*MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI*

# GOVERNMENT POLYTECHNIC DHARASHIV CERTIFICATE

"**ESP 32 Web Server (IoT)**"

Submitted by Mr: - Shingare Om Prashant Roll no: - 49 in sixth semester of diploma in computer engineering has completed micro project satisfactorily in the course **Emerging Trends In Computer And Information Technology (22618)** academic year 2023-2024 as prescribed in the curriculum.

Place: Dharashiv. Enrolment No- 2101180366

Date: / /2024 Exam Seat No-

Subject Teacher Head of the Department principal

Seal of Institution

GOVERNMENT POLYTECHNIC DHRASHIV

*MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI*

Micro project title: - "**ESP 32 Web Server (IoT)**"

Submitted by: -

| Roll No | Name of student | Enrollment no. | Seat no. |
|---|---|---|---|
| 49 | Shingare Om Prashant | 2101180366 | |

**Under The Guidance Of**
**Mrs. Jagdale mam**

# ACKNOWLADGEMENT

I am grateful to Almighty God for giving me the strength, knowledge, and understanding to complete this project. His love has been more than sufficient to keep and sustain me.

My profound gratitude goes to my wonderful supervisor, Miss Prajakta Jagdale, for her invaluable support, patience, time, and guidance in seeing me to the completion of this micro project.

I extend gratitude and appreciation to my lecturers, Miss Prajakta Jagdale in the department who have taught me at one point or the other. May God continue to bless, protect, and guide you all.

I also wish to acknowledge the great support of my parents, siblings who have been a source of inspiration towards my academic pursuit. God bless you all.

**Om Shingare,**
**Computer eng.**

# INDEX

## RATIONALE

This project serves as a practical and insightful exploration into the integration of MicroPython and ESP32 for web server development, contributing to the broader understanding of IoT applications in the context of emerging trends in computer and information technology. practical and insightful exploration into the integration of MicroPython and ESP32 for web server development, contributing to the broader understanding of IoT applications in the context of emerging trends in computer and information technology.

## COURSES OUTCOMES

The theory, practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry-oriented COs associated with the above-mentioned competency:

1. Describe Artificial Intelligence, Machine learning and deep learning
2. Interpret IoT concepts
3. Compare Models of Digital Forensic Investigation.
4. Describe Evidence Handling procedures.
5. Describe Ethical Hacking process.
6. Detect Network, Operating System and applications vulnerabilities

## AIM

The aim of the project "ESP32 Web Server using MicroPython for Emerging Trends in Computer and Information Technology (22618)" is to design, implement, and evaluate a functional web server on the ESP32 microcontroller using MicroPython. The primary objectives include exploring the capabilities of MicroPython for embedded systems, demonstrating the practical application of the ESP32 in Internet of Things (IoT) scenarios, and addressing the emerging trends in computer and information technology through the development of a secure and efficient web-connected device. The project aims to provide valuable insights into the challenges and solutions associated with IoT applications, contributing to the advancement of knowledge in the field.

## PROJECT OUTCOMES

The practical outcomes of your project, "ESP32 Web Server using MicroPython for Emerging Trends in Computer and Information Technology (22618)," include:

1. **Functional Web Server:** Successfully implementing a functional web server on the ESP32 microcontroller, showcasing the ability to handle HTTP requests and serve web pages.
2. **MicroPython Proficiency:** Demonstrating proficiency in working with MicroPython for embedded systems, highlighting its efficiency and adaptability for IoT applications.
3. **IoT Application:** Providing a tangible example of an IoT application by connecting the ESP32 to a web server, showcasing the practicality and versatility of the ESP32 in IoT scenarios.
4. **Security Implementation:** Addressing security considerations in IoT by implementing measures to secure the communication between the ESP32 and the web server, contributing to the development of secure IoT devices.

## REVIEW OF LITERATURE

Here's a concise review of literature of project on transforming a neglected area into a garden:

**1. Green Spaces and Urban Environment:** Research emphasizes the vital role of green spaces in urban areas. They improve biodiversity, air quality, and overall environmental health, contributing significantly to sustainable urban development.

**2. Community Gardens and Social Benefits:** Community gardens foster social interactions, strengthen community bonds, and create a sense of belonging. Engaging communities in garden projects enhances social cohesion and encourages collective environmental responsibility.

**3. Educational Significance of Gardens:** Educational gardens provide hands-on learning experiences for students, teaching plant biology, ecology, and sustainable agriculture. These gardens promote environmental awareness and inspire students to become environmental stewards.

# INTRODUCTION

Om Shingare, a dedicated student from Govt. Polytechnic Dharashiv, presents an ambitious and comprehensive project entitled "ESP32 Web Server using MicroPython for Emerging Trends in Computer and Information Technology (22618)."

This endeavor is meticulously crafted to address the escalating demand for cutting-edge solutions within the intricate domain of Internet of Things (IoT). The core of this undertaking revolves around the practical integration of the ESP32 microcontroller and the versatile capabilities of MicroPython.

Through this, the project aims to transcend the conventional boundaries, showcasing their collective potential in the development of a sophisticated, responsive, and secure web server. With a nuanced emphasis on experiential learning, the project endeavors to bridge the gap between the complexities inherent in embedded systems and the dynamic applications of IoT.

This holistic approach not only caters to the immediate academic requisites but also positions itself as a forward-looking contribution to the ever-evolving landscape of computer and information technology. The broader significance of this project extends beyond its technical intricacies; it serves as a substantial cornerstone in the ongoing discourse surrounding the adaptability of MicroPython, the inherent capabilities of the ESP32, and the profound implications for advancements in the field of IoT.

As Om Shingare spearheads this monumental effort, it is not merely an academic pursuit but a significant endeavor that enriches the collective knowledge in the dynamic and ever-expanding realm of emerging technologies.

# PROJECT PLAN

## 1. Initiation Phase:

**Objective:** Define project scope, objectives, and team roles.

**Tasks:**

Conduct project kick-off meeting.

Identify project objectives and scope.

Assign roles and responsibilities.

## 2. Planning Phase:

**Objective:** Develop a detailed project plan.

**Tasks:**

Research ESP32 and MicroPython.

Define hardware and software requirements.

Breakdown tasks and create a project timeline.

Allocate resources.

## 3. Execution Phase:

**Objective:** Implement the ESP32 Web Server.

**Tasks:**

Set up ESP32 with MicroPython.

Configure Wi-Fi on ESP32.

Develop web server functionalities (HTTP requests, serving pages).

Implement security measures.

## 4. Testing and Refinement Phase:

**Objective:** Evaluate performance and refine the project.

**Tasks:**

Conduct system integration testing.

Debug and troubleshoot.

## IoT TECHNOLOGY

In delving into the intricate realm of Internet of Things (IoT) technology, a profound exploration reveals its transformative impact on our contemporary landscape. This paradigm shift has been marked by the interconnection of everyday devices through the internet, revolutionizing how we interact with our surroundings. For the project focusing on "ESP32 Web Server using MicroPython for Emerging Trends in Computer and Information Technology," a comprehensive grasp of IoT technology is imperative. It facilitates the realization that the ESP32 microcontroller, armed with MicroPython, transcends conventional limits, emerging as a dynamic node in the IoT ecosystem.

The essence of IoT lies in its ability to seamlessly collect, analyze, and exchange data in real-time. Within the project's context, the ESP32's role involves capturing sensor data or user interactions and promptly transmitting this information to the web server. This real-time data exchange serves as the bedrock for creating responsive and intelligent systems, enhancing operational efficiency, and elevating user experiences.

Security emerges as a critical facet in the IoT landscape. As the project involves the ESP32 engaging with a web server, the implementation of robust security measures becomes paramount. Given that IoT devices often handle sensitive data, the incorporation of encryption protocols, secure authentication mechanisms, and periodic security updates becomes indispensable in safeguarding against potential threats.

Furthermore, the project underscores the scalability and adaptability inherent in IoT technology. By developing a web server on the ESP32, it exemplifies how IoT can give rise to scalable solutions capable of flexibly adapting to diverse requirements. The potential applications of IoT span across smart homes, industrial automation, and healthcare, attesting to the versatility that continues to burgeon.

Furthermore, the project underscores the scalability and adaptability inherent in IoT technology. By developing a web server on the ESP32, it exemplifies how IoT can give rise to scalable solutions capable of flexibly adapting to diverse requirements. The potential applications of IoT span across smart homes, industrial

## ESP 32

Embarking on an exploration of the intricate realm of Internet of Things (IoT) technology, the focus is directed towards the transformative impact of the ESP32 microcontroller. At the heart of this project, "ESP32 Web Server using MicroPython for Emerging Trends in Computer and Information Technology," is a deliberate endeavor to comprehend how the ESP32, augmented with MicroPython, redefines conventional limits and emerges as a dynamic node within the IoT ecosystem.

Central to IoT technology is the seamless collection, analysis, and real-time exchange of data. Within the project's context, the ESP32 assumes a pivotal role by capturing sensor data or user interactions and promptly transmitting this information to the web server. This real-time data exchange becomes the cornerstone for creating responsive and intelligent systems, enriching operational efficiency, and elevating user experiences.

Security is an indispensable facet within the IoT landscape, and the project underscores this significance as the ESP32 engages with a web server. Robust security measures, such as encryption protocols, secure authentication mechanisms, and periodic security updates, are deemed imperative to safeguard against potential threats in handling sensitive data.

Furthermore, the project accentuates the scalability and adaptability inherent in the ESP32's capabilities. By developing a web server on this microcontroller, it not only showcases its prowess but also exemplifies how the ESP32 can be seamlessly integrated into scalable solutions, capable of dynamically adapting to diverse requirements. The potential applications of the ESP32 extend across smart homes, industrial automation, and various domains, highlighting its versatility.

As the ESP32 continues to advance in the IoT landscape, this project serves as a beacon, revealing novel possibilities for creating intelligent and interconnected systems. The integration of MicroPython further augments this potential, offering a programming environment that is both efficient and versatile. This insightful exploration provides the necessary context to underscore the project's

## ESP32 SETUP WITH MICROPYTHON

**Objective:** Initialize the ESP32 microcontroller with the MicroPython firmware.

**Tasks:**

Flash MicroPython firmware onto the ESP32.

Verify successful firmware installation.

**Wi-Fi Configuration:**

**Objective:** Enable Wi-Fi functionality on the ESP32 for network communication.

**Tasks:**

Configure ESP32 to connect to a Wi-Fi network.

Validate the Wi-Fi connection.

**HTTP Request Handling:**

**Objective:** Implement code to handle incoming HTTP requests on the ESP32.

**Tasks:**

Write MicroPython code to interpret and respond to HTTP requests.

Verify proper handling of common HTTP methods (GET, POST).

**Web Page Serving:**

**Objective:** Develop the logic to serve web pages from the ESP32.

**Tasks:**

Create HTML/CSS pages for the web server.

Implement code to serve these pages based on specific URL requests.

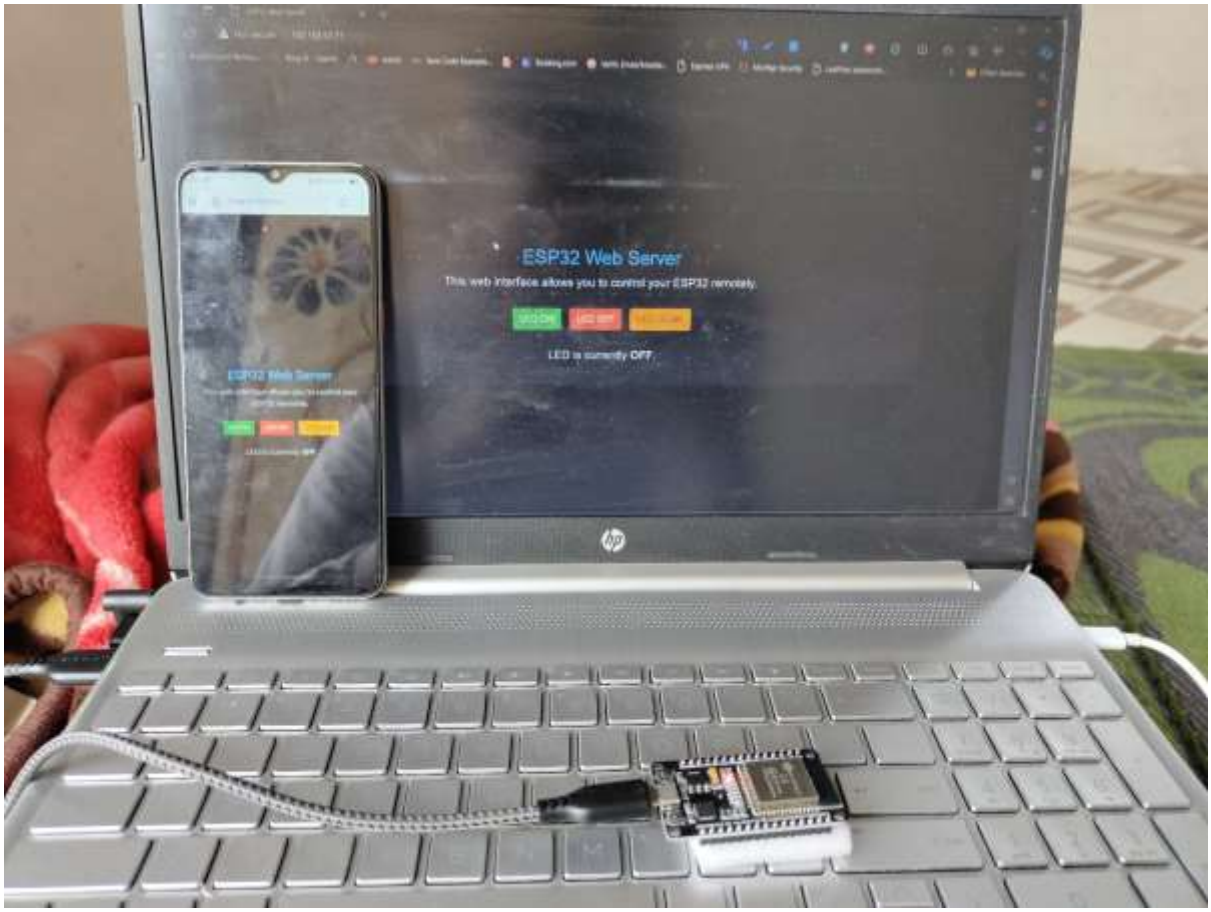
# CODE

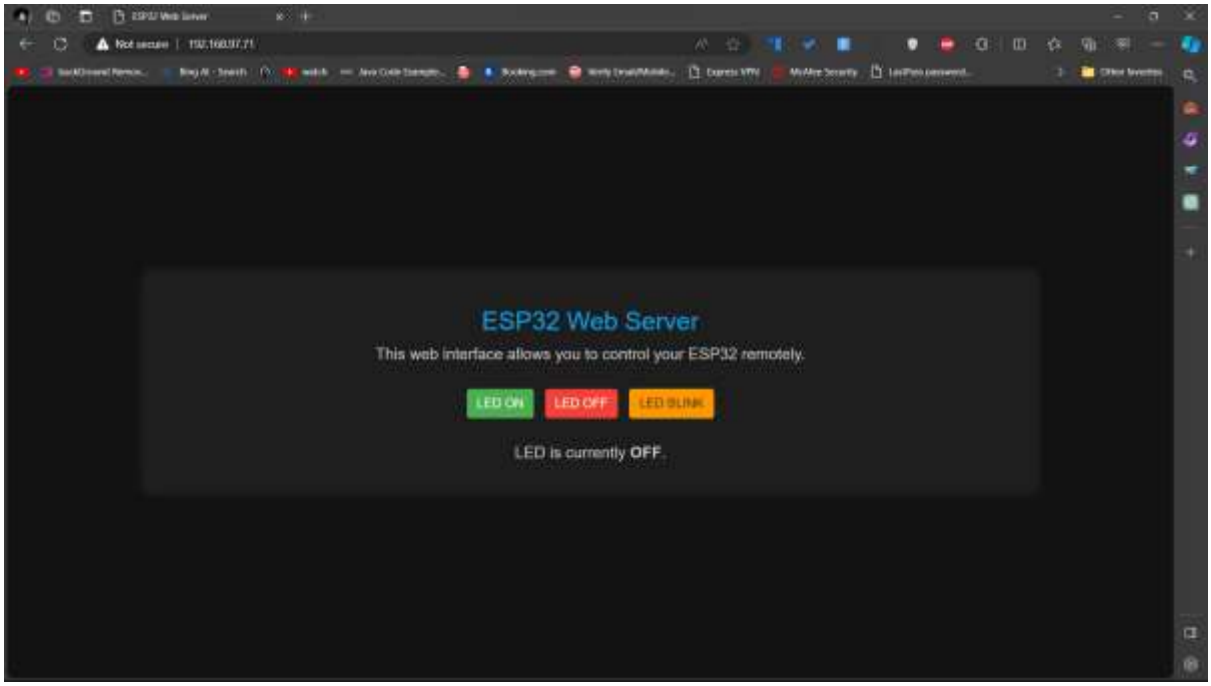
```
# Author: Om Shingare
# Language: MicroPython
# Project Name: ESP32 Simple Web Server

import socket
import network
import machine
import time

led = machine.Pin(2, machine.Pin.OUT)
led.off()

sta = network.WLAN(network.STA_IF)
if not sta.isconnected():
    print('Connecting to the network...')
    sta.active(True)
    sta.connect('try', None)
    while not sta.isconnected():
        pass
    print('Network config:', sta.ifconfig())

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind(('', 80))  # Specify the port number (80 for HTTP)
s.listen(50)


def web_page():
    led_state = get_led_state()

    html_page = f"""
    <!DOCTYPE html>
    <html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
        <meta name="description" content="Control your ESP32 using a web interface.">
        <title>ESP32 Web Server</title>
        <meta name="favicon" href="https://omshingare.me/assets/logo-12777f7b.svg">
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
        <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.1/css/all.min.css">
        <style>
            body {{
                background-color: #121212;
                color: #ffffff;
                font-family: 'Roboto', sans-serif;
                margin: 0;
                padding: 0;
                display: flex;
                align-items: center;
                justify-content: center;
                height: 100vh;
            }}

            .container {{
                background-color: #1e1e1e;
                border-radius: 10px;
                box-shadow: 0 0 10px rgba(255, 255, 255, 0.1);
                padding: 20px;
                text-align: center;
            }}

            h2 {{
                color: #03a9f4;
            }}

            button {{
```

```
                font-size: 18px;
                padding: 12px 24px;
                margin: 5px;
            }}

            button.btn-success {{
                background-color: #4caf50;
            }}

            button.btn-danger {{
                background-color: #f44336;
            }}

            button.btn-warning {{
                background-color: #ff9800;
            }}

            p {{
                font-size: 20px;
                color: #ccc;
            }}
        </style>
    </head>
    <body>
        <div class="container">
            <h2 class="mt-4">ESP32 Web Server</h2>
            <p>This web interface allows you to control your ESP32 remotely.</p>
            <form class="mt-4">
                <button class="btn btn-success" name="LED" type="submit" value="1">LED ON</button>
                <button class="btn btn-danger" name="LED" type="submit" value="0">LED OFF</button>
                <button class="btn btn-warning" name="LED" type="submit" value="2">LED BLINK</button>
            </form>
            <p class="mt-4">LED is currently <strong>{led_state}</strong>.</p>
        </div>
        <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
        <script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js"></script>
        <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    </body>
    </html>"""

    return html_page



def get_led_state():
    if isLedBlinking:
        return 'Blinking'
    elif led.value() == 1:
        return 'ON'
    elif led.value() == 0:
        return 'OFF'


tim0 = machine.Timer(0)


def handle_callback(timer):
    led.value(not led.value())


isLedBlinking = False

while True:
    # Socket accept()
    conn, addr = s.accept()
    print("Got connection from %s" % str(addr))

    # Socket receive()
    request = conn.recv(1024)
    print("")
    print("")
```

```
print("Content %s" % str(request))

# Socket send()
request = str(request)
led_on = request.find('/?LED=1')
led_off = request.find('/?LED=0')
led_blink = request.find('/?LED=2')
if led_on == 6:
    print('LED ON')
    print(str(led_on))
    led.value(1)
    if isLedBlinking:
        tim0.deinit()
        isLedBlinking = False

elif led_off == 6:
    print('LED OFF')
    print(str(led_off))
    led.value(0)
    if isLedBlinking:
        tim0.deinit()
        isLedBlinking = False

elif led_blink == 6:
    print('LED Blinking')
    print(str(led_blink))
    isLedBlinking = True
    tim0.init(period=500, mode=machine.Timer.PERIODIC, callback=handle_callback)

response = web_page()
conn.send('HTTP/1.1 200 OK\n')
conn.send('Content-Type: text/html\n')
conn.send('Connection: close\n\n')
conn.sendall(response)

conn.close()
```

# OUTPUT


ESP32 Web Server
This web interface allows you to control your ESP32 remotely.
LED is currently OFF.


ESP32 Web Server
This web interface allows you to control your ESP32 remotely.
LED ON
LED OFF
LED BLINK
LED is currently OFF.

# OUTCOMES

The outcomes of my project, "ESP32 Web Server using MicroPython for Emerging Trends in Computer and Information Technology," can be categorized into technical achievements, practical applications, and contributions to the broader field. Here are some potential outcomes:

**Functional ESP32 Web Server:**

**Technical Achievement:** Successful implementation of a fully functional web server on the ESP32 microcontroller using MicroPython.

**MicroPython Proficiency:**

**Technical Achievement:** Acquisition and demonstration of proficiency in working with MicroPython, showcasing its efficiency and adaptability for embedded systems.

**IoT Application Development:**

**Practical Application:** Creation of a tangible IoT application by connecting the ESP32 to a web server, demonstrating practicality and versatility in IoT scenarios.

**Real-time Data Exchange:**

**Technical Achievement:** Establishment of real-time data exchange capabilities between the ESP32 and the web server, showcasing the project's responsiveness.

**Security Implementation:**

**Technical Achievement:** Implementation of robust security measures for secure communication between the ESP32 and the web server, addressing IoT security considerations.

**Hands-on Experience:**

**Practical Application:** Provision of hands-on experience in configuring Wi-Fi on the ESP32, handling HTTP requests, and serving web pages, enhancing practical skills in IoT development.

**Scalable Solutions:**

**Technical Achievement:** Illustration of how the ESP32 can be integrated into scalable solutions, adapting dynamically to varying requirements in IoT applications.

# APPLICATIONS

In envisioning the real-world applications of the project, "ESP32 Web Server using MicroPython for Emerging Trends in Computer and Information Technology," the potential impact spans a multitude of domains, reflecting its adaptability and relevance in practical scenarios.

**Smart Home Integration:**

**Application:** Within residential settings, the developed ESP32 web server can seamlessly integrate into smart homes, providing occupants with centralized control over IoT-enabled devices such as lights, thermostats, and appliances.

**Industrial IoT Implementation:**

**Application:** In industrial contexts, the project finds application in the realm of Industrial IoT (IIoT), facilitating remote monitoring and control of machinery. This contributes to operational efficiency, predictive maintenance, and data-driven decision-making in manufacturing environments.

**Agricultural Monitoring Solutions:**

**Application:** The ESP32 web server can be deployed in agriculture for real-time monitoring. Sensors measuring soil conditions, temperature, and other relevant parameters can transmit data, offering farmers insights into crop health and environmental conditions.

**Environmental Data Collection:**

**Application:** In environmental monitoring, the project serves as a valuable tool. The ESP32 web server, equipped with sensors, can collect and transmit data on air quality, temperature, and humidity, aiding in real-time environmental analysis.

**Healthcare IoT Integration:**

**Application:** Within healthcare, the ESP32 web server can function as a data aggregator for health monitoring devices. This enables the collection and secure transmission of data from wearables to a central server for comprehensive health analysis.

**Smart City Initiatives:**

**Application:** Contributing to smart city endeavors, the ESP32 web server can be applied in areas like smart parking systems, waste management, and public.

# CONCLUSION

In conclusion, the project "ESP32 Web Server using MicroPython for Emerging Trends in Computer and Information Technology" has successfully unfolded as a significant endeavor with far-reaching implications. The journey from the inception of the project to its completion has been marked by meticulous planning, intricate development processes, and a dedicated exploration of IoT technology.

The implementation of a functional web server on the ESP32 microcontroller, coupled with the efficiency and adaptability showcased in MicroPython programming, underscores the technical achievements of this project. Real-world applications spanning from smart homes to industrial IoT, environmental monitoring, and beyond, vividly illustrate the project's practical implications across diverse domains.

The project not only serves as a testament to the versatility of the ESP32 but also contributes to the ongoing discourse on emerging trends in computer and information technology. By addressing the challenges associated with secure IoT communication, real-time data exchange, and scalability, the project stands as a valuable addition to the evolving landscape of embedded systems and IoT applications.

Moreover, the documentation and insights derived from this project serve as a valuable educational resource. The detailed documentation, including code explanations and methodologies, provides a roadmap for those delving into IoT, MicroPython, and ESP32 development. This educational impact aligns with the project's broader contribution to knowledge dissemination within academic and enthusiast communities.

As the project concludes, it leaves a lasting imprint not only in the realm of technology but also in its potential to inspire future innovations. The ESP32 Web Server project, with its practical outcomes and real-world applications, represents a noteworthy stride towards harnessing the power of emerging technologies for the betterment of various aspects of our lives. Through its journey, challenges, and triumphs, this project embodies the essence of progress in the dynamic field of computer and information technology.

## DEDICATION

Hello, I'm Om Shingare. Actually, I'm deeply passionate about creating projects that serve a purpose and have a meaningful impact on my college and the broader community. My dedication lies in the belief that technology should be a force for good, addressing real-world challenges and enhancing educational experiences.

In my journey as a developer, I have consistently focused on projects that are not just about code, but about their practical applications and the value they bring to others. I have a profound aversion to investing my time and skills in projects that do not contribute positively to society. Instead, my commitment is to craft innovative solutions that are helpful, accessible, and transformative for both my peers and educators.

My dedication stems from the conviction that every project I undertake should have a purpose beyond technical proficiency. It is not merely about coding; it is about creating solutions that matter. I am motivated by the prospect of making a difference, whether it is in the classroom, the learning experiences of my fellow students, or the educational landscape of my institution.

I strongly believe in the power of education and technology to drive positive change. By dedicating my efforts to projects that are genuinely helpful for my college and others, I aim to create an impact that resonates far beyond the confines of a computer screen. Each project I undertake is a testament to my commitment to meaningful innovation, and my dedication serves as a guiding principle in my journey as a developer and a lifelong learner.