A CS814 Course Project Report on

CAR RENTAL SYSTEM

Submitted by,

Mayur Khadse (202IS014)
Dange Sadik Pashamiyan (202CS005)

Under Guidance:
Mahendra Pratap Singh

Department of Computer Science and Engineering
National Institute of Technology Karnataka
P.O. Srinivasnagar, Surathkal, Mangalore-575 025
Karnataka, India
January 2021

TABLE OF CONTENTS

ABSTRACT

This Car Rental System project will enable the user to rent a vehicle. The user shall login to the system and check for availability of cars. The user specifies a type of car and the journey date and time. The Car Rental System shall check for the availability of the car and rent the car to the customer. The user can make payment online. The tool is designed using VB.net. All the data regarding the rental cars are stored in MySQL database. The user has to enter his name, address, phone details and check for the cars available for rent. The UI is very simple and the connectivity to back end is robust. The main advantage is that the user shall be able to choose a car depending on his budget And it is based on Role-Based Access Control (RBAC).The essence of Role-Based Access Control (RBAC) is that system permissions are assigned to defined "roles" rather than to individual users. Users acquire these permissions by virtue of being authorised to act in a categorised manner known as a "role". The driving motivation for RBAC is to simplify security policy administration while facilitating the definition of flexible, customised policies. Basic RBAC models have been successfully applied since the mainframe era, but emerging networked systems, which have greater numbers of users, roles, and program components, challenge the expressive power of these classical RBAC models.

INTRODUCTION

We developed this project to book a car on rent at the fare charges. In the present system all booking work is done manually and it takes very hard work to maintain the information of booking and cars. If you want to find which vehicle is available for booking then it takes a lot of time. It only makes the process more difficult and hard. The aim of the project is to automate the work performed in the car rental management system like generating daily bookings, records of car or cab available for booking, record of routes available, rental charges for cars for every route, store record of the customer.

Car rental management system is a car booking software that provides a complete solution to all your day-to-day car booking office running needs. This system helps you to keep the information of customers online. You can check your customer information any time by using this system. Car rental management system is a unique and innovative product. Using this you can also keep the information of the number of bookings. This helps you to track your business . Based on this information you can make decisions regarding your business development.

In order to facilitate users to rent a car as and when required we would like to design a car rental system. It is more feasible than owning a car and maintaining it. Car rental system serves people who don't own a car, who are out of town or owners of damaged cars who are awaiting repair or insurance compensation.

Scope:
1. Providing a car catalog to the Users so that they can choose the best option based on their concern.
2. Admin can manage the catalog by adding or removing the cars based on their availability and allows only authorized Users (Driver License) to rent a vehicle
3. Admin must make sure that the rented cars must have valid insurance.
4. Admin must allow the user to provide feedback at the end of every ride.

Objective:
1. To reduce the effort of booking a car in a conventional procedure.
2. To ease the search process of a customer who is in need of a car.
3. To provide services to the customers in order to achieve the best customer satisfaction.

Types of User :
1. Admin
2. Registered User
3. Guest

1. Admin : Admin is the user with more permissions of the website who can easily handle everything on the website. Admin as well as Registerd user can log in through the same login page.

> ❖ *Admin Features*
>> ➢ Admin can create vehicle brands
>> ➢ Manage Vehicle Brands(Edit, Delete)
>> ➢ Post Vehicle
>> ➢ Manage vehicle(Edit,Delete)
>> ➢ Manage Booking(Admin can confirm and Cancel Booking)
>> ➢ Manage Testimonials (Active and Inactive)
>> ➢ Manage Contact us Query
>> ➢ Admin Can see the details of registered users
>> ➢ Admin can also update the page contents
>> ➢ Admin can update the contact us details
>> ➢ Manage Subscribers
>> ➢ Admin Dashboard(Admin can view the count of reg users, total booking, total subscribers, total queries, etc)
>> ➢ Change Password(admin can change own password)
>> ➢ Logout

2. Registered User : Anyone having mail id and mobile number can register through the registration link available on the home page. After completion of successful registration through the registration process users can log in with valid email ID and password provided at the time of registration. Users can recover their own password if they forgot by providing some registered info.

> ❖ *Registered User Features*
>> ➢ Car Booking
>> ➢ View Car booking history
>> ➢ Update His/Her profile
>> ➢ Update his/her password
>> ➢ Post Testimonials
>> ➢ View Testimonials
>> ➢ Logout

3. Guest User : Guest users can only view the website and checkout the information about rental cars but they can't book it. Guest users can also inquire through the contact us page available on site.

This system gives following features to the customers :

Registration : in this system users have to become a registered user to rent a car. Users have to fill up his contact information like his name, mobile number, etc. in the registration form. After submitting the form he will be able to login the system.

Date Scheduling : Use can select the date of car booking. To book a car on a selected date he has to pick up place information and drop off place information.

Our car rental management system provides a total solution to the field of auto rental industry. The idea behind the proposed website is to develop a system that allows customers to rent any vehicle as per their needs on the selected date for required time and in any of the selected cities.

This service will work 24x7. This system works online so allow anyone to book or rent a vehicle in a city without any time restriction. This system allows the customer to board the vehicle at his choice of location, pay for the rental offline and could provide valuable feedback for improvement in services. With the help of a car rental management system you can maximize the revenue by increasing your reach to customers. The user can also keep the details of the car route. It helps him to make the decision of hiring a car for their journey. By using it your customers will be able to get estimated quotes, see estimated fare summary and make Car online reservations from your website, with just a few clicks. This system also keeps the information of cancellation and modification of booking in current month. This facility takes the admin control over booking of the car as he can cancel the car booking if the car is not available or there is some issue in booking a particular car booking. This system also helps the administrator to check the business of the company.

Requirements :

Language Used	PHP5.6, PHP7.x
Database	MySQL
User Interface Design	HTML, AJAX,JQUERY,JAVASCRIPT

Web Browser	Mozilla, Google Chrome, IE8, OPERA
Software	XAMPP / Wamp / Mamp/ Lamp (anyone)

Flow Structure of Website:
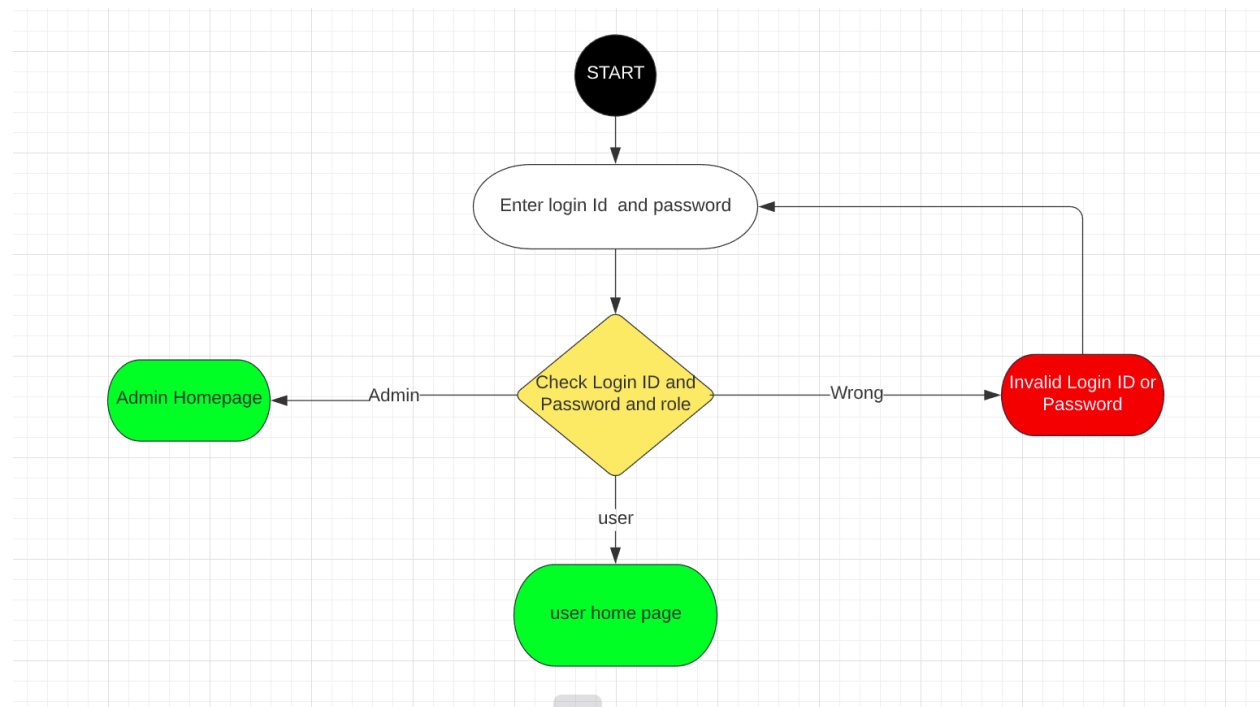
Entity Relation Diagram :

carrental tblsubscribers
- id : int(11)
- SubscriberEmail : varchar(120)
- PostingDate : timestamp

carrental tblcontactusinfo
- id : int(11)
- Address : tinytext
- EmailId : varchar(255)
- ContactNo : char(11)

carrental tbltestimonial
- id : int(11)
- UserEmail : varchar(100)
- Testimonial : mediumtext
- PostingDate : timestamp
- status : int(11)

carrental tblcontactusquery
- id : int(11)
- name : varchar(100)
- EmailId : varchar(120)
- ContactNumber : char(11)
- Message : longtext
- PostingDate : timestamp
- status : int(11)

carrental tblvehicles
- id : int(11)
- VehiclesTitle : varchar(150)
- VehiclesBrand : int(11)
- VehiclesOverview : longtext
- PricePerDay : int(11)
- FuelType : varchar(100)
- ModelYear : int(6)
- SeatingCapacity : int(11)
- Vimage1 : varchar(120)
- Vimage2 : varchar(120)
- Vimage3 : varchar(120)
- Vimage4 : varchar(120)
- Vimage5 : varchar(120)
- AirConditioner : int(11)
- PowerDoorLocks : int(11)
- AntiLockBrakingSystem : int(11)
- BrakeAssist : int(11)
- PowerSteering : int(11)
- DriverAirbag : int(11)
- PassengerAirbag : int(11)
- PowerWindows : int(11)
- CDPlayer : int(11)
- CentralLocking : int(11)
- CrashSensor : int(11)
- LeatherSeats : int(11)
- RegDate : timestamp
- UpdationDate : timestamp

carrental tblbooking
- id : int(11)
- BookingNumber : bigint(12)
- userEmail : varchar(100)
- VehicleId : int(11)
- FromDate : varchar(20)
- ToDate : varchar(20)
- message : varchar(255)
- Status : int(11)
- PostingDate : timestamp
- LastUpdationDate : timestamp

carrental Permissions
- id : int(11)
- role_id : int(11)
- permissions : varchar(100)

carrental tblrole
- id : int(11)
- role : varchar(100)

carrental tblbrands
- id : int(11)
- BrandName : varchar(120)
- CreationDate : timestamp
- UpdationDate : timestamp

carrental tblusers
- id : int(11)
- FullName : varchar(120)
- EmailId : varchar(100)
- Password : varchar(100)
- ContactNo : char(11)
- dob : varchar(100)
- Address : varchar(255)
- City : varchar(100)
- Country : varchar(100)
- RegDate : timestamp
- UpdationDate : timestamp
- role : int(11)

carrental tblpages
- id : int(11)
- PageName : varchar(255)
- type : varchar(255)
- detail : longtext

AUTHORIZATION

Role-based access control (RBAC) policy is used to assign a role to the users and according to the role permissions are assigned are given to users. Every user having the same role has the same set of permissions and if a user whose role is not assigned permission then that user cannot access that information.
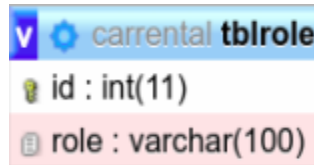
Need for RBAC based Authorization

In our application, there are two entities admin and registered users who have certain access permissions. Also, guest users can view all the cars available but if they want to book the cars then they are not allowed to do so. Only users who have accounts registered with our application i.e are registered users can do the booking of cars. Also, only the admin has access to the admin dashboard so that they can add vehicles and also confirm and reject booking done by users and this functionality in no case should be accessed by the user or guest. So for implementing such an authorization mechanism RBAC is perfect as different entities can be given different roles and hence only authorized personnel have access to relevant information.

Components of RBAC in our Application

We have created a table tbluser in our database to store all the users registered in our system and also a role is assigned to them.

carrental **tblrole**
id : int(11)
role : varchar(100)

In table tblrole we have stored id of user from table tbluser and its role so when user logins then according to their role the functionality is provided. So admin can get access to the admin dashboard and users with role as registered users can book cars as they wiil have option visible on homepage to book car whereas unauthorized user like guest userwhose entry is not present in the database does not have access to either functionality given to admin or registered and can just view cars.

CONCLUSION

In this project we have successfully implemented Role Based Access Control (RBAC) policy in the carrental management system using database approach. Carrental management system follows RBAC policy for authorization which makes sure that roles as assigned properly with the permission and no permissions should overlap. The web based car rental system has offered an advantage to both customers as well as Car Rental Company to efficiently and effectively manage the business and satisfy customers' needs at the click of a button.

REFERENCE

1. Sandhu, R., et al.: Role-based Access Control Models. IEEE Computer (1996).
2. https://en.wikipedia.org/wiki/Role-basedaccesscontrol
3. https://www.php.net/
4. https://docstore.mik.ua/orelly/webprog/webdb/ch08_03.htm
5. https://dev.mysql.com/doc/