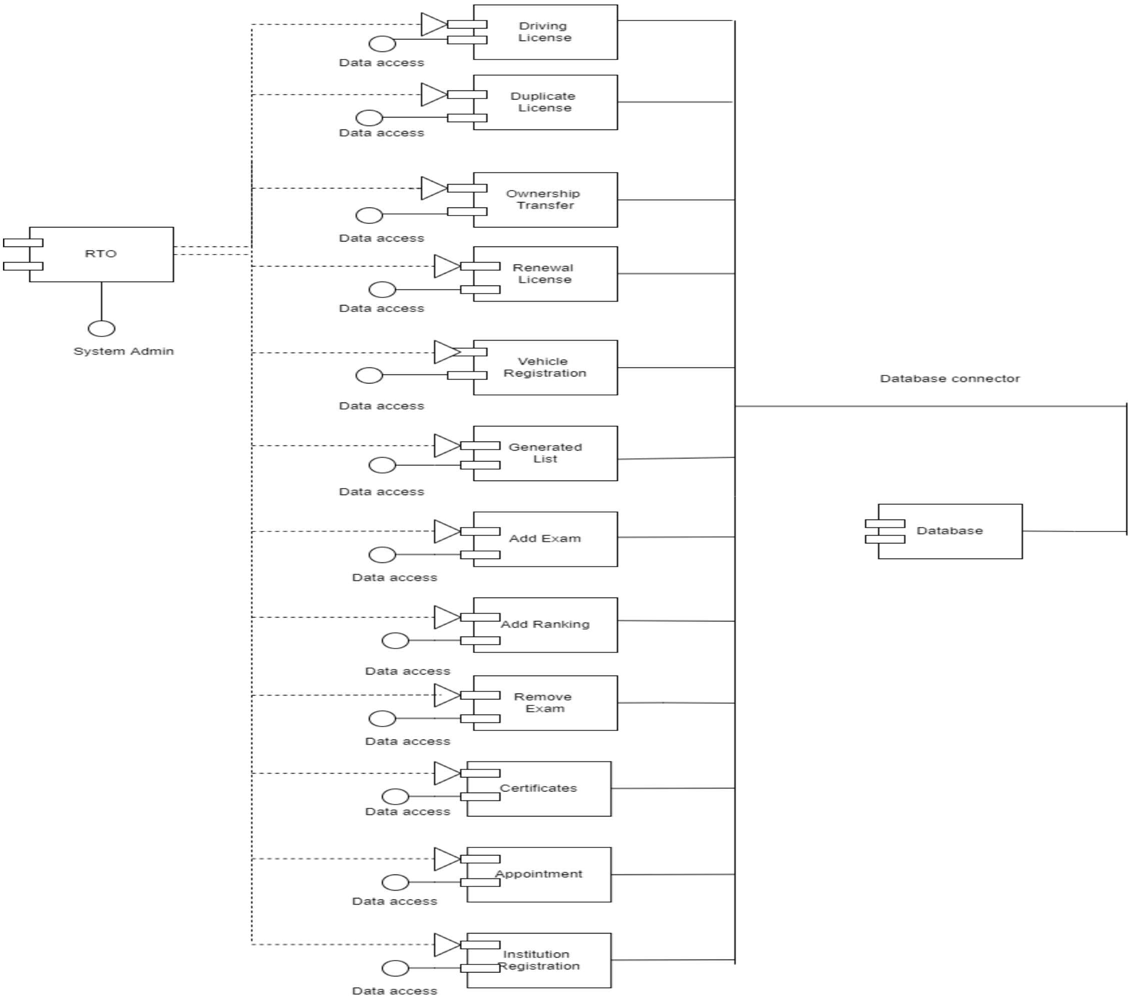


Fig 10: Component diagram of Regional Transport Authority System

4.2.6 OBJECT DIAGRAM

Class diagrams are a requirement for object diagrams because they are the source of class diagrams. An object diagram illustrates a specific instance of a class diagram. Diagrams of classes and objects share the same underlying concepts. The static view of a system, which is a quick snapshot of the system, can also be represented by object diagrams. Object diagrams are used, for instance, to show a group of items and links between them.

The objective of the object diagram is:

- Forward and reverse engineering.
- The system's object relationships
- A static view of an interaction.
- Recognize object behaviour and its relationship from a practical standpoint.

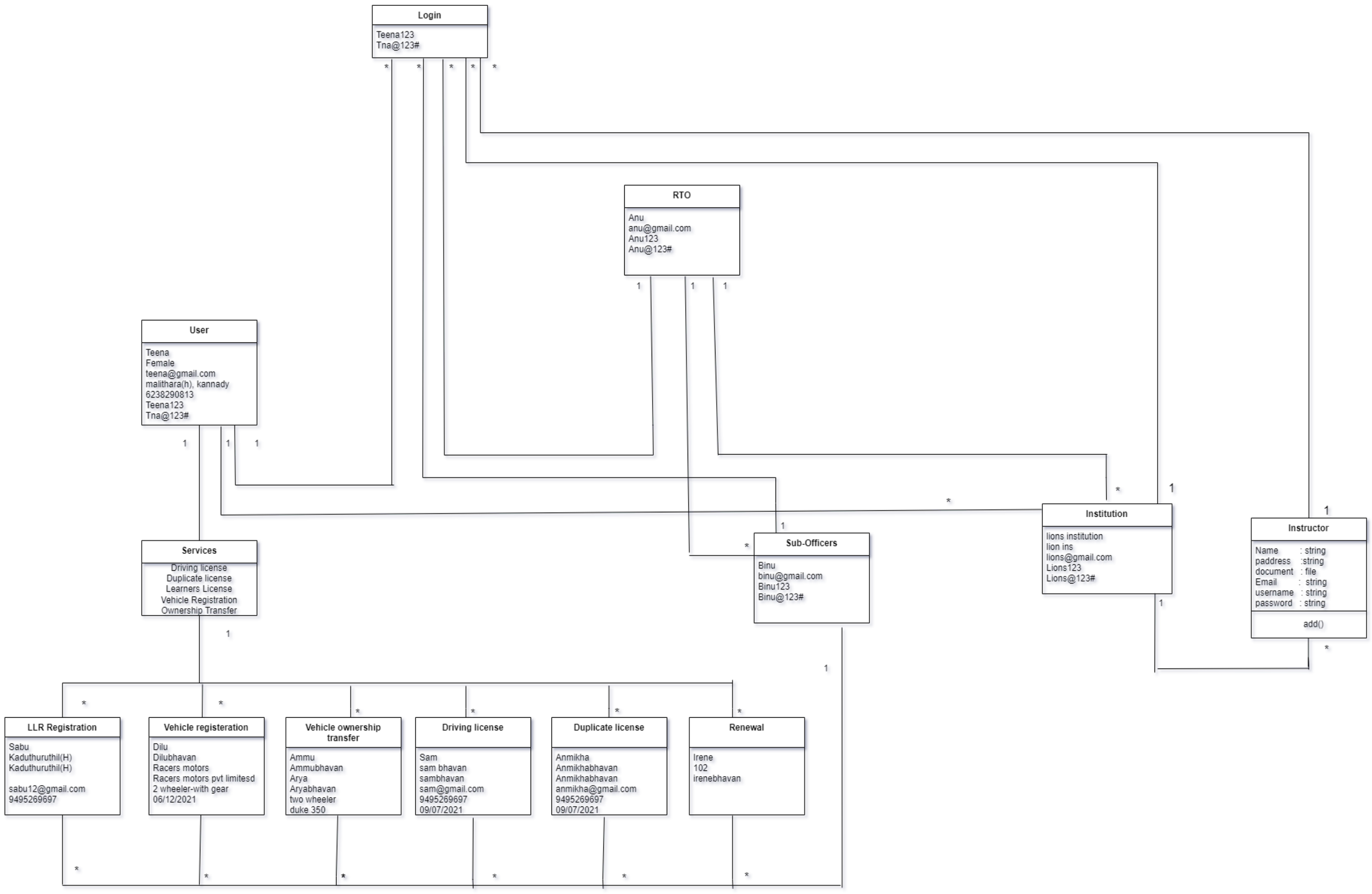


Fig 11: Object diagram of Regional Transport Authority System

4.2.7 DEPLOYMENT DIAGRAM

Deployment diagrams display the topology of a system's physical components, along with the locations of the software components. The static deployment view of a system is described using deployment diagrams. The main elements of deployment diagrams are nodes and the connections between them. It chooses how to load the software onto the hardware. In the real system architecture, where the software will function as a node, it connects the software architecture created during the design process. Since there are many nodes involved, communication channels are employed to show the link.

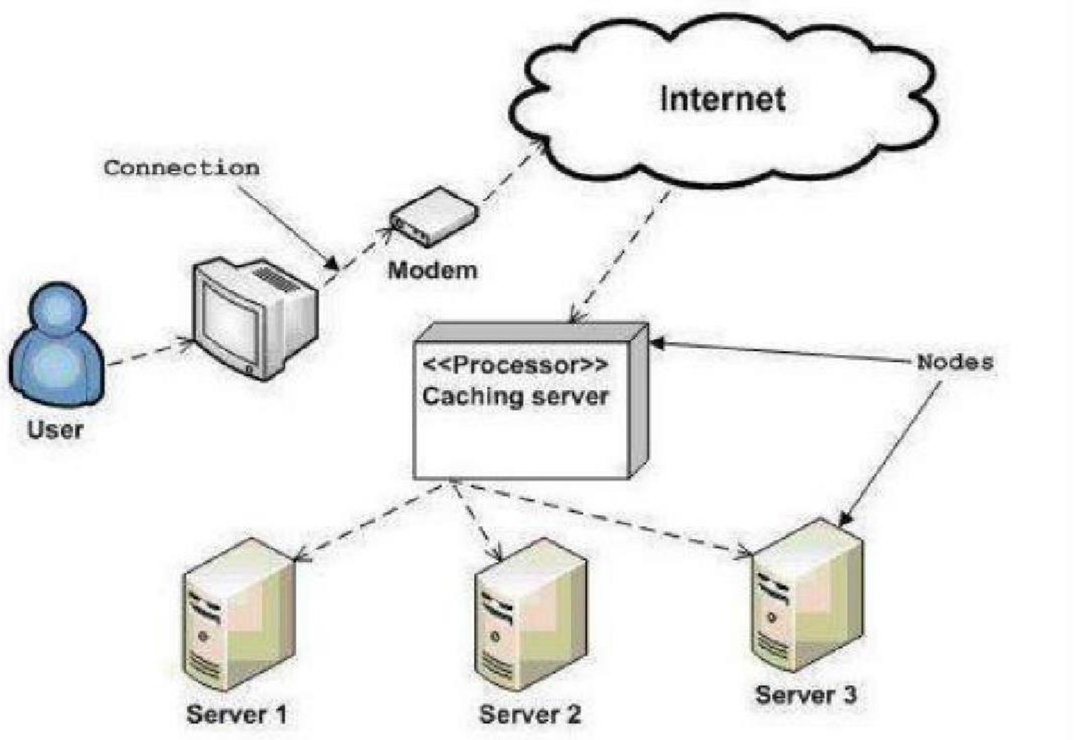


Fig 11: Deployment diagram of Regional Transport Authority System

4.2.8 STATE DIAGRAM

It describes various system components' statuses. Each system item or component has anown set of states. A state machine is depicted in a state chart diagram. State machines are devices that divide up an object's various states and control them in response to internal or external events. Throughout an object's existence, they define several states, and these states are altered by events. The reactive systems can be modelled with state chart diagrams. A system that reacts to internal or external events is known as a reactive system. The transfer of control from one state to another is depicted in a state chart diagram. States are described as a situation in which an object existing and undergoes a change in responseto an event. A state chart diagram's principal objective is to simulate an object's existence from creation to annihilation. For both forward and backward engineering of a system, state chart diagrams are employed. However, the main goal is to simulate the reactive system.

The major goals of using State chart diagrams are as follows:

- To mimic the dynamic nature of a system.
- To represent the lifetime of a reactive system.
- To list the various circumstances that an object might encounter throughout its lifetime.
- Construct a state machine to represent the states of an item.

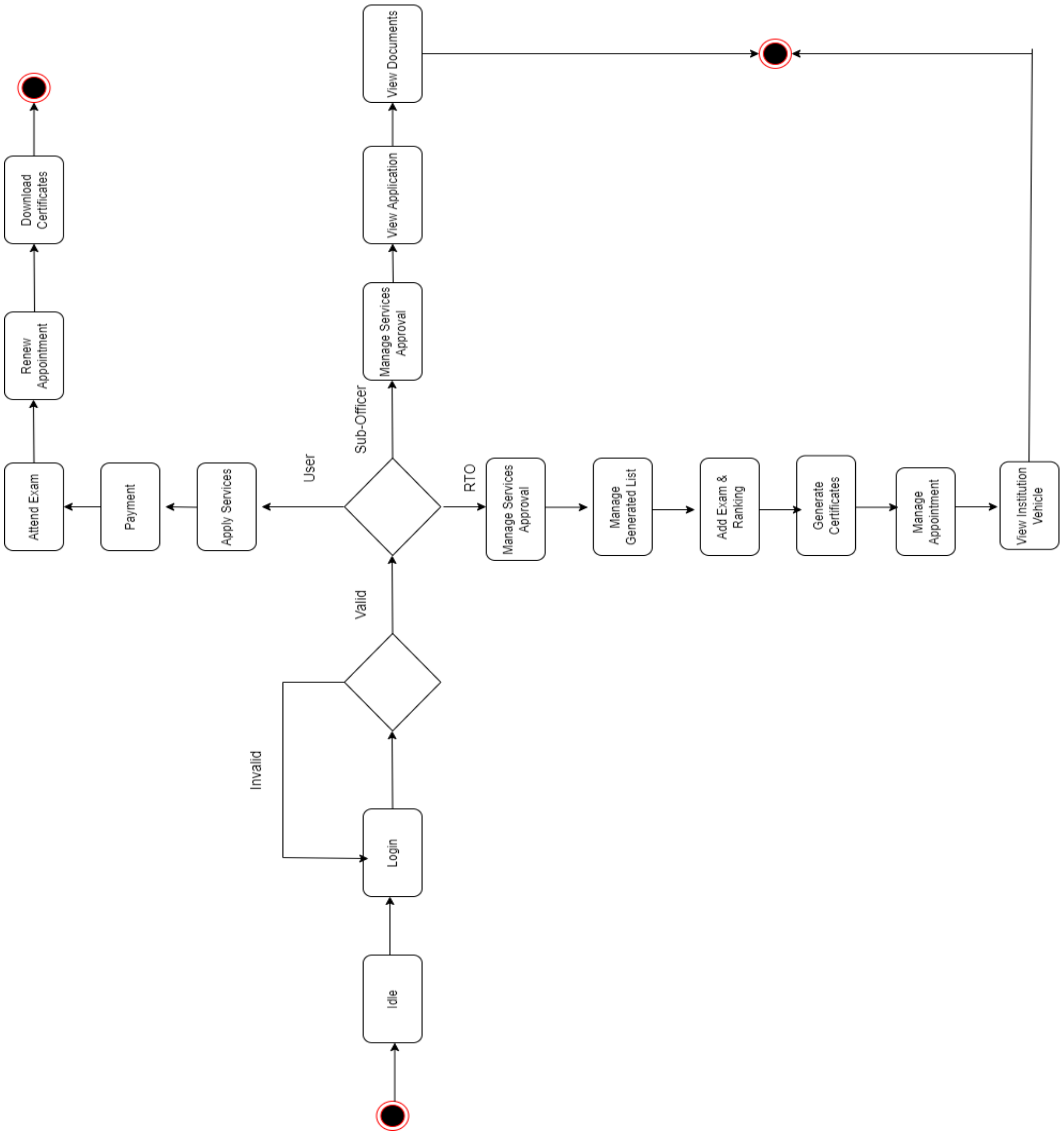
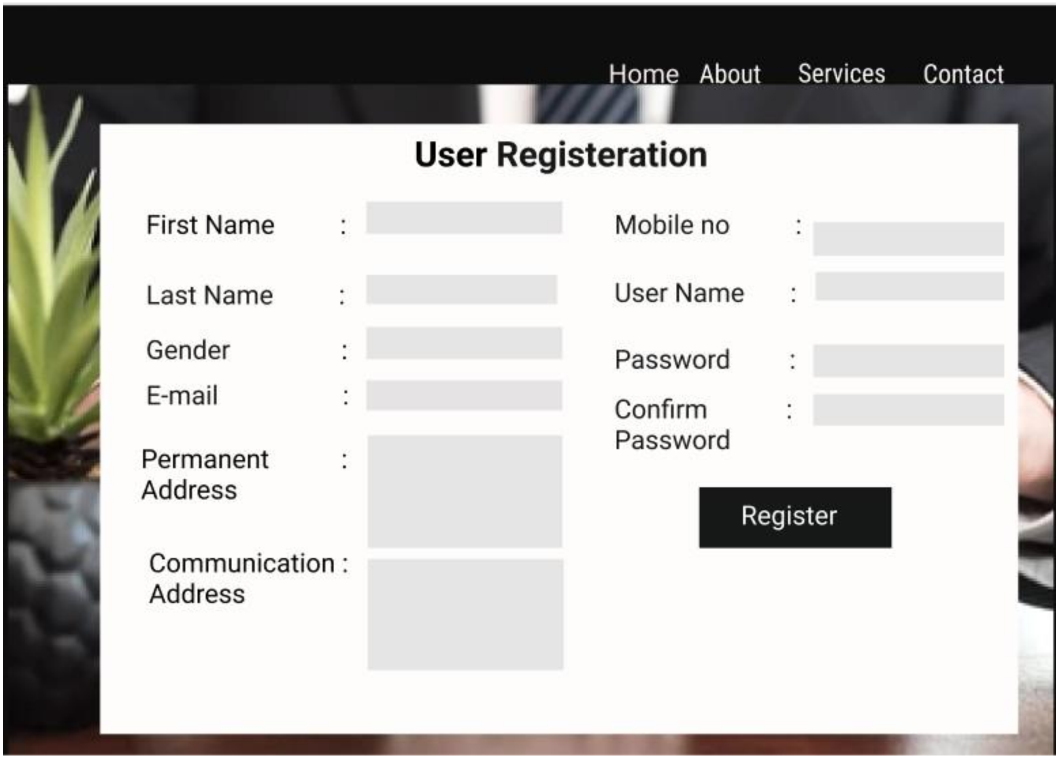


Fig 12: State diagram of Regional Transport Authority System

4.3 USER INTERFACE DESIGN

4.3.1 INPUT DESIGN

Form Name : User Registration



User Registeration

First Name :

Last Name :

Gender :

E-mail :

Permanent Address :

Communication Address :

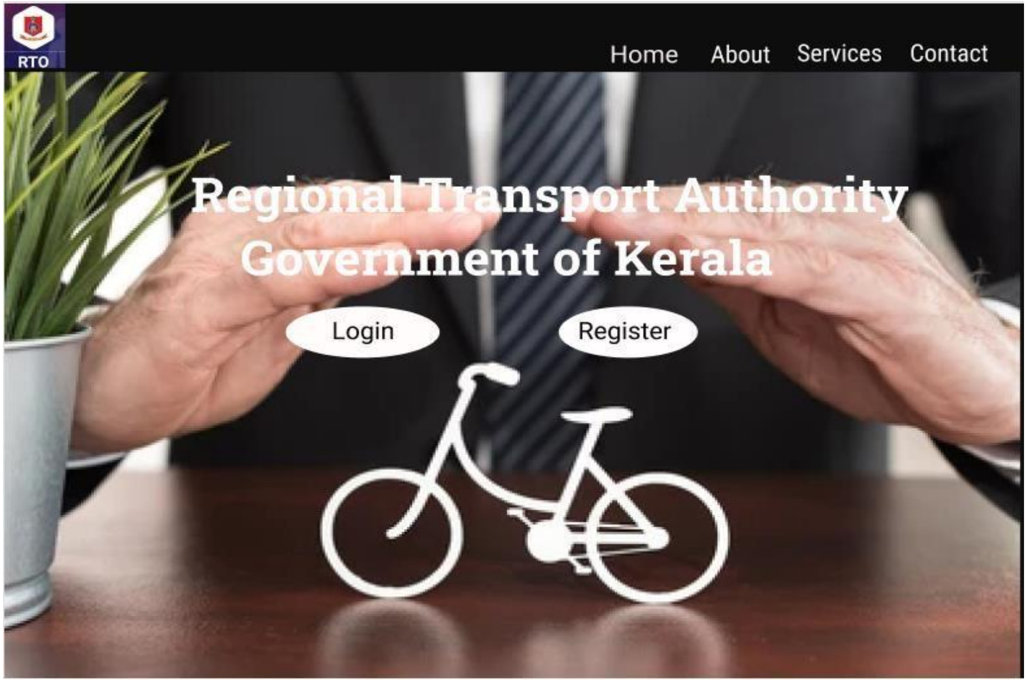
Mobile no :

User Name :

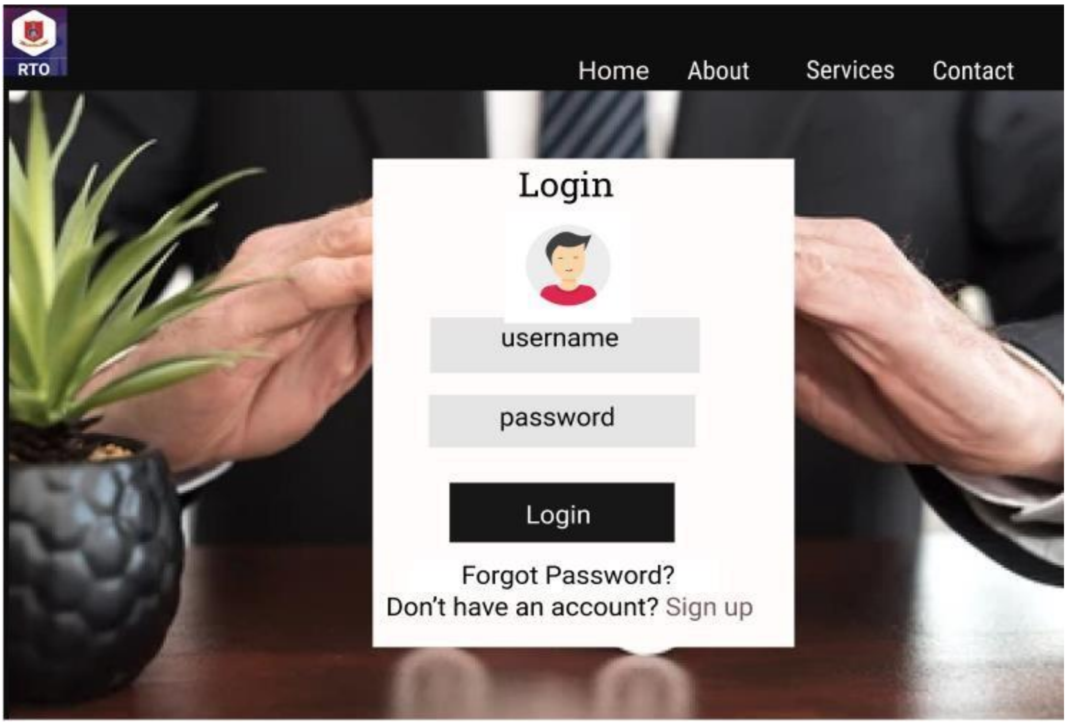
Password :

Confirm Password :

Form Name : User Homepage



Form Name : User Login



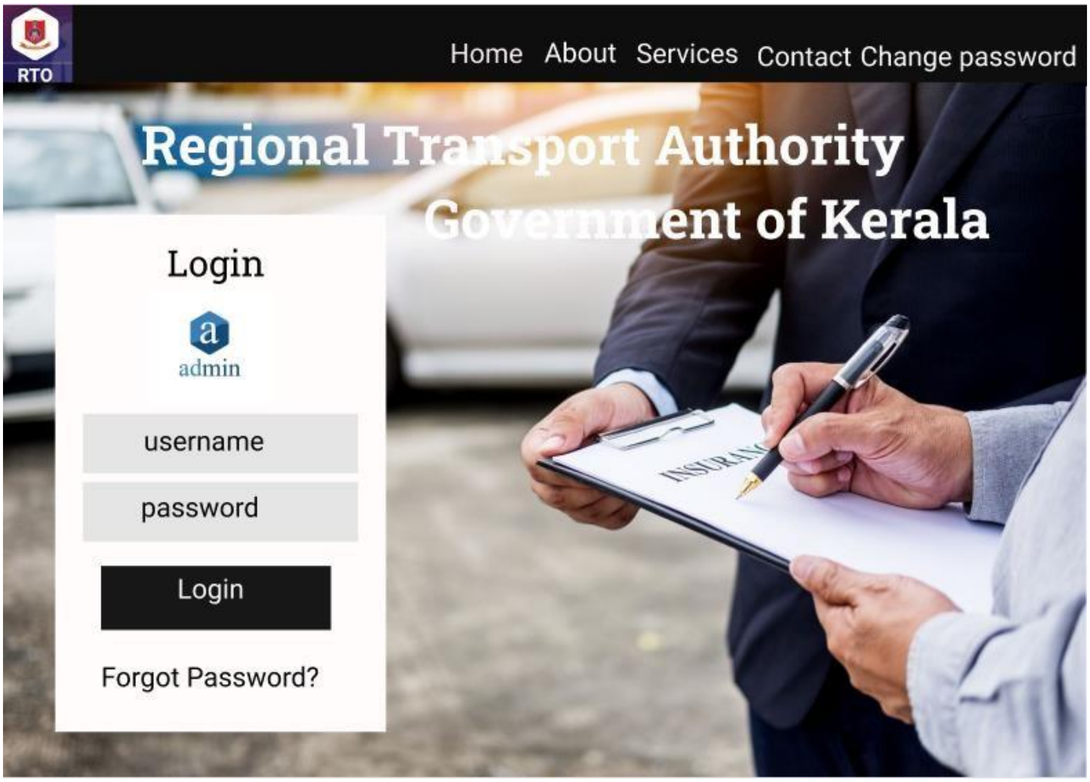
Login

username

password

Forgot Password?
Don't have an account? Sign up

Form Name : Admin Homepage



Login

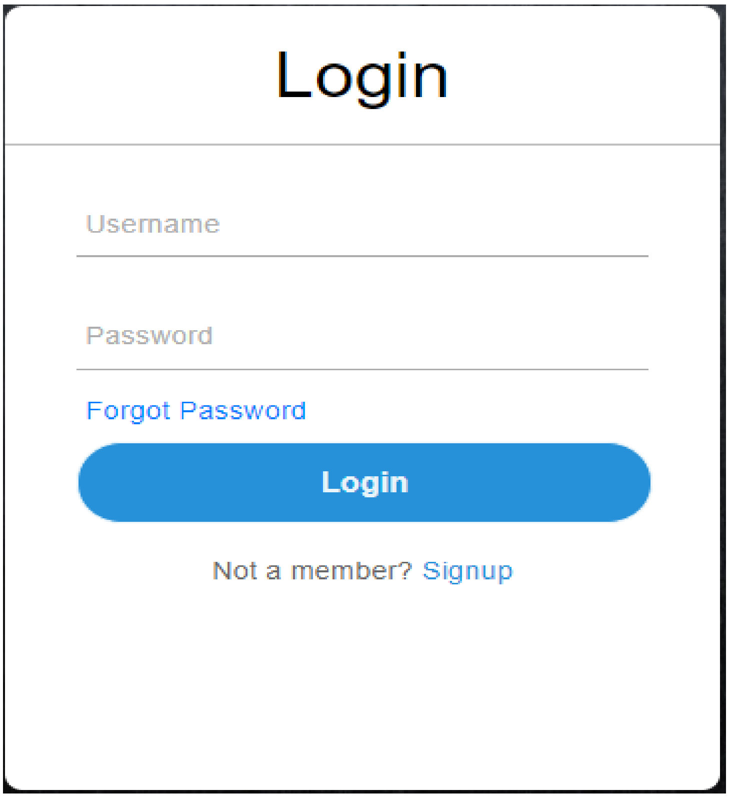
username

password

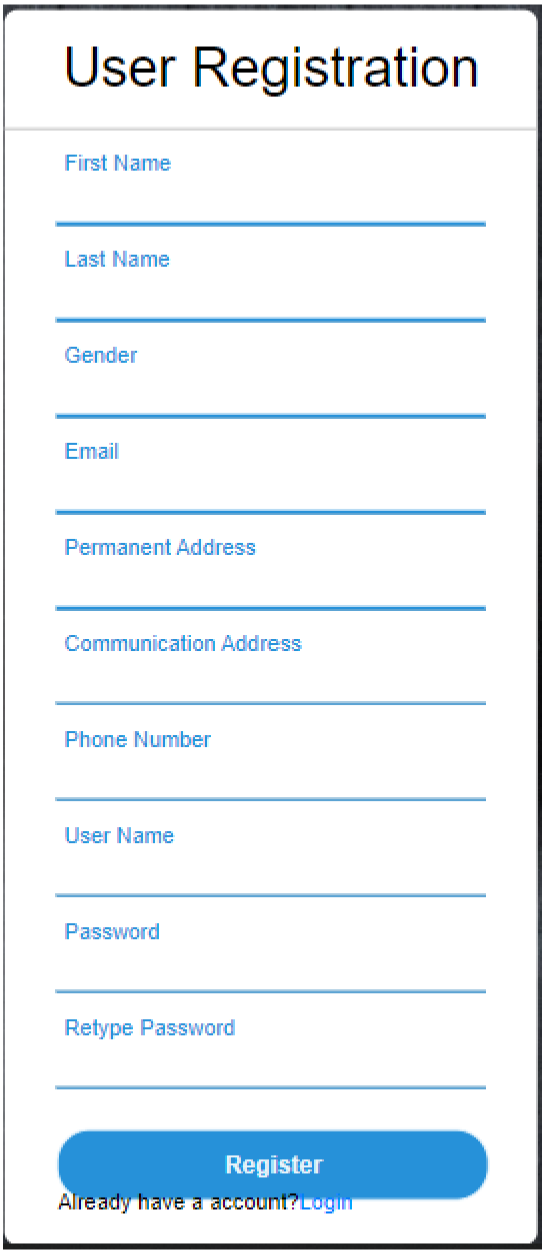
Forgot Password?

4.3.2 OUTPUT DESIGN

Form Name : User Login



Form Name : User Registration



User Registration

First Name

Last Name

Gender

Email

Permanent Address

Communication Address

Phone Number

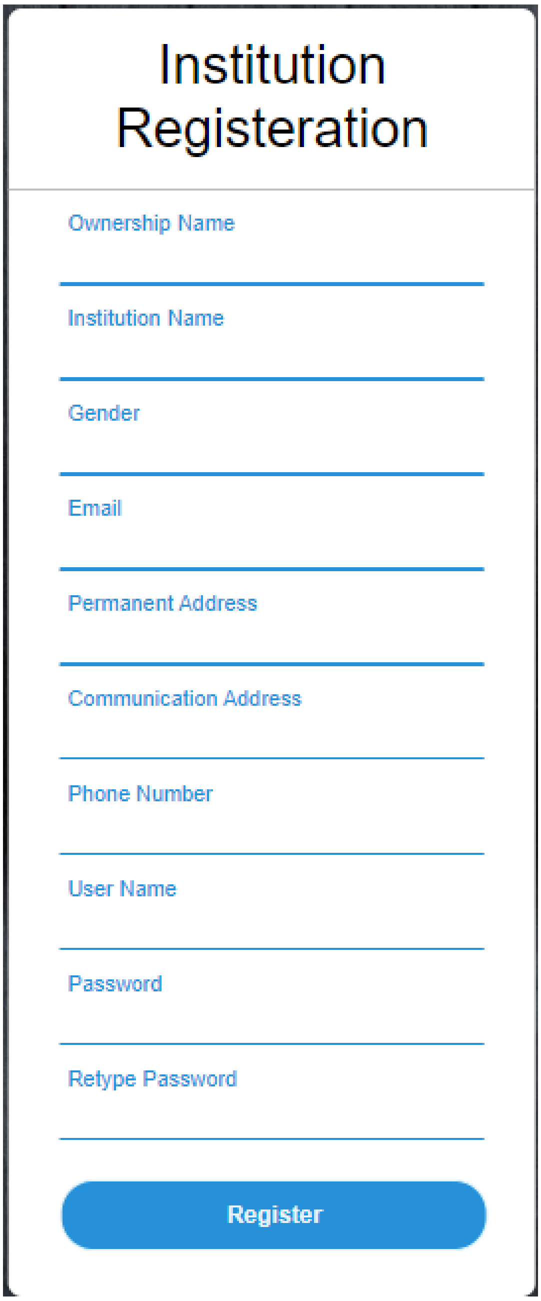
User Name

Password

Retype Password

Already have a account?Login

Form Name: Institution Registration



Institution Registeration

Ownership Name

Institution Name

Gender

Email

Permanent Address

Communication Address

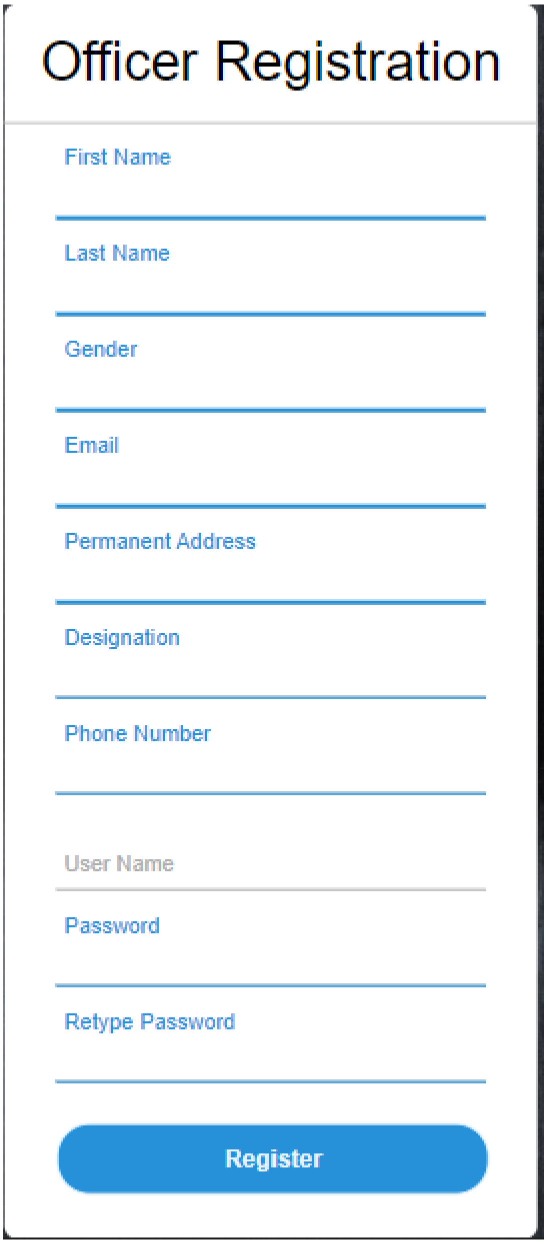
Phone Number

User Name

Password

Retype Password

Form Name: Subofficer Registration



Officer Registration

First Name

Last Name

Gender

Email

Permanent Address

Designation

Phone Number

User Name

Password

Retype Password

4.4 DATABASE DESIGN

A database is a structured system with the capacity to store information andallows users to retrieve stored information quickly and effectively. Any database's primary goal is its data, which demands protection.

There are two stages to the database design process. The user needs are obtained in the first step, and a database is created to as clearly as possible meet these criteria. This process, known as information level design, is carried out independently of all DBMSs.

The design for the specific DBMS that will be used to construct the system in issue is converted from an information level design to a design in the second stage. Physical Level Design is the stage where the characteristics of the particular DBMS that will be used are discussed. Parallel to the system design is a database design. The database's data arrangement aims to accomplish the two main goals listed below.

4.4.1 Data Integrity

4.4.2 Data independence

4.5.1 Relational Database Management System (RDBMS)

The database is displayed as a set of relations in a relational paradigm. A table or file of records with values can be compared to each relation. In formal relational model terminology, a row is referred to as a tuple, a column heading is referred to as an attribute, and the table is referred to as a relation. Numerous tables, each with a unique name, make up a relational database. Each row of a tale reflects a set of related values.

Relations, Domains & Attributes

A relation is a table. Tuples are the units of a table's rows. An ordered group of n elements is a tuple. Attributes are referred to as columns. The database already has links between all of the tables. This guarantees both the validity of entity relationships and referential ones. A domain D is a set of atomic values. A common method of defining a domain is by selecting a data type from which the domain's data values are derived. It is also useful to give the domain a name in order tomake the values of the domain simpler to comprehend. Each value in a relation is atomic and cannot be broken down.

Relationships

- Key is used to create table relationships. Primary Key and Foreign Key are the two principal keys that are most crucial. With the use of these keys, relationships for entity integrity and referential integrity can be created.
- Entity Integrity forbids the use of null values for any Primary Key.
- Referential Integrity enforces that no Primary Key can have null values.
- Referential Integrity: A Primary Key value in the same domain must correspond to each unique Foreign Key value. Super Key and Candidate Keys are additional keys.

4.5.2 Normalization

The simplest possible grouping of data is used to put them together so that future changes can be made with little influence on the data structures. The formal process of normalizing data structures in a way that reduces duplication and fosters integrity. Using the normalization technique, superfluous fields are removed and a huge table is divided into several smaller ones. Anomalies in insertion, deletion, and updating are also prevented by using it. Keys and relationships are two notions used in the standard form of data modelling. A row in a table is uniquely identified by a key. Primary keys and foreign keys are the two different kinds of keys. A primary key is an element, or set of components, in a table that serves as a means of distinguishing between records from the same table. A column in a table known as a foreign key is used to uniquely identify records from other tables. Up to the third normal form, all tables have been normalized.

As the name implies, it refers to arranging things in their natural state. The application developer uses normalization to provide a logical grouping of the data into suitable tables and columns so that users can immediately connect names to the data.By removing recurring groups from the data, normalization prevents data redundancy, which puts a heavy strain on the computer's resources. These include:

4.5.2.1 Normalize the data.

4.5.2.2 Choose proper names for the tables and columns.

4.5.2.3 Choose the proper name for the data.

First Normal Form

The First Normal Form requires that each attribute's domain only include atomic values and that each attribute's value in a tuple be a single value from that domain. In other words, 1NF forbids the use of relationships as attribute values within tuples or relations within relations. Under 1NF, only single atomic or indivisible values are allowed for attribute values. First Normal Form must be used to enter the data. Data can be divided into tables of a similar type in each table to achieve this. Repeated data groups were eliminated as a result. If only the constraints involving the primary key are satisfied, a relation is said to be in first normal form.

Second Normal Form

No non-key attribute in a relation where the primary key consists of multiple attributes should be functionally dependent on a major key component. This includes breaking down each partial key into its dependent attributes and generating a new relation for each one. Keep the original primary key and any attributes that are completely dependent on it in your database. With the use of this method, data that only depends on a small portion of the key can be eliminated. A relation is said to be in second normal form if and only if the primary key satisfies all the criteria for first normal form and all of the relation's non-primary key properties are wholly dependent on the primary key alone.

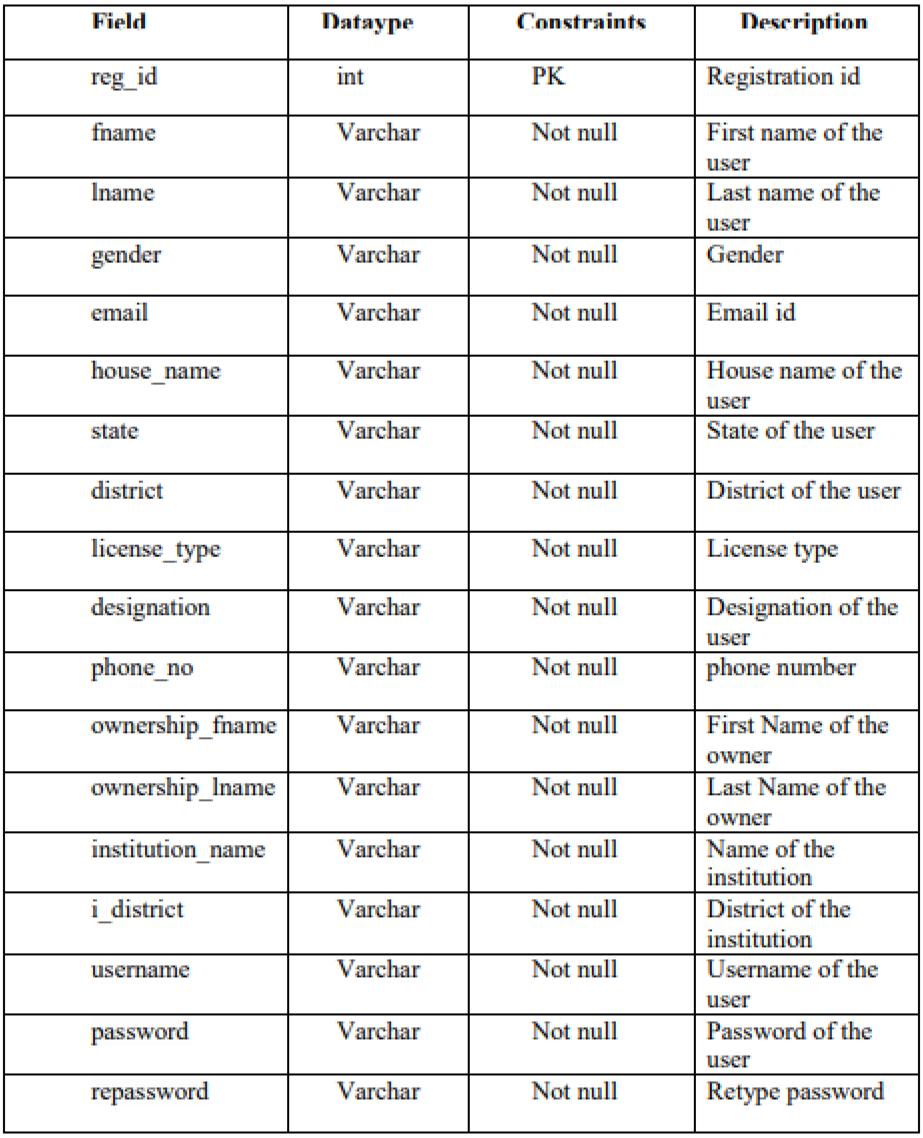
Third Normal Form

According to the Third Normal Form, a relation shouldn't have a non-key attribute that is functionally determined by another non-key attribute or by a group of non-key attributes. In other words, the primary key shouldn't be transitively dependent. The deconstructed non-key attributes are then arranged in relation to the non-key qualities that they functionally determine. Anything not fully dependent on the primary key is removed using this process. Only when a relation is in second normal form and, more importantly, when its non-key qualities do not depend on those of other non-key properties, is it said to be in third normal form.

TABLE DESIGN

Table No: 01

Name: tbl_register
Primary Key: reg_id
Foreign Key:
Description: To store registration information of user, sub-officer and Institution



Field	Dataype	Constraints	Description
reg_id	int	PK	Registration id
fname	Varchar	Not null	First name of the user
lname	Varchar	Not null	Last name of the user
gender	Varchar	Not null	Gender
email	Varchar	Not null	Email id
house_name	Varchar	Not null	House name of the user
state	Varchar	Not null	State of the user
district	Varchar	Not null	District of the user
license_type	Varchar	Not null	License type
designation	Varchar	Not null	Designation of the user
phone_no	Varchar	Not null	phone number
ownership_fname	Varchar	Not null	First Name of the owner
ownership_lname	Varchar	Not null	Last Name of the owner
institution_name	Varchar	Not null	Name of the institution
i_district	Varchar	Not null	District of the institution
username	Varchar	Not null	Username of the user
password	Varchar	Not null	Password of the user
repassword	Varchar	Not null	Retype password

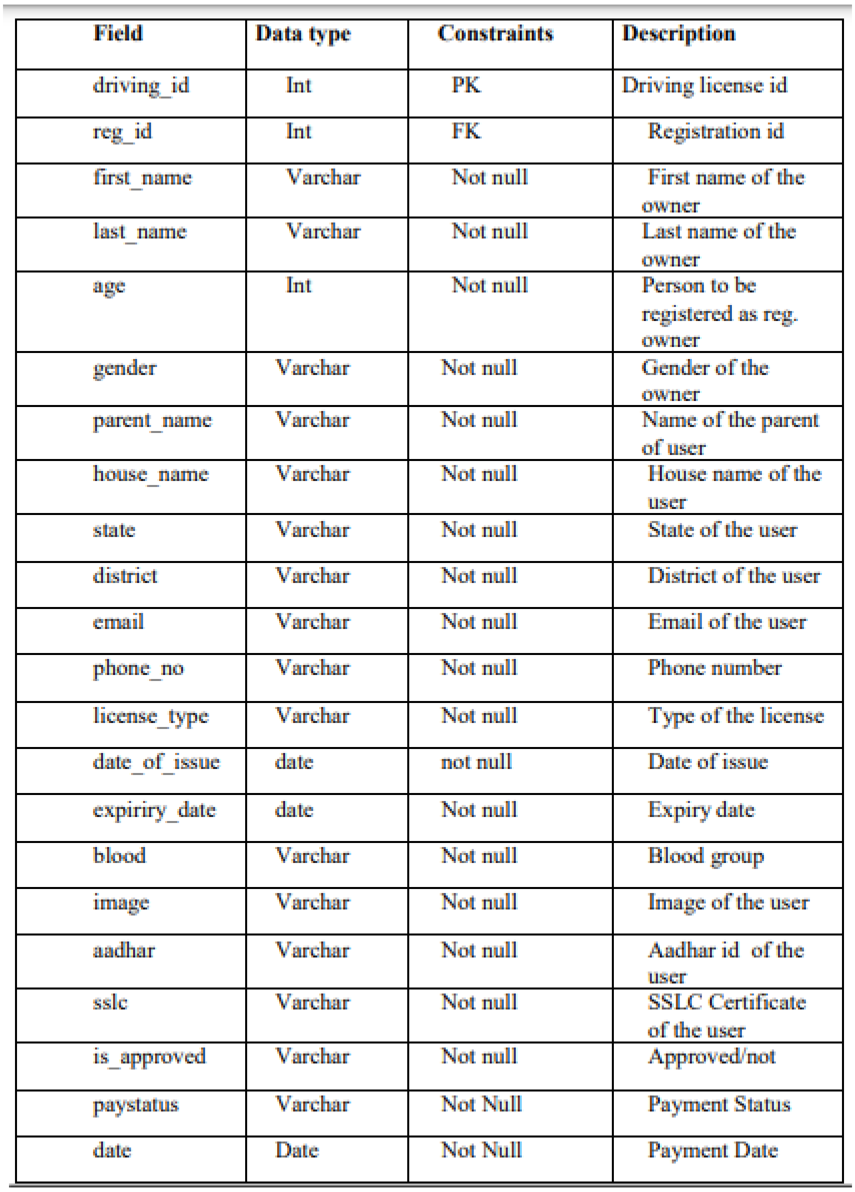
Table No: 02

Table Name: tbl_drivinglicense

Primary Key: driving_id Foreign

Key: reg_id

Table Description: keeps information about the license owner



Field	Data type	Constraints	Description
driving_id	Int	PK	Driving license id
reg_id	Int	FK	Registration id
first_name	Varchar	Not null	First name of the owner
last_name	Varchar	Not null	Last name of the owner
age	Int	Not null	Person to be registered as reg. owner
gender	Varchar	Not null	Gender of the owner
parent_name	Varchar	Not null	Name of the parent of user
house_name	Varchar	Not null	House name of the user
state	Varchar	Not null	State of the user
district	Varchar	Not null	District of the user
email	Varchar	Not null	Email of the user
phone_no	Varchar	Not null	Phone number
license_type	Varchar	Not null	Type of the license
date_of_issue	date	not null	Date of issue
expiriry_date	date	Not null	Expiry date
blood	Varchar	Not null	Blood group
image	Varchar	Not null	Image of the user
aadhar	Varchar	Not null	Aadhar id of the user
sslc	Varchar	Not null	SSLC Certificate of the user
is_approved	Varchar	Not null	Approved/not
paystatus	Varchar	Not Null	Payment Status
date	Date	Not Null	Payment Date

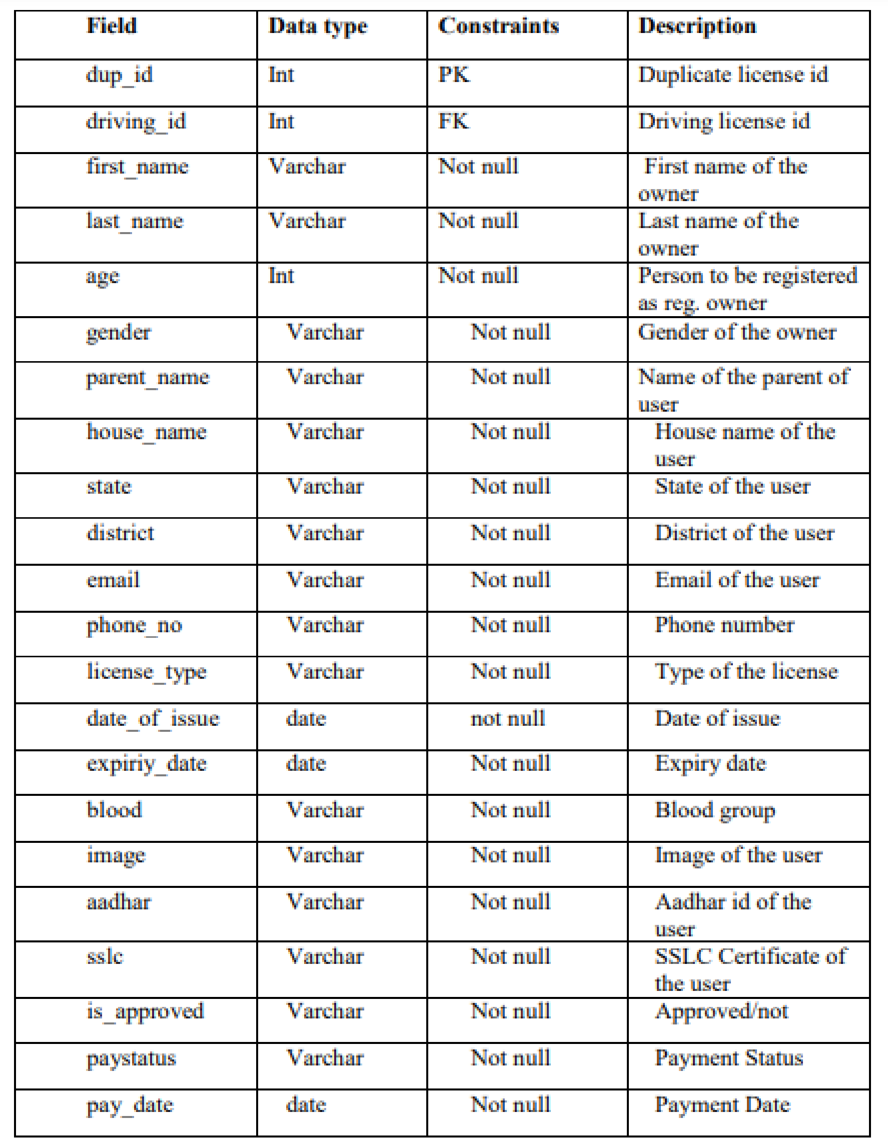
Table No: 03

Table Name: tbl_duplicatelicense

Primary Key: dup_id

Foreign Key: driving_id

Table Description: keeps information about duplication of license



Field	Data type	Constraints	Description
dup_id	Int	PK	Duplicate license id
driving_id	Int	FK	Driving license id
first_name	Varchar	Not null	First name of the owner
last_name	Varchar	Not null	Last name of the owner
age	Int	Not null	Person to be registered as reg. owner
gender	Varchar	Not null	Gender of the owner
parent_name	Varchar	Not null	Name of the parent of user
house_name	Varchar	Not null	House name of the user
state	Varchar	Not null	State of the user
district	Varchar	Not null	District of the user
email	Varchar	Not null	Email of the user
phone_no	Varchar	Not null	Phone number
license_type	Varchar	Not null	Type of the license
date_of_issue	date	not null	Date of issue
expiriy_date	date	Not null	Expiry date
blood	Varchar	Not null	Blood group
image	Varchar	Not null	Image of the user
aadhar	Varchar	Not null	Aadhar id of the user
sslc	Varchar	Not null	SSLC Certificate of the user
is_approved	Varchar	Not null	Approved/not
paystatus	Varchar	Not null	Payment Status
pay_date	date	Not null	Payment Date

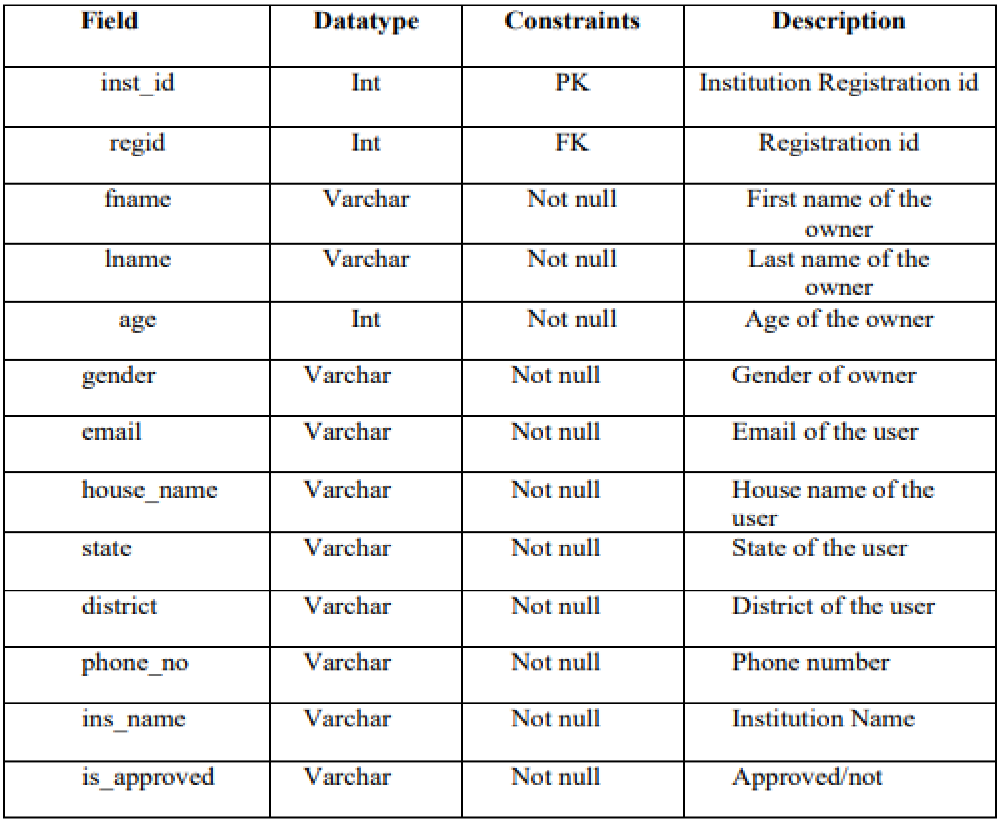
Table No: 04

Name: tbl_inst_reg

Primary Key: inst_id

Foreign Key: regid

Description: Keeps information about the Institution Registration



Field	Datatype	Constraints	Description
inst_id	Int	PK	Institution Registration id
regid	Int	FK	Registration id
fname	Varchar	Not null	First name of the owner
lname	Varchar	Not null	Last name of the owner
age	Int	Not null	Age of the owner
gender	Varchar	Not null	Gender of owner
email	Varchar	Not null	Email of the user
house_name	Varchar	Not null	House name of the user
state	Varchar	Not null	State of the user
district	Varchar	Not null	District of the user
phone_no	Varchar	Not null	Phone number
ins_name	Varchar	Not null	Institution Name
is_approved	Varchar	Not null	Approved/not

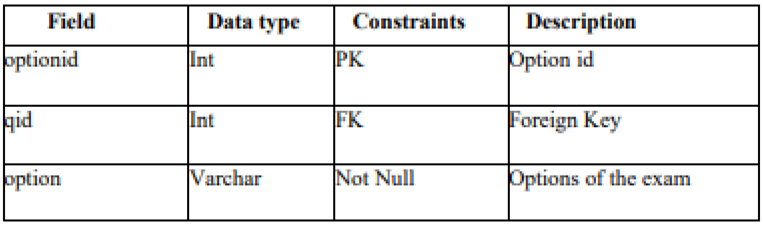
Table No: 05

Name: option

Primary Key: optionid

Foreign Key: qid

Description: stores the options of the test



Field	Data type	Constraints	Description
optionid	Int	PK	Option id
qid	Int	FK	Foreign Key
option	Varchar	Not Null	Options of the exam

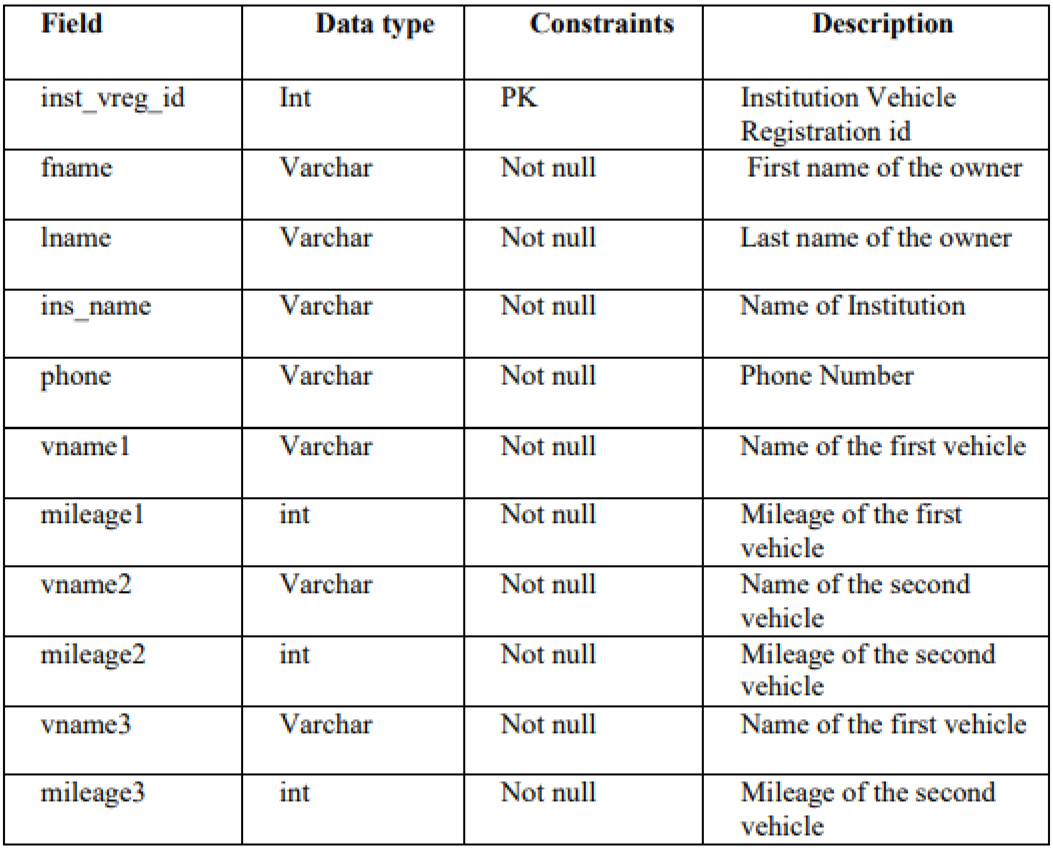
Table No: 06

Name: tbl_inst_vreg

Primary Key: inst_vreg_id

Foreign Key:

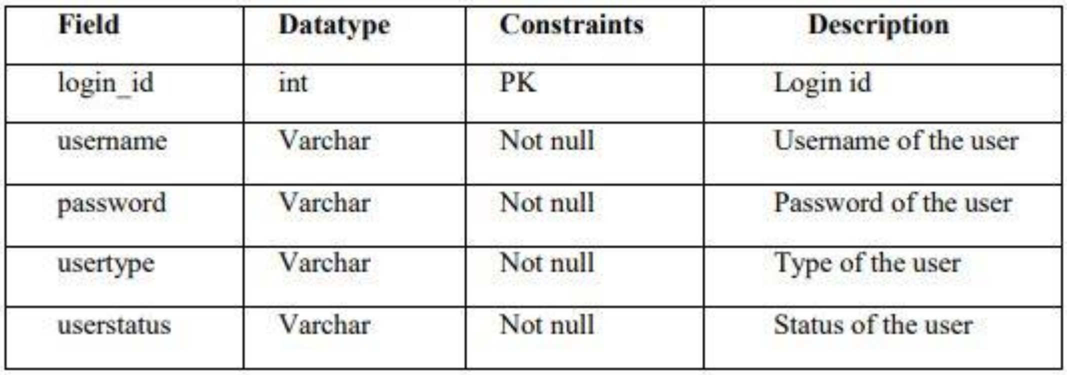
Description: keeps information about the Institution Vehicle Registration



Field	Data type	Constraints	Description
inst_vreg_id	Int	PK	Institution Vehicle Registration id
fname	Varchar	Not null	First name of the owner
lname	Varchar	Not null	Last name of the owner
ins_name	Varchar	Not null	Name of Institution
phone	Varchar	Not null	Phone Number
vname1	Varchar	Not null	Name of the first vehicle
mileage1	int	Not null	Mileage of the first vehicle
vname2	Varchar	Not null	Name of the second vehicle
mileage2	int	Not null	Mileage of the second vehicle
vname3	Varchar	Not null	Name of the first vehicle
mileage3	int	Not null	Mileage of the second vehicle

Table No: 07

Name: tbl_login Primary
Key: login_idForeign
Key:
Description: store the login details



Field	Datatype	Constraints	Description
login_id	int	PK	Login id
username	Varchar	Not null	Username of the user
password	Varchar	Not null	Password of the user
usertype	Varchar	Not null	Type of the user
userstatus	Varchar	Not null	Status of the user

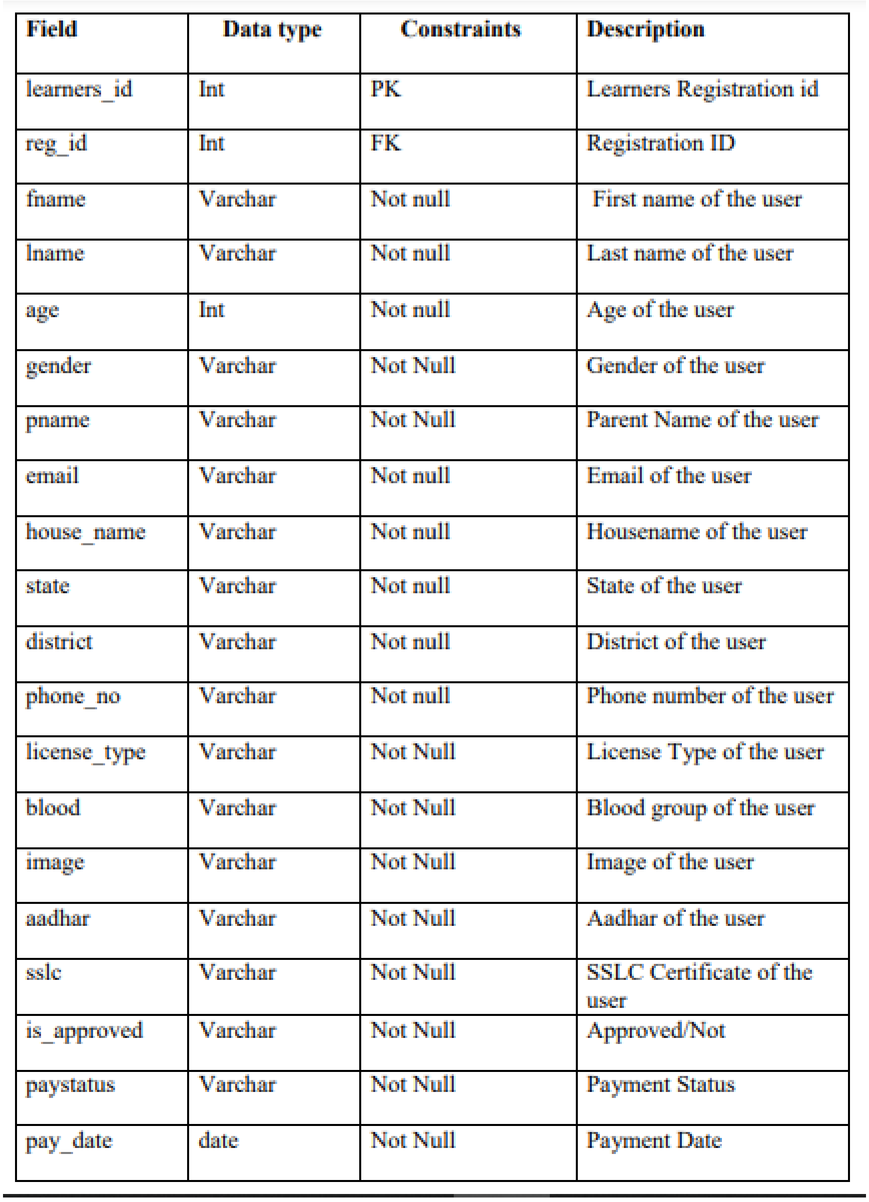
Table No: 08

Name: tbl_learners_reg

Primary Key: learners_id

Foreign Key: regid

Description: keeps information about the learners registration of the users



Field	Data type	Constraints	Description
learners_id	Int	PK	Learners Registration id
reg_id	Int	FK	Registration ID
fname	Varchar	Not null	First name of the user
lname	Varchar	Not null	Last name of the user
age	Int	Not null	Age of the user
gender	Varchar	Not Null	Gender of the user
pname	Varchar	Not Null	Parent Name of the user
email	Varchar	Not null	Email of the user
house_name	Varchar	Not null	Housename of the user
state	Varchar	Not null	State of the user
district	Varchar	Not null	District of the user
phone_no	Varchar	Not null	Phone number of the user
license_type	Varchar	Not Null	License Type of the user
blood	Varchar	Not Null	Blood group of the user
image	Varchar	Not Null	Image of the user
aadhar	Varchar	Not Null	Aadhar of the user
sslc	Varchar	Not Null	SSLC Certificate of the user
is_approved	Varchar	Not Null	Approved/Not
paystatus	Varchar	Not Null	Payment Status
pay_date	date	Not Null	Payment Date

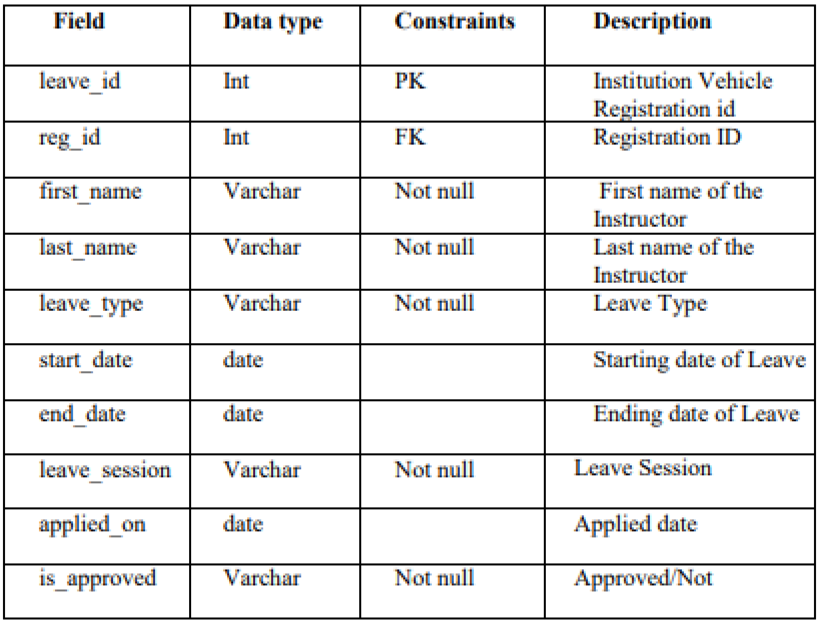
Table No: 09

Name: tbl_leave

Primary Key: leave_id

Foreign Key: reg_id

Description: keeps information about the Leave details of the instructor



Field	Data type	Constraints	Description
leave_id	Int	PK	Institution Vehicle Registration id
reg_id	Int	FK	Registration ID
first_name	Varchar	Not null	First name of the Instructor
last_name	Varchar	Not null	Last name of the Instructor
leave_type	Varchar	Not null	Leave Type
start_date	date		Starting date of Leave
end_date	date		Ending date of Leave
leave_session	Varchar	Not null	Leave Session
applied_on	date		Applied date
is_approved	Varchar	Not null	Approved/Not

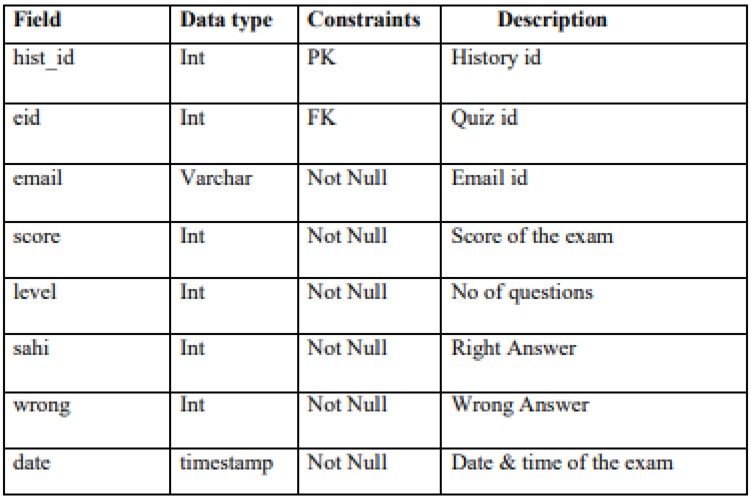
Table No: 10

Name: history

Primary Key: hist_id

Foreign Key: eid

Description: stores the history of the test



Field	Data type	Constraints	Description
hist_id	Int	PK	History id
eid	Int	FK	Quiz id
email	Varchar	Not Null	Email id
score	Int	Not Null	Score of the exam
level	Int	Not Null	No of questions
sahi	Int	Not Null	Right Answer
wrong	Int	Not Null	Wrong Answer
date	timestamp	Not Null	Date & time of the exam

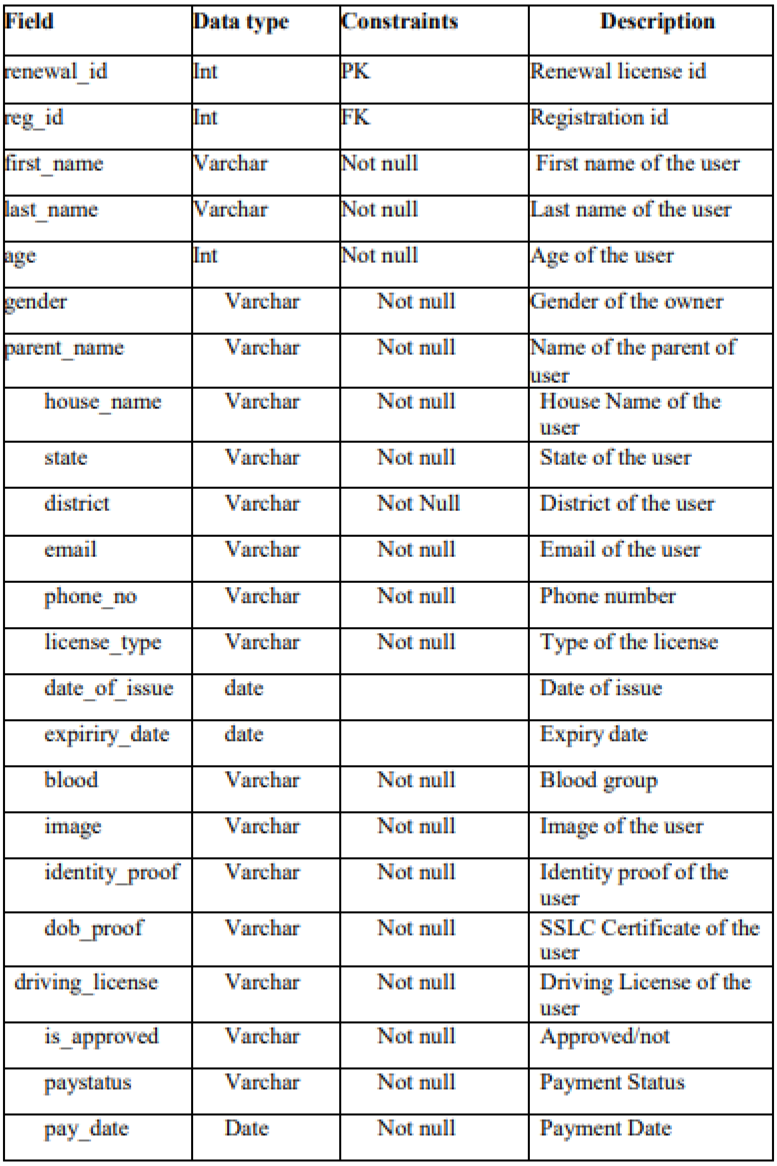
Table No: 11

Table Name: tbl_renewal

Primary Key: renewal_id

Foreign Key: reg_id

Table Description: keeps information about renewal of license



Field	Data type	Constraints	Description
renewal_id	Int	PK	Renewal license id
reg_id	Int	FK	Registration id
first_name	Varchar	Not null	First name of the user
last_name	Varchar	Not null	Last name of the user
age	Int	Not null	Age of the user
gender	Varchar	Not null	Gender of the owner
parent_name	Varchar	Not null	Name of the parent of user
house_name	Varchar	Not null	House Name of the user
state	Varchar	Not null	State of the user
district	Varchar	Not Null	District of the user
email	Varchar	Not null	Email of the user
phone_no	Varchar	Not null	Phone number
license_type	Varchar	Not null	Type of the license
date_of_issue	date		Date of issue
expiriry_date	date		Expiry date
blood	Varchar	Not null	Blood group
image	Varchar	Not null	Image of the user
identity_proof	Varchar	Not null	Identity proof of the user
dob_proof	Varchar	Not null	SSLC Certificate of the user
driving_license	Varchar	Not null	Driving License of the user
is_approved	Varchar	Not null	Approved/not
paystatus	Varchar	Not null	Payment Status
pay_date	Date	Not null	Payment Date

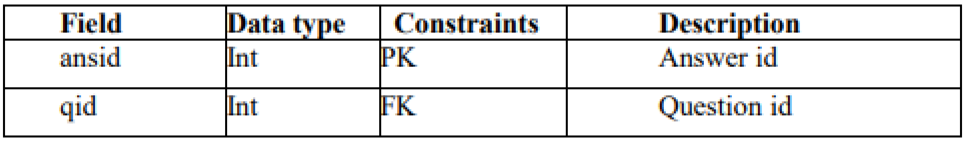
Table No: 12

Name: answer

Primary Key: ansid

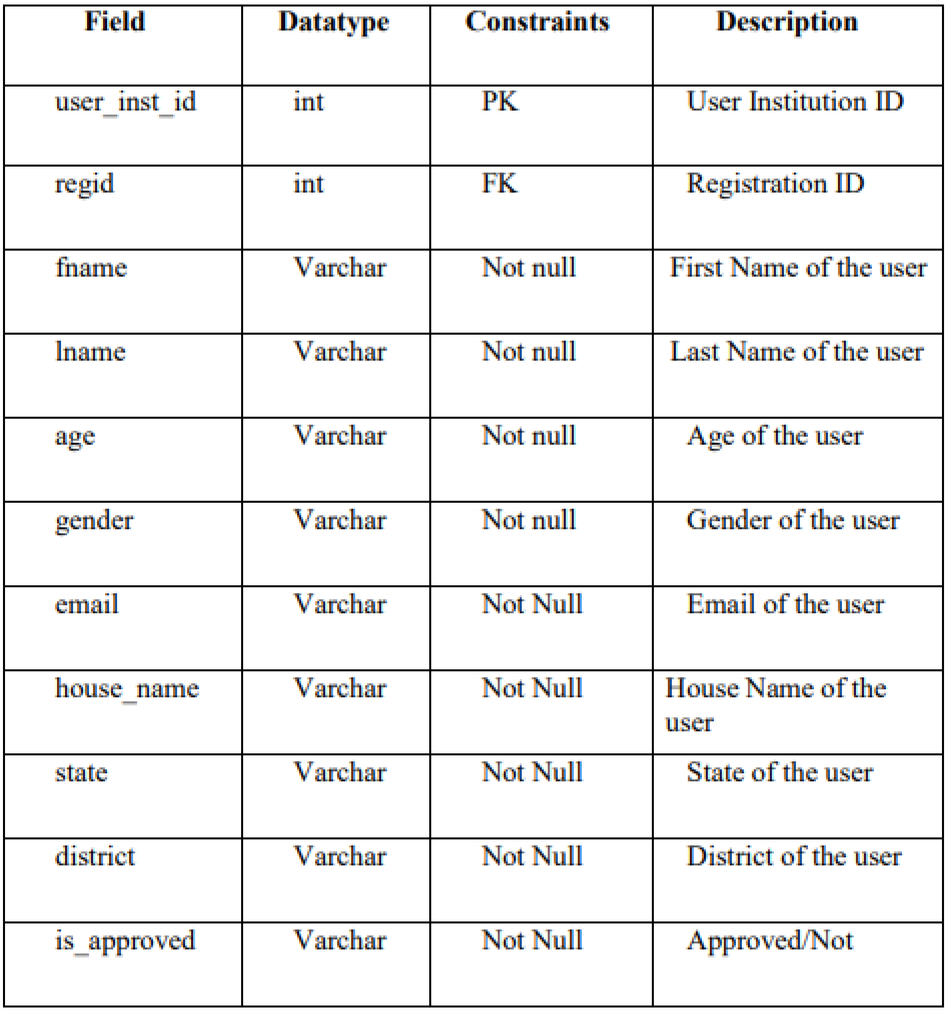
Foreign Key: qid

Description: stores the answers of the test



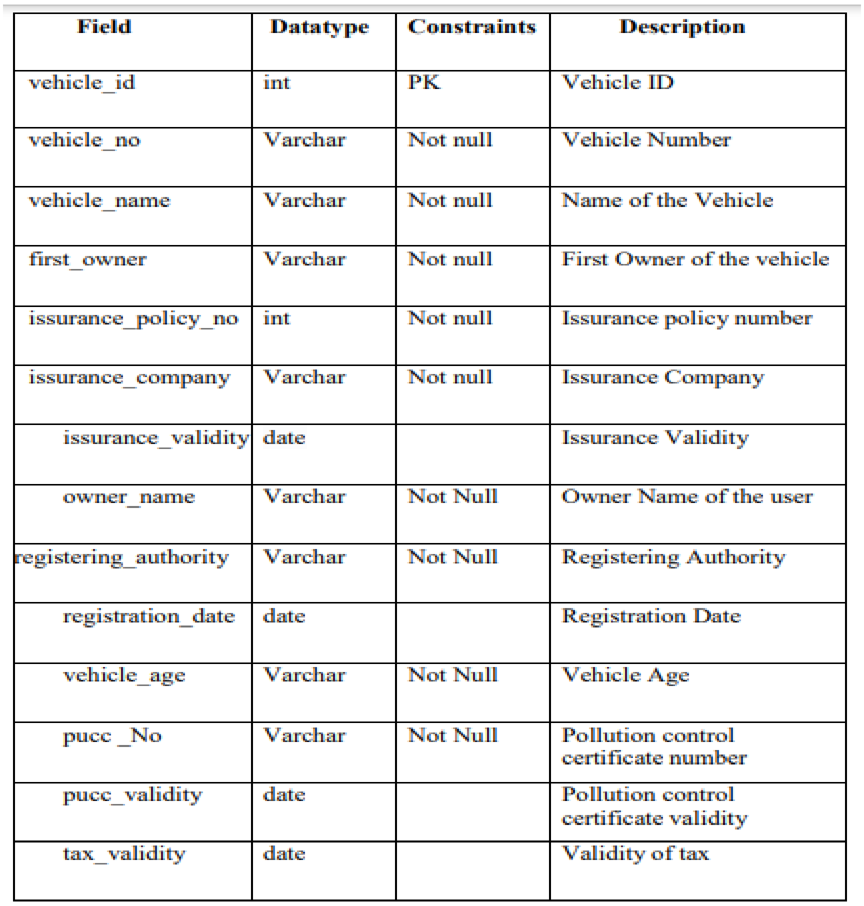
Field	Data type	Constraints	Description
ansid	Int	PK	Answer id
qid	Int	FK	Question id

Table No: 13
Name: tbl_user_inst_reg
Primary Key: user_inst_reg
Foreign Key: regid
Description: Store the Institution registration of the user



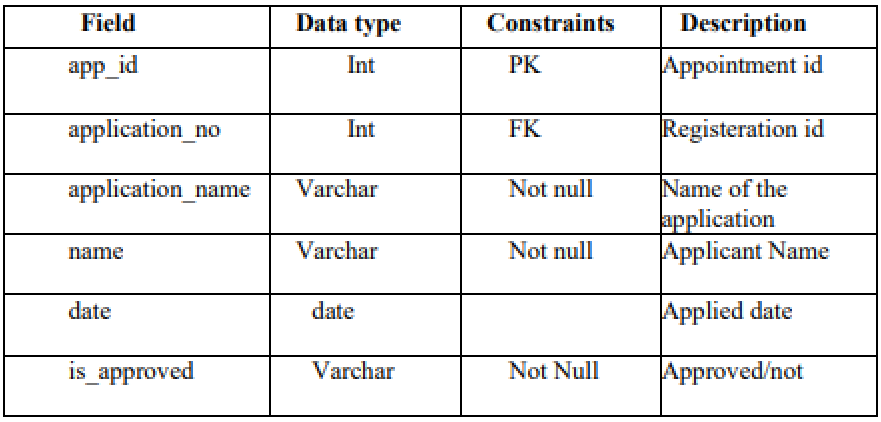
Field	Datatype	Constraints	Description
user_inst_id	int	PK	User Institution ID
regid	int	FK	Registration ID
fname	Varchar	Not null	First Name of the user
lname	Varchar	Not null	Last Name of the user
age	Varchar	Not null	Age of the user
gender	Varchar	Not null	Gender of the user
email	Varchar	Not Null	Email of the user
house_name	Varchar	Not Null	House Name of the user
state	Varchar	Not Null	State of the user
district	Varchar	Not Null	District of the user
is_approved	Varchar	Not Null	Approved/Not

Table No: 14
Name: tbl_vehicle_details
Primary Key: vehicle_id
Foreign Key:
Description: Store the details of the user



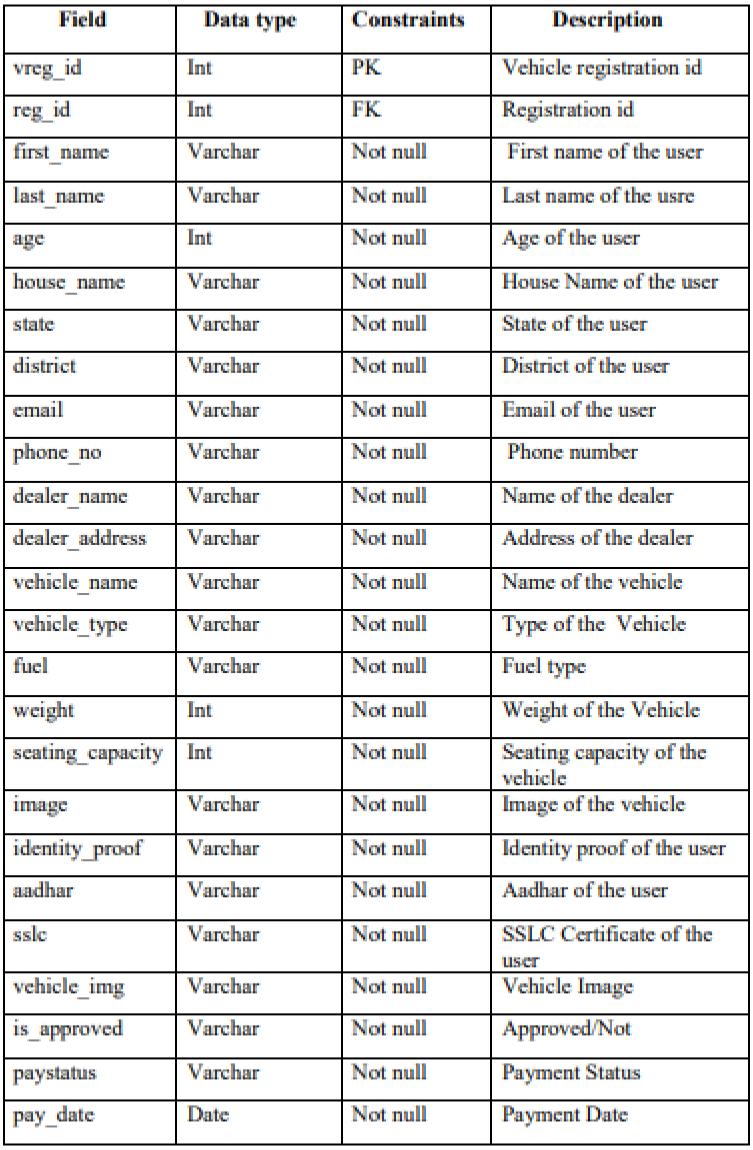
Field	Datatype	Constraints	Description
vehicle_id	int	PK	Vehicle ID
vehicle_no	Varchar	Not null	Vehicle Number
vehicle_name	Varchar	Not null	Name of the Vehicle
first_owner	Varchar	Not null	First Owner of the vehicle
issurance_policy_no	int	Not null	Issurance policy number
issurance_company	Varchar	Not null	Issurance Company
issurance_validity	date		Issurance Validity
owner_name	Varchar	Not Null	Owner Name of the user
registering_authority	Varchar	Not Null	Registering Authority
registration_date	date		Registration Date
vehicle_age	Varchar	Not Null	Vehicle Age
pucc _No	Varchar	Not Null	Pollution control certificate number
pucc_validity	date		Pollution control certificate validity
tax_validity	date		Validity of tax

Table No: 15
Name: tbl_appointment
Primary Key: app_id
Foreign Key: application_no
Description: stores the Appointment details



Field	Data type	Constraints	Description
app_id	Int	PK	Appointment id
application_no	Int	FK	Registeration id
application_name	Varchar	Not null	Name of the application
name	Varchar	Not null	Applicant Name
date	date		Applied date
is_approved	Varchar	Not Null	Approved/not

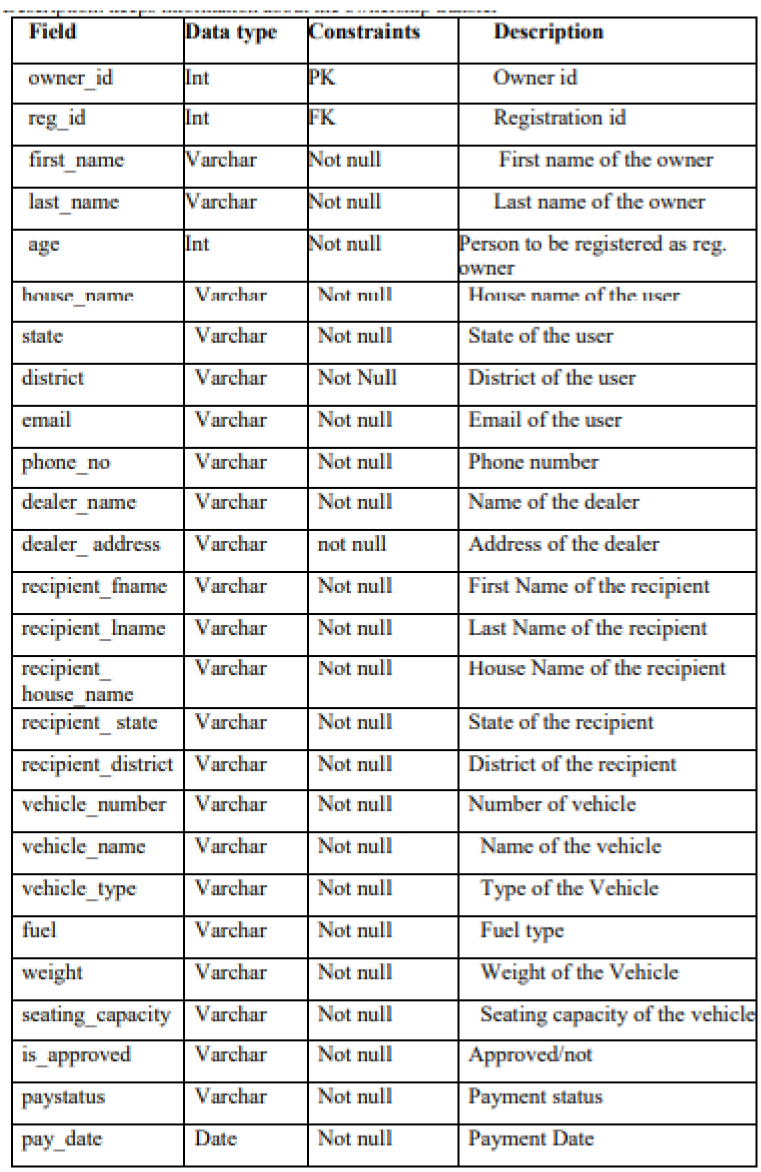
Table No: 16
Name: tbl_vreg
Primary Key: vreg_id
Foreign Key: reg_id
Description: keeps information about the registration of the vehicle



Field	Data type	Constraints	Description
vreg_id	Int	PK	Vehicle registration id
reg_id	Int	FK	Registration id
first_name	Varchar	Not null	First name of the user
last_name	Varchar	Not null	Last name of the usre
age	Int	Not null	Age of the user
house_name	Varchar	Not null	House Name of the user
state	Varchar	Not null	State of the user
district	Varchar	Not null	District of the user
email	Varchar	Not null	Email of the user
phone_no	Varchar	Not null	Phone number
dealer_name	Varchar	Not null	Name of the dealer
dealer_address	Varchar	Not null	Address of the dealer
vehicle_name	Varchar	Not null	Name of the vehicle
vehicle_type	Varchar	Not null	Type of the Vehicle
fuel	Varchar	Not null	Fuel type
weight	Int	Not null	Weight of the Vehicle
seating_capacity	Int	Not null	Seating capacity of the vehicle
image	Varchar	Not null	Image of the vehicle
identity_proof	Varchar	Not null	Identity proof of the user
aadhar	Varchar	Not null	Aadhar of the user
sslc	Varchar	Not null	SSLC Certificate of the user
vehicle_img	Varchar	Not null	Vehicle Image
is_approved	Varchar	Not null	Approved/Not
paystatus	Varchar	Not null	Payment Status
pay_date	Date	Not null	Payment Date

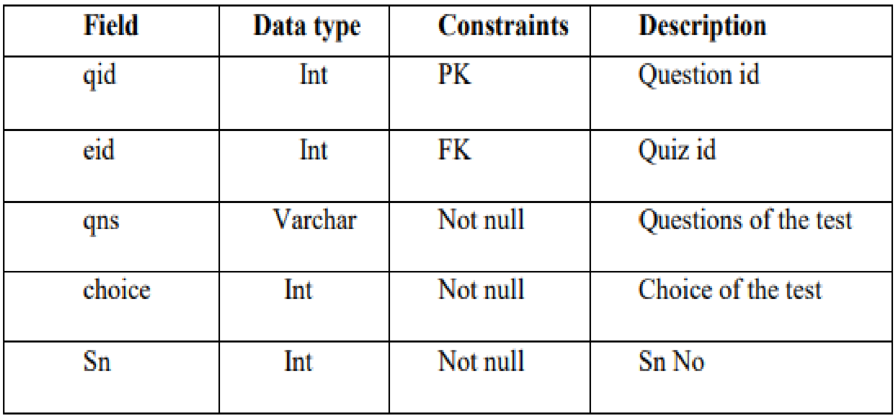
Table No: 17

Name: tbl_ownership
Primary Key: owner_id
Foreign Key:reg_id
Description: keeps information about the ownership transfer



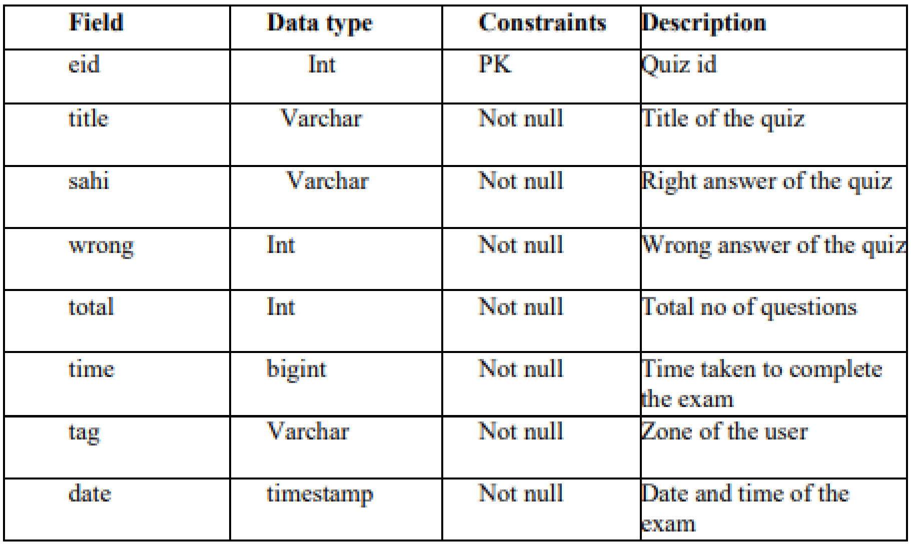
Field	Data type	Constraints	Description
owner_id	Int	PK	Owner id
reg_id	Int	FK	Registration id
first_name	Varchar	Not null	First name of the owner
last_name	Varchar	Not null	Last name of the owner
age	Int	Not null	Person to be registered as reg. owner
house_name	Varchar	Not null	House name of the user
state	Varchar	Not null	State of the user
district	Varchar	Not Null	District of the user
email	Varchar	Not null	Email of the user
phone_no	Varchar	Not null	Phone number
dealer_name	Varchar	Not null	Name of the dealer
dealer_ address	Varchar	not null	Address of the dealer
recipient_fname	Varchar	Not null	First Name of the recipient
recipient_lname	Varchar	Not null	Last Name of the recipient
recipient_ house_name	Varchar	Not null	House Name of the recipient
recipient_ state	Varchar	Not null	State of the recipient
recipient_district	Varchar	Not null	District of the recipient
vehicle_number	Varchar	Not null	Number of vehicle
vehicle_name	Varchar	Not null	Name of the vehicle
vehicle_type	Varchar	Not null	Type of the Vehicle
fuel	Varchar	Not null	Fuel type
weight	Varchar	Not null	Weight of the Vehicle
seating_capacity	Varchar	Not null	Seating capacity of the vehicle
is_approved	Varchar	Not null	Approved/not
paystatus	Varchar	Not null	Payment status
pay_date	Date	Not null	Payment Date

Table No: 18
Name: questions
Primary Key: qid
Foreign Key: eid
Description: stores the questions of the test



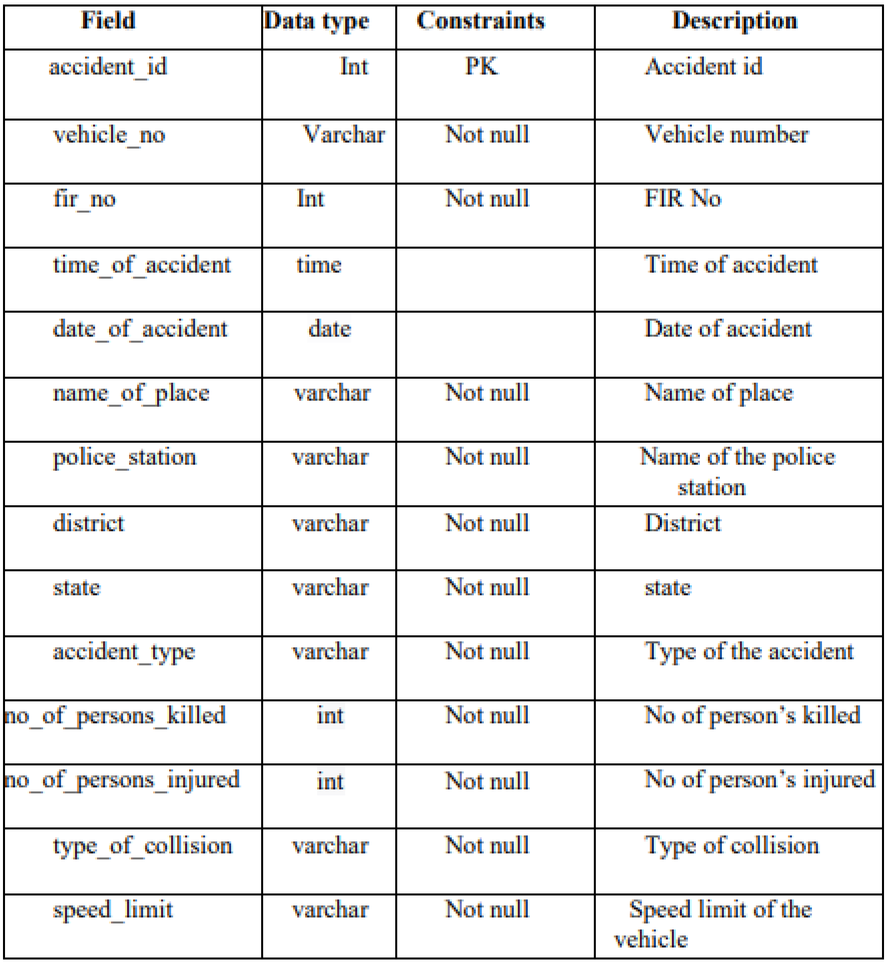
Field	Data type	Constraints	Description
qid	Int	PK	Question id
eid	Int	FK	Quiz id
qns	Varchar	Not null	Questions of the test
choice	Int	Not null	Choice of the test
Sn	Int	Not null	Sn No

Table No: 19
Name: quiz
Primary Key: eid
Foreign Key:
Description: stores the questions of the test



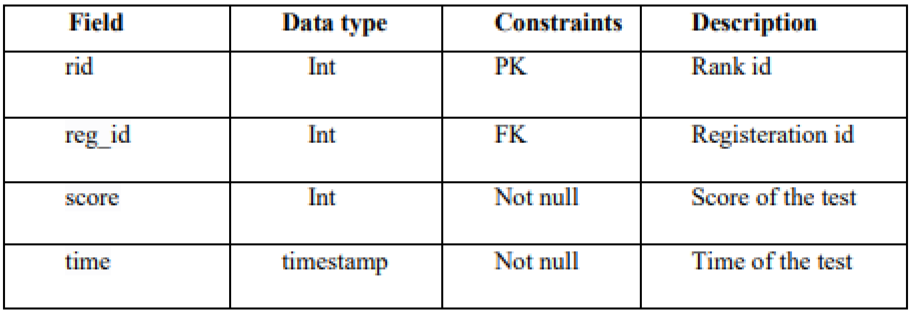
Field	Data type	Constraints	Description
eid	Int	PK	Quiz id
title	Varchar	Not null	Title of the quiz
sahi	Varchar	Not null	Right answer of the quiz
wrong	Int	Not null	Wrong answer of the quiz
total	Int	Not null	Total no of questions
time	bigint	Not null	Time taken to complete the exam
tag	Varchar	Not null	Zone of the user
date	timestamp	Not null	Date and time of the exam

Table No: 20
Name: tbl_accident_details
Primary Key: accident_id
Foreign Key:
Description: stores the accident details of the vehicle



Field	Data type	Constraints	Description
accident_id	Int	PK	Accident id
vehicle_no	Varchar	Not null	Vehicle number
fir_no	Int	Not null	FIR No
time_of_accident	time		Time of accident
date_of_accident	date		Date of accident
name_of_place	varchar	Not null	Name of place
police_station	varchar	Not null	Name of the police station
district	varchar	Not null	District
state	varchar	Not null	state
accident_type	varchar	Not null	Type of the accident
no_of_persons_killed	int	Not null	No of person's killed
no_of_persons_injured	int	Not null	No of person's injured
type_of_collision	varchar	Not null	Type of collision
speed_limit	varchar	Not null	Speed limit of the vehicle

Table No: 21
Name: rank
Primary Key: rid
Foreign Key: reg_id
Description: stores the ranking of the test



Field	Data type	Constraints	Description
rid	Int	PK	Rank id
reg_id	Int	FK	Registeration id
score	Int	Not null	Score of the test
time	timestamp	Not null	Time of the test

CHAPTER 5

SYSTEM TESTING

5.1 INTRODUCTION

Software testing is the process of carefully controlling the execution of software in order to determine whether it behaves as intended. The words verification and validation are frequently used in conjunction with software testing. Validation is the process of examining or evaluating a product, including software, to determine whether it complies with all relevant specifications. One type of verification, software testing, uses methods including reviews, analyses, inspections, and walkthroughs as well. Verifying that what has been specified matches what the user truly desired is the process of validation.

The processes of static analysis and dynamic analysis are additional ones that are frequently related to software testing. Static analysis examines the software's source code, searching for issues and obtaining statistics without actually running the code. Dynamic analysis examines how software behaves while it is running in order to offer data like execution traces, timing profiles, and test coverage details.

Testing is a collection of activities that can be planned ahead of time and carried out in a methodical manner. Testing starts with individual modules and progresses to the integration of the full computer-based system. There are many rules that can be used as testing objectives, and testing is necessary for the system testing objectives to be successful. As follows:

A program is tested by being run with the goal of identifying any errors.

- A successful test is one that finds an undetected error.
- A good test case is one that has a high likelihood of doing so.

If a test is successfully carried out in accordance with the aforementioned aims, it will reveal software bugs. Additionally, testing shows that the software functions seem to operate in accordance with the specification and that the performance requirements seem to have been satisfied.

There are three methods for program testing.

- For correctness
- For implementation efficiency
- For computational complexity

A program must be tested for correctness to guarantee that it operates as intended. This is significantly more difficult than it first appears to be, especially for large programs.

5.2 TEST PLAN

A test plan suggests a number of required steps that need be taken in order tocomplete various testing methodologies. The activity that is to be taken is outlined in the test plan. A computer program, its documentation, and associated data structures are all created by software developers. It is always the responsibility of the software developers to test each of the program's separate components to make sure it fulfils the purpose for which it was intended. In order to solve the inherent issues with allowing the builder evaluate what they have developed, there is an independent test group (ITG). The testing's explicit goals should be expressed in numerical terms. The test plan should contain details on the mean time to failure, the expense of identifying and resolving problems, the remaining defect density or frequency of occurrence, and the number of test work hours needed for each regression test.

The levels of testing include:

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

5.2.1 Unit Testing

The smallest unit of software design, the software component or module, is the focus of unit testing, which focuses verification efforts on it. Testing critical control routesto identify flaws inside the module's boundary is done using the component level design description as a reference. the specified untested area for unit testing and the test complexity level. Unit testing focuses on the white box, and multiple components may be tested at once. The modular interface is tested to ensure that data enters and exits the software unit under test correctly. The local data structure is reviewed to ensure that data temporarily stored maintains its integrity during each step of an algorithm's execution. To confirm that each statement in a module has been executed at least once, boundary conditions are evaluated. Finally, each method of error management is looked at.

Testing of data flow through a module interface are important before beginning any additional tests. If data cannot correctly enter and exit the system, all other tests are useless. The unit test's selective analysis of execution routes is a crucial task. In order to cleanly reroute or stop work when an error does occur, error handling channels must be set up and error scenarios must be anticipated in excellent design. Boundary testing is the last stage of unit testing. At its boundaries, software frequently fails.

In the Sell-Soft System, unit testing was carried out by treating each module as a distinct entity and subjecting them to a variety of test inputs. The internal logic of the modules had some issues, which were fixed. After coding each module is tested and run individually. All unnecessary code where removed and ensured that all modules are working, and gives the expected result.

5.2.2 Integration Testing

Integration testing is a methodical approach for creating the program's structure while also carrying out tests to find interface issues. The goal is to construct a program structure that has been determined by design using unit tested components. Theprogram as a whole is tested. Correction is challenging since the size of the overall program makes it challenging to isolate the causes. As soon as these mistakes are fixed, new ones arise, and the process repeats itself in an apparently unending cycle. All of the modules were integrated after unit testing was completed in the system to check for any interface inconsistencies. A distinctive program structure also developed when discrepancies in program structures were eliminated.

5.2.3 Validation Testing or System Testing

This marks the conclusion of the testing procedure. This required comprehensive testing of the system, which covered all forms, codes, modules, and class modules. System tests and black box testing are two common names for this kind of testing.

The primary focus of the black box testing approach is on the software's functional requirements. In this case, a software engineer can design sets of input conditions utilizing Black Box testing that will thoroughly test each programme requirement.

Erroneous or missing functions, interface faults, data structure or external data access errors, performance errors, initialization errors, and termination errors are the kind of issues that black box testing focuses on.

5.2.4 Output Testing or User Acceptance Testing

User approval of the system under consideration is tested; in this case, it must meetthe needs of the company. When developing, the program should stay in touch with the user and perspective system to make modifications as needed. With regard to the following points, this is done:

- Input Screen Designs,
- Output Screen Designs,

The aforementioned testing is carried out using a variety of test data. The preparation of test data is essential to the system testing process. The system under investigation is then put to the test using the prepared test data. Errors in the system are once again found during testing, fixed using the methods described above, and logged for use in the future.

Automation Testing

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

Benefits of Automation Testing

Detailed reporting capabilities - Test cases for different scenarios are carefully built for automation testing. These planned sequences can cover a lot of ground and produce in-depth reports that are simply impossible for a human to produce.
Improved bug detection - Finding bugs and other flaws in a product is one of the key reasons to test it. This procedure can be made simpler with automation testing. Additionally, it can examine a greater test coverage than perhaps people can.

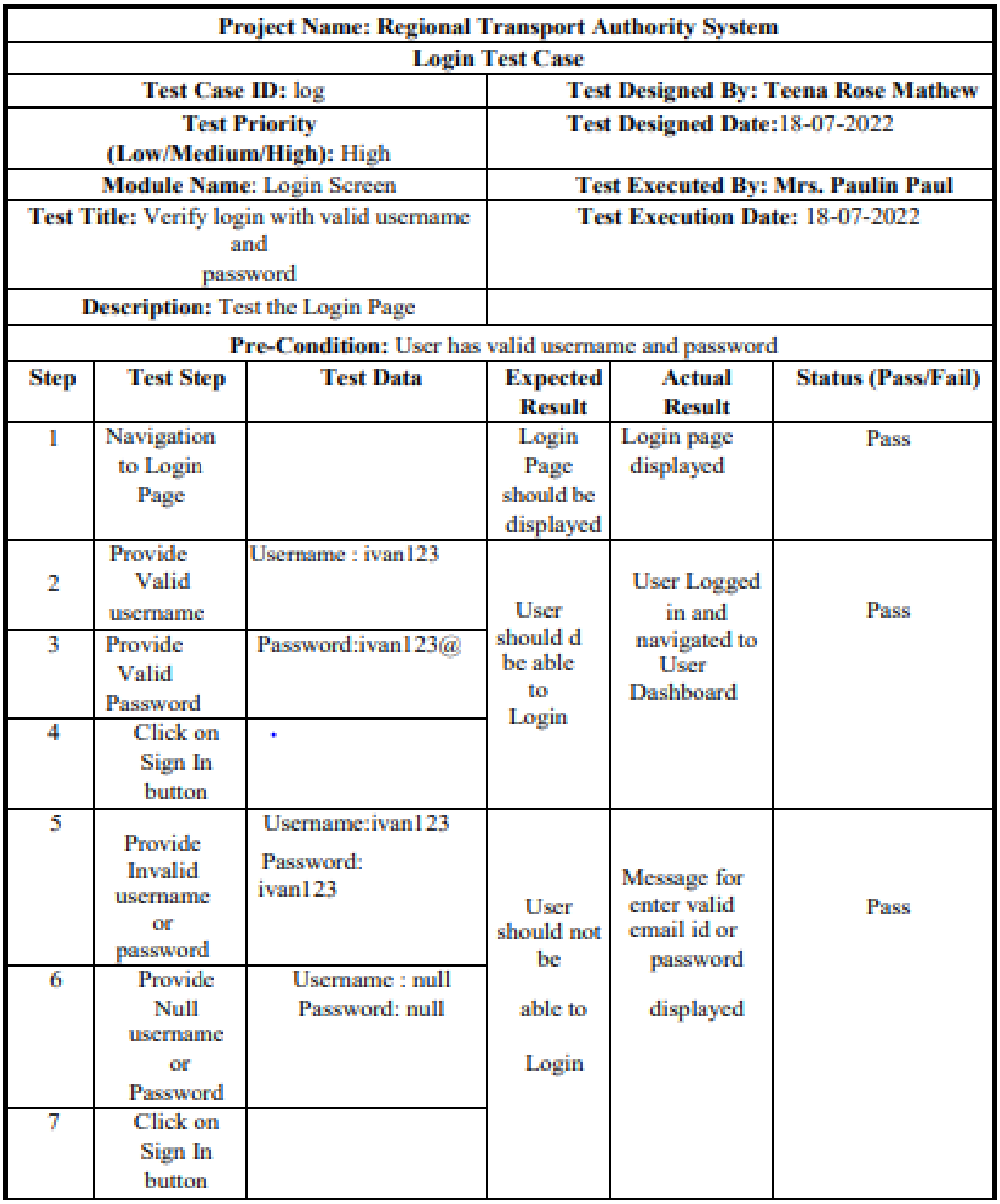
- Testing is a common component of most SaaS and tech firms' operations, which simplifies testing. The key is to keep things as basic as you can. Automation has a lot of advantages. The test scripts can be reused when automating test tools.
- Quickens the testing procedure - Machines and automated technology operate more quickly than people. This is why we employ them, along with increased accuracy. Your software development cycles are subsequently shortened by this.

- Lessens the requirement for human supervision - Tests may be conducted at any time of day, including overnight. Additionally, when done automatically, this can lessen the possibility of human error.

5.2.1 Selenium Testing

An open-source programme called Selenium automates web browsers. It offers a single interface that enables you to create test scripts in a number of different programming languages, including Ruby, Java, NodeJS, PHP, Perl, Python, and C#. Web application testing for cross-browser compatibility is automated using the Selenium testing tool. Whether they are responsive, progressive, or standard, it is utilized to assure high-quality web apps. Selenium is a free software programme.

Test cases for a Login Page



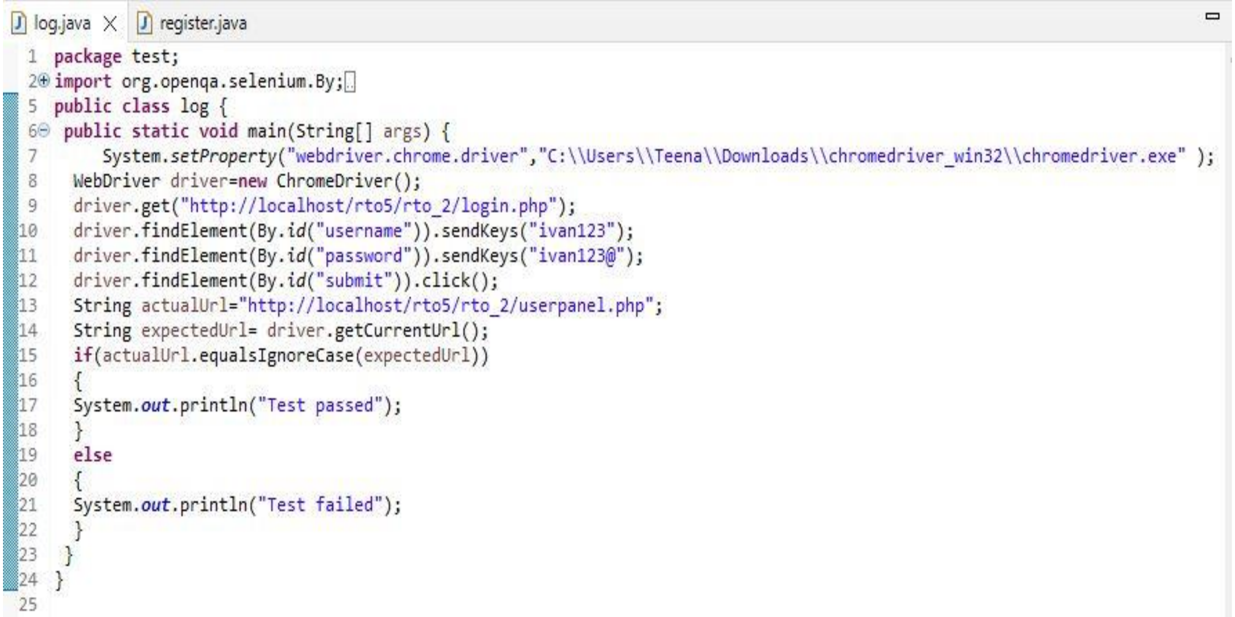
Project Name: Regional Transport Authority System	
Login Test Case	
Test Case ID: log	**Test Designed By: Teena Rose Mathew**
Test Priority (Low/Medium/High): High	**Test Designed Date:**18-07-2022
Module Name: Login Screen	**Test Executed By: Mrs. Paulin Paul**
Test Title: Verify login with valid username and password	**Test Execution Date:** 18-07-2022
Description: Test the Login Page	
Pre-Condition: User has valid username and password	

Step	Test Step	Test Data	Expected Result	Actual Result	Status (Pass/Fail)
1	Navigation to Login Page		Login Page should be displayed	Login page displayed	Pass
2	Provide Valid username	Username : ivan123	User should d be able to Login	User Logged in and navigated to User Dashboard	Pass
3	Provide Valid Password	Password:ivan123@			
4	Click on Sign In button				
5	Provide Invalid username or password	Username:ivan123 Password: ivan123	User should not be able to Login	Message for enter valid email id or password displayed	Pass
6	Provide Null username or Password	Username : null Password: null			
7	Click on Sign In button				

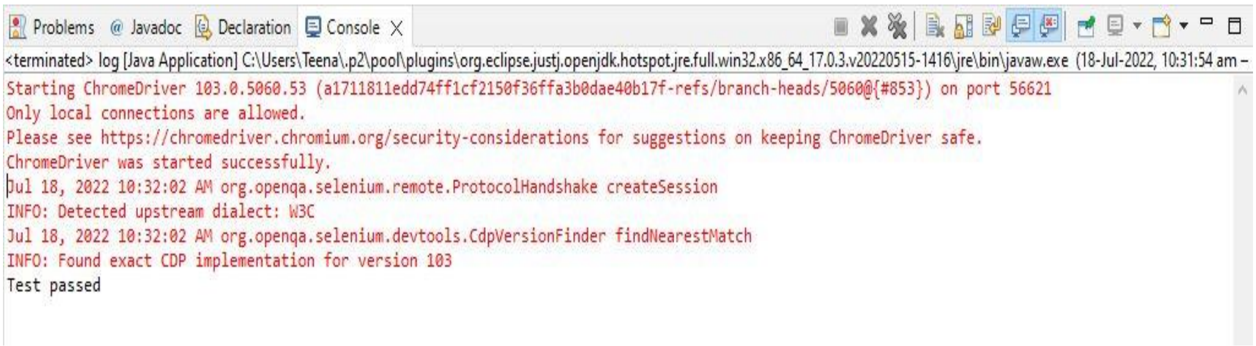
Post-Condition: User is validated with database and successfully login into account. The Account session details are logged in database.

Code package

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

System.setProperty("webdriver.chrome.driver","C:\\Users\\Teena\\Downloads\\chromedriver_win32\\chromedriver.exe" );
  WebDriver driver=new ChromeDriver();
  driver.get("http://localhost/rto5/rto_2/login.php");
  driver.findElement(By.id("username")).sendKeys("ivan123");
  driver.findElement(By.id("password")).sendKeys("ivan123@");
  driver.findElement(By.id("submit")).click();
  String actualUrl="http://localhost/rto5/rto_2/userpanel.php";
  String expectedUrl= driver.getCurrentUrl();
  if(actualUrl.equalsIgnoreCase(expectedUrl))
  {
  System.out.println("Test passed");
  }
  else
  {
  System.out.println("Test failed");
  }
 }
}
```

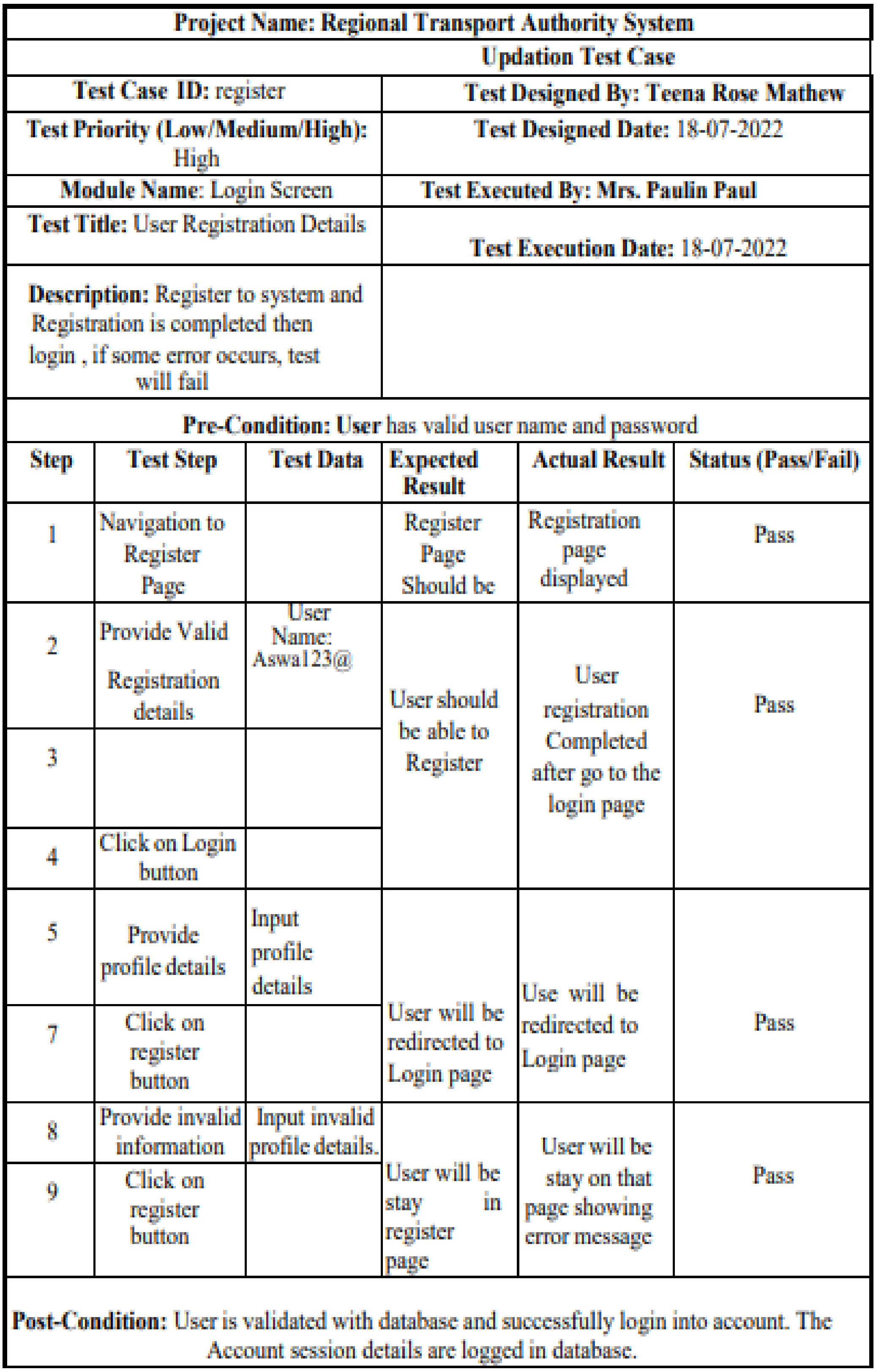


```
package test;
import org.openqa.selenium.By;
public class log {
 public static void main(String[] args) {
     System.setProperty("webdriver.chrome.driver","C:\\Users\\Teena\\Downloads\\chromedriver_win32\\chromedriver.exe" );
  WebDriver driver=new ChromeDriver();
  driver.get("http://localhost/rto5/rto_2/login.php");
  driver.findElement(By.id("username")).sendKeys("ivan123");
  driver.findElement(By.id("password")).sendKeys("ivan123@");
  driver.findElement(By.id("submit")).click();
  String actualUrl="http://localhost/rto5/rto_2/userpanel.php";
  String expectedUrl= driver.getCurrentUrl();
  if(actualUrl.equalsIgnoreCase(expectedUrl))
  {
  System.out.println("Test passed");
  }
  else
  {
  System.out.println("Test failed");
  }
 }
}
```



```
<terminated> log [Java Application] C:\Users\Teena\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (18-Jul-2022, 10:31:54 am –
Starting ChromeDriver 103.0.5060.53 (a1711811edd74ff1cf2150f36ffa3b0dae40b17f-refs/branch-heads/5060@{#853}) on port 56621
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Jul 18, 2022 10:32:02 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 18, 2022 10:32:02 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Test passed
```

Test cases for User Registration



Project Name: Regional Transport Authority System	
	Updation Test Case
Test Case ID: register	**Test Designed By: Teena Rose Mathew**
Test Priority (Low/Medium/High): High	**Test Designed Date:** 18-07-2022
Module Name: Login Screen	**Test Executed By: Mrs. Paulin Paul**
Test Title: User Registration Details	**Test Execution Date:** 18-07-2022
Description: Register to system and Registration is completed then login , if some error occurs, test will fail	
Pre-Condition: User has valid user name and password	

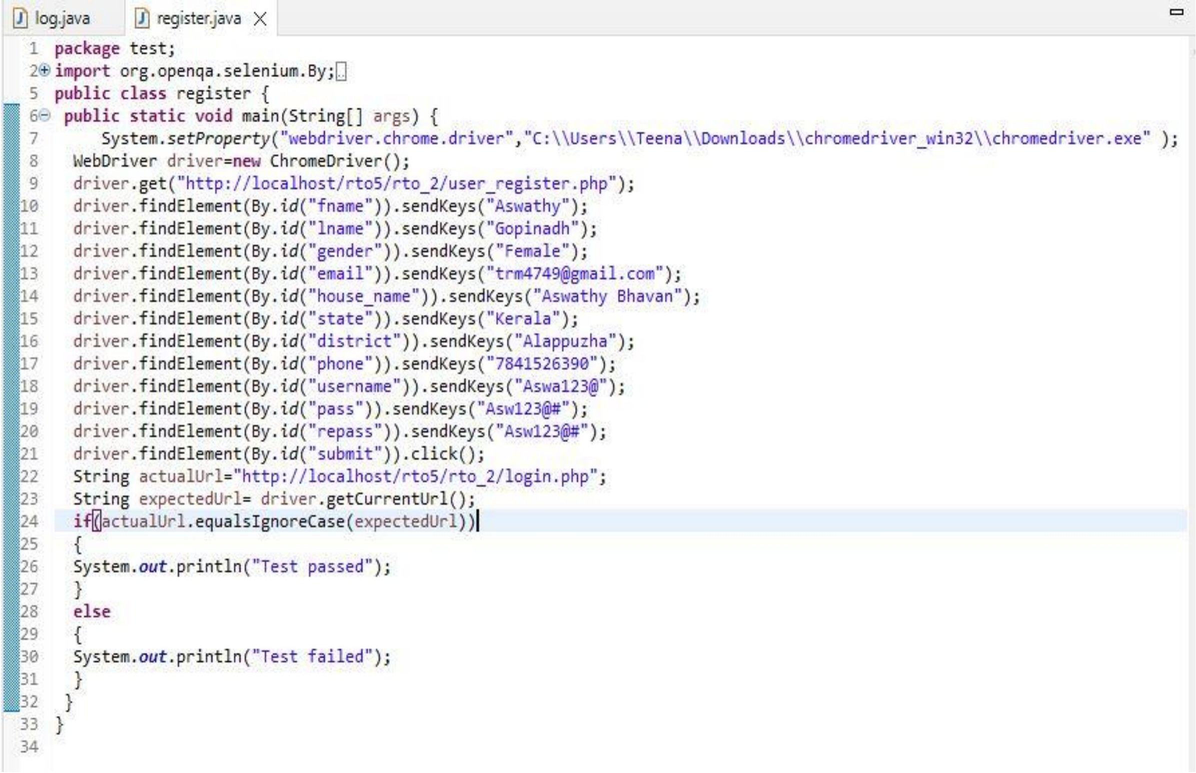
Step	Test Step	Test Data	Expected Result	Actual Result	Status (Pass/Fail)
1	Navigation to Register Page		Register Page Should be	Registration page displayed	Pass
2	Provide Valid Registration details	User Name: Aswa123@	User should be able to Register	User registration Completed after go to the login page	Pass
3					
4	Click on Login button				
5	Provide profile details	Input profile details	User will be redirected to Login page	Use will be redirected to Login page	Pass
7	Click on register button				
8	Provide invalid information	Input invalid profile details.	User will be stay in register page	User will be stay on that page showing error message	Pass
9	Click on register button				

Post-Condition: User is validated with database and successfully login into account. The Account session details are logged in database.

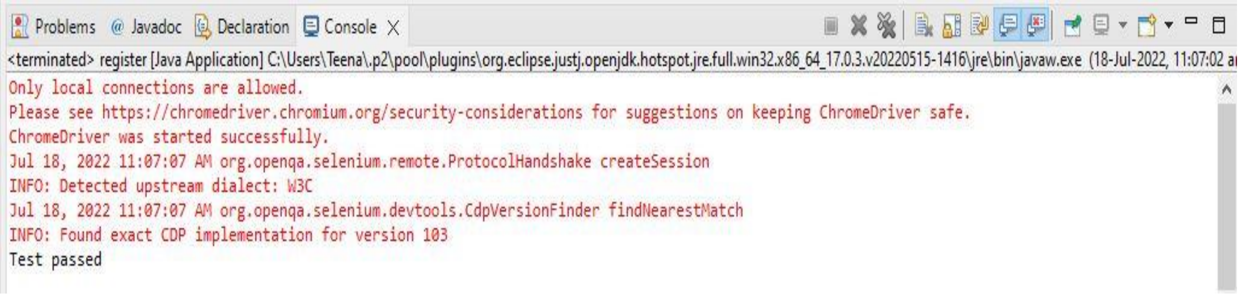
Code package

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

System.setProperty("webdriver.chrome.driver","C:\\Users\\Teena\\Downloads\\chromedr
iver_win32\\chromedriver.exe" );
  WebDriver driver=new ChromeDriver();
  driver.get("http://localhost/rto5/rto_2/user_register.php");
  driver.findElement(By.id("fname")).sendKeys("Aswathy");
  driver.findElement(By.id("lname")).sendKeys("Gopinadh");
  driver.findElement(By.id("gender")).sendKeys("Female");
  driver.findElement(By.id("email")).sendKeys("trm4749@gmail.com");
  driver.findElement(By.id("house_name")).sendKeys("Aswathy Bhavan");
  driver.findElement(By.id("state")).sendKeys("Kerala");
  driver.findElement(By.id("district")).sendKeys("Alappuzha");
  driver.findElement(By.id("phone")).sendKeys("7841526390");
  driver.findElement(By.id("username")).sendKeys("Aswa123@");
  driver.findElement(By.id("pass")).sendKeys("Asw123@#");
  driver.findElement(By.id("repass")).sendKeys("Asw123@#");
  driver.findElement(By.id("submit")).click();
  String actualUrl="http://localhost/rto5/rto_2/login.php";
  String expectedUrl= driver.getCurrentUrl();
  if(actualUrl.equalsIgnoreCase(expectedUrl))
  {
  System.out.println("Test passed");
  }
  else
  {
```

```
 System.out.println("Test failed");
  }
 }
}
```



```
package test;
import org.openqa.selenium.By;
public class register {
 public static void main(String[] args) {
     System.setProperty("webdriver.chrome.driver","C:\\Users\\Teena\\Downloads\\chromedriver_win32\\chromedriver.exe" );
  WebDriver driver=new ChromeDriver();
  driver.get("http://localhost/rto5/rto_2/user_register.php");
  driver.findElement(By.id("fname")).sendKeys("Aswathy");
  driver.findElement(By.id("lname")).sendKeys("Gopinadh");
  driver.findElement(By.id("gender")).sendKeys("Female");
  driver.findElement(By.id("email")).sendKeys("trm4749@gmail.com");
  driver.findElement(By.id("house_name")).sendKeys("Aswathy Bhavan");
  driver.findElement(By.id("state")).sendKeys("Kerala");
  driver.findElement(By.id("district")).sendKeys("Alappuzha");
  driver.findElement(By.id("phone")).sendKeys("7841526390");
  driver.findElement(By.id("username")).sendKeys("Aswa123@");
  driver.findElement(By.id("pass")).sendKeys("Asw123@#");
  driver.findElement(By.id("repass")).sendKeys("Asw123@#");
  driver.findElement(By.id("submit")).click();
  String actualUrl="http://localhost/rto5/rto_2/login.php";
  String expectedUrl= driver.getCurrentUrl();
  if(actualUrl.equalsIgnoreCase(expectedUrl))
  {
  System.out.println("Test passed");
  }
  else
  {
  System.out.println("Test failed");
  }
 }
}
```



```
<terminated> register [Java Application] C:\Users\Teena\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe  (18-Jul-2022, 11:07:02 a
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Jul 18, 2022 11:07:07 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 18, 2022 11:07:07 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Test passed
```

CHAPTER 6

IMPLEMENTATION

6.1 INTRODUCTION

The conceptual design is developed into a usable system during the project's execution phase. Since it assures users that the system will function as planned and be dependable and accurate, it can be viewed as the most crucial stage in developing a successful new system. User documentation and training are its main concerns. Usually, conversion happens either during or after the user's training. The process of converting a freshly revised system design into an operational one is known as implementation, which is essentially the act of putting a new system design into use.

The user department now bears the most of the workload, faces the most disruption, and has the biggest influence on the current system. If the implementation is not well thought out or managed, confusion and mayhem may result.

The entire process of moving from the old system to the new one is referred to as implementation. The new system can be completely different, replace an existing human or automated system, or simply be improved. A reliable systemthat satisfies organizational needs must be implemented properly. System implementation refers to the process of actually using the built system. This comprisesall the processes involved in switching from the old to the new system. Only after extensive testing and if it is determined that the system is operating in accordance with the standards can it be put into use. The system personnel assess the system's viability. As a system is put into place, the complexity of the system analysis and design work required to implement the three essential elements of education and training, system testing, and changeover will rise.

The following tasks are included in the implementation state:

- Careful planning.
- Examination of the system and its limitations.
- Development of transitional approaches.

6.2 IMPLEMENTATION PROCEDURES

The complete installation of the package in the intended environment is referred to as software implementation, as well as to the system's functionality and satisfaction of its intended applications. The software development project is frequently commissioned by someone who will not be using it. People have early reservations about the software, but we must watch out that they do not become more resistant by making sure that:

□ The new system's advantages must be known to the active user.

□ Their faith in the software is increased.

□ The user receives the appropriate instruction so that he feels confident using the application.

Before examining the system, the user must be aware that the server software needs to be running on the server in order to access the results.. The actual process won't happen if the server object is not operational and working on the server.

6.2.1 User Training

The purpose of user training is to get the user ready to test and modify the system. The participants must have confidence in their capacity to advance the objective and reap the rewards of the computer-based system. As systems get more complex, training becomes increasingly important. The user gains knowledge of how to enter data, handle error warnings, query the database, call up routines to generate reports, and do other critical tasks through user training.

6.2.2 Training on the Application Software

The user must first obtain the fundamental training in computer literacy, after which they must be taught how to operate the new application software. In addition to how the screens function, what sort of help is shown on them, what kinds of errors are created while entering data, how each entry is validated, and how to update the date that was entered, this will explain the core concepts of how to use the new system. Therefore, throughout the program's training on the application, the information required by the particular user or group to operate the system or a specific component of the system should be covered. This training maydiffer based on the user group and hierarchical level.

6.2.3 System Maintenance

The mystery of system development is maintenance. When a software product is in the maintenance stage of its lifecycle, it is actively working. A system should be properly maintained after it has been effectively implemented. System maintenance is a crucial phase in the software development life cycle. In order for a system to be able to respond to changes in the system environment, maintenance is required. Most importantly, software maintenance involves much more than just "Finding Mistakes."

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

A good web program for RTO work completion online is the Regional Transport Authority System. Here, we are creating the kinds of modules that lessen manual RTO labour while also saving the user's time. Reduce corruption in the transportation department noticeably. The license must be stored securely.

7.2 FUTURE SCOPE

The traffic police can also benefit from this technique. To better managehabitual traffic rule violators under the direction of the traffic police. The database of registration numbers and the history of drivers license holders are both in the possession of the traffic police. When a traffic police officer enters the details of a car that they have observed breaking traffic laws, it provides all of the vehicle's information, including the owner's name and address as well as the make, model, and other specifics of the vehicle. Additionally, the information on the holder of thedriver's license would be accessible. Therefore, increased penalties would be appliedto repeat offenders of traffic laws. Any fake license plates would be discovered right away.

CHAPTER 8

BIBLIOGRAPHY

REFERENCES:

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

WEBSITES:

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

CHAPTER 9

APPENDIX

9.1 Sample Code

Login.php

```
<?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($con,"insert into tbl_register(fname,lname,gender,email,paddress,caddress,phone_no,username,password,repassword)values('$fname','$lname','$gender','$email','$paddress','$caddress','$phone','$username','$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 Registeration</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">
    <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');
```

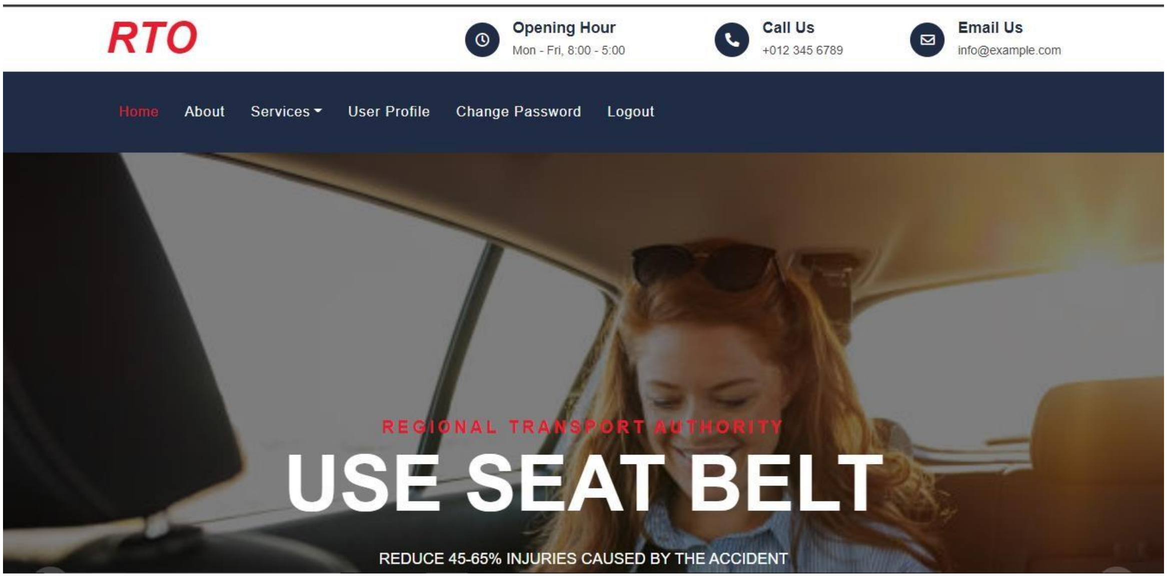
```
     return false;
   }

   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>
```

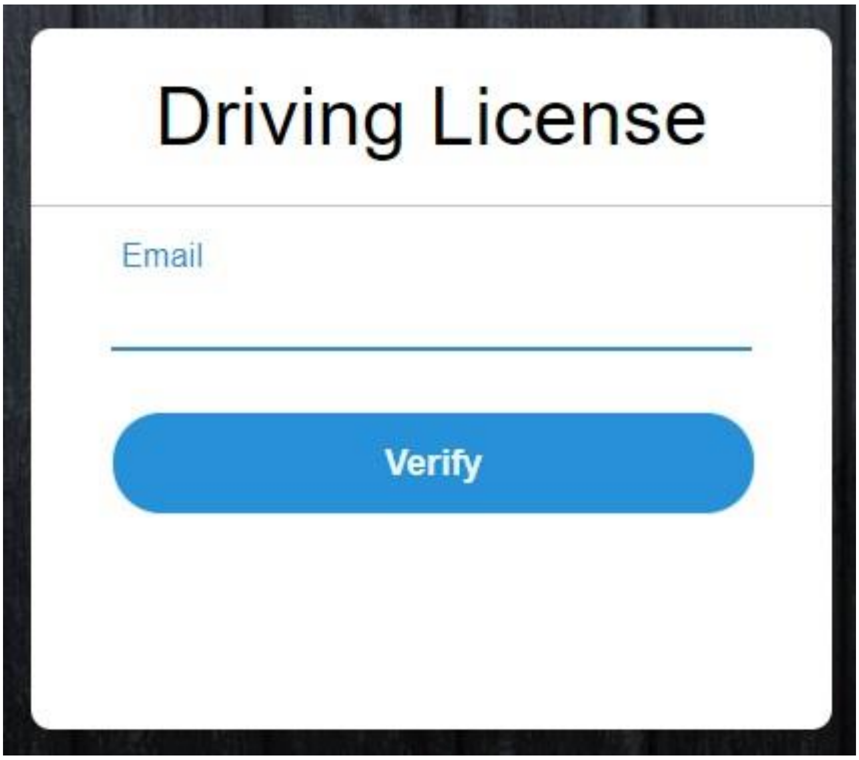
```
</body>

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

9.2 Screen Shots

User Home page

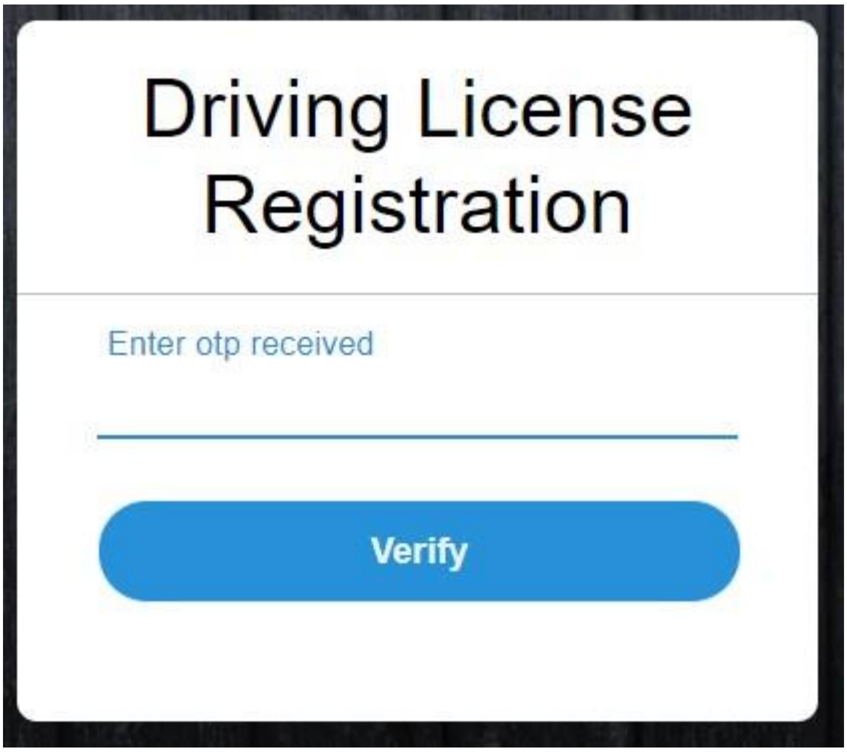


Driving License Registration page



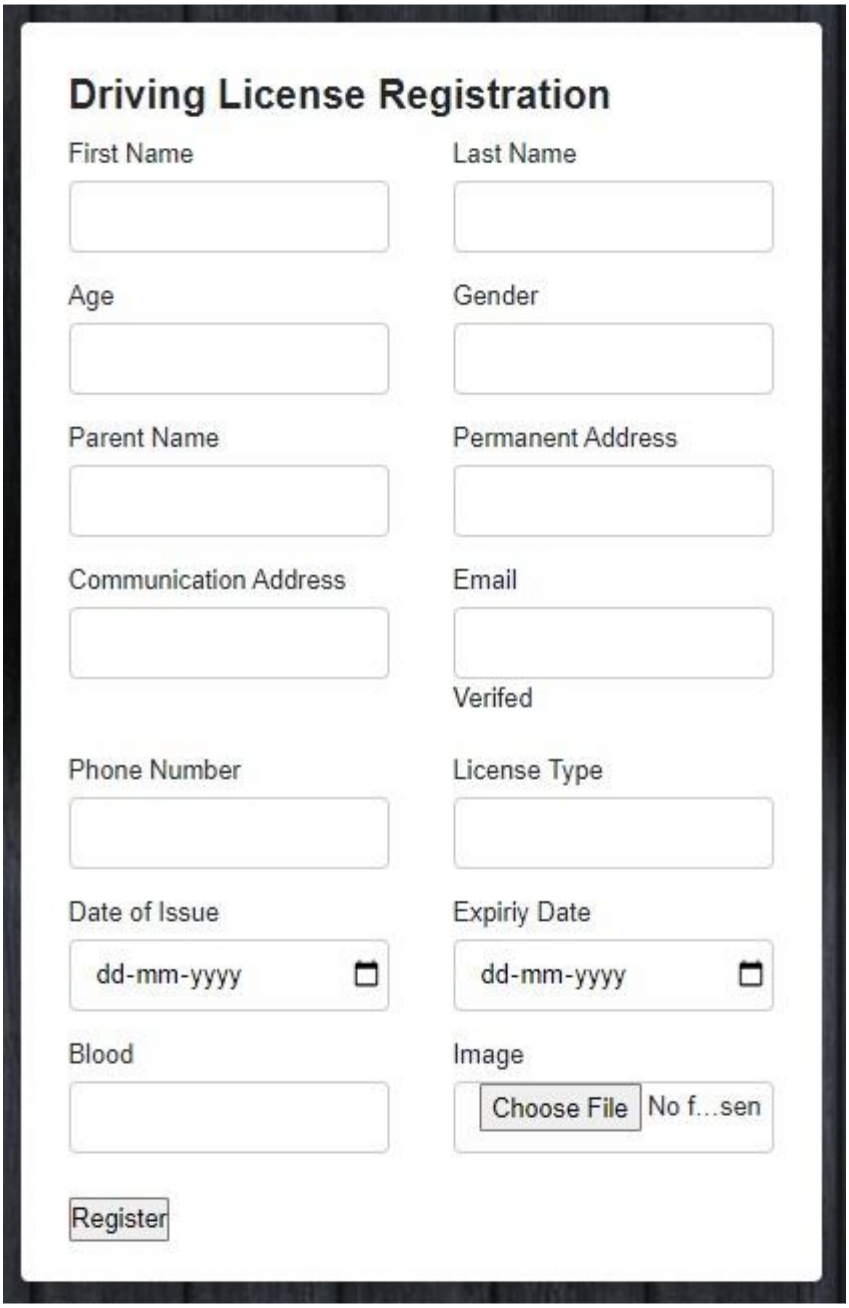
Driving License

Email



Driving License Registration

Enter otp received



Driving License Registration

First Name

Last Name

Age

Gender

Parent Name

Permanent Address

Communication Address

Email

Verifed

Phone Number

License Type

Date of Issue

dd-mm-yyyy

Expiriy Date

dd-mm-yyyy

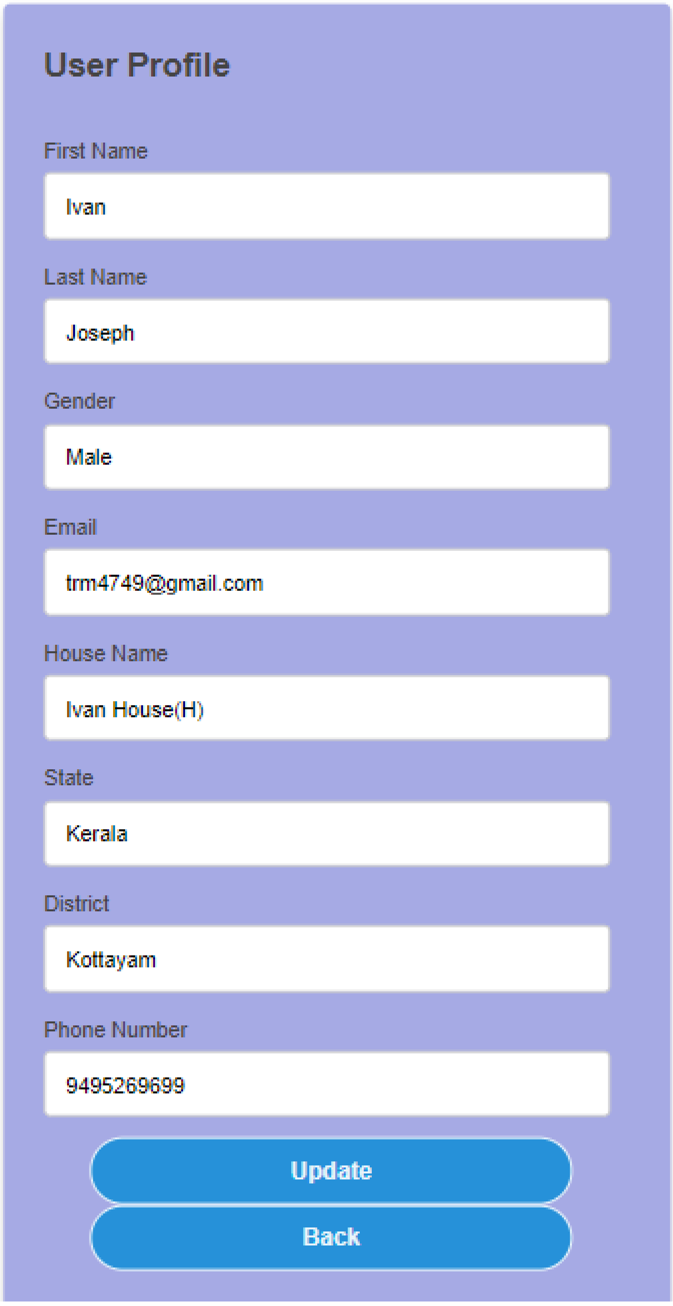
Blood

Image

Choose File No f...sen

Register

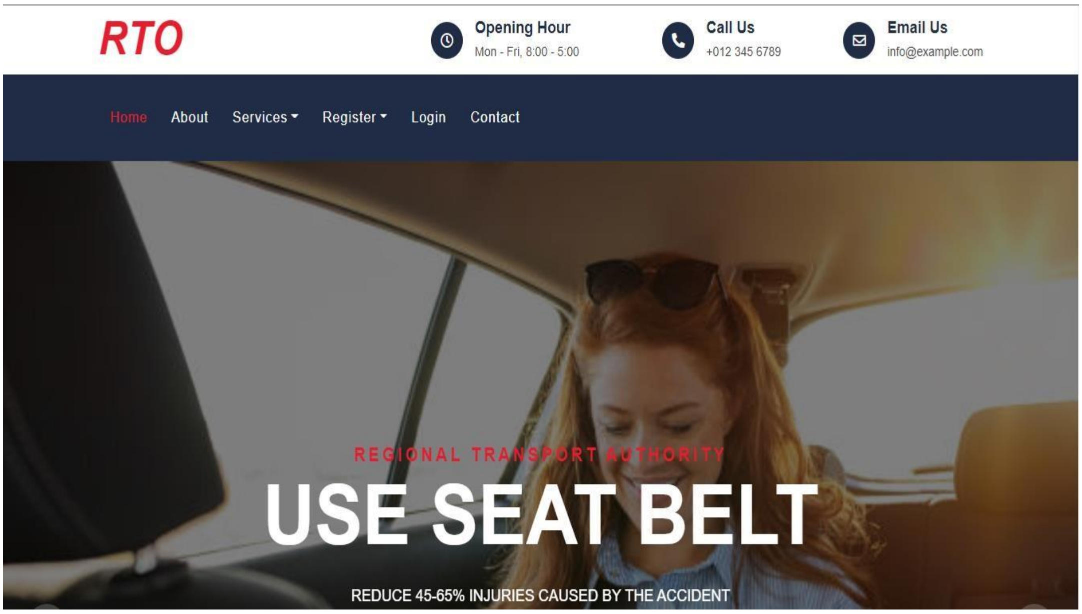
View Profile of User



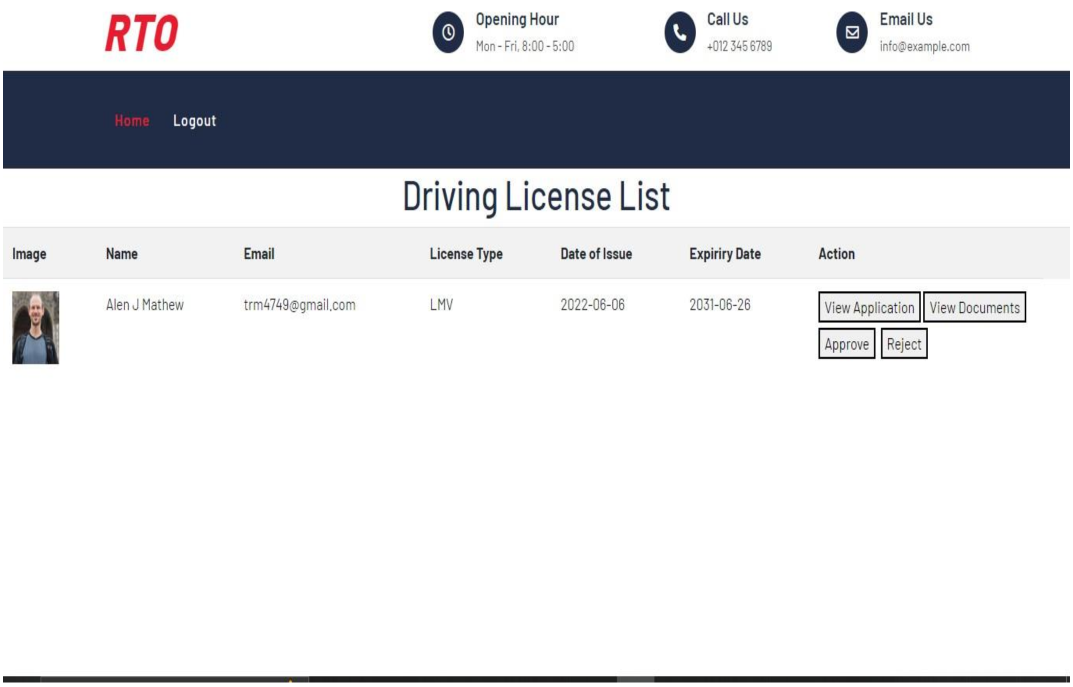
User Profile

First Name	Ivan
Last Name	Joseph
Gender	Male
Email	trm4749@gmail.com
House Name	Ivan House(H)
State	Kerala
District	Kottayam
Phone Number	9495269699

User Home Page



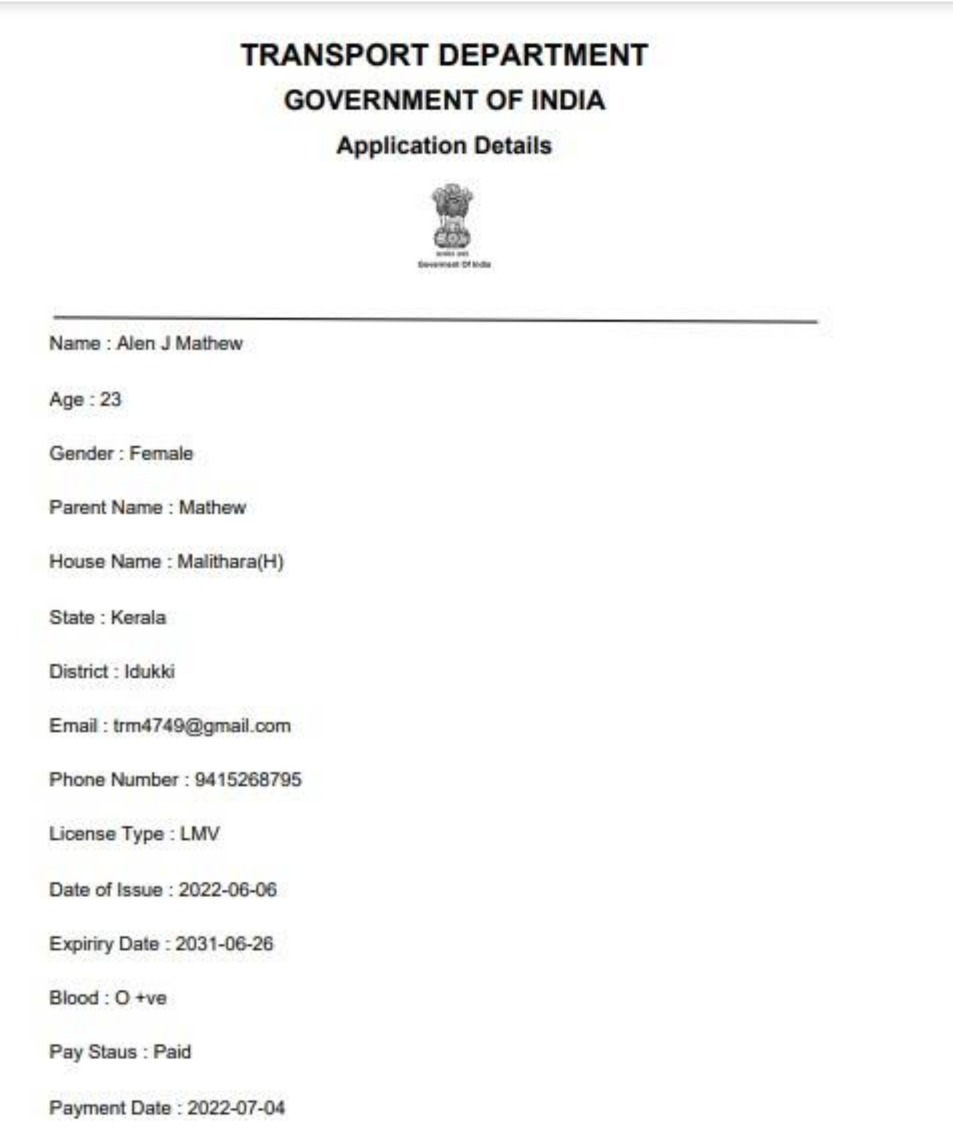
Sub-officer Driving license View page



Driving License List

Image	Name	Email	License Type	Date of Issue	Expiriy Date	Action
	Alen J Mathew	trm4749@gmail.com	LMV	2022-06-06	2031-06-26	View Application View Documents Approve Reject

Sub-officer View details in pdf format



TRANSPORT DEPARTMENT

GOVERNMENT OF INDIA

Application Details

Name : Alen J Mathew

Age : 23

Gender : Female

Parent Name : Mathew

House Name : Malithara(H)

State : Kerala

District : Idukki

Email : trm4749@gmail.com

Phone Number : 9415268795

License Type : LMV

Date of Issue : 2022-06-06

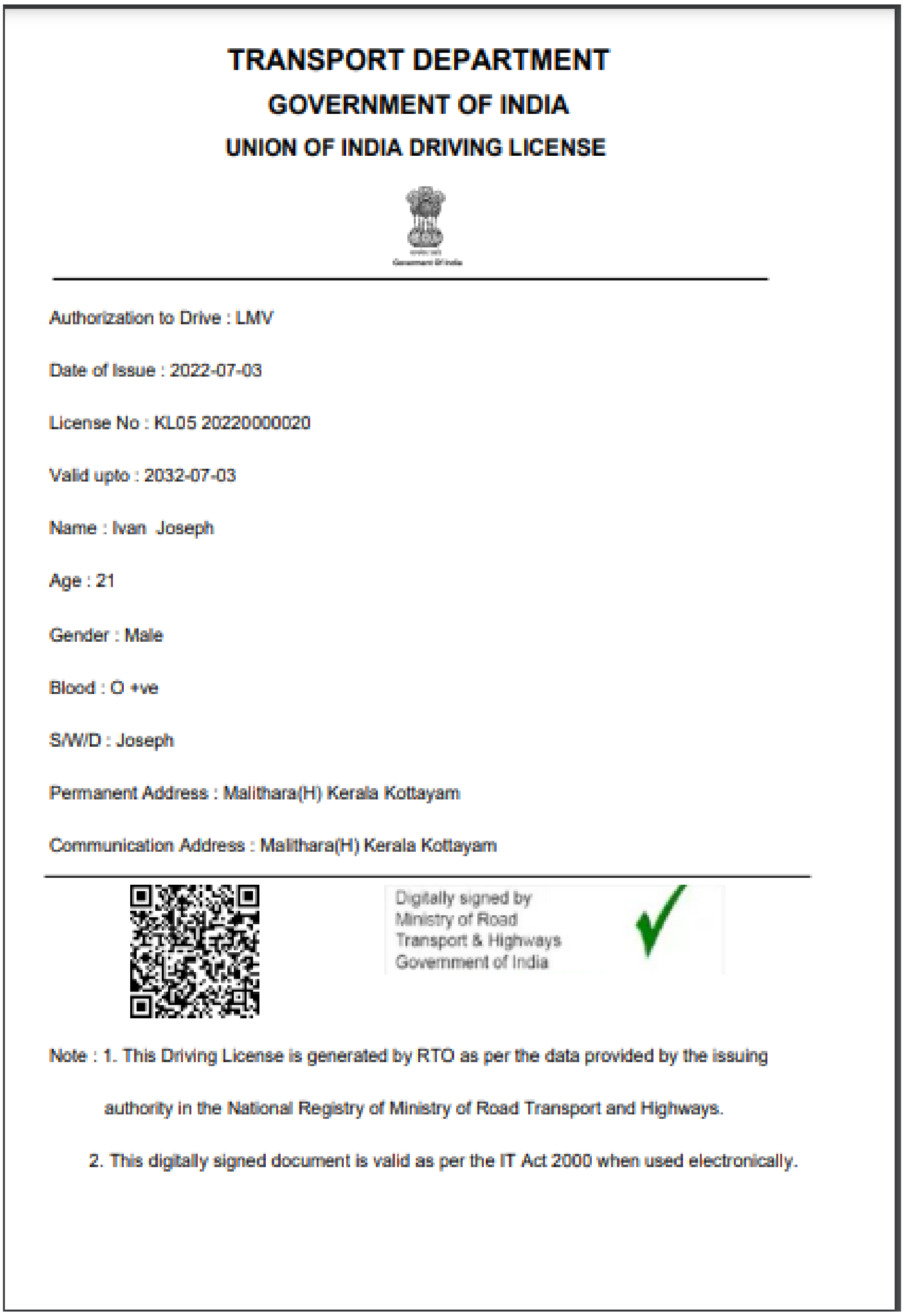
Expiriry Date : 2031-06-26

Blood : O +ve

Pay Staus : Paid

Payment Date : 2022-07-04

Certificate generation for driving license in pdf format



TRANSPORT DEPARTMENT

GOVERNMENT OF INDIA

UNION OF INDIA DRIVING LICENSE

Authorization to Drive : LMV

Date of Issue : 2022-07-03

License No : KL05 20220000020

Valid upto : 2032-07-03

Name : Ivan Joseph

Age : 21

Gender : Male

Blood : O +ve

S/W/D : Joseph

Permanent Address : Malithara(H) Kerala Kottayam

Communication Address : Malithara(H) Kerala Kottayam

Digitally signed by
Ministry of Road
Transport & Highways
Government of India

Note : 1. This Driving License is generated by RTO as per the data provided by the issuing authority in the National Registry of Ministry of Road Transport and Highways.

2. This digitally signed document is valid as per the IT Act 2000 when used electronically.