 **SIMATS SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

**CHENNAI-602105**

**Cab booking system**

**A CAPSTONE PROJECT REPORT**

*Submitted in the partial fulfillment for the award of the degree of*

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

**Submitted by**

**B.Prem krishna (192210064)**

**K.Vinod (192211822)**

**Under the Supervision of**

**Dr.A Moorthy**

**SEP 2024**

**DECLARATION**

We, **B.Prem krishna (192210064) and K.Vinod (192211822),**students of **‘Bachelor of Engineering in Computer Science And Engineering**, Department of Computer Science and Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, hereby declare that the work presented in this Capstone Project Work entitled  **Cab booking system** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics.

**B.Prem krishna (192210064)**

**K.Vinod (192211822)**

Date:

Place:

**CERTIFICATE**

This is to certify that the project entitled **“ Cab booking”** submitted by

**B.Prem krishna (192210064) and K.Vinod (192211822)** has been carried out under our supervision. The project has been submitted as per the requirements in the current semester of B. Tech ,Computer Science and Engineering.

Teacher-in-charge

Dr. A Moorthy

**Abstract:**

The cab booking system is designed to transform and streamline the process of reserving rides, providing a comprehensive and user-friendly platform for passengers and drivers alike. This system incorporates a variety of features aimed at enhancing the convenience, efficiency, and safety of urban transportation. At its core, the system facilitates secure user registration and authentication, ensuring that both passengers and drivers can access the platform with ease and confidence.

A central feature of the system is the real-time cab search and booking functionality. Passengers can quickly locate available cabs in their vicinity, with the option to book rides immediately or schedule them for future dates. This flexibility is particularly beneficial for users with varying transportation needs, whether they require an urgent ride or are planning ahead. Additionally, the system provides accurate fare estimation, taking into account factors such as distance, time, and current traffic conditions, thereby enabling passengers to make informed decisions before confirming a booking.

The management of drivers and cabs is streamlined through dedicated modules that handle driver profiles, cab details, and ride assignments. This ensures efficient coordination and optimal utilization of resources. The system also integrates multiple payment options, including cash, credit/debit cards, and digital wallets, catering to the diverse preferences of passengers. This versatility in payment methods simplifies transactions and enhances the overall user experience.

Real-time GPS tracking is another pivotal component of the system, allowing passengers to monitor the location of their cabs and providing drivers with optimized routes. This not only reduces waiting times but also improves the overall safety and reliability of the service. Additionally, a robust notification system keeps users informed with timely alerts regarding ride status, driver arrival, and payment confirmations.

Finally, the system includes a feedback and rating mechanism, enabling passengers to share their experiences and rate the service. This feedback is invaluable for maintaining high standards of service and for making continuous improvements. The primary objective of the cab booking system is to leverage modern technology to create a seamless, reliable, and efficient transportation solution that meets the needs of contemporary urban populations. Through this system, we aim to reduce the challenges associated with urban mobility and provide a superior transportation experience for all users.

**Introduction :**

In today's fast-paced urban environment, efficient and reliable transportation is essential for maintaining the flow of daily activities. Traditional methods of hailing cabs can be time-consuming and unpredictable, often leading to delays and inconvenience for passengers. To address these challenges, the development of a comprehensive cab booking system emerges as a crucial solution, offering a modern, digital approach to ride-hailing that aligns with the demands of contemporary society.

A cab booking system leverages advanced technology to provide a seamless and user-friendly platform for both passengers and drivers. By facilitating easy access to transportation services through mobile applications and web interfaces, this system enhances the convenience and efficiency of booking rides. Users can quickly find and book cabs in real time, eliminating the uncertainties associated with traditional methods. This not only saves time but also ensures a more reliable and stress-free travel experience.

The core of the cab booking system is its ability to connect passengers with drivers in a secure and efficient manner. Through features such as real-time GPS tracking, fare estimation, and multiple payment options, the system offers a comprehensive solution that caters to the diverse needs of urban commuters. Passengers can monitor the location of their cabs, receive accurate fare calculations based on distance and traffic conditions, and choose from a variety of payment methods. These functionalities collectively contribute to a more transparent and user-centric service.

Moreover, the cab booking system benefits drivers by providing a structured and organized platform for managing ride requests. Drivers receive ride assignments, optimized routes, and real-time navigation assistance, which help in maximizing their efficiency and earnings. The system also includes mechanisms for driver registration, profile management, and performance tracking, ensuring that high standards of service are maintained.

In addition to enhancing the ride-hailing experience, the cab booking system plays a significant role in improving urban mobility. By reducing waiting times, optimizing routes, and facilitating smoother transactions, the system contributes to a more sustainable and efficient transportation network. As cities continue to grow and evolve, the need for innovative solutions like the cab booking system becomes increasingly evident. This technology-driven approach not only meets the current demands of urban transportation but also sets the foundation for future advancements in the field.

In summary, the introduction of a cab booking system marks a significant step forward in revolutionizing urban transportation. By integrating cutting-edge technology and user-centric design, the system provides a reliable, efficient, and convenient solution for both passengers and drivers. As a result, it holds the potential to transform the way people navigate their cities, offering a modern alternative to traditional ride-hailing methods and paving the way for a smarter, more connected urban mobility ecosystem.

**Problem Description :**

In the contemporary digital era, the need for efficient and reliable transportation solutions has never been greater. The Car Booking System project is designed to address this need by developing a web-based application that allows users to book cars for various purposes seamlessly. Leveraging internet programming technologies, this project ensures that the application is accessible, scalable, and user-friendly, providing a comprehensive solution for modern transportation challenges.

The primary goal of the Car Booking System is to create a user-centric platform that facilitates easy and secure car bookings. The system will feature an intuitive interface, enabling users to search for available cars, book rides, manage their reservations, and provide feedback with ease. This focus on user experience is paramount, as the system aims to reduce the complexities traditionally associated with car rentals and ride-hailing services.

Security is a critical component of the Car Booking System. The project will implement robust authentication processes to ensure that user data is protected. Secure payment gateways will be integrated to safeguard financial transactions, providing users with peace of mind when booking and paying for rides. By prioritizing security, the system aims to build trust and reliability among its users.

Efficient resource management is another key objective of the project. The system will include features for managing car availability, assigning drivers, and planning routes effectively. This will not only maximize the utilization of available resources but also ensure that users experience minimal waiting times and optimal service. Real-time tracking and fare estimation will further enhance the user experience by providing transparency and convenience.

User feedback is invaluable for maintaining high standards of service. The Car Booking System will incorporate mechanisms for collecting and utilizing user reviews and ratings. This feedback will help in continuously improving the service, ensuring that it meets the evolving needs and expectations of users. By fostering a culture of continuous improvement, the system aims to stay ahead in the competitive landscape of car booking services.In summary, the Car Booking System project is poised to revolutionize the way users book and manage car rides. By integrating cutting-edge internet programming technologies and focusing on user-centric design, security, and efficient resource management, the system will offer a reliable, efficient, and convenient solution for modern transportation needs.

**Tool Description:**

**HTML5 :**

HTML5 plays a crucial role in the development of modern web-based applications, including cab booking systems. As the latest version of the HyperText Markup Language, HTML5 introduces a range of new features and capabilities that significantly enhance the functionality, performance, and user experience of web applications. In the context of a cab booking system, HTML5 provides the foundation for building a robust, interactive, and responsive user interface.

CSS3:

CSS3 introduces media queries, which allow developers to apply different styles based on the characteristics of the device or screen. This is essential for ensuring that the cab booking system is accessible and user-friendly on desktops, tablets, and smartphones. Media queries help in adjusting layouts, font sizes, and other visual elements to provide a consistent experience across different devices.

Cab Booking API:

In a cab booking system, APIs play a crucial role in ensuring a seamless user experience. The **Geolocation API** determines the user's location for pickup and drop-off, while **Maps APIs** (like Google Maps or Mapbox) display routes and track cabs. **Payment Gateway APIs** (such as Stripe or PayPal) handle secure financial transactions, and **Authentication APIs** (like OAuth or Firebase Authentication) manage user login and registration. Additionally, **SMS/Notification APIs** (such as Twilio or SendGrid) send updates and alerts, **Ride Management APIs** (e.g., Uber or Lyft) handle ride requests and assignments, and **Pricing and Fare Calculation APIs** estimate the cost of rides. Together, these APIs facilitate a smooth, interactive, and efficient cab booking process.

**Bootstrap:**

In cab booking systems, "bootstrap" refers to the foundational setup required to launch the service. This includes creating a responsive user interface using tools like Bootstrap, developing backend logic for ride management, setting up databases for storing user and ride data, integrating APIs for functionalities such as location tracking and payments, and implementing user authentication. The bootstrap phase ensures that the essential features are functional and stable before expanding the system further.

**Github :**

GitHub serves as the version control system for the Weather Application project. It facilitates collaboration among team members, enabling efficient tracking of changes, code reviews, and issue management. By using GitHub, the project maintains a well-organized and documented codebase, ensuring that all development efforts are coordinated and transparent.

**Visual Studio Code :**

In Visual Studio Code, a simple implementation for a cab booking system can be structured with two key operations: book a ride and view ride history. For the booking operation, you might use JavaScript to handle user input for ride details (e.g., pickup location, drop-off location) and send a request to a backend service to process the booking. For viewing ride history, you would retrieve past booking records from a database and display them in a user-friendly format. Both operations can be implemented using Node.js for backend logic and HTML/CSS for frontend interfaces, with Visual Studio Code providing an integrated environment for coding, debugging, and version control.

**Approach :**

**Requirement Analysis**

Requirement analysis for a cab booking system involves defining essential functionalities such as user registration and login, ride booking, real-time tracking, and viewing ride history. Drivers need features for managing ride requests, tracking earnings, and handling profile details. Administrative requirements include managing user and driver accounts, and generating analytics and reports. The system must ensure secure data handling, robust backend infrastructure, scalable performance, and seamless integration with external APIs for location tracking and payments. This comprehensive approach ensures the system is efficient, user-friendly, and meets the needs of all stakeholders.

**Design**

The design for a car booking system centers around a user-friendly interface and a robust backend. Users and drivers interact through a web or mobile application where users can book rides by entering pickup and drop-off locations, selecting vehicle types, and choosing payment methods. The backend handles ride requests, real-time tracking, and payment processing, while integrating with APIs for location services. The system features secure user and driver accounts, manages booking history, and provides administrative tools for user and driver management, along with analytics. Scalability and performance are key, ensuring the system can handle high volumes of concurrent requests efficiently.

**API Integration**

API integration for a cab booking system involves incorporating location services like Google Maps for route optimization and real-time tracking, payment gateways such as Stripe or PayPal for secure transactions, and notification services like Twilio or SendGrid for sending updates and confirmations. Additionally, integrating geocoding APIs for address conversion, driver matching services for efficient ride allocation, and analytics APIs for performance tracking ensures a comprehensive and seamless operation, enhancing both user and driver experiences.

**Development**

Development for a cab booking system involves designing and implementing a seamless user interface for booking and managing rides, integrating backend services for handling ride requests, real-time tracking, and payment processing. This includes setting up databases to store user and ride information, developing APIs for location services and notifications, and ensuring robust security measures for data protection. The development process also encompasses creating administrative tools for managing users and drivers, as well as implementing performance optimization to handle high traffic and scale efficiently.

**Testing**

Testing for a cab booking system involves verifying that all features such as ride booking, real-time tracking, and payment processing function correctly, ensuring the interface is user-friendly and intuitive. Performance testing is conducted to handle peak loads efficiently, while security testing focuses on protecting user data and preventing unauthorized access. Integration testing checks that APIs for maps, payments, and notifications work seamlessly, compatibility testing ensures the system functions across various devices and browsers, and regression testing confirms that updates do not introduce new issues. This comprehensive approach ensures a reliable, secure, and user-friendly system.

**Deployment**

Deployment in a car booking system involves preparing the production environment, configuring servers and databases, and deploying the application code. This includes migrating data, setting up monitoring tools to track system performance and errors, and ensuring proper access controls. Once live, a final round of testing ensures everything functions correctly. Ongoing support and maintenance are essential to address any issues and keep the system stable and secure for users and drivers.

**Feedback and Iteration**

Feedback and iteration in a car booking system involve collecting user and driver feedback on their experiences, identifying areas for improvement, and making necessary updates. After deployment, users provide insights into functionality, usability, and any issues they encounter, while drivers offer feedback on the system's operational aspects. This feedback is analyzed to prioritize enhancements and fix bugs. Iterative updates are then implemented to refine features, improve performance, and address any identified problems, ensuring the system continuously evolves to better meet user needs and maintain high service quality.

**IMPLEMENTATION**

**HTML CODE**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Cab Booking System</title>

<link rel="stylesheet" href="design.css">

</head>

<body>

<header id="top">

<nav>

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#login">Login</a></li>

<li><a href="#register">Register</a></li>

<li><a href="#driver">Driver</a></li>

<li><a href="#booking">Booking</a></li>

<li><a href="#payment">Payment</a></li>

<li><a href="#feedback">Feedback</a></li>

</ul>

</nav>

</header>

<main>

<section id="home">

<div class="Container">

<img src="cab.jpg" alt="Cab Booking"/>

<h2>Welcome to Cab Booking System</h2>

</div>

</section>

<section id="register">

<div class="box">

<center>

<h2>Register</h2>

</center>

<form action="registration.php" method="post" id="register\_form">

Name: <input type="text" name="name" required placeholder="Enter your name"/>

<br><br>

Email: <input type="email" name="email" required placeholder="Enter your email"/>

<br><br>

Password: <input type="password" name="password" required placeholder="Enter your password"/>

<br><br>

Confirm Password: <input type="password" name="confirm\_password" required placeholder="Confirm your password"/>

<br><br>

Phone Number: <input type="text" name="phone\_number" required placeholder="Enter your phone number"/>

<br><br>

Address: <input type="text" name="address" required placeholder="Enter your address"/>

<br><br>

Role:

<select name="role" required>

<option value="customer">Customer</option>

<option value="driver">Driver</option>

</select>

<br><br>

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

</form>

</div>

</section>

<section id="login">

<div class="box">

<center>

<h2>Login</h2>

</center>

<form action="login.php" method="post" id="login\_form">

Username: <input type="text" name="username" id="username" required placeholder="Enter your username"/>

<br><br>

Password: <input type="password" name="password" id="password" required placeholder="Enter your password"/>

<br><br>

<input type="submit" value="Login">

</form>

</div>

</section>

<section id="driver">

<div class="box">

<center>

<h2>Driver Registration</h2>

</center>

<form action="driver\_register.php" method="post" id="driver\_form">

Name: <input type="text" name="name" required placeholder="Enter your name"/>

<br><br>

License Number: <input type="text" name="license\_number" required placeholder="Enter your license number"/>

<br><br>

Vehicle Details: <input type="text" name="vehicle\_details" required placeholder="Enter your vehicle details"/>

<br><br>

Phone Number: <input type="text" name="phone" required placeholder="Enter your phone number"/>

<br><br>

Address: <input type="text" name="address" required placeholder="Enter your address"/>

<br><br>

<input type="submit" value="Register as Driver">

</form>

</div>

</section>

<section id="booking">

<div class="box">

<center>

<h2>Book a Cab</h2>

</center>

<form action="book.php" method="post" id="booking\_form">

Pickup Location: <input type="text" name="pickup" required placeholder="Enter pickup location"/>

<br><br>

Drop Location: <input type="text" name="drop" required placeholder="Enter drop location"/>

<br><br>

Date: <input type="date" name="date" required/>

<br><br>

Time: <input type="time" name="time" required/>

<br><br>

Vehicle Type:

<select name="vehicle" required>

<option value="car">Car</option>

<option value="van">Van</option>

<option value="truck">Truck</option>

</select>

<br><br>

Number of Passengers: <input type="number" name="passengers" required min="1" max="10"/>

<br><br>

<input type="submit" value="Book Now">

</form>

</div>

</section>

<section id="payment">

<div class="box">

<center>

<h2>Payment</h2>

</center>

<form action="payment.php" method="post" id="payment\_form">

Card Number: <input type="text" name="card\_number" required placeholder="Enter your card number"/>

<br><br>

Expiry Date: <input type="text" name="expiry\_date" required placeholder="MM/YY"/>

<br><br>

CVV: <input type="text" name="cvv" required placeholder="Enter CVV"/>

<br><br>

Name on Card: <input type="text" name="name\_on\_card" required placeholder="Enter your name on the card"/>

<br><br>

<input type="submit" value="Pay Now">

</form>

</div>

</section>

<section id="feedback">

<div class="box">

<center>

<h2>Feedback</h2>

</center>

<form action="feedback.php" method="post" id="feedback\_form">

Name: <input type="text" name="name" required placeholder="Enter your name"/>

<br><br>

Email: <input type="email" name="email" required placeholder="Enter your email"/>

<br><br>

Feedback: <textarea name="feedback" required placeholder="Enter your feedback" rows="5" cols="40"></textarea>

<br><br>

<input type="submit" value="Submit Feedback">

</form>

</div>

</section>

</main>

</body>

</html>

**CSS code :**

/\* Resetting default padding and margin \*/

\* {

padding: 0;

margin: 0;

box-sizing: border-box;

}

/\* Basic styles for the header \*/

header {

background-color: cyan;

height: 45px;

}

header nav ul {

display: flex;

justify-content: center;

align-items: center;

list-style: none;

padding: 0;

margin: 0;

}

header nav ul li {

margin-left: 20px;

}

header a {

text-decoration: underline;

color: brown;

font-weight: bold;

}

/\* Main section styles \*/

main {

padding: 20px;

text-align: center;

}

main h1 {

font-size: 2.5rem;

margin-bottom: 10px;

}

main p {

font-size: 1.25rem;

margin-top: 10px;

}

/\* Form styles \*/

form {

display: flex;

flex-direction: column;

align-items: center;

max-width: 600px;

margin: 0 auto;

}

form input, form select, form button {

width: 100%;

max-width: 400px;

margin-bottom: 10px;

padding: 10px;

border: 1px solid #ccc;

border-radius: 5px;

}

form button {

background-color: #4CAF50;

color: white;

border: none;

cursor: pointer;

}

form button:hover {

background-color: #45a049;

}

/\* Styling for individual pages \*/

#home {

background-color: pink;

}

#register {

background-color: yellow;

}

#login {

background-color: red;

}

#booking {

background-color: lightblue;

}

#payment {

background-color: lightgreen;

}

/\* Footer styles \*/

footer {

line-height: 40px;

display: flex;

justify-content: center;

font-size: 1rem;

background-color: #f1f1f1;

padding: 10px;

}

**PHP Code :**

<?php

$servername = "localhost"; // Change this to your server name

$username = "root"; // Change this to your database username

$password = ""; // Change this to your database password

$dbname = "cab booking"; // Change this to your database name

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

else

{

echo "Connected Successfully";

}

// Check if form is submitted

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$Name = $\_POST['name'];

$Email = $\_POST['email'];

$Password = $\_POST['password'];

$Phone\_number= $\_POST['phone\_number'];

$Address = $\_POST['address'];

$Role = $\_POST['role'];

$stmt = $conn->prepare("INSERT INTO registration VALUES (?, ?, ?, ?, ?, ?)");

$stmt->bind\_param("sssiss", $Name, $Email, $Password, $Phone\_number, $Address, $Role);

// Execute the statement

if ($stmt->execute()) {

echo "Registration successful!";

} else {

echo "Error: " . $stmt->error;

}

// Close the statement

$stmt->close();

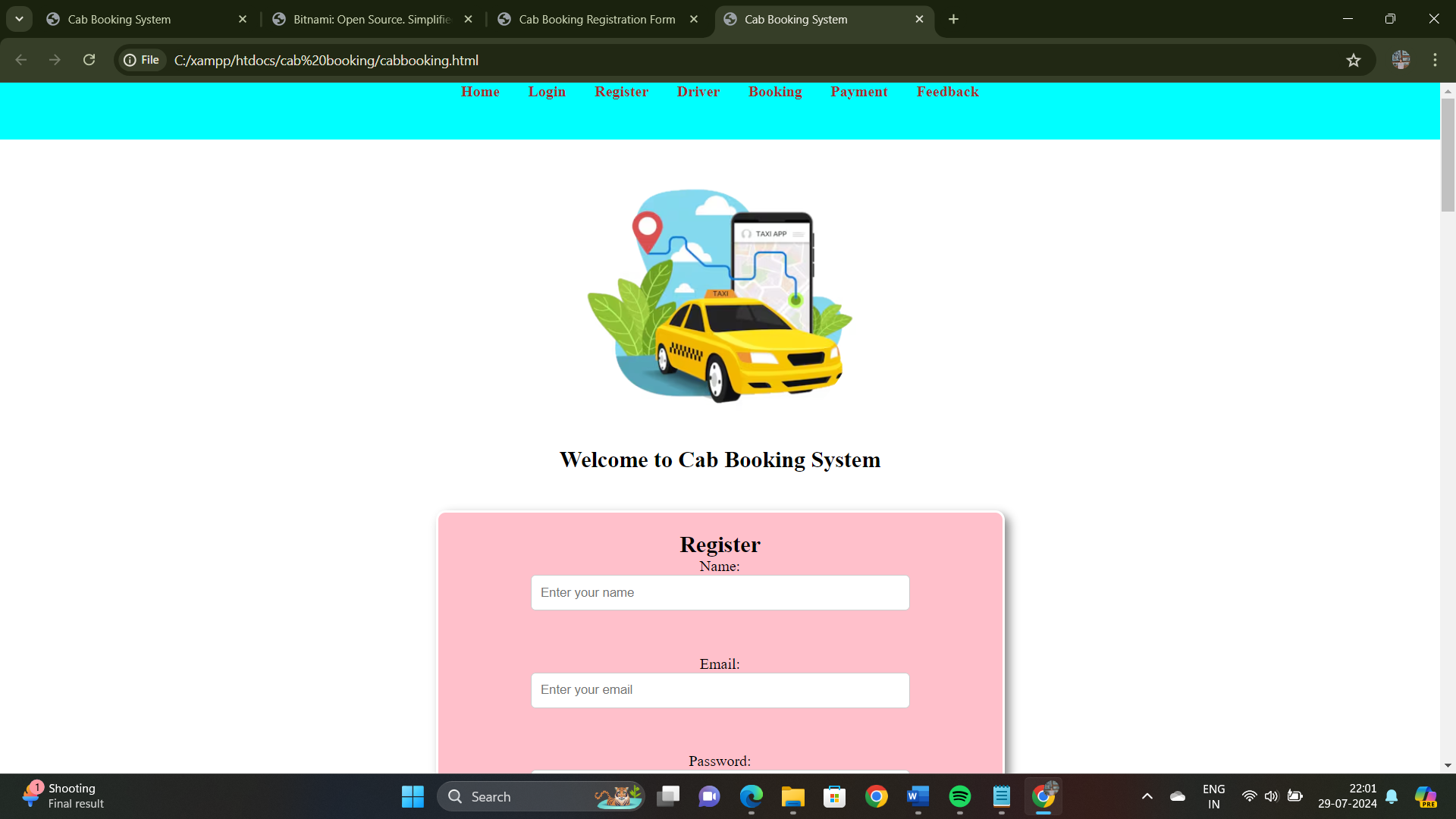
}

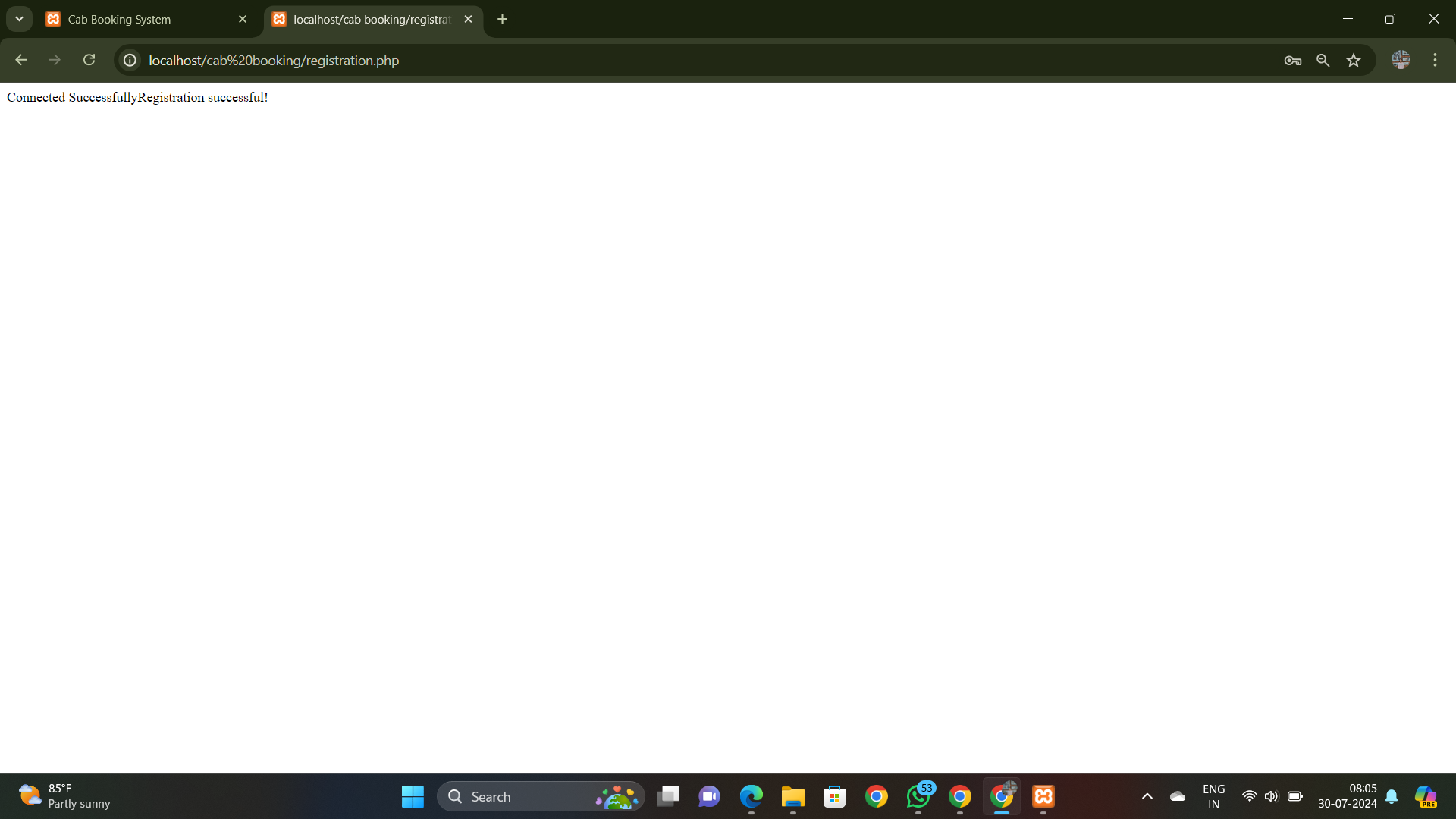
// Close the connection

$conn->close();

?>

**Output :**





**Conclusion :**

**Project Description**

The cab booking system will significantly enhance user experience by providing a seamless and intuitive interface for booking and tracking rides, leading to increased satisfaction and convenience. It will optimize resource utilization by efficiently managing car availability and driver assignments, thus minimizing wait times and maximizing operational efficiency. The system’s secure payment processing and robust authentication will improve safety and build trust, while its scalability will accommodate future expansions. Additionally, actionable insights from data analytics will inform business decisions and marketing strategies, contribute to a positive brand reputation, and support sustainability efforts by reducing fuel consumption and environmental impact.

**Practical Application**

The cab booking system's practical application lies in its ability to provide an efficient and user-friendly platform for urban transportation. It allows users to quickly book rides, track their cabs in real time, and make secure payments, all from their smartphones. This system improves operational efficiency by optimizing driver assignments and reducing idle times, leading to faster and more reliable service. It also enhances safety through secure authentication and payment processes, and offers valuable data insights for ongoing improvements. By integrating with mapping and geolocation technologies, the system streamlines urban mobility, making it a crucial tool for modern transportation networks.

**Future Improvements**

Future improvements for the cab booking system could include integrating advanced AI for demand prediction and dynamic pricing, expanding the fleet to include electric and autonomous vehicles, and enhancing integration with public transportation for multi-modal journey planning. Personalization features could be refined using user data, while safety could be bolstered with in-vehicle cameras and emergency alerts. Incorporating voice and gesture controls, leveraging blockchain for transaction transparency, and integrating augmented reality for navigation can further enhance user experience. Additionally, advanced analytics and reporting tools, along with sustainability initiatives, such as carbon offset options and incentives for eco-friendly vehicles, will ensure the system remains cutting-edge, efficient, and aligned with future transportation needs.