INDUSTRIAL GROUPS OF MANAGEMENT

ABSTRACT

The project “Industrial Groups of Management” is a web-based application. This Industrial Groups of Management is based on two business sectors such as Transportation Management and Builders Management. The Transportation Management provides the facility of online transport booking service around the country to individual and businesses. The material types that are mostly transported from various locations through road transport. This Application completely allow the users to choose the types of vehicles and destination of pick-up and delivery and scheduling dates. The user can see the information about Transportation with multimedia content (pictures, verbal) and the list of instruction about prices also available in the websites. If interested user can book the request through this application and then the admin will send the confirmation message to the respective user. The Builders Management provides the facility of online booking service the customers can add and book the individual Villas or Apartment based on their location, availability, and area. In this application this takes the workload simple for users to book a Villas or Apartment by giving the locations. The user can see the information about Villa or Apartments with multimedia content (pictures, videos, verbal) and if interested user can book the request through this application and then the admin will send the confirmation message to the respective user.

CHAPTER - I

INTRODUCTION

ABOUT THE PROJECT

The project entitled “Industrial Groups of Management” is a web-based application. This Industrial Groups of Management is based on two business sectors such as Transportation Management and Builders Management. Industrial Groups of Management is developed by html, css, php, mysql which mainly focus basic operations in industries like adding new user, booking the services, information facility to book the services. This Web application mainly used by clients of two sectors (Transport and Builders) and administrator. Admin can view the reports of transport and residential booking and details of users and admin maintains confirmation notification. Users can login into the user module to book the services and view the booking status at same module. The notification will be sent to the particular user about the confirmation of booking. This will help the users to know about confirmation of their booking. Industrial Groups of Management makes easy and requires less man power and less time for maintaining the records of the transportation and residential firm. In Industrial Groups of Management everything is computerized, user friendly is with good look and its totally transparent.

CHAPTER – II

2. SYSTEM STUDY

2.1 EXISTING SYSTEM

In existing system, it is hard for user to book the services in online. Without registering, user cannot able to book the services provider. The existing websites didn't focus on local transportation on transport management.

2.2 DISADVANTAGES OF EXISTING SYSTEM

- In existing system users have to register their details for viewing the content information, thus users sometimes feel annoying to register.
- Users have to pay, to list the services.

2.3 PROPOSED SYSTEM

- It is a web-application for transportation and builder's services both service at the same application. So, this web-application concentrates more on this.
- In proposed system users can view the list of services information's without registering the user. But this web-application will provide the user registration for the booking.
- In this module admin send the confirmation to the respective users.

2.4 ADVANTAGES OF PROPOSED SYSTEM

- Users don't need to pay or register, to view the services.
- A notification will be sent to user about booking and other details.
- The website will be in a user friendly so the user can feel easier to use the website.

2.5 PROBLEM DEFINITION AND DESCRIPTION

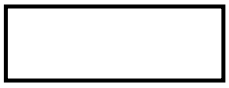
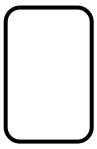
- This project will define the problem of the Industrial Groups of Management that will overcome the existing system.

3.3 DATA FLOW DIAGRAM

A data flow diagram is a graphical representation of the flow of data through an information system and is a key component of designing a successful database application. Thus, DFD is refined till each process performs a simple function.

A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformations and the lines represent data flow in the system. Individuals seeking to draft a data flow diagram must identify external inputs and outputs, determine how the inputs and outputs relate to each other, and explain with graphics how these connections relate and what they result in. This type of diagram helps business development and design teams visualize how data is processed and identify or improve certain aspects.

Data Flow Symbols:

Symbol	Description
	An entity. A source of data or a destination for data
	A process or task that is performed by the system.
	A data store, a place where data is held between processes.
	A data flows.

LEVEL 0

Fig 3.3.2 - Level – 0 Data Flow Diagram

LEVEL 1

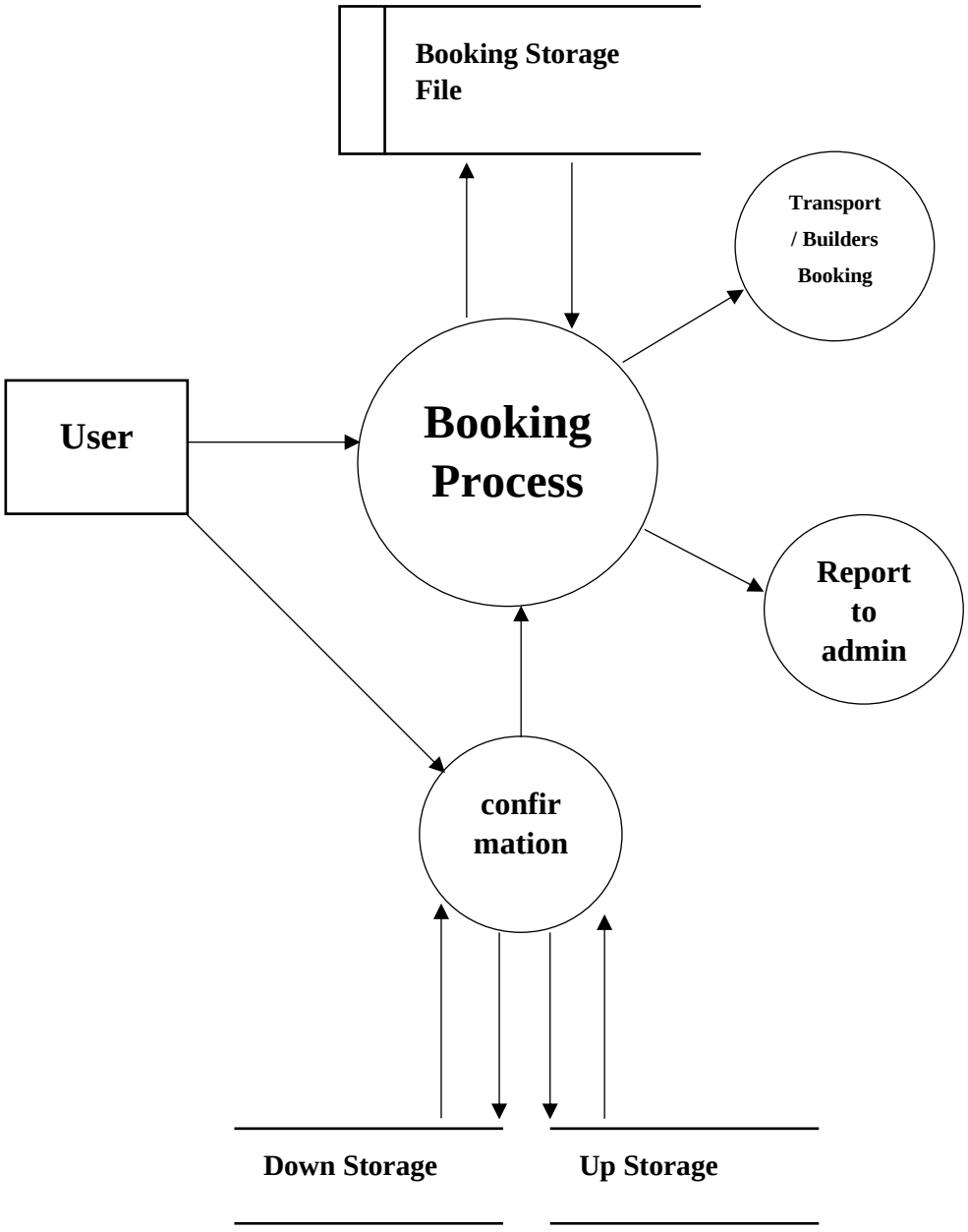


Fig 3.3.3 - Level – 1 Data Flow Diagram

LEVEL 2

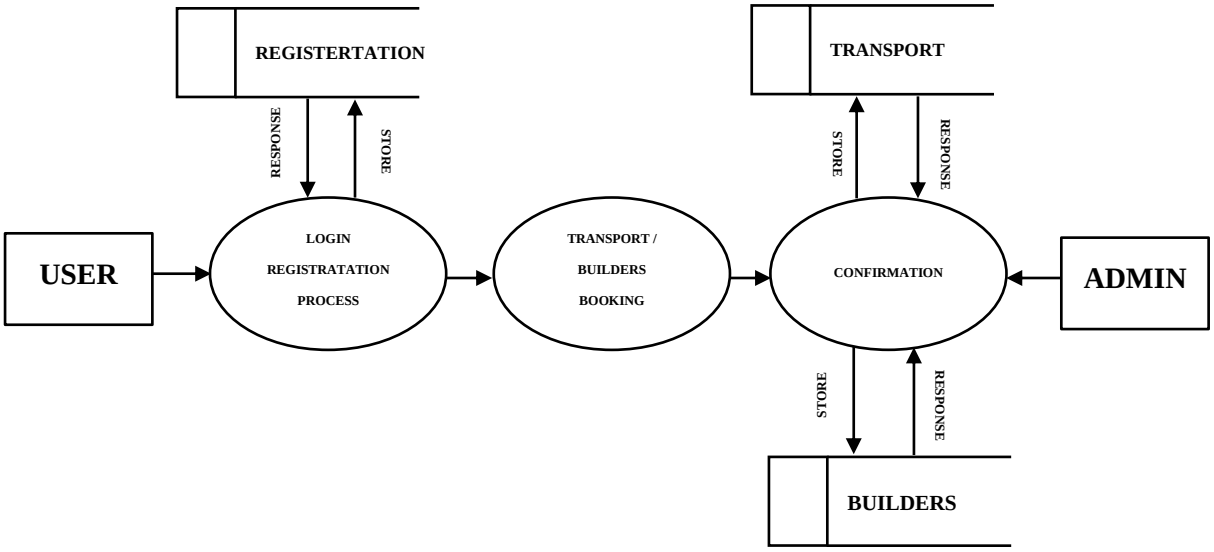


Fig 3.3.4 - Level – 2 Data Flow Diagram

CHAPTER – IV

4. SYSTEM DESIGN

4.1 ARCHITECTURE DESIGN

A system architecture or systems architecture is the conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system. System architecture can comprise system components, the externally visible properties of those components, the relationships (e.g., the behaviour) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture; collectively these are called architecture description languages (ADLs).

Various organizations define systems architecture in different ways, including:

- An allocated arrangement of physical elements which provides the design solution for a consumer product or life-cycle process intended to satisfy the requirements of the functional architecture and the requirements baseline.
- Architecture comprises the most important, pervasive, top-level, strategic inventions, decisions, and their associated rationales about the overall structure (i.e., essential elements and their relationships) and associated characteristics and behaviour.
- If documented, it may include information such as a detailed inventory of current hardware, software and networking capabilities; a description of long-range plans and priorities for future purchases, and a plan for upgrading and/or replacing dated equipment and software
- The composite of the design architectures for products and their life-cycle processes..

ARCHITECTURE DESIGN

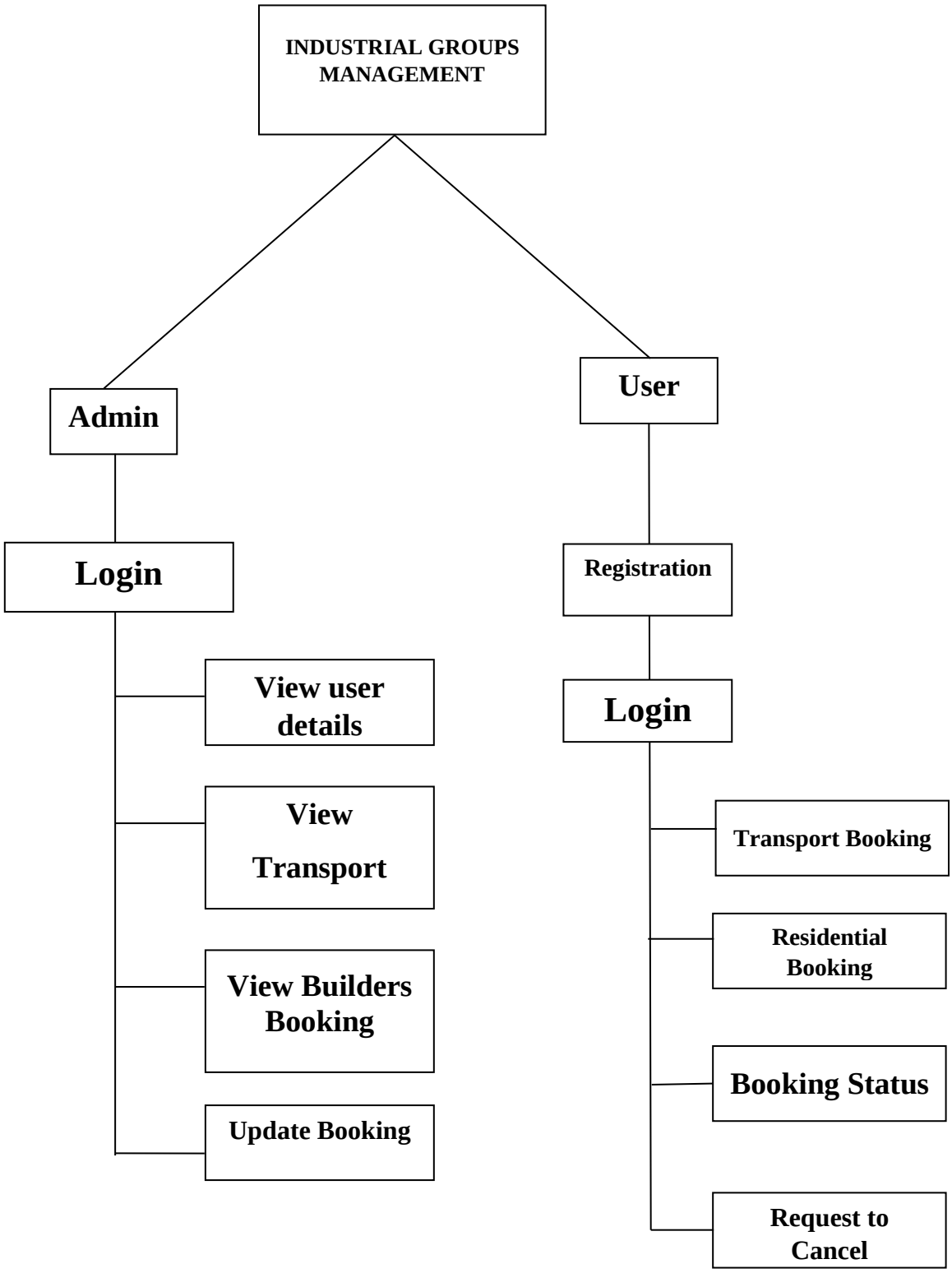


Fig 4.1 ARCHITECTURE DESIGN

4.2 I/O FORM DESIGN

INPUT DESIGN

Input design is the process of converting user-oriented description. Input design requires consideration of the needs of the data entry operator.

Three data entry considerations are:

- The field length must be documented.
- The sequence of fields must match the sequence of the field on the source Document.
- The data format must be identified to the data entry operator.

Main objective in input design is the process of converting user-oriented inputs to a computer-based format. The files and database are maintained through the timely and accurate input of data. Volume of information, frequency, accuracy and verification requirements are considering in the section of input design. Other objectives are to ensure that input is acceptable and understandable to the user.

OUTPUT DESIGN

The output screens are designed in the format replies to queries. An application is successful only when it can provide an efficient and effective report. The report generated must be useful to the organization and for the history reference. Each report has little at its top, which denotes query for which the report stands for. The report is printed in such a manner, that all the required information is given under the proper subheading and with enough space between the fields, so that they do not overlap one another. User is allowed to make queries based on any column from any table. Data may be retrieved from more than one related table. The query results may be displayed in two styles viz. Report and Grid. The user can specify any criteria for the query. Also record retrieved can be Store in the ascending or descending order of anything.

ADMIN LOGIN

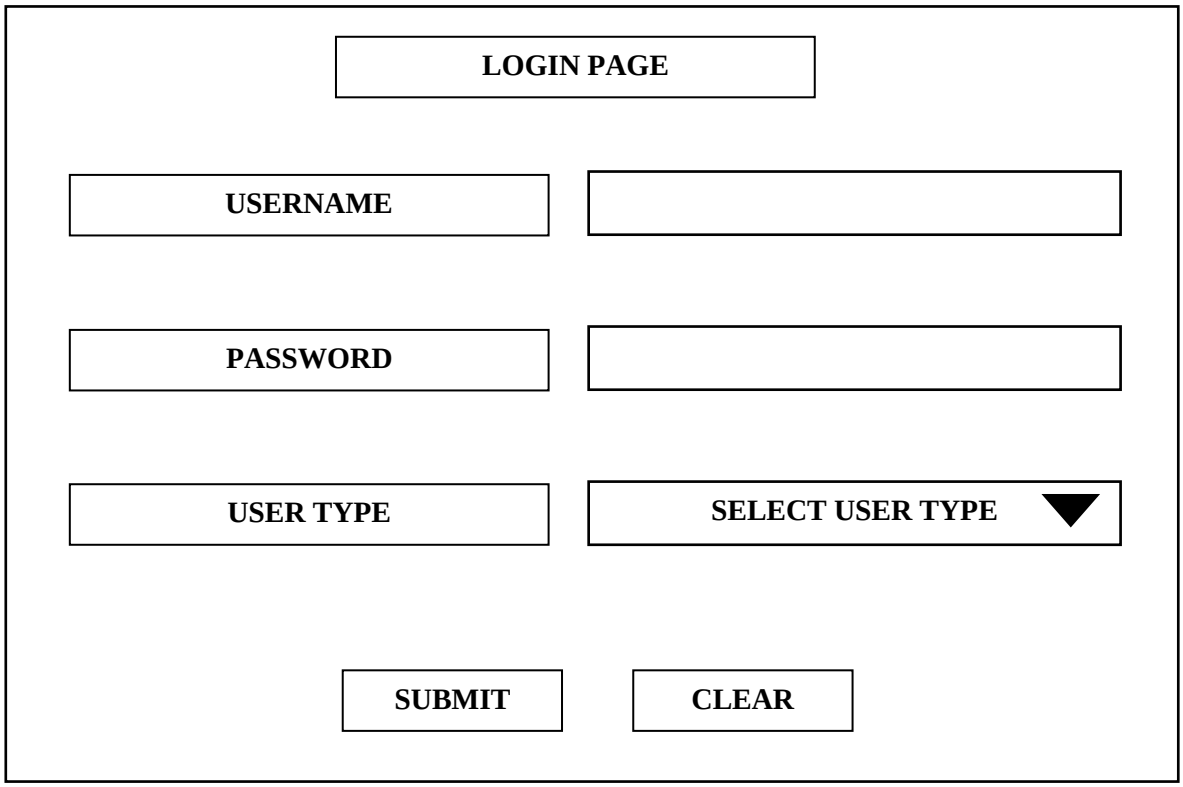


Fig 4.2.1 Admin Login

USER LOGIN

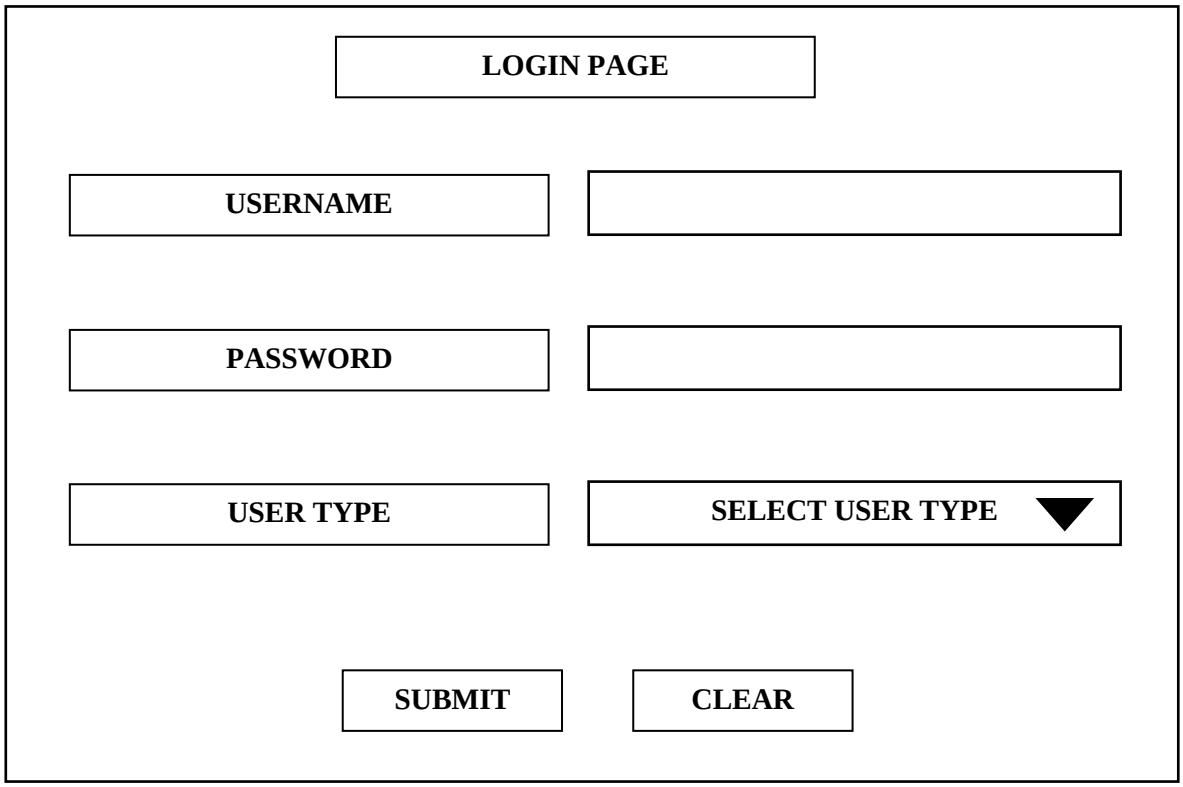


Fig 4.2.2 User Login

Registration

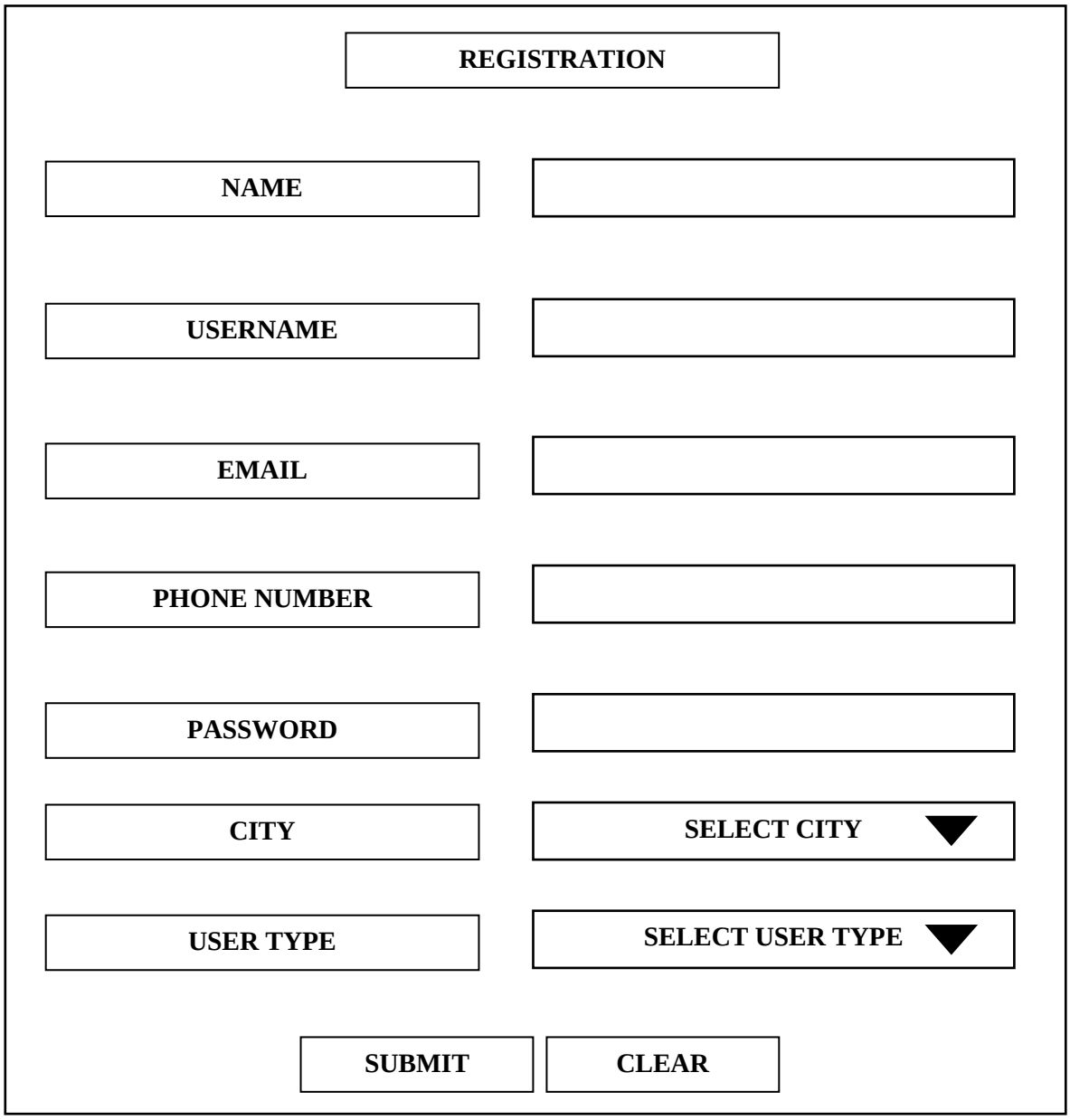


Fig 4.2.3 User Registration

Transport Registration

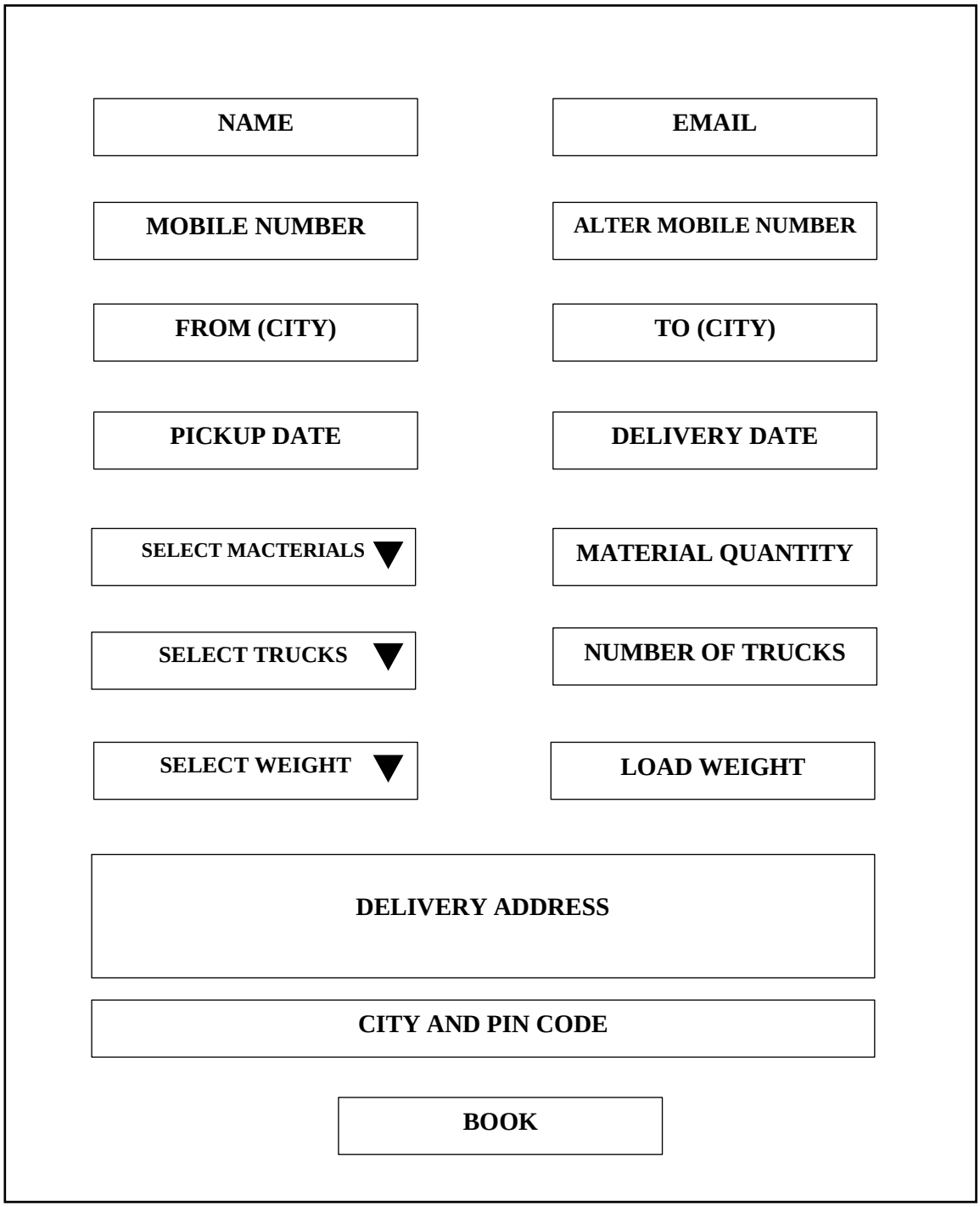


Fig 4.2.4 Transport Registration

Builders Registration

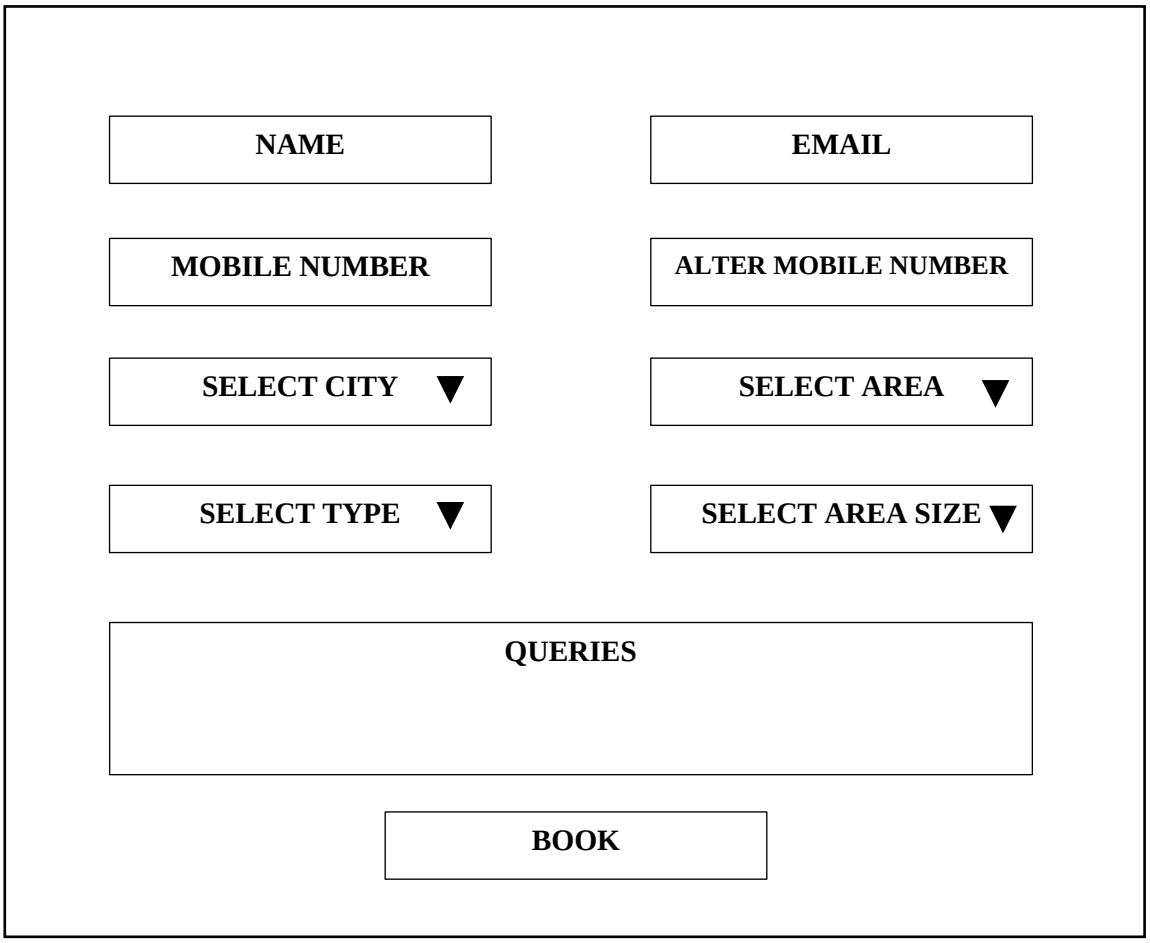


Fig 4.2.5 Builders Registration

4.3 TABLES

A table is a data structure that organizes information into rows and columns. It can be used to both store and display data in a structured format. For example, databases store data in tables so that information can be quickly accessed from specific rows. Websites often use tables to display multiple rows of data on page. Spreadsheets combine both purposes of a table by storing and displaying data in a structured format.

Databases often contain multiple tables, with each one designed for a specific purpose. For example, a company database may contain separate tables for employees, clients, and suppliers. Each table may include its own set of fields, based on what data the table needs to store. In database tables, each field is considered a column, while each entry (or record), is considered a row. A specific value can be accessed from the table by requesting data from an individual column and row.

TABLE NAME: tb_reg

Field	Type	Null
ename	Varchar (50)	No
adds	Varchar (50)	No
city	Varchar (50)	No
phone	Int (10)	No
uname	Varchar (50)	No
pass	Varchar (50)	No
utype	Varchar (75)	No

Table: 4.3.1

TABLE NAME: tb_trans

Field	Type	Null
id	int (11)	No
ename	varchar (50)	No
adds	varchar (100)	No
city	varchar (50)	No
phone	int (10)	No
phone1	int (10)	No
email	varchar (100)	No
city	varchar (50)	No
tcity	varchar (50)	No
pick	varchar (50)	No
deli	varchar (50)	No
cata	varchar (100)	No
tru	varchar (100)	No
moq	varchar (50)	No
nos	varchar (50)	No
lw	varchar (50)	No
low	varchar (50)	No
status	varchar (50)	No
uname	varchar (50)	No
request	varchar (50)	No

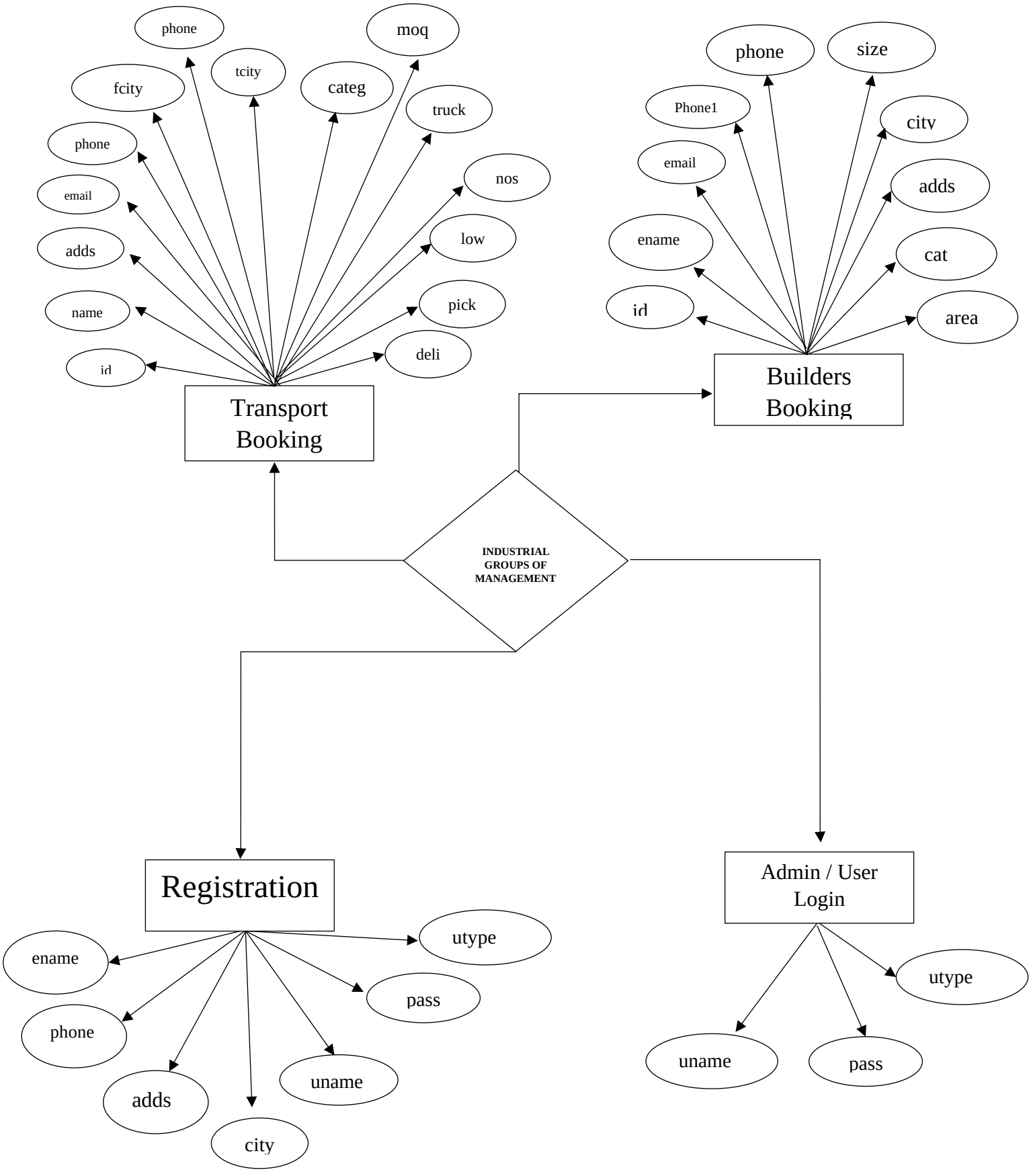
Table: 4.3.2

TABLE NAME: tb_build

Field	Type	Null
id	int (10)	No
ename	varchar (50)	No
adds	varchar (100)	No
city	varchar (50)	No
phone	int (10)	No
phone1	int (10)	No
email	varchar (50)	No
cat	varchar (50)	No
area	varchar (50)	No
size	varchar (25)	No
status	varchar (50)	No
uname	varchar (50)	No
request	varchar (50)	No

Table: 4.3.3

4.4 ENTITY RELATIONSHIP DIAGRAM



4.5 DATA DICTIONARY

Registration

Field name	Type	Description	Sample values
Ename	Varchar (50)	Store to the ename	Tamil
Uname	Varchar (50)	Store to the uname	Tamil20
Email	Varchar (50)	Store to the email	Tml@gmail.com
Phone	Int (10)	Store to the phone	9874561230
Pass	Varchar (50)	Store to the Pass	********
City	Varchar (50)	Store to the city	Trichy
Utype	Varchar (50)	Store to the utype	Transport

Table 4.5.1 tb_reg

Transport Booking

Field name	Type	Description	Sample values
Id	Int (10)	Store to the id	101
Ename	Varchar (50)	Store too the Ename	Tamil
Email	Varchar (100)	Store to the email	Tml@gmail.com
Phone	Int (10)	Store to the phone	1234567890
Phone 1	Int (10)	Store too the Phone1	0987456123
Fcity	Varchar (50)	Store too the Fcity	Trichy
Tcity	Varchar (50)	Store too the Tcity	Chennai
City	Varchar (50)	Store too the City	Trichy
Pick	Varchar (50)	Store too the Pick	8.03.2022
Deli	Varchar (50)	Store to the deli	10.03.2022
Cata	Varchar (50)	Store to the Cata	House hold
Moq	Varchar (50)	Store to the MOQ	10
Tru	Varchar (50)	Store to the Tru	Flatbed Truck
Nos	Varchar (50)	Store to the Nos	5
Lw	Varchar (50)	Store to the Lw	In Kgs
Low	Varchar (50)	Store to the Low	70kgs
Adds	Varchar (100)	Store to the Adds	Trichy
Status	Varchar (50)	Store to the status	Accepted
Request	Varchar (50)	Store to the request	Request to cancel

Table 4.5.2 tb_trans

Builders Booking

Field name	Type	Description	Sample values
Id	Int (10)	Store to the id	101
Ename	Varchar (50)	Store too the ename	Tamil
Email	Varchar (50)	Store to the email	Tml@gmail.com
Phone	Int (10)	Store to the phone	1234567890
Phone 1	Int (10)	Store too the Phone1	0987456123
Cat	Varchar (50)	Store too the cat	House
Area	Varchar (50)	Store too the area	Kk Nagar
City	Varchar (50)	Store too the city	Trichy
Size	Varchar (50)	Store too the Size	2000sqft
Adds	Varchar (75)	Store to the adds	House should be good
Status	Varchar (50)	Store to the status	Accepted
Request	Varchar (50)	Store to the request	Request to cancel

Table 4.5.3 tb_building

CHAPTER – V

5 SYSTEM DEVELOPMENT

5.1 FUNCTIONAL DOCUMENTATION

MODULES

- Admin
    - Login
    - View User Details
    - User Booking Details
- User
    - Register
    - Login
    - Transport Booking
    - Builders Booking
    - View Booking Status

MODULE DESCRIPTION

Admin

- **Login**

  Using this module admin enters user name and password and the system checks whether it is valid or not valid. If it is valid, then admin can login and if it is invalid, then the system will not allow admin to log in.

- **View User Details**

  In this module admin to know the information about user details etc.

- **View Booking Details**

  In this module admin to know the booking status of transport and builders.

User

- **Register**

  New users register through this module. User have to register themselves into the system to create an account. After registering successfully, they can then login into the system by entering our details.

- **Login**

  Using this module user enters user name and password and the system checks whether it is valid or not valid. If it is valid, then user can log in and if it is invalid, then the system will not allow user to log in.

- **Transport Booking**

  In this module, the user can book his/her transportation booking details like case name, phone number, email, city, truck type etc.

- **Builders Booking**

  In this module, the user can book his/her residential booking details like case name, phone number, email, city, house type etc.

- **View Booking Status**

  In this module used to view the booking status. The booking status contain user booking information’s etc. and this module is used to confirmation of user booking.

5.2 SPECIAL FEATURES OF LANGUAGE/UTILITY

FRONT END

Hypertext Preprocessor (a recursive acronym, originally personal home page) is a general-purpose scripting language that was originally designed for web development to produce dynamic web pages. For this purpose, PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module, which generates the web page document. As a general-purpose programming language, PHP code is processed by an interpreter application in command-line mode performing desired operating system operations and producing program output on its standard output channel. It may also function as a graphical application. PHP is available as a processor for most modern web servers and as a standalone interpreter on most operating systems and computing platforms.

PHP was originally created by Rasmus Lerdorfins 1995 and has been in continuous development ever since. The main implementation of PHP is now produced by the PHP Group and serves as the de facto standard for PHP as there is no formal specification. PHP is free software released under the PHP License.

CONTENTS

HISTORY

Rasmus Lerdorf, who wrote the original Common Gateway Interface component, and Andi Gutmans and Zeev Suraski, who rewrote the parser that formed PHP 3 PHP originally stood for personal home page. Its development began in 1994 when the Danish/Greenlandic programmer RasmusLerd or finitially created a set of Perl scripts he called 'Personal Home Page Tools' to maintain his personal homepage, including tasks such as displaying his resume and recording how much traffic his page was receiving 26 He rewrote these scripts as C programming language Common Gateway Interface (CGI) binaries, extending them to add the ability to work with web forms and to communicate with databases and called this implementation 'Personal Home Page/Forms Interpreter' or PHP/FI. PHP/FI could be used to build simple, dynamic web applications. Lerdorf released PHP/FI as 'Personal Home Page Tools (PHP Tools) version 1.0' publicly on June 8, 1995, to accelerate bug location and improve the code. This release already had the basic functionality that PHP has today. This included Perl-like variables, form handling, and the ability to embed HTML.

The syntax was similar to Perl but was more limited and simpler, although less consistent. A development team began to form and, after months of work and beta testing, officially released PHP/FI 2 in November 1997

ZeevSuraski and Andi Gutmans, two Israeli developers at the Technion IIT, rewrote the parser in 1997 and formed the base of PHP 3, changing the language's name to the recursive initialism PHP: Hypertext Preprocessor. Afterwards, public testing of PHP 3 began, and the official launch came in June 1998. Suraski and Gutmans then started a new rewrite of PHP's core, producing the Zend Engine in 1999. They also founded Zend Technologies in Ramat Gan, Israel

On May 22, 2000, PHP 4, powered by the Zend Engine 1.0, was released. As of August 2008 this branch is up to version 4.4.9. PHP 4 is no longer under development nor will any security updates be released.

On July 13, 2004, PHP 5 was released, powered by the new Zend Engine II. PHP 5 included new features such as improved support for object-oriented programming, the PHP Data Objects (PDO) extension (which defines a lightweight and consistent interface for accessing databases), and numerous performance enhancements. In 2008 PHP 5 became the only stable version under development. Late static binding had been missing from PHP and was added in version.

A new major version has been under development alongside PHP 5 for several years. This version was originally planned to be released as PHP 6 as a result of its significant changes, which included plans for full Unicode support. However, Unicode support took developers much 27 longer to implement than originally thought and the decision was made in March 2010 to move the project to a branch, with features still under development moved to trunk.

Changes in the new code include the removal of register_globals, magic quotes, and safe mode. The reason for the removals was that register_globals had given way to security holes, and the use of magic quotes had an unpredictable nature, and was best avoided. Instead, to escape characters, magic quotes may be replaced with the add slashes () function, or more appropriately an escape mechanism specific to the database vendor itself like mysql_real_escape_string () for MySQL. Functions that will be removed in future versions and have been deprecated in PHP 5.3 will produce a warning if used.

PHP currently does not have native support for Unicode or multi byte strings; Unicode support is under development for a future version of PHP and will allow strings as well as class, method, and function names to contain non-ASCII characters.

PHP interpreters are available on both 32-bit and 64-bit operating systems, but on Microsoft Windows the only official distribution is a 32-bit implementation, requiring Windows 32-bit compatibility mode while using Internet Information Services (IIS) on a 64-bit Windows platform. As of PHP 5.3.0, experimental 64-bit versions are available for MS Windows.

USAGE

PHP is a general-purpose scripting language that is especially suited to server-side web development where PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content. It can also be used for command-line scripting and client-side GUI applications.

PHP can be deployment most web servers, many operating systems and platforms, and can be used with many relational database management systems (RDBMS). It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use. PHP primarily acts as a filter, taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML. Since PHP 4, the PHP parser compiles input to produce byte code for processing by the Zend Engine, giving improved performance over its interpreter predecessor.

ABOUT MYSQL

MySQL is the world's most popular open-source database software, with over 100 million copies of its software downloaded or distributed throughout its history. With its superior speed, reliability, and ease of use, MySQL has become the preferred choice for Web, Web 2.0, SaaS, ISV, Telecom companies and forward-thinking corporate IT Managers because it eliminates the major problems associated with downtime, maintenance and administration for modern, online applications.

Many of the world's largest and fastest-growing organizations use MySQL to save time and money powering their high-volume Web sites, critical business systems, and packaged

software — including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, Wikipedia, and Booking.com.

The flagship MySQL offering is MySQL Enterprise, a comprehensive set of productiontested software, proactive monitoring tools, and premium support services available in an affordable annual subscription.

MySQL is a key part of LAMP (Linux, Apache, MySQL, PHP / Perl / Python), the fastgrowing open-source enterprise software stack. More and more companies are using LAMP as an alternative to expensive proprietary software stacks because of its lower cost and freedom from platform lock-in.

MySQL was originally founded and developed in Sweden by two Swedes and a Finn:David Axmark, Allan Larsson and Michael "Monty" Widenius, who had worked together since the 1980's. More historical information on MySQL is available on Wikipedia.

OUR CONTINUED MYSQL VALUES

We want the MySQL database to be:

• The best and the most-used database in the world for online applications

• Available and affordable for all

• Easy to use

• Continuously improved while remaining fast, secure and reliable

• Fun to use and improve

• Free from bugs

We want the people working on MySQL to:

• Subscribe to the Open-Source philosophy

• Aim to be good citizens

• Prefer partners that share our values and mindset

• Answer email and give assistance to users, customers, partners and co-workers

• Be a virtual organization, networking with others.

If you want to know more or have a request or proposal for us, please visit our contacts page to learn how to best reach us.

MySQL is a relational database management system (RDBMS)[1] that runs as a server providing multi-user access to a number of databases. MySQL is officially pronounced /maɪˌɛskjuːˈɛl/ ("My S-Q-L"),[2] but is often also pronounced /maɪˈsiːkwəl/ ("My Sequel"). It is named after developer Michael Widenius' daughter, My. The SQL phrase stands for Structured Query Language.

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

Free-software projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Some free software project examples: Joomla, WordPress, MyBB, phpBB, Drupal and other software built on the LAMP software stack. MySQL is also used in many high-profile, large-scale World Wide Web products, including Wikipedia, Google (though not for searches) and Face book.

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP web application software stack—LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python".

MySQL is used in some of the most frequently visited web sites on the Internet, including Flickr, Nokia.com, YouTube and as previously mentioned; Wikipedia,[10] Google [11] and Face book.

PLATFORMS AND INTERFACES

MySQL is written in C and C++. Its SQL parser is written in yacc, and a home brewedlexical analyzer named sql_lex.cc.

MySQL works on many different system platforms, including AIX, BSDi, FreeBSD,HPUX, eComStation, i5/OS, IRIX, Linux, Mac OS X, Microsoft Windows, NetBSD, NovellNetWare, OpenBSD, OpenSolaris, OS/2 Warp, QNX, Solaris, Symbian, SunOS,

SCOOpenServer, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists.

Many programming languages with language-specific APIs include libraries for accessing MySQL databases. These include MySQL Connector/Net for integration with Microsoft's Visual Studio (languages such as C# and VB are most commonly used) and the ODBC driver for Java. In addition, an ODBC interface called My ODBC allows additional programming languages that support the ODBC interface to communicate with a MySQL database, such as ASP or ColdFusion. The HTSQL - URL based query method also ships with a MySQL adapter, allowing direct interaction between a MySQL database and any web client via structured URLs. The MySQL server and official libraries are mostly implemented in ANSI C/ANSI C++.

MANAGEMENT AND GRAPHICAL FRONTENDS

MySQL Workbench in Windows, displaying the Home Screen which streamlines use of its full capabilities.

MySQL is primarily an RDBMS and therefore ships with no GUI tools to administer MySQL databases or manage data contained within. Users may use the included command linetools, or download MySQL frontends from various parties that have developed desktop software and web applications to manage MySQL databases, build database structure, and work with data records.

OFFICIAL

The official MySQL Workbench is a free integrated environment developed by MySQL AB, that enables users to graphically administer MySQL databases and visually design database structure. MySQL Workbench replaces the previous package of software, MySQL GUI Tools. Similar to other third-party packages, but still considered the authoritative MySQL frontend, MySQL Workbench lets users manage the following:

DATABASE DESIGN & MODELLING

SQL development – replacing MySQL Query Browser Database administration – replacing MySQL Administrator MySQL Workbench is available in two editions, the regular free and open source Community Edition which may be downloaded from the MySQL website,

and the proprietary Standard Edition which extends and improves the feature set of the Community Edition

THIRD-PARTY

Third-party proprietary and free graphical administration applications (or "frontends") are available that integrate with MySQL and enable users to work with database structure and data visually. Some well-known front ends, in alphabetical order, are:

Adminer – a free MySQL front end written in one PHP script, capable of managing multiple databases, with many CSS skins available. DB Edit – a free front end for MySQL and other databases.

Db Forge GUI Tools — a set of tools for database management that includes separate applications for schema comparison and synchronization, data comparison and synchronization, and building queries.

Distinguishing features

MySQL implements the following features, which some other RDBMS systems may not:

Multiple storage engines, allowing one to choose the one that is most effective for each table in the application (in MySQL 5.0, storage engines must be compiled in; in MySQL 5.1, storage engines can be dynamically loaded at run time):

Native storage engines (MyISAM, Falcon, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, Cluster, Berkeley DB, EXAMPLE, Maria, and InnoDB, which was made the default as of 5.5)

CHAPTER – VI

6. TESTING

6.1 TESTING PROCESS

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

TYPES OF TESTS

- Unit testing
- Integration testing
- Functional test
- System Test
- Validation Testing

UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

FUNCTIONAL TEST

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input	:	identified classes of valid input must be accepted.
Invalid Input	:	identified classes of invalid input must be rejected.
Functions	:	identified functions must be exercised.
Output	:	identified classes of application outputs must be exercised.
Systems/Procedures	:	interfacing systems or procedures must be invoked.

SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

VALIDATION TESTING

At the culmination of the black box testing software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of test i.e., validation succeeds when the software function in a manner that can be reasonably accepted by the customer.

TEST FOR LOGIN AND REGISTRATION MODULE

Test for login form this form is used for login of user and admin. In this we enter the username password and user type if all these are correct login page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for username and password.

Test for account creation this registration form is used for new account creation when user does not fill the registration form completely it asks again to fills the whole form when he/she fills the registration form fully it gets redirected to login page.

6.2 TEST DATA AND OUTPUT

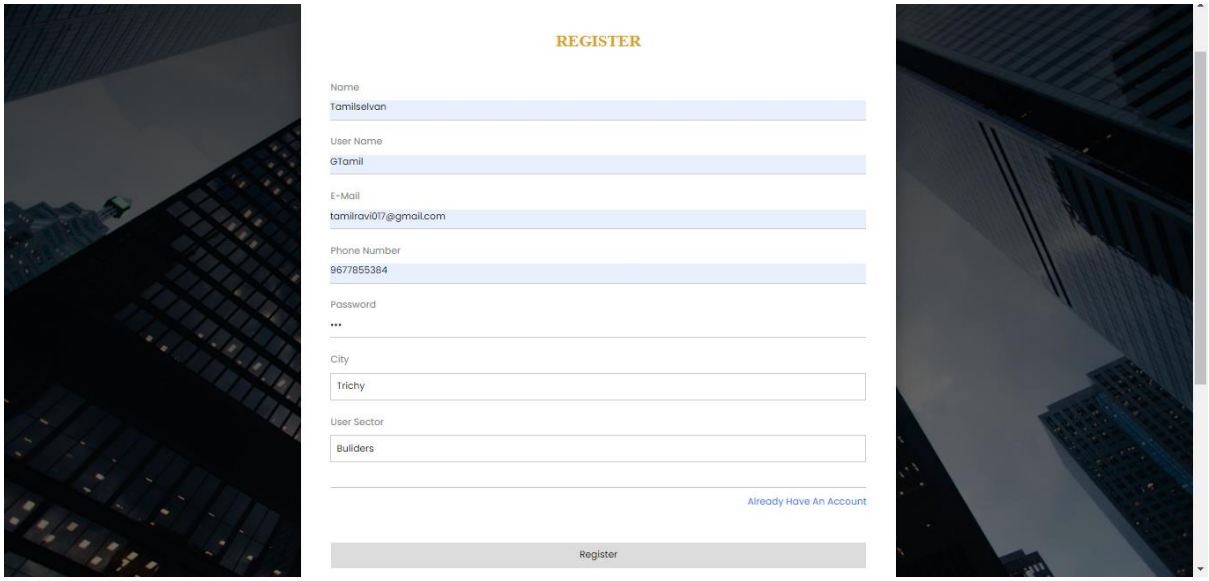


Fig 6.2.1 Registration Form

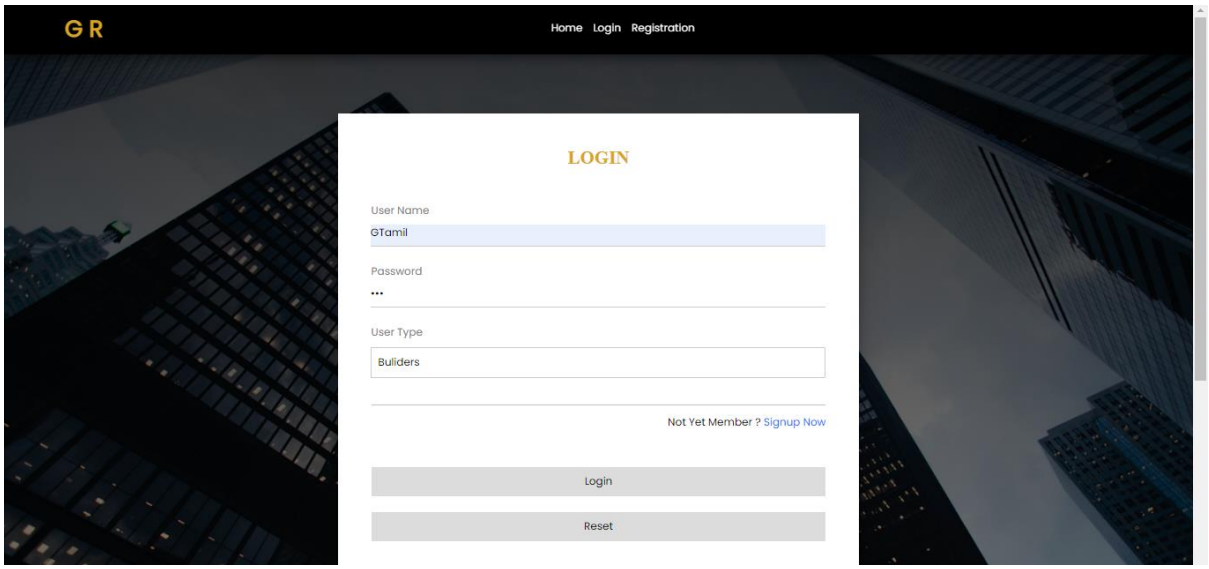


Fig 6.2.2 Login Form

CHAPTER – VII

7. USER MANUAL

7.1 HARDWARE REQUIRMENT

- System: Any 64bit Processors
- Hard disk: 500 GB
- Ram: 2 GB or above
- Keyboard type: Any System Keyboard
- Mouse: Any System Mouse

7.2 SOFTWARE REQUIREMENTS

- Operating system: window 7 or above
- Coding language: html, css, JavaScript, php
- Data base: MySQL

7.3 INSTALATION PROCEDURE

Step 1: Download the files. Download the latest **PHP**5 ZIP package from www.**php**.net/downloads.**php.**

Step 2: Extract the files.

Step 3: Configure **php.**

Step 4: Add C:**php** to the path environment variable.

Step 5: Configure **PHP** as an Apache module.

Step 6: Test a **PHP** file

7.4 SAMPLE I/O

HOMEPAGE

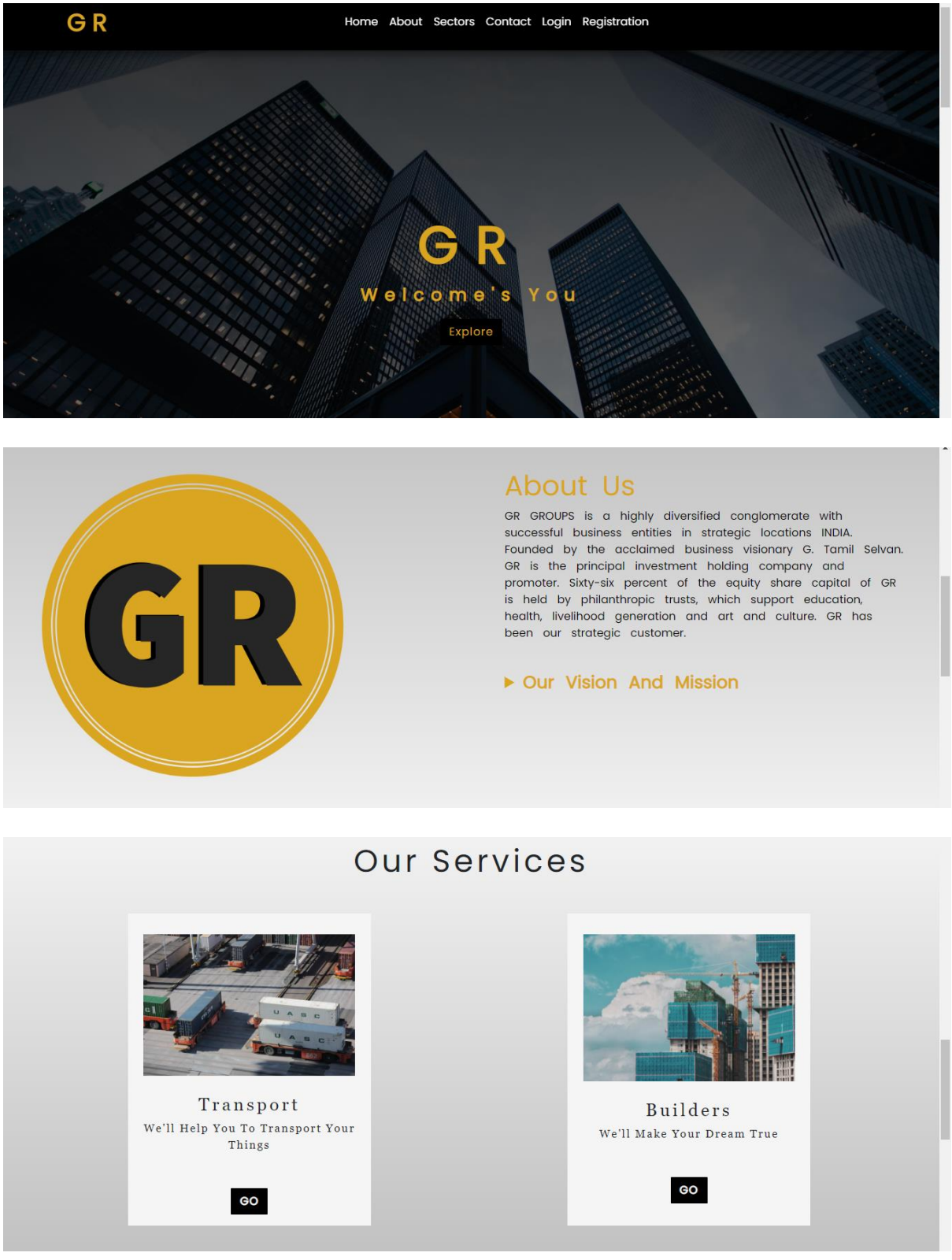


Fig 7.4.1 Home page

USER LOGIN PAGE

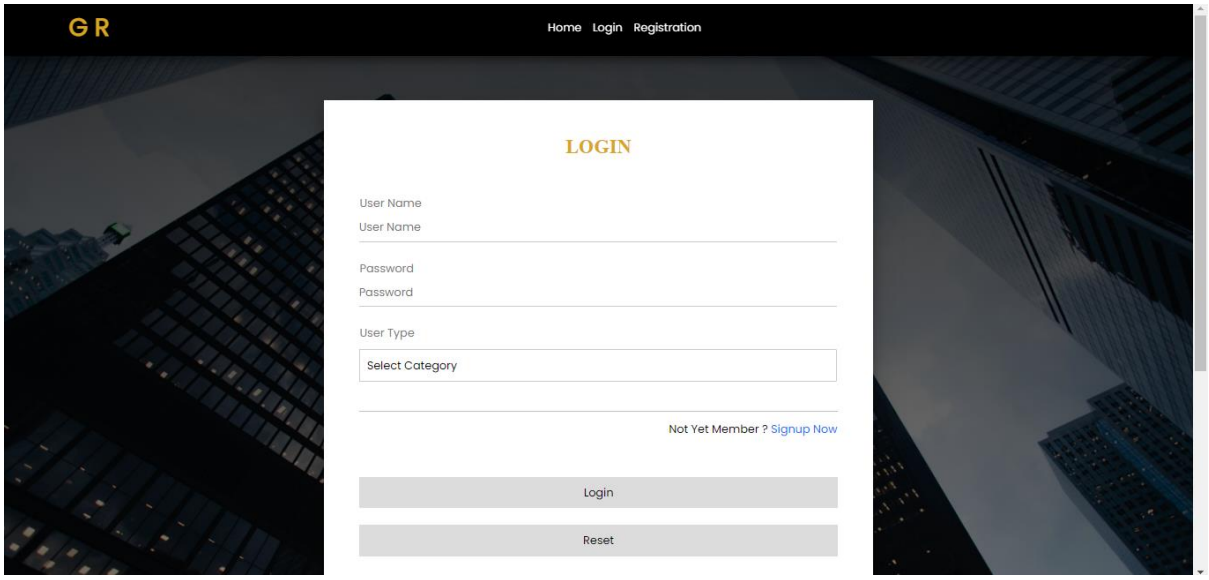


Fig 7.4.2 User Login

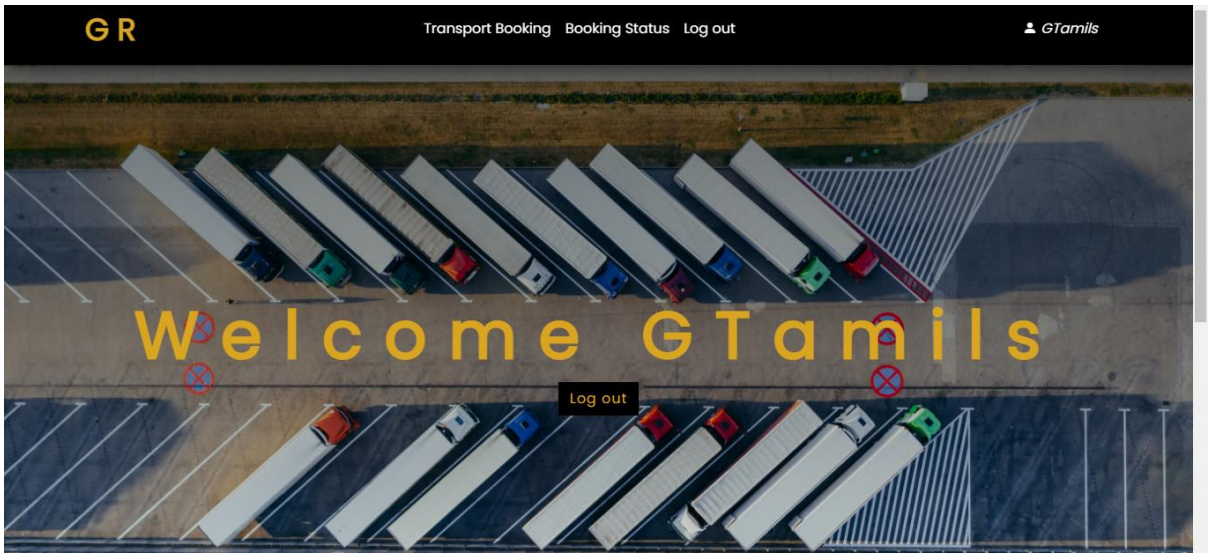


Fig 7.4.3 Transport User Home Page

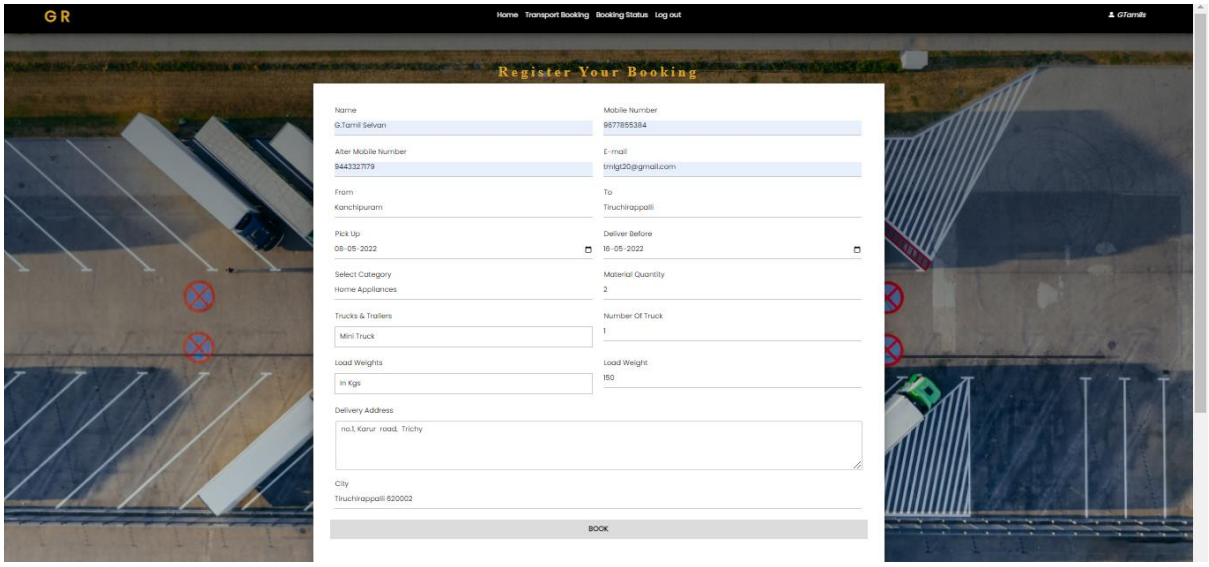


Fig 7.4.4 Transport Booking

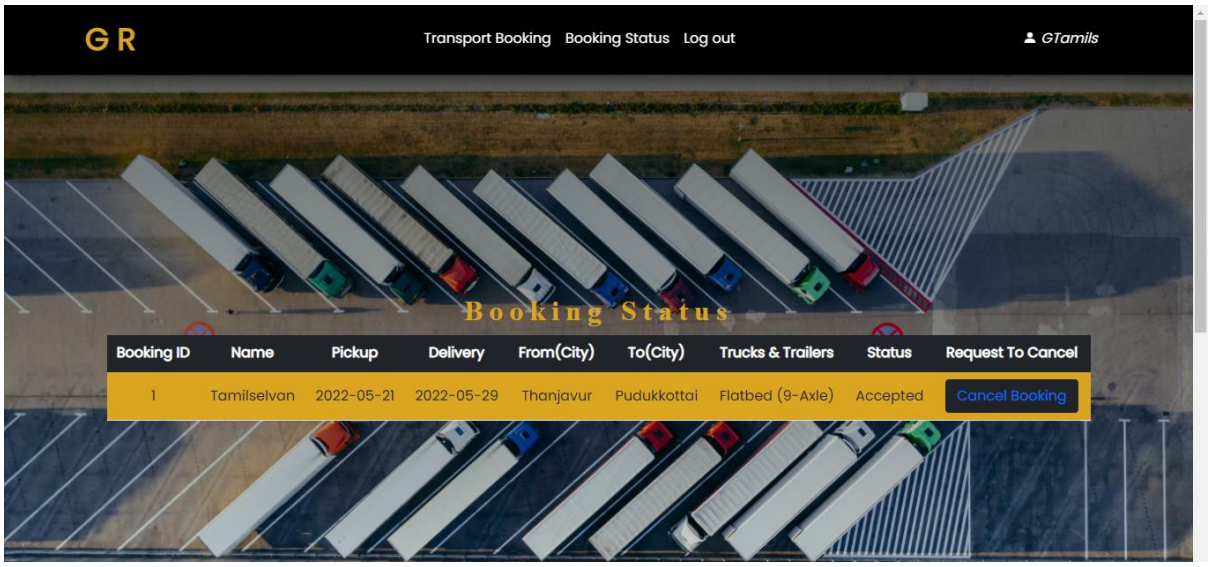


Fig 7.4.5 Transport Booking Status

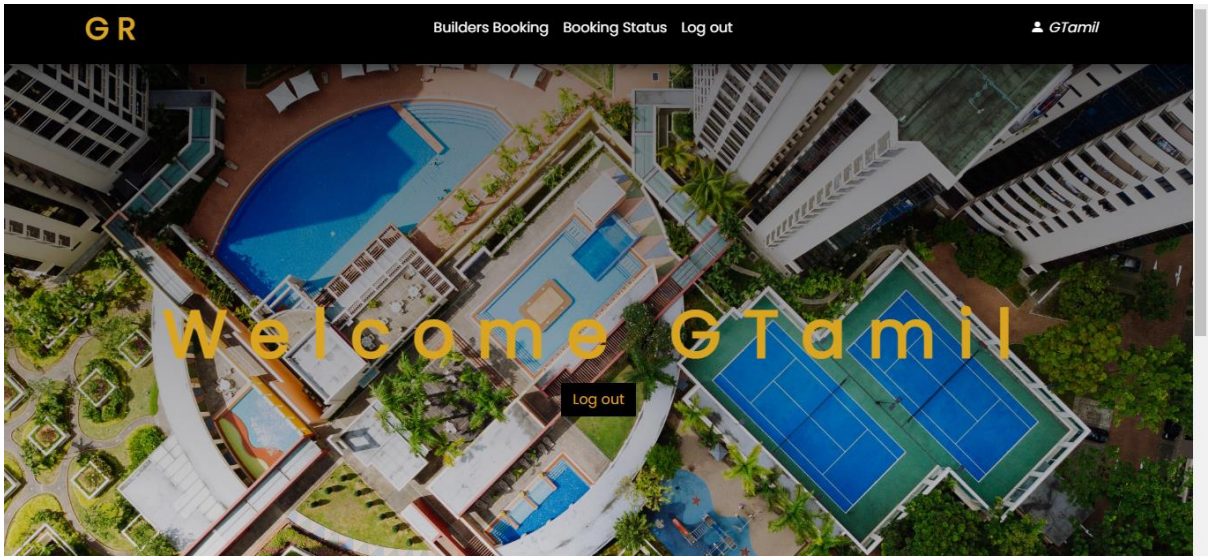


Fig 7.4.6 Residential User Home Page

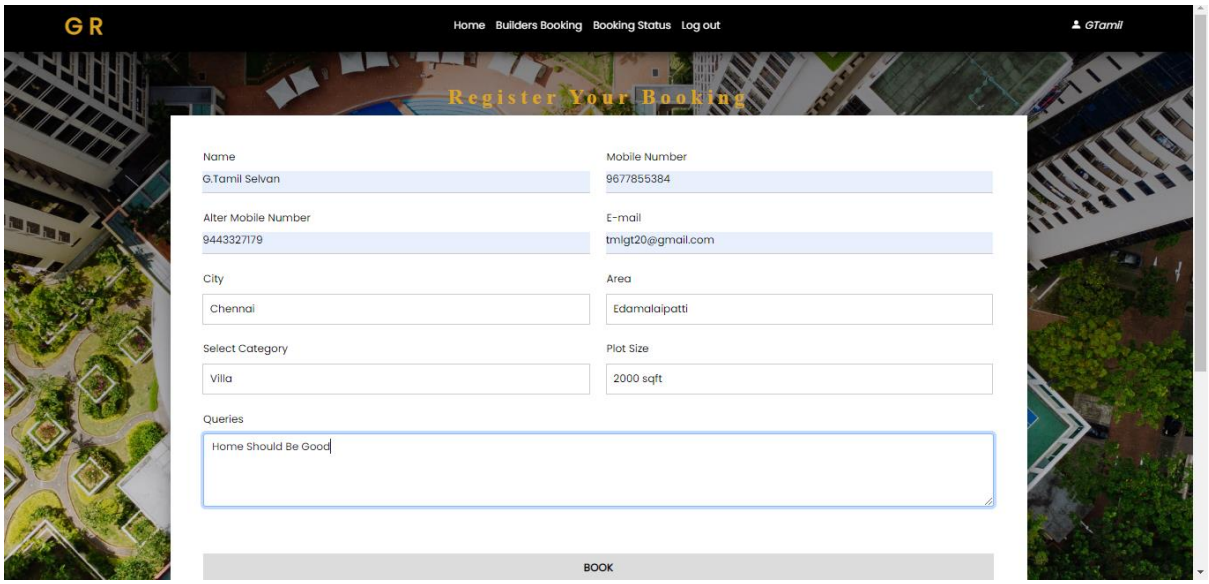


Fig 7.4.7 Residential Booking

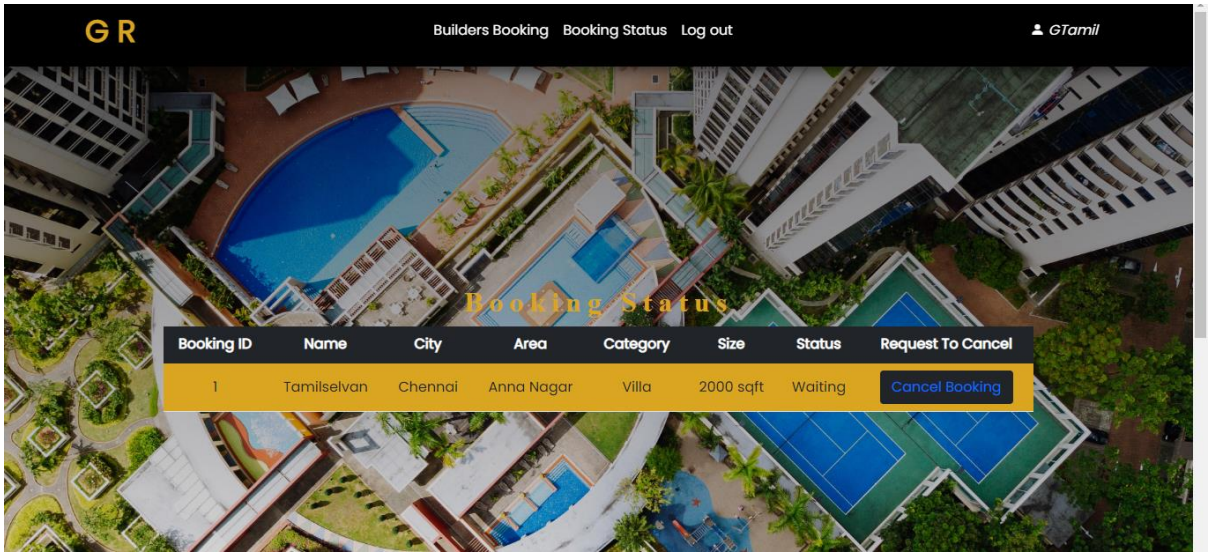


Fig 7.4.8 Residential Booking Status

ADMIN LOGIN PAGE

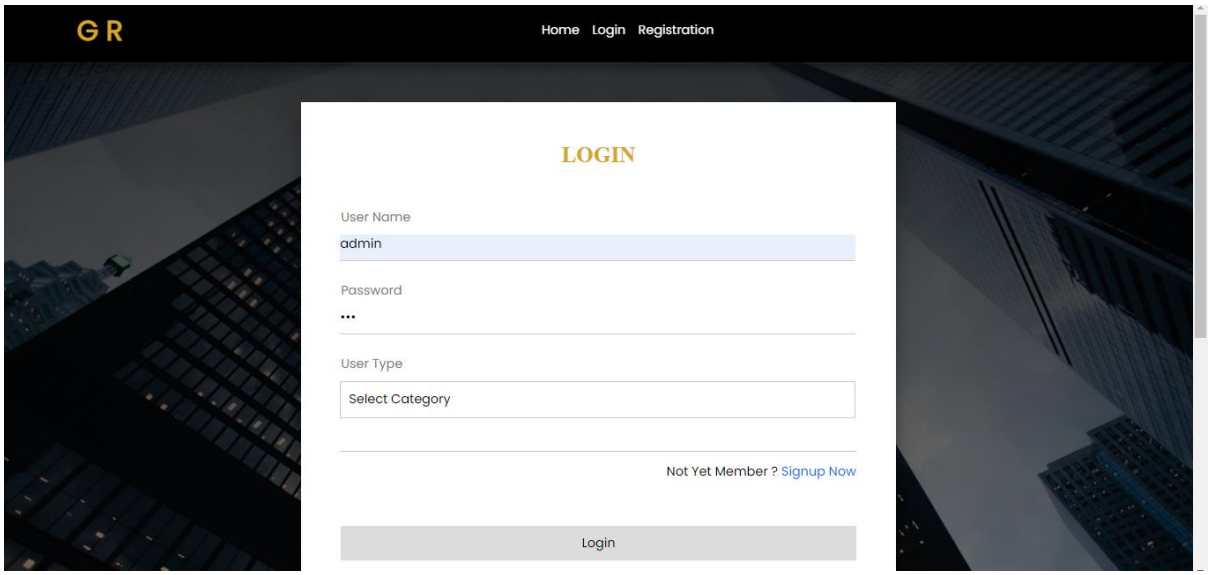


Fig 7.4.9 Admin Login

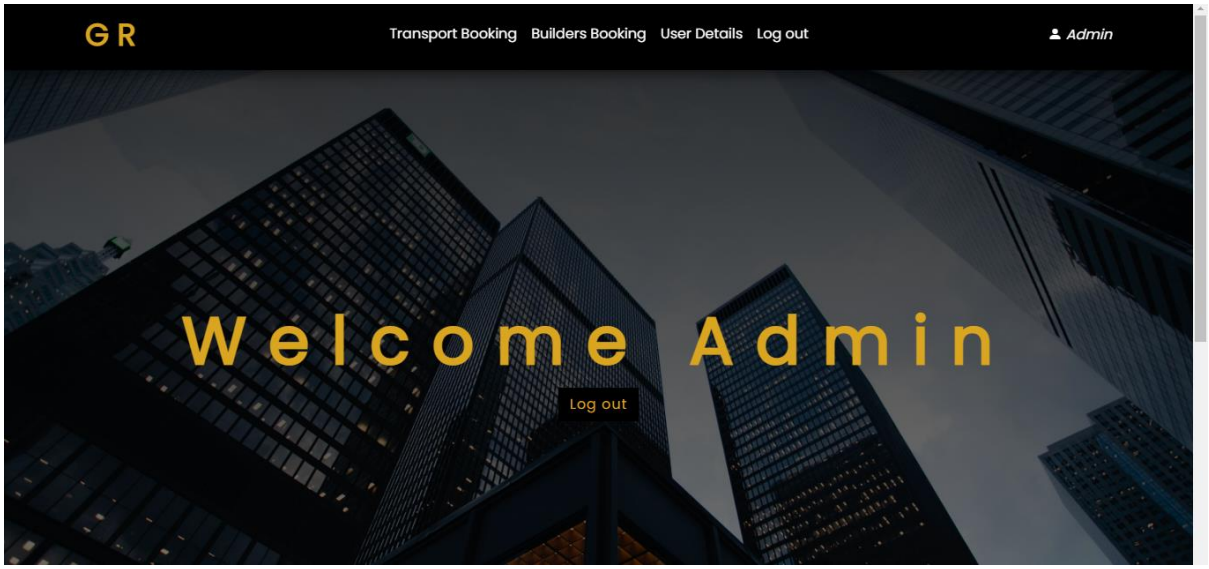


Fig 7.4.10 Admin Home Page

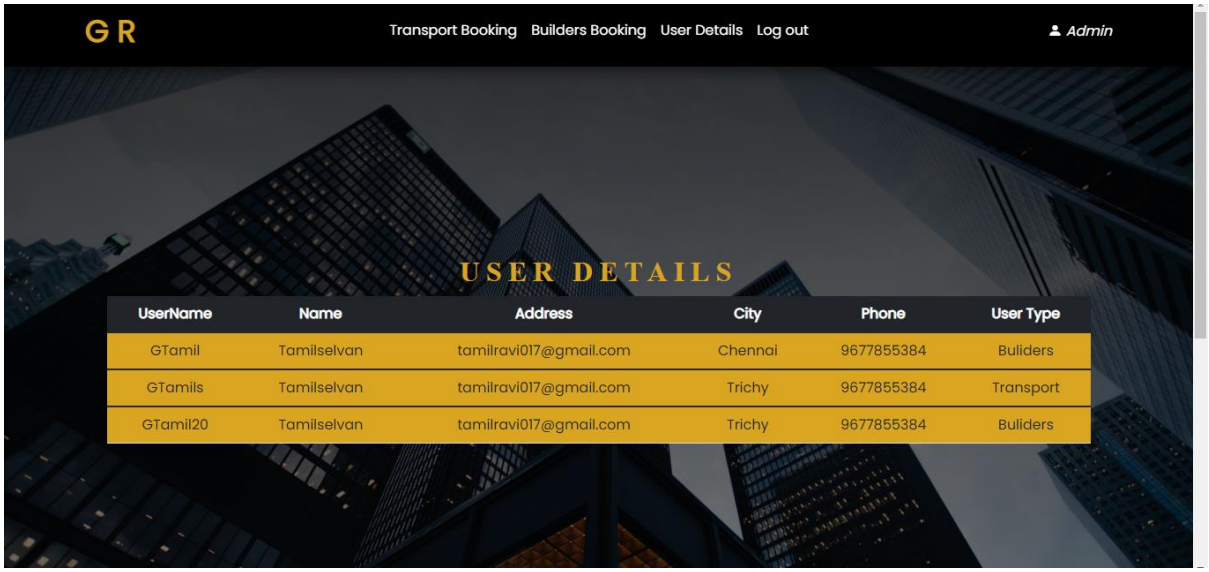


Fig 7.4.11 User Details

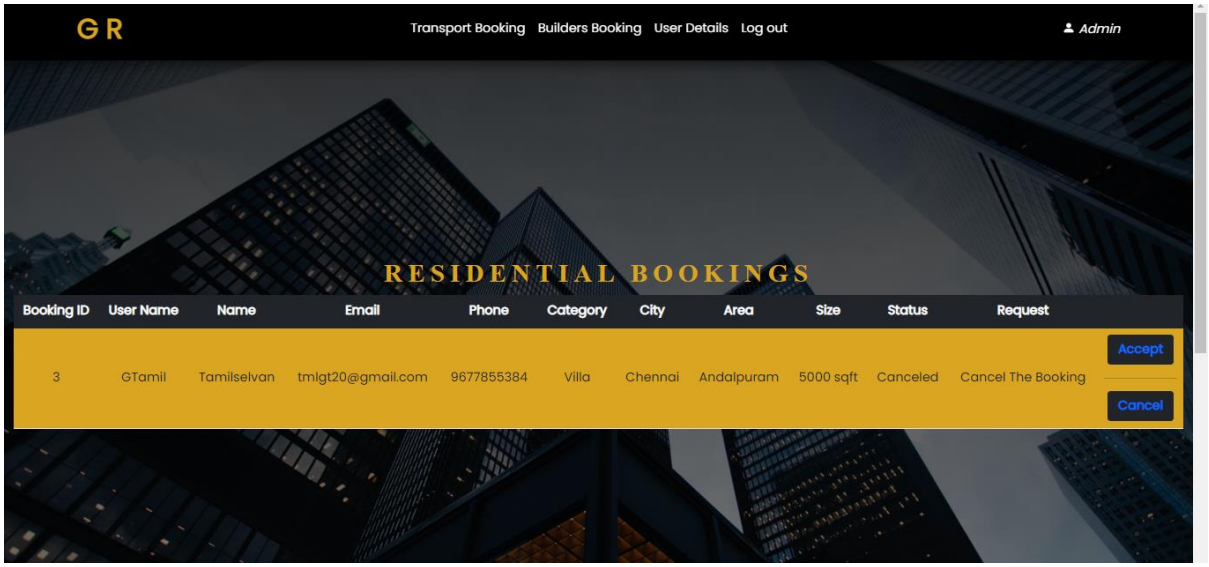


Fig 7.5.12 Residential Booking Report

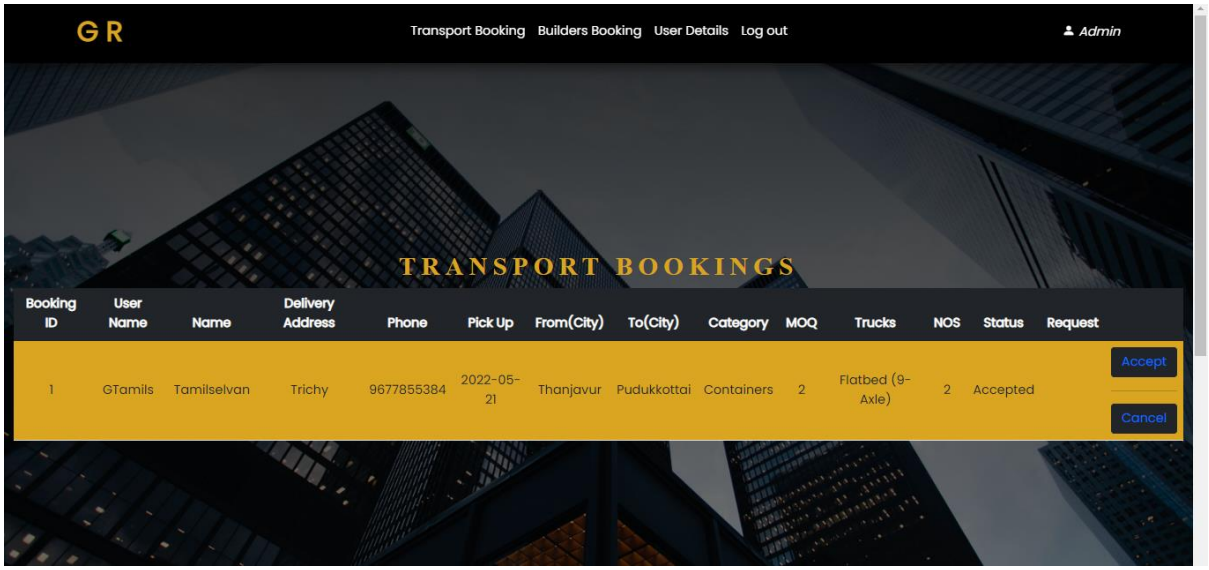
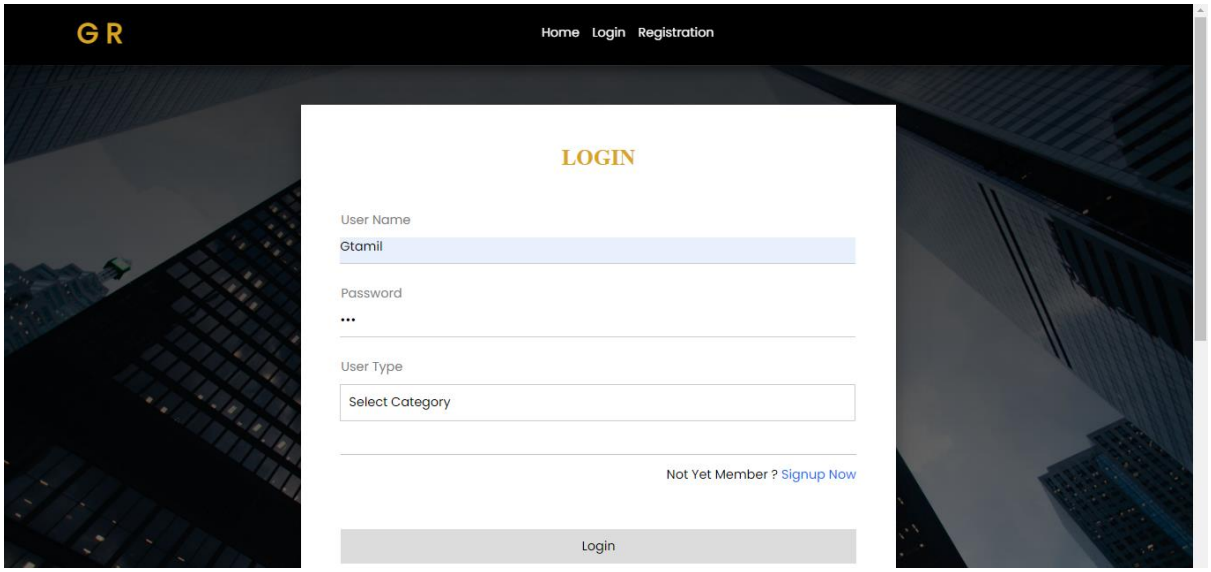
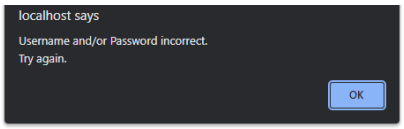


Fig 7.4.13 Transport Booking Report

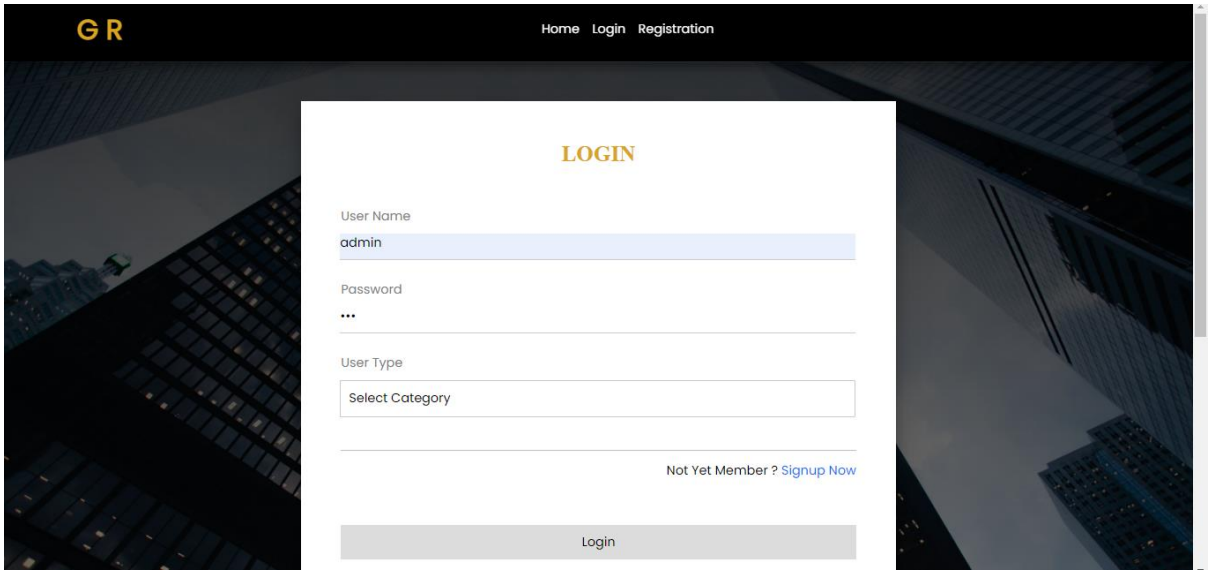
7.5 ERROR MESSAGES

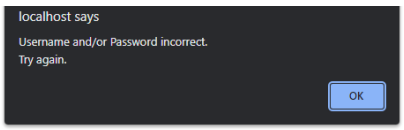
User Login





Admin Login


G R
Home Login Registration
LOGIN
User Name
admin
Password
User Type
Select Category
Not Yet Member ? Signup Now
Login


localhost says
Username and/or Password incorrect.
Try again.
OK

CHAPTER – VIII

8. CONCLUSION

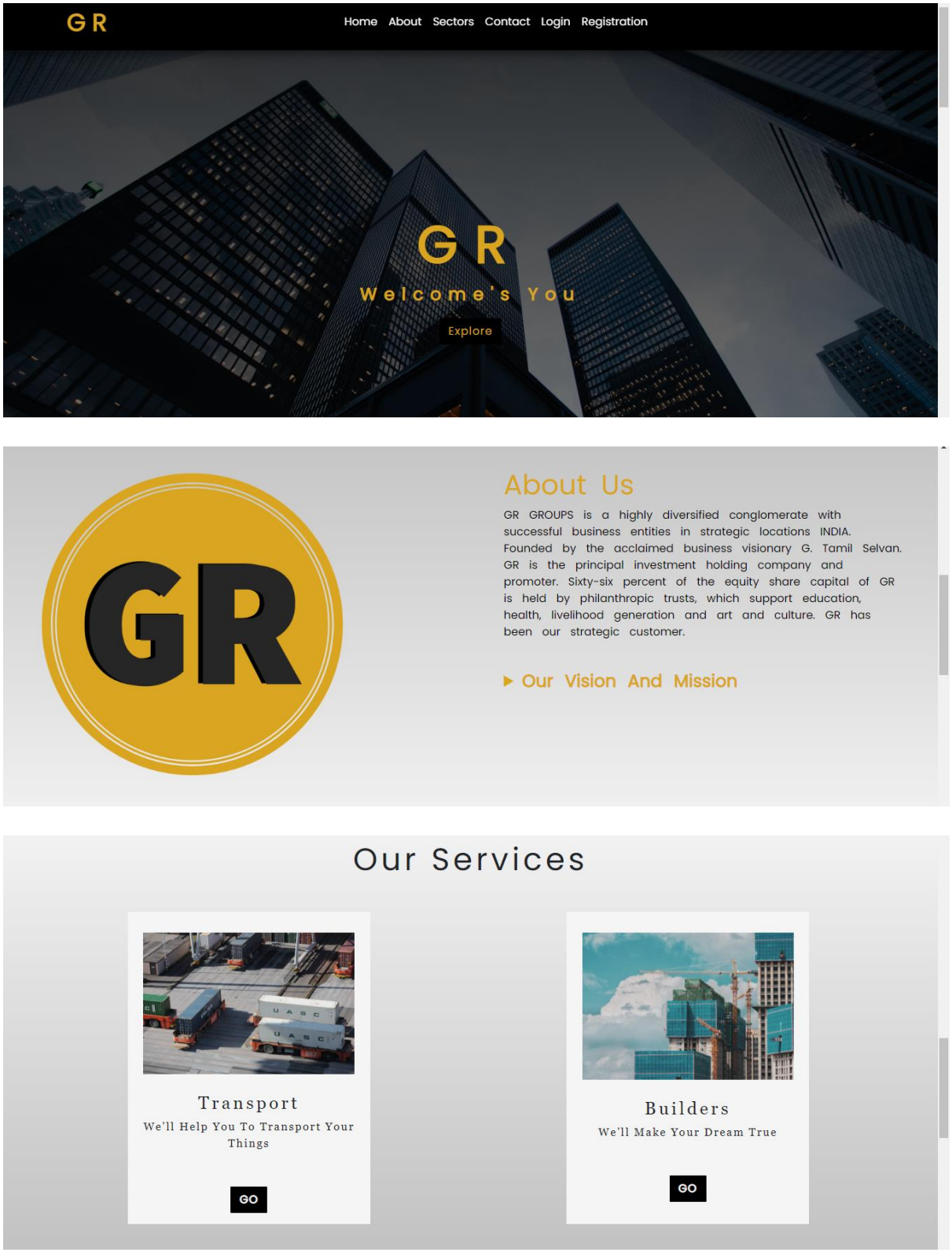
8.1 SUMMARY OF THE PROJECT

The Complete Industrial Groups of Management was successfully completed and then tested. Almost all the objectives of the project were meet and there by a trial run of the project giving sample data gives good result. This system has been developed to reduce the Problem for Industrial Groups of Management. This system can adopt any changes in the future also. Thus, this software is developed successfully to fulfil the objectives and satisfies the requirements of the user.

8.2 FUTURE ENHANCEMENT

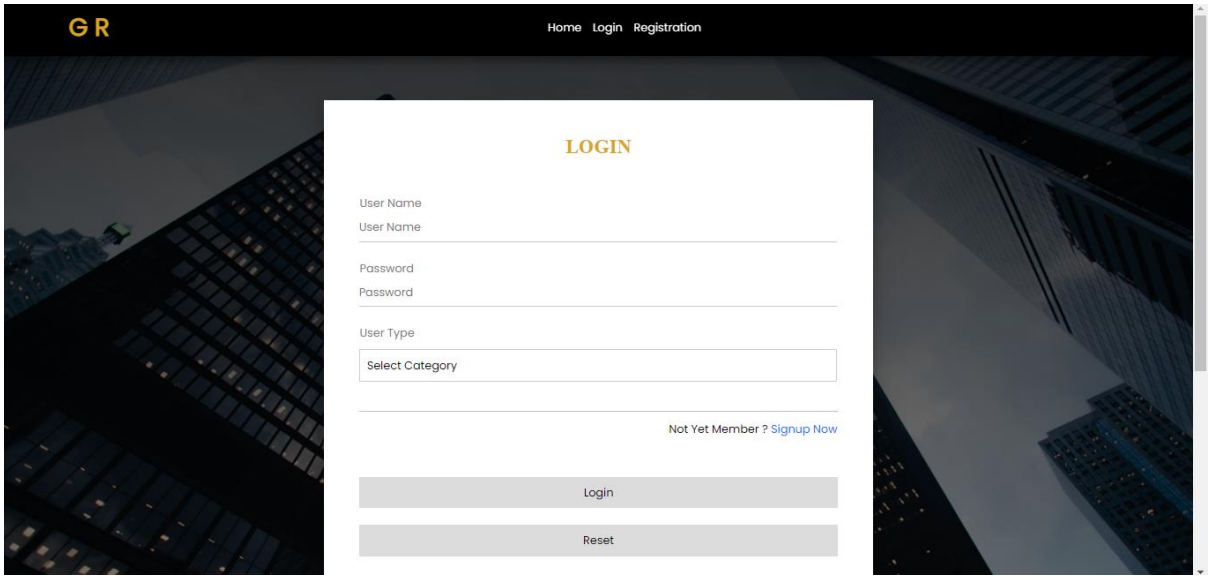
In this project, notifications are sent to the user about the booking to user's dashboard. In future, the notification will be sent to the user registered email directly.

HOMEPAGE

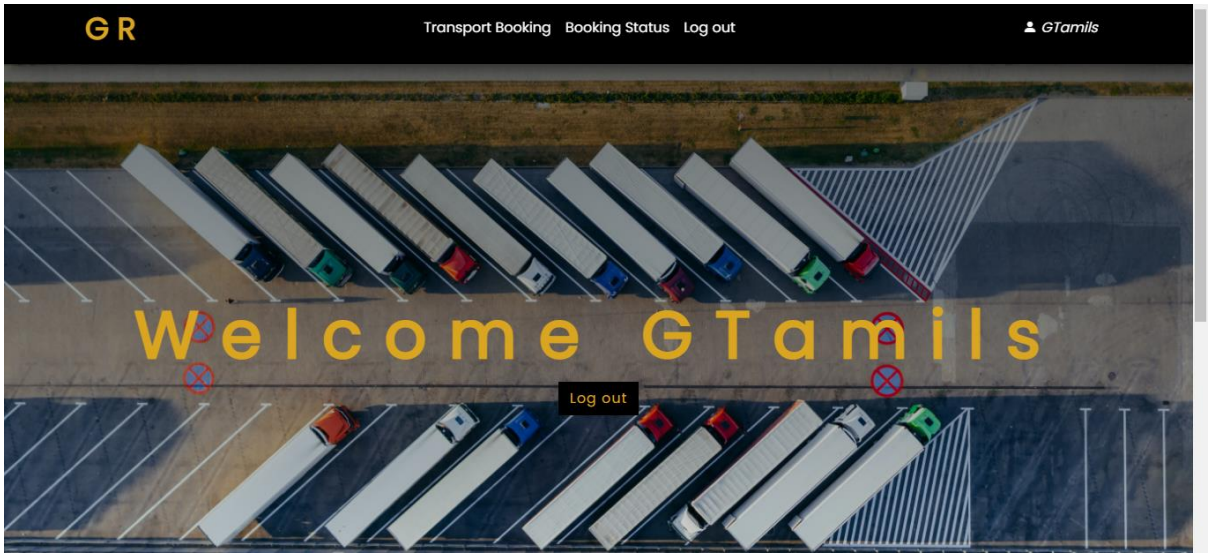


Home page

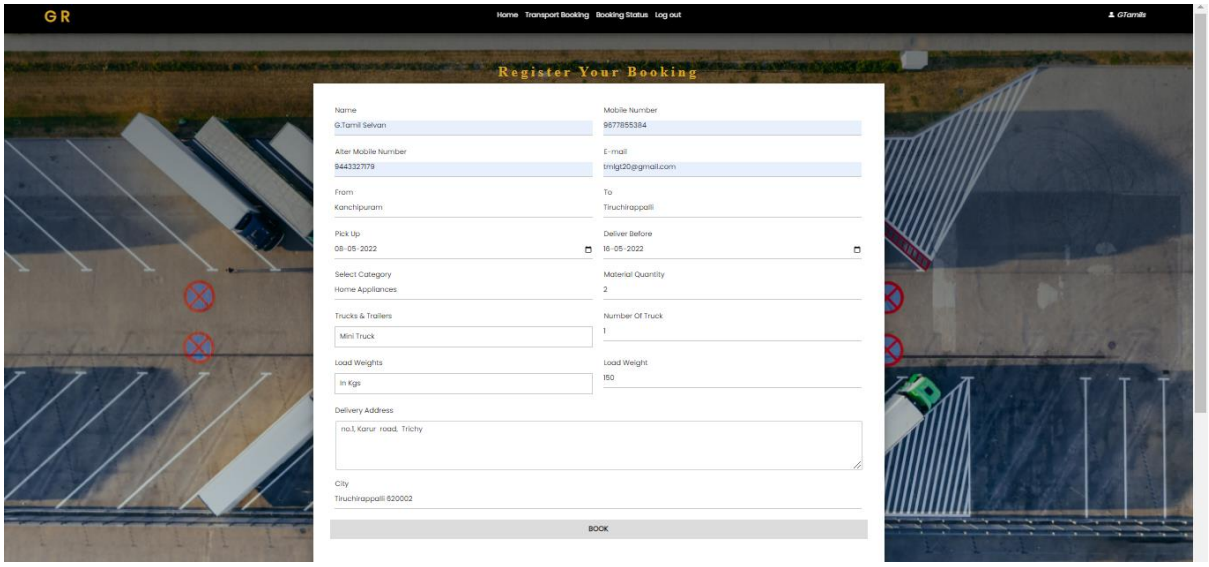
USER LOGIN PAGE



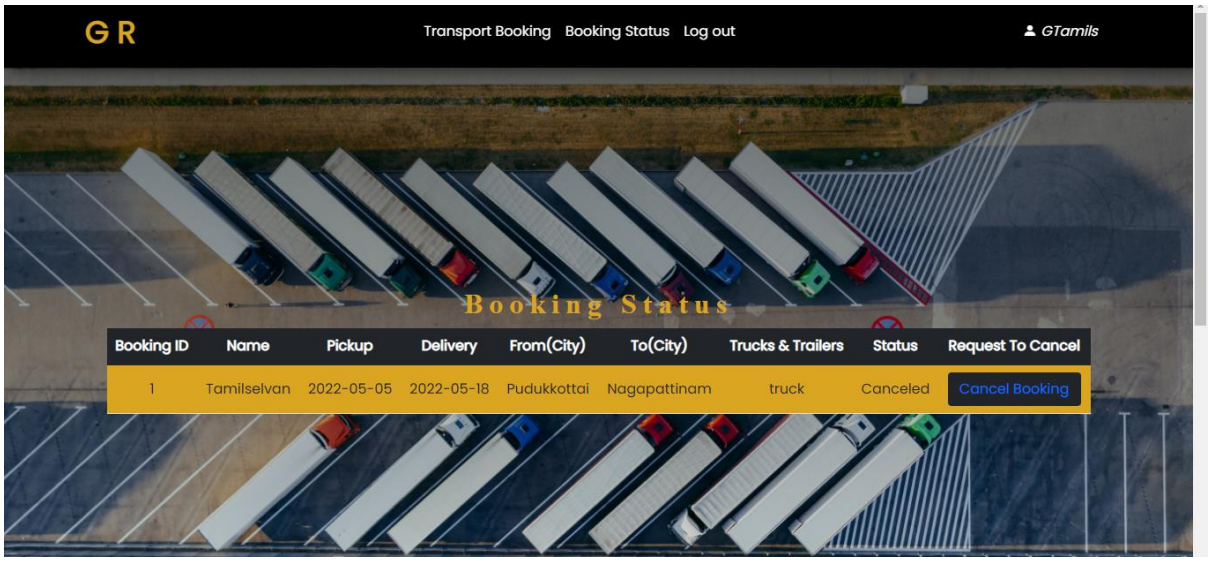
User Login



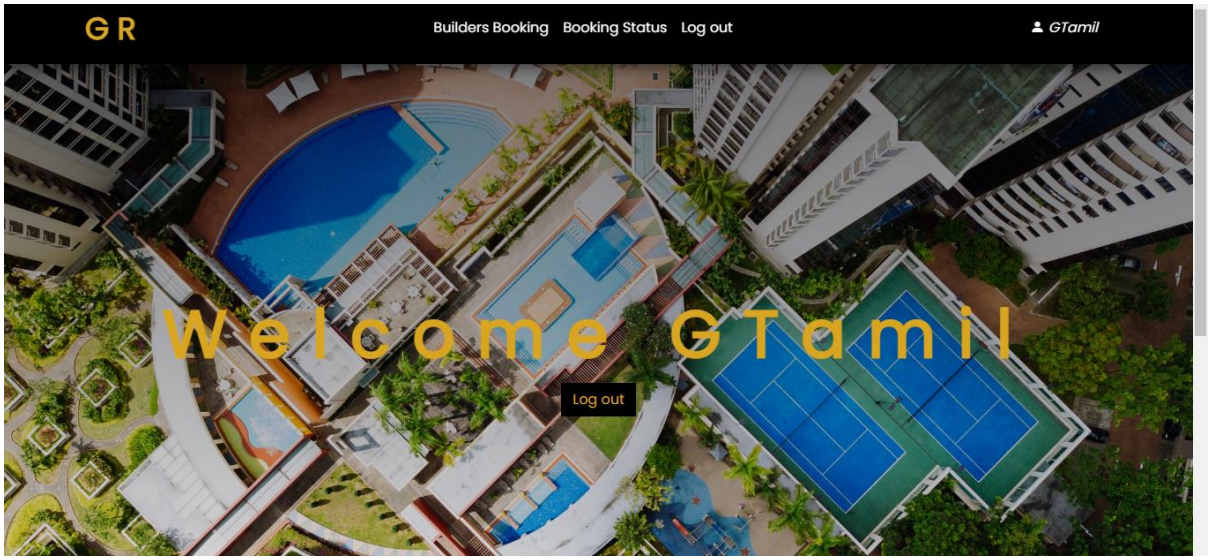
Transport User Home Page



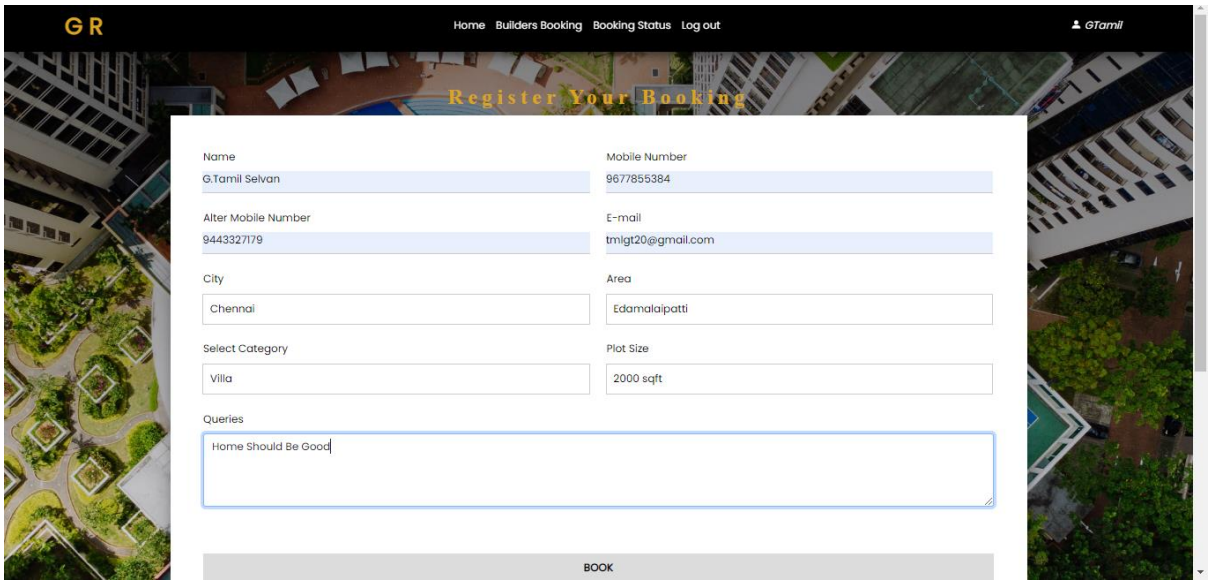
Transport Booking



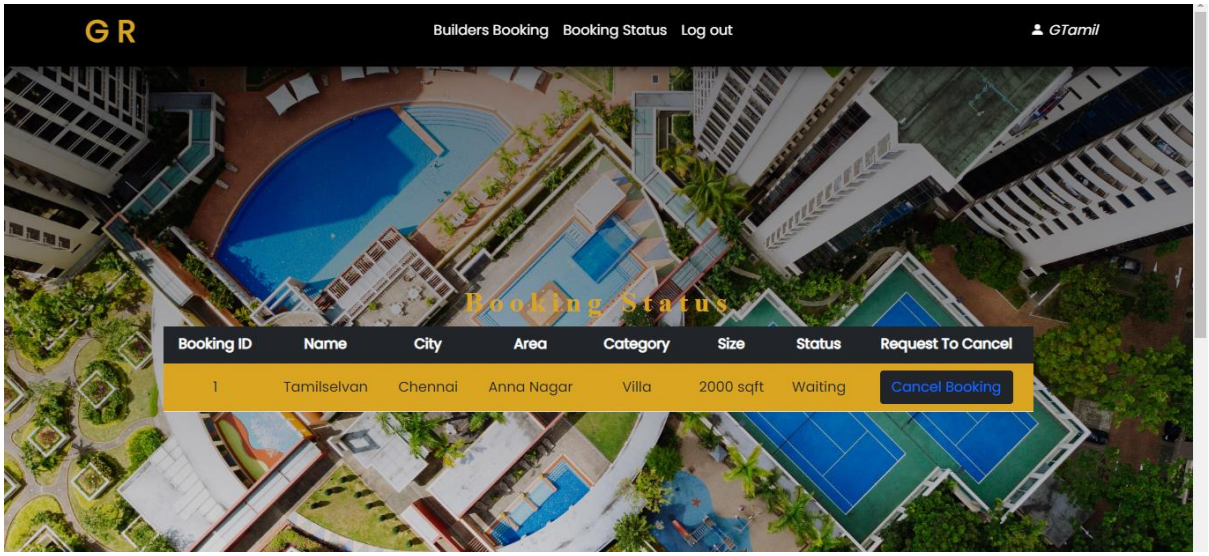
Transport Booking Status



Residential User Home Page

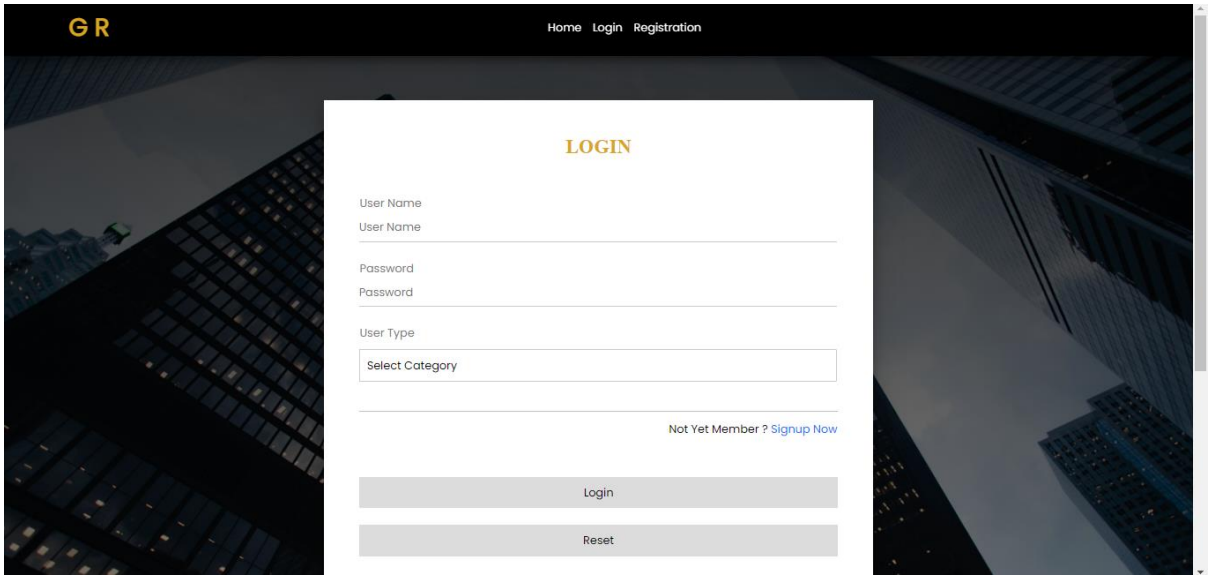


Residential Booking

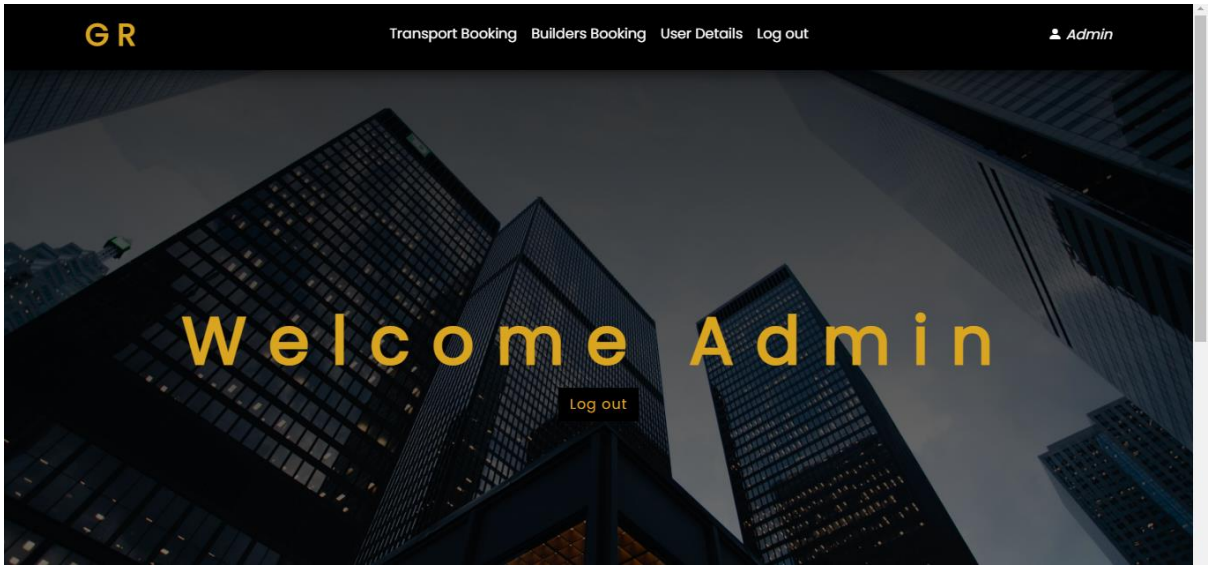


Residential Booking Status

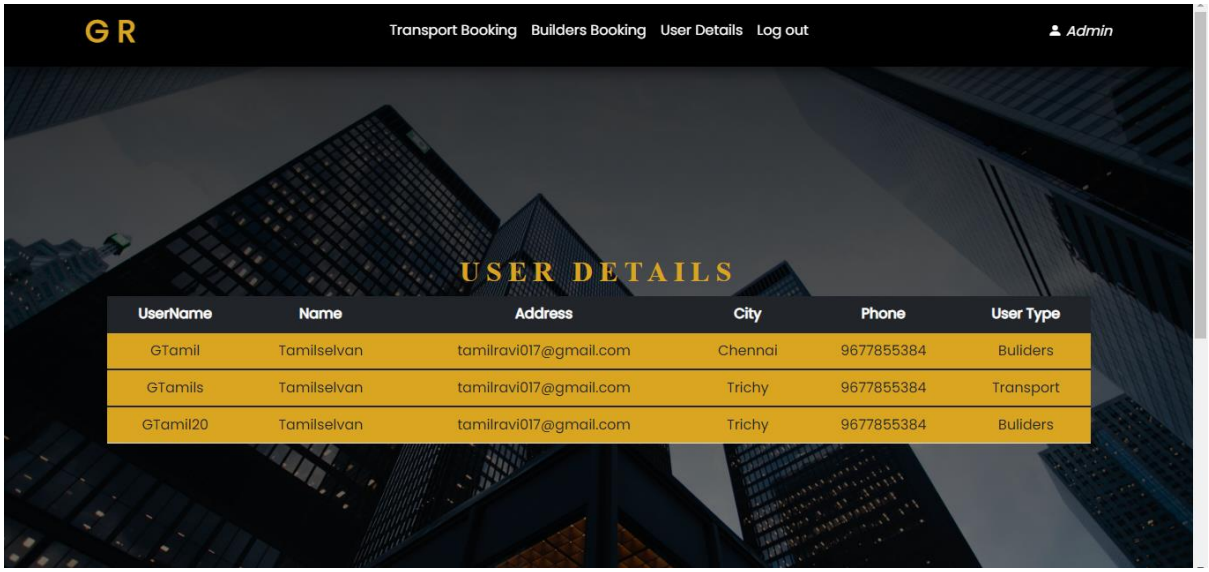
ADMIN LOGIN PAGE



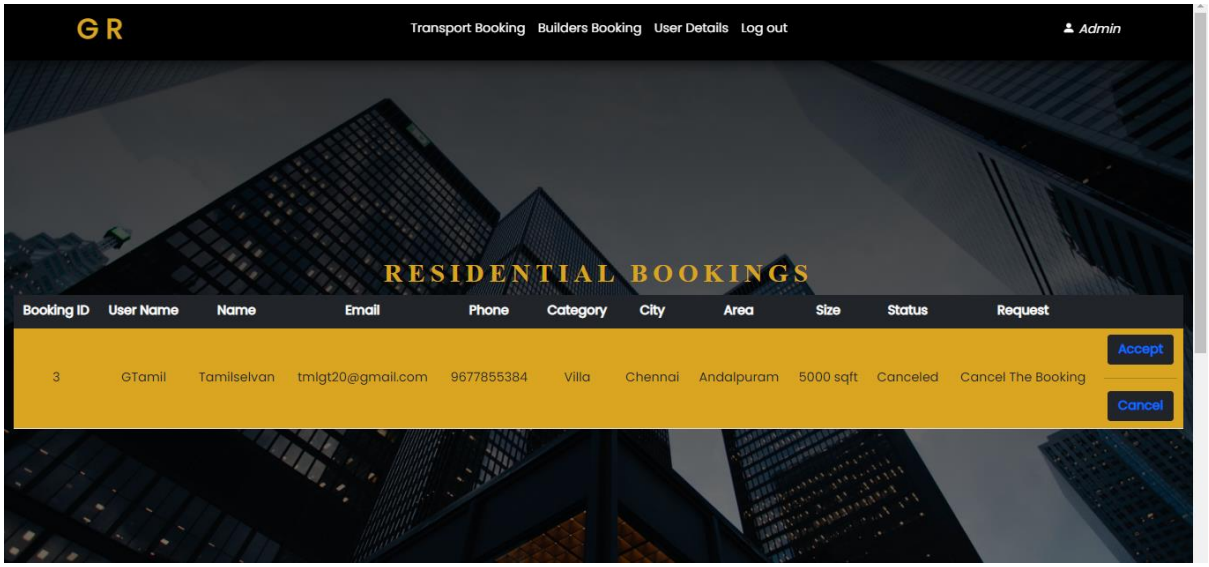
Admin Login



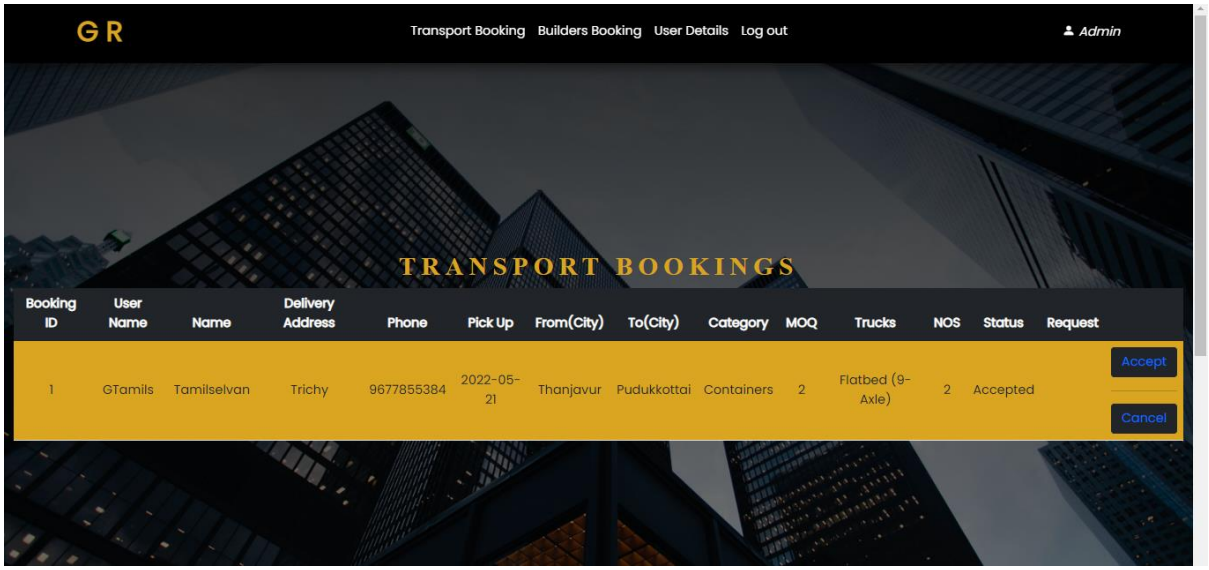
Admin Home Page



User Details



Residential Booking Report



Transport Booking Report