



**SDJ INTERNATIONAL
COLLEGE**

Bachelor of Computer Applications (BCA)

Project Report

**BCA Sem VI
AY 2024-25**

Eclipse

by

Exam No.	Name of Student
5429	KALATHIYA ANSH ASHVINBHAI
5847	VALA KARTIK DHARMENDRBHAI
5842	VAGHELA KEVAL MUKESHBHAI
5431	KALSARIYA JATIN KURJIBHAI
5497	LATHIYA YAGNIK JAYSUKHBHAI

Project Guide by :

Prof. Hardik Kabra

Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project work. Whatever we have done is only due to such guidance and assistance.

We would not forget to thank I/C Principal Dr. Aditi Bhatt, Head of Department Prof. Nainesh Gathiyawala and Project guide Prof. Hardik Kabra, and all other Assistant professors of SDJ International College, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

We are extremely grateful to her for providing such a nice support and guidance though she/he had busy schedule managing the college dealings.

We are thankful and fortunate enough to get support and guidance from all Teaching staffs of Bachelor of Computer Application Department which helped us in successfully completing our project work. Also, we would like to extend our sincere regards to all the non-teaching staff of Bachelor of Computer Application Department for their timely support.

ANSH A. KALATHIYA (5429)

KARTIK D. VALA (5847)

KEVAL M. VAGHELA (5842)

JATIN K. KALSARIYA (5431)

YAGNIK J. LATHIYA (5497)

INDEX

Sr. No.	Description	Page No.
1.	Introduction	01
	1.1 Project Summary	02
	1.2 Project Technical Profile	03
2.	Scope & Planning	05
	2.1 Requirement Analysis	06
	2.2 Feasibility Study	10
	2.3 Timeline Chart	12
	2.4 Future Development	14
	2.5 Technologies Details	15
3.	Designing	19
	3.1 Data Flow Diagram	20
	3.2 Use Case Diagram	25
	3.3 Document Data Model Diagram	27
	3.4 Database Design	28
	3.5 Data Dictionary	29
	3.6 User Interface	33
4.	Testing	45
	4.1 Functional Testing	46
	4.2 Environment Testing	47
5.	Conclusion	48
6.	Bibliography	50

1. Introduction

1.1 Project Summary

- **Eclipse** is a comprehensive tour and travel website designed for travelers and adventure seekers.
- Using this platform, users can plan and manage their trips efficiently, explore destinations, and stay updated on their travel itinerary.
- Users can search and compare rental cars based on location, price, car type, and rental duration.
- The platform offers a variety of vehicles, including economy, luxury, SUVs, and vans, to suit different travel needs.
- The website offers useful tools like travel cost estimators and itinerary planners to help users make informed decisions.
- Customers can book cars for short-term or long-term rentals, with options for daily, weekly, or monthly rentals.
- The website provides pickup and drop-off location selection, allowing users to choose convenient rental locations, including airports and city centers.
- Users can view rental policies, pricing details, and insurance options before confirming their booking.
- The platform includes a customer support chat for booking assistance and rental inquiries.
- Secure online payment options ensure a hassle-free booking process. Travelers can discover popular attractions, restaurants, and local experiences for their chosen destinations.
- Privacy is a priority, with security features like end-to-end encryption and password protection to ensure user data is safe.

1.2 Project Technical Profile

Fields	Descriptions
Project Title	Eclipse (Tour & Travels)
Definition	Eclipse is a travel and car rental website that lets users plan trips and rent vehicles as per their needs. It offers destination planning, a variety of rental cars and secure booking options for a seamless travel experience.
Developed For	S. D. J. International College, Vesu, Surat
Project Guide	Prof. Hardik Kabra
Front End	Microsoft visual studio
Back End	MySQL
Programming Language	PHP5, HTML
Operating System	Microsoft Windows 10
Submitted By	<ol style="list-style-type: none"> 1. KALATHIYA ANSH ASHVINBHAI 2. VALA KARTIK DHARMENDRBHAI 3. LATHIYA YAGNIK JAYSUKHBHAI 4. KALSARIYA JATIN KURJIBHAI 5. VAGHELA KEVAL MUKESHBHAI

Hardware & Software Requirement:

At Development Time...

❖ Hardware Requirement:

- Intel® Core™ i5-3340M CPU @ 2.70GHz
- Minimum 8.0 GB DDR3 RAM
- 64-bit Operating System
- 128 GB Hard Disk Drive

❖ Software Requirement:

- Microsoft Visual Studio
- XAMPP (64 bit)
- Working Internet Connection
- SQLite Database

2. Scope & Planning

2.1 Requirement Analysis

Requirement Gathering:

The **Eclipse** car rental website will provide users with a seamless platform to rent vehicles for their travels. Users can search for available cars, choose from various booking plans, and make secure online payments. The rental cost is calculated per kilometer based on the car model, ensuring flexible pricing options. The website will support multiple payment gateways and allow users to manage their bookings, view history, and select pickup and drop-off locations. A customer support system with live chat and automated assistance will be available to handle inquiries and booking issues.

On the backend, Eclipse will use a MySQL database to store user information, booking details, and car listings securely. The platform will be built using modern web technologies like PHP, JavaScript, and Bootstrap, ensuring a responsive and efficient user experience. Security measures such as data encryption, secure logins, and role-based access control will be implemented to protect user information. Additionally, features like auto-sync for offline bookings, cloud storage for data backup, and API integrations for payment processing and location services will ensure reliability and scalability. Future enhancements may include a mobile app, GPS tracking, and a loyalty rewards program to enhance user engagement.

Requirement Analysis:

The **Eclipse** car rental website requires a well-defined requirement analysis to ensure efficient system performance, security, and user convenience. The primary objective is to provide a seamless car rental service where users can browse available vehicles, compare pricing, and book cars based on their travel needs. The pricing structure is based on a per-kilometer cost, which varies according to the car model.

Users should have the flexibility to choose rental plans (hourly, daily, weekly, or monthly) and select pickup/drop-off locations. The website should also support secure online payments with multiple transaction options and maintain a booking history for users. Additionally, an admin panel is required to manage car listings, pricing, and bookings, ensuring smooth operational control.

From a technical perspective, Eclipse will use PHP, MySQL, and JavaScript to create a responsive and scalable platform. The MySQL database will securely store user details, car availability, and booking records, while cloud storage will be used for backup and synchronization of offline data. Security features like data encryption, secure login authentication, and role-based access will be implemented to protect sensitive information.

The system will also integrate Google Maps API for selecting rental locations and estimating distances. To ensure reliability and performance, the website will be optimized for high availability and scalability, capable of handling multiple users simultaneously. Future enhancements may include a mobile app, GPS tracking, customer reviews, and loyalty programs to improve user engagement and customer satisfaction.

1. Functional Requirement Analysis:

1.1 User Management:

The system must allow users to create accounts, log in securely, and manage their profiles. User authentication will be handled using secure login mechanisms such as email and password authentication. Additional features like password recovery, OTP verification, and profile management (updating contact details, payment preferences, etc.) should be available.

1.2 Car Search and Booking:

Users should be able to search for rental cars based on various parameters such as car type (economy, luxury, SUV), location, price, and availability. The system

will allow users to choose rental plans (hourly, daily, weekly, or monthly) and calculate the rental cost based on a per-kilometer pricing model, which varies depending on the car type. After selecting a vehicle, users can confirm their booking by choosing pickup and drop-off locations and making an online payment.

1.3 Payment Processing:

The system should support multiple online payment methods, including credit/debit cards, UPI, net banking, and digital wallets. Secure payment gateway integration (e.g., PayPal, Stripe, Razorpay) will be required. After booking, users should receive an automated invoice detailing the rental cost, taxes, and total amount paid. The system must also handle refunds and cancellations per the company's rental policies.

1.4 Admin Panel for Vehicle & Booking Management:

The admin panel should allow administrators to add, update, or remove vehicles from the rental database. Admins can set rental rates, manage bookings, and view customer transactions. The system should provide an overview of available and rented cars, ensuring that rental operations are efficiently managed.

1.5 Pickup and Drop-off Locations:

Users should have the flexibility to select pickup and drop-off locations. The system will integrate Google Maps API for location-based services, allowing users to find nearby rental stations and estimate distances between locations for cost calculation.

1.6 Customer Support & Assistance:

The system should include a live chat feature and a support ticketing system for users to raise issues or inquiries related to their bookings. This ensures better user engagement and prompt resolution of customer concerns.

2. Non-Functional Requirement Analysis:

2.1 Performance and Scalability:

The website should be optimized for fast loading times and support multiple users simultaneously without performance issues. The backend must be designed to handle large volumes of booking requests efficiently.

2.2 Security & Data Protection:

The system will implement secure login authentication, data encryption, and role-based access control to protect user and financial information. Payment transactions will be processed using PCI DSS-compliant payment gateways to ensure data security.

2.3 Database Management:

The system will use MySQL as the primary database to store information on users, bookings, cars, and transactions. Data backup mechanisms will be in place to prevent data loss, and cloud storage will be used for real-time synchronization of offline and online data.

2.4 Availability and Reliability:

The system must have 99.9% uptime, ensuring continuous availability for users. Auto-sync functionality should update offline data to the online database once the connection is restored.

2.5 Compliance and Legal Considerations:

The platform must adhere to GDPR compliance and data privacy regulations to protect user information. Clear terms and conditions for renting cars, cancellation policies, and liability clauses must be implemented.

2.2 Feasibility Study

1. Technical Feasibility:

Technical feasibility assesses whether the project can be successfully developed using available technology, software, and resources. The Eclipse website is technically feasible as it is designed to work on any device with a web browser and internet connection.

It will be built using PHP, JavaScript, MySQL, and Bootstrap, ensuring compatibility with various platforms. The website will also integrate Google Maps API for location services and secure payment gateways for transactions.

Modern cloud-based hosting solutions, such as AWS or Firebase, will provide high availability and scalability.

Given the availability of skilled developers and the necessary technical components, the project is feasible from a technical perspective.

2. Economic Feasibility:

Economic feasibility determines whether the project is cost-effective in terms of development and long-term operation. The Eclipse platform is expected to be financially viable due to its low development costs and high potential for revenue generation through car rental bookings.

The main cost components include website development, hosting, payment gateway integration, database management, and security implementations. These costs are mostly one-time investments, while maintenance and server costs are recurring but manageable.

The platform's revenue model, which may include rental fees, commission-based earnings, and premium booking features, ensures a return on investment, making it economically feasible.

3. Operational Feasibility:

Operational feasibility assesses whether the system will be effectively used by end-users and if it aligns with business needs. The Eclipse car rental platform is designed for ease of use, with simple navigation, an intuitive booking system, and secure payment processing.

Users will have multiple car rental plans, pickup/drop-off location selection, and online payment options, ensuring a smooth rental experience. Administrators will have full control over managing vehicles, bookings, and customer interactions. Since the system meets business and customer needs efficiently, operational feasibility is confirmed.

4. Time Feasibility:

Time feasibility evaluates whether the project can be completed within an acceptable timeframe.

The Eclipse Car Rental Website is designed with a structured development plan, ensuring that all major functionalities—including user registration, car booking, payment integration, and admin management—are implemented within the estimated 80 to 90 days.

With proper project management, including task prioritization and agile development methodologies, the system can be delivered on schedule, making it time-feasible.

5. Legal Feasibility:

Legal feasibility ensures that the system complies with relevant laws and regulations. The Eclipse platform adheres to data protection laws, ensuring that user information, payment details, and rental history are stored securely.

2.3 Timeline Chart

When Scheduling of a software project is done. The planner begins with a set of tasks to be performed. If automated tools are used; the work breakdown is input as a task network or task outline. Effort, duration and start date are then input for each task. In addition, tasks may be assigned to specific individuals.

Work Tasks	Month	Dec		Jan				Feb				Mar			
	Week	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. Requirement Gathering and Analysis															
1.1 Requirement Gathering															
1.2 Identifying Needs															
1.3 Requirement Analysis															
2. Scope and Planning															
2.1 Information Gathering															
2.2 Problem Specification															
2.3 Feasibility Study															
2.4 Risk analysis															
2.5 Scheduling Chart															
3. Designing															
3.1 Database Design															
3.2 Use Case Design															
3.3 Data Flow Diagram															
3.4 Document Data Model Diagram															
3.5 Data Dictionary															
3.6 User Interface															
4. Coding & Logic Development															
4.1 Coding for Modules															
4.2 Implement Logic for Application															
4.3 Add Security for App data															
4.4 Finalize Application															
5. Testing & Reviewing															
5.1 Testing of Application															
5.2 Review Application & Bug Fixings															
6. Documentation															

2.4 Future Development

- To enhance security, we will implement advanced cybersecurity measures to protect user data, including secure payment gateways, encrypted transactions, and fraud detection mechanisms.
- Introducing additional privacy features, such as two-factor authentication (2FA) and user-controlled data access, will provide users with greater control over their personal and booking details.
- Integrating UPI and digital wallets for seamless payments will allow users to pay instantly without additional transaction fees, improving the booking experience.
- An offline booking request feature will be added, enabling users to reserve vehicles even when they have a weak internet connection, ensuring uninterrupted service.
- To simplify the car rental process, we will introduce AI-based recommendations to suggest the best vehicle based on a user's preferences, travel distance, and budget.
- Enhancing customer support by adding live chat, in-app messaging, and AI chatbots will improve communication and help resolve issues quickly.
- Expanding data export options, including support for PDF, CSV, and Excel formats, will allow users to download and manage their booking history more effectively.

2.5 Technologies Details

This project will be developed in Microsoft Visual Studio Code – IDE with PHP as the front end and MySQL as the back end. The application works on web-based platforms, allowing users to access it from any device with an internet connection.

Front end and back end are generalized terms that refer to the initial and end stages of handling a system. The front end is responsible for collecting user inputs and displaying relevant information, while the back end processes data, handles transactions, and stores records securely. The front end serves as an interface between the user and the back end, ensuring smooth interaction and accessibility.

The main reason for forming a combination of these two ends is to provide different functionalities for managing bookings, processing payments, and handling user requests.

The front end handles user interactions and displays car rental options, while the back end securely stores and processes all booking and transaction details. This ensures a seamless, efficient, and user-friendly experience for customers using the Eclipse Car Rental Website.

Main Programming Language:

- ❖ Programming Language: PHP

Different Programming Environment:

- ❖ Front end: HTML, CSS, JavaScript, Bootstrap (for a responsive UI)
- ❖ Back end: MySQL

Other Tools:

- ❖ JSON, API

Microsoft Visual Studio:

Microsoft Visual Studio is a powerful integrated development environment (IDE) developed by Microsoft for creating web, mobile, desktop, and cloud-based applications.

It supports multiple programming languages such as C#, PHP, JavaScript, and Python, providing developers with advanced tools for coding, debugging, testing, and deployment. Its built-in IntelliSense feature offers code suggestions, syntax highlighting, and auto-completion, making development faster and more efficient.

The IDE also integrates seamlessly with version control systems like Git and Azure DevOps, enabling smooth collaboration among developers.

In the Eclipse Car Rental Website, Microsoft Visual Studio Code is used as the primary code editor for developing the PHP-based platform with a MySQL database. It provides a lightweight yet powerful environment for writing, testing, and debugging code while supporting front-end technologies like HTML, CSS, and JavaScript. With its extension support and real-time collaborative features like Live Share, it enhances productivity and ensures a seamless development experience.

The tool's flexibility and cloud integration capabilities make it ideal for building scalable web applications, ensuring a robust and user-friendly experience for car rental services.

Flutter is an open-source UI software development kit created by Google. It is used to develop cross-platform applications for Android, iOS, Linux, macOS, Windows, Google Fuchsia, and the web from a single codebase.

APIs:

An application programming interface (API) is a way for two or more computer programs to communicate with each other. It is a type of software interface, that offers a service to other pieces of software. A document or standard that describes how to build or use such a connection or interface is called an API specification. A computer system that meets this standard is said to implement or expose an API. The term API may refer either to the specification or to the implementation.

In contrast to a user interface, which connects a computer to a person, an application programming interface connects computers or pieces of software to each other. It is not intended to be used directly by a person (the end-user) other than a computer programmer who is incorporating it into the software.

An API is often made up of different parts which act as tools or services that are available to the programmer. A program or a programmer that uses one of these parts is said to call that portion of the API.

The calls that make up the API are also known as subroutines, methods, requests, or endpoints. An API specification defines these calls, meaning that it explains how to use or implement them.

One purpose of APIs is to hide the internal details of how a system works, exposing only those parts a programmer will find useful and keeping them consistent even if the internal details later change. An API may be custom-built for a particular pair of systems, or it may be a shared standard allowing interoperability among many systems.

JSON:

JSON (JavaScript Object Notation, pronounced) is an open standard file format and data interchange format that uses human-readable text to store and transmit data objects consisting of attribute–value pairs and arrays (or other serializable values). It is a common data format with diverse uses in electronic data interchange, including that of web applications with servers.

JSON is a language-independent data format. It was derived from JavaScript, but many modern programming languages include code to generate and parse JSON-format data. JSON filenames use the extension. json. Any valid JSON file is a valid JavaScript (.js) file, even though it makes no changes to a web page on its own.

Douglas Crockford originally specified the JSON format in the early 2000s. He and Chip Morningstar sent the first JSON message in April 2001.

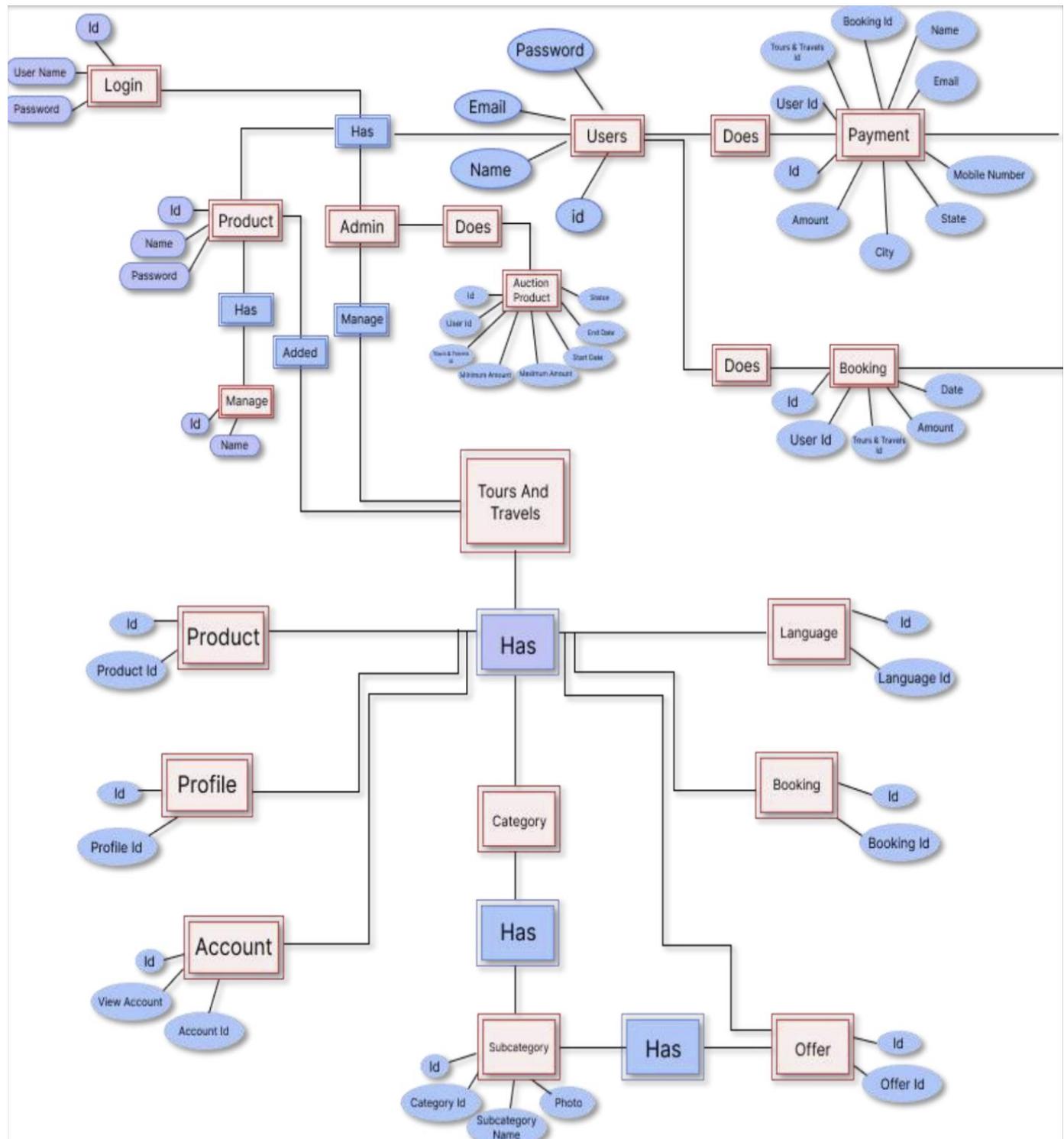
MySQL:

MySQL is an open-source relational database management system (RDBMS) developed by Oracle Corporation. It is widely used for storing, managing, and retrieving structured data in various applications, including web development, enterprise systems, and cloud applications. MySQL uses Structured Query Language (SQL) to interact with databases, making it easy to create, update, and manage data efficiently.

MySQL is known for its scalability, security, and high performance, making it a preferred choice for applications that require handling large amounts of data. It supports ACID (Atomicity, Consistency, Isolation, Durability) compliance, ensuring data integrity and reliability. MySQL is compatible with various programming languages like PHP, Python, Java, and C++, and is often used in combination with web technologies like Apache and PHP (LAMP stack) for dynamic website development.

3. Designing

❖ Entity Relationship Diagram (ER Diagram)



E-R Diagram:

- Entity Relational model is a model for identifying entities to be represented in the database and representation of how those entities are related.
- The ER data model specifies enterprise schema that represents the overall logical structure of a database graphically.
- ER Diagrams are used to model real-world objects like a person, a car, a company and the relation between these real-world objects.

Entity:



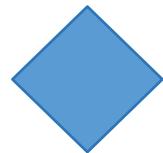
An Entity may be any object, class, person or place. In the ER diagram, an entity can be represented as rectangles.

Attribute:



The Attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.

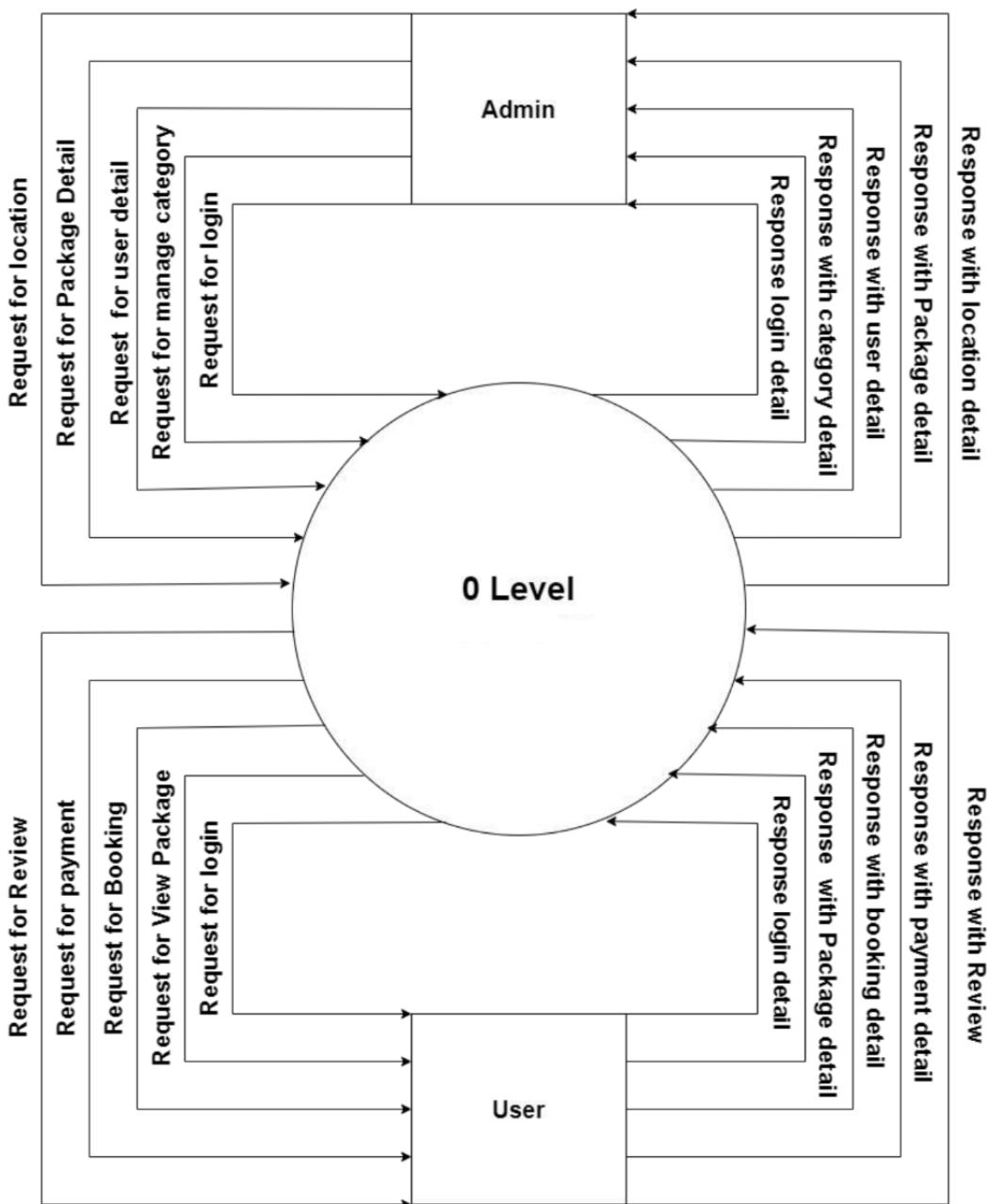
Relationship:

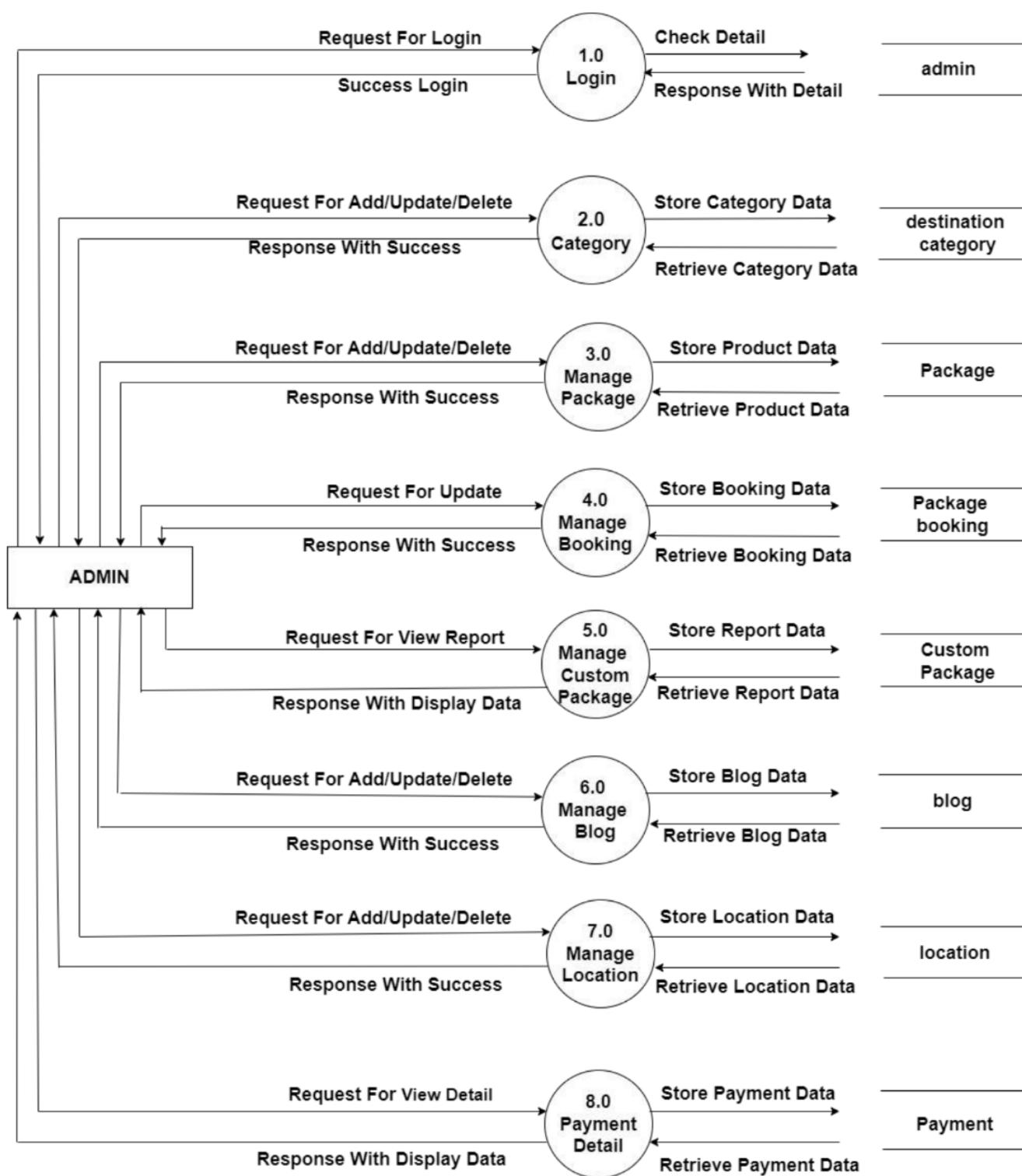


A Relationship is used to describe the relation between entities. Diamond or rhombus is used to represent the relationship.

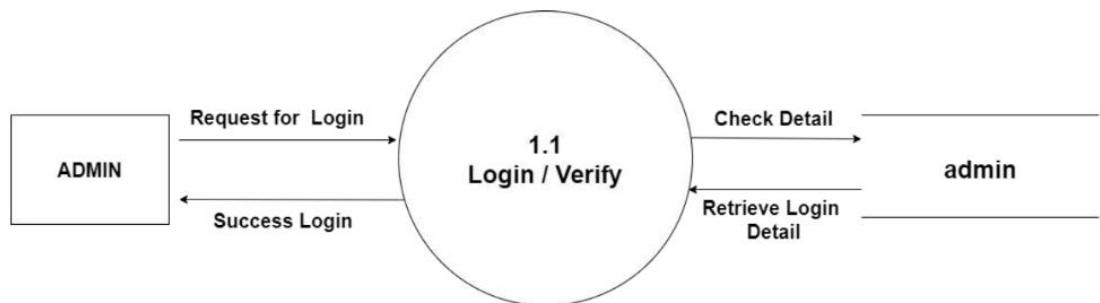
3.1 Data Flow Diagram

- ❖ Context Level Data flow diagram (Level - 0):

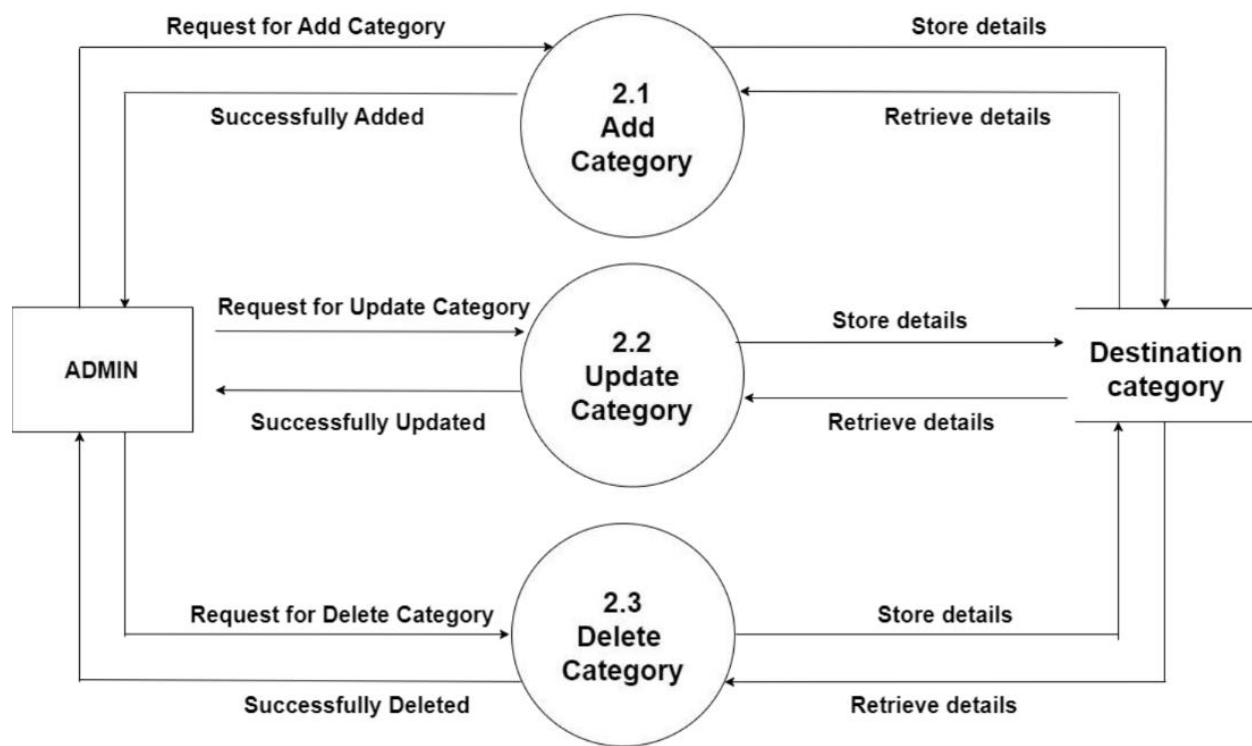


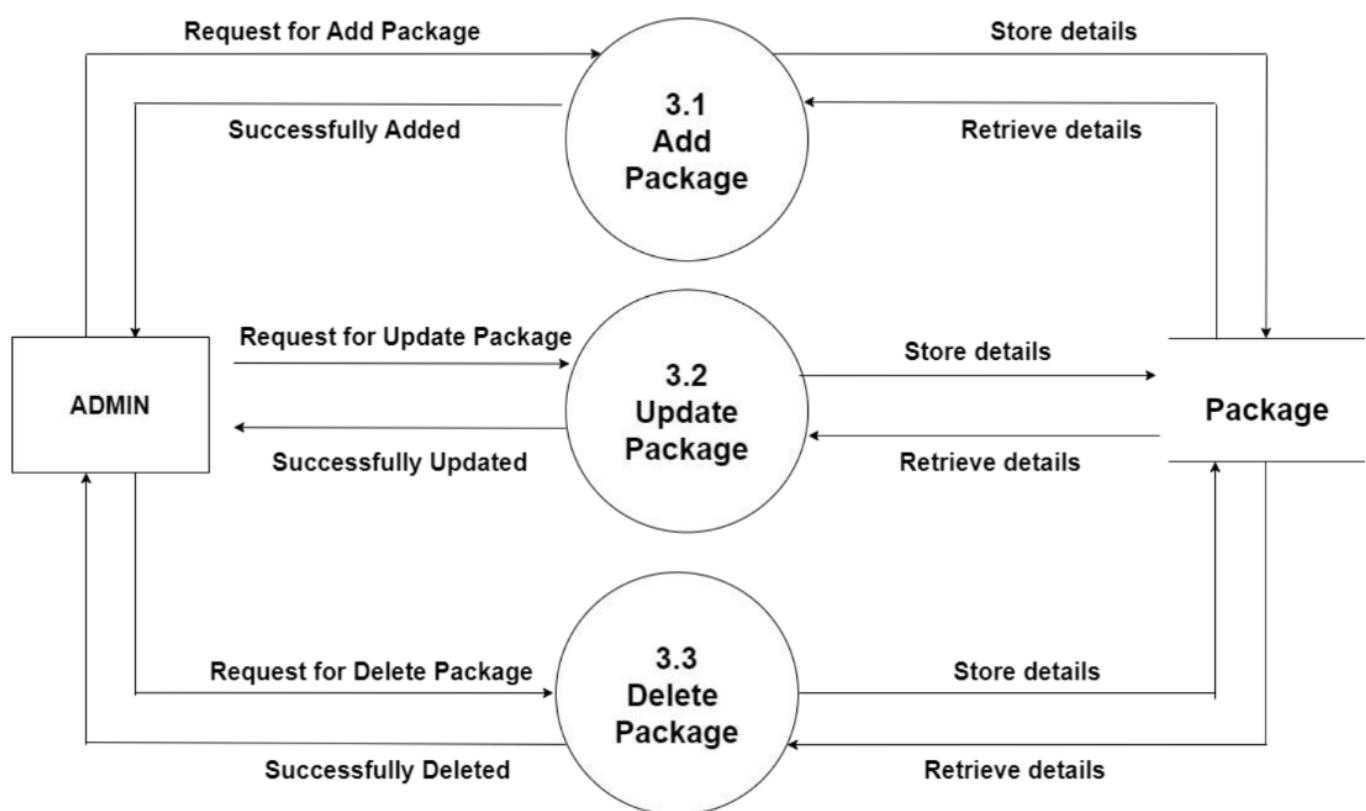
❖ 1st Level Data Flow Diagram: Admin side


❖ 1st Level Data Flow Diagram: User Side

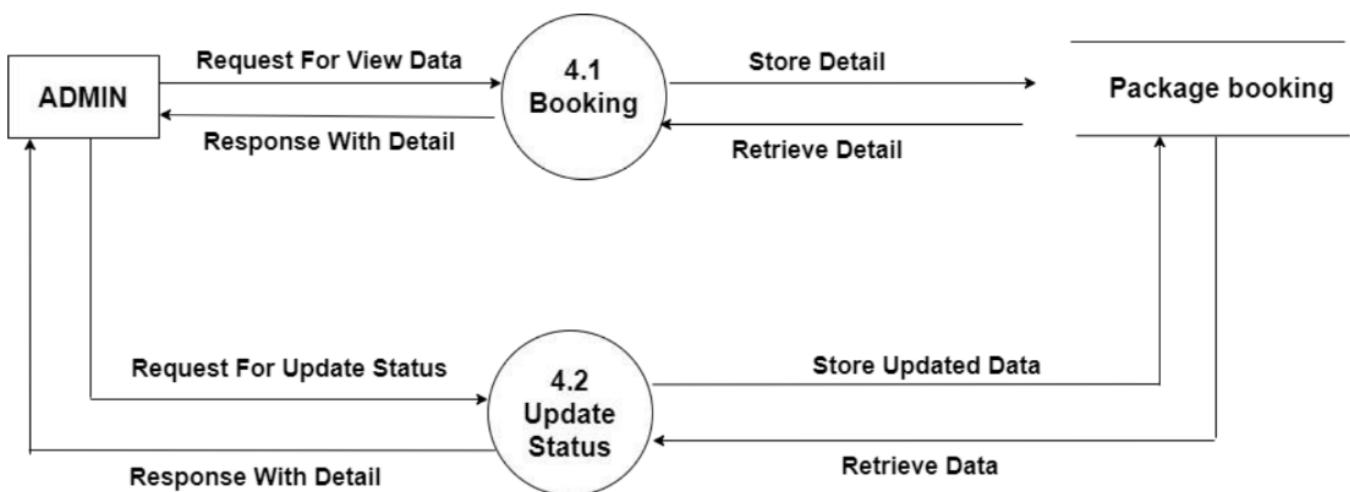


❖ 2nd Level Category Flow For Admin

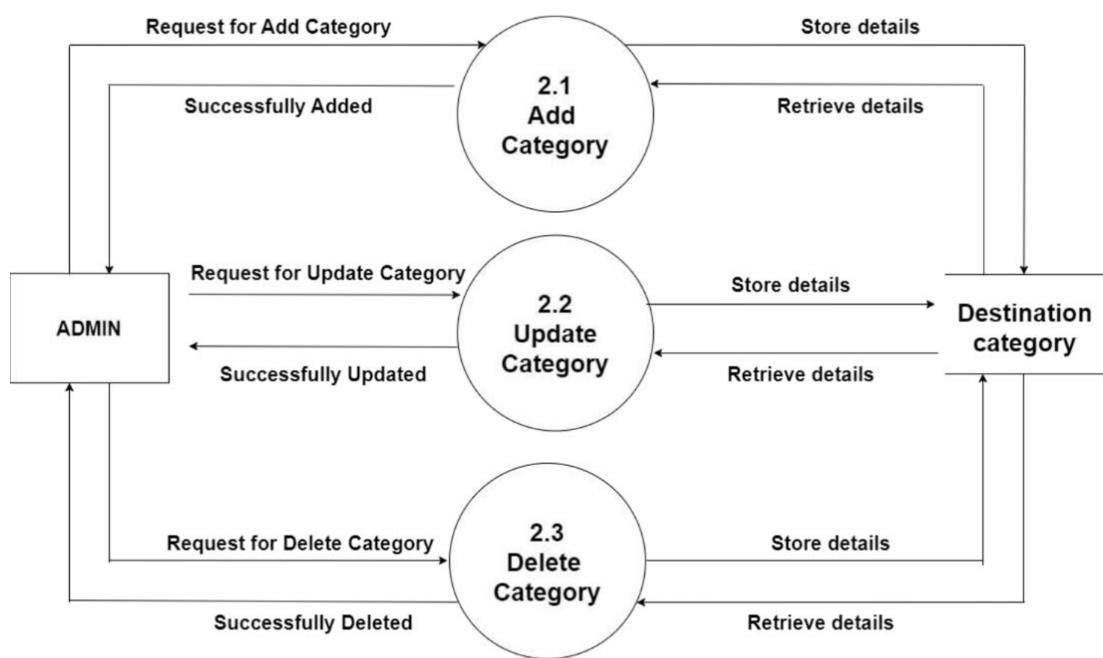


❖ 2nd Level Package Flow For Admin


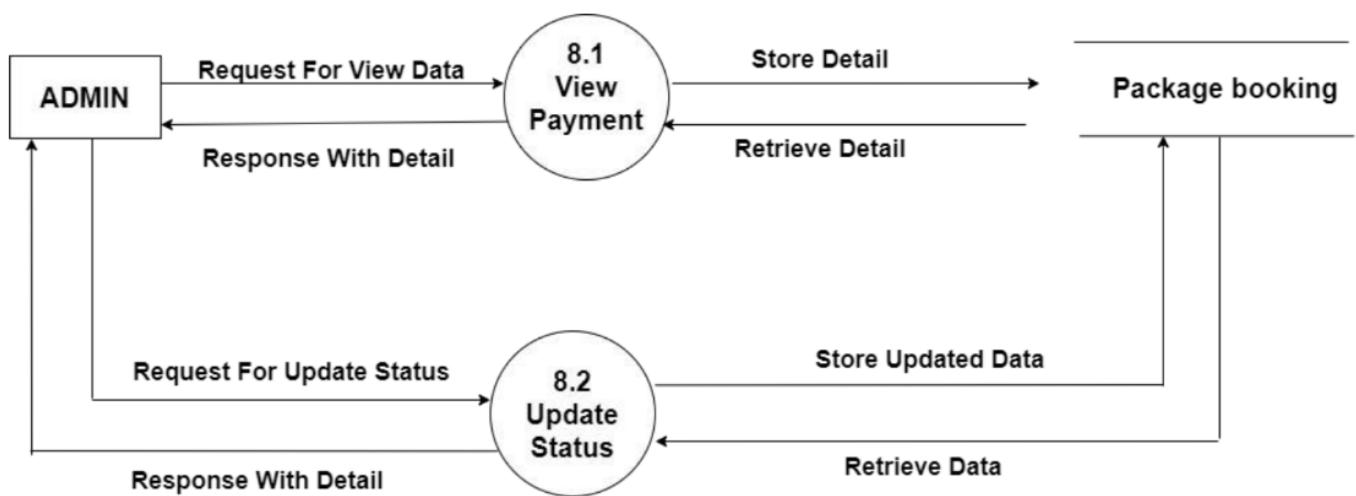
❖ 2nd Leval Booking Flow For Admin

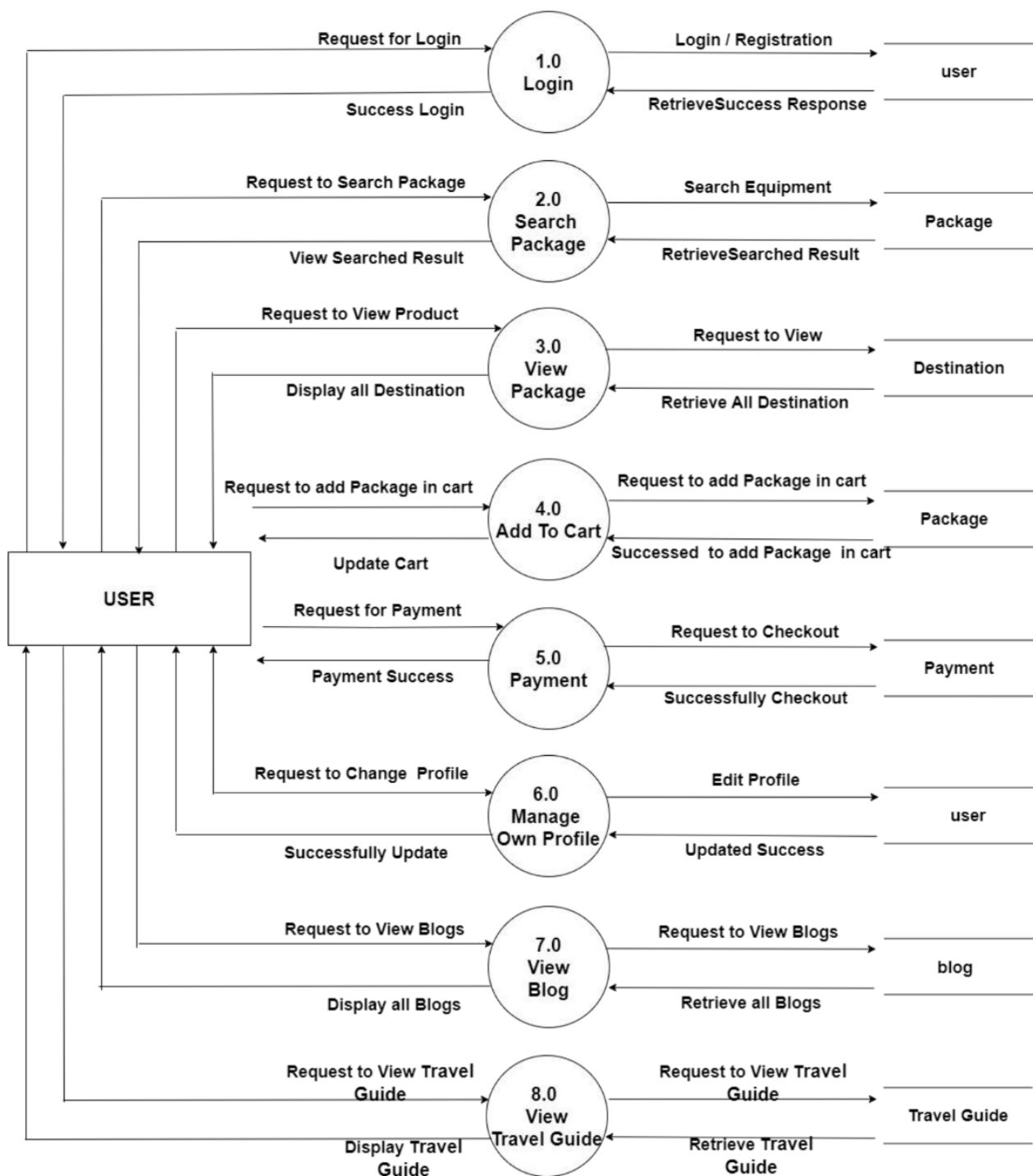


2nd Level blog management Flow For Admin

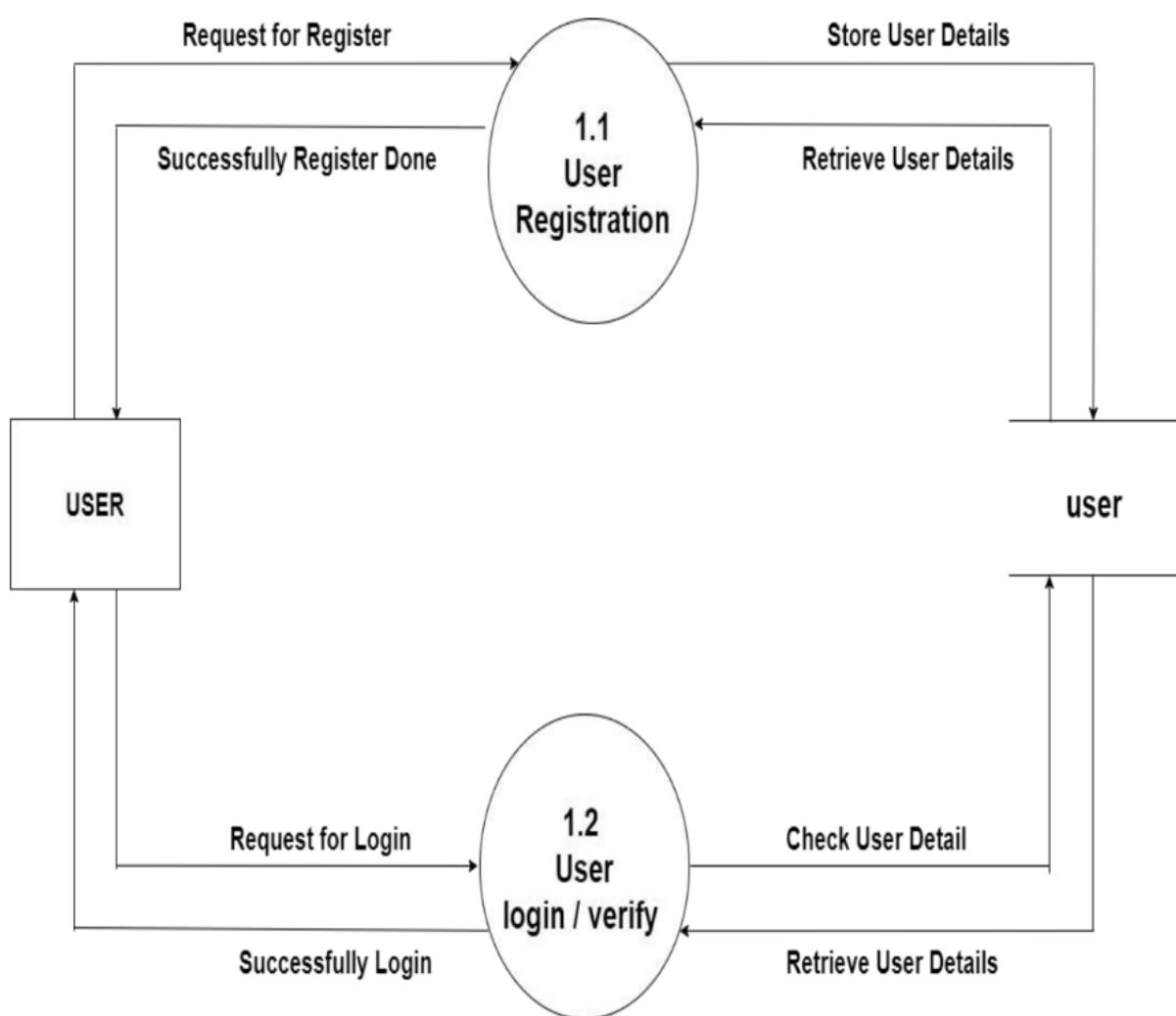


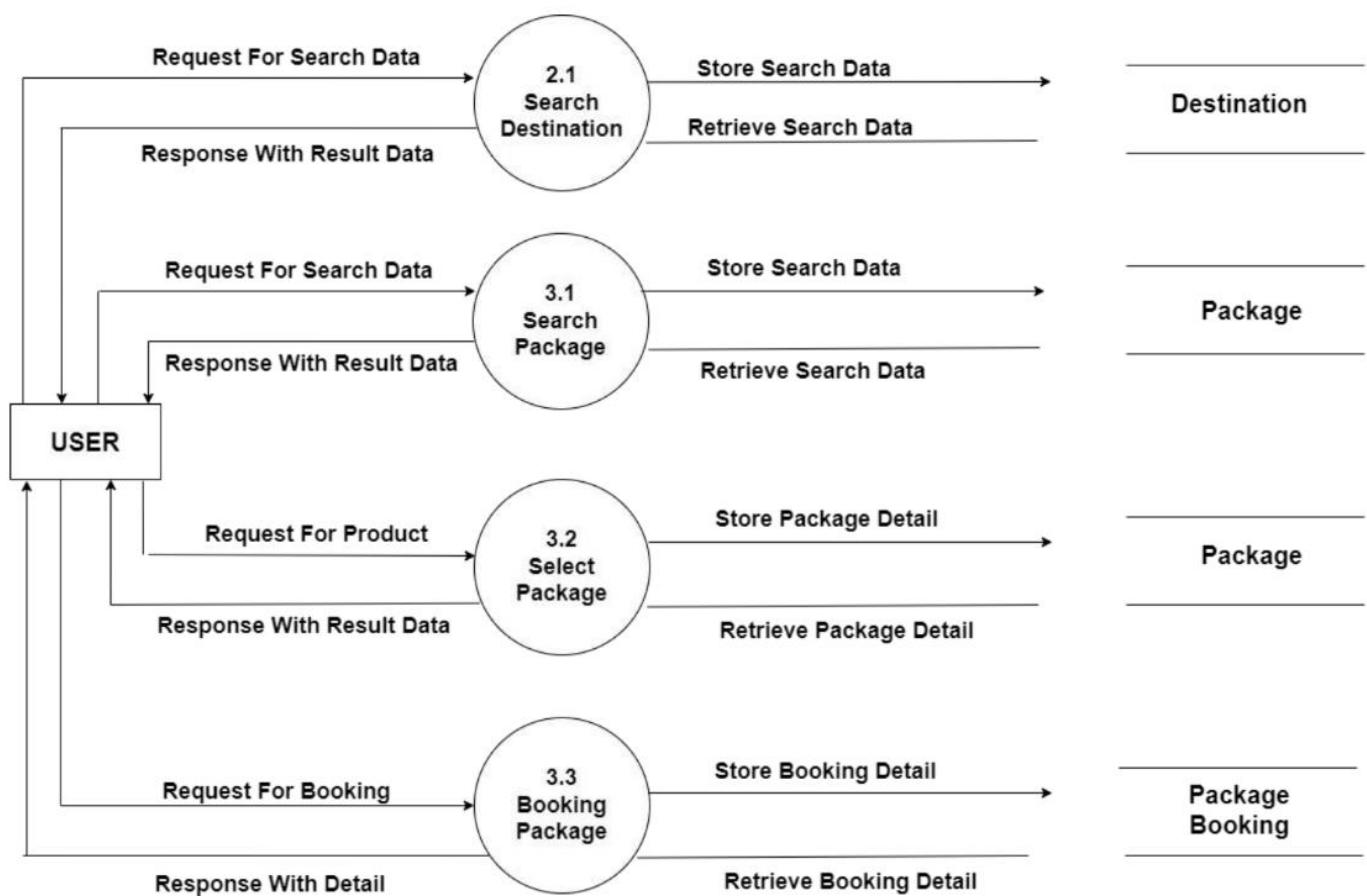
❖ 2nd Level Payment Management Flow For Admin



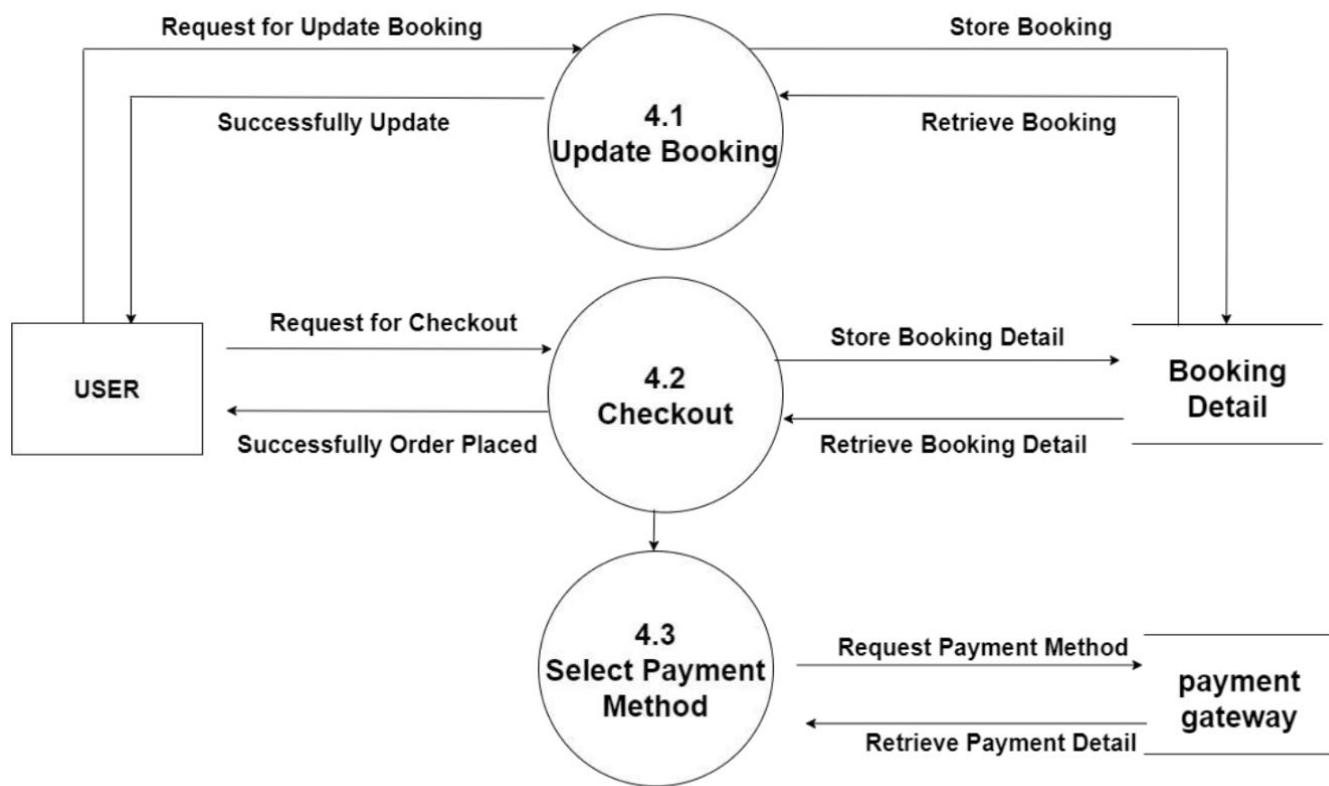
❖ 1st Level Diagram for user


❖ 2nd Level Login Flow For User

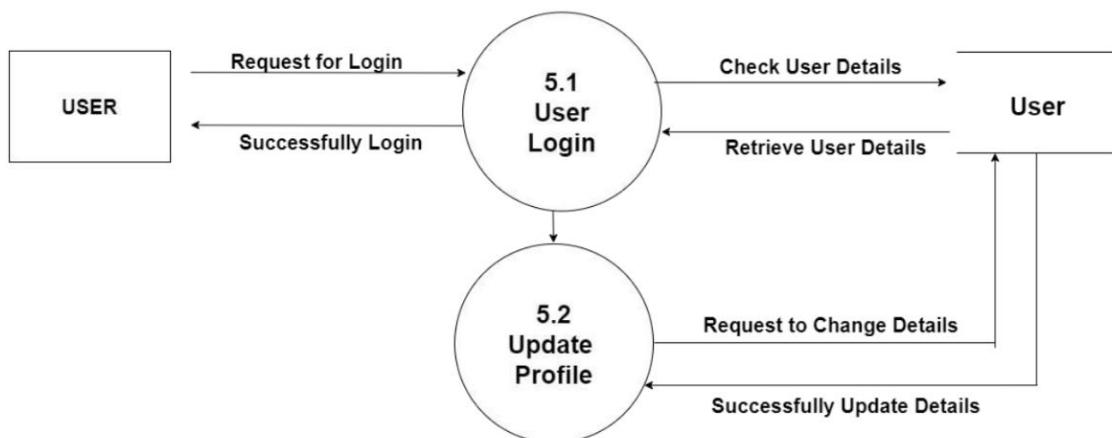


❖ 2nd Level Pacakge Flow For User


❖ **2nd Level Payment Flow For User**

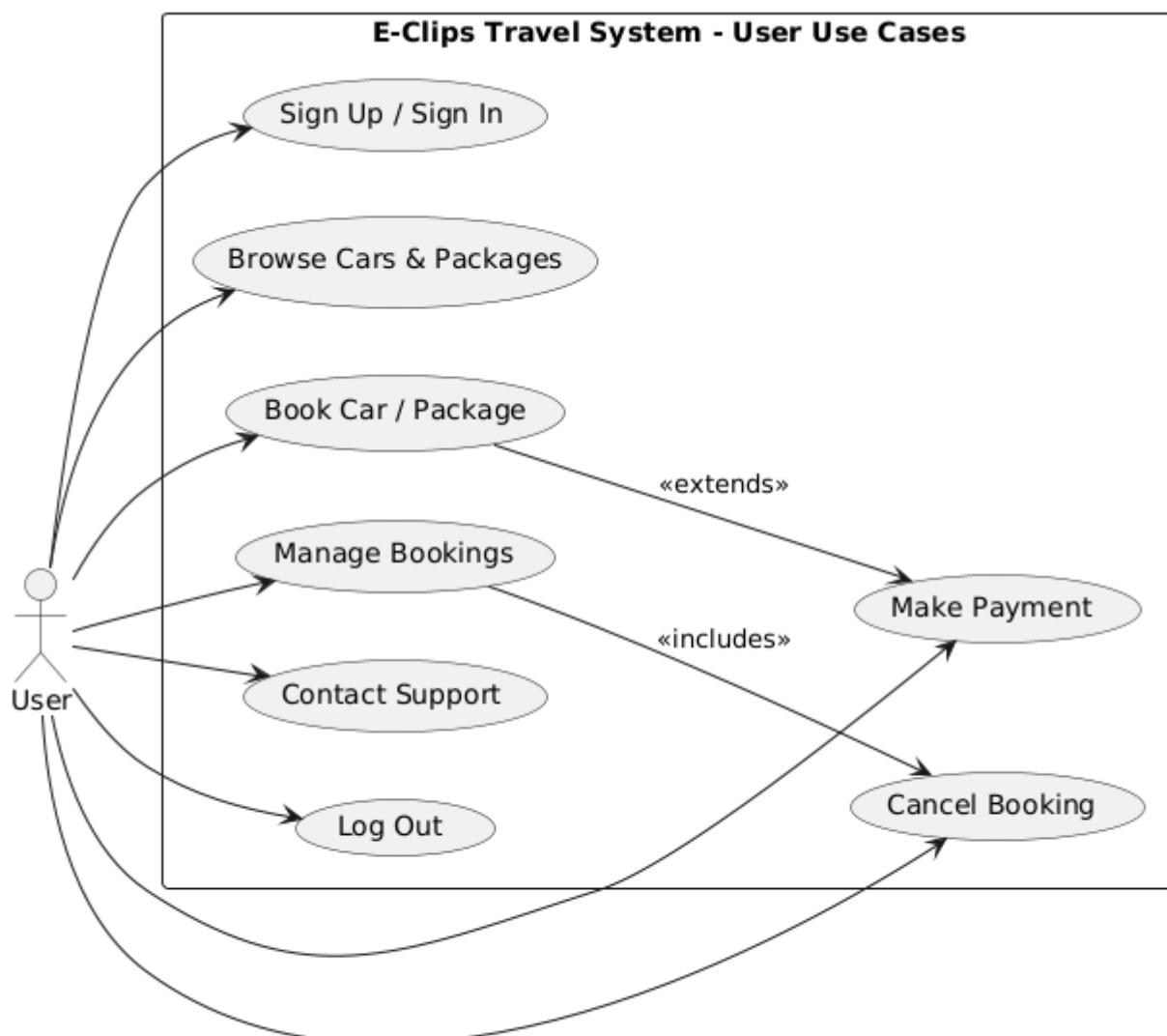


❖ **2nd Level Profile Management Flow For User**

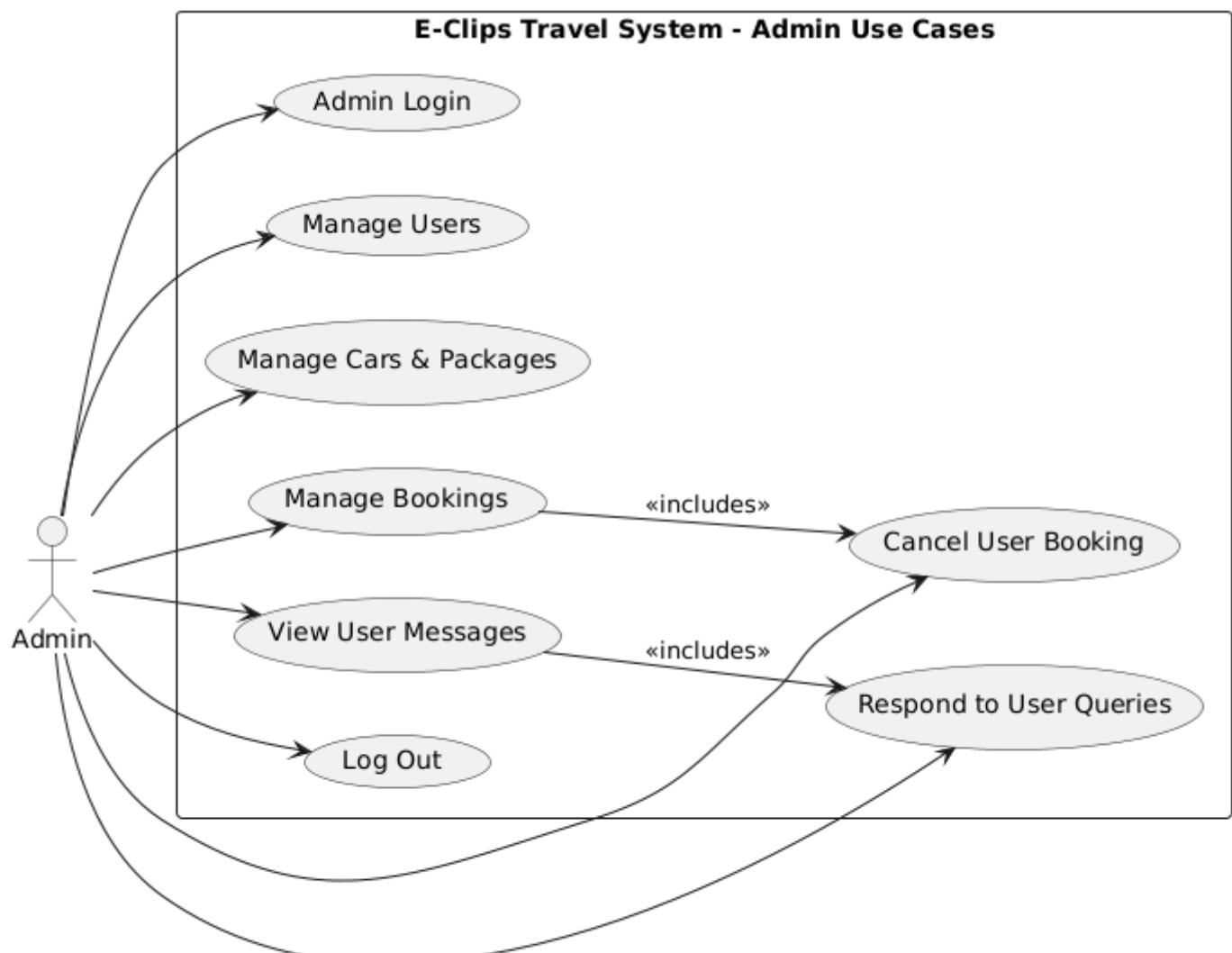


3.2 Use Case Diagram

❖ Use Case Diagram: User

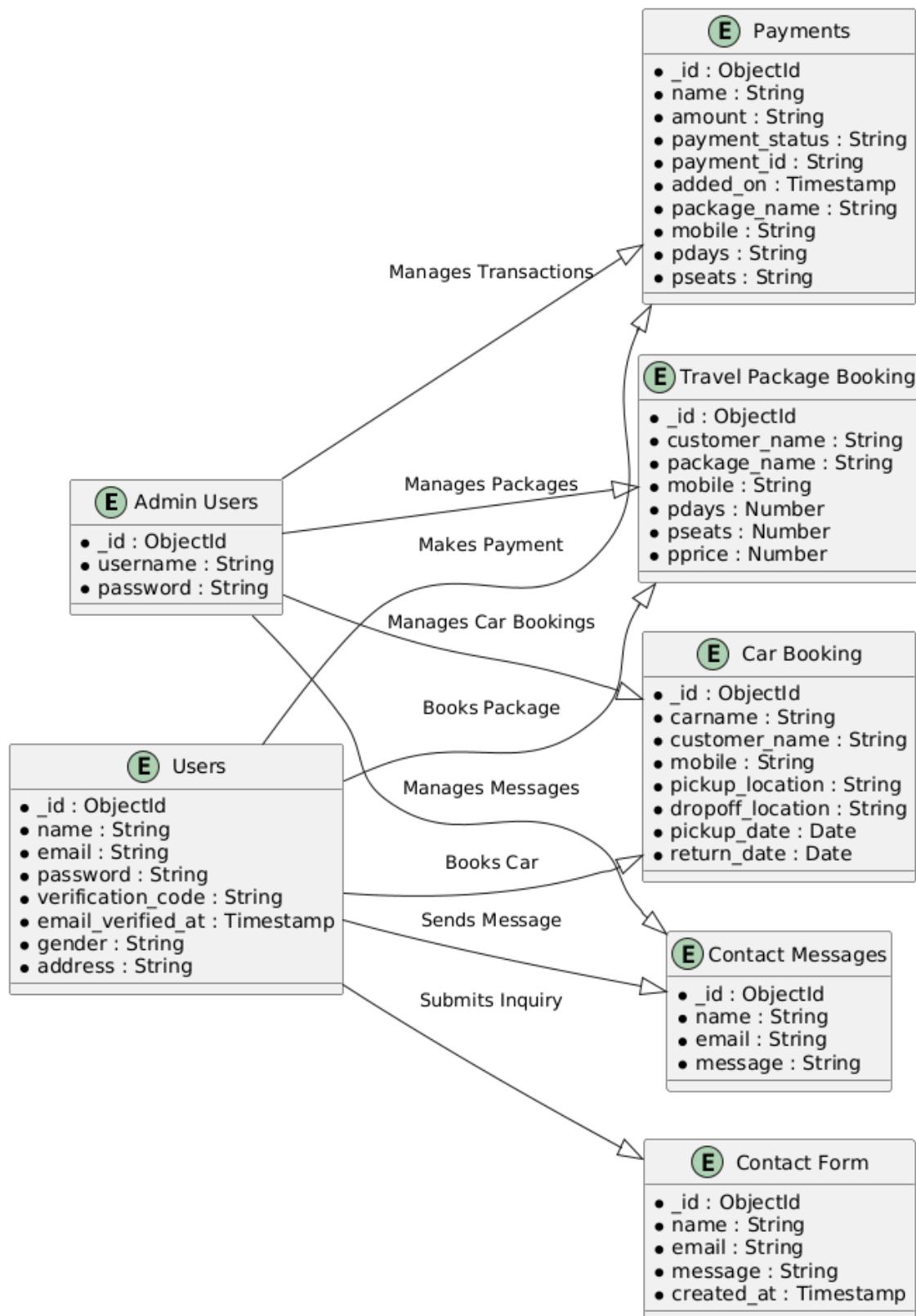


❖ Use Case Diagram: Group Admin



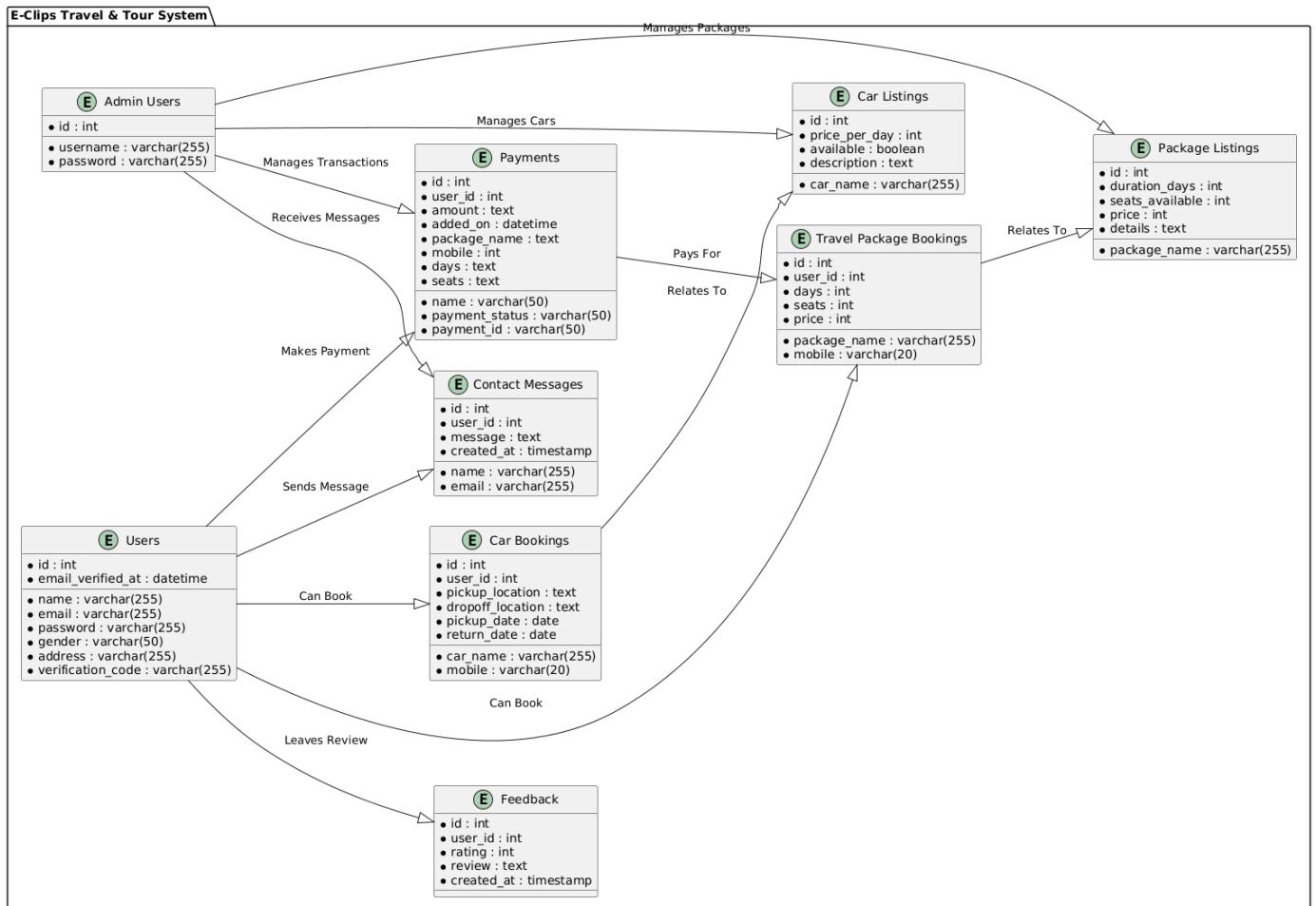
3.3 Document Data Model Diagram

❖ Documents Data Model Diagram (Class Diagram):



3.4 Database Design

❖ Database Design:



3.5 Data Dictionary

Collection Name: admin_users

Description: This collection contains all data of user.

FIELD NAME	FIELD TYPE	DESCRIPTION
id<<PK>>	int	Primary Key - Unique Admin User ID
username	varchar(255)	Admin username
password	varchar(255)	Admin password

Collection Name: contact_form

Description: This collection contains all data of user contact.

FIELD NAME	FIELD TYPE	DESCRIPTION
id<<PK>>	int	Primary Key - Unique Contact ID
name	varchar(255)	User's name
email	varchar(255)	User's email
message	text	Message content
created_at	timestamp	Record creation time

Collection Name: pay

Description: This collection contains all data of payments.

FIELD NAME	FIELD TYPE	DESCRIPTION
id<<PK>>	int	Primary Key - Unique Payment ID
name	varchar(50)	Payer's name
amount	text	Payment amount
payment_status	varchar(50)	Status of payment
payment_id	varchar(50)	Transaction ID
added_on	datetime	Date and time of payment
pname	text	Package name
mobile	Int(11)	User's mobile number
pdays	text	Package duration in days
pseats	text	Number of seats booked

Collection Name: tb_form

Description: This collection contains data of user current balance.

FIELD NAME	FIELD TYPE	DESCRIPTION
id<<PK>>	int	Primary Key - Unique Form ID
carname	text	Car name
name	text	User's name
mobile	text	User's mobile number
pickuplocation	text	Pickup location
dropofflocation	text	Drop-off location
pickupdate	text	Pickup date
returndate	text	Return date

Collection Name: tb_package

Description: This collection contains all data package.

FIELD NAME	FIELD TYPE	DESCRIPTION
id<<PK>>	int	Primary Key - Unique Package ID
name	text	User's name
pname	text	Package name
mobile	int(11)	User's mobile number
pdays	int(11)	Package duration in days
pseats	int(11)	Number of seats booked
pprice	int(11)	Package price

Collection Name: tbl_con

Description: This collection contains data.

FIELD NAME	FIELD TYPE	DESCRIPTION
id<<PK>>	int	Primary Key - Unique Contact ID
name	text	User's name
email	text	User's email
message	text	Message content

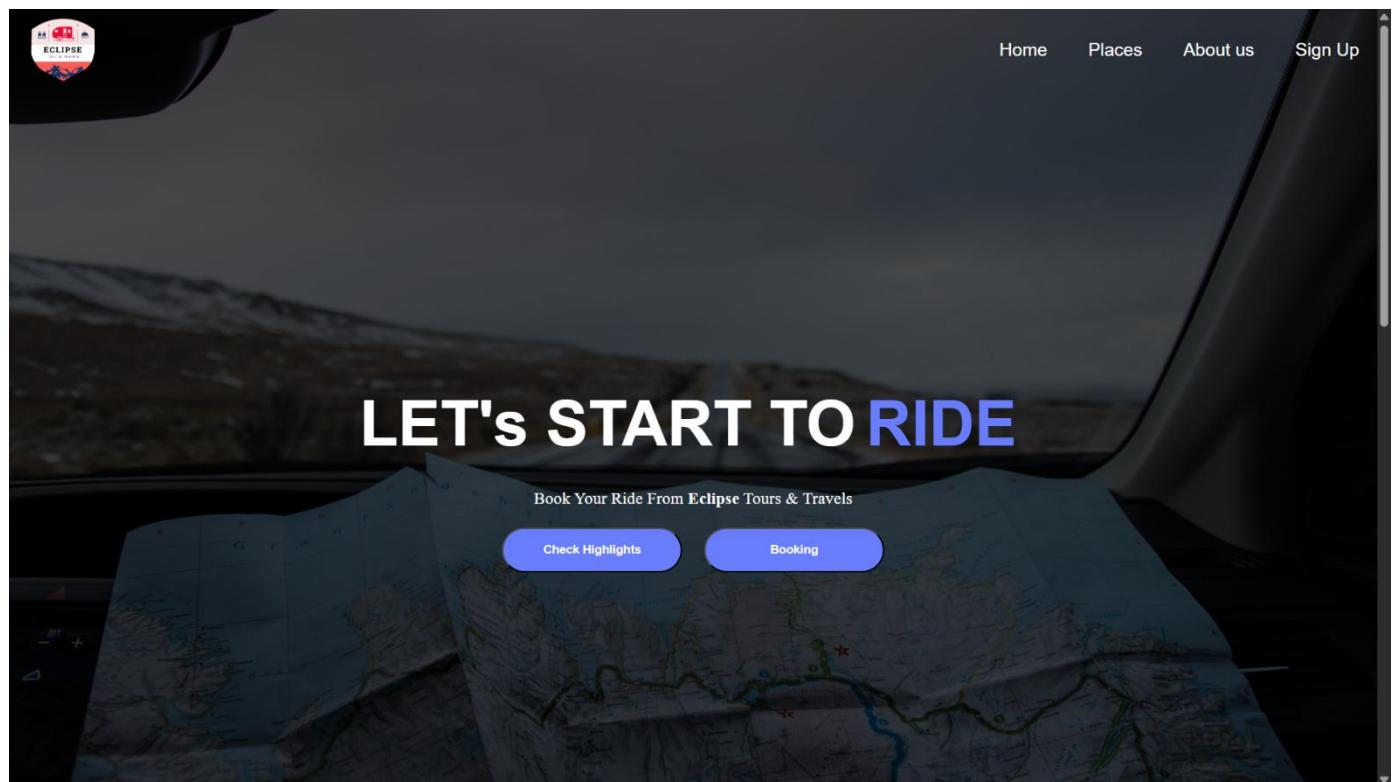
Collection Name: *tbl_signup*

Description: This collection contains data of signup.

FIELD NAME	FIELD TYPE	DESCRIPTION
id<<PK>>	int	Primary Key - Unique User ID
name	text	User's full name
email	text	User's email
password	text	User's password
verification_code	text	Email verification code
email_verified_at	datetime	Email verification timestamp
gender	varchar(52)	User's gender
address	varchar(50)	User's address

3.6 User Interface

❖ Welcome Screen:



Why Choose Us

Variety of Car Brands

We offer a wide range of car brands to choose from.

Best Rated Guarantee

We provide the best-rated guarantee for your satisfaction.

24/7 Customer Support

Our customer support is available.

Best Rented Car

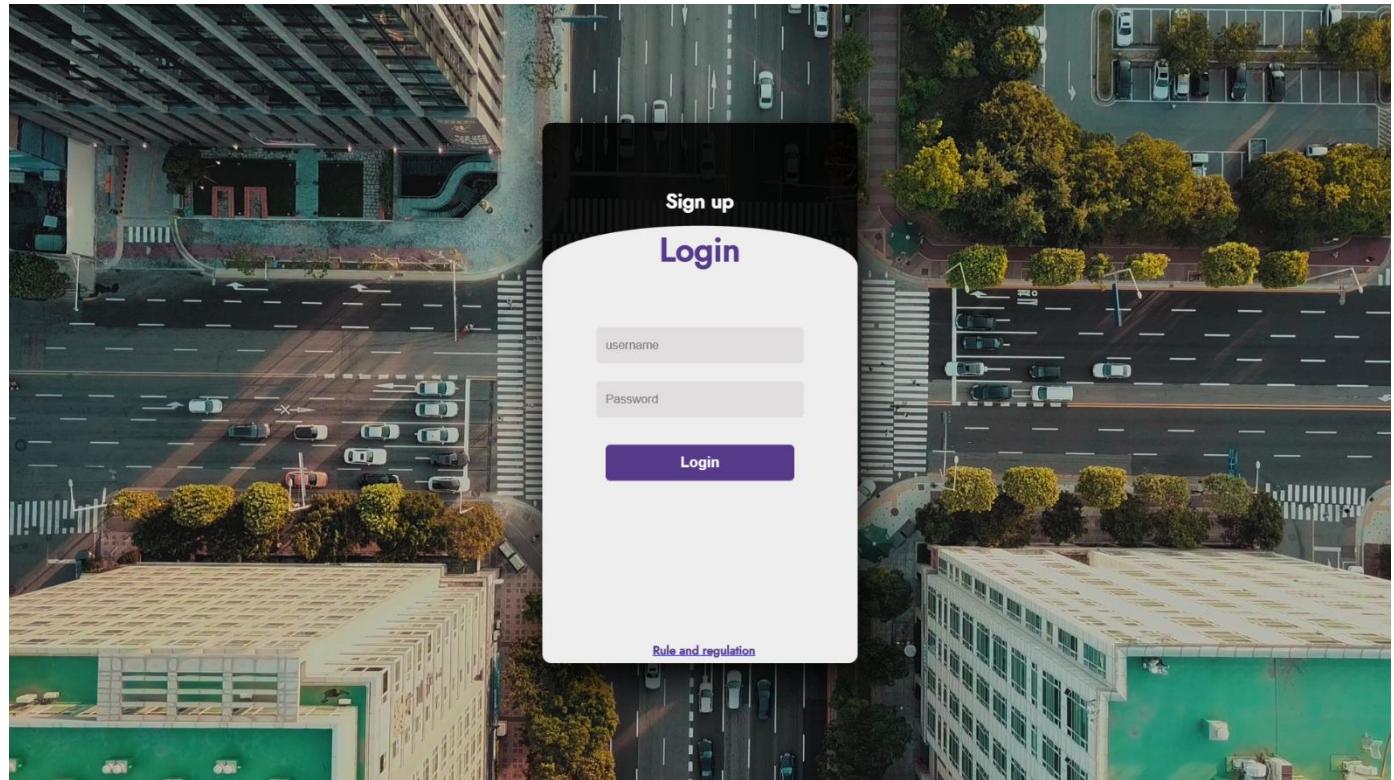
[Book Now !](#)

[Book Now !](#)

[Book Now !](#)

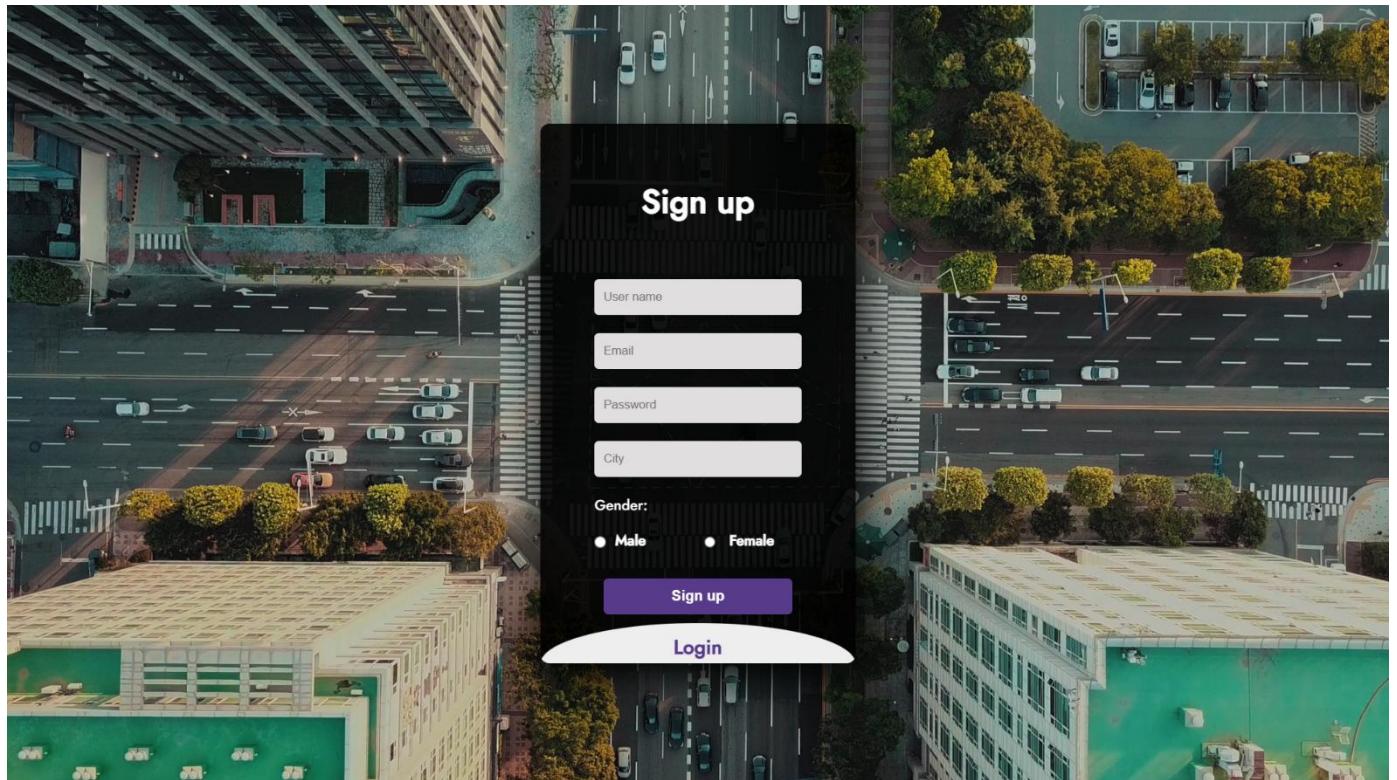
[Welcome Screen]

❖ **Sign In Screen:**



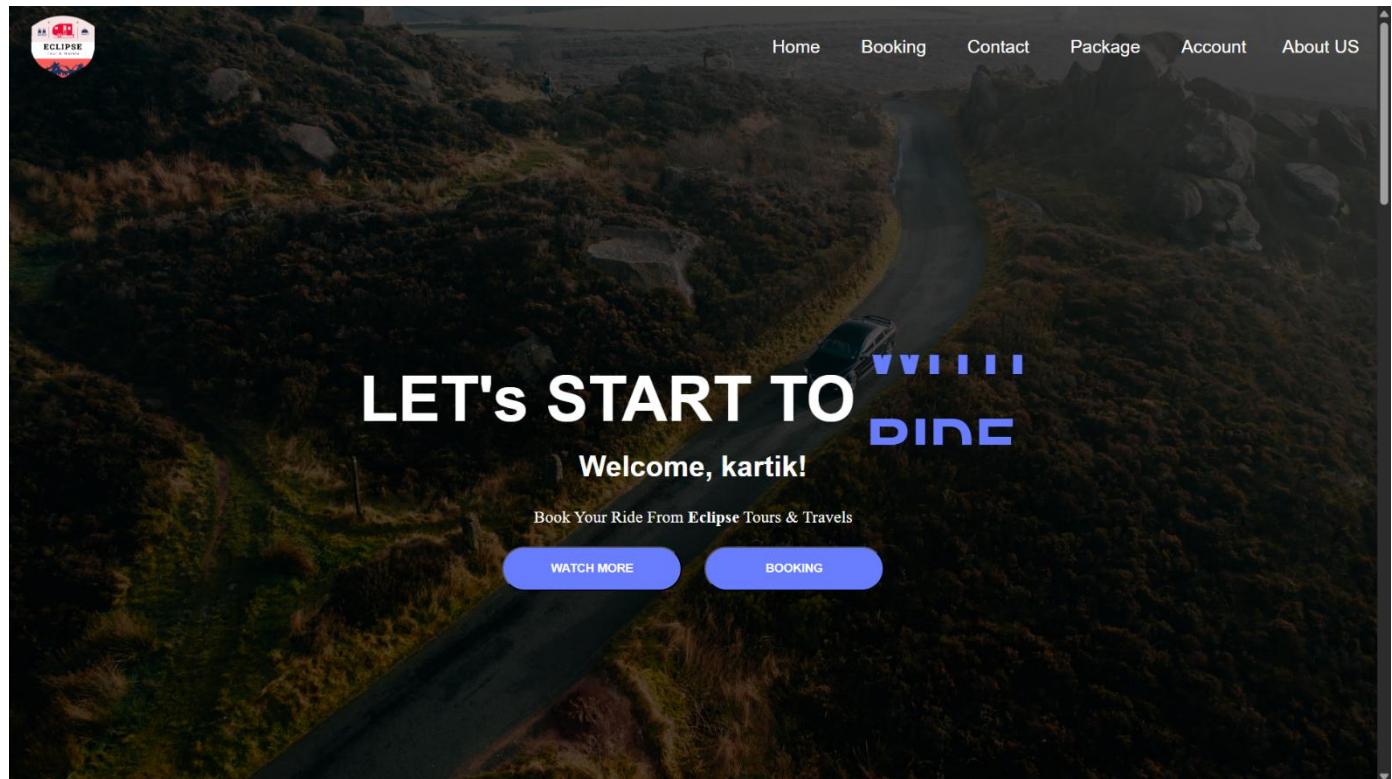
[Sign In Screen]

❖ **Sign Up Screen:**



[Sign Up Screen]

❖ Home Screen:



Why Choose Us



Variety of Car Brands

We offer a wide range of car brands to choose from.



Best Rated Guarantee

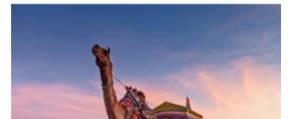
We provide the best-rated guarantee for your satisfaction.



24/7 Customer Support

Our customer support is available.

Best Place To Visit



[Home Screen]

Our Team



Safe Destination

Safety

Choose safe and reliable transportation options, whether it's public transport, reputable taxi services, or well-reviewed car rental companies

Mo. XXX-XXXX-XXX

[Contact](#)



Research the Destination

travel

Before you travel, research your destination thoroughly. Look for travel advisories and government warnings, if any, related to the place you plan to visit

Mo. XXX-XXXX-XXX

[Contact](#)



Vaccinations and Health Precautions

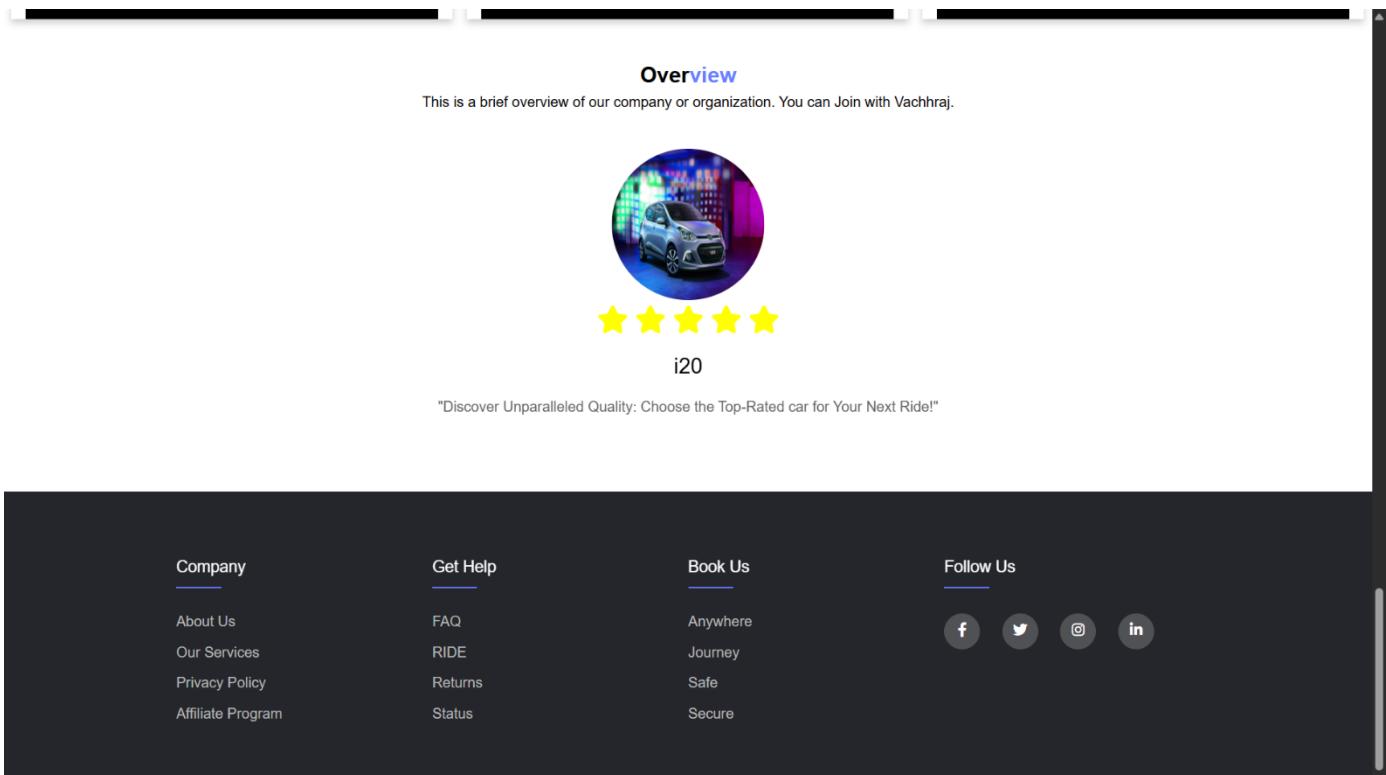
Plans

Check if there are any required or recommended vaccinations. Carry any necessary medications, prescriptions, and a first-aid kit tailored to your specific needs.

Mo. XXX-XXXX-XXX

[Contact](#)

Overview



Overview
This is a brief overview of our company or organization. You can Join with Vachhraj.



★★★★★

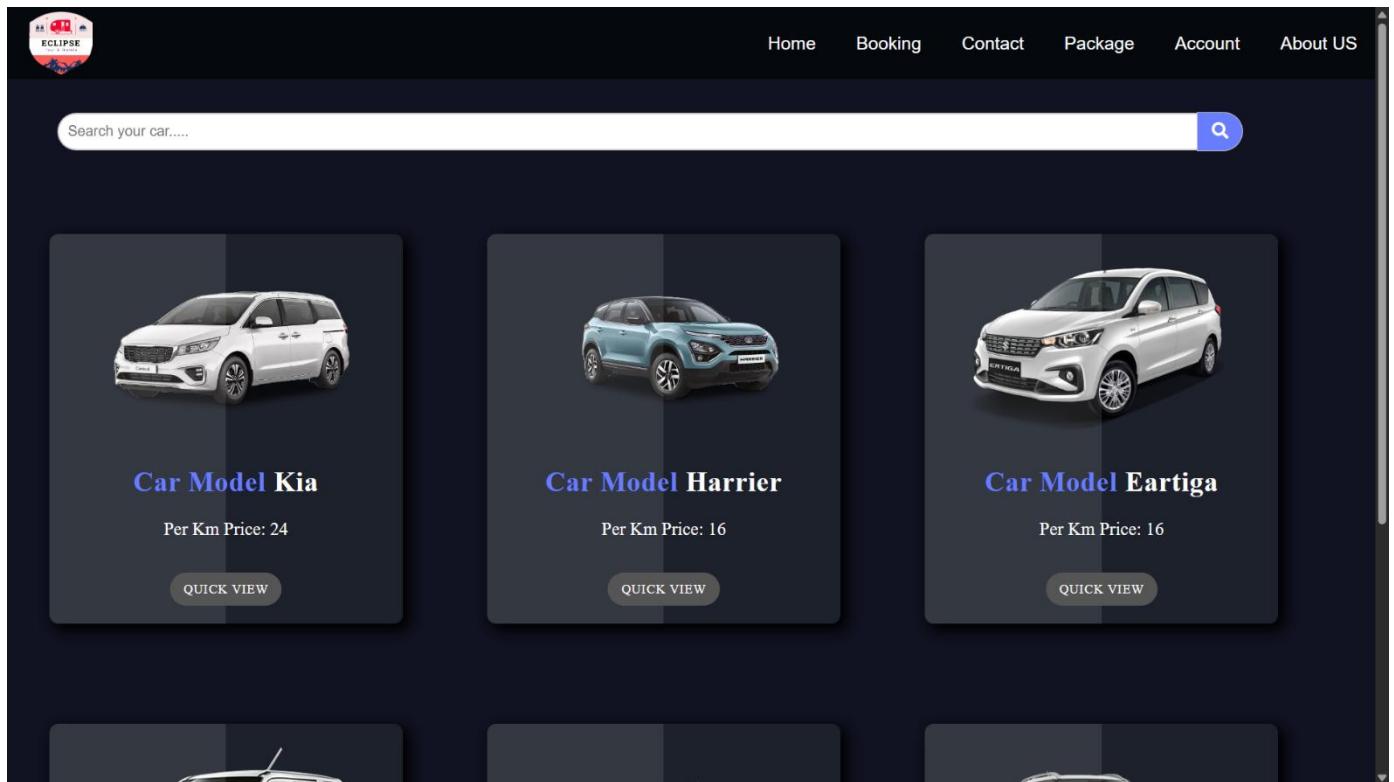
i20

"Discover Unparalleled Quality: Choose the Top-Rated car for Your Next Ride!"

Company	Get Help	Book Us	Follow Us
About Us	FAQ	Anywhere	
Our Services	RIDE	Journey	
Privacy Policy	Returns	Safe	
Affiliate Program	Status	Secure	

[Home Screen]

❖ **Booking Screen:**



The image shows a screenshot of a car rental booking website. At the top, there is a navigation bar with the SDJ International College logo on the left and menu items: Home, Booking, Contact, Package, Account, and About Us. Below the navigation bar is a search bar with the placeholder "Search your car....." and a magnifying glass icon. The main content area displays three car models in a grid format:

- Car Model Kia** (white car image): Per Km Price: 24. [QUICK VIEW](#)
- Car Model Harrier** (blue car image): Per Km Price: 16. [QUICK VIEW](#)
- Car Model Ertiga** (white car image): Per Km Price: 16. [QUICK VIEW](#)

Below the main grid, there are three smaller, partially visible car images.

Car Model Kia

Per Km Price: 24

[QUICK VIEW](#)

Car Model Harrier

Per Km Price: 16

[QUICK VIEW](#)

Car Model Eartiga

Per Km Price: 16

[QUICK VIEW](#)



Car Model Tavera

Per Km Price: 19

[QUICK VIEW](#)

Car Model Swift

Per Km Price: 13

[QUICK VIEW](#)

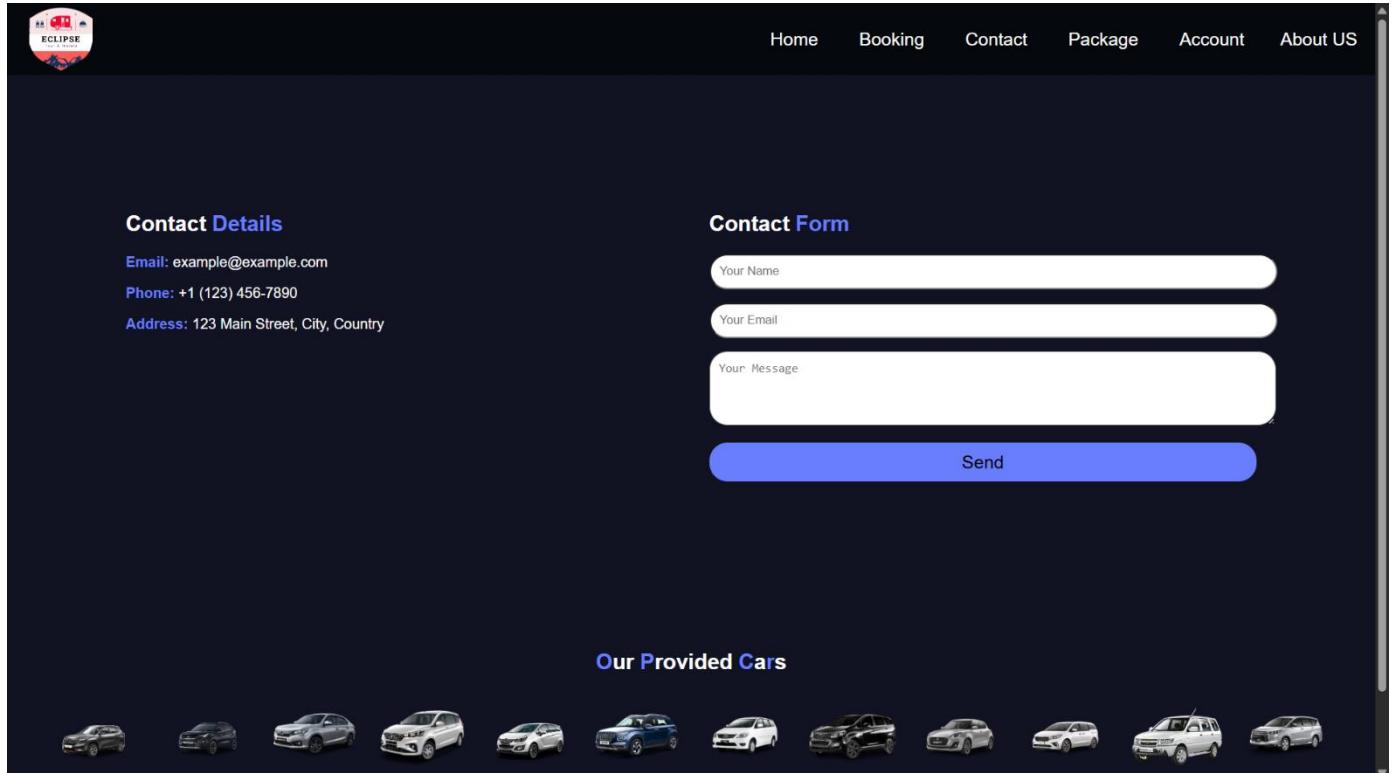
Car Model Venue

Per Km Price: 19

[QUICK VIEW](#)

[Booking Screen]

❖ **Contact Screen:**



The screenshot shows a dark-themed contact page. At the top, there is a navigation bar with a logo on the left and links for Home, Booking, Contact, Package, Account, and About US. Below the navigation bar, there are two main sections: 'Contact Details' on the left and 'Contact Form' on the right. The 'Contact Details' section contains placeholder text for Email, Phone, and Address. The 'Contact Form' section includes input fields for Your Name, Your Email, and Your Message, followed by a blue 'Send' button. At the bottom, there is a section titled 'Our Provided Cars' featuring a grid of twelve small car images.

Contact Details

Email: example@example.com
Phone: +1 (123) 456-7890
Address: 123 Main Street, City, Country

Contact Form

Your Name

Your Email

Your Message

Send

Our Provided Cars

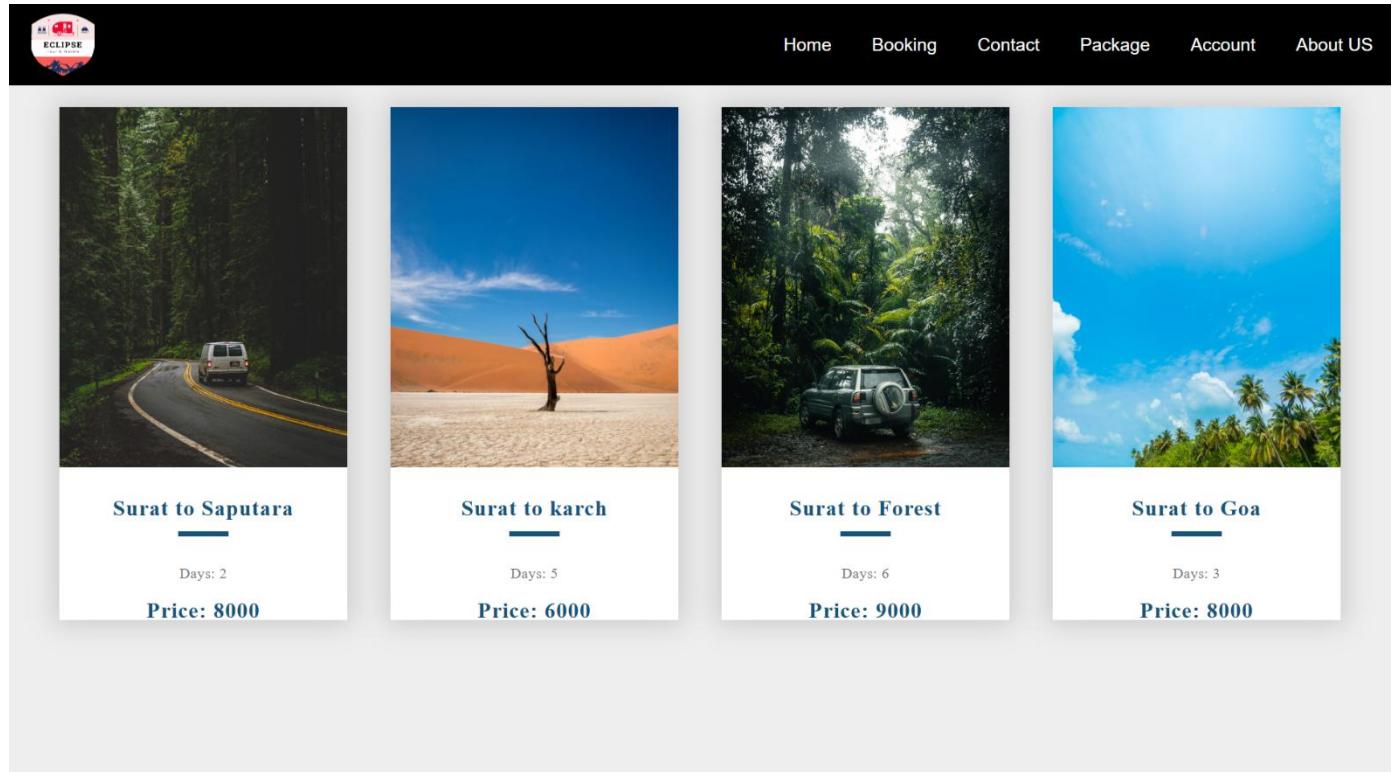


[Contact Screen]

❖ **Change Password Screen:**

[Profile Screen]

❖ Package Screen:



Surat to Saputara
Days: 2
Price: 8000

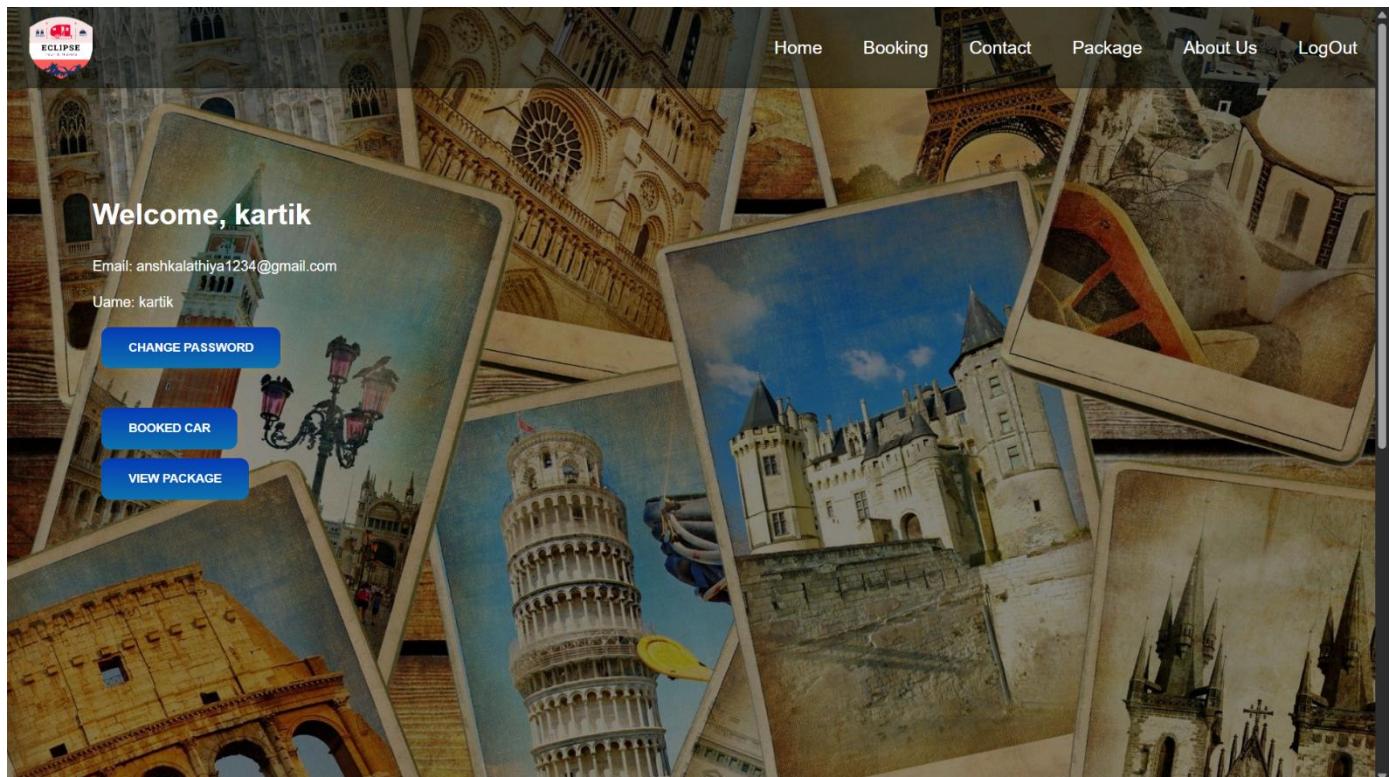
Surat to karch
Days: 5
Price: 6000

Surat to Forest
Days: 6
Price: 9000

Surat to Goa
Days: 3
Price: 8000

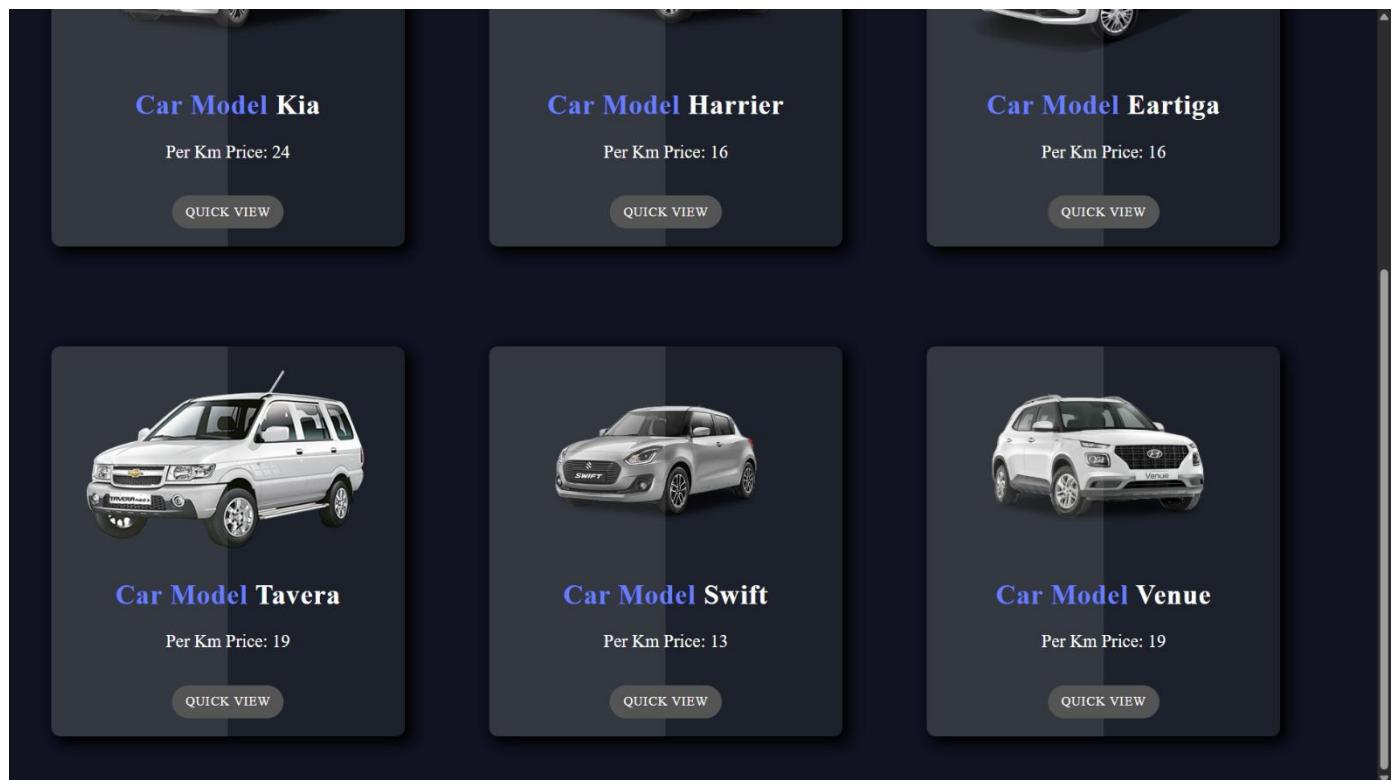
[Package Screen]

❖ **Profile Screen:**

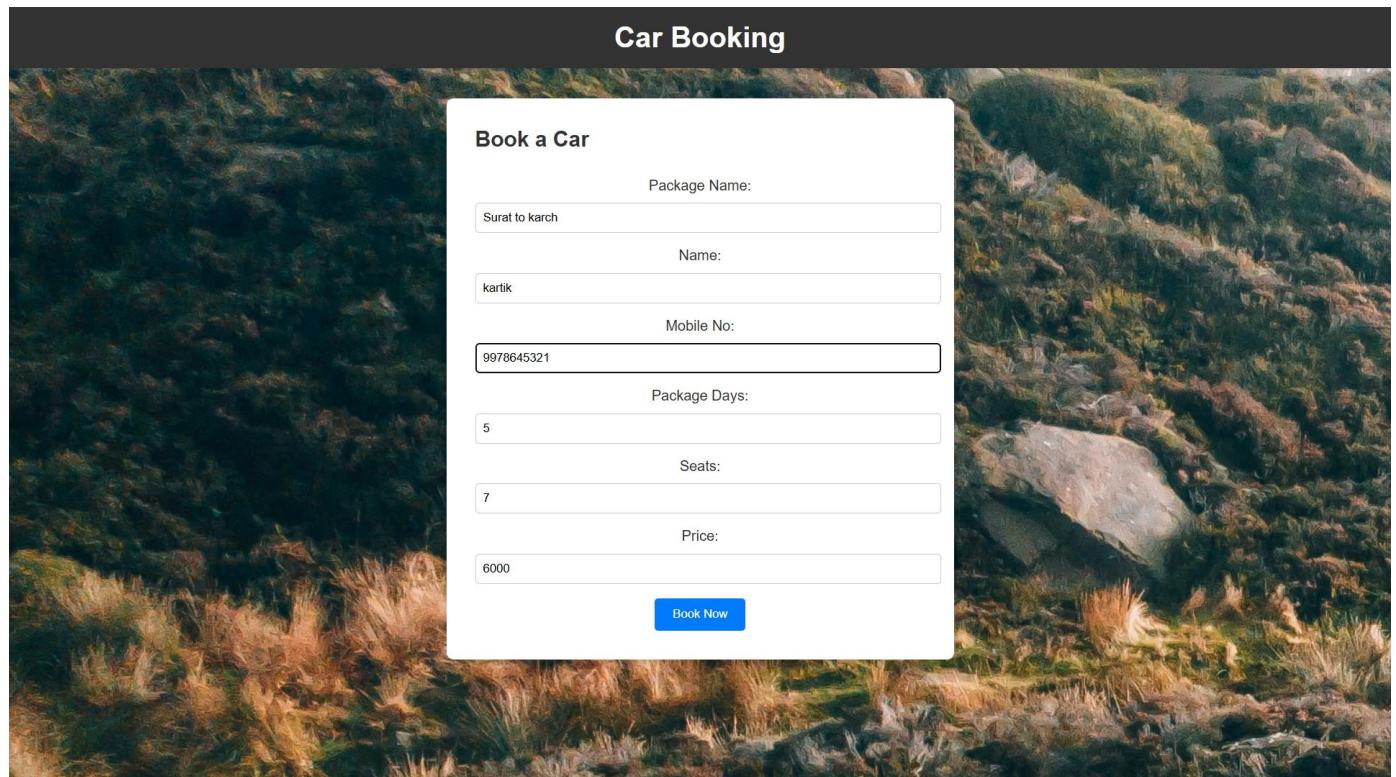


[Profile Screen]

❖ **Car Booking Screen**

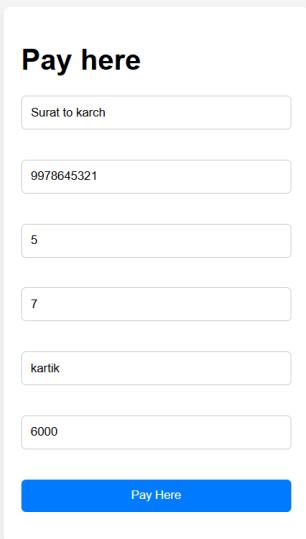


Car Model	Per Km Price
Kia	24
Harrier	16
Eartiga	16
Tavera	19
Swift	13
Venue	19



[Car Booking Screen]

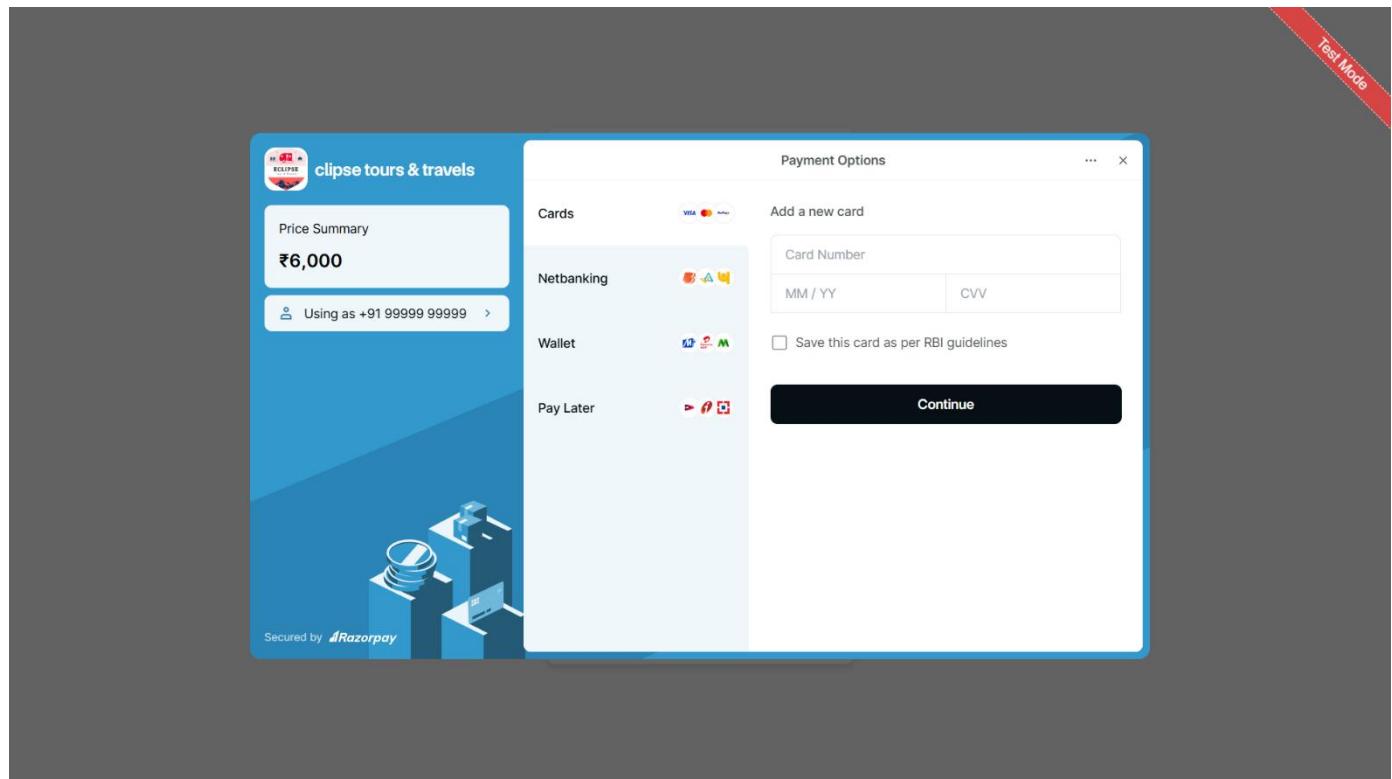
Payment Screen:



The image shows a payment screen titled "Pay here". It contains the following fields:

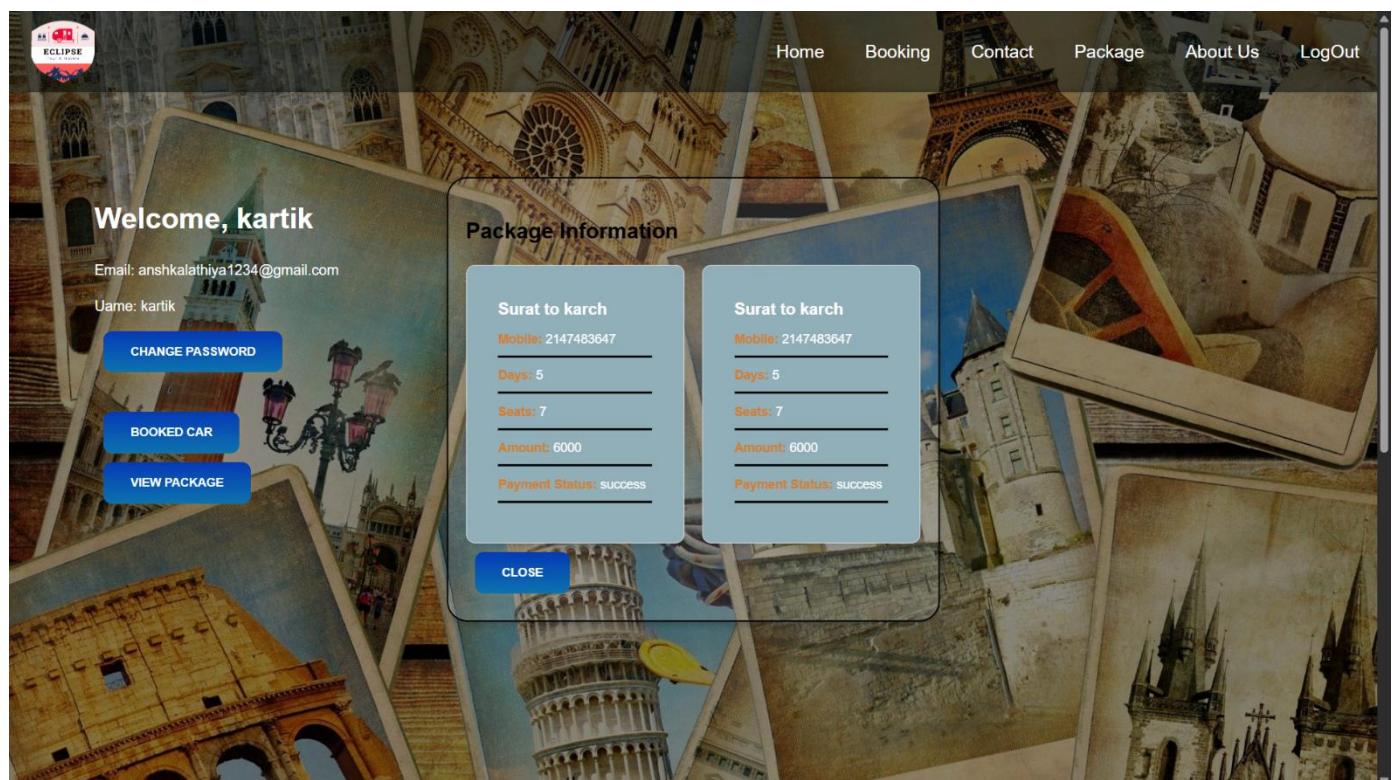
- Surat to karch
- 9978645321
- 5
- 7
- kartik
- 6000

At the bottom is a blue "Pay Here" button.



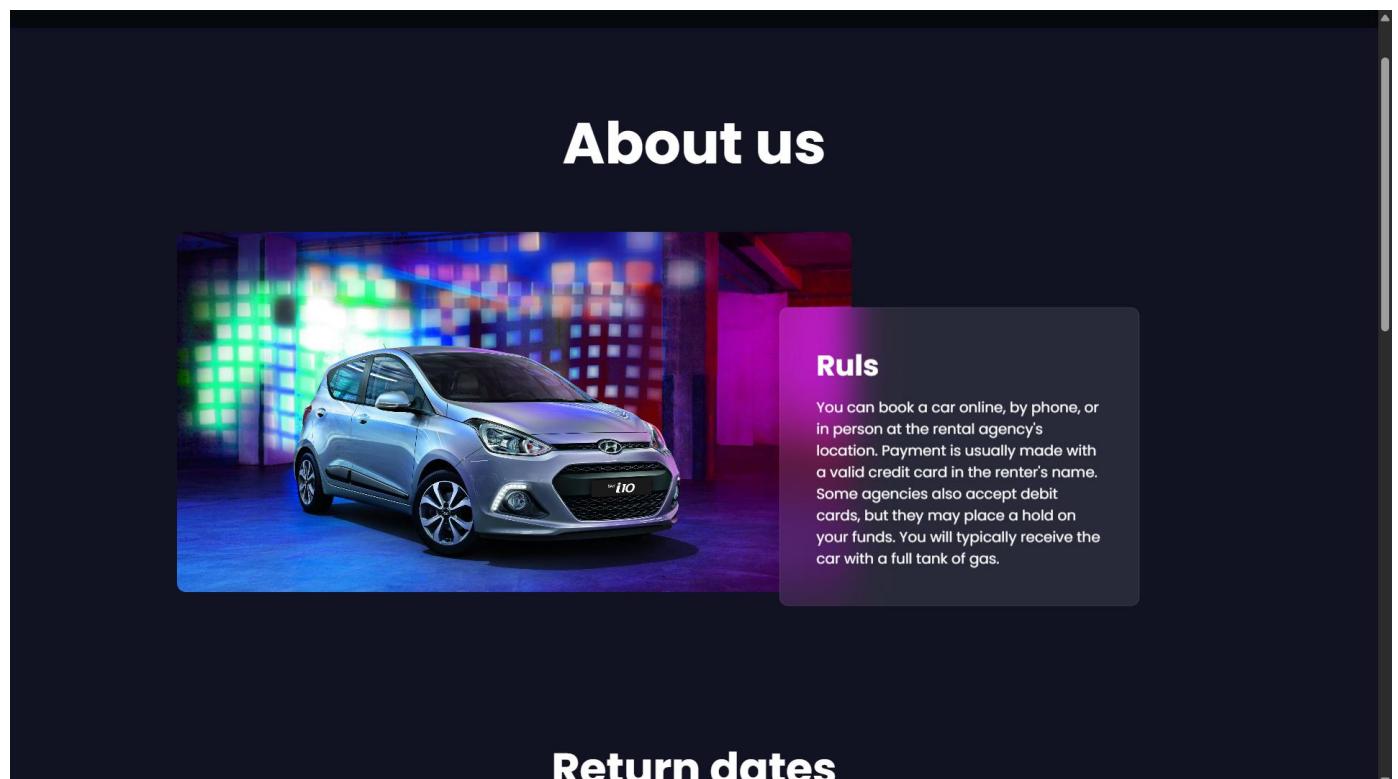
[Payment Screen]

❖ Booked Package Screen:



[Booked Package Screen]

❖ **About Us Screen:**



**Description**

When returning the rental car, make sure to do so on time and in the condition specified in the rental agreement. Agencies may charge late fees or penalties for damages. Many rental agencies offer discounts for frequent renters, loyalty program members, and various promotions. It's a good idea to look for these to save money.

Follow Us On Social Media

 Facebook



[About Us Screen]

4. Testing

4.1 Functional Testing

Functional testing is a quality assurance (QA) process and a type of black-box testing that verifies whether the application meets the specified requirements. In this approach, test cases are created based on the system's functionality without considering its internal code structure.

The testing process involves providing input and analyzing the output to ensure the system behaves as expected. Since functional testing focuses on what the system does, it helps ensure that all essential user interactions work smoothly.

- User registration and verification functions work properly.
- Sign-in and sign-out processes authenticate users and load data correctly.
- User profile updates and retrieval functions operate without issues.
- Car search and filter features display results accurately based on user preferences.
- Booking and payment processes function correctly with multiple payment options.
- Rental cost calculations based on distance and car type are accurate.
- Booking modification and cancellation work as expected.
- Notifications and reminders for upcoming rentals are sent at the correct time.
- Transaction history and invoice generation function correctly.
- Exporting rental history and booking details is seamless.
- Admin panel functions, including car management, user management, and rental approvals, work properly.
- Customer support and inquiry submission features respond correctly.
- Security features like password changes and account deletions work without issues.

4.2 Environment Testing

The test environment is set up according to the requirements of the Eclipse Car Rental Website, allowing for thorough testing of all features and functionalities.

A well-configured test environment helps prevent errors, unexpected crashes, or performance issues, ensuring a seamless user experience.

- The Eclipse Car Rental Website is tested on multiple browsers, including Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari, to ensure cross-browser compatibility.
- The website is tested on desktop, laptop, tablet, and mobile devices to verify responsive design and usability.
- The system is checked on different screen resolutions to ensure a consistent user experience across all devices.
- Functional and performance testing is conducted in various network conditions, including high-speed and low-speed connections, to test loading times and data retrieval efficiency.
- Database testing is performed on MySQL to verify that user data, bookings, and transactions are stored and retrieved accurately.
- Security testing is conducted to ensure safe online payments, secure user authentication, and protection against cyber threats.

5. Conclusion

Conclusion

The Eclipse Website is a comprehensive, user-friendly, and efficient platform designed to provide a seamless car rental experience for travelers. It offers users the ability to rent vehicles of their choice based on their needs, budget, and destination.

The website simplifies the process of booking a car by offering various rental plans, multiple vehicle options, and flexible payment methods, making it accessible and convenient for all types of users.

With a secure and scalable MySQL database, the platform efficiently stores and manages user details, bookings, transactions, and payment records. The integration of multiple online payment gateways ensures safe and hassle-free transactions.

The rental cost is calculated based on a per-kilometer rate, which varies depending on the car model, providing transparent pricing for customers. Additionally, the website supports real-time availability checks to help users find the best car options instantly.

From a technical standpoint, the website is built using PHP and MySQL, ensuring robustness, scalability, and efficient data management. It is optimized for desktop and mobile devices, ensuring a seamless experience across different platforms. Security features such as SSL encryption and secure payment processing protect users' data from potential cyber threats.

In conclusion, the Eclipse Car Rental Website is a reliable and advanced car rental solution that meets the needs of modern travelers. By offering a smooth and secure booking experience, it enhances customer satisfaction while maintaining operational efficiency. With future enhancements such as mobile app integration, AI-powered recommendations, and loyalty programs, the platform aims to further improve its services and expand its user base.

6. Bibliography

Bibliography

YouTube Links:

- <https://youtu.be/Dr3EGJ0rpt4?si=wAJfegwAHxOrqRVI>
- https://youtu.be/88F_vXq0CFA?si=DmDLr1XEvMYhCvi3
- https://youtube.com/playlist?list=PLbGui_ZYuhiihdSW-kg50d0L4or1DWvko&si=B5fOcaW1JhtuJACQ
- <https://youtube.com/playlist?list=PL7SuOAlyxrsnO4wHQ9JLD9eYUZWO8yOCh&si=G9-NQIHM5MDfELCd>

Reference Links:

- <https://chatgpt.com/c/67da6568-3fa4-800a-876b-5a2da397b512>
- <https://github.com/projectworldsofficial/Tours-and-travels-in-php>