Faculty of Electrical Technology and Engineering

DESIGN AND OPTIMIZATION OF RESERVATION PARKING SYSTEM USING APP

AINA ALISA BINTI GHAZALI

Bachelor of Electrical Engineering Technology with Honours

2023

DESIGN AND OPTIMIZATION OF RESERVATION PARKING SYSTEM USING APP

AINA ALISA BINTI GHAZALI

A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor of Electrical Engineering Technology with Honours

Faculty of Electrical Technology and Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2023

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI DAN KEJUTERAAN ELEKTRIK

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : DESIGN AND OPTIMIZATION OF RESERVATION PARKING SYSTEM USING APP

Sesi Pengajian : 2023

Saya AINA ALISA BINTI GHAZALI mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (✓):

☐	**SULIT***	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	**TERHAD***	(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
☒	**TIDAK TERHAD**	

Disahkan oleh:

(TANDATANGAN PENULIS)
Alamat Tetap:

(COP DAN TANDATANGAN PENYELIA)

Tarikh: 13 / 1 / 2024

Tarikh: 14 Januari 2024

*CATATAN: Jika laporan ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I declare that this project report entitled “DESIGN AND OPTIMIZATION OF RESERVATION PARKING SYSTEM USING APP” is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : 

Student Name : AINA ALISA BINTI GHAZALI

Date : 13/1/2024

APPROVAL

I approve that this Bachelor Degree Project II (PSM2) report entitled DESIGN AND OPTIMIZATION OF RESERVATION PARKING SYSTEM USING APP" is sufficient for submission.

Signature : 

Supervisor Name : MOHAMAD HANIFF BIN HARUN

Date : 14 Januari 2024

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Electrical Engineering Technology (Industrial Automation & Robotics) with Honours.

Signature : 

Supervisor Name : MOHAMAD HANIFF BIN HARUN

Date : 14 Januari 2024

Signature :

Co-Supervisor :

Name (if any)

Date :

DEDICATION

This research is wholeheartedly devoted to our adores parents, who have been a constant source of inspiration for me and given the fortitude to keep going even when I felt like giving up. My parents continue to support me on a variety of levels, including spiritually, emotionally, financially and emotionally.

I would want to express our gratitude to our brothers, sisters, relative, mentors, friends and classmates who provided me with words of encouragement and guidance to help me complete my study. I am also wanted to say thank you my supervisor and my friends who have encouraged, guided and inspires me to complete this project.

And last, we would like to thank the Almighty God for blessing me with a long and healthy life, as well as for granting us wisdom, power of mind, protection and abilities. This book is dedicated to you, God. All of these are available to you free of charge.

ABSTRACT

In this new era, technology were develop rapidly and with the usage of smartphone increasing dramatically. So, all the things can access and searching thru our phone by using fingertips. Next, this parking apps are software application that design to help drivers locate and reserve parking quickly and easily. While, can make payment for parking fee every time and everywhere thru the apps that have been created. It will help a lot because before this parking ticket fee need to pay at the kiosk that have been provide at certain place only so it is quite complicated. After that, for the current solution that have been face now is with the green and red lamp system. The green lamp means that parking lot is empty while the red lamp means there are already have car that park there. It also provide the LCD display that show how many empty parking that are still available. Other than that, it already have apps that can show you or find available parking space thru it so there is the improvement that already have and much better from previous solution. Therefore, the proper solution that I can improve is to add on gate that can be close if the person make a payment for the parking that have been reserved so from that no one can use the parking space accept the person that have reserved that space using id that have been given after payment to open the gate. Lastly, expected outcome project is the user can find available parking space, make a reservation and payment fee for the parking using their mobile devices.

ABSTRAK

Dalam era baru ini, teknologi berkembang pesat dan dengan penggunaan telefon pintar meningkat secara mendadak. Jadi, semua perkara itu boleh diakses dan dicari melalui telefon kita dengan menggunakan hujung jari. Seterusnya, aplikasi tempat letak kereta ini ialah aplikasi perisian yang direka untuk membantu pemandu mencari dan menempah tempat letak kereta dengan cepat dan mudah. Manakala, anda boleh membuat bayaran untuk bayaran letak kereta setiap masa dan di mana-mana melalui aplikasi yang telah dibuat. Ia akan banyak membantu kerana sebelum ini bayaran tiket parking perlu dibayar di kiosk yang telah disediakan di tempat-tempat tertentu sahaja jadi agak rumit. Selepas itu, bagi penyelesaian semasa yang dihadapi sekarang ialah dengan sistem lampu hijau dan merah. Lampu hijau bermakna tempat letak kereta kosong manakala lampu merah bermakna sudah ada kereta yang diletakkan di situ. Ia juga menyediakan paparan LCD yang menunjukkan berapa banyak ruang parkir kosong yang masih tersedia. Selain itu, ia sudah mempunyai aplikasi yang boleh menunjukkan kepada anda atau mencari ruang letak kereta yang tersedia melaluinya supaya terdapat peningkatan yang sudah ada dan lebih baik daripada penyelesaian sebelumnya. Oleh itu, penyelesaian yang sewajarnya yang boleh saya perbaiki ialah menambah pada pintu pagar yang boleh ditutup sekiranya orang tersebut membuat bayaran untuk tempat letak kereta yang telah ditempah supaya daripada itu tiada sesiapa boleh menggunakan tempat letak kereta tersebut menerima orang yang telah menempah ruang itu menggunakan ID yang telah diberikan selepas pembayaran untuk membuka pintu pagar. Akhir sekali, hasil yang diharapkan daripada projek itu ialah pengguna boleh mencari tempat letak kereta yang tersedia, membuat tempahan dan membayar yuran tempat letak kereta menggunakan peranti mudah alih mereka.

ACKNOWLEDGEMENTS

Alhamdulillah, grateful to divine with His grace I can finished the work I do. As finally I were able to finished my project proposal that has been given by my supervisor. This task had been done with all effort by my own even thought I have faced a few problem while doing this project. Luckily, all the problems could be settled and I were able to adapt properly and wisely

In preparing this report, I was in contact with many people and academicians. They have contribute towards my understanding and thought. In particular, I wish to express my sincere appreciation to my project supervisors, Sir Mohamad Haniff Bin Harun for encouragement, guidance critics, friendship, advices and motivation.

My fellow undergraduate students should also be recognized for their support. My sincere appreciation also extend to all my colleagues and other who have provided assistance at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space.

Finally, I am grateful to all my family members, for supporting steadfastly and their appreciated advice through my project completion.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

δ	-	Voltage angle
	-	
	-	
	-	
	-	
	-	
	-	
	-	

LIST OF ABBREVIATIONS

V	-	Voltage
	-	
	-	
	-	
	-	
	-	
	-	
	-	

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

This chapter will discuss the project's basic concepts. It will include the project background, problem statement, objectives, and scope of work. In addition, a simple work flow will be explained, from the start of operation to the end. This project is mostly comprised of Internet-of-Things (IoT) devices, which are required to achieve the Industry Revolution 4.0 (IR 4.0)

1.2 Background

The reservation parking system using an app is a technological solution designed to streamline and optimize the process of reserving parking spaces. It aims to address the challenges associated with finding available parking spots in busy urban areas, commercial establishments, and other high-demand locations. The background of this system can be traced back to the growing urbanization and increased vehicle ownership, which have led to a significant rise in the demand for parking spaces. Outdated parking systems often rely on first-come, first-served basis, resulting in frustration, wasted time, and unnecessary fuel consumption as drivers search for an available spot.

To overcome these challenges, the idea of a parking reservation system developed, using the future potential of mobile applications and smart technologies. The system allows users to reserve parking spots in advance,

ensuring their availability upon arrival quickly and easily. It offers convenience, time savings, and a more efficient utilization of parking spaces. The underlying technology of the reservation parking system typically involves a combination of mobile apps, cloud-based platforms, and real-time data. Users can download the dedicated app onto their smartphones, which provides them with a user-friendly interface to search for and reserve parking spaces based on their desired location, date, and duration.

Parking providers, such as parking lots, garages, or even individual property owners, can integrate their facilities into the system. They can manage their available parking inventory, set pricing and time slots, and monitor the occupancy in real-time. The system employs various technologies to facilitate the reservation process. It may utilize sensors or cameras to monitor and update the availability of parking spots in real-time by using LCD display and thru the apps. Additionally, it may incorporate online payment gateways, allowing users to pay for their reservations securely through the app every time and anywhere.

Benefits of a reservation parking system include improved efficiency in parking space utilization, reduced traffic congestion, and enhanced user experience. It provides a seamless and convenient way for drivers to plan their parking needs in advance, reducing stress and uncertainty associated with finding parking. Other than that, this system also make an improvement for reservation parking system by adding gate for reservation to avoid taking parking spaces that have been booked illegally.

Overall, the background of the reservation parking system using an app lies in the need to address the challenges of urban parking and provide a modern, user-friendly solution that optimizes parking space management and enhances the overall parking experience for both drivers and parking providers.

1.3 Problem Statement

The outdated parking system faces several challenges that prevent efficient utilization of parking spaces and cause frustration for drivers. These challenges include limited parking availability, time-consuming search for available parking spots, traffic congestion caused by vehicles movement.

To address these issues, for the reservation parking system using an app can be defined as time consuming for searching parking which is driver frequently waste time and fuel searching for parking spaces, especially in congested urban areas or high- locations. This inefficient search process contributes to traffic congestion and frustrates both drivers.

Next, uncertainty and frustration for drivers. It mean the uncertainty of finding parking spaces creates stress and frustration for drivers, particularly during peak hours or when visiting popular place when struggling to find available parking spot. This negatively impacts the overall user experience and can deter potential customers from visiting the place. Furthermore, improper payment .Outdated parking systems often lack convenient payment methods which is the drivers need to pay parking fee at the kiosk that have been placed only. Drivers may face difficulties in making

payments. Besides, inefficient parking space management which is the outdated parking systems often rely on a first-come, first-served basis so to avoid from misunderstanding or take the reservation parking illegally, we create a gate specific for reservation parking space.

To address these problems, a reservation parking system using an app aims to provide a user-friendly and efficient solution. It seeks to optimize parking space utilization, reduce searching time, provide real-time information, offer convenient payment method, and enhance accessibility and equity in parking facilities.

1.4 Project Objective

The main aim of this project is to propose a development of reservation parking system by using app that can be lead from smart phone. Specifically, the objective are as follow:

a) To design reservation parking system based on android system using ESP32 microcontroller.

b) To develop the convenience application that user can reserve parking spot by their own anytime and anywhere.

c) To analyse the effectiveness of the system based on the time consume, effectiveness payment method and easiness to find parking spot for the users.

1.5 Scope of Project

There is a significant scope for this project. To avoid any uncertainty of this project due to some limitations and constraints, the scope of project are as follow:

a) This system is tested in open space parking with limited number of parking.

b) This system is tested at Mydin MITC for research purpose.

c) This system is using ESP32 microcontroller and MIT App Inventor.

a)

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The reservation parking system using an app has emerged as a promising solution to address the challenges associated with outdated parking systems. With the growth of smartphones and advancements in technology, this innovative approach enhance mobile applications to streamline and optimize the process of reserving parking spaces. In order to gain a deeper understanding of this topic and explore its various dimensions, a thorough literature review is conducted.

Next, this literature review aims to critically examine the existing body of knowledge related to the reservation parking system using an app. By reviewing education articles, research studies, conference papers, journal and relevant publications, this section provides a comprehensive overview of the current state of research and the key findings in this domain. Besides, the literature review will explore deeply or thoroughly into several important aspects of the reservation parking system using an app. It will explore the evolution of parking systems, highlighting the limitations of traditional approaches and the need for a more efficient and user-friendly solution. The review will discuss the key components required for a successful reservation parking system, such as mobile applications, real-time data, cloud platforms, and online payment gateways.

Moreover, the literature review will examine the benefits and advantages of the reservation parking system for various stakeholders. It will explore the impact on drivers, including time savings, convenience, and reduced frustration, as well as the benefits for parking providers, such as improved space utilization and revenue generation. Additionally, the literature review will examine case studies and examples of successful implementations of reservation parking systems using mobile apps. By examining real-world scenarios in different urban areas or commercial establishments, the review will highlight the impact of these systems on parking efficiency and user satisfaction.

Furthermore, the literature review will identify and discuss the challenges and considerations associated with implementing a reservation parking system using an app. It will address issues related to infrastructure requirements, system integration, data security, user adoption, and equitable access to parking spaces. Other than that, the literature review will explore emerging trends and technologies in the reservation parking system domain, providing insights into potential future developments and innovations. It will identify areas for further research and improvement in the system, including the integration of autonomous parking, predictive analytics, and smart city initiatives.

Lastly, this report aims to consolidate existing knowledge, identify research gaps, and provide a solid foundation for further analysis and discussion on the reservation parking system using an app. The findings from the literature review will inform the subsequent sections of the report,

enabling a deeper understanding of the topic and facilitating informed recommendations and conclusions.

2.2 Smart Parking System with Application

A car parking system is a mechanical device that increases the parking capacity within a parking lot. In Malaysia, there are different types of parking systems, including the smart parking system. The smart parking system can be divided into five main categories which is Parking Guidance and Information System (PGIS), transit-based information system, smart payment system, e-parking, and automated parking [9]. These systems utilize advanced sensing devices such as high-resolution cameras, vehicle counting equipment, and sensitive sensors embedded in the pavement. These devices are used to determine the availability of parking spaces, and more advanced sensing systems are being developed to analyse and transmit real-time information to a database.

The smart payment system in parking can reduce the need for maintenance and staffing, as well as improve traffic control [3]. This system offers different payment methods, including contact and contactless options, as well as mobile device payments. Contact methods involve using smart cards, debit cards, or credit cards, while contactless methods use Automated Vehicle Identification (AVI) tags with Radio Frequency Identification (RFID) technology. With contact methods, drivers must physically interact with automated parking machines or parking meters to make payment. However, concerns exist regarding the privacy and security of driver

information, as debit and credit cards contain personal and confidential data. To address these concerns, various solutions such as cryptography, detection and evasion techniques, and temporary deactivation methods have been developed to enhance data security and protect against potential threats [5]. Implementing these solutions ensures that drivers' data remains secure and minimizes the risk of unauthorized access by third parties**.**

2.3 The Common Problem on Car Parking System Nowadays

Insufficient information on parking availability and pricing, lack of updates on parking space availability, inefficiency in ticketing processes, and inadequate signage are prevalent issues in car parking systems. Many conventional systems fail to display parking prices, leading to customer dissatisfaction and surprise charges. Drivers waste time searching for parking due to the lack of real-time updates on available spaces. Inefficient ticketing methods, such as requiring drivers to roll down their windows, result in long queues, especially during busy periods. Additionally, inadequate signage poses a risk to drivers as they navigate through the parking area, causing confusion and potential accidents.

2.4 Type of Parking System

As in this era, the parking system develops from time to time. A modern parking solution that is transforming urban parking. This modern method uses mobile apps, artificial intelligence, and sensor technology to optimise parking procedures. With real-time data, drivers can easily find available spaces, which eases traffic and boosts overall productivity. In order to ensure a great user experience, Smart Park includes contactless payment methods. The technology reduces environmental effect by increasing utilisation of space with sustainability in mind. With Smart Park, an innovative technology that redefines sustainability, accessibility, and ease of use in urban settings, you are able to accept the future of parking.

2.4.1 Manual parking coupon

Parking reservation is a manual method involving physical coupons. These coupons are purchased by the public and displayed on their car's dashboard to indicate their right to use a parking space. However, this method faced challenges related to user ethics, parking management, and resource wastage. Users sometimes used already used coupons, highlighting ethical concerns. To tackle this, council staff manually checked the displayed coupons for illegal parking. However, this solution proved inefficient, consuming time and labour resources. Additionally, the coupon system itself required mass production of paper coupons, further adding to resource consumption. From the user's perspective, finding available

parking spaces and the need to physically scratch open the coupons added to time consumption and inconvenience.

2.4.2 Automated Parking System

The automated parking system is designed to allow drivers to bring their vehicles to a designated bay, where computer-controlled mechanisms take over and place the cars in allocated spaces. This system is more efficient than conventional parking lots as it maximizes car storage capacity by stacking vehicles on top of each other. It offers additional safety measures as patrons do not need to enter the parking area [9]. Ensuring the vehicles are returned undamaged is a crucial aspect of this system. To achieve this, a three-level software design consisting of the Logical Layer, Safety Layer, and Hardware Abstraction Layer is implemented. Various components such as conveyer belts, rotatable lifts, and shuttles work in coordination to ensure the successful and safe storage and retrieval of vehicles.

2.4.3 E-Parking System

E-parking offers the advantage of allowing drivers to check the availability of parking spaces and make reservations at their desired destination. This system ensures that drivers can confirm the availability of vacant parking spots before they arrive at the parking facility. E-parking can be accessed through SMS or the internet, providing convenient options for users. Additionally, this system incorporates smart payment methods, eliminating the hassle of payment for drivers. Reservations can be made

through mobile phones [4] or internet-enabled devices such as PDAs or computers. Some researchers have even enhanced the system by incorporating fuzzy logic, allowing drivers to accept or reject parking reservations based on their preferences.

2.4.4 Smart Car Parking With Reservation System Using QR Generator

In major cities worldwide, the scarcity of parking spaces has become a significant issue due to the increasing number of vehicles. This problem is further compounded by the fact that even low-income individuals now own cars. To address this challenge, smart car parking systems (SPS) have been developed. These systems aim to optimize parking space utilization and reduce the need for manual intervention. The proposed SPS in this project focuses on providing an efficient and autonomous car parking system controlled by an Android application. Users can register through the app, book available parking slots, and receive confirmation with a QR code. The system also incorporates business rules such as grace periods for reservation, offering alternative spots if available, and notifying customers of reservation expiration. The architecture consists of user and server modules, facilitating communication, reservation management, and billing. The SPS aims to provide certainty in finding parking spots while maximizing utilization in high-density areas.

The SPS project proposes a smart car parking system to address the insufficient parking space issue in urban areas. It introduces an Android application for users to register, book parking slots, and receive confirmation via QR codes. The system implements business rules such as

grace periods, alternative spot offerings, and notifications for reservation expiration. With a user module for communication and a server module for administration, the SPS optimizes parking utilization and provides a reliable parking solution in high-density areas. Below shown the Figure 2.1 of Proposed Architecture of The Smart Parking System with Reservation

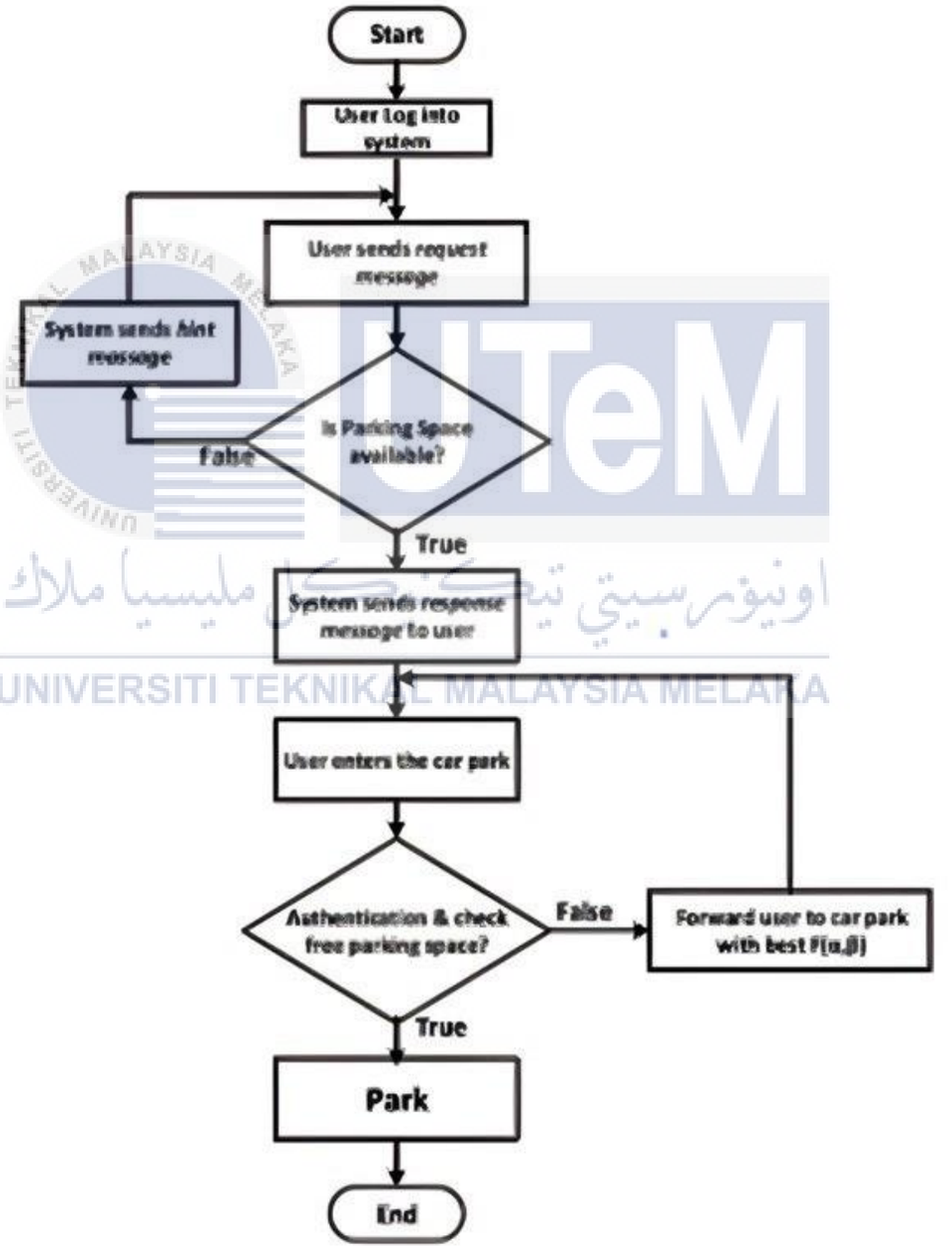


Figure 2-1 Proposed Architecture of the Smart Parking System with Reservation

Inefficient parking management is essential for a positive mall experience. Conventional systems lack real-time availability information and specific parking slot details, resulting in congestion as drivers search for preferred spots near the entrance. Payment is typically made upon exit with varying fee structures, and some malls still use manual ticketing. Researchers propose improvements using floor plans and swim lane diagrams. The enhanced system features reserved parking accessed through an app, confirmed by scanning a QR code. Upon leaving, drivers check the app for their total parking duration and make payment accordingly. Scanning the QR code again clears their data from the parking system.

This research paper introduces the Smart Valet Parking reservation system (SVPRS), which aims to replace traditional valet services in various locations like landmarks, airports, malls, restaurants, and cafes. The SVPRS comprises a hardware device, an online platform, a cloud-based server, and a mobile app for user access. Through the app, users can check available parking spaces, select one, and make a reservation by paying online. Each reservation is assigned a unique QR code, which is sent to the user for scanning upon arrival at the parking lot. LED lights at the parking lots indicate their availability (green) or occupancy (red). The system utilizes a smart objective detector to detect the presence of a car, exchange real-time data with the server, and control the LED lights. An alerting device is installed to remind users to scan the QR code within 3 minutes.

The system can be connected to the Road and Transport Authority (RTA) and traffic police to enforce parking reservation policies and issue fines for violations. If a reserved parking spot remains unoccupied after a

20-minute waiting period, the reservation is cancelled, and a partial refund is given. Additionally, the SVPRS offers GPS guidance to direct users to their reserved parking spot and back to their vehicle, facilitated by a pre-designed algorithm developed by the app company. Overall, this smart parking solution aims to improve convenience, reduce search time, encourage social distancing, generate revenue for city management or valet companies, and utilize data-driven cyber-physical systems (CPS) principles. Below shown Figure 2.2 Smart Valet Reservation Parking System (SVRPS) Smartphone Application Interface and Figure 2.3 Hardware of SVRPS components.

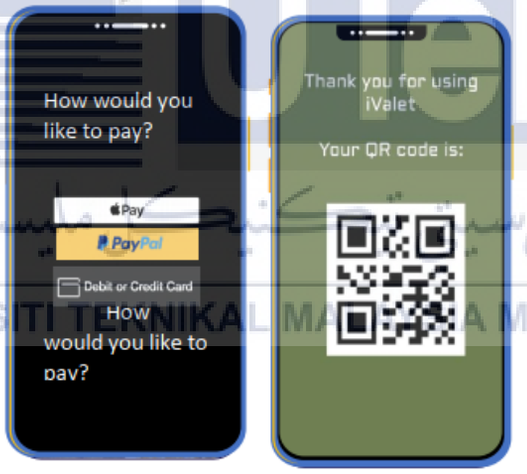


Figure 2-2 Smart Valet Reservation Parking System (SVRPS) smartphone application

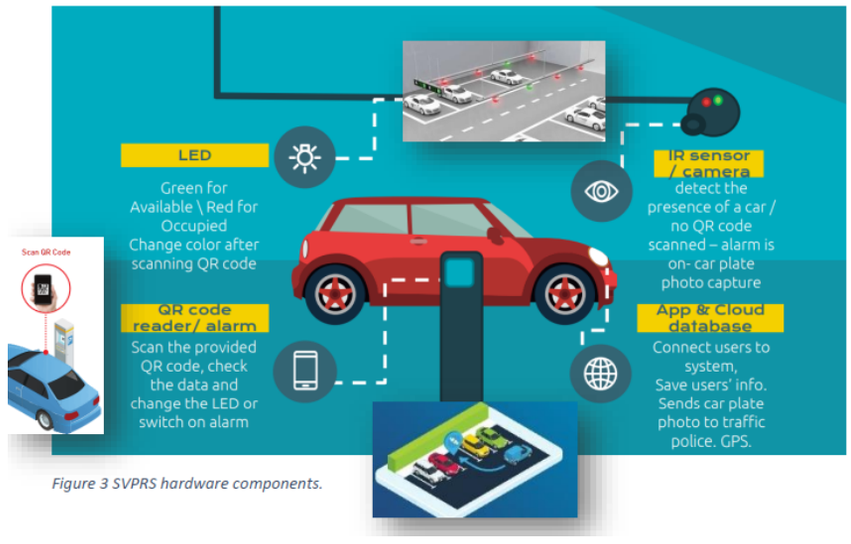


Figure 2-3 Hardware of SVRPS components.

2.4.5 Smart Parking System Using IoT

This research paper focuses on the issue of finding parking spaces in smart cities, particularly in densely populated areas like metro cities. The growing number of vehicles and inadequate parking infrastructure create difficulties for drivers in locating available parking spots. To address this problem, the authors propose an Intelligent Parking System (IPS) that utilizes an Internet of Things (IoT) framework to gather real-time data and suggest suitable parking locations to users. The system includes a mobile application that enables users to check parking spot availability and make reservations. The study explores various scenarios for finding and parking vehicles in the correct places. The IPS is implemented using Raspberry Pi, NodeMCU, RFID, and IR sensors. The paper discusses the findings, demonstrating the effectiveness of the IPS. The introduction highlights the parking challenges specific to Indian cities and the need for an efficient parking solution.

The proposed framework aims to monitor parking space availability in real-time, display nearby parking options to users, and facilitate

reservations. It addresses the difficulties faced by non-local individuals and those with time constraints, preventing them from parking illegally and saving their valuable time. The system benefits both users and parking space owners. The architecture consists of three main components: application, cloud, and interface. The mobile application provides real-time data obtained from a cloud database, indicating the availability of parking slots. The authors' contributions include the development of a unique mobile application for the smart parking system and the proposal and implementation of various use cases. The paper provides a detailed description of the methodology, materials used, simulation, and experimental results. It concludes by discussing potential future directions for the research. Below shown Figure 2.4 Architecture of The Smart Parking System

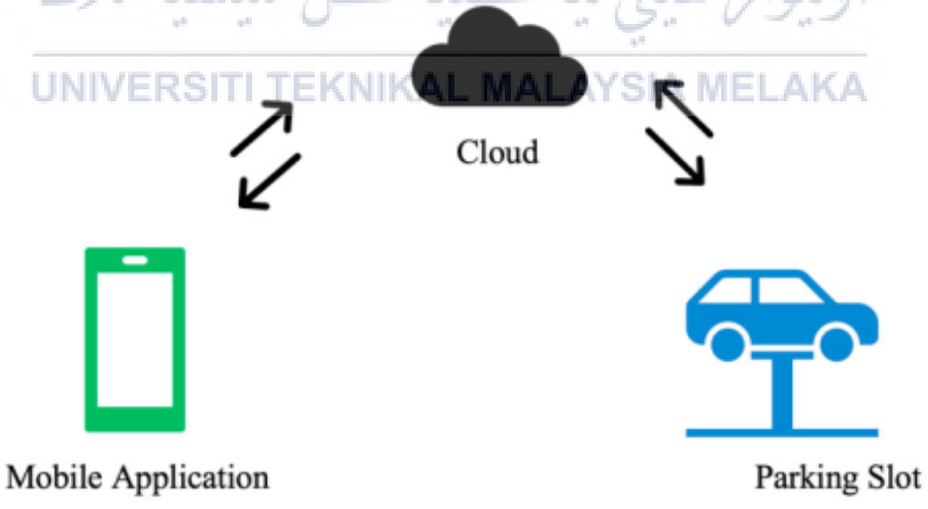


Figure 2-4 The Architecture of the Smart Parking System

2.5 An IoT- based E-Parking System for Smart Cities

The increasing number of vehicles on the road and the mismanagement of parking spaces have caused parking issues and traffic congestion. To tackle this problem, an automated smart parking management system is needed. This system would assist drivers in finding suitable parking spaces, leading to reduced fuel consumption and air pollution. Previous research has focused on features such as checking parking availability, making reservations, real-time navigation, and detecting occupancy. However, limited attention has been given to real-time identification of improper parking and automated payment collection. This paper presents an IoT-based E-parking system that incorporates a parking meter component. It enables immediate identification of improper parking, estimates the duration of parking for each vehicle, and automates the payment collection process. Additionally, the system offers city-wide smart parking management, including information on parking availability and reservation systems. It is known as Parking Meter-based E-parking (PM-EP). Below shown Figure 2.4 Network Architecture of Proposed System.

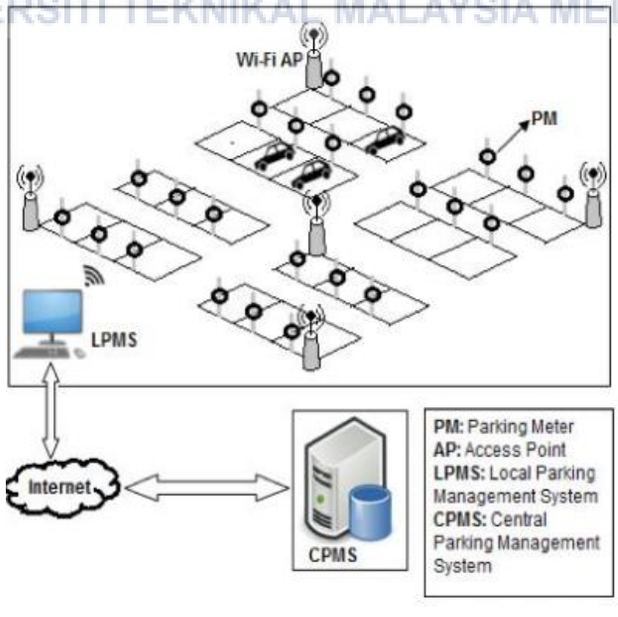


Figure 2-5 Network Architecture of Proposed System

2.6 Developing a Smart Parking Management System Using the Internet of Things

Inefficient parking search processes lead to wasted time, effort, and costs, particularly for individuals who prioritize punctuality. Smart cities utilize modern technologies to efficiently manage urban parking facilities. To address these challenges and optimize parking management, a smart parking management system (SPMS) has been developed. The primary focus of the system is to enhance the search for available parking spaces and alleviate congestion at entry points. It enables users to conveniently search and book parking spaces online in advance. The SPMS integrates a range of technologies to streamline parking management, offering services such as parking search, reservations, and payment processing. Advanced features encompass notifications, statistical analysis, and real-time monitoring of parking availability. The system employs occupancy sensors and automatic number plate recognition (ANPR) cameras for effective access control. Below shown Figure 2.5 Entity Relationship Diagram of Smart Parking System.

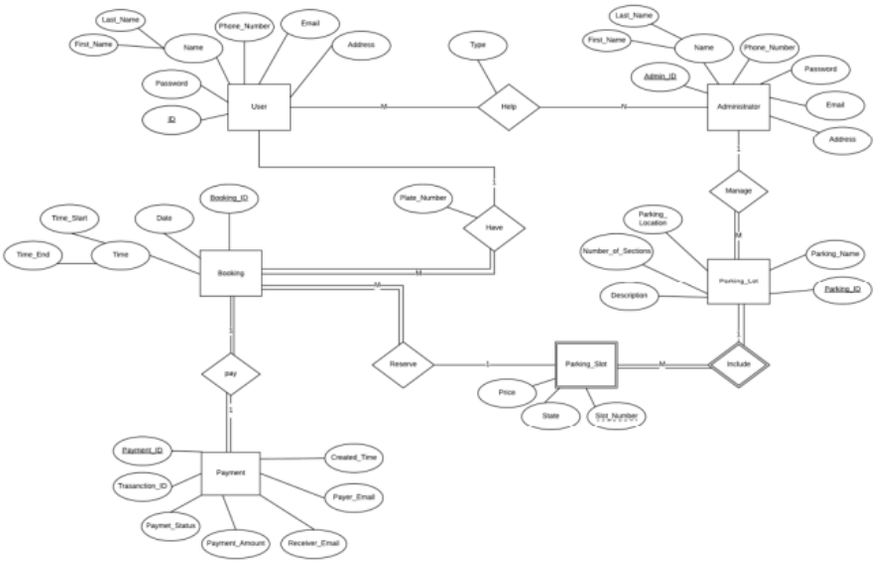


Figure 2-6 Entity Relationship Diagram of Smart Parking System

2.7 An iot based intelligent system for real-time parking monitoring and automatic billing

The parking industry is undergoing a transformation with the adoption of new technologies aimed at reducing congestion in cities. Smart parking systems utilize sensor networks to detect vehicle occupancy and provide real-time information about available parking spaces. Various types of vehicle detectors, including inductive loops, acoustic sensors, infrared sensors, and ultrasonic sensors, are used to collect parking data. However, video camera sensors have limitations related to weather, cost, and data transmission. Magneto-resistive sensors combined with wireless networks are popular due to their accuracy, but they face challenges with electromagnetic interference and battery life. To address these issues, a parking sensor system has been proposed that optimizes energy consumption and achieves high accuracy in monitoring occupancy by utilizing darkness and Signal Strength Indicator (RSSI) measurements. Despite advancements, existing wireless sensors remain costly and intrusive to install.

Research on smart parking payment systems is limited, but some companies are developing patents for payment solutions using cameras or RFID transceivers. However, these approaches can be complex, expensive, and may not provide information on available parking spaces. Another approach involves using two cameras to record vehicle entrances and exits, along with a method for obtaining and displaying information on vacant parking spaces. This paper introduces a smart sensor system that enables

outdoor parking monitoring and payment without user interaction and without the need to install components on each parking lot. The proposed system offers advantages such as reliable detection and payment, reduced system complexity and installation costs, and extended battery life through decreased power consumption. Below shown Figure 2.6 Proposed System Architecture, Wireless Occupancy Sensor, Wireless Gateway, Data Storage and Processing Unit.

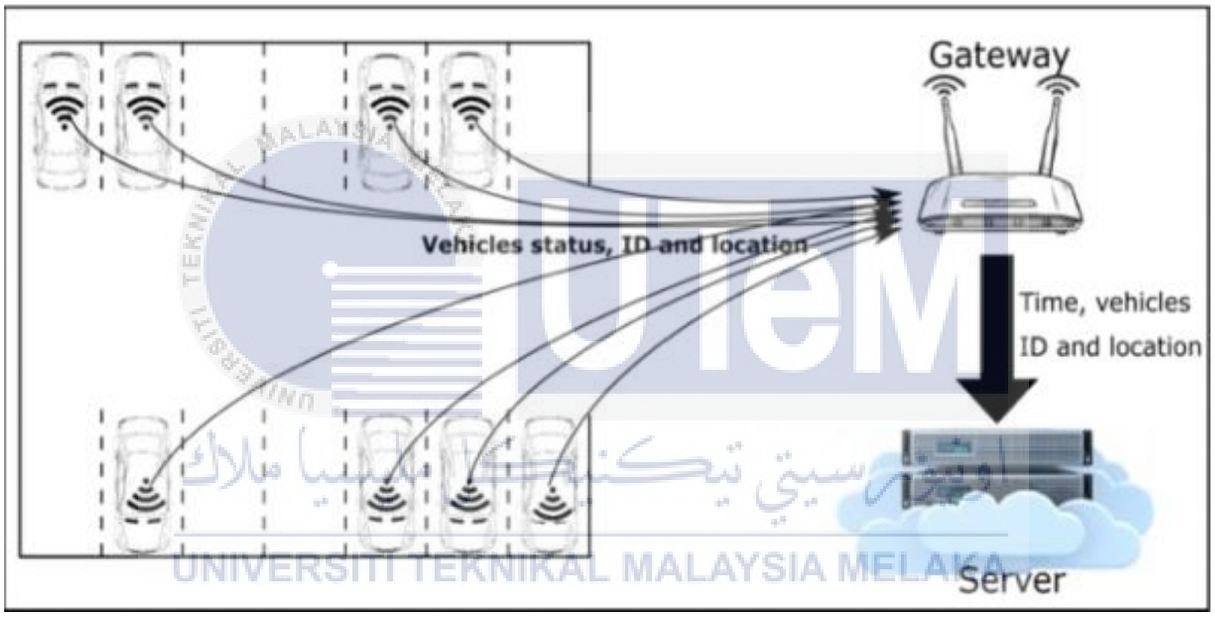


Figure 2-7 Proposed System Architecture, Wireless Occupancy Sensor, Wireless Gateway, Data Storage and Processing Unit

2.8 Analysis of parking system used in shopping mall

In Malaysia, various shopping malls utilize different types of smart parking systems. The selection of these systems is based on factors such as the mall's location, purpose, benefits, drawbacks, and suitability. Table 2.1 provides information on the specific types of smart parking systems employed in shopping malls across Malaysia.

Table 2-1 Type of smart parking systems used in shopping mall in Malaysia

Shopping Mall	Smart Parking System
1. Suria KLCC	Parking Guidance System
2. Mid Valley Megamall	E-Payment System
3. Sogo	E-Payment System
4. Quill City Mall	Qsmart Parking System
5. Plaza Low Yat	Smart Payment System

2.9 Benefits offered by smart parking solutions

In today's digital era, people heavily rely on on-demand applications that have become an integral part of their daily lives. These apps serve various purposes, such as e-Learning, food ordering, ride-hailing, and alarm setting, from morning till night. While there are apps available for car repairs, it only makes sense to have a dedicated app for parking as well. Smart parking solutions have gained significant global demand and are rapidly growing, with a projected compound annual growth rate (CAGR) of 14 percent according to Statista. The development of a car parking app offers several benefits to users, including optimized parking spaces, reduced traffic congestion and related issues, decreased parking time, the ability to find suitable and affordable parking areas, and enhanced vehicle safety.

2.10 Solutions using iot based parking

The smart parking system operates by transmitting real-time information about available and occupied parking spaces through a web or mobile application. Each parking spot is equipped with an IoT device

comprising sensors and microcontrollers. Users can receive instant updates on the availability of parking spaces and make informed decisions about which spot to choose. This solution offers a range of benefits, including reduced traffic congestion and improved fuel efficiency, particularly in urban areas where parking can be a challenge. The overall process of the smart parking IoT system can be represented by a basic flowchart. Below shown Figure 2.8 Flow Chart of IoT Based Parking

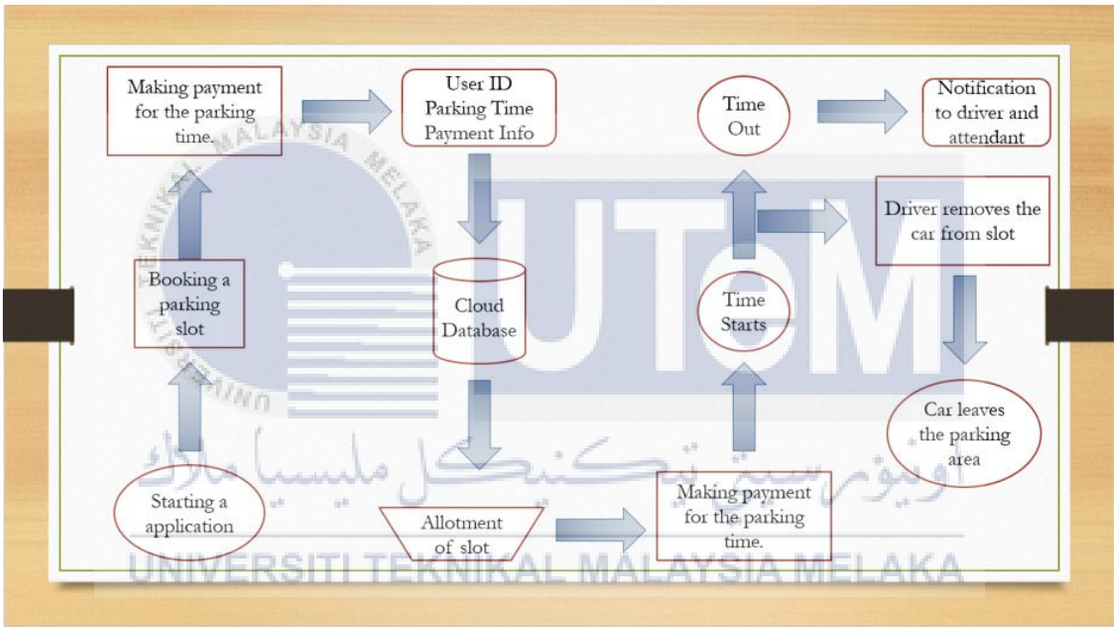


Figure 2-8 Flow Chart of IoT Based Parking

2.11Comparison of previous studies

From Table 2.2 shows previous studies that have different ideas for implementing related projects. There are several article references searched to know more and better about parking system reservation.

Table 2-2 Comparison of Previous Studies

Article Reference	Component	Implementation	Software	Advantage and Disadvantage
Smart Parking System-Belgium	NodeMCU ESP8266, IR Sensor, Servo Motor	Controller with IoT	Blynk	ADVANTAGE - Saving financial cost -Capable for android user DISADVANTAGE -IR Sensor nee to clean regularly -Not capable for IOS
A Smart Parking Reservation System-iValet	Camera, QR Code Reader, Alarm, IR Sensor	Controller with IoT	GPS, Android OS	ADVANTAGE - tracking the owner car. -Capable for Android

				DISADVANTAGE -High cost -Not Capable for IOS
Smart Application for car Parking System-NAKHON	Arduino UNO, IRSensor, NodeMCU ESP8266	Controller with IoT	MIT App Inventor	ADVANTAGE -Cormmercialist DISADVANTAGE -Quite expensive
An IoT Assisted Intelligent Parking System (IPS) for Smart Cities.1	Raspberry Pi, NodeMCU ESP8266, RFID and IR Sensor	Controller with IoT	Wi-fi	ADVANTAGE - Cormmercialist. -Capable for IOS and Android DISADVANTAGE -Quite Expensive

2.12 Summary

Based on the previous researcher works and the theories, these information that related to this project gained knowledge about the methods used by the previous researchers. Therefore, comparison of methods applied, advantage and disadvantage are made to show the similarities and differences methods applied by previous researchers. Lastly, this reservation parking system give an accurate about data of availability parking space in real-time to user so they able to know the real status of the parking space through smartphones over the Wi-Fi connectivity.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter, this chapter describes the methods for completing the project over time. The goal of this chapter is to provide information and confirmation about how the project was completed. To develop this project the sequence from designing and developing of reservation parking system using app capability will be shown step-by-step.

In this chapter, the software and hardware which have related to the design and development of reservation parking system using app capability will be stated out too. The design and construction of reservation parking system using app capability and mainly on detection the number of parking space that available and display it on the application. Otherwise, it can make a reservation thru apps and receive the notification thru the application.

3.2 Selecting and Evaluating Tools for a Sustainable Development

It is critical to carefully pick and evaluate the tools and technologies that will be used to collect and analyze data when developing and implementing a PMS project with a focus on sustainability. This includes a variety of methodological concerns, such as checking the accuracy and reliability of sensors, evaluating the compatibility of various tools and software, and taking into consideration of the project's environmental impact. Furthermore, it is critical to address the project's social and

economic implications, such as ensuring that the data is accessible and intelligible to a wide range of stakeholders and weighing the costs and benefits of various tool options.

3.3 Block Diagram

A block diagram helps with the project's concept. It almost resembles a flowchart but with a more straightforward presentation. Block diagrams commonly represent the intake, process, and output of design projects while also simplifying the parts and resources that were utilized.

3.3.1 System Block Diagram

Figure 3.1 shows the microcontroller's block diagram that contains the entire explanation of the project concept from input to output. The microcontroller ESP32 which is the major component of this project is utilized to regulate the input and output data throughout all of the other components. It will collect and transport data, as well as gather and provide commands. For the input, there are Ultrasonic. The function of Ultrasonic is to detect an object while the ESP32 will be used for networking and data collection for the software. All sensor data will be uploaded to the software from the ESP32, which will then transfer the data to the output which is to the software, where the data will be saved there from the software to the apps on the phone. For another output, the data will cause the component to function according to the program that has been selected. The command for the red lamp and green lamp is to indicate either the parking space available

or not. While, the LCD display will display the available parking space on that time and for servo motor use to turn down the gate when the owner of reservation come to park their car.

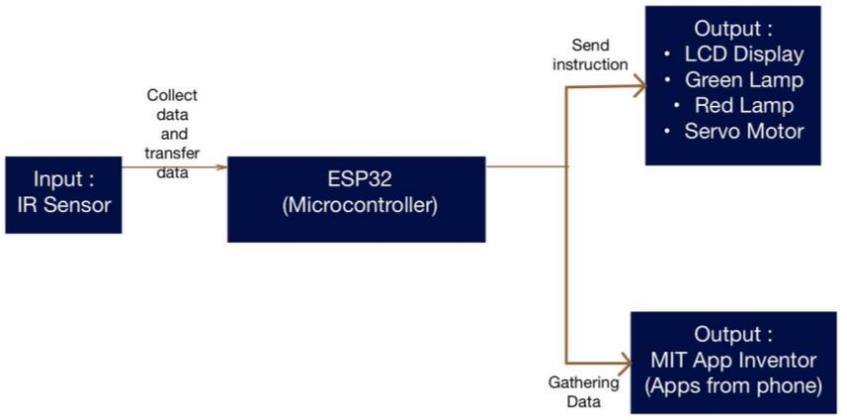


Figure 3-1 System Block Diagram

3.4 Project workflow

Project flowcharts are used to visually represent and explain the system's flow. The flowchart accomplishes its goal in the most basic way by defining the project's working criteria. Using flowcharts, the processes of the project had to be laid out as a diagram that depicted the system from the beginning to the finish

3.4.1 Flowchart of the project

It is very important that to make sure the progress of this project is followed step by-step and complete the tasks that arranged in schedule. Therefore, a flowchart will be drew. A workflow or process can be graphically represented using something called a flowchart. It involves carrying out a series of distinct steps in a predetermined order. In most cases, a flowchart will display the steps as a series of boxes of varying types. Steps are linked so that anyone can examine the flowchart and follow its directions in the appropriate order from the very beginning to the very end. This is

accomplished by connecting lines and directed arrows. It is also useful for a variety of other objectives, including as documenting, Utilizing a flowchart allows for the development of an understanding of how a process is carried out. Figure 3.2 shows the flowchart of this project

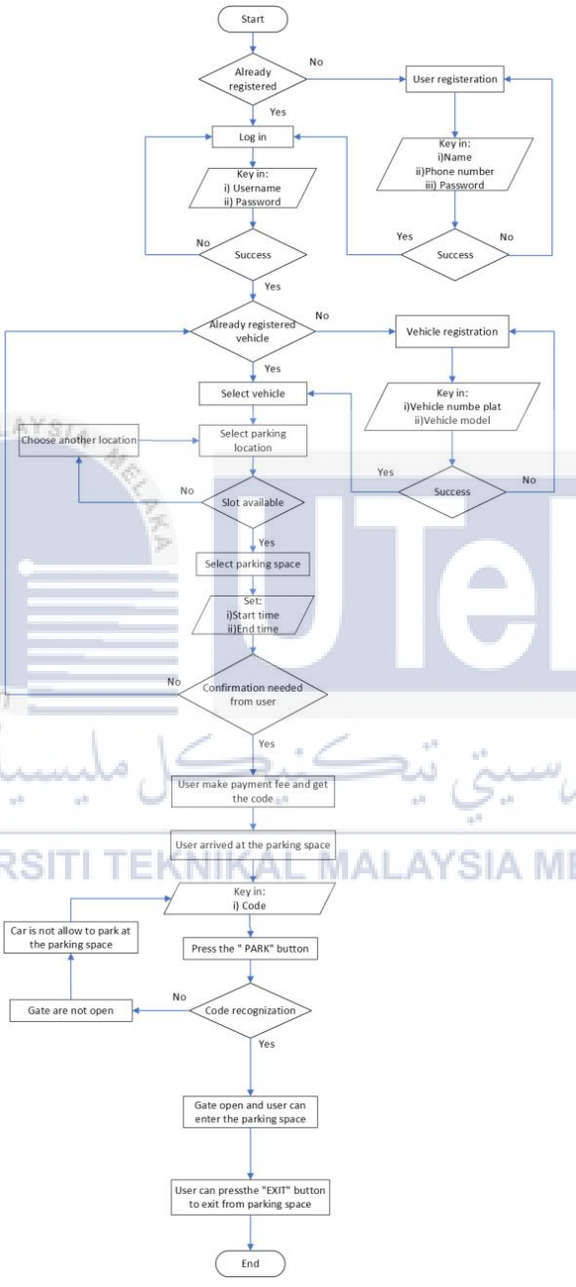


Figure 3-2 Flowchart of project

3.4.2 Flowchart of the registration user to the application

By referring Figure 3.3 below, shows the flowchart of registration which is start with the registration from user first before go thru to the Log in process. While in the registration process, user should key in their name, phone number, email for safety purpose, receiving any information about the reservation parking spot and else. The, the user also need to key in their password n plat number. Next, if the process successful the user may go to Log in process to log in the application by insert their used id that indicated as their full name and password that have been set by user self. Otherwise, if the user is not successful the registration process, user need to repeat the same step until finish for the registration process or user can cancel and exit from the application without register anything. By log in the application, user can proceed to the next step which is reservation process.

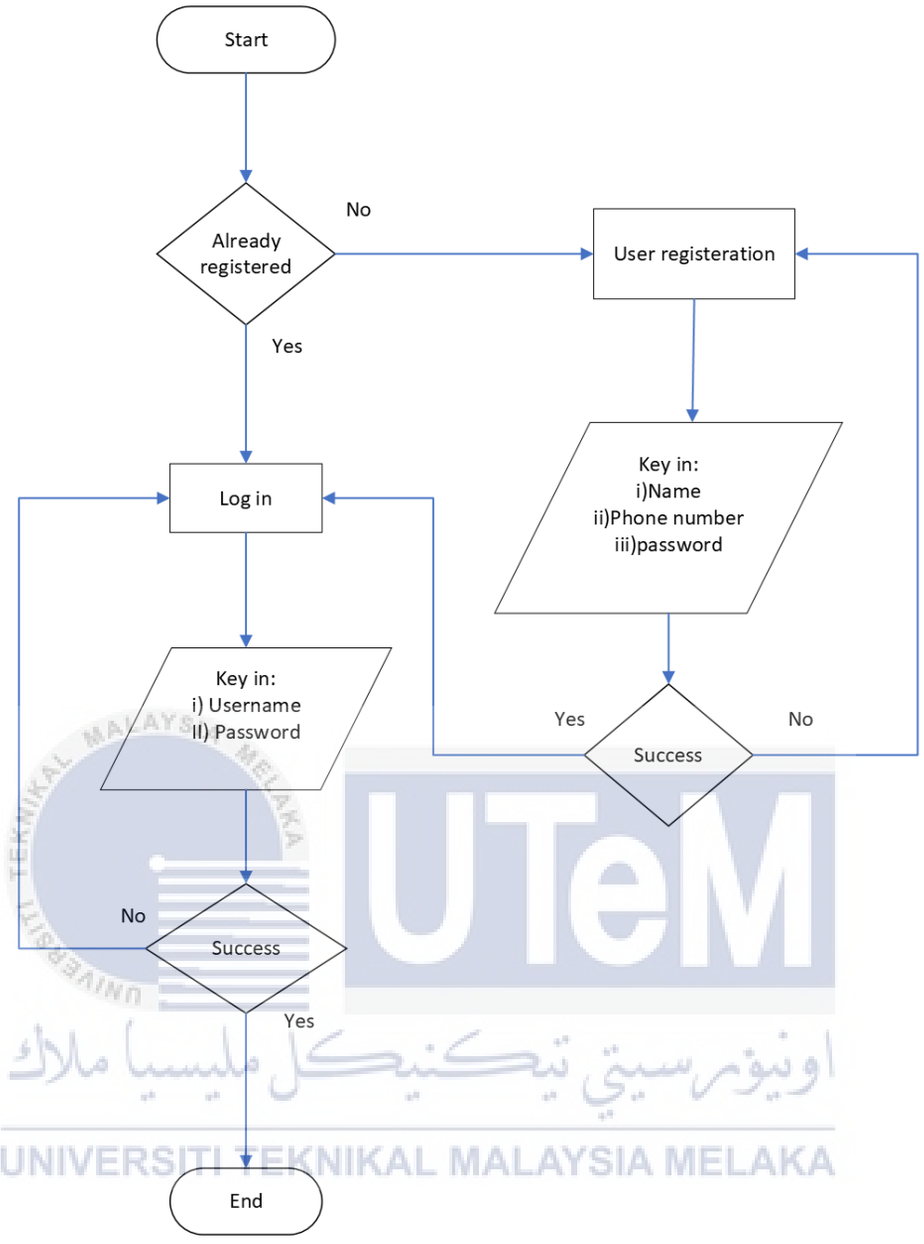


Figure 3-3 Flowchart of Registration

3.4.3 Flowchart of the reservation parking space using application

By referring Figure 3.4 below, shows the flowchart of reservation parking space using application which is when the user success in registration and log in process, the user can start by selecting the location of parking. After that, if the location of parking that have been selected by user have the available parking space, the user need to insert the details of start time and end time for the reservation but if there is no slot available at the parking location that have been selected, the user may choose another location of parking until find the location of parking that have available parking space. After finish key in the start time and end time process, the user need to do some confirmation about the reservation details such as the location have been selected, parking slot that have been given and the duration of the reservation before proceed to the payment process. After finish the payment process, the user will get the code of reservation parking slot that have been reserved for entering the parking slot purpose.

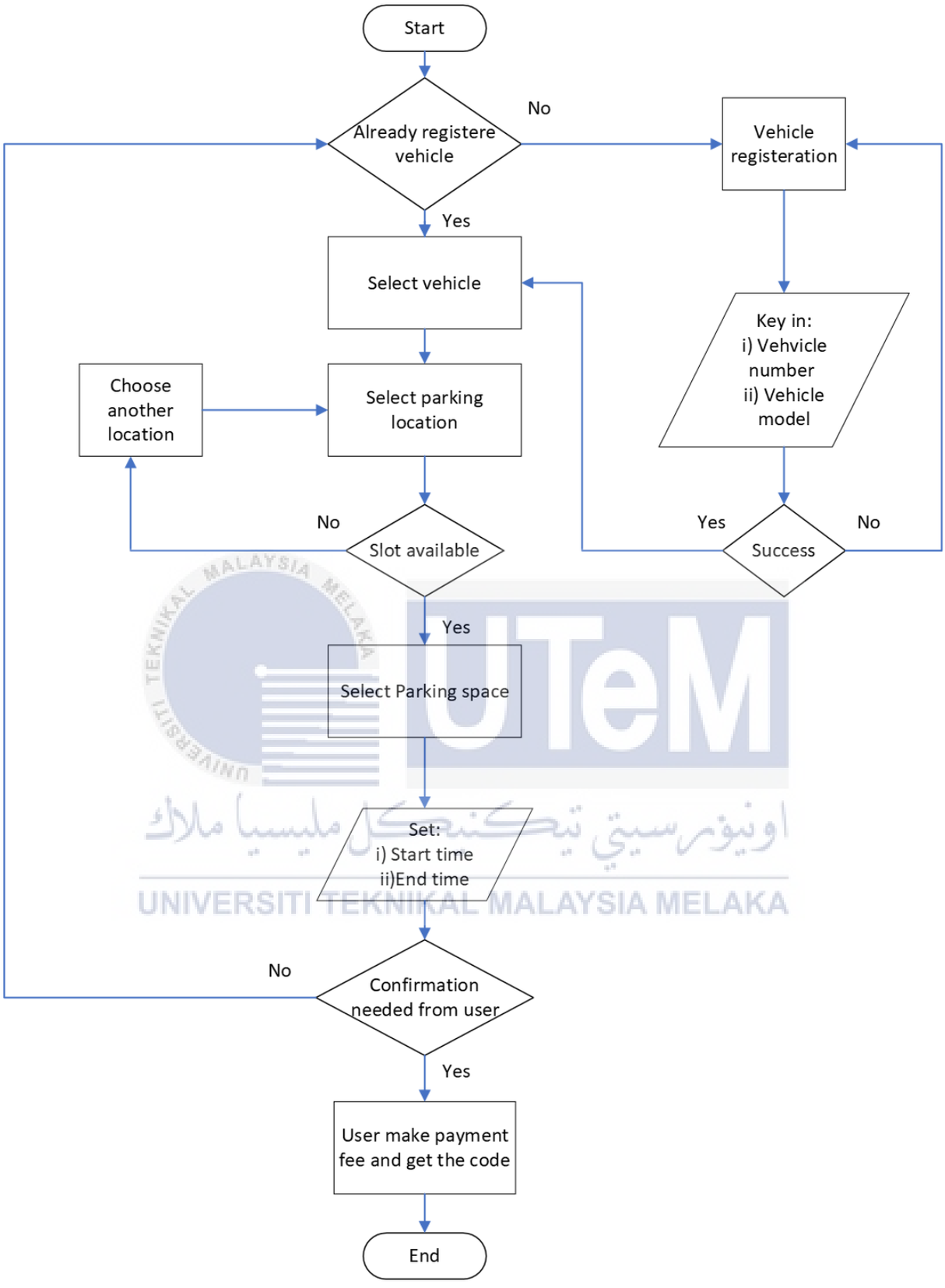


Figure 3-4 Flowchart of Reservation

3.4.4 Flowchart of the reservation parking space using application

By referring Figure 3.5 below, shows the flowchart car enter the parking space and out from parking space. In this situation user must be already make the reservation parking space thru the application that have been provided. When the user are arrived at the parking space that have been booked, the user need to key in the code that already received at the end of the reservation process and press the button 'PARK' that already display at the application interface. When the code is recognized the gate will open and user can park the car at the parking space that have been booked and a few second the gate will close back. If the code is not recognize, the gate are not open and the car is not allowed to park at the parking space. If in this case, the user need to key in again the code correctly. If the user want to exit the parking space the user need to press the "EXIT" button to exit from the reservation parking space. As usual, the gate will open and user can exit the car at the parking reservation space that have been booked and a few second the gate will close back

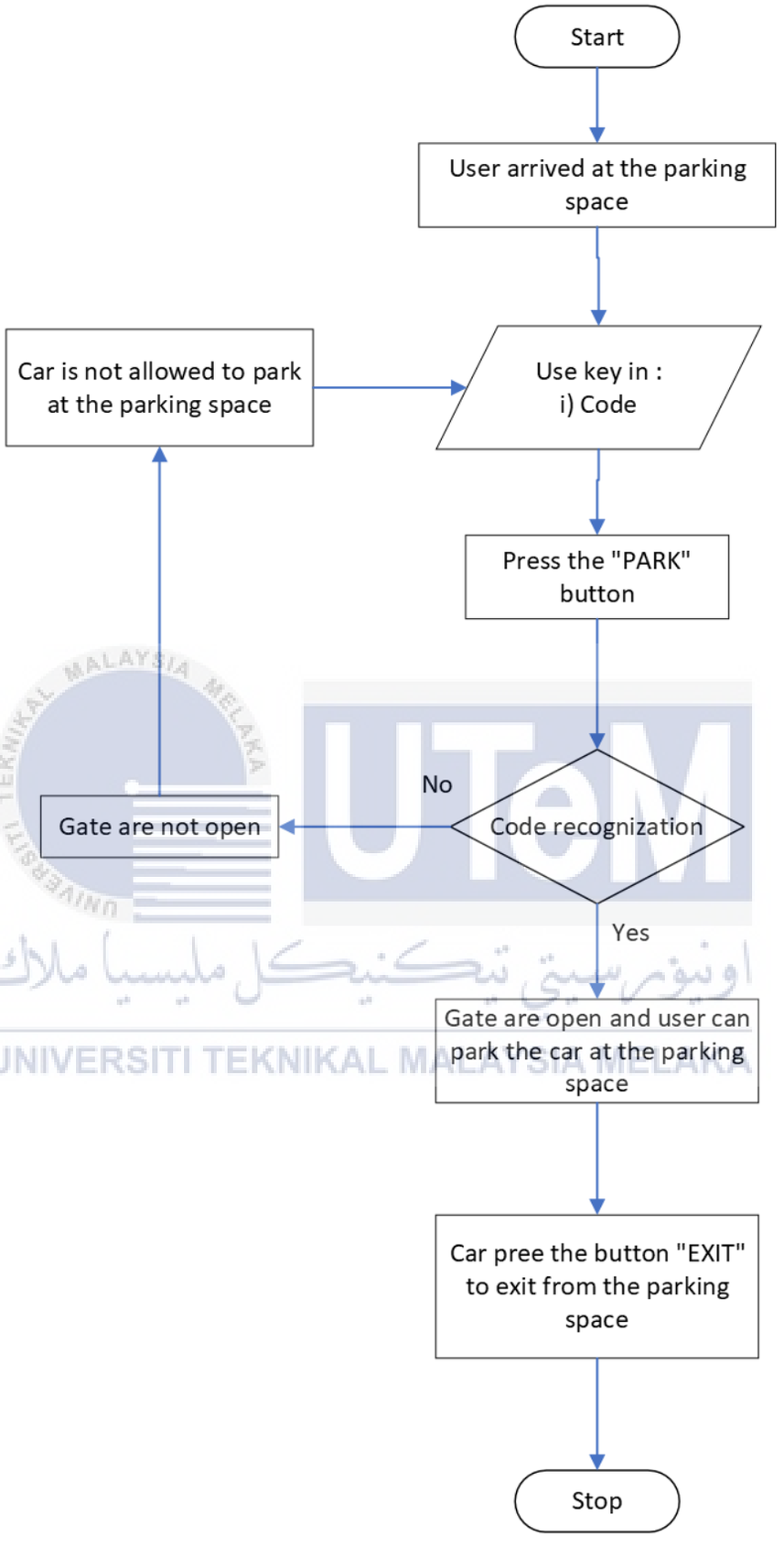


Figure 3-5 Flowchart of Car Enter the Parking Space

3.4.5 Development of IoT in reservation parking system using app general process flow

Figure 3.6 shown the process of the project which is it will be start by starting the program. Once the program is begun, it will establish a link between the controller and the network, where the sensor parameters will be set and read before it begins verifying the process that send to the ESP32 and application. The system will examine the output data to determine the status of parking space. If the parking space is not available, the ESP32 will send the data to the application interface of smartphone and LCD display by using IoT or specific name is Internet. As we know the IoT things that have been created and embedded with the sensor, software and other technology components that function to connecting and exchange the data from the device and the system over the internet. Same goes to when the parking slot is available, the same process will be carried out. Simultaneously, all gathered data will be transferred to the ESP32 and it will be transmitted to the user interface and LCD display.

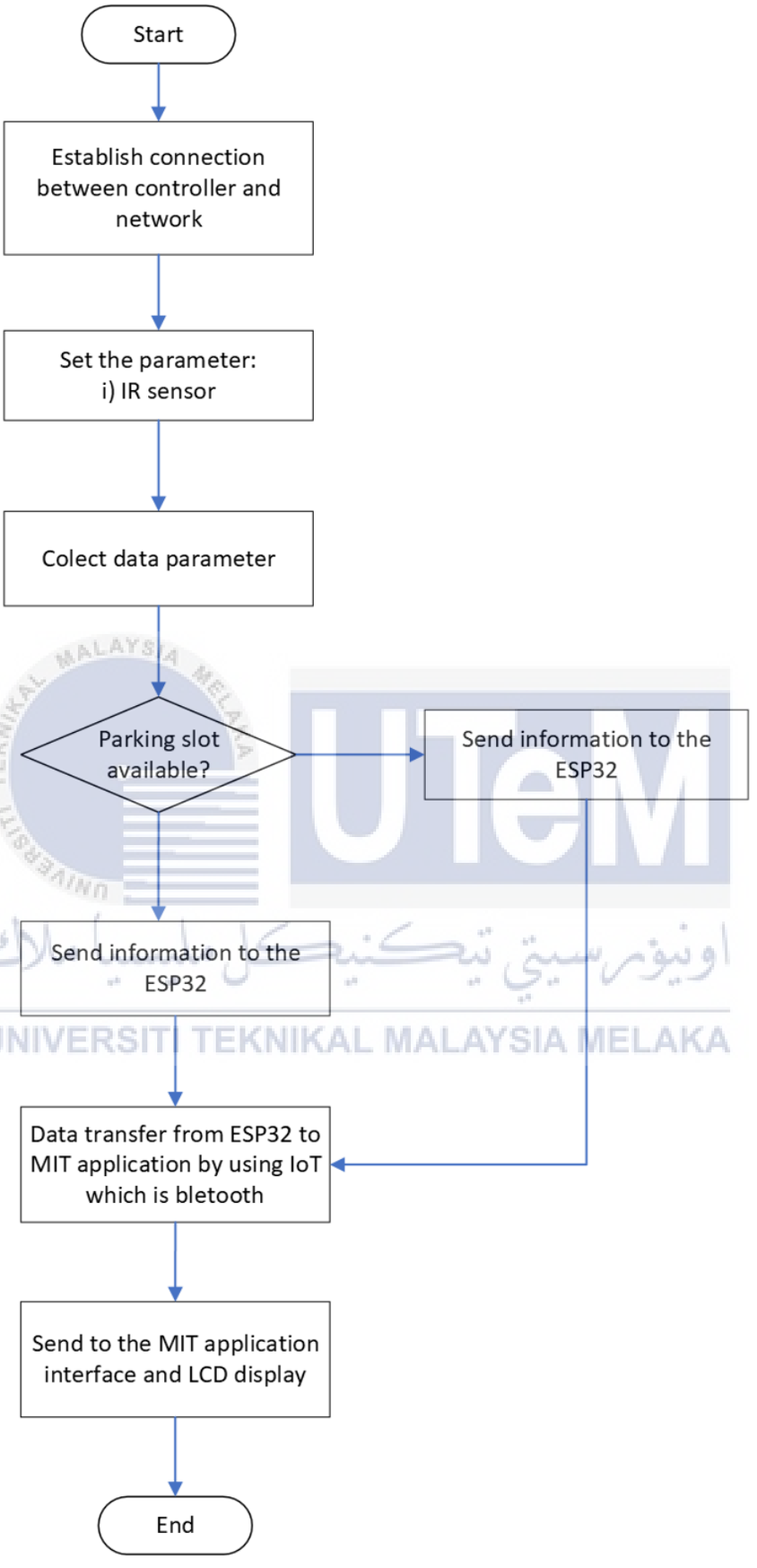


Figure 3-6 Development of IoT in reservation parking system using app general

3.5 Experimental setup

Simulation tests serve as crucial assessments, replicating real-world scenarios within a controlled environment to evaluate system performance, identify vulnerabilities, and validate functionalities. By imitating various conditions and interactions, these tests provide valuable insights into a system's behavior and responses. In this simulation test, Arduino UNO is the main component and act as microcontoller and Ultrasonis (HC-SR04) as proximity sensor in this test simulation. I fact, for the real hardware was ESP32 as microcontroller and IR sensor as proximity sensor.

i. The sensors are connected to the microcontroller board based on their respective pin configurations as shown in figure 3.6 below.

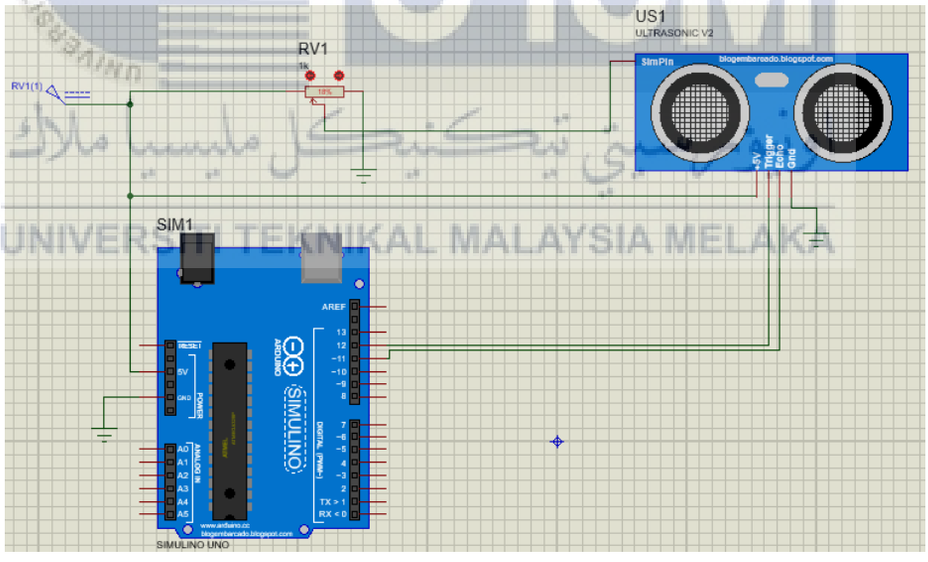


Figure 3-7 Connection Ultrasonic Sensor to the Microcontroller Board

ii. Software code is written to read data from sensors, and establish communication with the chosen IoT platform that shown in figure 3.7 below.

```
// Define the trigger and echo pins
const int triggerPin = 2;
const int echoPin = 3;
long duration;
int distance;
void setup()
{
   // Initialize serial communication
  Serial.begin(9600);

  // Set trigger pin as output and echo pin as input
  pinMode(triggerPin, OUTPUT);
  pinMode(echoPin, INPUT);
}

void loop()
{
  // Clear the trigger pin
  digitalWrite(triggerPin, LOW);
  delayMicroseconds(2);

  // Send a 10 microsecond pulse to the trigger pin
  digitalWrite(triggerPin, HIGH);
  delayMicroseconds(10);
```

Done compiling.

```
  // Set trigger pin as output and echo pin as input
  pinMode(triggerPin, OUTPUT);
  pinMode(echoPin, INPUT);
}

void loop()
{
  // Clear the trigger pin
  digitalWrite(triggerPin, LOW);
  delayMicroseconds(2);

  // Send a 10 microsecond pulse to the trigger pin
  digitalWrite(triggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerPin, LOW);

  // Read the duration of the echo pulse
  duration = pulseIn(echoPin, HIGH);

  // Wait for a short delay before the next measurement
  delay(500);
}
```

Done compiling.

Figure 3-8 Programming code for Ultrasonic Sensor (HC-SR04)

iii. The microcontroller's software code is distributed to make sure it functions properly and read to collect the relevant data.

3.5.1 Schematic Diagram

By referring to Figure 3.9 below where the schematic diagram of ultrasonic sensor to detect the object or in this projet will be use to detect the presence of the car. For this schematic diagram, the microcontroller was SIMULINO UNO ATMEGA328P as the testing microcontroller before proceed to real hardware which is using ESP32 and for HC-SR04 (Ultrasonic Sensor) also only use for the simulation. For the real hardware will use IR sensor.

The interface of HC-SR04 (Ultrasonic Sensor) and SIMULINO UNO ATMEGA328P will be connected. For the connection, pin 11 of Arduino (~11) will be connected to pin Echo of HC-SR04 (Ultrasonic Sensor). The reasons of connecting pin 11 of Arduino its is because the signal from this sensor will be received from this pin to measure the time taken for the ultrasonic sensor signal to transfer back after detected the object. If the microcontroller such as ESP32, it will be more suitable to connect with D5 (GPIO 2) and same goes to IR sensor to the microcontoller. Any changes of the microcontroller need to refer back the datasheet of microcontroller itself. For the Trig pin of ultrasonic sensor will be connected with pin 12 of Arduino. The reasons of connected pin is because this is the most suitable pin for sending the Ultrasonic signal to initiate the distance measurement. For microcontroller ESP32, it will more suitable to connect with D5 (GPIO 2) of the microcontroller.

Besides that, the VCC of HC-SR04 (Ultrasonic Sensor) will connected to the Arduino pin 5V as to give power supply for the Arduino to turn on. The GND of HC-SR04 (Ultrasonic Sensor) will be connect to the pin GN in

Arduino. Figure 3.9 below shows the schematic diagram of HC-SR04 (Ultrasonic Sensor).

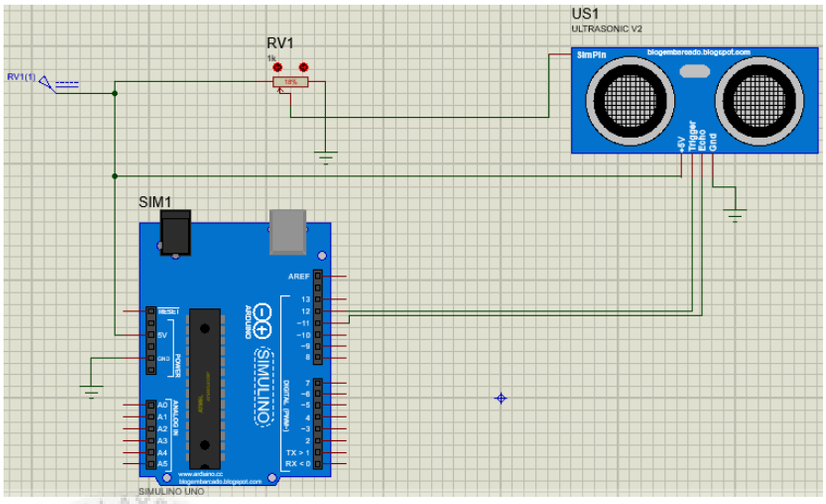


Figure 3-9 Schematic Diagram

Arduino. Figure 3.9 below shows the schematic diagram of HC-SR04 (Ultrasonic Sensor).

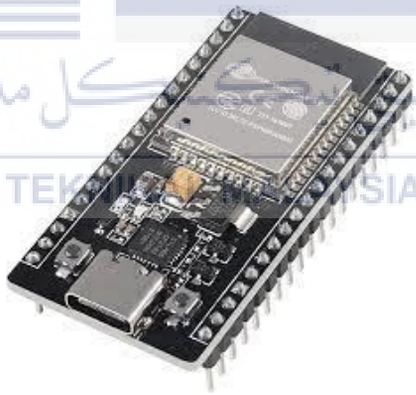
3.6 Hardware

In this project which is Reservation Parking System using Apps is a combination of hardware components perfectly managed to completely change the parking experience. The hardware that have been selected in this project was ESP32 as microcontroller and LCD display to display the availablity of parking space in real time. Other than that, LED as indicator to present for the parking space availablity and IR sensor use to detect the present of the car at the parking space. In additional, ESP32 will

communicate with the Apps software to give a good experience for user to use and provide the real time status.

3.6.1 ESP32

By referring on Figure 3.10 that have been shown below, the ESP32 development board comes is equipped with the ESP32 chip, which indeed contains a Tensilica Xtensa 32-bit LX6 microprocessor. This component act as microcontroller for this project. This component also provide the connection of the internet to the project and can be used as a Wi-Fi that connect to Wi-Fi and as an access point that create a hotspot that allows data to be easily retrieved and uploaded to the IoT. The ESP32 is a standalone microcontroller and Wi-Fi system-on-chip (SoC) developed by Espressif Systems.



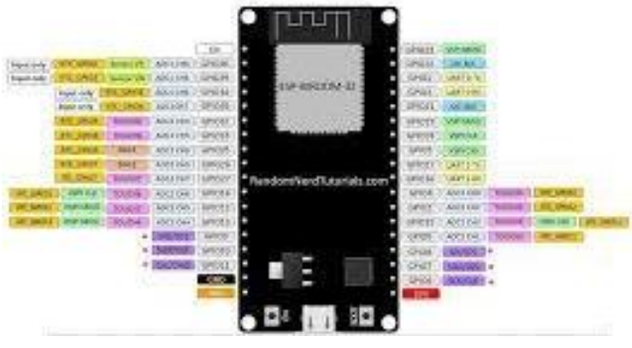


Figure 3-10 ESP 32

3.6.2 IR Sensor (Infrared Sensor)

In the Figure 3.11 below shown the IR sensor. As have been clear that this sensor is commonly used for various purposes, including proximity sensing, object detection, and obstacle avoidance. Infrared sensors can work based on reflection, where the emitted infrared light reflects off an object and is detected by the sensor, or they can work based on interruption, where the interruption of the infrared beam triggers a response. In this project which is reservation parking system using application, the IR sensor detects vehicle presence in parking spaces, triggering actions such as updating availability, reserving spots, and providing users with confirmation and directions through the application. Overall, it enables real-time monitoring and management of parking reservations.

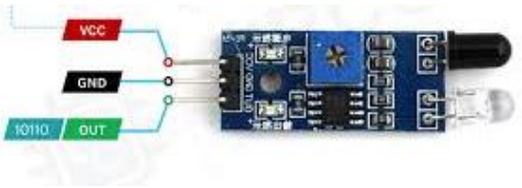


Figure 3-11 IR Sensor (Infrared Sensor)

3.6.4 16 x2 Display LCD

By referring Figure 3.12 that shown below, there a LCD Display 16x2 that commonly use to display some words or numbering and act as output. So, the function of 16x2 LCD display in a reservation parking system using applications serves as an output interface to provide information to users. It displays relevant details of parking availability. The LCD display can show clear and concise messages in a readable format, allowing users to quickly access information about available parking spaces, any other important instructions or notifications. By presenting information directly on the LCD display, users can easily navigate the parking system, make informed decisions, and have a seamless parking experience.

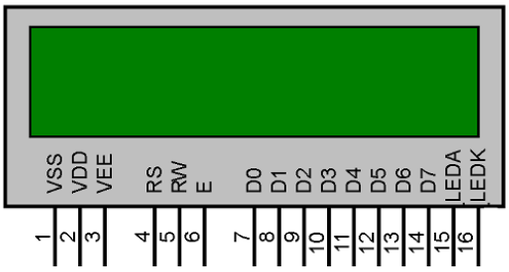


Figure 3-12 16 x2 display LCD

3.6.5 Servo motor 360 degree

Below shown the Figure 3.13 of Arduino Tower Pro 9g SG90 Plastic Gear Micro Servo Motor 360 degree. The function of a 360-degree servo motor in this project which is reservation parking system using applications is to control the movement of gates. The servo motor will be connected to the parking space entrance and allowing them to be opened and closed automatically based on the instructions received from the application. By rotating the servo motor to a specific angle, the gates can be smoothly and precisely operated, granting access to authorized vehicles and ensuring proper security measures. This enables a seamless and automated entry and exit process for vehicles in the parking system, enhancing convenience and efficiency for both users and administrators.

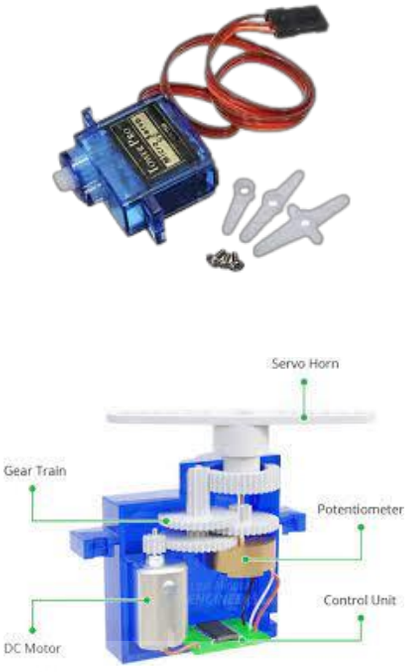


Figure 3-13 Arduino Tower Pro 9g SG90 Plastic Gear Micro Servo Motor 360 Degre

3.7 Software

Software is a set of instruction, data or program that used to operate the system and execute specific task. It different with the hardware, which describe the physical aspects of a system. Software is a generic term used to refer to application and program that run on a device. By that in this project , 3 tools of software that have been use to operate the system which is programming tool, simulation tools and application software tool.

3.7.1 Programming Development

By referring Figure 3.14 Programming Code using Arduino IDE (Integrated Development Environment) Software. For this project coding have been created using using Arduino IDE for programming code. It is because offers several advantages over other programming tools. The Arduino IDE is specifically designed for programming Arduino boards, providing a user-friendly to develop the coding for this project and simplified environment for writing, compiling, and uploading code. Back to our main purpose using Arduino IDE for programming code development in this project is because it is also offers a wide range of libraries and examples that make it easy to interface with various Arduino components and sensors such as that will be used in this project is IR Sensor, Servo Motor, LCD Display and LED. So, from this library the Arduino code can be used to handle various tasks such as reading IR Sensor data, controlling actuators like Servo Motors for gate operation, communicating with the application, and managing the overall system logic. Overall, The Arduino IDE's compatibility with Arduino boards ensures seamless integration and easy code uploading, enabling the implementation of the reservation parking system's functionality efficiently and reliably.

```
GABUNG_SEMUA
#include <LiquidCrystal.h>
#include <Servo.h>

Servo myservo;
const int echoPin = 12; // Echo Pin of Ultrasonic Sensor
const int pingPin = 11; // Trigger Pin of Ultrasonic Sensor
const int rs = A4, en = A5, d4 = A3, d5 = A2, d6 = A1, d7 = A0;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

int led1 = 7;
int led2 = 6;

int iLineNumber = 0;
int iCursor=0;
int iChar;

char * LargeText= " PARKING SPACE AVAILABLE ";

void setup()
{

  Serial.begin(9600); // Starting Serial Communication
  pinMode(pingPin, OUTPUT); // Initializing pin 11 as output
  pinMode(echoPin, INPUT); // Initializing pin 12 as input
```

Figure 3-14 Programming Code using Arduino IDE

3.7.2 Simulation Development

In Figure 3.15 below shown the simulation development that have been used in this project. So, Proteus 8 Professional software being use in this project for simulation part. The reason is because it is a comprehensive simulation and design software widely used in the field of electronics. In this project which is reservation parking system using applications, Proteus allows for the creation and testing of virtual electronic circuits, including Arduino UNO as microcontrollers for testing part, Servo Motor, LED, LCD Display and Ultrasonic Sensors, providing an accurate representation of real-world behaviour. It enables to verify the functionality and performance of the system before implementing it physically, ensuring proper integration and addressing any potential issues. Proteus 8 Professional software aids in

the efficient design and testing of the reservation parking system, reducing development time and enhancing overall system reliability.

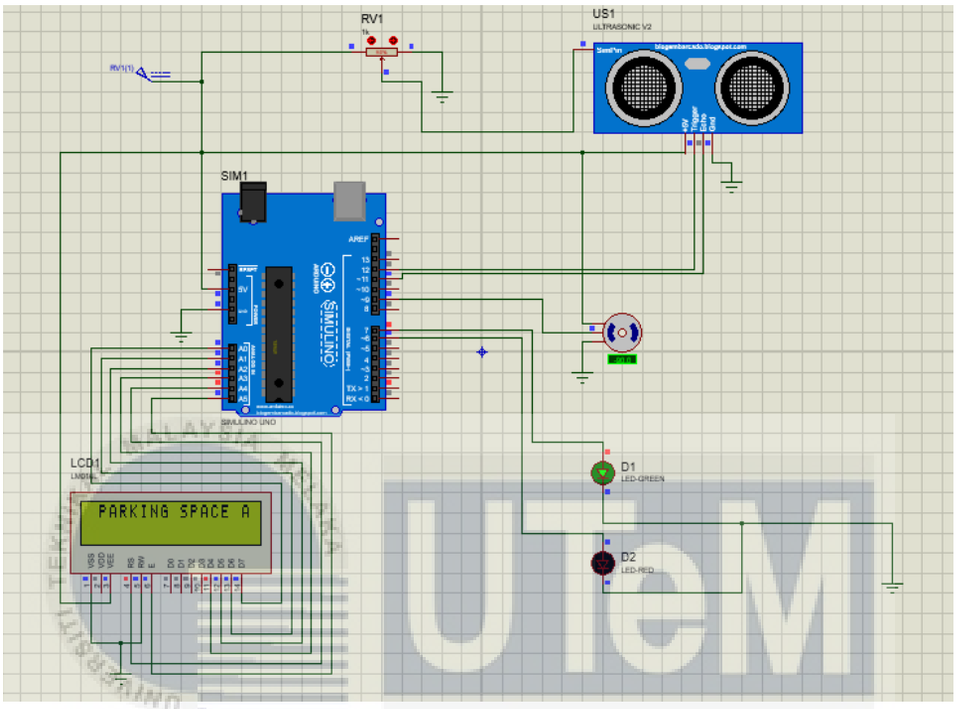


Figure 3-15 Proteus 8 Professional Software

3.7.3 Software Development

By referring Figure 3.16 below, the MIT App Inventor have been selected to use in this project as software to create an applications. In the project which is reservation parking system using applications, MIT App Inventor can be utilized to create the user interface and functionality for the mobile application used by the users. It easy to design screens, define interactions, and integrate features such as real-time parking availability updates, reservation management, and navigation to parking spots. MIT App Inventor's intuitive interface and drag-and-drop functionality enable the creation of user-friendly and engaging applications for the reservation parking system, without the need for extensive coding knowledge. So it can be conclude that MIT App Inventor is the powerful tool for developing mobile applications without need any experience in coding language.

Figure 3-16 MIT App Inventor

3.8 Summary

In this chapter, will represent on which tools will be used for the project and overall progress and process. A flowchart for the process of reservation parking system thru apps, a flowchart for the process of application and a block diagram for system of the reservation parking using app. A quick explanation about the flowcharts included at Figures 3.2, 3.3, and 3.4 so there are detailing about the project and explanation of each process. The Arduino UNO is the primary device that will take data from the input and deliver data to the output according to the block diagram which is also show about the other components functionality for input and output. But, the Arduino UNO and HC-SR04 (Ultrasonic Sensor) only for testing and simulations main component only. For the real hardware and prototype will be ESP 32 and IR sensor.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This section describes the results, analysis, and discussion for all the data collected from the system to determine the performance of the system. In this chapter, it shows the result of hardware, application, and durability of hardware prototype, weather station results with forecast website and the result analysis of weather station.

4.2 Results and Analysis

In this subtopic, there are a few sets of data and graphs will be shown to prove that the prototype of this project is functioning well.

4.2.1 Application System Functionality

System of Application The term "functionality" describes a software system's abilities and functions. The app that have been created for this project is functionality and efficacy in fulfilling user requirements are assessed by taking into account all of the features, actions, and interactions that it can carry out. Assuring the system's effectiveness, usefulness, and overall worth requires an understanding each of its functionality in this Reservation Parking App.

Figure 4-1 Log-In Page

The login screen for a parking reservation app is shown in Figure 4.1. The user interface has an easy-to-use layout with password and username input areas. It also has a "Sign Up" option for new users and a "Forgot Password" link. The website is visually appealing and encourages users to make effective use of the app's features.

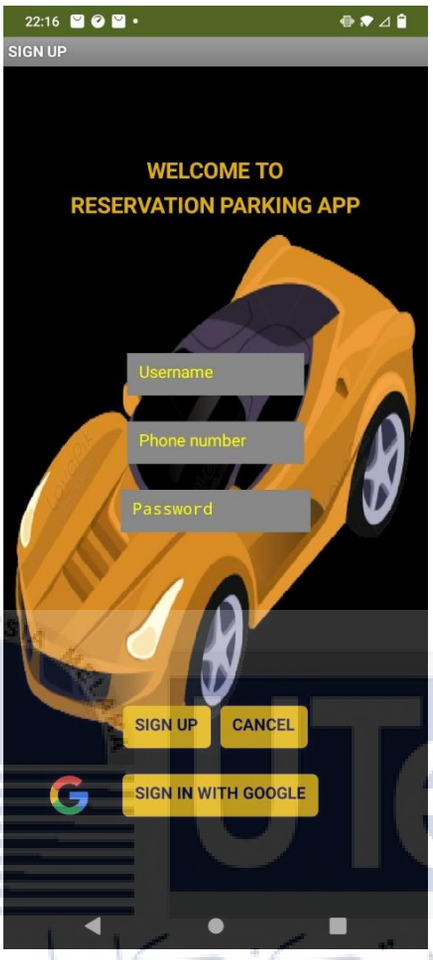


Figure 4-2 Registration Page

The app's registration screen is shown in Figure 4.2. Users can create an account on this website by entering their phone number, password, and username. For those who would rather, there is now the option to sign in using Google, which offers a quick and easy way to register.

\

Figure 4-3 Menu

In Figure 4.3, the application's menu is displayed, offering users four distinct options to choose from "Find Parking" where the feature allows users to search and locate available parking spaces within the app's network. Secondly, "Vehicle Registration": users can access and update their vehicle information such as plate number. Thirdly, "Vehicle Reservation" this option enables users to book a parking space for their registered vehicle once all necessary details have been provided. Lastly, "Reservations": Users can view and manage their existing parking reservations through this option, including cancellations or modifications if required.

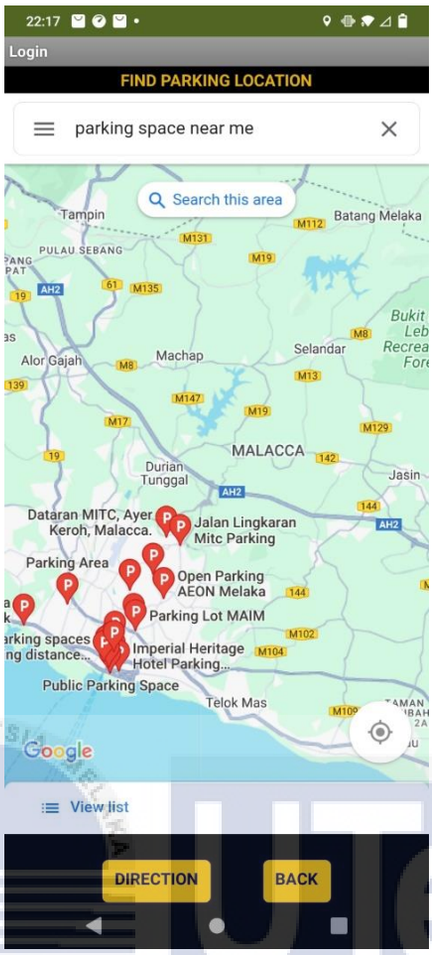


Figure 4-4 Find location

The application shows a map in Figure 4.4 when the user selects the "Find Parking" option. The locations of parking spots that are available and that the application can access are displayed on the map. Users are able to locate convenient and easily accessible parking options by looking at the map and identifying parking spots in the required area.

Figure 4-5 Vehicle Registration

The car registration page is shown in Figure 4.5. Users must enter their plate number and car model on this page. After completing the required fields, users can choose to either "Register" to finish the registration procedure or "Back" to return to the application's main menu without registering their car.

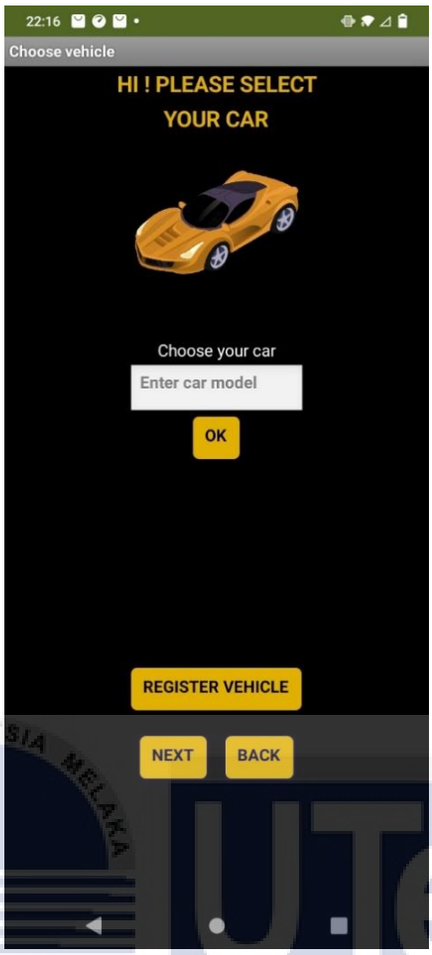


Figure 4-6 Vehicle Selection

Once registration is complete, the car selection screen appears, as shown in Figure 4.6. Users are invited to input the plate number of their car on this page. Users can move forward by using the "Next" button after entering the number plate. Users can designate the car they wish to reserve or manage within the application on this screen. “Back” button press will guide the user to the main menu interface of this application. “Register Vehicle” button press for the user that not yet registered the vehicle.

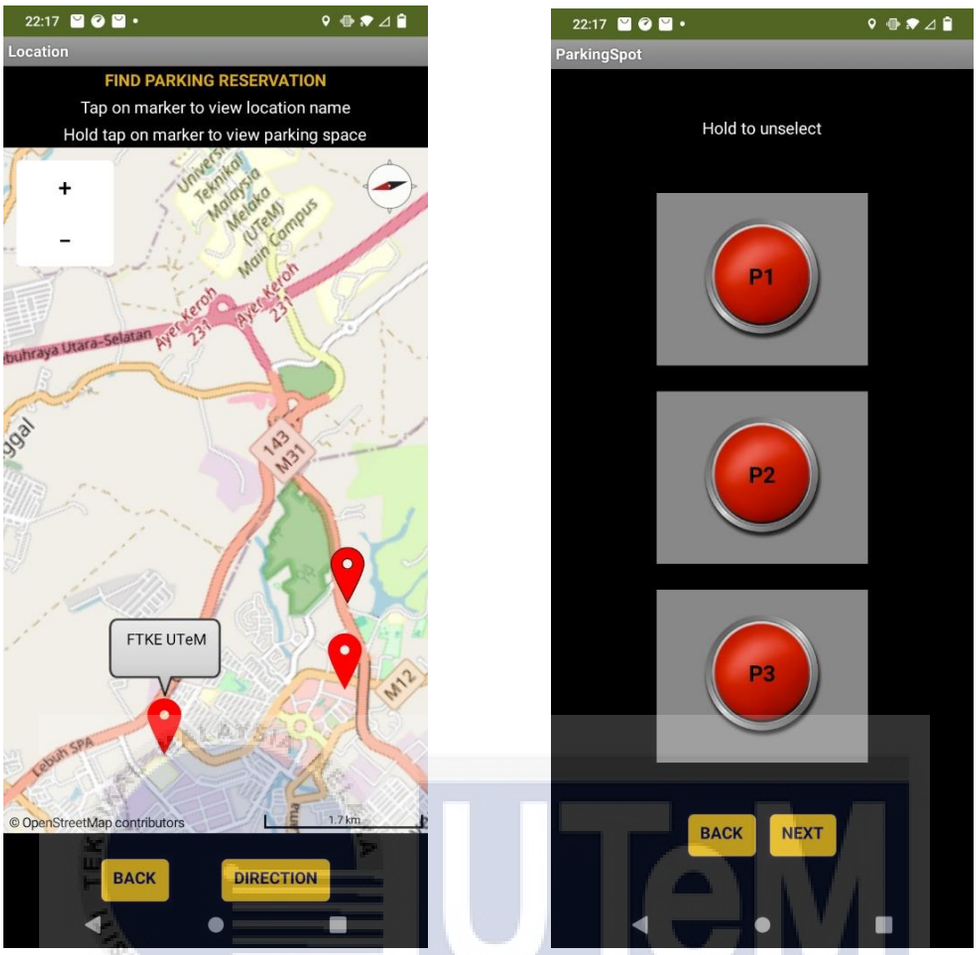


Figure 4-7 Reservation Parking

The page for parking reservations is shown in Figure 4.7. There are markers or pins on this page that indicate where parking is available. Tap and hold the marker, the page transitions to a new screen. On this screen, users are prompted to choose specific parking spot from options like P1, P2, and P3. After choosing a preferred parking space, users press “NEXT” button to next page for set date and time.

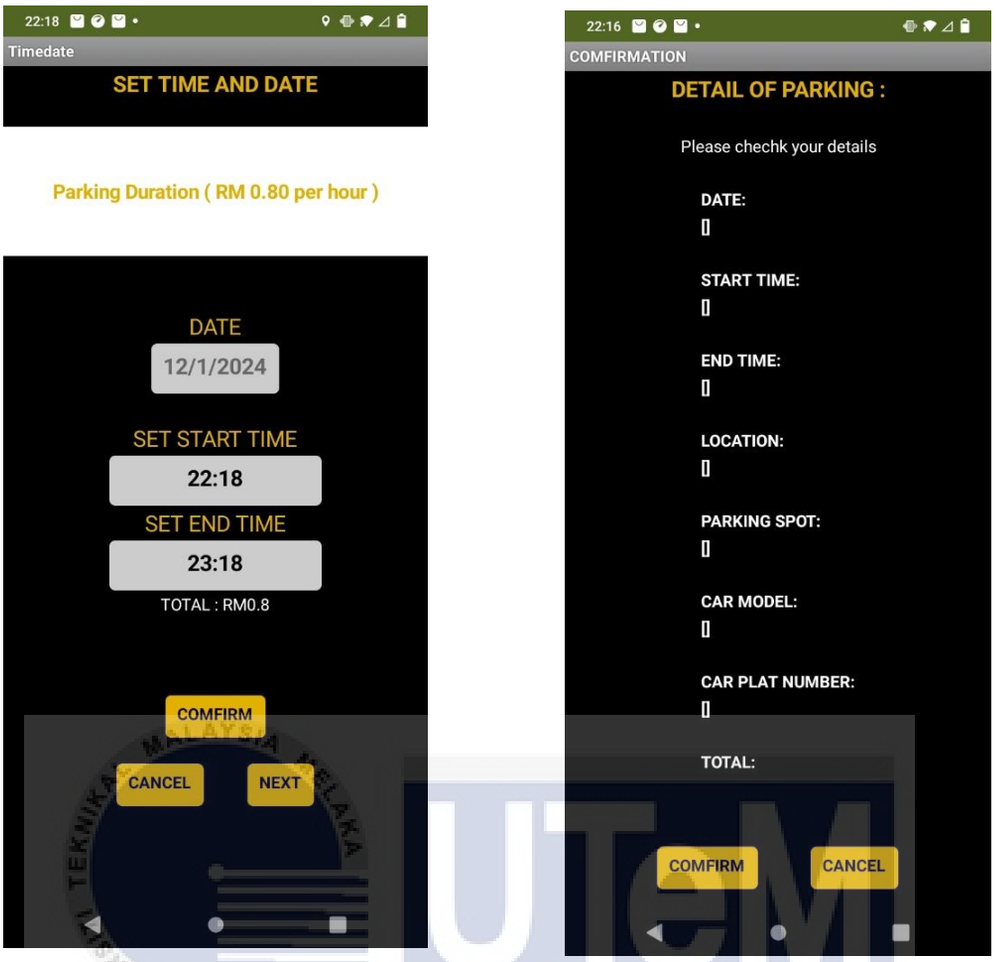


Figure 4-8 Date & Time

The "Set Date and Time" feature is displayed on the page in Figure 4.8. When making a parking reservation, users can enter a precise date and time. Furthermore, there is information regarding the parking cost, which is shown at RM 0.80 per hour, above the screen. This enables customers to enter the period of their reservation and get the estimated cost of their parking reservation. The parking rates for the selected date and time are made transparent and understandable by this function. Total of the parking fee will appear once press confirm button. When users press next button, confirmation page will appear and this is where all the information for the parking registration is showed. When button"Comfirm" click, it proceed to the payment method.

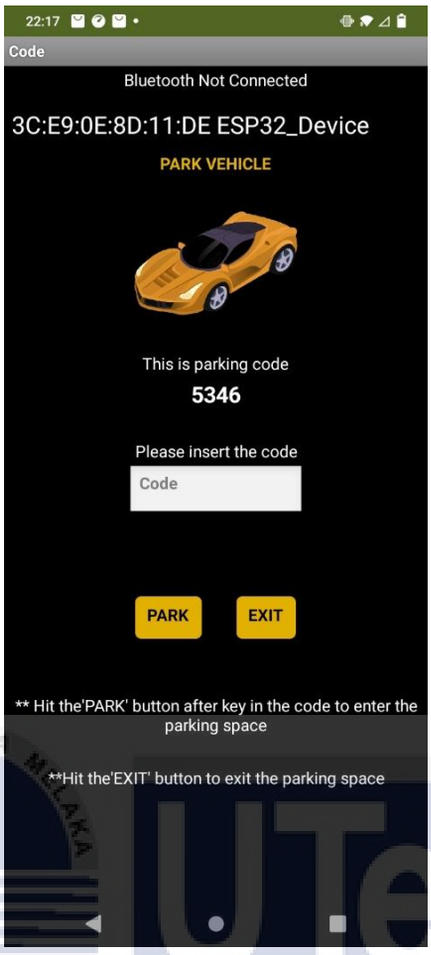


Figure 4-9 Code

Once payment is complete, the random code will be appeared, as shown in Figure 4.9. The code is needed to insert when the user arrived at the reservation parking space location. When user hit the “Park” button it will send the signal to the microcontroller which is ESP32 and move the servo motor as agate to open and the user can entry the parking space. But if the code is wrong the gate will not open. The “Exit” button function is when the user wants to exit from the parking space the user needs to hit that button.

4.2.2 Hardware Prototype View

This section demonstrates the real hardware prototype display view from the top, left, front, right and back angles as well. The major component required in Figure below is the ESP 32 microcontroller, which is connected to IR sensor, Led, Servo motor and LCD display.

Connecting the ESP32 microcontroller with the component that have been mentioned before is part of the wiring procedure for the hardware prototype. For precise control, GPIO pins on the microcontroller are used to connect the ESP32 with the component. Determine that the ESP32 and the application are connected via Bluetooth.

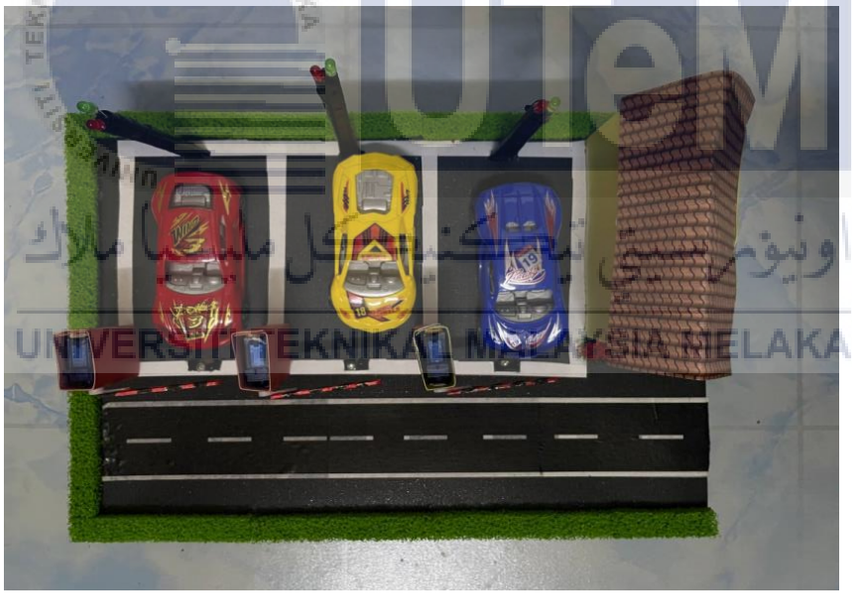


Figure 4-10 Top View

Figure 4-11 Left Side View

Figure 4-12 Right Side View

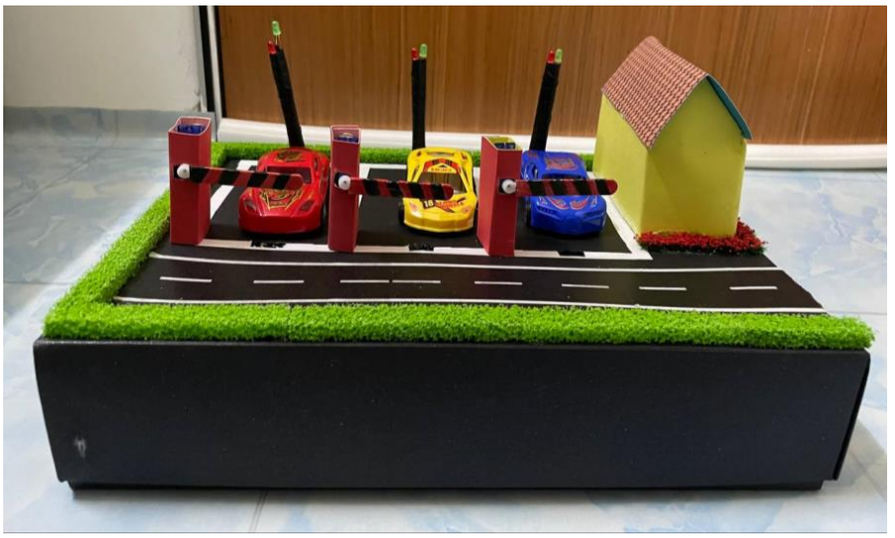


Figure 4-13 Front view

Figure 4-14 Back View

4.3 Trial Analysis

This trial analysis involves systematically assessing and evaluating outcomes during a preliminary test or experiment. For this project, cloud DB as database stored for the application. As for now the cloud DB is free cloud to use but have a limitation for a client number. Below show the table 4.1 the effectiveness of application.

Table 4-1 Parking application success rate assessment

	Successful	**Not successful**
People using the application	48	2

Figure 4.15 illustrates the usage of a parking application by individuals. Among 50 attempts made, 48 were successful, while 2 ended in failure. The primary reason for the unsuccessful attempts was the clients reaching their maximum capacity. The application proved effective for the majority, but steps may need to be taken to address the issue of client limitations to enhance overall user experience and satisfaction.

Figure 4-15 illustrates the usage of a parking application by individuals.

4.4 Data Analysis based on light rays

For this project, sometimes light rays also can affect the effectiveness of IR sensor to detect the object. It is because IR sensor works by detecting infrared radiation emitted or reflected by objects. There are two types of IR which are emitter and receiver. The emitter sends out infrared light while the receiver measures the reflected or emitted light.

Data analysis based on light rays involves interpreting information obtained from the behavior of light waves. Table 4.2 provide the results from 3 types of light rays which is yellowish, dark, and normal light rays that have been test by this project.

Table 4-2 types of light rays that have been tested by this project.

Type of light rays	Detect Object	Accuracy
Yellowish	Detect	High accuracy
Dark	Sometimes not	Low accuracy
Normal	Detect	High accuracy

4.5 Output Result

The Serial Monitor function at Arduino IDE as a real-time connection between an ESP32 board and a computer is made possible by the Arduino IDE's Serial Monitor feature. By showing information about variable values, sensor readings, and other outputs, it facilitates debugging by displaying data received by the ESP32 microcontroller. This capability is essential for improving and debugging this reservation parking project.

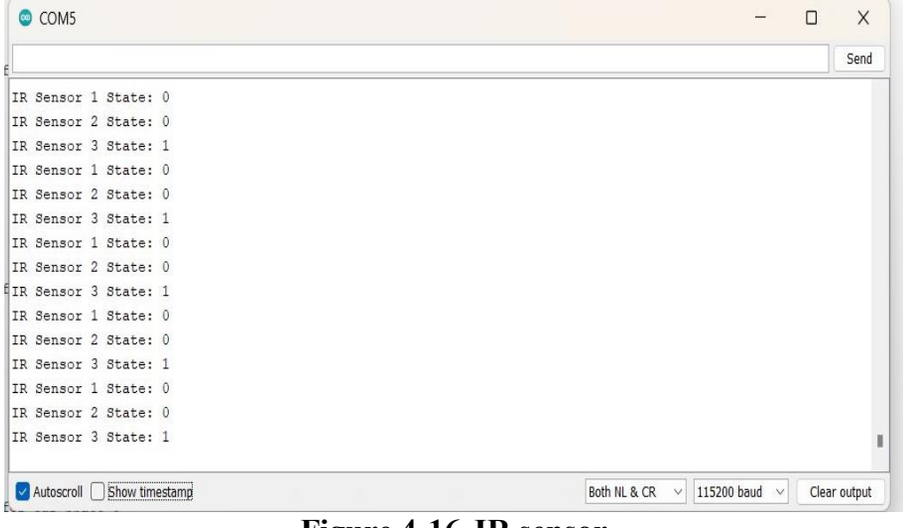


Figure 4-16 IR sensor

This is the output for IR sensor that shown in Figure 4.16. In general, IR sensor is to detect the object. This IR sensor will be receiving and analysing the infrared signal emitted or reflected by the object. As Figure above shown the IR sensor detection. IR sensor will ‘LOW’ condition if detect the object and it will be present ‘0’ in serial monitor. While, when IR sensor did not detect the object, it will be in ‘HIGH’ condition and will represent to’1’ at serial monitor.

Figure 4-17 IR sensor and Red Led

This Figure 4.17 shown IR sensor and red led. This figure shown that when red led turn on it will display ‘1’ and IR sensor is in ‘LOW’ condition and will represent as ‘0’ at the serial monitor. With this condition that is mean IR sensor detect the object.

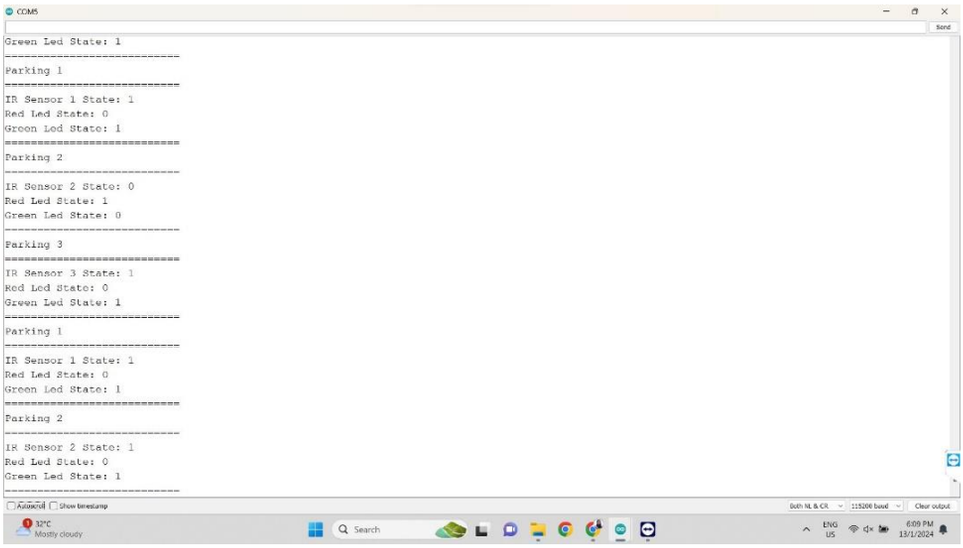


Figure 4-18 IR sensor and green Led and red Led

This Figure 4.18 shown IR sensor and green led. This figure shown that when green led turn on and it will display '1' that is mean IR sensor is in 'HIGH' condition and will represent as '1' at the serial monitor. With this condition that is mean IR sensor did not detect the object and a few conditions with red led turn on it will display '1' and IR sensor is in 'LOW' condition and will represent as '0' at the serial monitor. With this condition that is mean IR sensor detect the object.

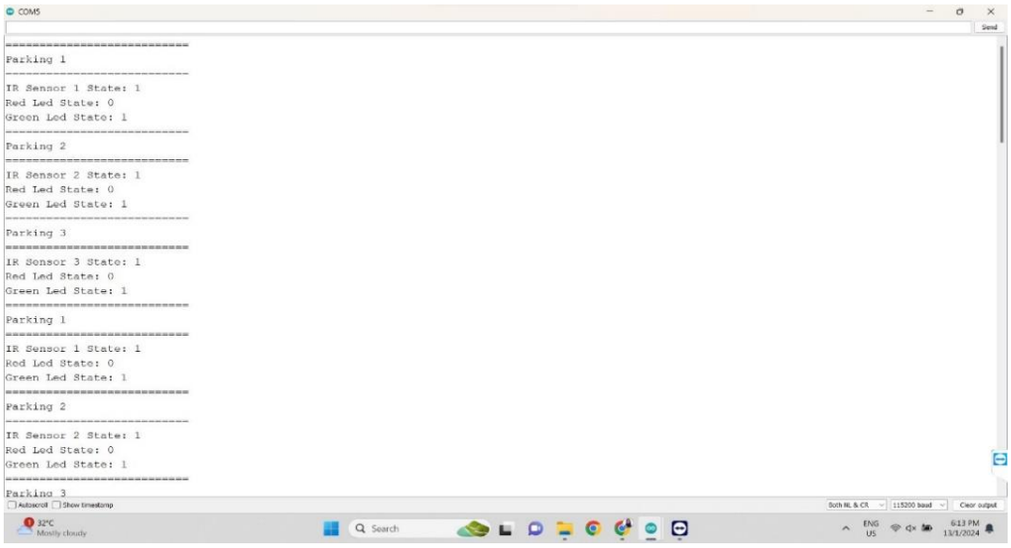


Figure 4-19 IR sensor and green Led

This Figure 4.19 shown IR sensor and green led. This figure shown that when green led turn on and it will display '1' that is mean IR sensor is in 'HIGH' condition and will represent as '1' at the serial monitor. With this condition that is mean IR sensor did not detect the object.

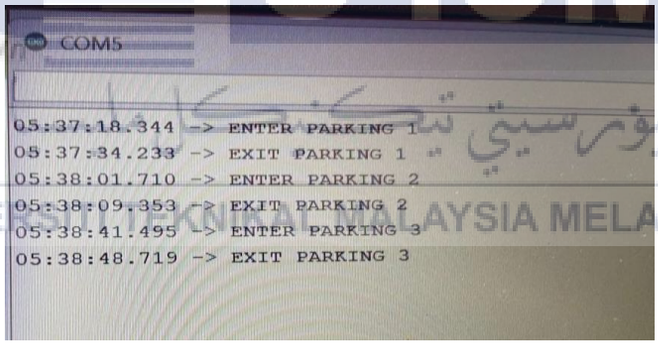


Figure 4-20 Test the button that have been created at MIT App Inventor

This Figure 4.20 shown the output for servo motor that move '0' to '90' degree. This output is communicated between ESP32 and application MIT App Inventor by using Bluetooth. As have been mentioned above our reservation parking space have 3 which is P1, P2 and P3. When user choose P1 for reservation parking space then the user hit the "PARK" button to

enter the parking space if the user want to exit just hit “EXIT” button to exit from the reservation parking space.

4.6 Price Range of Prototype

Parking Reservation that developed in this project is consider as a cheap but working well and good to use for nowadays. The component or software which include to develop this Reservation Parking project are as follow:

- ESP32 microcontroller
- MIT App Inventor
- IR sensor
- Jumper/Wire
- 12v DC Supply
- Led
- LCD display
- Servo motor SG90
- Regulator
- ESP32 Housing

Table 4-3 Below shows the price of each component/software which needed

	QTY	PRICE
IR SENSOR	6	RM30
ESP 32	1	RM25
SERVO MOTOR	3	RM15
GREEN LED	3	RM 0.30
RED LED	3	RM0.30
JUMPER WIRE		RM30
LCD	1	RM22
HOUSING	1	RM18
REGULATOR	1	RM8
12V DC BATTERY SUPPLY	1	RM62
TOTAL		**RM210.60**

From this price range above, we can conclude that the components and software which included in doing this project in below than RM 300. Although software is using MIT App Inventor which is no need to pay any fee, but it have a limited screen but it is not a big problem at all.

4.7 Summary

From the summary of this topic can conclude that the development of this project works no matter in hardware or software, and it gives benefits to the people. From this chapter, it proved that with the main component in this project, microcontroller ESP32, IR sensor, Led, LCD display, Servo

motor and MIT App Inventor as IoT - Based cloud platform, they work well, and the data of the monitoring object can be sent to smartphone without hesitation.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The reservation parking system using a mobile application has proven to be a valuable solution for optimizing parking management and enhancing user experience. This system offers numerous benefits such as reducing the time spent searching for parking spaces, increasing convenience for drivers, improving traffic flow, and maximizing parking space utilization. The application-based reservation system provides a seamless and efficient process for users to find and secure parking spots in advance.

The mobile application allows users to access real-time information about available parking spaces, including their location, availability status, and pricing. This empowers users to make informed decisions and plan their parking in advance, resulting in time and effort savings. The system also offers flexibility in choosing parking options based on proximity to destinations or specific amenities.

Furthermore, the reservation parking system enhances the overall user experience by providing a seamless and contactless parking process. Users can easily make reservations, pay for parking, and receive digital tickets or QR codes for accessing designated parking areas. This eliminates the need for physical tickets or cash transactions, promoting social distancing and reducing direct contact.

From a management perspective, the reservation parking system provides valuable insights and data on parking space utilization, peak hours, and user preferences. This information can be leveraged to optimize parking operations, allocate resources effectively, and implement dynamic pricing strategies for maximizing revenue. It is essential to acknowledge the challenges and limitations associated with the reservation parking system, such as the initial investment required, ensuring a reliable network infrastructure, and addressing privacy and security concerns related to user data.

In conclusion, the reservation parking system utilizing a mobile application offers significant advantages for both users and parking management. It simplifies the parking process, reduces congestion, improves user convenience, and optimizes parking space utilization. With continued technological advancements and system enhancements, the reservation parking system has the potential to revolutionize the parking experience and contribute to smarter and more sustainable cities.

5.2 Contribution Research

The "Reservation Parking System Using Apps" represents a significant contribution to the field of smart urban mobility and parking management. This innovative project integrates a range of hardware components, including the ESP32 microcontroller, IR sensors, LEDs, servo motors, and an LCD display, with a user-friendly mobile app. The real-time communication and

synchronization between these elements create a seamless parking experience for users.

One notable contribution is the enhancement of user convenience and efficiency in parking space reservation. The integration of the ESP32 microcontroller with the app streamlines the entire process, from user authentication to selecting parking spaces, setting reservation parameters, and making secure payments. This not only optimizes the utilization of parking spaces but also minimizes the time and effort required for users.

Furthermore, the system's incorporation of IR sensors, LEDs, and servo motors contributes to real-time status indicators for parking availability and secure gate control. The deployment of six LEDs per parking space which is three red indicating occupancy and three green indicating availability to provide users with a visual the purpose that provides relevance. The servo motors actuate the gate mechanism, enhancing security and automating the entry and exit processes.

Additionally, the LCD display further augments user experience by presenting vital information such as available parking spaces. The project's user authentication system, reservation workflow, and payment confirmation significantly contribute to the realm of secure, efficient, and user-friendly parking solutions.

In conclusion, the "Reservation Parking System Using Apps" project makes noteworthy contributions to the optimization of urban parking management, combining hardware innovation with an intuitive app interface to redefine user experience, enhance security, and streamline parking processes in contemporary urban environments.

5.3 Future Work

The "Reservation Parking System Using Apps" involves exploring advanced features to further elevate the system's functionality. Firstly, by offering more precise real-time data, machine learning techniques for parking space availability prediction could improve user experience. Additionally, integrating a camera-based license plate recognition system would streamline entry and exit processes, reducing reliance on manual code input and the fee penalty for the user that exceed from the tome that have been set.

Further developments may include a comprehensive user management system within the app, allowing users to track their parking history, preferences, and receive personalized notifications. Exploring the integration of renewable energy sources, such as solar power, for the system's components could contribute to sustainability goals.

To enhance security measures, implementing advanced encryption protocols for data transmission between the ESP32 microcontroller and the app could strengthen the system's resilience against potential threats. Integration with smart city initiatives could foster a more interconnected urban infrastructure, leading to better traffic management and urban planning.

Moreover, future iterations might involve scalability considerations to accommodate a larger number of users and parking spaces, making the

system adaptable to varying urban landscapes. Finally, continuous updates and user feedback analysis would be crucial for refining the app interface, ensuring it remains intuitive and user-friendly. These proposed advancements collectively aim to position the Reservation Parking System as a cutting-edge, adaptable, and sustainable solution in the evolving landscape of urban mobility and smart city development.

REFERENCES

[1] Dr Y Raghavender Rao," Automatic Smart Parking System using Internet of Things
(IOT)" International Journal of Engineering Technology Science and Research, Vol.4, No.5, pp.225-258,May 2017

[2] Abdul Ahad, Zishan Raza Khan, Syed Aqeel Ahmad,"Intelligent Parking System"
Scientific Research Publishing, Vol.4, No.2, pp. 160-167, May 2016

[3] A. Rajalakshmi and H. Shahnasser, "Internet of Things using Node-Red and alexa," in *2017 17th International Symposium on Communications and Information Technologies (ISCIT)*, 2017, pp. 1–4. doi: 10.1109/ISCIT.2017.8261194.

[4] T. Raj, Y. Cheema, V. K. Singh, A. Kumar, and S. N. Gupta, "Automatic shopping trolley using IOT," *Mater Today Proc*, vol. 79, pp. 362–366, 2023, doi: https://doi.org/10.1016/j.matpr.2022.12.152.

[5] A. V Zinkevich, "ESP8266 Microcontroller Application in Wireless Synchronization Tasks," in *2021 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*, 2021, pp. 670–674. doi: 10.1109/ICIEAM51226.2021.9446411.

[6] A. Vishwakarma, A. Sahu, N. Sheikh, P. Payasi, S. K. Rajput, and L. Srivastava, "IOT Based Greenhouse Monitoring And Controlling System," in *2020 IEEE Students Conference on Engineering & Systems (SCES)*, 2020, pp. 1–6. doi: 10.1109/SCES50439.2020.9236693.

[7] A Complete Guide On Car Parking App Development in August 29, 2021 https://appdukaan.com/car-parking-app-development/

[8] IoT-based Smart Parking System – A Step Towards Building Smart City in May 28, 2022 https://www.airtel.in/blog/business/iot-based-smart-parking-system-a-step-towards-building-smart-city/

[9] H Wang and W Hei (n.d), A Reservation- Based Smart Parking System, The 1st International Workshop on Cyber- Physical Networking System

[10] Z PALA and N INAN (n.d), *Smart Parking by Using RFID Technology, Department of Electrical and Electronics Engineering Yuzuncu Yil University, Turkey.*

[11] Roojai.com. (2019, April 6). *5 Must-have Apps for Car Lovers! Definitely good*. [On- line]. Available: http://www.roojai.com/article/car-parts-and-car-accessories/5-app-for-car/

[12] Bureau of Road Maintenance Administration. (2022, March 4). Department of Highways. *Thailand Highway Traffic* [On-line]. Available: http://bmm.doh.go.th/website/index.php/2-uncategorised/190- thailand-highway-traffic#.

[13] Thai Car License Plates good. (2022, March 4). [On-line]. Available: https://driving.information.in.th/license-plates.html.

[14] Vehicle registration plates of Thailand. (2022, March 4). [On-line]. Available: https://www.wikiwand.com/en/Vehicle_registration_plates_of_Thailand

[15] Including License Plates of Each Type. (2022, March 4). [On-line]. Available: https://www.asiadirect.co.th/adbnewsdetail.php?newId=62.

[16] MIT App Inventor. (2021, January 15). [On-line]. Available: https://appinventor.mit.edu/.

APPENDICES

APPENDIX 1 Gantt Chart

PSM 1 Gantt chart

No.	Task/ Activity	Week												
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	Project briefing														
2	Submission of project title														
3	Hardware and sensor finalization														
4	Literature Review														
5	Components Ordering														
6	EMG signal validation														
7	Report writing														
8	Submission of the final report														
9	Preparation and Presentation														

Pla

Actua

PSM 2 Gantt chart

WEEK/ ACTIVITY	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Details calculation and theory concept														
Project design finalization														
Hardware and sensor finalzation														
Assemmble hardware part														
Collecting the data														
Final documentation and report writing														
PSM 2 drraft submission														
Preparation and Presentation														
Submission of final report														

APPENDIX 2 Coding

Coding for testing Ultrasonic Sensor

```
// Define the trigger and echo pins
const int triggerPin = 2;
const int echoPin = 3;
long duration;
int distance;
void setup()
{
// Initialize serial communication
Serial.begin(9600);
// Set trigger pin as output and echo pin as input
pinMode(triggerPin, OUTPUT);
pinMode(echoPin, INPUT);
}
void loop()
{
// Clear the trigger pin
digitalWrite(triggerPin, LOW);
delayMicroseconds(2);
// Send a 10 microsecond pulse to the trigger pin
```



```
digitalWrite(triggerPin, HIGH);
delayMicroseconds(10);
```

```
digitalWrite(triggerPin, LOW);

// Read the duration of the echo pulse

duration = pulseIn(echoPin, HIGH);

// Wait for a short delay before the next measurement

delay(500);

}
```

Coding for overall project

```
#include <ESP32Servo.h>

#include <ESP32PWM.h>

#include "BluetoothSerial.h"

#include <LiquidCrystal_I2C.h>

#include <Wire.h>


LiquidCrystal_I2C lcd(0x27,16,2);

Servo myservo1;

Servo myservo2;

Servo myservo3;

BluetoothSerial SerialBT;

int servo1Pin = 23;

int servo2Pin = 26;

int servo3Pin = 27;


const int irSensorPin1 = 5;

const int irSensorPin11 = 17;

const int redLedPin1 = 4;

const int greenLedPin1 = 13;


const int irSensorPin2 = 18;

const int irSensorPin22 = 16;

const int redLedPin2 = 25;

const int greenLedPin2 = 14;
```

```
const int irSensorPin3 = 19;

const int irSensorPin33 = 2;

const int redLedPin3 = 15;

const int greenLedPin3 = 12;


int prevIrState1 = HIGH;

int prevIrState11 = HIGH;

int prevIrState2 = HIGH;

int prevIrState22 = HIGH;

int prevIrState3 = HIGH;

int prevIrState33 = HIGH;



int availableSlots = 3;  // Initially, all slots are available

char displayText[20];     // Adjust the array size based on your LCD size

unsigned long lastCarDetectedTime = 0;  // Variable to store the time when a car is

detected by IR Sensor 1




void setup() {

 Serial.begin(115200);

 lcd.init();       // Initialize LCD

 lcd.backlight();
```

```
SerialBT.begin("ESP32_Device"); //Bluetooth device name
ESP32PWM::allocateTimer(0);
ESP32PWM::allocateTimer(1);
ESP32PWM::allocateTimer(2);
ESP32PWM::allocateTimer(3);
myservo1.setPeriodHertz(50);    // standard 50 hz servo
myservo2.setPeriodHertz(50);    // standard 50 hz servo
myservo3.setPeriodHertz(50);    // standard 50 hz servo

myservo1.attach(servo1Pin, 500, 2400);
myservo2.attach(servo2Pin, 500, 2400);
myservo3.attach(servo3Pin, 500, 2400);

pinMode(irSensorPin1, INPUT);
 pinMode(irSensorPin11, INPUT);
pinMode(redLedPin1, OUTPUT);
pinMode(greenLedPin1, OUTPUT);

pinMode(irSensorPin2, INPUT);
pinMode(irSensorPin22, INPUT);
pinMode(redLedPin2, OUTPUT);
pinMode(greenLedPin2, OUTPUT);

pinMode(irSensorPin3, INPUT);
pinMode(irSensorPin33, INPUT);
```

```
pinMode(redLedPin3, OUTPUT);
pinMode(greenLedPin3, OUTPUT);

digitalWrite(redLedPin1, LOW);
digitalWrite(greenLedPin1, HIGH);

digitalWrite(redLedPin2, LOW);
digitalWrite(greenLedPin2, HIGH);

digitalWrite(redLedPin3, LOW);
digitalWrite(greenLedPin3, HIGH);

lcd.home();
lcd.print("AVAILABLE");
lcd.setCursor(0, 1);
lcd.print("SPACE: ");
lcd.setCursor(7, 1);  // Adjust the cursor position based on your LCD size
lcd.print("      ");  // Clear the existing value
lcd.setCursor(7, 1);
sprintf(displayText, "%d", availableSlots);
lcd.print(displayText);



}
```

```
void updateLCD() {
  lcd.setCursor(7, 1);  // Adjust the cursor position based on your LCD size
  lcd.print("      ");  // Clear the existing value
  lcd.setCursor(7, 1);
  sprintf(displayText, "%d", availableSlots);
  lcd.print(displayText);
}

void loop() {

// Read the state of IR sensor for car space 1
int irState1 = digitalRead(irSensorPin1);
int irState11 = digitalRead(irSensorPin11);
// Read the state of IR sensor for car space 2
int irState2 = digitalRead(irSensorPin2);
int irState22 = digitalRead(irSensorPin22);
// Read the state of IR sensor for car space 3
int irState3 = digitalRead(irSensorPin3);
int irState33 = digitalRead(irSensorPin33);

if (Serial.available()) {
   SerialBT.write(Serial.read());
}
if (SerialBT.available()) {
```

```
  Serial.write(SerialBT.read());
}
delay(5);

if (SerialBT.available()) {
  char command = SerialBT.read();
  switch (command) {

    case 'd':
      myservo1.write(180);
      Serial.println("ENTER PARKING 1");
      delay(10000);
      while (irState1 == HIGH && irState11 == LOW ){
          delay(50);
          }
      lastCarDetectedTime = millis();
      while (irState11 == LOW && (millis() - lastCarDetectedTime) <= 5000) {
          delay(50);
          }
      lcd.setCursor(13, 1);
      lcd.print("P1");
      myservo1.write(90);
      digitalWrite(redLedPin1, HIGH);
      digitalWrite(greenLedPin1, LOW);
      availableSlots = max(0, availableSlots - 1);
```

```
    break;


case 'a':
    myservo1.write(180);
    Serial.println("EXIT PARKING 1");
    delay(10000);
    while (irState1 == LOW && irState11 == HIGH ){
          delay(50);
          }
    lastCarDetectedTime = millis();
    while (irState11 == LOW && (millis() - lastCarDetectedTime) <= 5000) {
          delay(50);
          }
    lcd.setCursor(13, 1);
    lcd.print("P1");
    myservo1.write(90);
    digitalWrite(redLedPin1, LOW);
    digitalWrite(greenLedPin1, HIGH);
    availableSlots = min(3, availableSlots + 1);
    lcd.setCursor(13, 1);
    lcd.print("    ");
    break;


case 'e':
    myservo2.write(180);
```

```
  Serial.println("ENTER PARKING 2");
  delay(10000);
  while (irState2 == HIGH && irState22 == LOW ){
      delay(50);
      }
  lastCarDetectedTime = millis();
  while (irState22 == LOW && (millis() - lastCarDetectedTime) <= 5000) {
      delay(50);
      }
  lcd.setCursor(13, 1);
  lcd.print("P2");
  myservo2.write(90);
  digitalWrite(redLedPin2, HIGH);
  digitalWrite(greenLedPin2, LOW);
  availableSlots = max(0, availableSlots - 1);
  break;

case 'b':
  myservo2.write(180);
  Serial.println("EXIT PARKING 2");
  delay(10000);
  while (irState2 == LOW && irState22 == HIGH ){
      delay(50);
      }
  lastCarDetectedTime = millis();
```

```
  while (irState22 == LOW && (millis() - lastCarDetectedTime) <= 5000) {
        delay(50);
        }
  lcd.setCursor(13, 1);
  lcd.print("P2");
  myservo2.write(90);
  digitalWrite(redLedPin2, LOW);
  digitalWrite(greenLedPin2, HIGH);
  availableSlots = min(3, availableSlots + 1);
  lcd.setCursor(13, 1);
  lcd.print("  ");
  break;

case 'f':
  myservo3.write(180);
  Serial.println("ENTER PARKING 3");
  delay(10000);
  while (irState3 == HIGH && irState33 == LOW ){
        delay(50);
        }
  lastCarDetectedTime = millis();
  while (irState33 == LOW && (millis() - lastCarDetectedTime) <= 5000) {
        delay(50);
        }
  lcd.setCursor(13, 1);
```

```
    lcd.print("P3");
    myservo3.write(90);
    digitalWrite(redLedPin3, HIGH);
    digitalWrite(greenLedPin3, LOW);
    availableSlots = max(0, availableSlots - 1);
    break;
  case 'c':
    myservo3.write(180);
    Serial.println("EXIT PARKING 3");
    delay(10000);
    while (irState3 == LOW && irState33 == HIGH ){
        delay(50);
        }
    lastCarDetectedTime = millis();
    while (irState33 == LOW && (millis() - lastCarDetectedTime) <= 5000) {
        delay(50);
        }
    lcd.setCursor(13, 1);
    lcd.print("P3");
    myservo3.write(90);
    digitalWrite(redLedPin3, LOW);
    digitalWrite(greenLedPin3, HIGH);
    availableSlots = min(3, availableSlots + 1);
    lcd.setCursor(13, 1);
    lcd.print("    ");
```

```
        break;
      default:
        // Handle unexpected command if needed
        break;




}
// Update previous states
prevIrState1 = irState1;
prevIrState11 = irState11;
prevIrState2 = irState2;
prevIrState22 = irState22;
prevIrState3 = irState3;
prevIrState33 = irState33;

updateLCD();  // Update LCD display with the current available slots

  // Add a delay to control the update rate
delay(500);
}
}
```