**Internet of Things**

**Phase 5:Documentation & Submission**

**Project title: public transpotation optimization**

**submitted by :**

**P.Karthikeyan**

**K.mohamed Imran**

**N.udhayakumar**

**a.karuppasamy**

**g.devaAbstract :**

* Public transportation plays a crucial role in urban mobility, addressing congestion, environmental concerns, and providing affordable transportation options. This abstract outlines the key aspects of optimizing public transportation systems. Optimization involves improving efficiency, accessibility, and sustainability. Key strategies include enhancing route planning, integrating real-time information systems, fostering multimodal transportation solutions, and employing data-driven decision-making. The success of these strategies is essential to create more attractive and efficient public transportation networks, encouraging increased ridership and promoting urban sustainability. This abstract provides an overview of the importance of optimizing public transportation and highlights key approaches to achieve these goals.

**Project Objectives:**

* **Enhance Accessibility:** Improve the accessibility of public transportation services to a wider population, including individuals with disabilities and those in underserved communities.
* **Efficiency Improvement:** Increase the overall efficiency of public transportation systems to reduce travel times, waiting times, and congestion.
* **Cost Reduction:** Identify cost-effective strategies to reduce operational costs, which may include fuel consumption, maintenance, and staffing.
* **Rider Experience:** Enhance the overall rider experience by focusing on comfort, safety, and convenience.
* **Environmental Sustainability:** Implement eco-friendly measures to reduce the carbon footprint of public transportation, such as the adoption of electric vehicles and the reduction of greenhouse gas emissions.
* **Intermodal Integration:** Foster integration between different modes of transportation (e.g., buses, trains, trams) to create a seamless, interconnected transit system.
* **Digital Services:** Develop and implement digital solutions for riders, including mobile apps, real-time tracking, and electronic payment options.
* **Data-Driven Decision Making:** Utilize data analytics and forecasting to make informed decisions about route planning, scheduling, and resource allocation.
* **Safety Measures:** Enhance safety measures for both riders and staff, such as surveillance systems, emergency response protocols, and maintenance standards.
* **Community Engagement:** Involve the community in the decision-making process, gather feedback, and ensure that the public transportation system meets the needs and preferences of the local population.
* **Financial Viability:** Ensure the long-term financial sustainability of public transportation through revenue generation, fare structure optimization, and funding diversification.
* **Adaptation to Future Needs:** Design the system with scalability and flexibility in mind to adapt to evolving transportation demands, technologies, and population growth.
* **Benchmarking:** Continuously evaluate and benchmark the system’s performance against industry standards and best practices.
* **Stakeholder Collaboration:** Collaborate with government agencies, private sector partners, and other stakeholders to achieve project objectives.

**IoT Sensors Deployment:**

* **Smart Buses/Trains:** Equip public transportation vehicles with IoT sensors to monitor their real-time location, speed, and maintenance needs. This data can be used to optimize routes, reduce fuel consumption, and ensure timely maintenance.
* **Passenger** **Counting Sensors:** Install sensors at vehicle entrances and exits to track passenger numbers. This data can help adjust schedules, dispatch additional vehicles during peak times, and improve passenger safety.
* **Traffic and Weather Sensors:** Integrate IoT sensors that monitor traffic conditions and weather. This information can be used to reroute vehicles, anticipate delays, and provide real-time updates to passengers.
* **Smart Stops/Shelters:** Deploy IoT sensors at bus and train stops to provide real-time information on arrivals, departures, and any delays. This enhances passenger convenience and helps reduce congestion.
* **Payment and Ticketing:** Implement contactless payment systems and IoT-enabled ticketing for smoother transactions and improved fare collection.
* **Energy Efficiency:** Monitor energy usage on vehicles and at transportation hubs using IoT sensors. This can lead to energy-saving strategies and reduce operating costs.
* **Predictive Maintenance:** Use IoT sensors to monitor the condition of vehicles, such as engine performance and wear and tear. Predictive maintenance can help avoid breakdowns and improve vehicle longevity.
* **Security and Surveillance:** Install security cameras and sensors on vehicles to enhance passenger safety and deter criminal activity. This data can also be used for incident investigation.
* **Data Analytics and AI:** Utilize data collected from IoT sensors to perform analytics and implement AI-driven algorithms. These can optimize routes, schedules, and maintenance plans based on historical and real-time data.
* **Passenger Feedback:** Collect feedback from passengers through IoT-enabled kiosks or mobile apps to improve services, identify areas for enhancement, and enhance customer satisfaction.
* **Environmental Impact:** Monitor the environmental impact of public transportation, including emissions and energy consumption, to support sustainability goals.
* **Emergency Response:** In case of emergencies, IoT sensors can quickly transmit data to authorities for immediate response and passenger safety.
* **Data Security:** Ensure that the data collected from IoT sensors is securely stored and transmitted to protect passenger privacy.

**Platform Development :**

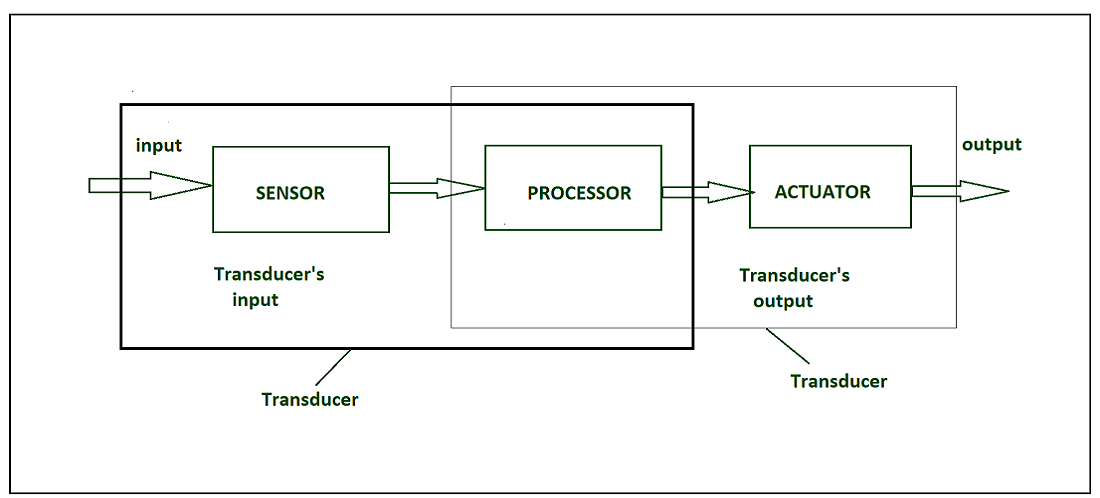
* **Project Planning:** Define your project scope, objectives, and target audience. Determine the specific goals of the optimization, such as improving route efficiency, reducing costs, or enhancing user experience.
* **Data Collection:** Gather data on existing public transportation systems, including routes, schedules, passenger data, and real-time information. This may involve partnerships with local transit authorities or data providers.
* **Data Analysis:** Use data analytics and modeling techniques to understand the current system’s performance and identify areas for improvement.
* **Algorithm Development:** Create algorithms to optimize various aspects of public transportation, such as route planning, scheduling, and resource allocation. Consider factors like demand, traffic conditions, and vehicle capacity.
* **User Interface (UI) and User Experience (UX) Design:** Develop an intuitive and user-friendly interface for passengers, transit operators, and administrators. This may include mobile apps, web portals, and dashboards.
* **Integration with Real-Time Data:** Implement mechanisms to integrate real-time data, such as GPS tracking, traffic updates, and passenger feedback, to ensure dynamic optimization.
* **Testing and Validation:** Conduct extensive testing to ensure the platform’s effectiveness, accuracy, and reliability. This may involve simulation and real-world testing.
* **Security and Privacy:** Implement robust security measures to protect sensitive data and user privacy.

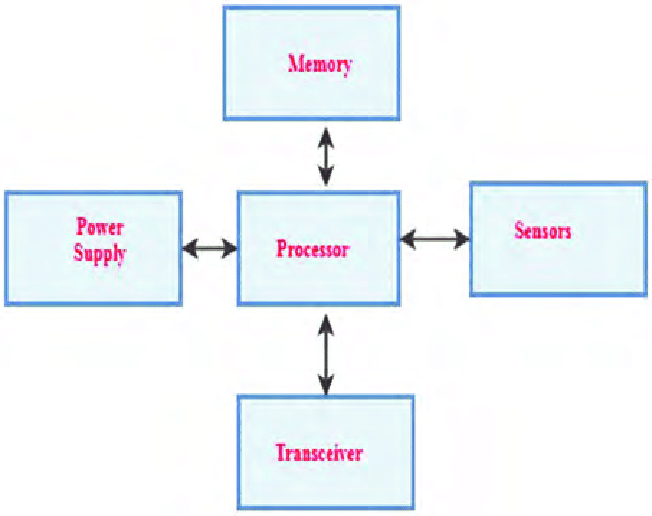
* **Scalability:** Design the platform to handle increasing data volumes and user traffic as public transportation needs evolve.
* **Maintenance and Updates:** Plan for ongoing maintenance, bug fixes, and updates to adapt to changing requirements and technologies.
* **Deployment:** Roll out the platform to specific transit agencies or regions and monitor its performance in a real-world setting.
* **Feedback and Improvement:** Gather feedback from users and operators to make continuous improvements to the platform.
* **Cost-Benefit Analysis:** Assess the economic benefits of the optimization platform, such as cost savings and improved service quality.
* **Regulatory Compliance:** Ensure compliance with local and national regulations and standards related to public transportation.
* **Marketing and Adoption:** Promote the platform to increase user adoption and encourage transit agencies to implement it.
* **Partnerships:** Collaborate with transit authorities, technology providers, and other stakeholders to gain support and resources.

**Code Implementation :**

* **Data Collection:** Gather data on routes, schedules, stops, passenger demand, and any other relevant information. This data can come from various sources, including APIs or datasets provided by transportation authorities.
* **Data Preprocessing:** Clean and format the collected data for analysis. This may involve handling missing values, converting formats, and ensuring data consistency.
* **Graph Representation:** Create a graph representation of the transportation network. Nodes represent stops or stations, and edges represent routes or connections between them. You can use libraries like Networks in Python for this.
* **Define Objectives:** Clearly define your optimization objectives. This could be minimizing travel time, maximizing service coverage, or reducing costs. The choice depends on the goals of your optimization.
* **Optimization Algorithm Selection:** If it’s a simple problem, you might use basic algorithms like Dijkstra’s or A\* for route planning.For more complex problems, consider using advanced optimization algorithms such as genetic algorithms, simulated annealing, or linear programming.
* **Implement Algorithms:** For route planning, you can implement algorithms like Dijkstra’s or A\* to find the shortest path between two stops.For scheduling optimization, you might use linear programming to optimize routes and schedules while considering constraints.
* **Integration with Real-Time Data:** If available, incorporate real-time data to adapt your plans and schedules dynamically.
* **User Interface:** Develop a user-friendly interface for passengers and transportation staff to access and interact with the optimized plans and schedules.
* **Testing and Validation:** Thoroughly test your code using historical data and, if possible, in a controlled environment. Validate the results to ensure they meet your optimization objectives.
* **Deployment:** Once your code is working well, deploy it to the transportation system, and continuously monitor and update it based on changing conditions and feedback.

**Daigrams :**





**Wokwi Simulation :**

#define BLYNK\_TEMPLATE\_ID "TMPL26V4fGv5q"

#define BLYNK\_TEMPLATE\_NAME "Test"

#define BLYNK\_AUTH\_TOKEN "XEHxNF\_Ur1Nt2p7wB5B20dNI1ZUwj34P"

#include <WiFi.h>

#include <WiFiClient.h>

#include <BlynkSimpleEsp32.h>

int duration1 = 0;

int distance1 = 0;

int duration2 = 0;

int distance2 = 0;

int dis1 = 0;

int dis2 = 0;

int dis\_new1 = 0;

int dis\_new2 = 0;

int entered = 0;

int left = 0;

int inside = 0;

#define LED 2

#define PIN\_TRIG1 15

#define PIN\_ECHO1 14

#define PIN\_TRIG2 13

#define PIN\_ECHO2 12

BlynkTimer timer;

char auth[] = BLYNK\_AUTH\_TOKEN;

char ssid[] = "Wokwi-GUEST";   // your network SSID (name)

char pass[] = "";

#define BLYNK\_PRINT **Serial**

long get\_distance1() {

  // Start a new measurement:

  digitalWrite(PIN\_TRIG1, HIGH);

  delayMicroseconds(10);

  digitalWrite(PIN\_TRIG1, LOW);

  // Read the result:

  duration1 = pulseIn(PIN\_ECHO1, HIGH);

  distance1 = duration1 / 58;

  return distance1;

}

long get\_distance2() {

  // Start a new measurement:

  digitalWrite(PIN\_TRIG2, HIGH);

  delayMicroseconds(10);

  digitalWrite(PIN\_TRIG2, LOW);

  // Read the result:

  duration2 = pulseIn(PIN\_ECHO2, HIGH);

  distance2 = duration2 / 58;

  return distance2;

}

void myTimer() {

**Serial**.println("100");

  dis\_new1 = get\_distance1();

  dis\_new2 = get\_distance2();

  if (dis1 != dis\_new1 || dis2 != dis\_new2){

**Serial**.println("200");

    if (dis1 < dis2){

**Serial**.println("Enter loop");

      entered = entered + 1;

      inside = inside + 1;

      digitalWrite(LED, HIGH);

      Blynk.virtualWrite(V0, entered);

      Blynk.virtualWrite(V2, inside);

      dis1 = dis\_new1;

      delay(1000);

      digitalWrite(LED, LOW);

    }

    if (dis1 > dis2){

**Serial**.println("Leave loop");

      left = left + 1;

      inside = inside - 1;

      Blynk.virtualWrite(V1, left);

      Blynk.virtualWrite(V2, inside);

      dis2 = dis\_new2;

      delay(1000);

    }

  }

}

 void setup() {

**Serial**.begin(115200);

  pinMode(LED, OUTPUT);

  pinMode(PIN\_TRIG1, OUTPUT);

  pinMode(PIN\_ECHO1, INPUT);

  pinMode(PIN\_TRIG2, OUTPUT);

  pinMode(PIN\_ECHO2, INPUT);

  Blynk.begin(auth, ssid, pass, "blynk.cloud", 8080);

  timer.setInterval(1000L, myTimer);

}

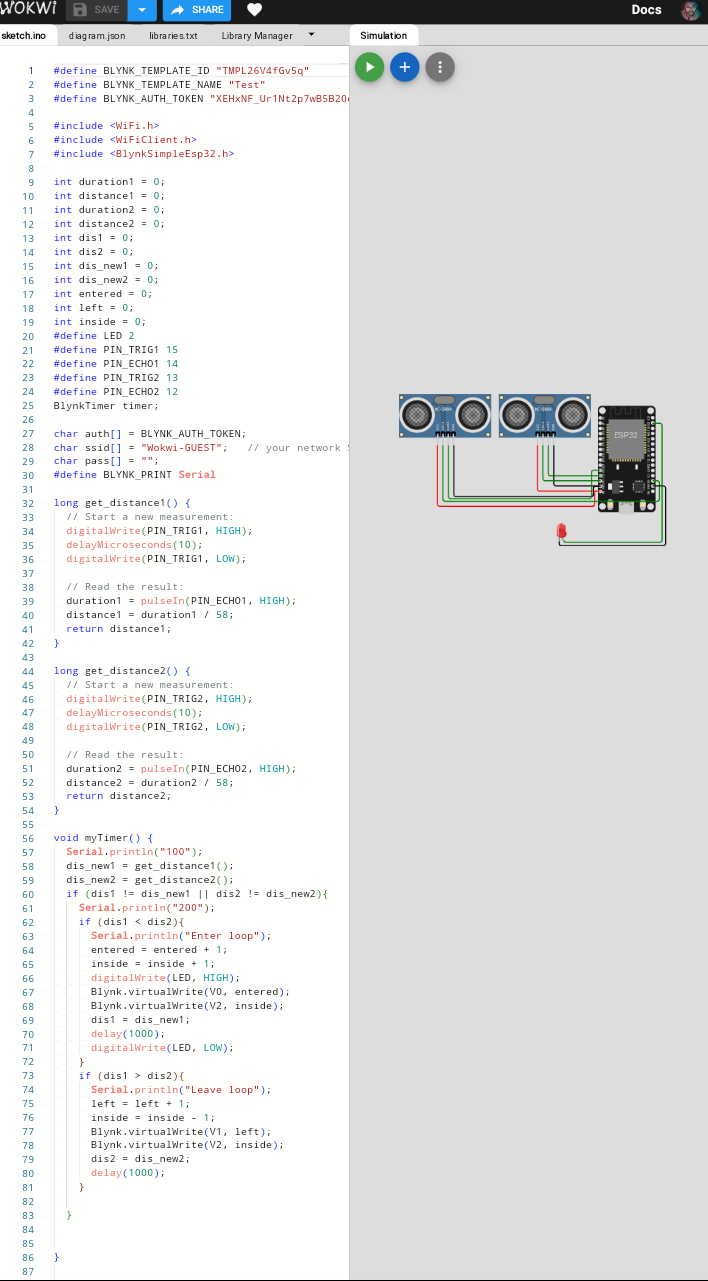
void loop() {

  Blynk.run();

  timer.run();

}

**Wokwi Simulation Output :**

****

**Web Platform Development :**

**HTML :**

<!DOCTYPE html>

<html lang=”en”>

<head>

<meta charset=”UTF-8”>

<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>

<title>Real-Time Transit Information</title>

<link rel=”stylesheet” href=”styles.css”>

</head>

<body>

<header>

<h1>Real-Time Transit Information Platform</h1>

</header>

<nav>

<ul>

<li><a href=”#live-map”>Live Map</a></li>

<li><a href=”#sensor-data”>Sensor Data</a></li>

</ul>

</nav>

<section id=”live-map”>

<h2>Live Map</h2>

<div id=”map”></div>

</section>

<section id=”sensor-data”>

<h2>Sensor Data</h2>

<div id=”sensor-display”></div>

</section>

<script src=”script.js”></script>

</body>

</html>

**CSS :**

Body {

Font-family: Arial, sans-serif;

Margin: 0;

Padding: 0;

Background-color: #f0f0f0;

}

Header {

Background-color: #333;

Color: #fff;

Text-align: center;

Padding: 1rem 0;

}

Nav {

Background-color: #444;

Color: #fff;

Text-align: center;

Padding: 1rem 0;

}

Nav ul {

List-style: none;

Padding: 0;

}

Nav ul li {

Display: inline;

Margin-right: 20px;

}

Nav a {

Text-decoration: none;

Color: #fff;

Font-weight: bold;

}

**JavaScript :**

// Simulated sensor data (replace with real data retrieval logic)

Const sensorData = {

Ultrasonic: “45.2 meters”,

Magnetic: “73 units”,

Infrared: “120 units”,

Camera: “No data available”,

groundLoop: “2 loops detected”,

pavement: “Dry”,

wireless: “Strong signal”,

radar: “Moderate traffic”,

gps: “Latitude: 40.7128, Longitude: -74.0060”,

weather: “Temperature: 70°F, Sunny”,

trafficDensity: “Medium”

};

// Function to display sensor data on the page

Function displaySensorData() {

Const sensorDisplay = document.getElementById(“sensor-display”);

sensorDisplay.innerHTML = “<h3>Sensor Data</h3>”;

for (const sensor in sensorData) {

const sensorValue = sensorData[sensor];

sensorDisplay.innerHTML += `<p>${sensor}: ${sensorValue}</p>`;

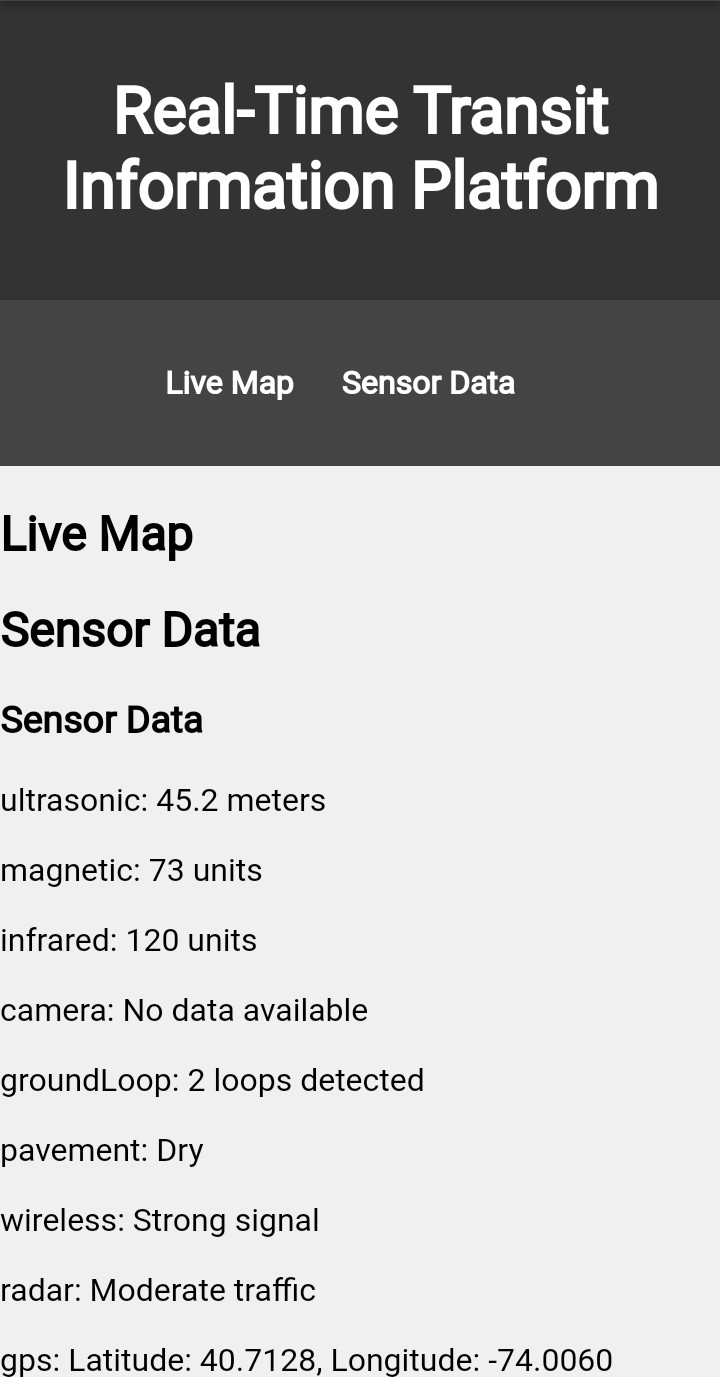
}

}

// Call the function to display sensor data

displaySensorData();

**Web Platform Development Output :**

****

**A real-time transit information system can significantly enhance public transportation services and passenger experience in several ways:**

* **Accurate Arrival Times:** Passengers can access real-time information on the arrival and departure times of buses, trains, or trams, reducing uncertainty and wait times.
* **Efficient Route Planning:** These systems provide up-to-date route information, helping passengers plan their journeys more efficiently, including finding alternate routes during disruptions.
* **Reduced Crowding:** Passengers can see how full a vehicle is in real-time, which enables them to choose less crowded options, enhancing comfort and safety.
* **Improved Safety:** Real-time data can help passengers stay informed about any safety alerts, incidents, or emergencies, allowing them to make informed decisions.
* **Mobile Apps and Notifications:** Many transit systems offer mobile apps with real-time information, enabling passengers to receive alerts and updates on their smartphones, enhancing convenience.
* **Accessibility:** Real-time information can be particularly useful for passengers with disabilities, ensuring they can plan their trips and make necessary arrangements.
* **Increased Ridership:** Reliable real-time information can attract more passengers to use public transportation, reducing congestion and pollution in cities.
* **Operational Efficiency:** Transit agencies can use real-time data to optimize their operations, such as adjusting schedules based on demand, which can lead to cost savings.
* **Customer Feedback:** Passengers can provide feedback or report issues in real-time, allowing transit agencies to address concerns promptly.
* **Integration with Other Services:** Real-time transit data can be integrated into other services like ride-sharing apps, making it easier for passengers to combine different modes of transportation.

**Conclusion:**

* The implementation of this real-time transit information system can significantly improve public transportation services and enhance the passenger experience. By deploying IoT sensors, developing a robust platform, and providing accurate, real-time data, public transportation becomes more reliable, efficient, and user-friendly, contributing to the overall improvement of urban mobility.

**THANK YOU**