Palestine Polytechnic University

College of Information Technology and Computer Engineering

Department of Computer Engineering

Smart Car Wash System

Project Team

Husam Salameh

Tariq Qawasmeh

Supervisor

Dr. Elayan Abu Gharbyeh

June 2025

Acknowledgment

In the name of Allah, the Most Gracious, the Most Merciful, who granted us strength, perseverance, and wisdom to complete this project. We extend our heartfelt gratitude to Him for His blessings and guidance throughout this journey. We wish to express our deepest appreciation to our graduation project supervisor, Dr. Elayan Abu Gharbyeh, whose invaluable guidance, encouragement, and expertise have been instrumental in shaping our work. His mentorship has inspired us to strive for excellence and overcome challenges with determination. We are profoundly grateful to our families for their unwavering support, love, and patience throughout this project. Their belief in our capabilities and sacrifices have been the foundation of our success. Mom, Dad, and all our loved ones, this achievement is as much yours as it is ours.

We also thank our friends, colleagues, and everyone who contributed to our journey, whether through words of encouragement, technical assistance, or moral support. Your contributions have enriched our experience and made this milestone possible.

Finally, we acknowledge every individual and organization that played a role in making this project a reality. From those who provided resources to those who inspired us with their work, your impact is deeply appreciated. As we celebrate this accomplishment, we carry with us gratitude for every step taken and every lesson learned along the way.

Abstract

The Smart Car Wash System is an integrated system developed to improve the efficiency of car wash services by relying on modern technologies and automation. This system aims to provide a smooth and fast customer experience and reduce reliance on human resources, leading to reduced operational costs and improved service quality.

The system provides users with an easy-to-use online platform through which they can book a car wash appointment at a time convenient for them. After selecting an appointment and paying electronically using a bank card, the user receives a unique QR code for the reservation, which they can then use upon arrival at the car wash station. Upon arrival at the station, the QR code is scanned by an installed camera to verify the validity of the reservation. The car is then automatically entered into the wash process. An ultrasonic sensor detects the presence of the car, activating the soap pump, followed by the water pump. The system automatically shuts down after the wash is complete, ensuring safe and organized operation.

The system includes an administrative control panel that enables administrators to monitor all reservations, control user data, manage supervisor accounts, and cancel or modify any reservation. All reservations are displayed in an organized manner for easy follow-up and quick decision-making.

The system has been developed to include a feature to prevent overlapping appointments, as new reservations are not allowed until 15 minutes after the previous reservation has expired. This mechanism helps reduce waiting times and ensures efficient operations without delays or congestion.

الملخص

نظام غسيل السيارات الذكي هو نظام متكامل تم تطويره لتحسين كفاءة خدمات غسيل السيارات من خلال الاعتماد على التقنيات الحديثة والأتمتة. يهدف هذا النظام إلى توفير تجربة سلسة وسريعة للعملاء، وتقليل الاعتماد على العامل البشري، مما يؤدي إلى تقليل التكاليف التشغيلية وتحسين جودة الخدمة.

يوفر النظام للمستخدمين منصة إلكترونية سهلة الاستخدام يمكن من خلالها حجز موعد لغسيل السيارة حسب الوقت المناسب لهم. بعد اختيار الموعد والدفع الإلكتروني باستخدام البطاقة البنكية، يحصل المستخدم على رمز QR خاص بالحجز، يُستخدم لاحقًا عند الوصول إلى محطة الغسيل. عند وصول السيارة إلى المحطة، يتم مسح رمز QR عبر كاميرا مثبتة للتأكد من صحة الحجز. بعدها، تدخل السيارة بشكل تلقائي إلى سير الغسيل. يقوم حساس الموجات فوق الصوتية باكتشاف وجود السيارة، ما يؤدي إلى تفعيل مضخة الصابون، تليها مضخة المياه، ثم يتوقف النظام تلقائيًا بعد انتهاء عملية الغسيل، مما يضمن تشغيلًا آمنًا ومنظمًا. ويشتمل النظام على لوحة تحكم إدارية تُمكّن المسؤول من مراقبة جميع الحجوزات، والتحكم في بيانات المستخدمين، وإدارة حسابات المشرفين، بالإضافة إلى القدرة على إلغاء أو تعديل أي حجز. وتُعرض جميع الحجوزات بشكل منظم لتسهيل المتابعة واتخاذ القرارات السريعة.

تم تطوير النظام ليشمل خاصية منع التداخل بين المواعيد، حيث لا يُسمح بحجز جديد إلا بعد مرور 15 دقيقة على انتهاء الحجز السابق. تسهم هذه الآلية في تقليل الانتظار وضمان سير العمل بكفاءة دون تأخير أو ازدحام.

Contents

Acknowledgment 2
Abstract 3
الملخص 4
CHAPTER 1: INTRODUCTION 11
1.1 Preface 11
1.2 Problem Statement 11
1.3 Project Aims and Objectives 12
1.4 Project Requirements 13
1.4.1 Functional Requirements 13
1.4.2 Non-Functional Requirements: 13
1.5 System Description 14
1.6 Project Limitations/Constraints: 15
1.7 Project Schedule 16
1.8 Report Outline 16
CHAPTER TWO: THEORETICAL BACKGROUND: 17
2.1 Preface: 17
2.2 Theoretical Background Concepts: 17
2.2.1 Embedded Systems : 17
2.2.2 QR Code Recognition: 17
2.2.3 Automation and Control Logic: 18
2.2.4 Conveyor Belt Mechanism: 18
2.3 Literature Review: 18
2.4 Summary 19
CHAPTER THREE: SYSTEM DESIGN 20
3.1 Preface 20
3.2 System Components and Design Alternatives 20
3.2.1 Hardware Components 20
3.2.2 Software Components: 26
3.3 Conceptual system description 28
3.4 Algorithms and methodologies 30

3.4.1 Flow chart 30
3.4.2 Use Case Diagram 31
3.4.3 Sequence Diagram 32
3.5 Schematic diagram 33
3.6 Summary 34
CHAPTER FOUR: IMPLEMENTATION AND TESTING 35
4.1 Preface 35
4.2 Implementation issues 35
4.3 Implementation challenges 37
4.4 Software Implementation 38
4.4.1 Database 38
4.4.2 Login page 39
4.4.3 Main page 39
4.4.4 Create an account 40
4.4.5 The main interface for the customer account. 40
4.4.6 New reservation 41
4.4.7 Reservation request history. 41
4.4.8 Customer Information. 42
4.5 Hardware Implementation 43
4.6 Validation and testing 43
4.6.1 Unit Testing: 44
4.6.2 Software Testing: 45
4.6.3 Hardware Testing: 48
4.7 Summary 51
CHAPTER FIFTH : RESULTS AND DISCUSSION 52
5.1 Detailed analysis of the results/experiments 52
5.2 Justifications of the obtained results 54
5.3 Summary 55
CHAPTER SIX: CONCLUSION AND FUTURE WORK 56
6.1 Conclusion 56
6.2 Future work 57

References 58
Appendix A 60
Appendix B 64
Appendix C 65
Appendix D 67

List of Figures

Figure1 .1 System Description.....15
Figure3. 1 Raspberry Pi 4B[3].....22
Figure3. 2 Ultrasonic Sensor[4].....22
Figure3. 3 Relay 4 channel[7].....23
Figure3. 4 Water Pump[8] 24
Figure3. 5 Conveyor belt & DC Motor[9].....25
Figure3. 6 Gate Actuator[10].....26
Figure3. 7 System Block Diagram.....29
Figure3. 8 System conceptual diagram.....29
Figure3. 9 Flowchart.....30
Figure3. 10 Use Case Diagram.....31
Figure3. 11 Sequence Diagram.....32
Figure3. 12 Schematic Diagram 33
Figure4. 1 Types Of Pumps 35
Figure4. 2 Power Converters 36
Figure4. 3 Get QR Code 36
Figure4. 4 Generating A QR Code 37
Figure4. 5 Database Tables.....38
Figure4. 6 Login Page.....39
Figure4. 7 Main Page.....39
Figure4. 8 Create an account 40
Figure4. 9 The main interface for the customer account. 40
Figure4. 10 New reservation.....41
Figure4. 11 Reservation Request History.....41
Figure4. 12 Customer Information. 42
Figure4. 13Connect The Electronic Components.....43
Figure4. 14 Hardware Testing 44

Figure4. 15 QR Testing.....45
Figure4. 16 Adding A Customer.....45
Figure4. 17 Verifying the Addition Customer.....46
Figure4. 18 Adding A New Administrator.....46
Figure4. 19 Customer Search.....47
Figure4. 20 Displaying The QR Code.....47
Figure4. 21 Ultrasonic sensor testing.....48
Figure4. 22 Camera testing.....49
Figure4. 23 Test the soap pump.....49
Figure4. 24 Test the start of the washing.....50
Figure4. 25 Hardware Testing.....50
Figure5. 1 Conveyor Belt.....52
Figure5. 2 Pumps in Belt.....52
Figure5. 3 Customer Account.....53
Figure5. 4 Sending A Reservation.....53
Figure5. 5 smart car wash system.....54

List of Table

Table1. 1 Project Schedule 16
Table2. 1 Comparison with other projects..... 19
Table3. 1 Differences between microcontrollers.....21
Table3. 2 Differences between cameras 23
Table3. 3 Comparison Between Pumps.....24

CHAPTER 1: INTRODUCTION

1.1 Preface

In this project, we are implementing a smart car wash system that integrates automation and smart technologies, which are essential components of the service sector. The car wash sector, in particular, is witnessing significant developments aimed at enhancing customer convenience, reducing manual labor, and improving operational efficiency. This project introduces a smart car wash system that integrates online booking, QR code verification, and automated washing mechanisms based on real-time sensor inputs.

1.2 Problem Statement

Ideally, car wash services should provide customers with a fast, reliable, and automated solution. Customers should be able to book and pay online. A car wash should be completed with minimal waiting time and high service quality, all while ensuring resource efficiency (water, time, and labor).

Currently, many car washes rely heavily on manual labor, leading to inefficiencies, longer waiting times, and higher operational costs. Customers often need to visit the car wash in person, wait in line, or deal with outdated systems that don’t offer online booking or payment options. This creates customer dissatisfaction and operational bottlenecks.

This project addresses these issues by designing and implementing a smart car wash system that integrates online booking, automated verification, and real-time operation based on sensor inputs. The goal is to improve service efficiency, enhance customer experience, and reduce reliance on manual labor.

1.3 Project Aims and Objectives

In this project, we propose a system that aims to provide the following features:

1. The system aims to develop a smart car wash system capable of automating the car wash process and enhancing customer convenience. To achieve this aim, the following objectives should be accomplished:

- (a) Providing the system with the ability to identify cars by recognizing each customer's QR code via a camera upon arrival at the car wash gate.
- (b) Automate the gate's operation, ensuring it opens only for authorized cars.
- (c) Activate water and washing mechanisms automatically when the car approaches the wash area.

2. We aim to design a user-friendly platform for booking and managing car wash appointments. To achieve this aim, the following objectives should be accomplished:

- (a) Develop a website that allows users to schedule appointments and track the status of their bookings.
- (b) Integrate a notification system to inform users about the progress of their car wash.

1.4 Project Requirements

1.4.1 Functional Requirements

The functional requirements define the core operations and behaviors that the Smart Car Wash System must perform to achieve its objectives. These requirements include:

1. The system allows customers to book online by entering their bank card details.
2. Upon successful booking, the system generates a unique QR code for the customer to use when visiting the car wash.
3. The camera is used to scan and validate the QR code against the recorded bookings.
4. The system detects the presence of a vehicle at pre-defined points using ultrasonic sensors.
5. Automatic pump control based on vehicle detection.
6. Restrictions on bookings on the website prevent any conflict between customer bookings.
7. Electronic payment for reservations using Visa card, but we do not have API from the bank.

1.4.2 Non-Functional Requirements:

1. High performance efficiency in terms of vehicle detection, pump operation and conveyor belt movement.
2. The system must be able to handle a large number of reservations and automatically organize them.
3. The system must support expansion to include multiple car wash stations with minimal modifications.
4. User data must be securely stored using encryption.

5. The user interface must be user-friendly and accessible to users without technical expertise.
6. The web interface must be compatible with modern browsers and mobile devices.

1.5 System Description

The smart car wash system is an integrated solution utilizing the latest car wash technologies. Appointments are booked online and electronic payment is made via bank or Visa cards to enhance customer convenience, improve service efficiency, and ensure consistent quality. This is achieved through a combination of web technologies, QR code recognition, real-time sensors, and automated mechanical components.

Once the vehicle enters the car wash, it is placed on a conveyor belt that moves it through the wash stages without manual intervention. Sensors detect the vehicle's location to activate the water pumps, soap dispensers, and dryers at the appropriate time.

All components are connected and communicate wirelessly to ensure a seamless process from booking to completion. The goal is to reduce manual labor, improve customer convenience, and enhance operational efficiency.

Reservation restrictions are also imposed on the website to prevent conflicts between customer reservations. This means that if a previous reservation has been made, it can only be made 15 minutes after the reservation has ended, at which point a new reservation is made to ensure the system's efficiency and effectiveness.

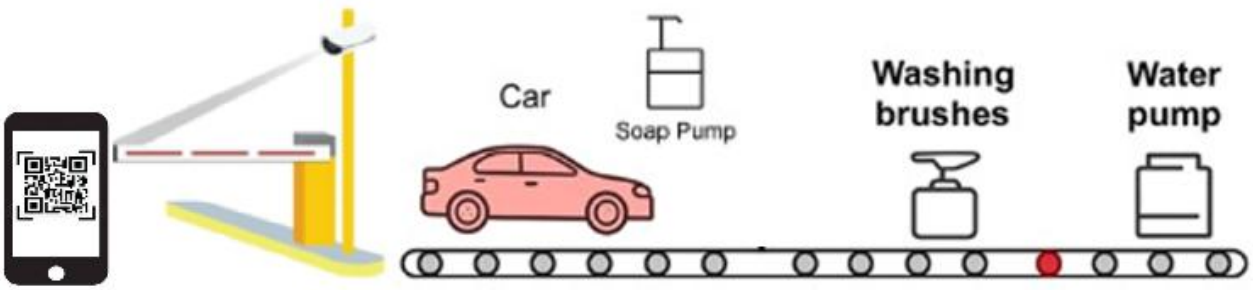

Figure1 .1 System Description

1.6 Project Limitations/Constraints:

- Limited to cars that can be detected via the camera and sensors.
- Requires stable internet for seamless communication between devices and booking systems.
- The system will initially support payments through specific gateways (like PayPal) and may need updates for new payment methods.
- Weather conditions might affect the sensor performance and require calibration.

1.7 Project Schedule

The system implementation and operation tasks are distributed along the first and second semesters, as summarized in Table 1.1.

Table1. 1 Project Schedule

	First Semester			**Second Semester**			
Week	1-4	5-10	11-15	1-4	5-10	11-14	15
Selection of Project Idea							
Collecting the Data							
System Design							
System Implementation							
System Testing							
System Operation							
Documentation							

1.8 Report Outline

This report is organized as follows: Chapter 2 presents the theoretical background and supporting technologies for the smart car wash project. It provides an overview of the key concepts and a literature review that compares our system to similar solutions. Chapter 3 details the system design, discussing the hardware and software components in detail. It also covers the theoretical background and algorithms, and includes a block diagram illustrating the system structure and operation. Chapter 4 concludes, summarizing the project's achievements, limitations, and future directions.

CHAPTER TWO: THEORETICAL BACKGROUND:

2.1 Preface:

This chapter will introduce the foundational theories and technologies behind the Smart Car Wash project. It provides an overview of the concepts and recent advancements in automation as they pertain to automated car wash system. By grounding the project in these theoretical frameworks, the chapter sets the stage for understanding the technology choices and design decisions made.

2.2 Theoretical Background Concepts:

Here are some basic theoretical concepts that underpin a Smart Car Wash System. These concepts combine principles from computer science, control systems, embedded systems to create an automated and intelligent system:

2.2.1 Embedded Systems:

A combination of hardware and software designed for specific control functions. In this system, a Raspberry Pi microcontroller acts as the central controller, responsible for processing sensor inputs

2.2.2 QR Code Recognition:

QR (Quick Response) codes store encoded data that can be easily scanned and interpreted by a camera and image-processing software. Upon booking via the website, a QR code is generated and used for identifying the user at the car wash entrance. A camera connected to the Raspberry Pi reads the QR code, triggering the start of the washing process.

2.2.3 Automation and Control Logic:

The sequence of operations detecting the car, reading the QR code, moving the conveyor, and activating pumps is governed by a predefined control algorithm within the Raspberry Pi. This logic ensures that each process runs in the correct order and duration.

2.2.4 Conveyor Belt Mechanism:

A conveyor belt is a continuously moving band used to transport objects. In the car wash, it moves the car through various cleaning stages. The conveyor motor is activated once the system verifies the presence and identity of the car.

2.3 Literature Review:

Many projects have been designed in the field of automated car washing, each with different focuses and technological implementations. Notable examples include the Automated Car Wash with Parking System (2023) [1] and the Automatic Car Washing Machine (2019). [2] These projects have incorporated various technologies, such as IoT, servo motors, ultrasonic sensors, and high-pressure cleaning systems, each aiming to improve operational efficiency and reduce resource consumption. The following section compares these projects in detail, highlighting key differences and advancements made by the current Smart Car Wash project, as summarized in Table 2.1.

Table2. 1 Comparison with other projects

Feature	Automated Car Wash with Parking System	Automatic Car Washing Machine	Smart Car Wash System
Year	2023	2019	2024-2025
Automated Gate Control	Yes, servo motor	No	Yes, by recognizing the QR code through the camera
Generate QR	No	No	Yes, for booking through website
Online Booking	No	No	Yes, for website
Real-time Feedback	Not included	Not included	Yes, through sensors
Processing unit	Raspberry Pi	Mega Arduino	Raspberry Pi
Electronic payment	No	No	Yes, for visa card

2.4 Summary

In this chapter, the theoretical background and recent developments related to automated car washes are discussed. Key concepts such as integration between system components, rapid response code generation, and automated control of the washing process, which form the foundation of a smart car wash system, are highlighted. This theoretical foundation provides insight into the innovations and improvements this project offers compared to previous similar projects.

CHAPTER THREE: SYSTEM DESIGN

3.1 Preface

This chapter provides an in-depth exploration of the design framework for the Smart Car Wash Project, highlighting the integration of hardware and software components that power the system. It examines the selection process for system components, presents alternative designs, and explains the methodologies employed to achieve seamless operation. Through detailed diagrams, algorithms, and conceptual models, the chapter aims to offer a thorough understanding of the technical architecture and the decisions that shape the system's functionality.

3.2 System Components and Design Alternatives

The Smart Car Wash Project requires a combination of hardware and software components to achieve its goals of automation, efficiency, and user convenience. This section provides a detailed analysis of the required system components, design alternatives, and the rationale behind the selected options.

3.2.1 Hardware Components

3.2.1.1 Microcomputers

As the core of the Smart Car Wash system, the microcontroller oversees and regulates various components. It facilitates communication between sensors, cameras, relays, and other hardware to ensure smooth and synchronized operation. Several microcontroller options were considered, as shows in Table 3.1.

Table3. 1 Differences between microcontrollers

Feature	Arduino	Raspberry Pi
Type	Microcontroller board	Single-board computer
Processor	8-bit AVR (e.g., ATmega328P)	Powerful, capable of running a full operating system
GPU	None	Broadcom VideoCore VI
RAM	2KB to 8KB	2GB, 4GB, or 8GB LPDDR4
Storage	There is no built-in storage, Flash memory only	MicroSD USB
Number of Pins	14 digital, d analog	40 GPIO
I/O Ports	Simple embedded systems like blinking LEDs, sensors	Complex systems like ioT hubs, AI and multimedia apps
Power Supply	Limited to program memory (e.g., 32KB on Arduino Uno)	Expanded via SD card
Wireless Connectivity	Requires modules for Wi-Fi, Bluetooth, or Ethernet	Built-in connectivity options (in some models)

The Raspberry Pi 4 Model B is a powerful single-board computer developed by the Raspberry Pi Foundation. It features a quad-core ARM Cortex-A72 processor running at 1.5 GHz, and it comes with options for 2GB, 4GB, or 8GB of LPDDR4 RAM, making it suitable for multitasking and high-performance applications. The board includes dual micro HDMI ports that support 4K video output, USB 3.0 ports, Gigabit Ethernet, and built-in Wi-Fi and Bluetooth. It also has a 40-pin GPIO header for connecting sensors and modules, which makes it ideal for embedded systems and electronics projects. The Raspberry Pi 4B runs a variety of operating systems, including Raspberry Pi OS (Linux), and can be used

for education, media centers, IoT, and even desktop computing. as show in figure 3.1.

Figure3. 1 Raspberry Pi 4B[3]

3.2.1.2 Ultrasonic Sensors

The ultrasonic sensor is a distance measuring device utilized in the Smart Car Wash system to detect and monitor the car's position. The sensor emits ultrasonic waves, which bounce back upon hitting an object, allowing the system to calculate the distance based on the time taken for the echo to return. When the car reaches specific points in the system, the ultrasonic sensor triggers actions such as activating water sprays. Additionally, it plays a role in ensuring that cars are correctly positioned on the conveyor belt, and optimizing the washing process, as show in figure 3.2 .

Figure3. 2 Ultrasonic Sensor[4]

3.2.1.3 Camera

The camera is an essential part of the smart car wash system. It is responsible for capturing a QR code generated on the website for the customer who has booked a car wash appointment. The code is recognized by the camera to initiate the wash process. Table 3.2 below details the available camera options.

Table3. 2Differences between cameras

Feature	Raspberry Pi Camera Module	Logitech C920	ESP32 Camera (OV2640)
Image Sensor	CMOS	CMOS	OV2640
Video Resolution	1080p@30fps, 720p@60fps	1080p@30fps	1080p@30fps, 720p@60fps
Compatibility	Compatible with Raspberry Pi	Compatible via USB	Compatible with ESP32
Price	$25-$35	$70-$100	$10-$20

We have selected the Raspberry Pi Camera Module [5] for its high resolution, seamless Raspberry Pi compatibility, cost-effectiveness, and compact design, making it a better choice than the ESP32 Camera and Logitech C920 [6], it provides a simple and effective solution for our needs.

3.2.1.4 Relay 4 channel

The relay board can be used to turn lights, fans, and other devices on and off while maintaining isolation from the microcontroller. The relay coil has a 5V voltage, allowing it to be used directly with any 5V voltage regulator. This relay board allows you to control high-power devices (up to 10A) via the built-in relay. The relay is controlled via a 1 x 3 connector, making it suitable for servo cables and easy to connect to many development boards. It was used in a car wash project to operate water and soap pumps and also to operate motors, as show in figure 3.3.

Figure3. 3 Relay 4 channel[7]

3.2.1.5 Water Pump

We used DC Water Pumps to pump water to the car wash. These pumps are efficient, reliable, and capable of providing the necessary pressure for effective cleaning. The water pumps are controlled by the raspberry pi via a relay, ensuring that water is delivered automatically when needed. The following table 3.3 shows the difference between two types of pumps. The Micro Peristaltic Ink pump was tested because it is the most suitable for the project, as show in figure 3.4.

Figure3. 4 Water Pump[8]

Comparison: Micro Peristaltic Ink Pump vs Micro Oxygen Pump

Table3. 3 Comparison Between Pumps

Feature	**Micro Peristaltic Ink Pump (DC 5V)**	**Micro Oxygen Pump (Fish Tank)**
Pump Type	Peristaltic Pump	Diaphragm or Centrifugal (Mini Air/Water Pump)
Working Principle	Uses rollers to squeeze flexible tubing to move liquid	Uses diaphragm or fan to push air or water
Fluid Type	Liquids only (e.g., ink, water, medicine)	Primarily air, sometimes small water volumes
Flow Precision	Very high precision, controllable flow	Low precision, continuous low-pressure flow
Output Pressure	Low to medium, steady and accurate	Very low, suitable for aquarium oxygenation
Common Uses	Printers, medicine dispensing, dosing systems	Fish tank aeration, small air circulation

Viscous Liquid Handling	Yes, handles low-viscosity fluids well	No, not suitable for viscous or thick fluids
Power Supply	DC 5V	DC 5V or 12V
Noise Level	Relatively low	Moderate – noticeable vibration/sound
Maintenance	Easy – only tubing needs replacement	Minimal unless it breaks

3.2.1.6 Conveyor belt -DC motor

The Conveyor Belt is used in the Smart Car Wash project to transport vehicles We selected a DC motor-powered conveyor system for its efficiency, reliability, and ease of integration with the RPI. The conveyor belt is controlled by a motor connected to a relay, which is managed by the RPI. This ensures smooth movement of the vehicles from the entry point to the washing area. The automated system reduces manual labor and ensures consistent, efficient operation, enhancing the overall performance of the car wash, as show in figure 3.5.

Figure3. 5 Conveyor belt & DC Motor[9]

3.2.1.7 Gate Actuator

To control the entry and exit of vehicles in the Smart Car Wash project, we selected a Gate Actuator. This actuator provides precise and reliable movement, allowing the gate to open and close smoothly based on vehicle detection. The actuator is controlled by the Raspberry Pi via a relay, ensuring automated operation. When a vehicle is detected and authorized, the actuator opens the gate,

this system ensures smooth entry and exit, enhancing the overall automation of the process, as show in figure 3.6.

Figure3. 6 Gate Actuator[10]

3.2.2 Software Components:

This section will provide some information about the main programs used in our project.

3.2.2.1 Raspbian

Although Raspberry Pi can utilize different types of Linux distributions for its default operating system (OS), Raspbian is the recommended one. It is a free operating system based on Debian, optimized for the Raspberry Pi hardware. It comes with over 35,000 packages; precompiled software bundled in a good format for easy installation on Raspberry Pi hardware.

A free operating system based on Debian optimized for the Raspberry Pi hardware. Coming with over 35,000 packages, pre-compiled software. This operating system provides significantly faster performance for applications that make heavy use of floating-point arithmetic operations. [11]

3.2.2.2 Thonny

Thonny is an integrated development environment (IDE) designed specifically to simplify the Python programming experience, with a focus on ease of use and smooth learning. It features a beginner-friendly interface, making it an ideal choice for students and people entering the field of programming for the first time. Thonny provides basic and advanced tools to write, run, and debug code

effectively. Thonny is a powerful learning tool, offering features that make learning fun and simple. It is designed to help users gain a deeper understanding of the basics of Python by visually and systematically displaying programming operations. For example, you can follow the execution of code step by step and see how values inside variables change and how resources are allocated inside memory. [12]

3.2.2.3 XAMPP

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible. XAMPP is regularly updated to the latest releases of Apache, MariaDB, PHP and Perl. It also comes with a number of other modules, including OpenSSL, phpMyAdmin, MediaWiki, Joomla, WordPress and more. [13]

3.3 Conceptual system description

The following figure 3.7 represents a block diagram of a smart car wash system that uses a Raspberry Pi as the central control unit. Bookings are made in advance through a website where users log in and set a time and date. A QR code is then generated which is used to identify the vehicle upon arrival at the car wash via a camera connected to the controller. After the booking is confirmed the first ultrasonic sensor detects the vehicle's entry onto the conveyor belt. The controller then gradually activates the wash components through relays, the conveyor belt is first activated followed by the soap pump to spray detergent. The wash brushes and water pump are then activated and the ultrasonic sensor then detects the vehicle's arrival at the end of the belt and this system automates the entire car wash process with high efficiency and without direct human intervention.

The following figure 3.8 illustrates the concept of the System Conceptual Diagram, which is considered one of the basic elements in systems analysis and design. This diagram was used to understand and represent the basic components of the car wash system.

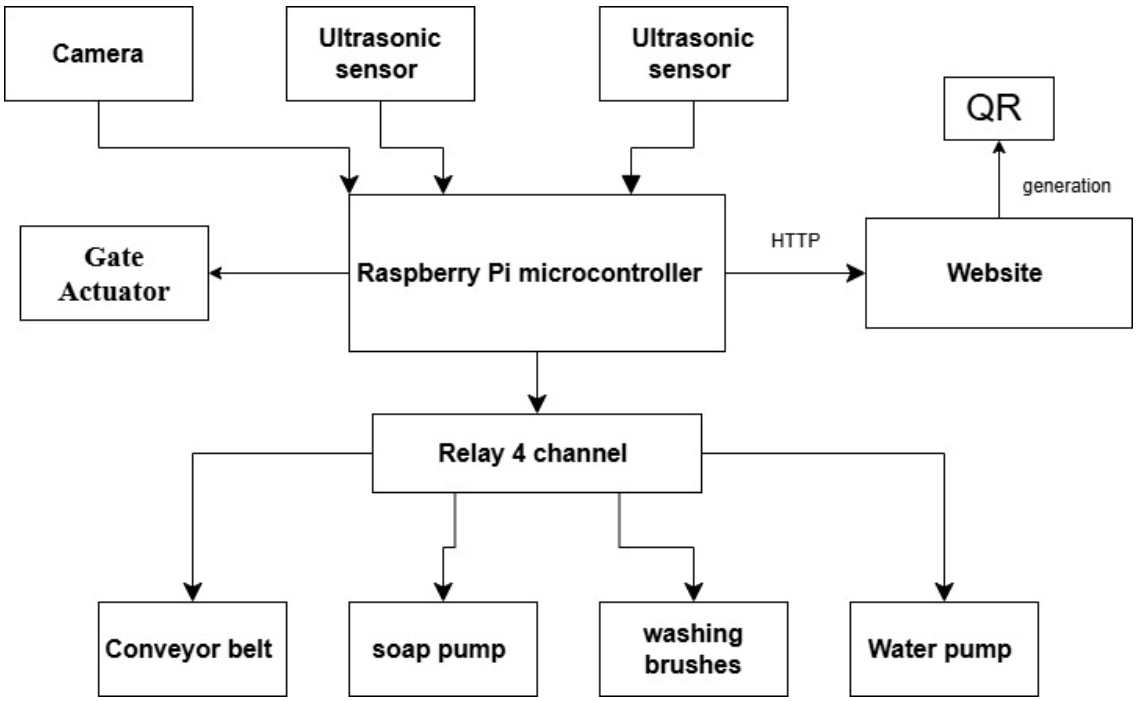

Figure3. 7 System Block Diagram

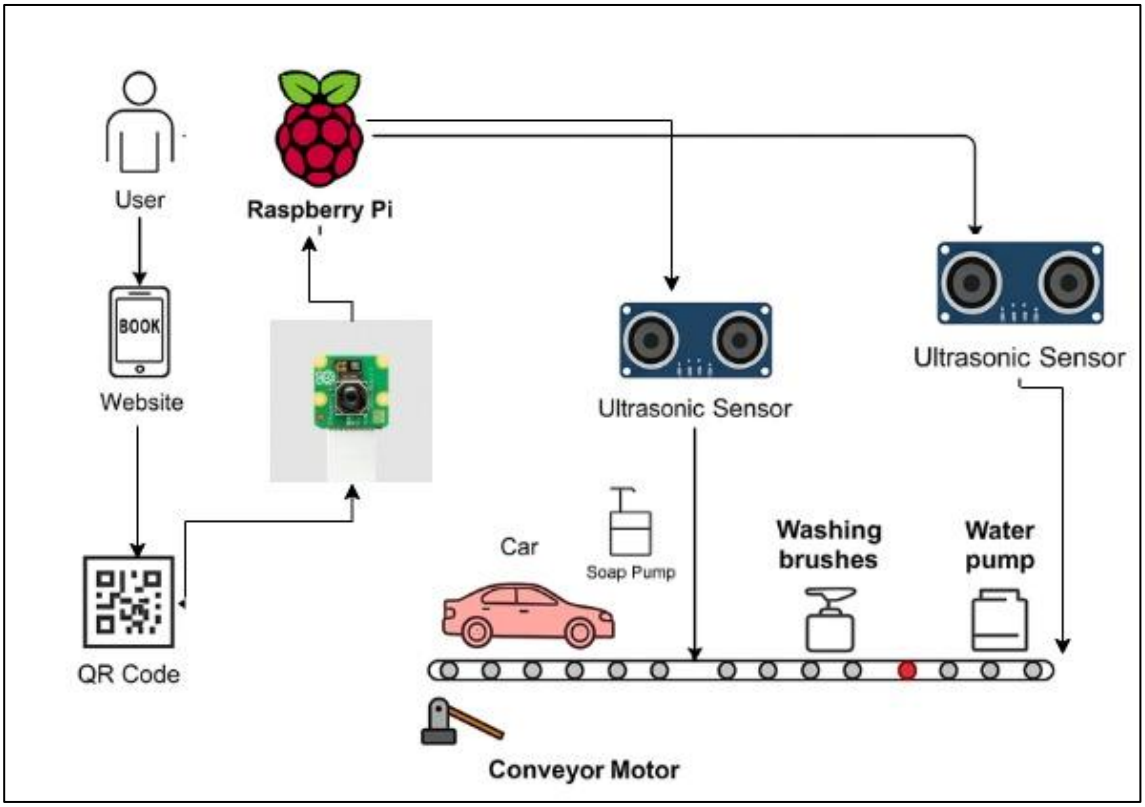

Figure3. 8 System conceptual diagram

3.4 Algorithms and methodologies

3.4.1 Flow chart

Figure 3.9 below shows a flowchart of the processes for a smart car wash system, starting from booking online and obtaining the QR code, until the start and end of the wash process.

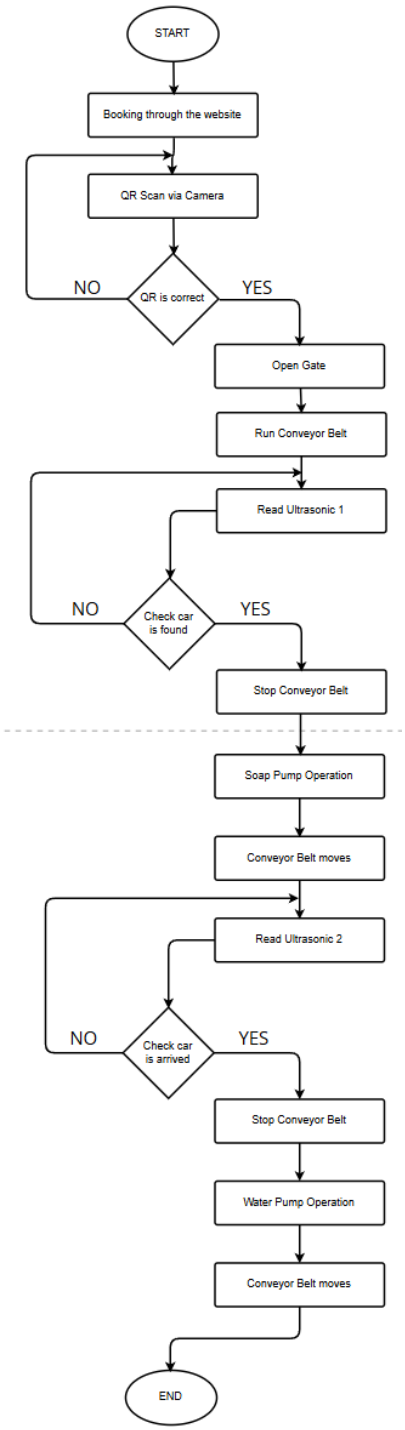

Figure3. 9 Flowchart

3.4.2 Use Case Diagram

Figure 3.10 shows the use case of the Car Wash Booking and Entry Process:

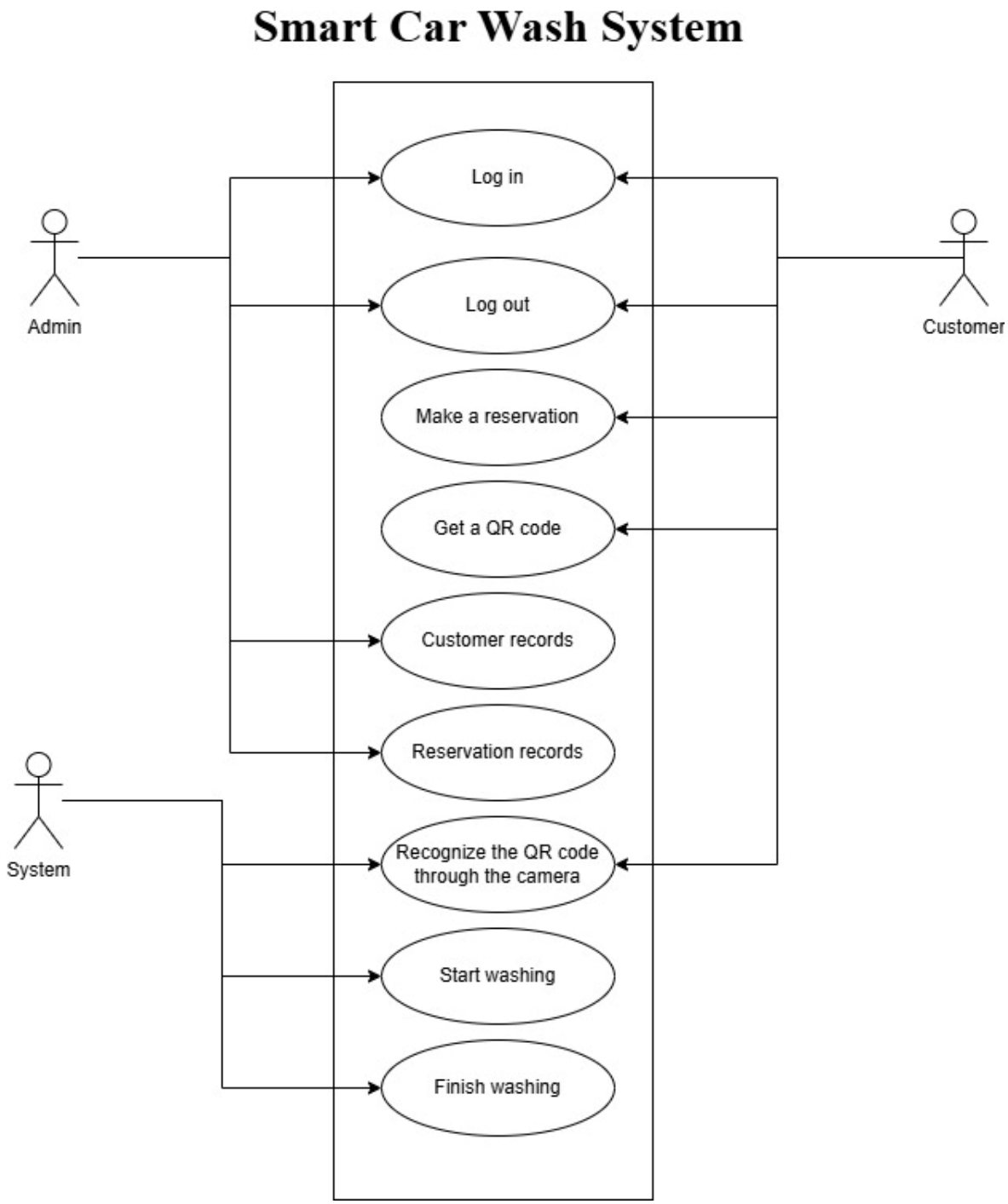

Figure3. 10 Use Case Diagram

3.4.3 Sequence Diagram

Figure 3.11 shows the sequence diagram of the reservation and car wash process.

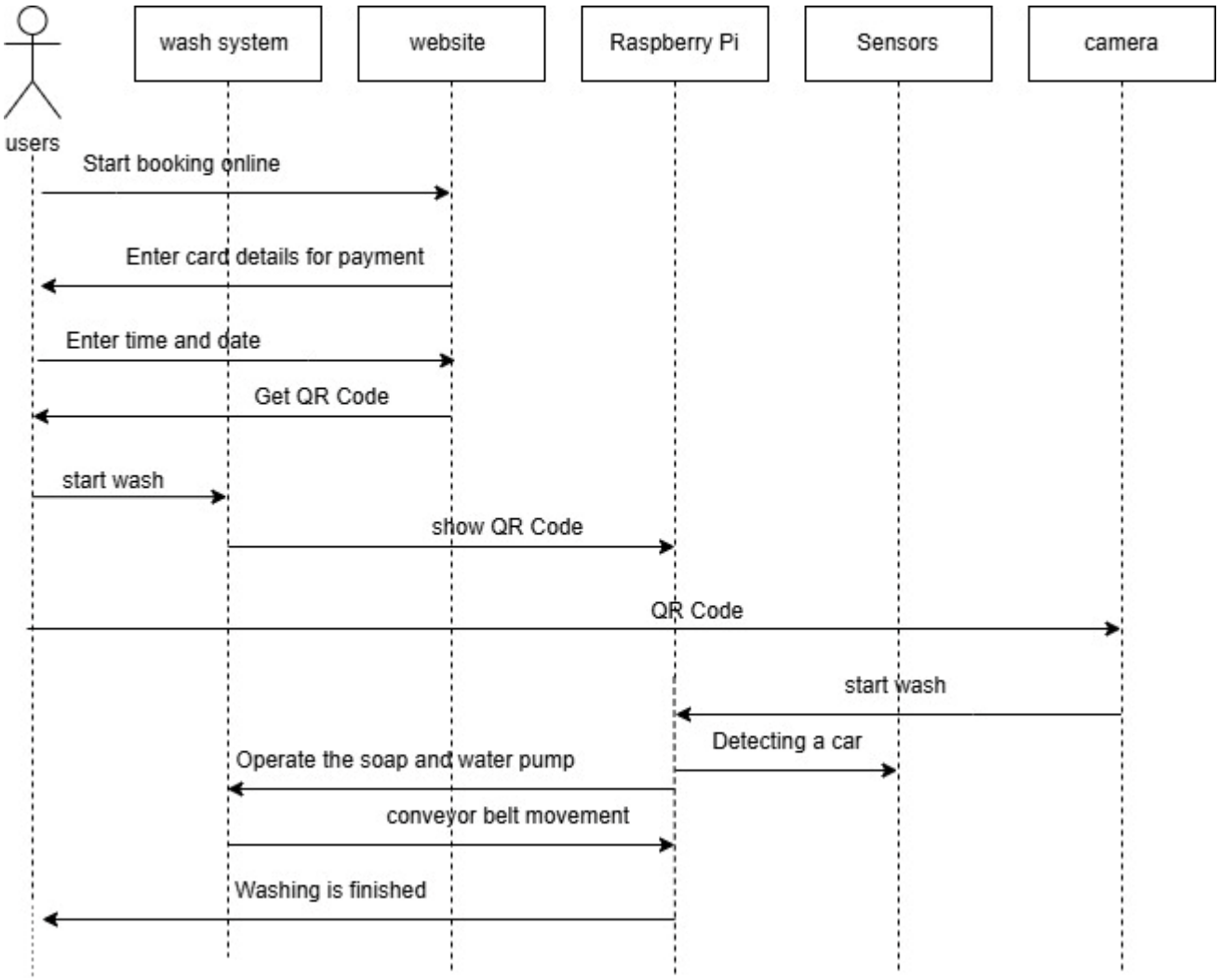

Figure3. 11 Sequence Diagram

3.5 Schematic diagram

Figure 3.12 shows the connection between the main components in smart car wash project.

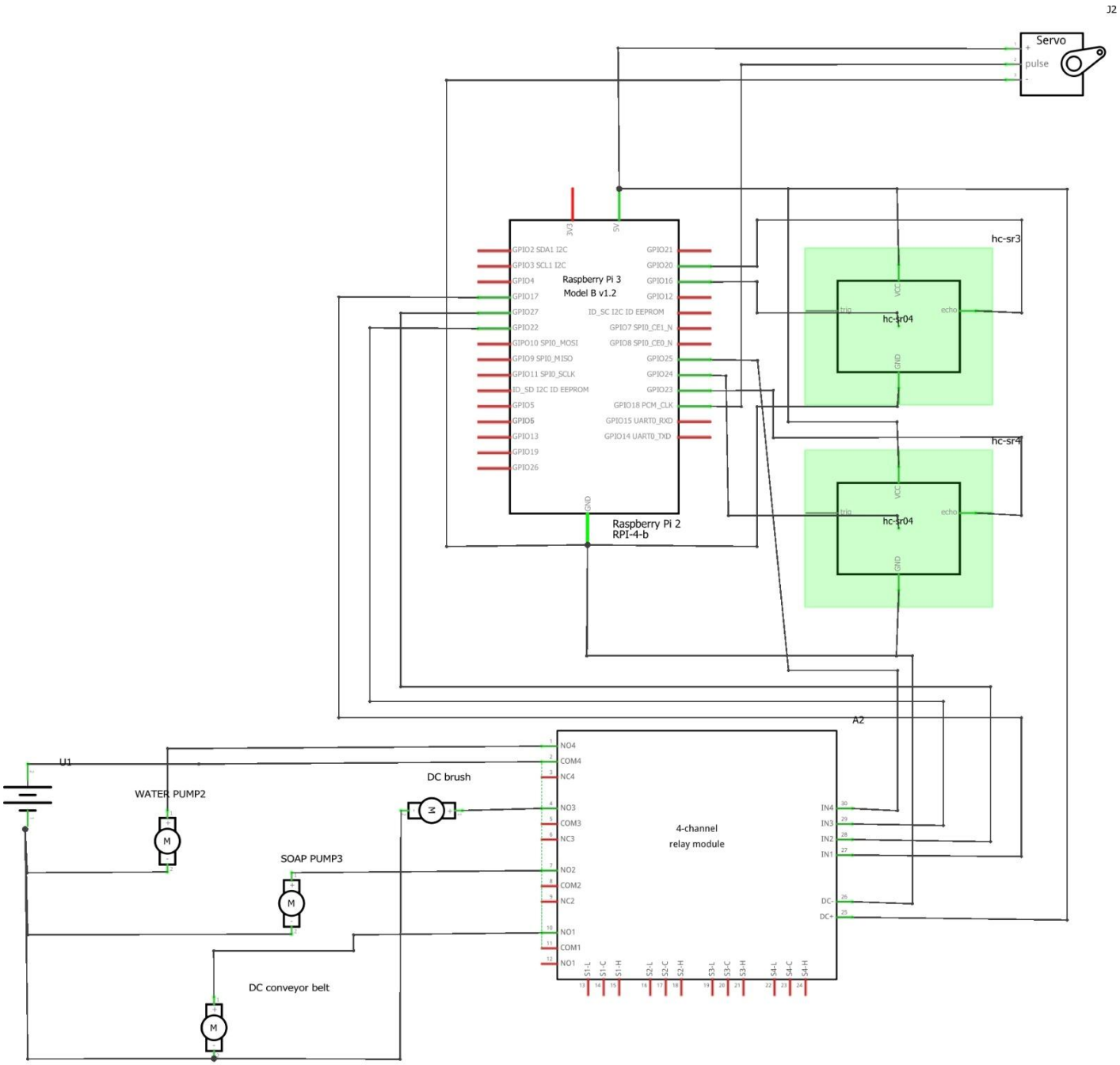

Figure3. 12 Schematic Diagram

3.6 Summary

This chapter covered the hardware and software components of the Smart Car Wash system, evaluating alternatives and selecting optimal solutions. Key designs were illustrated with block diagrams, schematic diagrams, and algorithms to demonstrate system functionality and flow. Essential methodologies, including flowcharts and use cases, were presented to ensure clarity and guide implementation.

CHAPTER FOUR: IMPLEMENTATION AND TESTING

4.1 Preface

In this chapter, we will discuss the implementation of the software and hardware components of a smart car wash project. We will begin by discussing implementation issues, reviewing the project website, including the connections of the electronic components, reviewing implementation challenges, and validating and testing the project.

4.2 Implementation issues

During the implementation of the smart car wash system, we encountered several issues, including:

1. Purchasing a number of dedicated water and soap pumps to ensure the best pump in terms of pumping power, size, and efficiency, as show in figure 4.1.

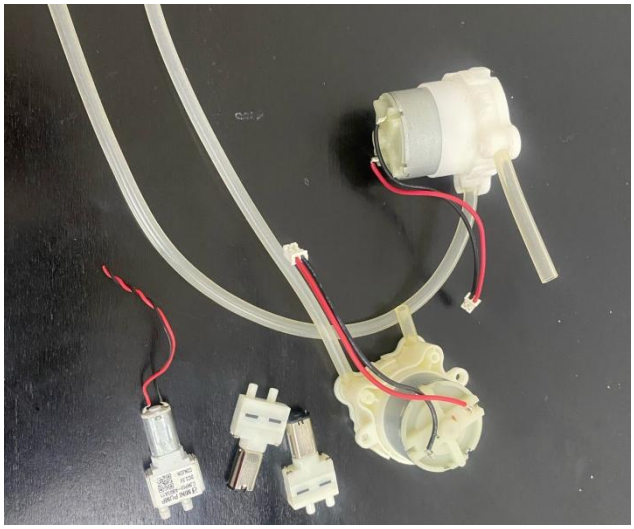

Figure4. 1 Types Of Pumps

2. Sensor accuracy: Small errors in sensor values led to vehicle undetected or incorrect timing for dispensing water and soap.
3. Integrating the website with the hardware required a significant amount of time.

4. Another implementation issue was determining the placement of sensors on the model, which required us to reposition the sensors multiple times to achieve the best placement.
5. Ensuring the power needed to operate the pumps and sensors to prevent them from burning out, as show in figure 4.2.

Figure4. 2 Power Converters

6. Poor network performance or internet outages led to delayed system commands or data loss.
7. Imposing restrictions on bookings where there is no booking conflict.
8. How to get QR code from website to make reservation inside laundry, as show in figure 4.3.

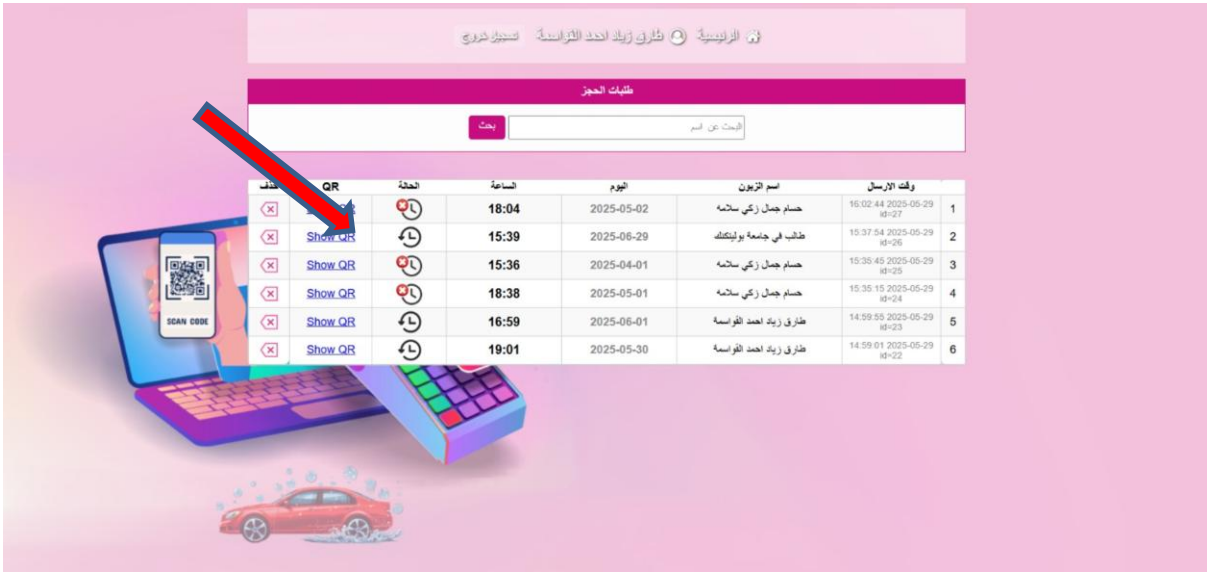

Figure4. 3 Get QR Code

4.3 Implementation challenges

We faced many challenges during the implementation phase, the most prominent of which was the process of randomly generating a QR code for each person who booked on the website as shows in figure 4.4. The difficulty was in the process of creating it and linking it to the time and date in order to avoid errors such as giving the same QR code to more than one person or booking at the same time. In addition, we faced a problem in designing the conveyor belt for the car as we had to choose the best materials to make the belt so that the movement would be precise and consistent and also to determine the location of each sensor to perform specific tasks such as operating the water or soap pumps.

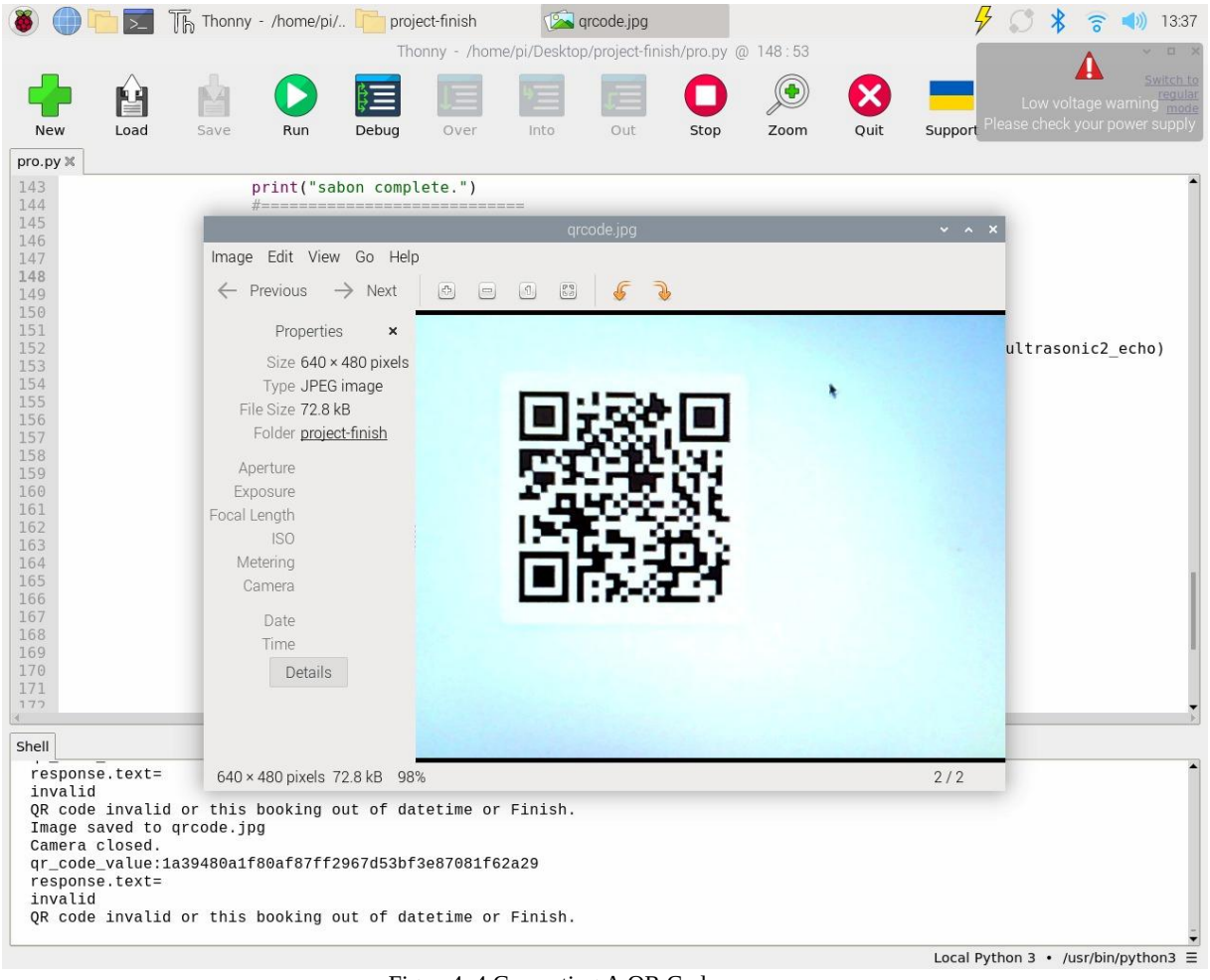

Figure4. 4 Generating A QR Code

4.4 Software Implementation

At this stage, designs and plans are transformed into a practical application, including programming and hardware connection. All specifications and requirements are implemented to create a complete, usable software system, ensuring seamless interaction with the microcontroller. The best methods are also researched and the most appropriate software libraries are selected to achieve optimal implementation.

4.4.1Database

The following figure 4.5 shows the database tables for the website that was designed. It contains several tables, including admin, customer, and customer booking.

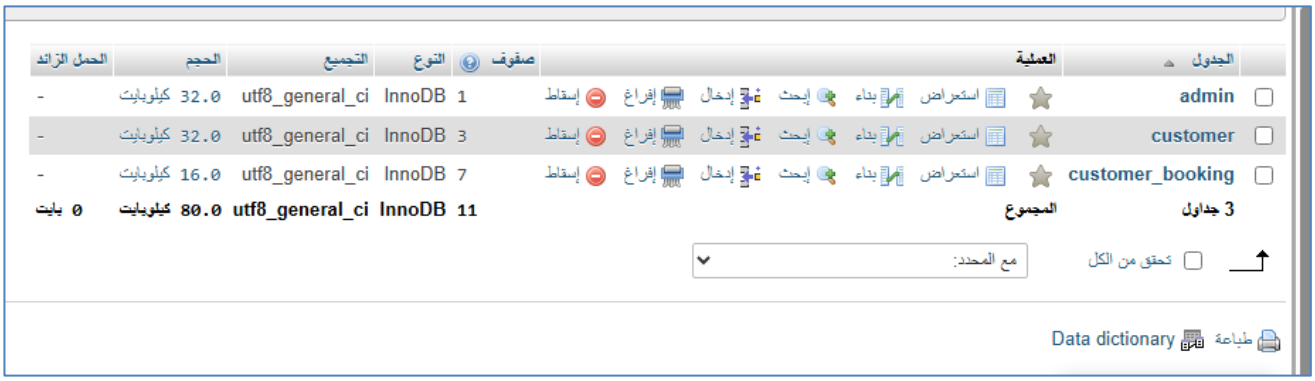

Figure4. 5 Database Tables

4.4.2 Login page

The following figure 4.6 shows the login page of the website, where you can log in to the website using username and password.

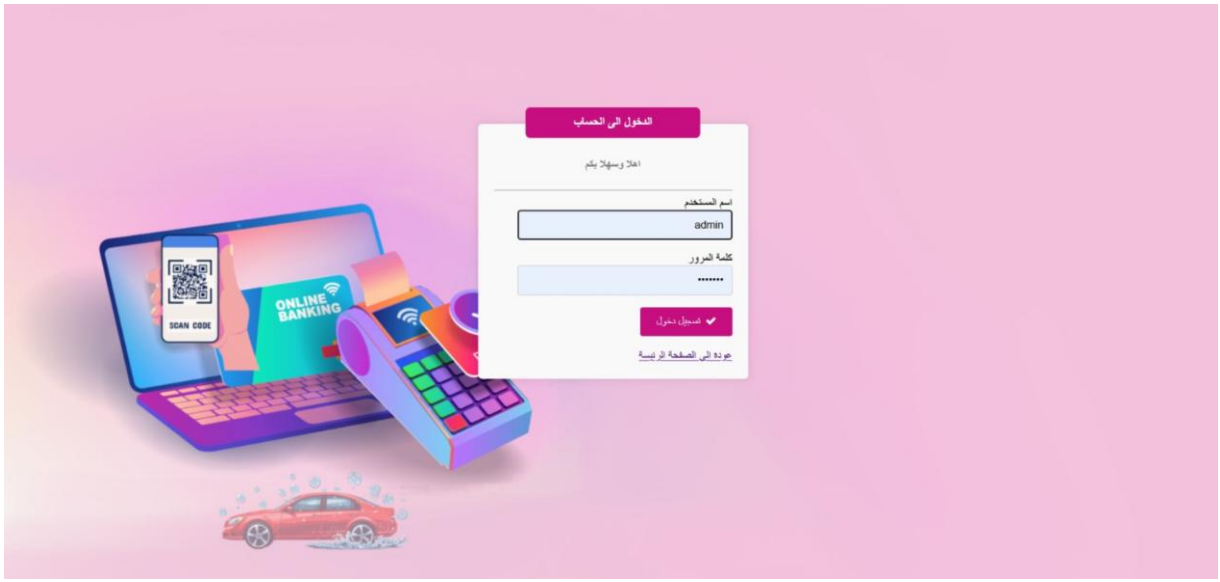

Figure4. 6 Login Page

4.4.3 Main page

The following figure 4.7 shows the main interface of the site, through which customer accounts and management accounts can be viewed, in addition to all reservations.

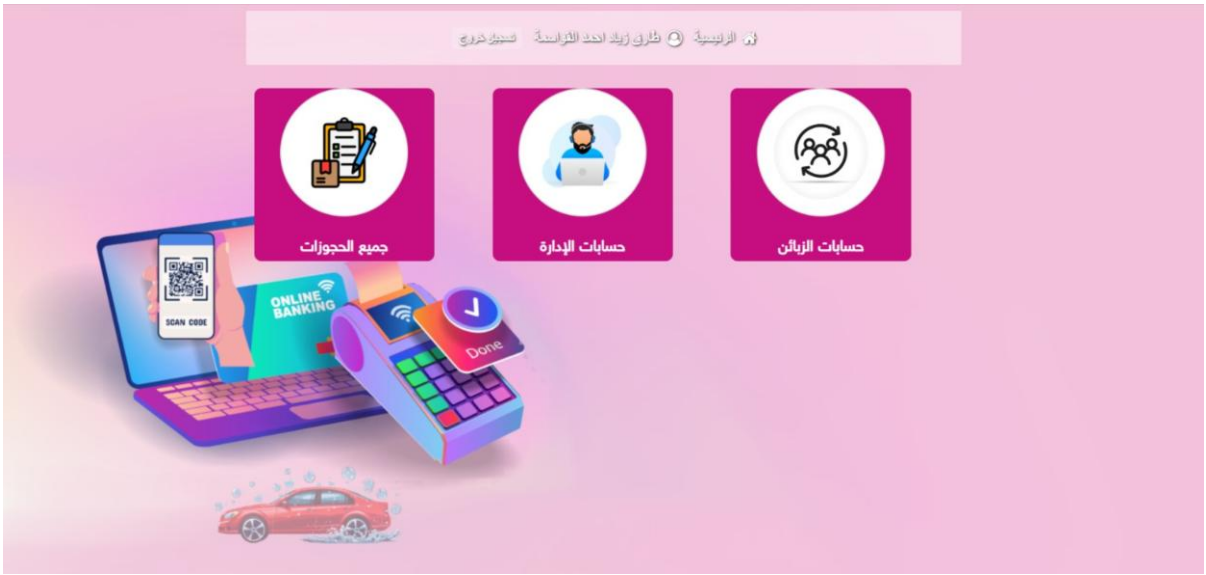

Figure4. 7 Main Page

4.4.4 Create an account

Figure 4.8 represents the interface for creating a customer account. Through this interface, data is entered, including the full name, user name, and password. The password is then confirmed, and the account is then created.

Figure4. 8 Create an account

4.4.5 The main interface for the customer account.

The following figure 4.9 shows the main interface of the customer account, through which he can make a new reservation or view the history of his existing reservations.

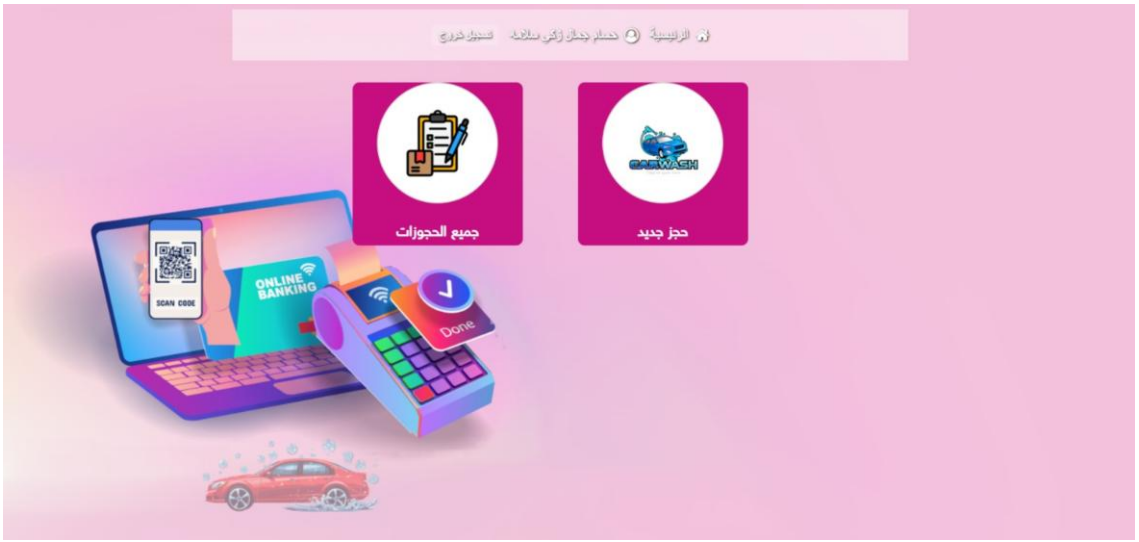

Figure4. 9 The main interface for the customer account.

4.4.6 New reservation

The following figure 4.10 shows a new reservation made by the customer, where the time and date are specified and the reservation is sent.

Figure4. 10 New reservation

4.4.7 Reservation request history.

Figure 4.11 below shows a history of reservation requests made by a customer, showing the time, date, submission time, customer name, status, and QR code.

Figure4. 11 Reservation Request History

4.4.8 Customer Information.

The following figure 4.12 shows the customer data, where the bank card data, name, username, and password can be modified.

Figure4. 12 Customer Information.

4.5 Hardware Implementation

Implementing the system requires connecting the electronic components to the RPI controller to program them to meet all the requirements of the smart car wash system. The figure 4.13 shows how to connect the electronic components to operate the system and ensure accurate readings and communication between them.

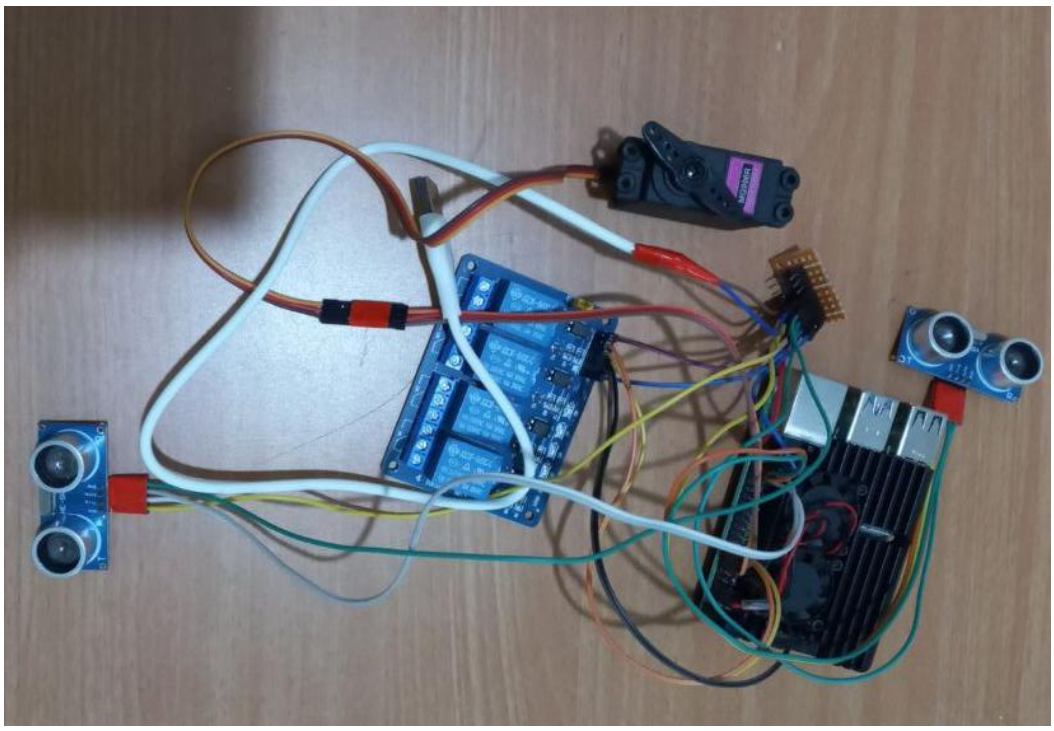

Figure4. 13Connect The Electronic Components

4.6 Validation and testing

To ensure the smart car wash system operates efficiently and accurately according to all design specifications, we conducted comprehensive verification and testing of all system components. The goal was to verify that all hardware components, software functions, and system interactions functioned correctly under various operating conditions.

4.6.1 Unit Testing:

Each hardware component (water pump, ultrasonic sensors, servo motors, soap pump, controller, and relays) was tested to ensure proper operation and accurate readings as show in figure 4.14. The unit was then linked to the software unit by first booking online, then obtaining a QR code for each customer and scheduling an appointment and time as show in figure 4.15.

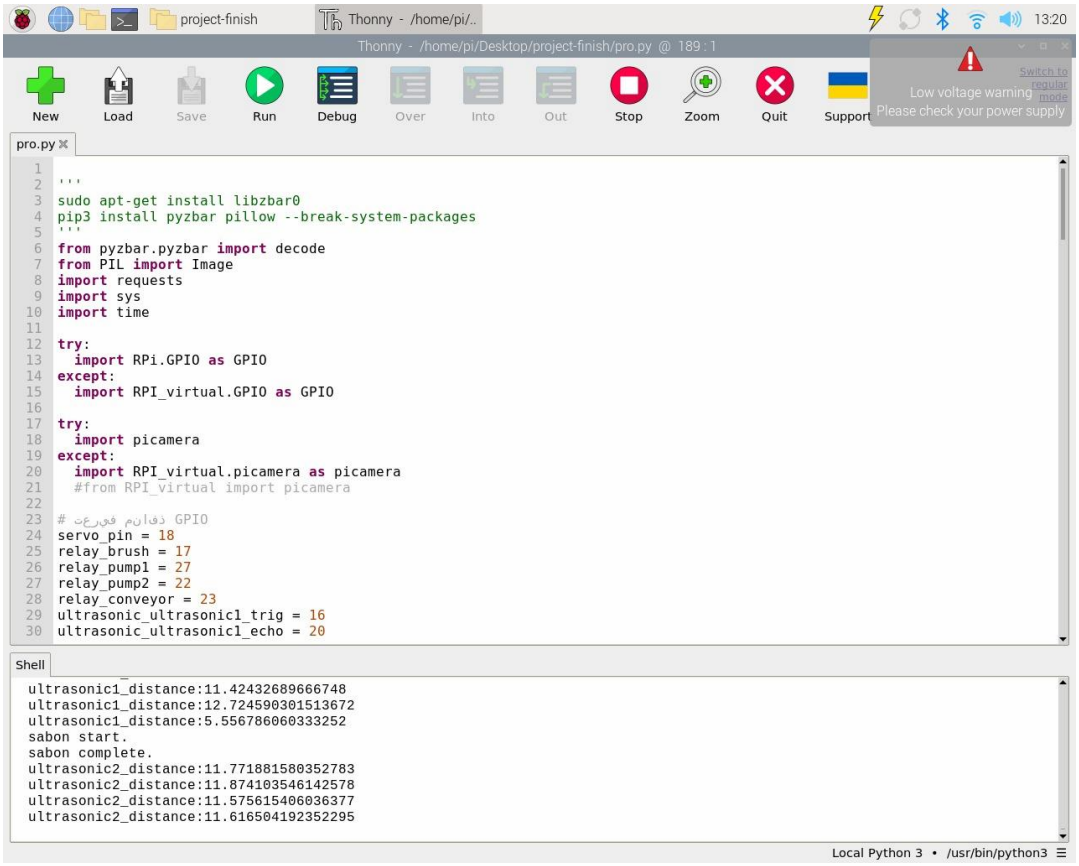

Figure4. 14 Hardware Testing

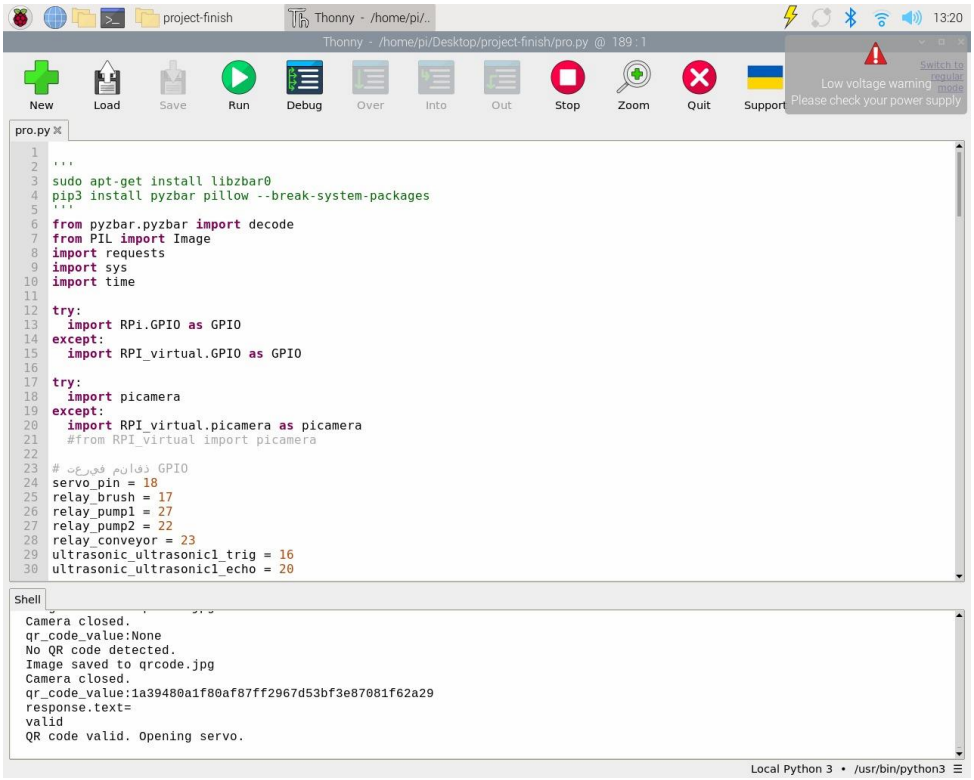

Figure4. 15 QR Testing

4.6.2 Software Testing:

At this stage a series of website tests were conducted to ensure its readiness for actual use and compliance with all system requirements. The testing process aims to verify performance and ease of use security and compatibility with various devices and browsers. The following figure 4.16 illustrates the process of adding a customer to the website by entering and saving their data and verifying the completion of the addition.

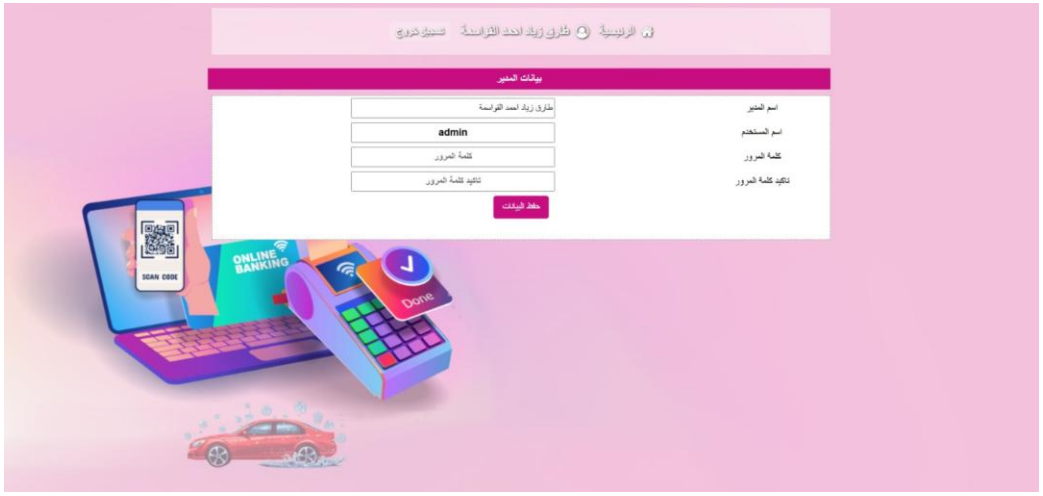

Figure4. 16 Adding A Customer

The following figure 4.17 illustrates the process of verifying the addition of a new customer.

Figure4. 17 Verifying the Addition Customer

The following figure 4.18 illustrates the process of adding a new administrator to the website by entering his data, saving it, and verifying that the addition process is complete.

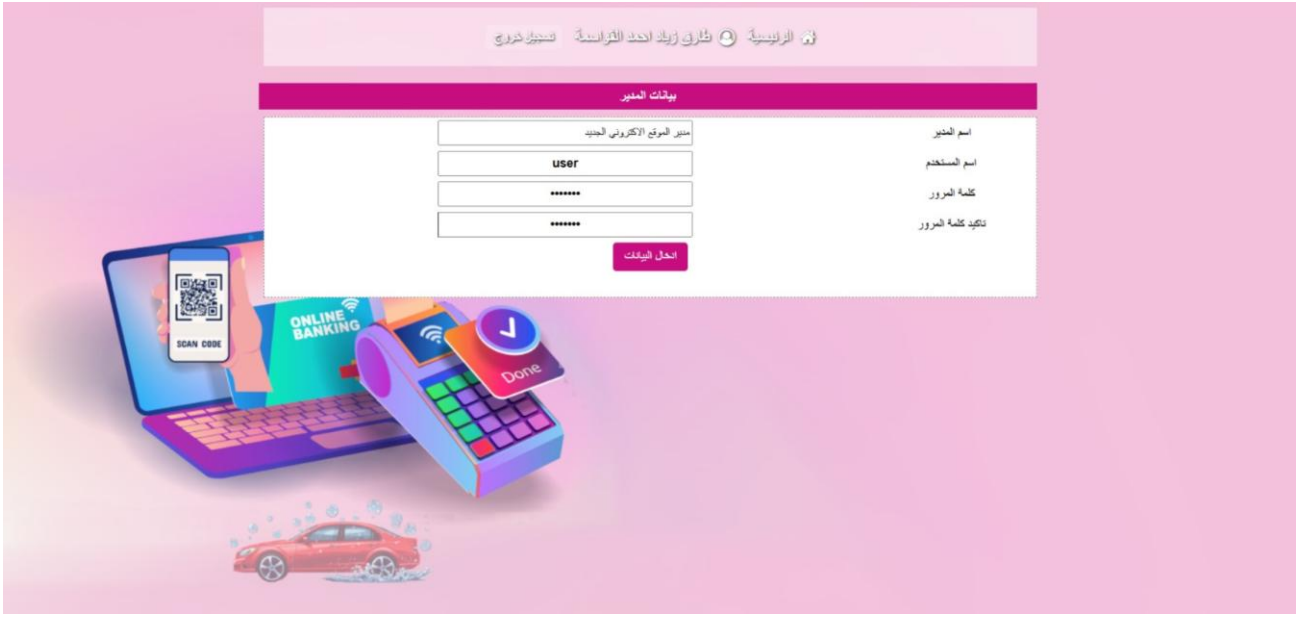

Figure4. 18 Adding A New Administrator

The following figure 4.19 shows the customer search field in the reservation requests being tested by entering the customer's name and pressing the search button to ensure that the search process is correct.

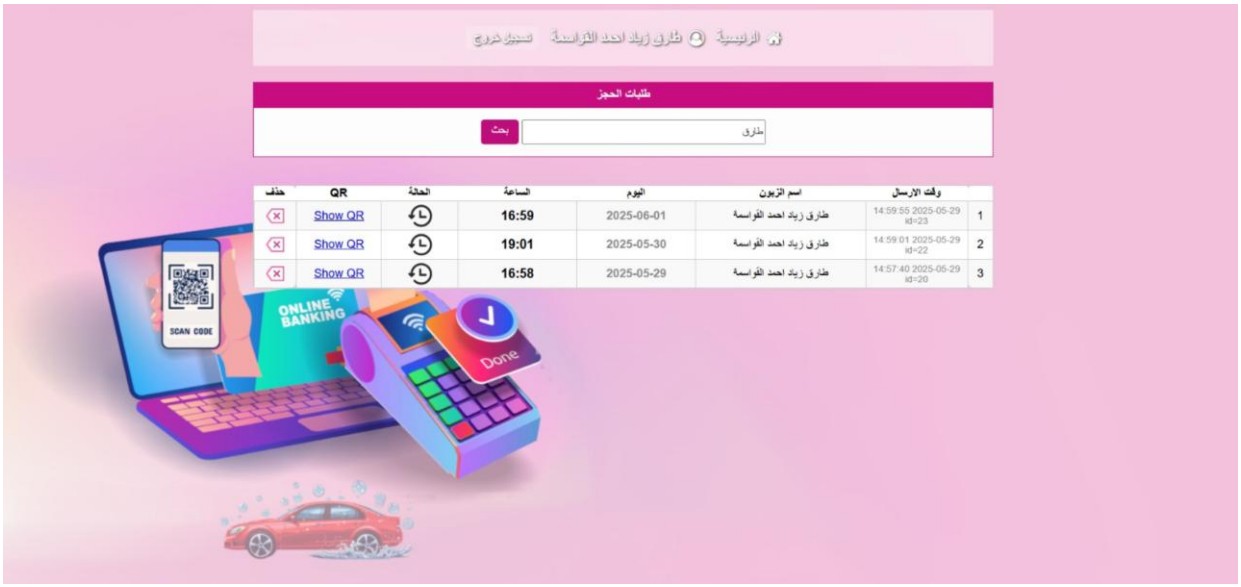

Figure4. 19 Customer Search

Figure 4.20 shows a test of displaying the QR code for each customer in a new tab.

Figure4. 20 Displaying The QR Code

4.6.3 Hardware Testing:

The following figure 4.21 shows how to test the ultrasonic sensor and obtain different readings, in addition to starting or stopping the wash.

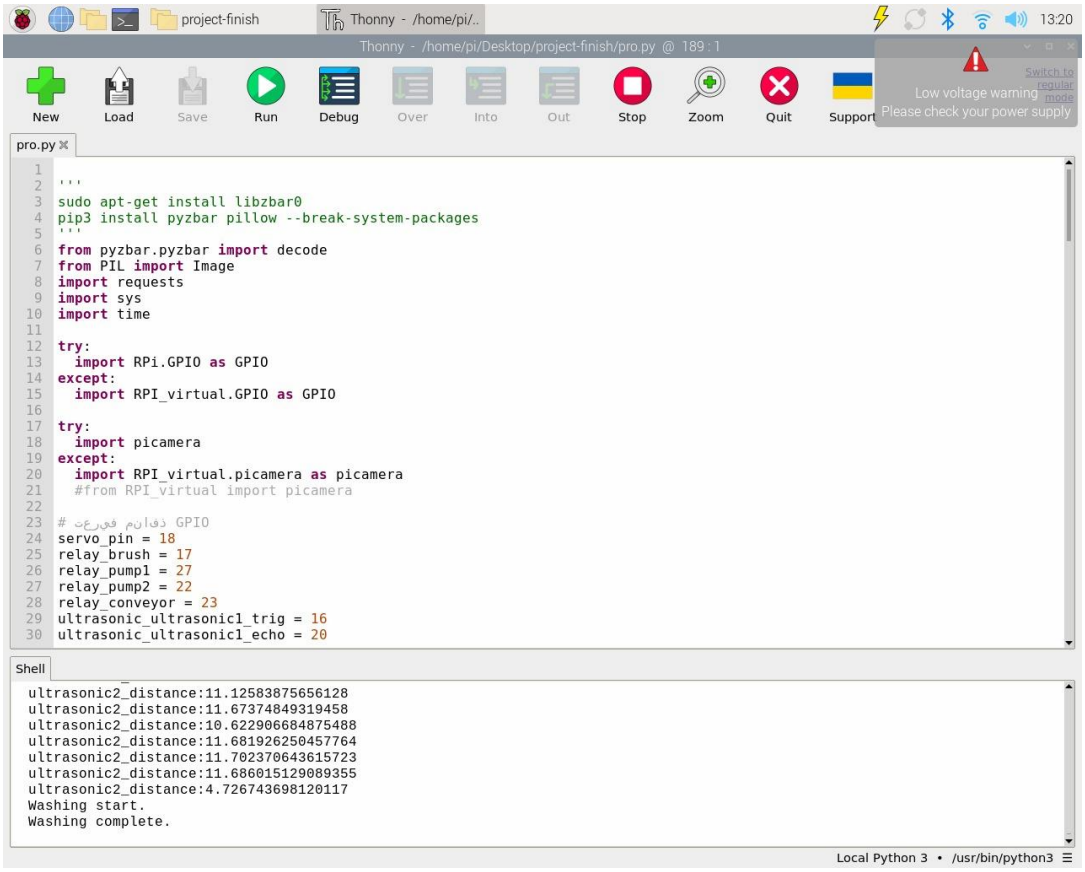

Figure4. 21 Ultrasonic sensor testing

The following figure 4.22 shows how the camera tests the QR code reading, determining whether the code is recognized or not. If it is recognized, the QR code value is given.

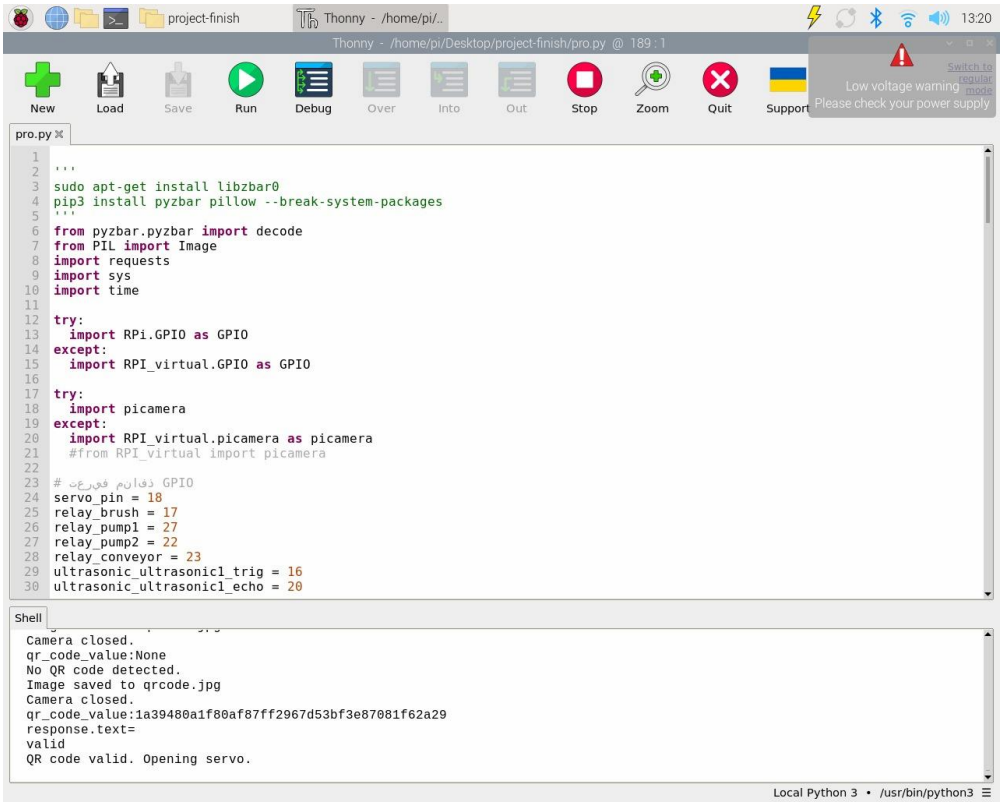

Figure4. 22 Camera testing

The following figure 4.23 shows how to test the soap pump. The vehicle is detected and then the soap pump is turned on.

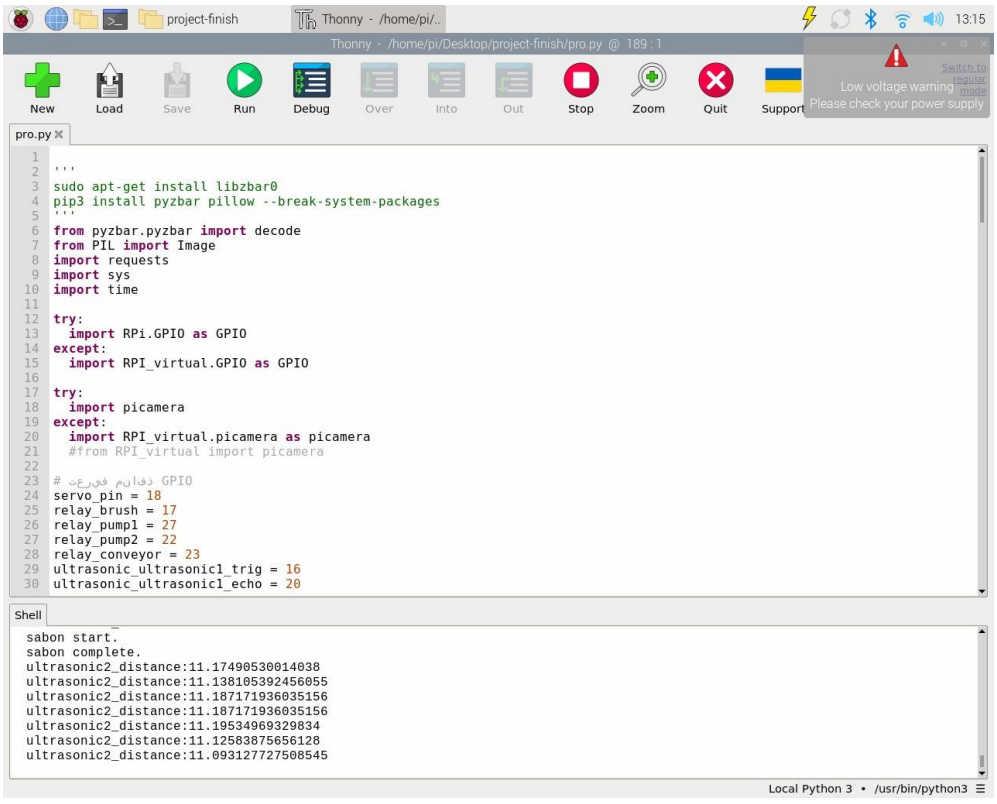

Figure4. 23 test the soap pump

The following figure 4.24 shows how to test the start of the washing process, where the presence of a vehicle is determined and then the washing process begins.

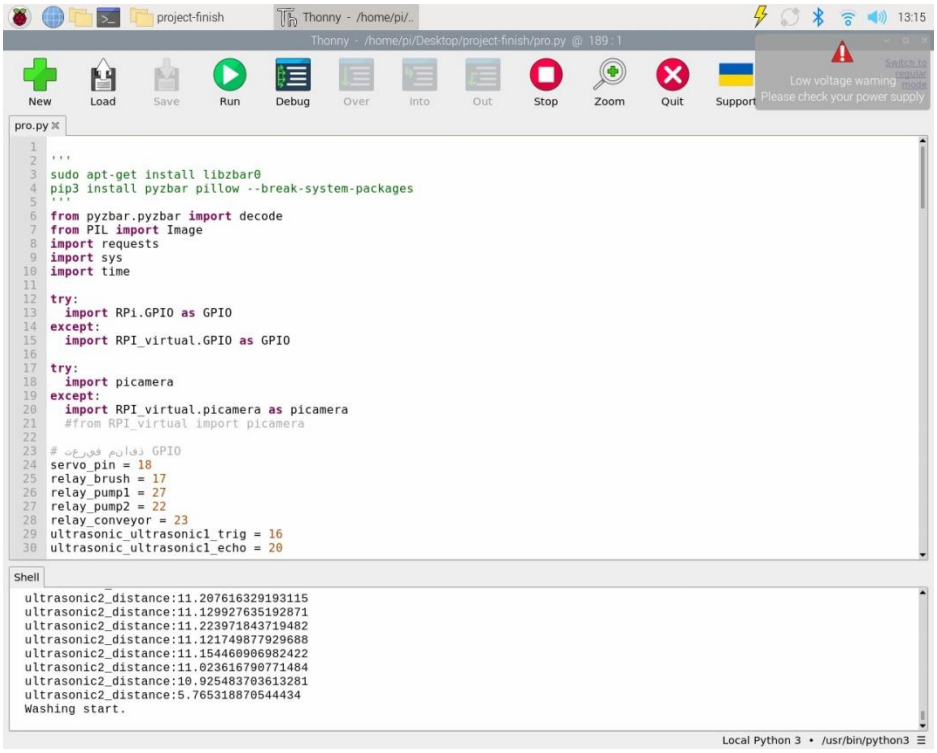

Figure4. 24 test the start of the washing

System testing involves installing all components on the model and ensuring that they work and perform as shown in the figure 4.25, which shows the conveyor belt, the motors for movement, and the location of the pumps.

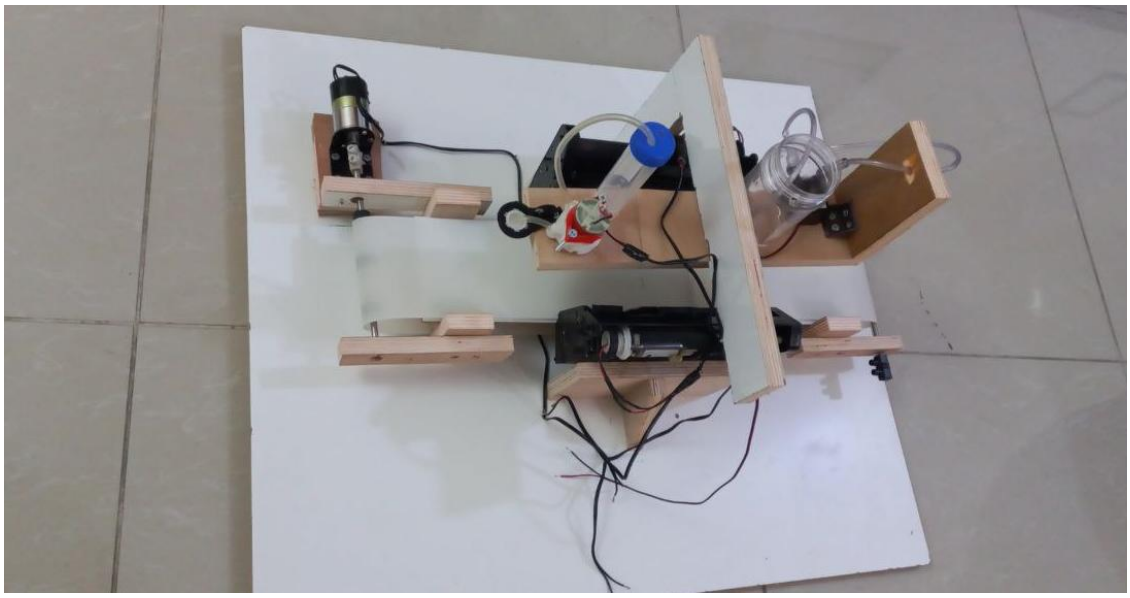

Figure4. 25 Hardware Testing

4.7 Summary

In this chapter, we discussed implementation issues by identifying the problems we faced during work. We also discussed how to implement the software and hardware components, in addition to presenting the challenges we faced when working on implementing the project.

CHAPTER FIFTH : RESULTS AND DISCUSSION

5.1 Detailed analysis of the results/experiments

The results of the smart car wash project were analyzed based on a set of key factors that form the foundation of the project, including mechanical, electronic, and software aspects, to achieve optimal system performance. These include:

1) We manufactured a conveyor belt to move cars through the various wash stages automatically and systematically as show in figure 5.1 .

Figure5. 1 Conveyor Belt

2) Two types of pumps were selected for water and soap, based on the quantities they pump, as water is pumped in larger quantities than soap as show in figure 5.2 .

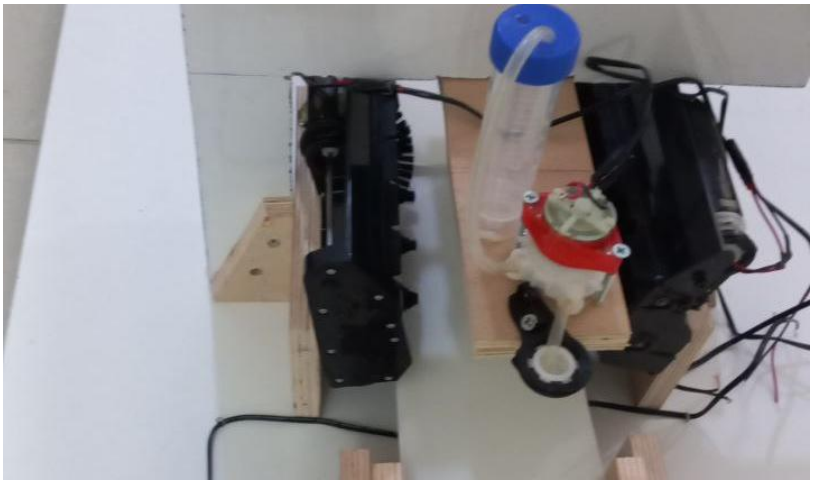

Figure5. 2 Pumps in Belt

3) The customer can log into their account and make a new reservation or view their reservation records, as shown in the figure 5.3.

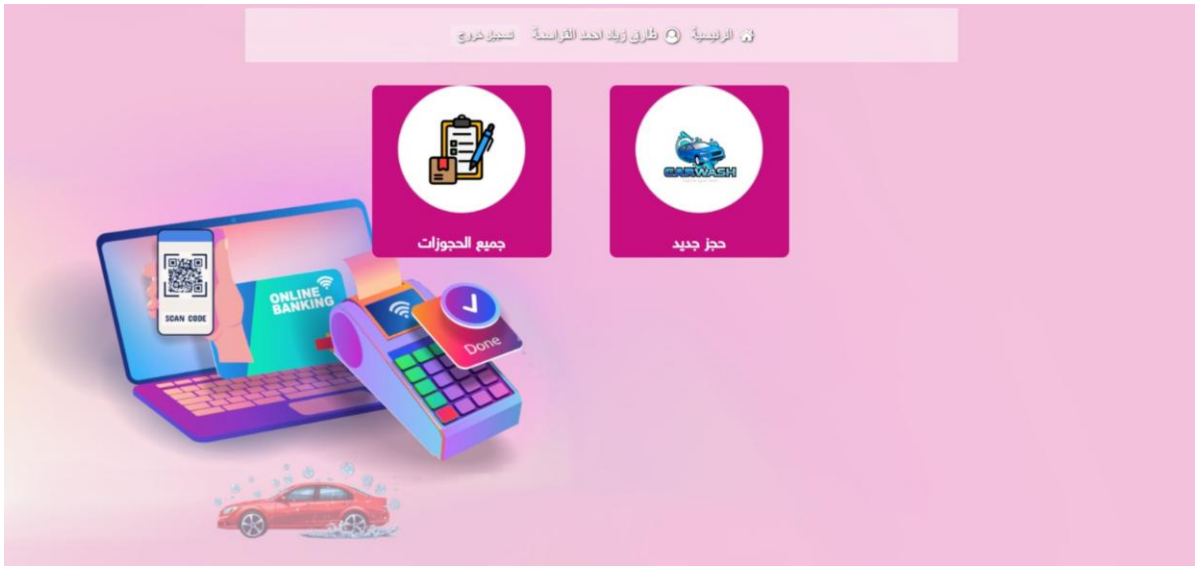

Figure5. 3 Customer Account

4) The following figure 5.4 shows the process of sending a reservation by the customer, specifying the time and date.

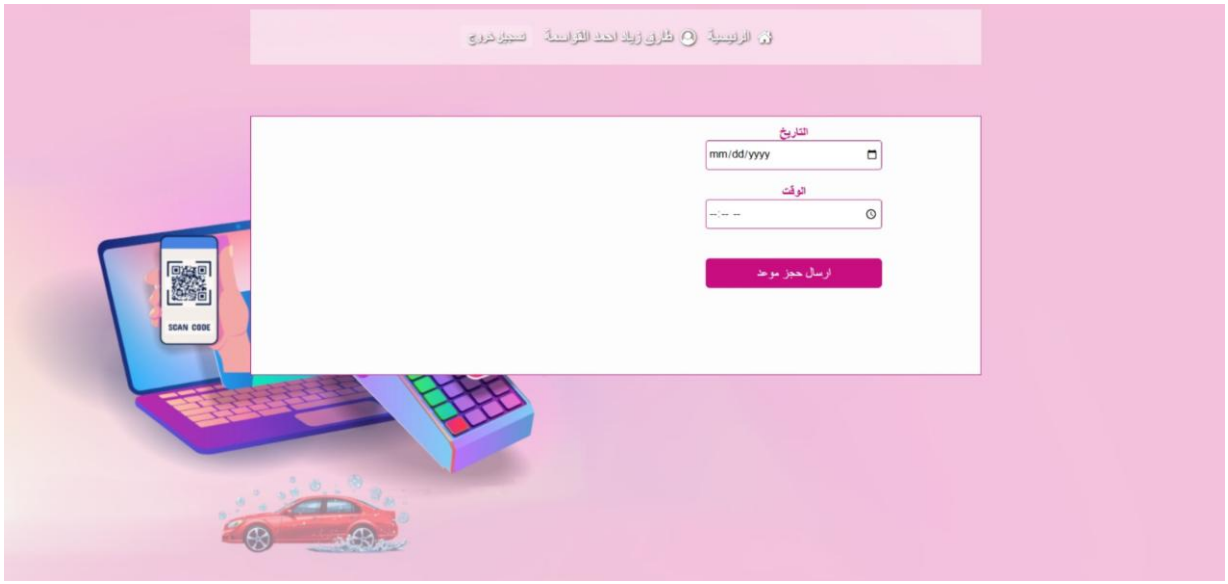

Figure5. 4 Sending A Reservation

5) Ultrasonic sensors were installed at various points, one at the entrance and the other at the car wash area.

5.2 Justifications of the obtained results

The results obtained through the experiments are consistent with the project's objectives and requirements for a smart car wash system as shows in figure 5.5. Several justifications were obtained for the results, including:

1. A model of the project was designed using a bus bar mounted with electronic components and pumps, and the bar's operation was verified to be correct and accurate.
2. After testing a number of pumps two types were developed suitable for water and soap based on the specified pumping quantities for each.
3. A website was designed to manage customer accounts and reservations allowing the manager to select a suitable time and date for each customer.
4. Through study and analysis sensors were installed on the model based on the performance of each sensor.
5. Based on multiple studies this system was developed in a meticulous and effective manner to achieve all the desired objectives.

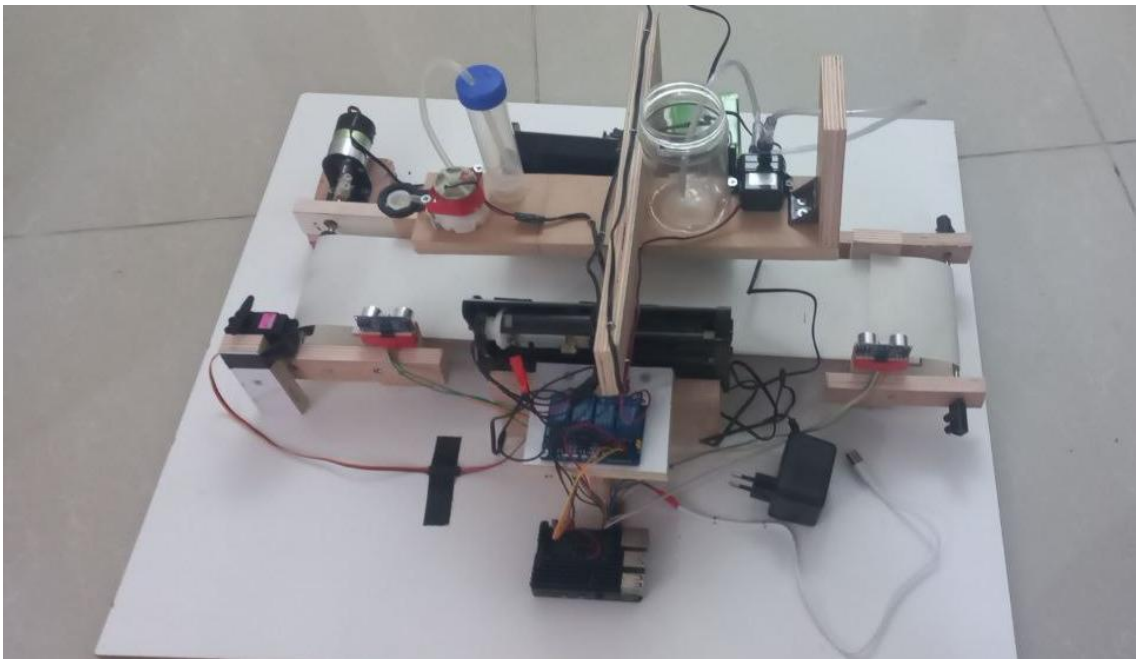

Figure5. 5 smart car wash system

5.3 Summary

This chapter presents the project results, demonstrating the system's ability to efficiently manage the car washing process through accurate vehicle detection, automated control of pumps and conveyor movement, and smooth user interaction via the online booking platform. The integration of mechanical systems, sensors, and web technologies has proven effective in building a reliable, user-friendly, and scalable smart car wash solution.

CHAPTER SIX: CONCLUSION AND FUTURE WORK

6.1 Conclusion

In conclusion our smart car wash project successfully achieved its primary objectives by designing and developing a pre-booked car wash prototype that automated the car wash process enhanced service efficiency and improved the overall customer experience. By using a conveyor belt and ultrasonic sensors and high-performance pumps connected to a Raspberry Pi microcontroller and creating an online reservation website the system demonstrated high reliability reduced resource consumption and provided a smooth and organized workflow.

The system addressed many of the problems faced by car wash operators, as traditional systems lack pre-booking systems and customer time and appointment management. Our project addressed all of these problems and challenges by developing a prototype smart car wash system.

In addition to an easy-to-use website for both educated and uneducated people a smooth and easy interactive interface has been designed to enable customers to book a slot in the laundry by specifying a suitable time and date.

Despite the minor challenges we faced in ensuring optimal sensor performance in all environmental conditions the project demonstrated the feasibility and effectiveness of building a fully automated smart car wash service. The system provides a scalable foundation for future improvements aimed at increasing automation, sustainability and user convenience.

6.2 Future work

Future work on the smart car wash system focuses on several key areas to develop the project and increase its efficiency and effectiveness including:

1) Working to deploy the system on the largest possible scale enabling it to be adopted in all car washes.
2) Adding more accurate and effective sensors to enhance the system's capabilities for example use IR sensor .
3) Adding a notification feature to inform customers of upcoming car washes.
4) Developing a mobile phone application for pre-booking.
5) Developing the system to include artificial intelligence, such as automatic vehicle recognition (truck or regular vehicle) for yolo library .
6) Adding a secure online payment system to enable customers to pay during the reservation process.

References

[1] Automated Car Wash with Parking System (2023). Retrieved November 2024, from https://repository.najah.edu/collections/15d6ab60-d6dd-4333-bb0a-2db1a16d5a77

[2] Automatic Car Washing Machine (2019). Retrieved November 2024, from https://scholar.ppu.edu/handle/123456789/7165

[3] Raspberry Pi Camera Module Datasheet. Retrieved December 2024, from https://www.raspberrypi.org/documentation/accessories/camera.html

[4] HC-SR04 Ultrasonic Sensor Datasheet. Retrieved November 2024, from https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf

[5] Logitech C920 Webcam Specifications. Retrieved December 2024, from https://www.logitech.com/en-us/products/webcams/c920-hd-pro-webcam.960-001335.html

[6] ESP32 Camera (OV2640) Module Datasheet. Retrieved December 2024, from https://www.espressif.com/sites/default/files/documentation/esp32-camera-specification.pdf

[7] Relay Module Datasheet: 1-Channel 5V Relay. Retrieved December 2024, from https://components101.com/sites/default/files/component_datasheet/Relay-Module-Datasheet.pdf

[8] DC Water Pump Datasheet. Retrieved December 2024, from https://5.imimg.com/data5/SELLER/Doc/2021/4/RH/JK/XD/1833510/r385-dc-9-12v-water-pump-maximum-lift-3-meters.pdf

[9] DC Motor Datasheet. Retrieved December 2024, from https://infodatasheet.com/jgy-370-dc-motor-datasheet.html

[10] Gate Actuator Datasheet. Retrieved December 2024, from https://www.asahi-america.com/wp-content/uploads/2024/07/Series_19_MAV_Modulating_Gate_Valve_Datasheet pdf.

[11] Raspbian Operating System. Retrieved December 2024, from

https://www.raspberrypi.org/software/operating-systems/

[12] Thonny IDE. Retrieved December 2024, from

https://thonny.org

[13] XAMPP – Apache Friends. Retrieved December 2024, from

https://www.apachefriends.org/index.html

Appendix A

Raspberry code for the system (python code):

```
'''
sudo apt-get install libzbar0
pip3 install pyzbar pillow --break-system-packages
'''
from pyzbar.pyzbar import decode
from PIL import Image
import requests
import sys
import time

try:
  import RPi.GPIO as GPIO
except:
  import RPI_virtual.GPIO as GPIO

try:
  import picamera
except:
  import RPI_virtual.picamera as picamera
  #from RPI_virtual import picamera

# تعريف منافذ GPIO
servo_pin = 18
relay_brush = 17
relay_pump1 = 27
relay_pump2 = 22
relay_conveyor = 23
ultrasonic_ultrasonic1_trig = 16
ultrasonic_ultrasonic1_echo = 20
ultrasonic_ultrasonic2_trig = 19
ultrasonic_ultrasonic2_echo = 26

# إعداد منافذ GPIO
GPIO.setmode(GPIO.BCM)

GPIO.setup(servo_pin, GPIO.OUT)

GPIO.setup(relay_brush, GPIO.OUT)
GPIO.setup(relay_pump1, GPIO.OUT)
GPIO.setup(relay_pump2, GPIO.OUT)
GPIO.setup(relay_conveyor, GPIO.OUT)

GPIO.setup(ultrasonic_ultrasonic1_trig, GPIO.OUT)
GPIO.setup(ultrasonic_ultrasonic1_echo, GPIO.IN)
```

```
GPIO.setup(ultrasonic_ultrasonic2_trig, GPIO.OUT)
GPIO.setup(ultrasonic_ultrasonic2_echo, GPIO.IN)

servo = GPIO.PWM(servo_pin, 50)  # 50Hz PWM


GPIO.output(relay_brush,1)#off
GPIO.output(relay_pump1,1)#off
GPIO.output(relay_pump2,1)#off
GPIO.output(relay_conveyor,1)#off


# دوال مساعدة
def move_servo(angle):
    duty = angle / 18 + 2
    servo.start(0)
    servo.ChangeDutyCycle(duty)
    time.sleep(3)
    servo.ChangeDutyCycle(0)

move_servo(90)

def read_qr_code():
    cam_index=0 # on PRI_real use 0 or 1 just
    with picamera.PiCamera(camera_num=cam_index) as camera:
         camera.resolution = (640, 480)
         #camera.start_preview()
         #time.sleep(2)
         camera.capture('qrcode.jpg')
         #camera.stop_preview()

    image = Image.open('qrcode.jpg')
    decoded_objects = decode(image)

    if decoded_objects:
        return decoded_objects[0].data.decode('utf-8')
    else:
        return None

def verify_qr_code(qr_code_value):
    url = 'http://192.168.214.48/2025/student/project2025-
1/university/ppu/car-wash/web/CAR-WASH-web/api.php'

    data = {'qr_code': qr_code_value}
    try:
        response = requests.post(url, data=data)
        print("response.text="+str(response.text))
        if response.text.strip() == 'valid':
            return True
```

```
        else:
            return False
    except requests.exceptions.RequestException as e:
        print(f"Error connecting to server: {e}")
        return False

def get_distance(trig, echo):
    GPIO.output(trig, True)
    time.sleep(0.00001)
    GPIO.output(trig, False)
    ultrasonic1_time = time.time()
    stop_time = time.time()
    while GPIO.input(echo) == 0:
        ultrasonic1_time = time.time()
    while GPIO.input(echo) == 1:
        stop_time = time.time()
    time_elapsed = stop_time - ultrasonic1_time
    distance = (time_elapsed * 34300) / 2
    return distance


try:
    while True:
        try:
            qr_code_value = read_qr_code()
            print("qr_code_value:"+str(qr_code_value))
            if qr_code_value:
                if verify_qr_code(qr_code_value):
                    #============================1
                    print("QR code valid. Opening servo.")
                    move_servo(0)
                    GPIO.output(relay_conveyor,0)#on

                    ultrasonic1_distance = 999
                    while ultrasonic1_distance>10:
                        ultrasonic1_distance =
get_distance(ultrasonic_ultrasonic1_trig, ultrasonic_ultrasonic1_echo)
                        time.sleep(0.5)
                        print("ultrasonic1_distance:"+str(ultrasonic1_distan
ce))


                    GPIO.output(relay_conveyor,1)#off
                    move_servo(90)
                    #============================

                    #============================2
                    print("sabon start.")
                    GPIO.output(relay_pump1, 0)#on
```

```
                    time.sleep(5)
                    GPIO.output(relay_pump1, 1)#off
                    print("sabon complete.")
                    #============================

                    #============================3
                    GPIO.output(relay_brush, 0)#on
                    GPIO.output(relay_conveyor,0)#on

                    ultrasonic2_distance=999
                    while ultrasonic2_distance>10:
                        ultrasonic2_distance =
get_distance(ultrasonic_ultrasonic2_trig, ultrasonic_ultrasonic2_echo)
                        time.sleep(0.5)
                        print("ultrasonic2_distance:"+str(ultrasonic2_distan
ce))

                    GPIO.output(relay_brush, 1)#off

                    GPIO.output(relay_conveyor,1)#off
                    #============================

                    #============================4
                    print("Washing start.")
                    GPIO.output(relay_pump2, 0)#on
                    time.sleep(5)
                    GPIO.output(relay_pump2, 1)#off
                    print("Washing complete.")
                    #============================

                    #============================5
                    GPIO.output(relay_conveyor,0)#on
                    time.sleep(2)
                    GPIO.output(relay_conveyor,1)#off
                    #============================

                else:
                    print("QR code invalid or this booking out of datetime
or Finish.")
            else:
                print("No QR code detected.")
        except KeyboardInterrupt:
            print("\nProgram interrupted by user.")
            break  #  stop loop

        time.sleep(2)
except Exception as e:
    print(f"Unexpected error: {e}")
```

```
finally:
    GPIO.cleanup()
    print("GPIO cleaned up. Program terminated.")
    sys.exit(0)
```

Appendix B

Code QR Screen (PHP Code):

```
<!DOCTYPE html>
<html lang="ar">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>QR Code Generator</title>
    <style>
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            margin: 0;
            background-color: #f4f4f4;
        }
        .qr-container {
            text-align: center;
            padding: 20px;
            background: white;
            box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);
            border-radius: 10px;
        }
    </style>
</head>
<body>

<div class="qr-container">
    <?php
    if (isset($_GET['data'])) {

        $data = urlencode($_GET['data']);
        $svg = file_get_contents("https://api.qrserver.com/v1/create-
qr-code/?data={$data}&size=200x200&format=svg");
        echo $svg;
```

```
    } else {
        echo "<p>لم يتم توفير بيانات لإنشاء QR Code</p>";
    }
    ?>
</div>

</body>
</html>
```

Appendix C

Booking Code (PHP Code):

```
<?php
include "chech_auth.php";

$return_data=array();

$customer_id=intval($_SESSION['user_id']);


if (isset($_REQUEST['action']) && $_REQUEST['action'] ==
'save_booking')
{

  $wash_date = mysqli_real_escape_string($connect,$_POST['wash_date']);
  $wash_time = mysqli_real_escape_string($connect,$_POST['wash_time']);

  //============================
  $wash_date = $_POST['wash_date']; // تاريخ الحجز بصيغة 'Y-m-d'
  $wash_time = $_POST['wash_time']; // الوقت المحجوز بصيغة 'H:i:s'

  $sql = "SELECT COUNT(*) AS total_bookings
          FROM `customer_booking`
          WHERE TIMESTAMP(wash_date, wash_time)
                BETWEEN TIMESTAMP('$wash_date', '$wash_time') -
INTERVAL 15 MINUTE
                    AND TIMESTAMP('$wash_date', '$wash_time') +
INTERVAL 15 MINUTE";

  $result = mysqli_query($connect, $sql);
  $row = mysqli_fetch_assoc($result);
```

```php
  $this_time_wash_count = $row['total_bookings'];



  if ($this_time_wash_count >= 3)
  {
      $return_data["status"] = "error";
      $return_data["note"] = "اختر وقت اخر لأن الحد الأقصى من الحجوزات تم الوصول إليه.";
      echo json_encode($return_data);
      exit();
  }
  //============================

  $serial_QR=bin2hex(random_bytes(20));

  $sql="INSERT INTO customer_booking
  (wash_date,wash_time, customer_id,serial_QR)
  VALUES
  ('$wash_date', '$wash_time',$customer_id,'$serial_QR')";


  $insert = mysqli_query($connect, $sql);

  if($insert)
  {
    $return_data["status"]="ok";
    $return_data["note"]="تم الحجز بنجاح";
    echo json_encode($return_data);
    exit();

  }



}
?>
```

Appendix D

Code to receive data from Raspberry (PHP Code):

```php
<?php

include "config.php";

date_default_timezone_set('Asia/Jerusalem');

$qr_code_value = $connect->real_escape_string($_POST['qr_code']);

$current_date = date('Y-m-d');
$current_time = date('H:i:s');

// SQL: تحقق أن التاريخ هو تاريخ اليوم وأن الوقت الحالي بين وقت الحجز ووقت الحجز + ساعة
$sql = "SELECT * FROM customer_booking
        WHERE serial_QR = '$qr_code_value'
        AND status = 1
        AND wash_date = '$current_date'
        AND TIME('$current_time') BETWEEN wash_time AND
ADDTIME(wash_time, '01:00:00')";

// سجل الاستعلام في ملف
file_put_contents("output.debug", "$sql\n", FILE_APPEND);

$result = $connect->query($sql);

if ($result && $result->num_rows > 0) {
    echo 'valid';

    // تحديث الحالة إلى "تم الاستخدام"
    $update_sql = "UPDATE customer_booking SET status = 2 WHERE
serial_QR = '$qr_code_value'";
    $connect->query($update_sql);

} else {
    echo 'invalid';
}

?>
```

نموذج إقرار وتوثيق استخدام أدوات الذكاء الاصطناعي في مشروع التخرج

اسم الطالب / أسماء الطلبة: حسام سلامه / طارق القواسمة	عنوان المشروع: Smart Car Wash System
	اسم المشرف الأكاديمي: د. عليان أبو غربية
الكلية / الدائرة: كلية تكنولوجيا المعلومات وهندسة الحاسوب	السنة/الفصل الدراسي: 2026 / 2025

الغرض من الاستخدام	اسم الأداة	أماكن الاستخدام في التقرير (فصول / صفحات)	الأمر الأساسي المستخدم (Prompt)	هل تم تعديل الناتج؟	نسبة الاستخدام التقديرية	ملاحظات
توليد نصوص	☐ ChatGPT	Chapter 1 / 1.3 Project Aims and Objectives Chapter 2 / 2.2 Theoretical Background Concepts Chapter 3 / 3.2 System Components and Design Alternatives	Summarize the concepts and design process of the smart car wash system	نعم	15%	- تم استخدامه لصياغة الأهداف والتمهيد للمشروع - المساعدة في كتابة الشرح وتحليل النظريات والأدوات المستخدمة - توضيح اختيار المكونات للمشروع
تدقيق لغوي	☐ Grammarly	كامل التقرير	—	نعم	10%	تم استخدامه في تحسين القواعد اللغوية في التقرير
	☐ Quillbot			نعم	10%	ساعد في اغادة صياغة الجمل وتحسين النص

- تلخيص عملية التصميم واختيار مكونات المشروع - تلخيص النتائج التي تم الحصول عليها من الاختبارات	15%	نعم	Summarize the key differences between components and results obtained from testing	Chapter 3 / 3.2.1 Hardware Components Chapter 5 / 5.2 Justifications of the obtained results	☐ ChatGPT	تلخيص محتوى
استخدم لتحليل النتائج ومناقشة البيانات	10%	نعم	Summarize test results and analyze user data	Chapter 5 / 5.1 Detailed analysis of the results/experiments	☐ Excel Copilot	تحليل بيانات
المساعدة في انشاء صورة تمثيلية لمشروع المغسلة الالية لغسيل السيارات	5%	نعم	Edit an image repressing car wash process automation	Chapter 1 / Figure 1.1 Chapter 3 / Figure 3.8	☐ DALL·E	رسم مخططات / أشكال
برمجة الموقع برمجة الراسبيري	20%	نعم	Generate login, registration and main page Generate Python code for wash process	Chapter 4 / 4.4 Software Implementation	☐ ChatGPT	كتابة أكواد برمجية
المساعدة في تلخيص وتنظيم المراجع المستخدمة داخل التقرير	5%		Help generate citations for references on car wash automation	Chapter 2 / 2.2 Chapter 3 / 3.2.2	☐ ChatGPT	توثيق مراجع

إقرار فريق مشروع التخرج:

نقر بأن استخدام أدوات الذكاء الاصطناعي تم بشكل مسؤول وبما يتوافق مع السياسات الجامعية، وقد راجعنا وحررنا كافة المخرجات بما يعكس فهمنا الشخصي.

Group team: Husam Salameh Signature: …….

Tariq Qawasmeh Signature: …….

Date: 22/6/2025