JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

EZY PARK (ANDRIOD APPLICATION)

Prof. Urvashi Rakholiya [1], Ashish Panchal [2], Jaymin Shah [3]

[1,2,3]Department of Computer Science and Engineering,
Parul Institute of Technology, Vadodara, Gujarat, India

ABSTRACT: This research paper describes the design and development of an android application that allows users to easily book their parking slots in advance. The app is designed to make finding and reserving parking spots more convenient and efficient in urban areas. Features such as real-time availability, ability to cancel or modify bookings are included to enhance user experience. The app aims to alleviate the stress and frustration often associated with parking in urban areas by providing a reliable, efficient platform for booking parking spaces.

KEYWORDS: Android application, Location-based, Parking spot reservation, User account, Booking history, Real-time availability.

INTRODUCTION

Car parking has become a significant challenge in urban areas due to a lack of parking facilities and a growing number of vehicles. This leads to drivers roaming around the city during peak hours, causing traffic, wasting time and money. As technology advances, there is a growing need to improve existing technologies. Our goal is to create an application that not only improves safety, saves space-time and fuel, but also provides an extra layer of detail for those who need a proper assessment of parking requirements, space availability, and traffic flow. The rapid economic and population growth in urban areas has led to an increase in the use of private vehicles, making vehicle parking a major concern for governments. In India, 28 out of 1000 people own a car, which translates to about 61 million registered cars. This has resulted in tremendous pressure for parking spaces, leading to various problems such as traffic jams, illegal parking, and parking disputes. It is estimated that on average, automobiles only operate for 400 hours per year, and are stationary for the rest of the time (95%). This data reflects the high demand for parking spaces. However, research shows that legal parking spaces are underutilized, while illegal parking spaces are on the rise. The desire to park as close to our destination as possible often results in long periods of time spent searching for parking, especially in metro cities like Mumbai and Bangalore.

NEED OF STUDY

There is a need to study and develop an Android parking application because it has the potential to address many of the issues and challenges associated with traditional parking methods. One major issue that it can help solve is the inconvenience of finding a parking space, particularly in densely populated urban areas. By providing drivers with the ability to easily locate available parking spaces, an app can reduce the time and effort spent searching for a spot. Another issue that it can help address is the frustration of having to constantly return to a parking meter to add more time or risk receiving a parking ticket. An app can enable users to pay for and extend their parking time remotely, eliminating the need to go back to the parking meter. This can also help reduce the risk of getting a ticket for overstaying one's welcome. Additionally, an app can help lower the environmental impact of parking. By providing an option for electronic payments and eliminating the need for paper receipts and coins, it can help reduce waste and encourage more sustainable practices. Furthermore, it can make it easier for parking lot operators to manage their lots, as they can track usage and revenue through the app. Overall, an android parking application can provide a lot of benefits, making the parking experience more convenient, efficient, and environmentally-friendly. It can solve problems like convenience, efficiency, and environment and making it important tool for any city or town.

PROBLEM DEFINITION

The problem of inadequate parking spaces and the related issues such as traffic jams, illegal parking, and parking disputes in urban areas has become increasingly prevalent. Drivers often face inconvenience and frustration when searching for available parking spots, particularly in crowded areas. Additionally, the task of constantly monitoring the vehicle that is parked in cities or in no-parking and remembering to move one's vehicle can be tedious and can lead to the risk of getting no-parking fine.

To address these challenges, this research proposes the development of an Android-based parking application called EZY Park. The objective of this application is to provide a more convenient and efficient parking experience for drivers by helping them locate available parking spots and pay for parking remotely.

The main objectives of this research include:

- Design and development of EZY Park, an android-based parking application, to aid drivers in finding parking spots and paying for parking remotely.

- Evaluation of the effectiveness of EZY Park in reducing the time and effort spent searching for parking spots as well as the risk of receiving parking tickets.

- Analysis of the impact of EZY Park on the environment through reducing paper waste and increasing electronic payments.

- Study of the benefits that EZY Park provides to parking lot operators in managing their lots, such as tracking usage and revenue.

- This research aims to provide a user-friendly and efficient solution to current parking challenges by developing a parking application that can benefit both drivers and parking lot operators alike.

PROPOSED WORK

In order to build a seamless and hassle-free experience for all the people who struggle to find parking spots for their vehicles, we designed EZY PARK Mobile Application keeping the user needs.

EXISTING SYSTEM

Understanding the problems

1. In heavily populated areas, parking spots can be scarce.
2. The practice of parking on the side of the road not only creates a hindrance for maintenance workers and drivers, but also blocks important ways.
3. Sometimes it is difficult to recognize if the parking spot is authorized or not.
4. Many parking places the employee would keep a notebook use physical tokens or tickets to manage all the parking space.
5. Fear of vehicle damage People forgetting the area where they have parked their vehicle.
6. Parking is very expensive but we get to know the price only after reaching the parking spot.

PROPOSED SYSTEM

Some of the features were as follows:

- The ability to choose and reserve a parking spot in advance.
- Real-time information on parking slot availability.
- Option to extend the duration of the parking reservation.

LITERATURE REVIEW

- In recent years, Android parking apps have gained popularity as a convenient way for drivers to find and pay for parking. These apps often allow users to locate available parking spaces, view prices and payment options, and make payments directly from their mobile device. One study found that the use of a parking app can significantly decrease the time spent searching for a parking spot, leading to a better overall user experience and potentially reducing traffic congestion. Another study discovered that parking apps can increase parking turnover and revenue for parking operators. Despite these benefits, concerns have been raised about the security and privacy of personal and financial information entered into these apps. It is crucial for developers to implement proper security measures to protect user data. In conclusion, Android parking apps offer a convenient and efficient solution for finding and paying for parking, but it is essential for developers to prioritize the security and privacy of their users.

- Some parking apps offer additional features, such as the ability to reserve a parking space in advance or to set reminders for when a parking meter is about to expire. These features can help users save time and avoid parking tickets. Some cities and towns have implemented parking apps as part of a wider smart city initiative, aiming to improve the efficiency and sustainability of urban transportation. These apps can be integrated with other transportation systems, such as public transit, bike-sharing, and car-sharing, to provide a more comprehensive mobility solution. Some researchers have examined the social and economic impacts of parking apps, including the potential to reduce parking demand and costs, increase accessibility and mobility for underserved communities, and support local businesses. While Android parking apps have gained widespread adoption, some users may still prefer traditional methods of paying for parking, such as using coins or credit cards at parking meters. It is important for developers to consider the needs and preferences of different user groups when designing and marketing their apps.

- Some parking apps allow users to pay for parking using their smartphone's mobile wallet or a dedicated parking payment account. This can be more convenient than carrying around cash or coins, and it can also reduce the risk of losing or forgetting a physical parking ticket. Some parking apps offer loyalty programs or rewards for frequent users, such as discounts on parking fees or the ability to earn points that can be redeemed for other perks. These programs can incentivize users to continue using the app and can also provide valuable data for the parking operator. Some parking apps allow users to leave ratings and reviews for parking facilities, which can help other users make informed decisions about where to park. These reviews can also provide feedback to the parking operator about the quality and maintenance of their facilities. While Android parking apps have the potential to improve the parking experience for users, it is important for developers to consider the potential negative impacts on parking attendants and other workers. Some critics have raised concerns about the automation and digitization of parking services, and it is important to ensure that workers are treated fairly and have access to training and opportunities for advancement.

Methodology

1. Identify the problem: The first step in developing an Android parking application is to identify the specific challenges that the app aims to address. This might include issues such as difficulty finding parking, long search times, or high parking fees.

2. Research existing solutions: Next, it is important to research existing solutions for the identified problems, including both mobile apps and traditional methods of finding and paying for parking. This can provide valuable insights into the strengths and weaknesses of different approaches, and help inform the design and features of the app.

3. Define the target audience: To tailor the app to the needs and preferences of its users, it is important to define the target audience. This might include considerations such as the types of devices they use, their location, and their preferred payment methods.

4. Design the app: Based on the research and target audience, the app should be designed with a user-friendly interface and a clear workflow for finding and booking parking. The design should also consider issues such as security and privacy, as well as the need for any additional features or integrations.

5. Test the app: Before launching the app, it is important to thoroughly test it to ensure that it functions as intended and meets the needs of the target audience. This might include user testing, usability testing, and performance testing.

6. Launch the app: Once the app has been tested and refined, it is ready to be launched and made available to users. This might involve submitting the app to app stores, promoting the app through marketing campaigns, and providing customer support.

7. Monitor and update the app: After launching the app, it is important to monitor its performance and user feedback, and to make updates and improvements as needed. This can help ensure that the app continues to meet the needs of its users and remains competitive in the market.

Fundamentals

Android applications are primarily developed using the Java programming language; However, they are not executed through a standard Java Virtual Machine (JVM). Google has created a custom virtual machine named Dalvik, which is responsible for interpreting and executing Java bytecode. All Java classes must be converted into a Dalvik compatible instruction set before they can be executed on an Android device. The Dalvik virtual machine takes the generated Java class files and combines them into one or more Dalvik Executable (.dex) files. This process allows for efficient storage and memory usage by reusing duplicate information from multiple class files, which is crucial in mobile environments.

Application Framework

1. Home screen: The home screen of the app should provide a clear overview of the available parking options, including the location, price, and availability of each space. This might include a map view or a list view, as well as filters and search capabilities to help users find the most suitable parking option.

2. Booking flow: The app should provide a clear and easy-to-use workflow for booking a parking space, including selecting the desired location and duration, entering payment information, and confirming the booking.

3. Payment processing: The app should support a range of payment options, such as mobile wallets, credit cards, and dedicated parking payment accounts. It should also securely handle the processing of payments and provide receipts for users.

4. User account: The app should allow users to create and manage a personal account, including storing payment information and booking history. It should also provide options for users to customize their preferences, such as notification settings and preferred payment methods.

5. Data and analytics: The app should track and analyze key metrics, such as booking and payment data, to help the parking operator optimize their operations and provide a better experience for users. It should also provide tools for users to track their own parking usage and expenses.

6. Navigation: The app should provide clear and intuitive navigation, allowing users to easily find the information and features they need. This might include a main menu, tabs, or other navigation elements.

7. Notifications: The app should provide notifications to users to help them manage their parking experience, such as reminders when their parking meter is about to expire or when their booking is about to end. Users should be able to customize their notification settings within the app.

8. Maps and directions: The app should provide maps and directions to help users locate the parking space they have booked. This might include integration with a mapping service such as Google Maps.

9. Security: The app should prioritize the security and privacy of its users, including measures such as secure payment processing and data encryption. It should also comply with relevant laws and regulations related to data protection.

Software Implementation for developing Android Application

To implement an Android application in Android Studio, the following general steps could be followed:

1. Install Android Studio: First, Android Studio needs to be installed on the development machine. This can typically be done by downloading the software from the Android developer website and following the prompts to install it.

2. Create a new project: In Android Studio, a new project can be created by selecting "File" > "New" > "New Project". This will open a wizard that guides the user through the process of setting up the project, including choosing a name and location for the project and selecting the target Android version and build tools.

3. Design the user interface: Next, the user interface for the app can be designed using the layout editor in Android Studio. This might involve creating layouts using a combination of views, such as text views, buttons, and image views, and arranging them using a layout manager.

4. Implement the app logic: After the user interface is complete, the app logic can be implemented using Java or Kotlin code. This might involve writing code to handle user input, perform calculations, access data from a database, or communicate with external servers or services.

5. Test the app: Before releasing the app, it is important to test it thoroughly to ensure that it functions as intended and meets the needs of the target audience. This might involve running unit tests, conducting user testing, and debugging any issues that arise.

6. Package the app: Once the app is ready for release, it can be packaged into an APK (Android Package Kit) file, which is the file format used to distribute Android apps. This can be done by selecting "Build" > "Build APK" in Android Studio.

7.Publish the app: Finally, the app can be published to an app store, such as the Google Play Store, or distributed directly to users through other channels. This might involve creating a listing for the app, providing screenshots and other promotional materials, and setting a price or other terms for the app.

Android Studio Software

Android Studio is a development environment for building Android apps. It is the official integrated development environment (IDE) for Android app development, and it is available for download from the Android developer website.

Android Studio includes a range of tools and features that are designed to help developers build and test Android apps more efficiently. Some of the key features of Android Studio include:

- A layout editor: This tool allows developers to design the user interface of their app using a drag-and-drop interface.
- A code editor: This provides support for writing and editing code in Java or Kotlin, including syntax highlighting, refactoring, and debugging tools.
- A built-in emulator: This allows developers to test their app on a virtual device, without the need for a physical device.
- A range of plugins: Android Studio supports a range of third-party plugins that can be installed to add additional functionality to the development environment.
- Android Studio is based on the IntelliJ IDEA development environment, and it includes many of the same features and tools, such as a code editor, a debugger, and support for version control systems.
- Android Studio is designed to be user-friendly and intuitive, with a clean and organized interface that makes it easy to navigate and find the tools and resources needed for app development.
- Android Studio provides a range of built-in templates and code samples that can help developers get started with common tasks and patterns, such as creating an app with a navigation drawer or implementing a login flow.
- Android Studio includes a range of tools and features for testing and debugging Android apps, including a built-in emulator, a profiler for analyzing performance, and support for testing frameworks such as JUnit.
- Android Studio integrates with the Android developer tools, such as the Android SDK (Software Development Kit) and the Android NDK (Native Development Kit), which provide the libraries, tools, and resources needed for building Android apps.
- Android Studio is regularly updated with new features and improvements, and developers can stay up-to-date with the latest releases by using the built-in update mechanism or by downloading the latest version from the Android developer website.
- Android Studio supports a range of programming languages, including Java, Kotlin, and C++, and it provides specialized tools and features for each language. For example, Kotlin support includes features such as syntax highlighting, refactoring, and code completion.
- Android Studio includes a range of tools for developing and testing Android apps on different devices and platforms, including support for different screen sizes, resolutions, and orientations, as well as support for different versions of Android.
- Android Studio integrates with Google Play, the official app store for Android apps, and it provides tools and resources for publishing and promoting apps on the platform. This includes features such as the Play Console, which allows developers to track the performance and status of their apps, and the Play Billing Library, which provides tools for implementing in-app purchases and subscriptions.

- Android Studio supports a range of third-party tools and services that can be integrated into the development process, such as testing frameworks, performance analytics tools, and cloud-based services.

- Android Studio provides a range of resources and documentation for developers, including API reference documentation, code samples, and design guidelines, as well as a community forum where developers can ask questions and share knowledge with each other.

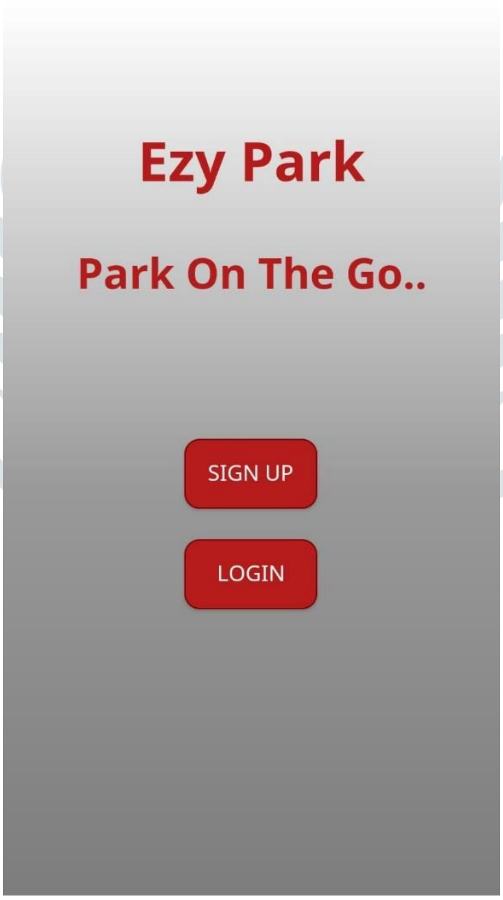
Overall, Android Studio is a comprehensive and powerful tool for developing Android apps, and it provides a range of features and resources to help developers build high-quality and successful apps.

IMPLEMENTATION

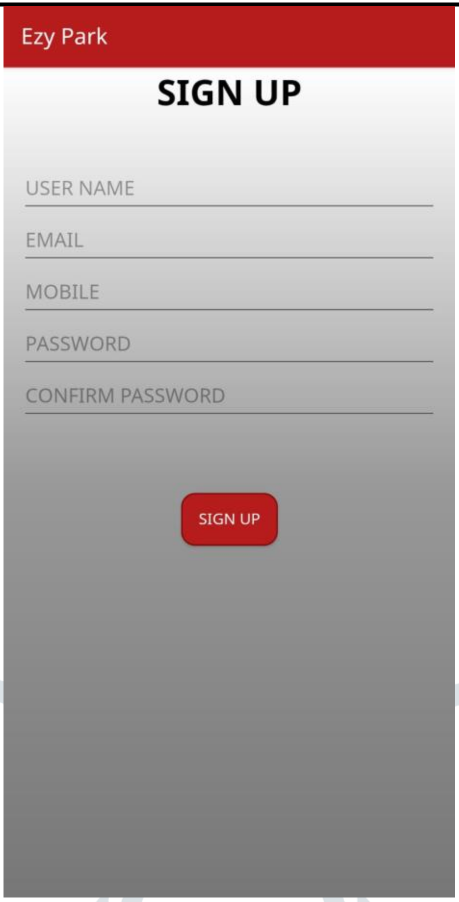
ANDROID - Android is a mobile operating system based on Linux, primarily used in smartphones and tablet devices that use ARM processors. It consists of a kernel built on Linux, with middleware, libraries, and APIs written in C, and application software built on an application framework that uses Java-compatible libraries. To create new applications for the Android operating system, developers use the Android Software Development Kit (SDK) which is primarily written in the Java programming language. Android utilizes the Dalvik virtual machine, which uses just-in-time compilation to execute code written in Dalvik Executable (DEX) format. This is typically translated from Java bytecode. In summary, the process of developing apps for android is based on the Android SDK, Java and Dalvik Virtual Machine.

1. Signup/Login : Initially, the user has to register his details with the application for the first time. This is a one-time registration. The user has to enter details like user name, phone number and email- id. All this data will be stored. Booking for slots mandatory has to be done two hours prior to arrival.
2. Parking Slot booking: The user is provided with multiple parking locations. User has to select one of the locations provided where user (he/she) desires to park the vehicle.
3. Slot booking confirmation: After the successful booking the user will get the receipt that is generated for the confirmation of successful payment.

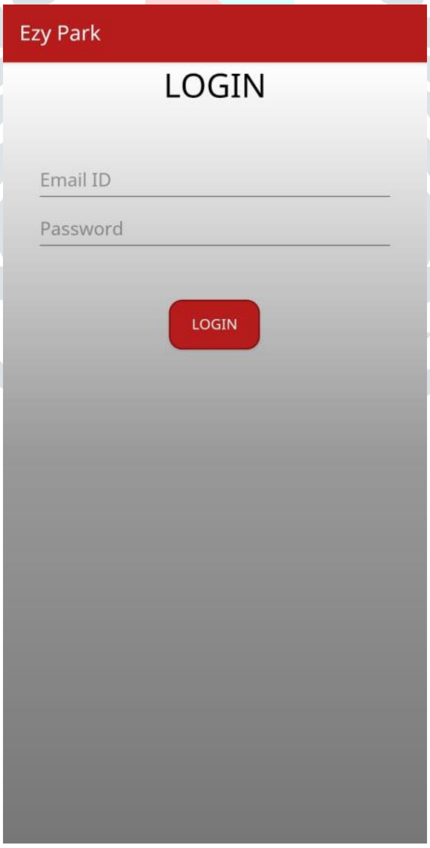
Results



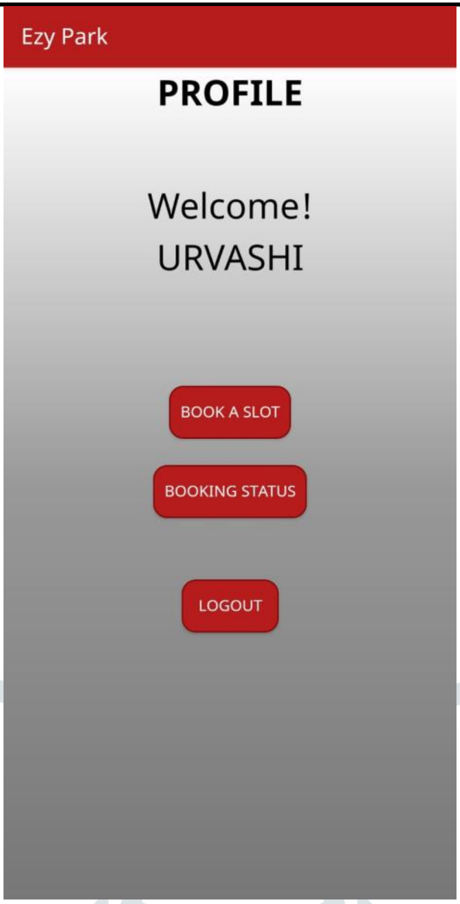
1. Home Page



2.Signup



3.Login

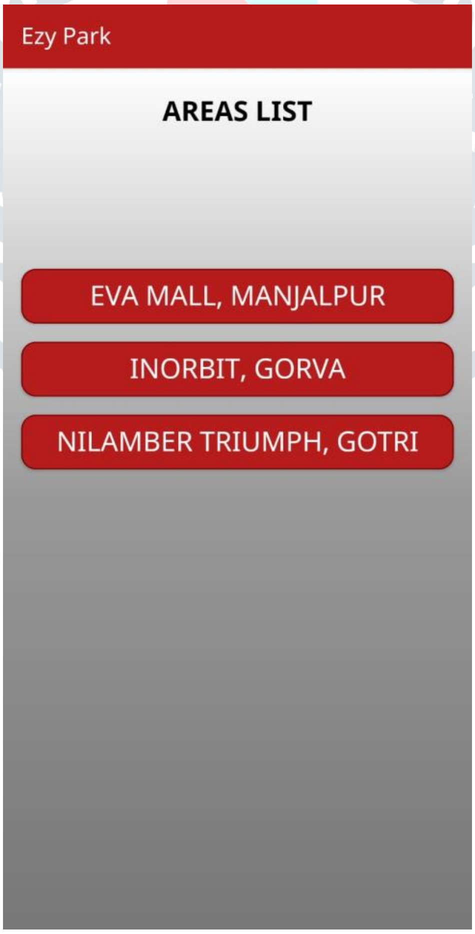


Ezy Park

PROFILE

Welcome!
URVASHI

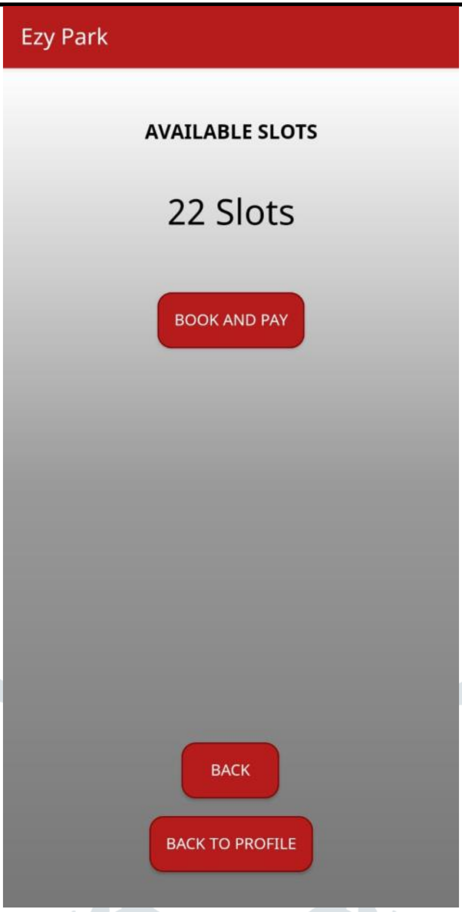
4.Profile



Ezy Park

AREAS LIST

5.Area Selection

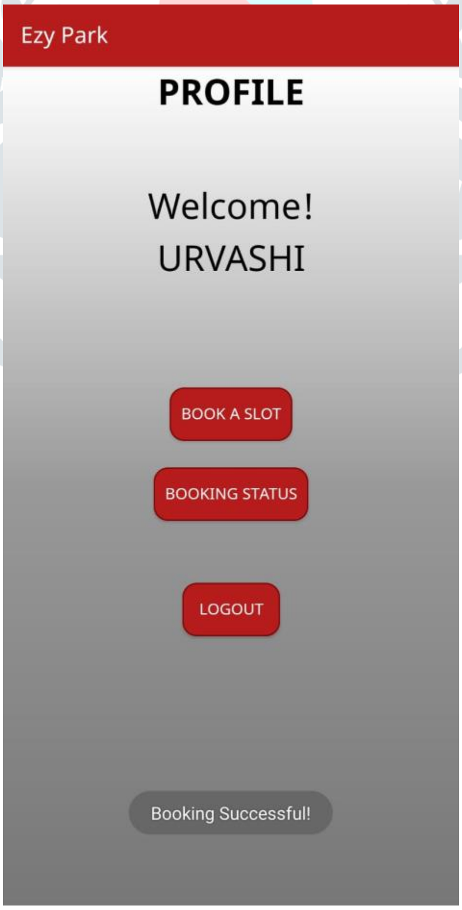


Ezy Park

AVAILABLE SLOTS

22 Slots

6.Slot Booking



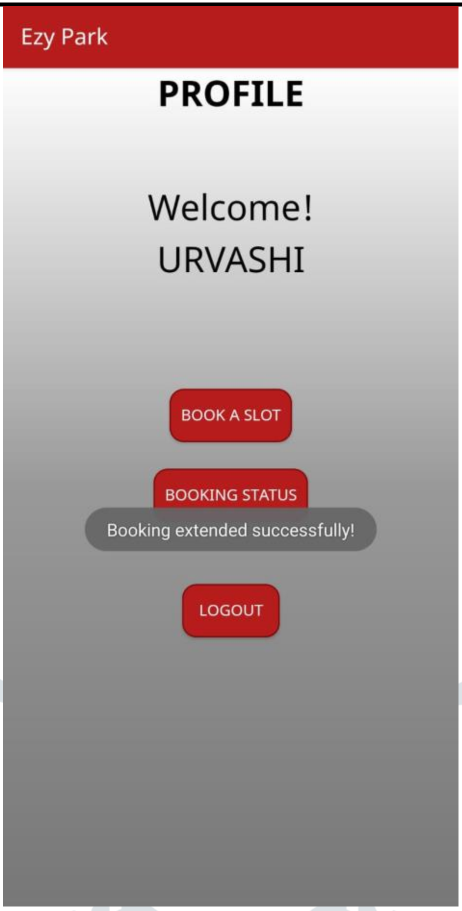
Ezy Park

PROFILE

Welcome!
URVASHI

Booking Successful!

7.Booking Successful

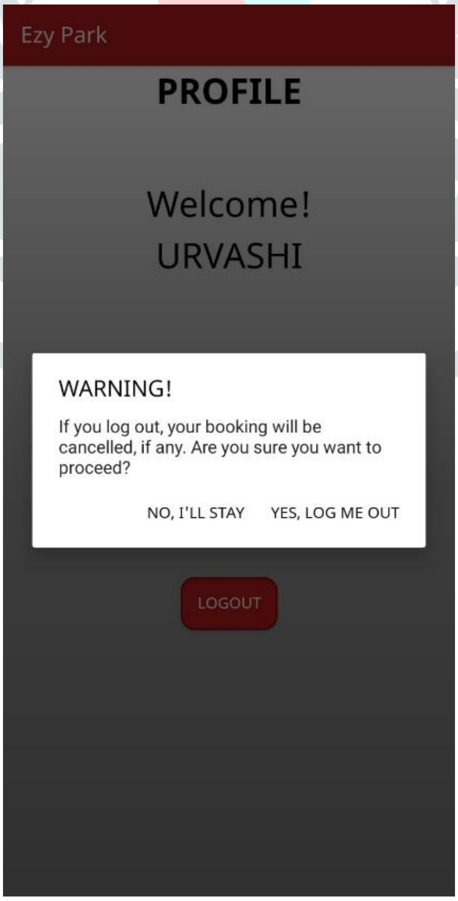

Ezy Park

PROFILE

Welcome!
URVASHI

Booking extended successfully!

8. Booking Extended Successful


Ezy Park

PROFILE

Welcome!
URVASHI

WARNING!

If you log out, your booking will be cancelled, if any. Are you sure you want to proceed?

9.App Exit

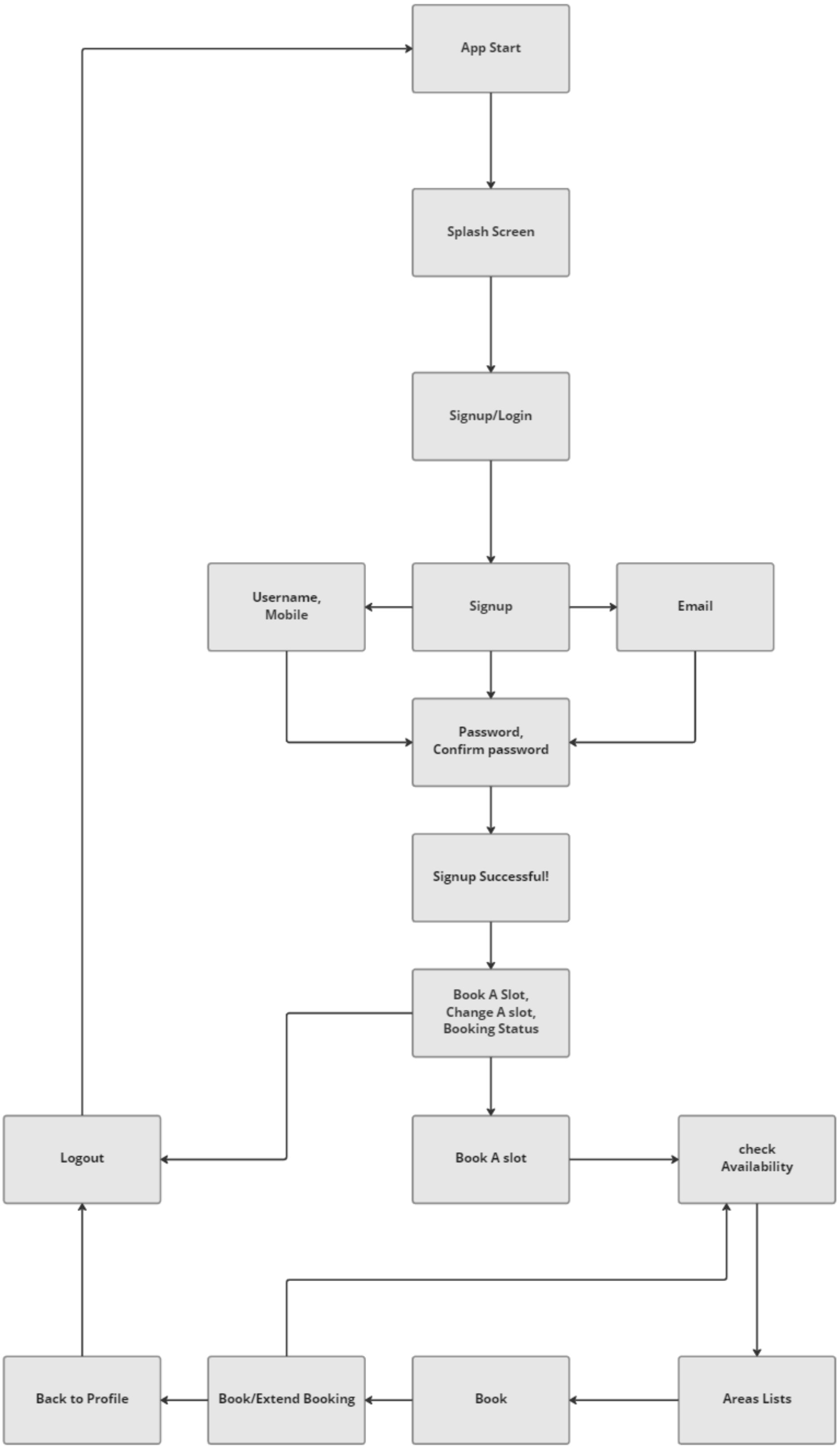

App Start
Splash Screen
Signup/Login
Username, Mobile
Signup
Email
Password, Confirm password
Signup Successful!
Book A Slot, Change A slot, Booking Status
Logout
Book A slot
check Availability
Back to Profile
Book/Extend Booking
Book
Areas Lists

FIG. Main

Conclusions and Discussions

The Application EZY Park is used for booking parking slots prior without great effort just by the user using their phones. Various methodologies of smart parking have been implemented to provide better services to the end-users and improve the overall management of the existing parking system. After analyzing the different solutions, we came to a conclusion that we can solve the existing problem of parking our cars by just booking an online parking slot, having easy access, finding alternate and nearest parking areas & all these problems will be solved by just our one application. Building the Application from scratch was a great learning experience. We learned and discovered new things at every step of the process. After spending a lot of time developing a minimal, easy-to-use app, we have a potential solution that has always meant to been there.

In conclusion, The EZY Park Android app offers a convenient and efficient solution for booking parking slots in advance. By leveraging various smart parking technologies, the app helps users find and reserve parking spaces with minimal effort, and it also helps improve the overall management of the parking system. The development of the EZY Park app was a valuable learning experience, and the resulting solution has the potential to solve many of the common problems that drivers face when trying to find parking. Overall, the EZY Park app represents a promising approach to addressing the challenges of parking in modern cities.
One of the key benefits of the EZY Park app is its user-friendly interface and intuitive design. By making it easy for users to find and book parking slots, the app helps reduce the frustration and stress that can come with searching for parking.

REFERENCES

1. A mobile application leveraging QR-codes to support efficient urban parking, Alessio Bechini; Francesco Marcelloni; Armando Segatori, IEEE/ 2018. [1]

2. Intelligent Systems for Car Parking with Image Processing, Yusnita R., Norbaya Fariza and Basharuddin Norazwinawati, IEEE/ 2012. [2]

3. Integration of RFID and WSN technologies in a smart parking system, Mainetti i. L., Palano L., Patrono L., Stefanizzi M. L. and Vergallo R., IEEE/ 2014. [3]

4. A Vision-based car parking sy his paper introduces a novel algorithm that increases the efficiency of the current cloud-based smart-parking system and develops a network architecture based on the Internet-of-Things technology, Al-Absi Hamada R. H., Daniel Devaraj Justin Dinesh, Sebastian Patrick and Vooi Voon Yap, IEEE/ 2016. [4]

5. A Cloud-Based Smart-Parking System Based on Internet-of-Things Technologies. (Aswathy James ; Prince Abraham ; IEEE/2018). [5]

6. Verification of smart guiding system to search for parking space viaDSRC communication, Hsu C. W., Shih M. H., Huang H. Y., Shiue Y. C. and Huang S. C., IEEE/ 2012. [6]

7. Internet of Things (IoT). Newton, MA, USA: TechTarget, Wigmore, IEEE/ 2014. [7]

8. Android Application for S-Park System, In this paper, a prototype of smart parking (S-Park) system is designed that allows drivers to effectively find and reserve the empty parking spaces using an android application connected to Internet, Gaurav, Kate, et al, IEEE/ 2015. [8]

9. Smart Routing: A Novel Application of Collaborative Path-Finding to Smart Parking Systems, C. Rhodes, W. Blewitt, C. Sharp, G. Ushaw, G. Morga. [9]

10. Reserve Parking and Authentication of Guest Using QR Code, Ahmed Abdullah Ahmed; Omar Ismael Al-Sanjary; Saranya Kaeswaren. [10]