THIRUMALAI ENGINEERING COLLEGE

The Right Place to Enrich Your Career...

Approved by AICTE, New Delhi & Affiliated to Anna University

An ISO 9001 : 2008 Certified Institution

SMART PARKING

PHASE 4: DEVELOPMENT PART 2

A Project report submitted in partial fulfilment of the requirements for the degree of B.Tech in Information technology

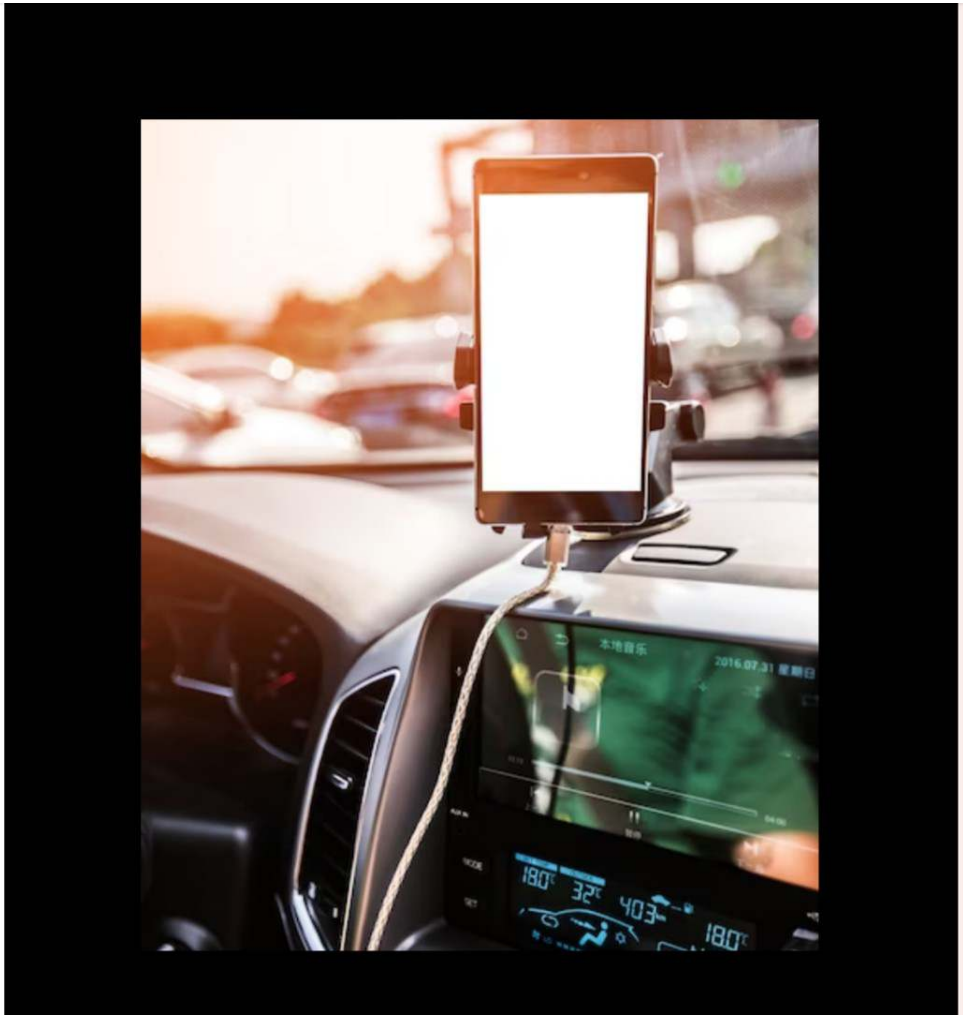
BY

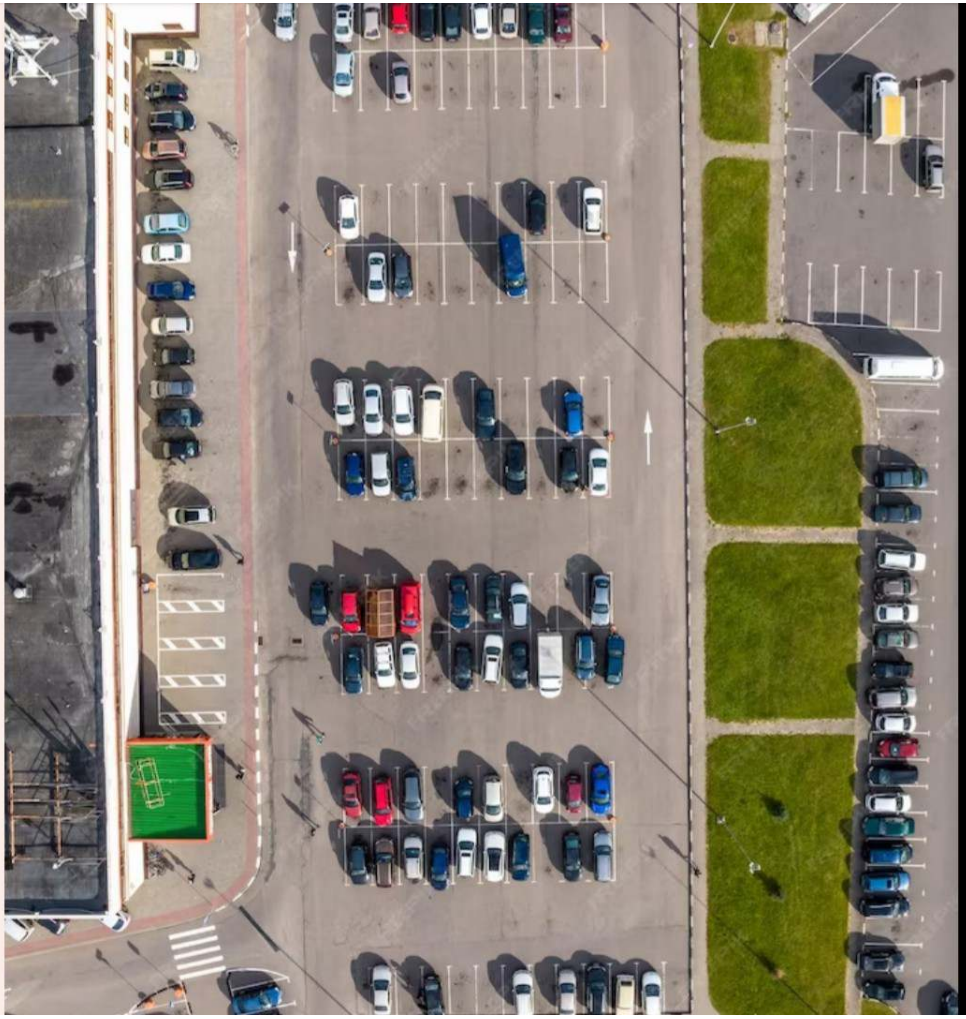
BOOBALAN.M (513221205003)

Creating a Real-time Parking Availability App using Flutter: Leveraging Mobile App Development Frameworks to Interact with Raspberry Pi

Introduction

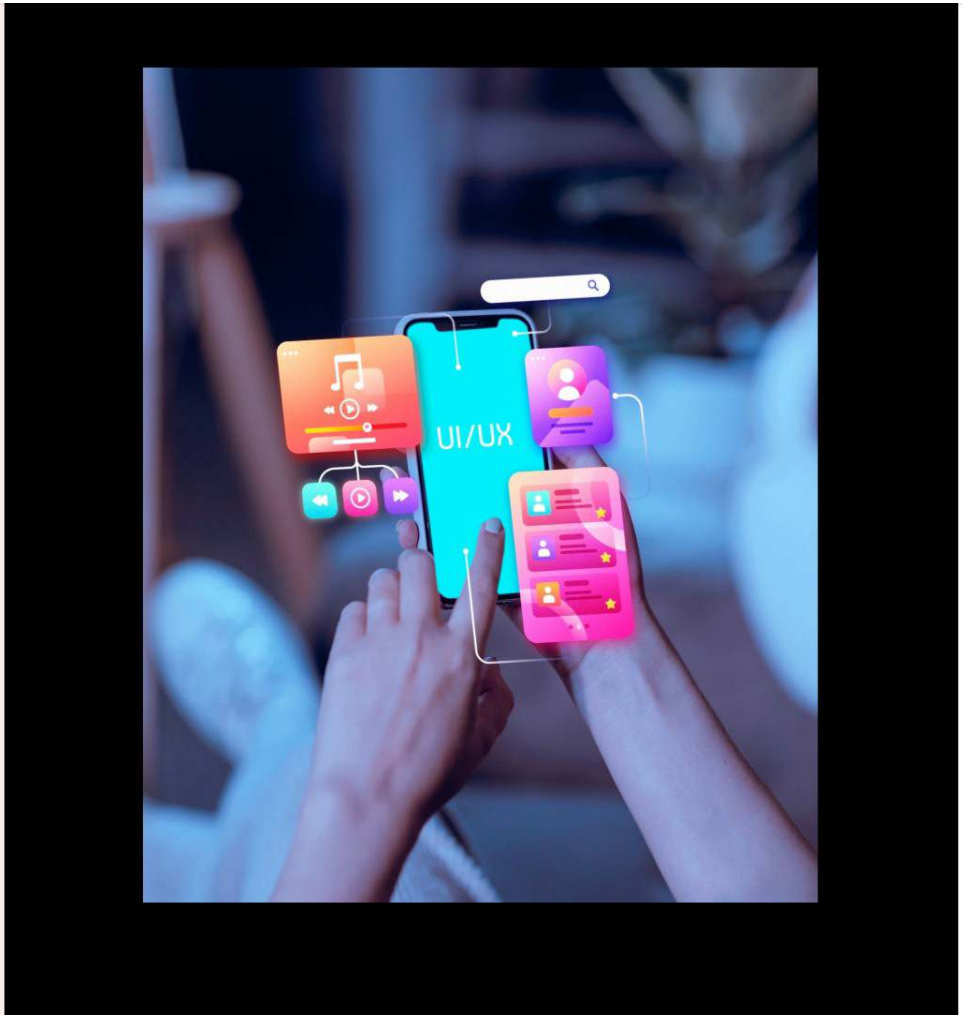
Welcome to the presentation on creating a real-time parking availability app using Flutter and Raspberry Pi. This presentation will explore the process of leveraging mobile app development frameworks to interact with Raspberry Pi for building an efficient parking app. We will discuss the benefits, challenges, and implementation details of this innovative solution.





Understanding the Problem

Parking availability is a common issue in urban areas. *Limited parking spaces* and lack of real-time information often lead to frustration and wasted time for drivers. This app aims to address these challenges by providing a user-friendly interface that shows **real-time parking availability** using data from Raspberry Pi sensors installed in parking lots.

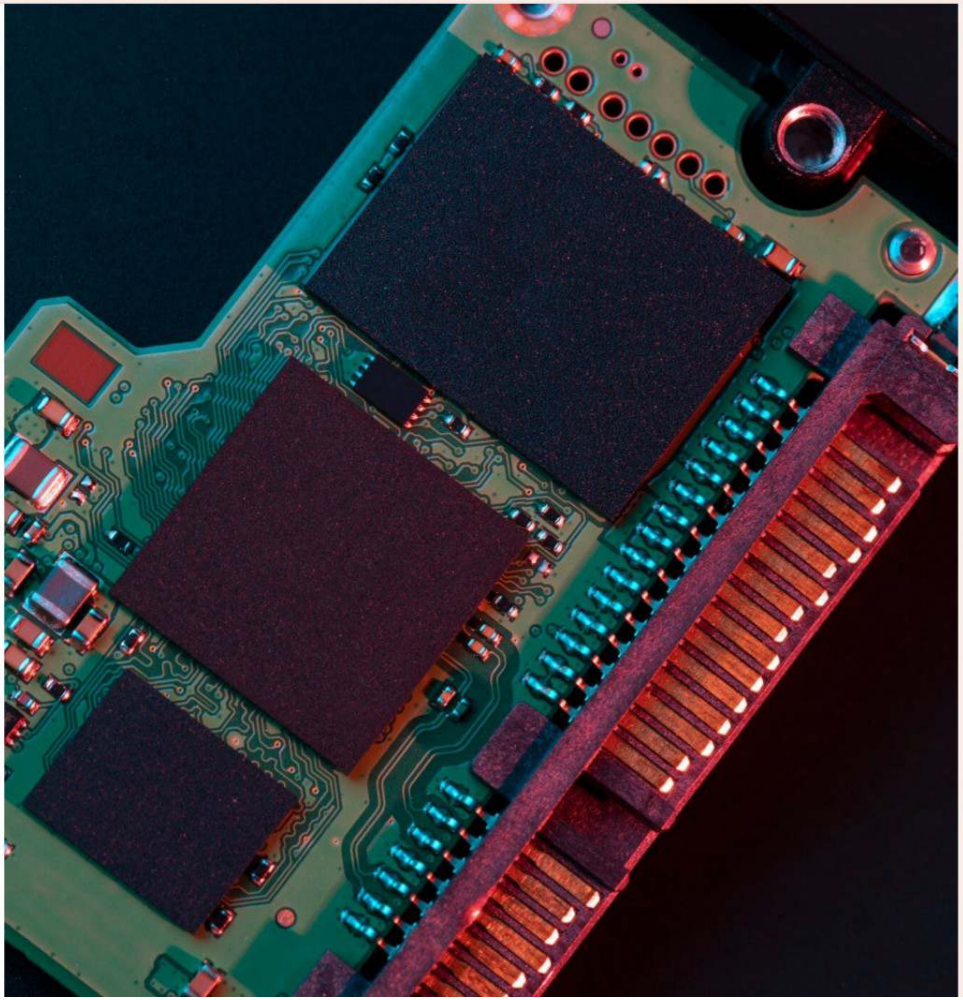


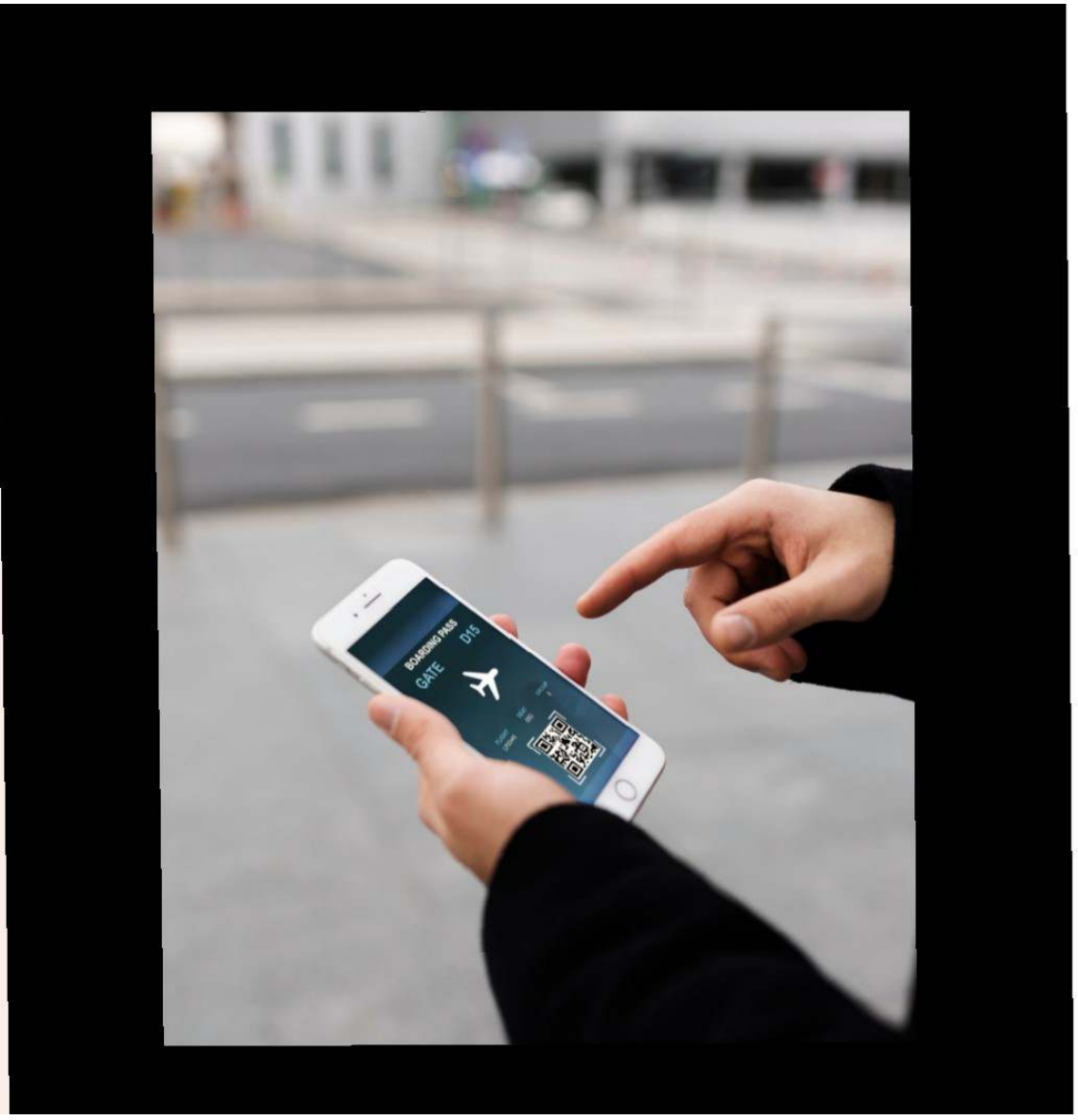
Leveraging Flutter for Mobile App Development

Flutter is a powerful **cross-platform mobile app development framework** developed by Google. It enables developers to create beautiful and high-performance apps for both Android and iOS platforms using a single codebase. By leveraging Flutter, we can ensure a consistent and seamless experience for users of our parking availability app.

Interacting with Raspberry Pi

Raspberry Pi, a small yet powerful single-board computer, can be used to collect and process parking data. By connecting Raspberry Pi to sensors in parking lots, we can gather information on available parking spaces. Our Flutter app will communicate with Raspberry Pi through APIs, allowing users to access real-time parking availability information.



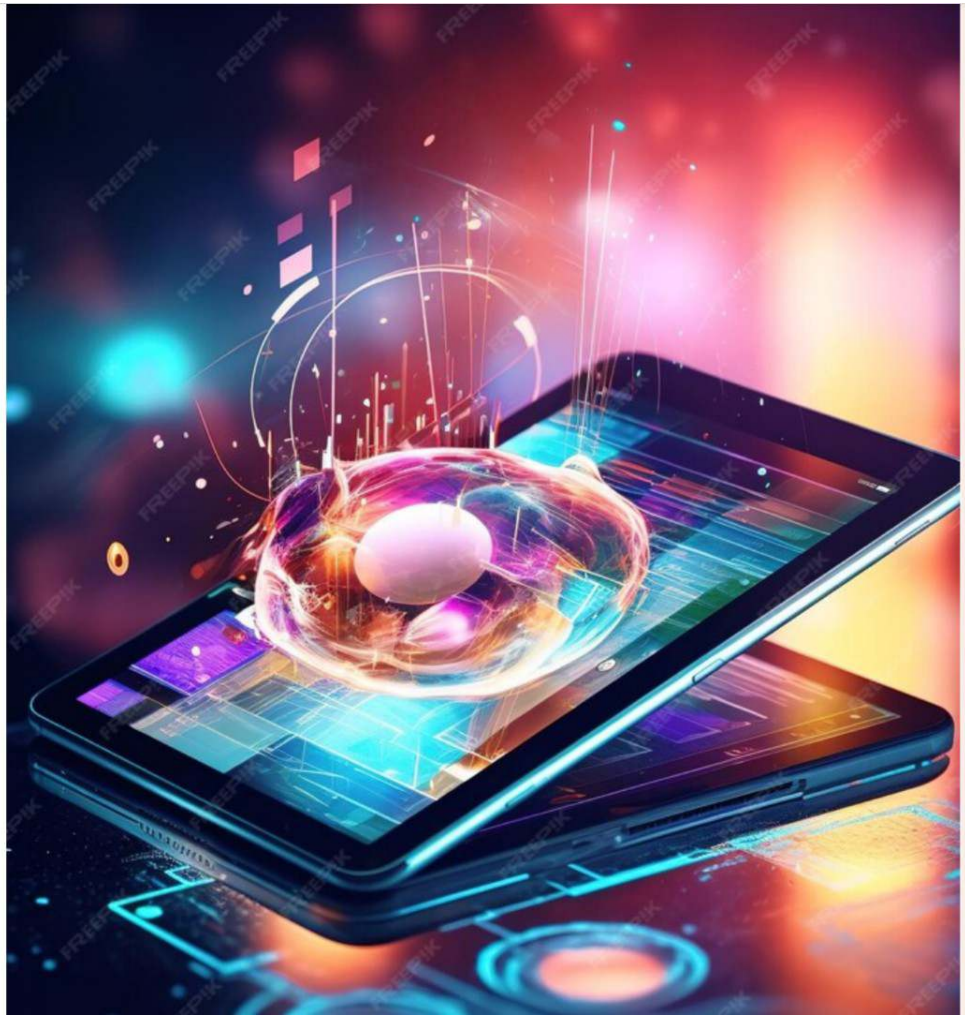


Designing the User Interface

A well-designed user interface is crucial for a successful app. Our parking availability app will have an intuitive and user-friendly interface that displays **real-time availability** of parking spaces. We will focus on creating a clean and modern design, with clear indicators for available and occupied spaces, ensuring a hassle-free parking experience for users.

Implementing Real-Time Updates

Real-time updates are essential for providing accurate parking availability information. Our app will continuously receive data from Raspberry Pi sensors and update the availability status in real-time. By leveraging Flutter's reactive programming model and WebSocket communication, we can ensure instant updates and a seamless user experience.

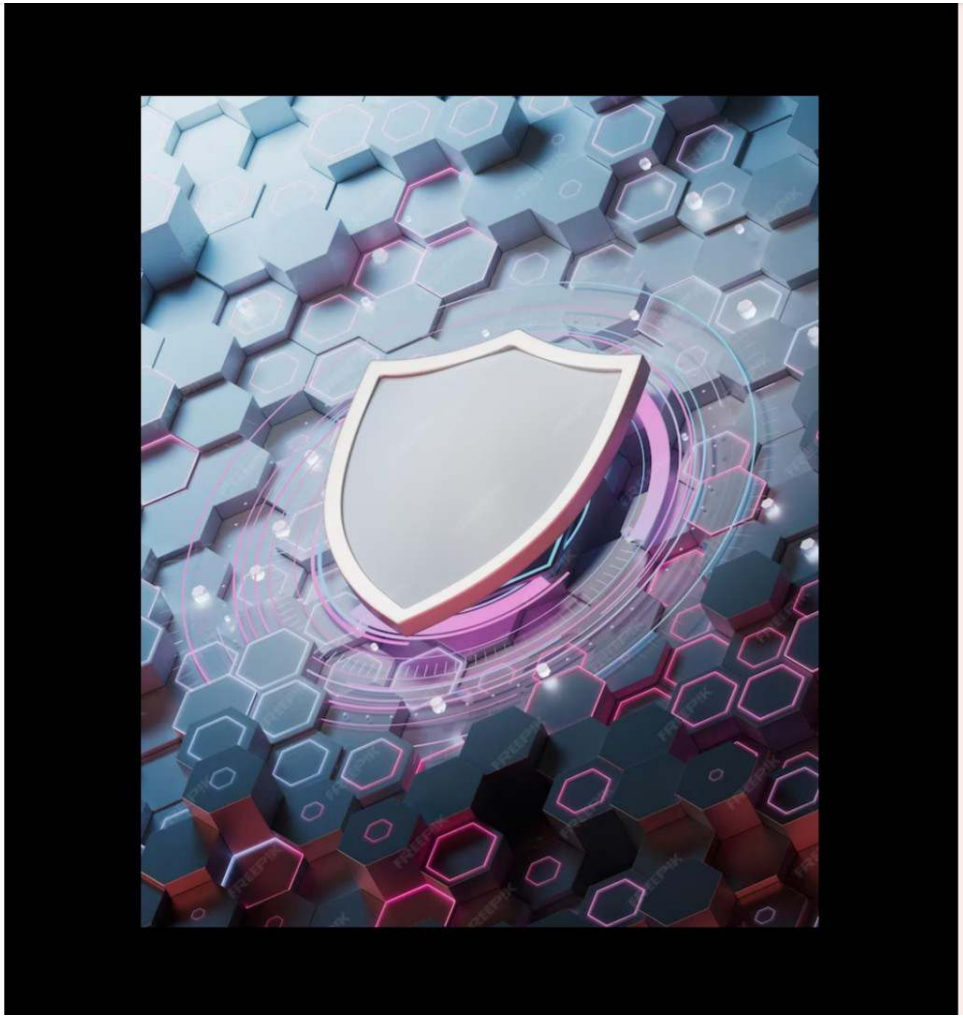


Handling User Queries and Navigation

To enhance user experience, our app will include features such as **search functionality**, **navigation to parking lots**, and **user feedback**. Users can search for specific parking lots, get directions, and provide feedback on parking availability. These features will make our app more user-centric and convenient for drivers.

Ensuring Security and Privacy

Security and privacy are of utmost importance when dealing with user data. Our app will implement robust security measures to protect user information and ensure secure communication between the app and Raspberry Pi. We will use encryption techniques, secure APIs, and follow best practices to maintain user trust and data privacy.



Future Enhancements and Scalability

Our parking availability app has immense potential for future enhancements and scalability. We can integrate additional features like **payment integration**, **reservation system**, and **integration with smart parking systems**. By leveraging the power of Flutter and Raspberry Pi, we can continuously improve and expand the capabilities of our app to meet evolving user needs.