IOT BASED PORTABLE TRACKING DEVICE

INTERDISCIPLINARY PROJECT II

Submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering

By

KAVIRITHANYA E V

Reg. No–42110613

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY (DEEMED TO BE UNIVERSITY)

CATEGORY - 1 UNIVERSITY BY UGC

Accredited with Grade “A++” by NAAC I Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600119

APRIL - 2025

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
CATEGORY - 1 UNIVERSITY BY UGC
Accredited with Grade "A++" by NAAC I Approved by AICTE
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Interdisciplinary Project II Report is the Bonafide work of **KAVIRITHANYA E V (Reg. No–42110613)** who carried out the Project entitled **"IOT BASED PORTABLE TRACKING DEVICE"** under my supervision from January 2025 to April 2025.

Internal Guide
Dr. S. PRAYLA SHYRY, M.E., Ph.D.,

Head of the Department
Dr. L. LAKSHMANAN, M.E., Ph.D.,

Submitted for Interdisciplinary Project Viva Voce Examination held on __________

Internal Examiner **External Examiner**

DECLARATION

I, **KAVIRITHANYA E V (Reg. No–42110613)**, hereby declare that Interdisciplinary Project II Report entitled "**IOT BASED PORTABLE TRACKING DEVICE"** done by me under the guidance of **Dr. S. PRAYLA SHYRY, M.E., Ph.D.,** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering.**

DATE: 26/04/2025

PLACE: Chennai **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph. D.**, **Dean**, School of Computing, and **Dr. L. Lakshmanan, M.E., Ph.D.,** Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. S. PRAYLA SHYRY, M.E, Ph.D.,** for her valuable guidance, suggestions, and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the Interdisciplinary project II.

TRAINING CERTIFICATE

TECHKNOTS ACADEMY LLP

CERTIFICATE

OF COMPLETION

This is to certify that Kavirithanya E V(42110613) has successfully completed the professional training course on IoT-Smart Home Automation: Design & Implementation using Cloud Computing for 6 Days and completed a mini project under Smart Home Automation which was organized by the department of Computer Science in association with Techknots Academy at Sathyabama University. This achievement reflects dedication, diligence, and a deep understanding of the subject matter.

Nixon E
Founder & Director
Techknots Academy LLP

ABSTRACT

In the modern era of smart technology, real-time location tracking has become essential for various applications such as asset tracking, personal safety, and navigation. This project focuses on building a GPS-based tracking system using IoT components including the ESP8266 microcontroller, Neo 6M GPS module, piezo buzzer, and jumper wires. The main objective is to accurately capture the device's geographical coordinates and reflect them in an interactive and user-friendly front-end interface. The GPS module retrieves the latitude and longitude of the device, and these values are displayed on the front end when the user clicks on the "FIND NOW" button. The coordinates are then marked on a digital map, enabling the user to visualize the current location of the device in real time. The entire system operates through wireless communication and ensures efficient and fast data transmission. An additional feature includes a buzzer that is controlled remotely via the "Buzz Device" button. This function is especially useful when the user is physically near the device but needs an auditory signal to locate it precisely. The hardware components are programmed using the Arduino IDE. This system offers a compact, cost-effective, and reliable solution for location tracking, integrating both software and hardware in a seamless manner. It highlights the practical use of IoT in solving real-world problems by combining live tracking with interactive control features.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1. BACKGROUND

With the rapid growth of technology and increased mobility in daily life, the need for effective tracking solutions has become more critical than ever. People often misplace essential belongings such as bags, wallets, or keys, and there is a rising demand for tools to monitor the whereabouts of vehicles, pets, children, or elderly family members. While commercial products like Apple AirTag and Tile offer location tracking, these devices typically rely on Bluetooth connectivity and depend on nearby smartphone ecosystems, which significantly limit their range and usefulness in rural or low-connectivity environments. Such systems also lack flexibility for DIY customizations or integration with open-source tools, making them unsuitable for personalized use cases. Moreover, their reliance on proprietary platforms restricts control over data privacy, functionality, and adaptability.

To address these challenges, GPS and IoT technologies provide a more scalable and accessible alternative. By integrating the Neo-6M GPS module for accurate positioning and the NodeMCU ESP8266 microcontroller for Wi-Fi-based data transmission, developers can create a low-cost, portable tracking system that operates independently of Bluetooth or mobile networks. This opens the door for more inclusive applications, particularly in regions with poor infrastructure or limited access to premium tracking services. The project leverages these technologies to deliver a practical and affordable solution for real-time location monitoring, while also incorporating features like a piezo buzzer for locating nearby items, further enhancing its usability in everyday life.

1.2. PROBLEM STATEMENT

In today’s increasingly mobile and fast-paced world, the need to reliably track and locate valuable items, vehicles, and loved ones has become a growing concern. Existing commercial tracking solutions, such as Bluetooth-based devices or cellular trackers, often come with limitations including short range, high cost, or dependence on cellular networks, making them unreliable or inaccessible in remote or low-connectivity areas. This creates a gap in providing an affordable, efficient, and widely available solution for real-time tracking. There is a pressing need for a compact, low-cost device that can deliver accurate geolocation information without relying on Bluetooth or cellular infrastructure. Addressing this issue, the proposed project aims to develop a GPS and IoT-based tracking system using the Neo-6M GPS module and NodeMCU ESP8266, capable of transmitting real-time location data over Wi-Fi to a web or mobile interface, while also including a manual buzzer function for locating nearby items.

1.3. OBJECTIVES

- To design and implement a GPS-based tracking system using ESP8266 and Neo GPS module.
- To retrieve accurate real-time latitude and longitude coordinates from the GPS module.
- To transmit the location data wirelessly and display it on a user-friendly front-end interface.
- To mark the captured coordinates on a digital map for easy location visualization.
- To enable users to access the location information remotely using Blynk IoT platform.
- To integrate a buzzer module that can be triggered remotely to assist in locating the device.
- To ensure seamless communication between hardware components through Wi-Fi connectivity.
- To use Arduino IDE for programming the hardware and managing system logic.
- To create a compact, portable, and low-cost solution suitable for real-world tracking needs.
- To demonstrate the effective use of IoT in real-time monitoring and location-based applications.

1.4. SCOPE OF THE PROJECT

This project aims to develop a portable, real-time GPS tracking device using IoT technology to deliver accurate location data without relying on Bluetooth or cellular networks. Built with the Neo-6M GPS module and NodeMCU ESP8266 microcontroller, the system ensures affordability and ease of use. It transmits geolocation data via Wi-Fi to a web or mobile interface, allowing users to monitor the location of personal belongings, vehicles, or loved ones from anywhere with internet access. A manually activated piezo buzzer helps users locate nearby items through sound.

Designed for applications such as luggage tracking, elderly or child monitoring, and vehicle security, the system can also be adapted for small-scale commercial use. The project involves hardware integration, data communication, and a simple front-end for real-time visualization. Future improvements may include features like geofencing, power management, and smart alerts for enhanced functionality.

1.5. SIGNIFICANCE OF THE STUDY

The significance of this project lies in its ability to provide a cost-effective, portable, and reliable real-time tracking solution, overcoming the limitations of existing commercial trackers. Unlike Bluetooth-based devices that require close-range connectivity or cellular-based trackers that rely on costly subscriptions, this system uses GPS and Wi-Fi to offer location tracking without such constraints, making it ideal for rural areas or places with limited connectivity.

By using affordable components like the Neo-6M GPS module and NodeMCU ESP8266, the project demonstrates how low-cost hardware can solve real-world problems through IoT. With its potential applications in personal safety, asset security, and ease of use, this project contributes to the development of smart tracking technologies, offering an accessible, customizable alternative for personal and small-scale commercial use.

CHAPTER 2
LITERATURE SURVEY

2.1. GPS BASED TRACKING SYSTEMS

- ***Authors:*** Kunal Maurya, M. Singh, N. Jain
- ***Reference:*** Maurya, K., Singh, M., & Jain, N. (2012). Real Time Vehicle Tracking System using GSM and GPS Technology. *International Journal of Electronics and Computer Science Engineering*.
- ***Summary:***

  This study proposed a vehicle tracking system using GPS for location tracking and GSM for communication. The system sends the location coordinates to a mobile device via SMS. While effective for long-distance tracking, it incurs operational costs due to the use of GSM and lacks real-time data visualization on maps.

2.2. IOT-BASED LOCATION TRACKING

- ***Authors:*** R. Kumar, M. P. Rajasekaran
- ***Reference:*** Kumar, R., & Rajasekaran, M. P. (2016). IoT-based smart GPS device for child tracking. *In IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology*.
- ***Summary:***

  This study explored the use of IoT-enabled GPS tracking for public transport monitoring, leveraging Wi-Fi-enabled microcontrollers and cloud services. The system eliminated the need for GSM, offering real-time location updates that can be accessed via the internet, improving efficiency and reducing operational costs.

2.3. BLUETOOTH-BASED TRACKERS

- ***Author:*** Apple Inc.
- ***Reference:*** Apple Inc. (2021). AirTag Technology Overview.

- ***Summary:***

  Devices such as Apple AirTag and Tile use Bluetooth Low Energy (BLE) for locating personal items. These devices rely on nearby smartphones within the same ecosystem to update and report location data. However, their range is limited to 10–30 meters, and their performance degrades in low-density or rural areas.

2.4. GPS MODULE NEO-6M

- ***Author:*** Ublox AG
- ***Reference:*** Ublox AG. (2013). NEO-6 Data Sheet: GPS Modules for Embedded Applications.
- ***Summary:***

  The Neo-6M GPS module is widely used in GPS-based applications due to its cost-effectiveness, accuracy, and ease of integration with microcontrollers such as Arduino and NodeMCU. It provides location updates with a positional accuracy of about 2.5 meters, making it ideal for real-time tracking applications. It is commonly integrated with cloud services like ThingSpeak or Firebase for remote monitoring.

2.5. ESP8266 FOR IOT APPLICATIONS

- ***Authors:*** R. Patel, R. Agrawal
- ***Reference:*** Patel, R., & Agrawal, R. (2018). IoT Based Real-Time Vehicle Tracking System Using NodeMCU. *International Journal of Scientific Research in Computer Science and Engineering*.
- ***Summary:***

  The NodeMCU ESP8266 microcontroller is widely used in IoT applications due to its built-in Wi-Fi capabilities and support for various communication protocols. Researchers have employed the ESP8266 to create low-cost, Wi-Fi-enabled tracking systems that publish data to cloud dashboards, offering real-time tracking solutions for vehicles.

2.6. WEB-BASED GPS TRACKING INTERFACES

- ***Authors:*** S. R. Jadhav, P. A. Patil
- ***Reference:*** Jadhav, S. R., & Patil, P. A. (2019). Real-Time GPS Vehicle Tracking System Using Web-Based Application. *International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)*.
- ***Summary:***

  This study presents a GPS tracking system integrated with a web-based interface for real-time vehicle monitoring. The system captures GPS data using a GPS module and transmits it via Wi-Fi to a web server. Users can access live location updates through a browser, making it accessible across multiple platforms. This eliminates the need for mobile applications or GSM modules and enhances visual tracking with interactive maps.

2.7. LOW-POWER IOT TRACKING DEVICES

- ***Authors:*** J. Kim, H. Park
- ***Reference:*** Kim, J., & Park, H. (2020). Design of Low-Power IoT Devices for Real-Time Location Monitoring. *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9503–9511. doi: 10.1109/JIOT.2020.2982398
- ***Summary:***

  This paper focuses on optimizing power consumption in IoT-based GPS trackers for longer battery life. The researchers implemented deep sleep modes in ESP-based devices and used efficient data transmission techniques to reduce energy usage. The design supported cloud connectivity and was tested for applications like wearable GPS trackers and smart luggage. The study highlights power optimization as a key factor in portable tracker performance.

CHAPTER 3
REQUIREMENT ANALYSIS

3.1. FUNCTIONAL REQUIREMENTS:

I. ***Real-Time Location Tracking:***

The system must continuously track the device's location using a Neo-6M GPS module. The device must update the real-time location data (latitude and longitude) to a connected web or mobile interface via NodeMCU ESP8266 (Wi-Fi-enabled). The GPS coordinates should be sent periodically (e.g., every 10 seconds) to the cloud or server to ensure up-to-date information.

II. ***User Interface for Location Viewing:***

The system must allow users to view the current location of the device on a map interface on a web or mobile app. The user should be able to zoom in/out, view real-time coordinates, and track the device’s location over time. The app should display the location status, including the device's current position and any historical location updates.

III. ***Piezo Buzzer Alert:***

The device must be equipped with a piezo buzzer that can be manually triggered via the web or mobile app. When the "Buzz Device" button is pressed, the device should sound the buzzer to assist the user in locating the item or individual associated with the device. The buzzer should sound for a user-defined period (e.g., 5 seconds).

IV. ***Geofencing Capability:***

The system must allow users to define a geofence for the device. If the device enters or exits the geofence, it should trigger the buzzer to alert the user. The geofence radius should be configurable by the user (e.g., 50m, 100m, etc.).

V. ***Power Management:***

The device should be designed to operate on a rechargeable battery that lasts for a reasonable duration (e.g., 12-24 hours depending on usage). The device should have a low-power mode to conserve battery when idle or when no user interactions are detected.

VI. ***Wi-Fi Connectivity:***

The device should connect to a Wi-Fi network using the NodeMCU ESP8266 microcontroller. Wi-Fi connection stability should be ensured for continuous transmission of location data. In case of Wi-Fi disconnection, the device should attempt to reconnect automatically.

VII. ***Manual Buzzer Control:***

Users must be able to manually activate the buzzer via the mobile interface. The buzzer should be activated when the user presses Buzzer button in the interface.

VIII. ***Location Data Logging:***

The system should log historical location data for a defined period (e.g., last 7 days) on the cloud. Users should have the option to view the location history for tracking the device's movement over time.

3.2. NON-FUNCTIONAL REQUIREMENT:

Non-functional requirements relate to how the system performs its functions under constraints:

I. ***Performance:***

The system should update the location data to the web/mobile app within 5-10 seconds of receiving the GPS coordinates. The buzzer should activate within 1-2 seconds after receiving the trigger command.

II. ***Scalability:***

The system should support adding multiple devices to the same network, allowing tracking of several items simultaneously.

III. ***Reliability:***

The location data should be reliable and accurate within the limits of the GPS module (i.e., within 2-5 meters of the actual location). The system should be capable of handling intermittent Wi-Fi disconnections and automatically reconnecting.

IV. ***Security:***

All communication between the device and the mobile/web interface should be encrypted to ensure data privacy and security.

V. ***User Experience:***

The mobile/web interface should be user-friendly and intuitive, allowing users to easily track the device, configure geofences, and activate the buzzer. Notifications should be timely and easy to understand, providing relevant location information.

VI. ***Battery Efficiency:***

The device must be optimized to minimize energy consumption without sacrificing functionality. It should have a low-power mode during idle periods. The system must be able to operate for at least 12 hours on a single charge, under normal usage conditions.

3.3. HARDWARE REQUIREMENTS:

To fulfill the system's design, the following hardware components are required:

1. ***GPS Module: Neo-6M***

- **Purpose**: Provides real-time geolocation data (latitude and longitude).
- **Specifications**:
  - Accuracy: Around 2.5 meters
  - Operating Voltage: 3.3V or 5V
  - Interface: UART (Serial communication)
  - Update Rate: 1Hz (updates every second)

2. *Microcontroller: NodeMCU ESP8266*

- **Purpose**: Manages communication between the GPS module, buzzer, and mobile/web interface.
- **Specifications**:
  - Connectivity: Built-in Wi-Fi (ESP8266 chipset)
  - Logic Voltage: 3.3V
  - Power Supply: 5V via USB
  - Processing Speed: 80 MHz
  - GPIO Support: Yes, for external devices like buzzers
  - *Storage:* Flash memory for firmware and data transmission

3. *Piezo Buzzer*

- ***Purpose****:* Provides an audible alert when triggered.
- ***Control Mechanism:*** Activated through a digital output pin on the NodeMCU using simple HIGH/LOW logic or digitalWrite() command in Arduino
- ***Specifications****:*
  - Operating Voltage: 5V
  - Sound Output: High-pitched tone
  - Current Consumption: Low (suitable for battery-powered systems)
  - Size: Compact and easily integrable

4. *Power Supply*

- ***Purpose****:* Provides portable and reliable power to the system.
- ***Specifications****:*
  - Type: Rechargeable Lithium Polymer (Li-Po) or Lithium-Ion (Li-Ion) battery
  - Capacity: 3.7V, 2000mAh (recommended)
  - Backup Time: 12–24 hours of continuous operation
  - Charging Module: TP4056 USB charging circuit

5. ***Enclosure/Case***

- ***Purpose:*** Protects internal electronic components during field deployment.
- ***Specifications:***
    - Material: Durable and lightweight (plastic or ABS)
    - Design: Waterproof or water-resistant for outdoor use

6. ***Jumper Wires***

- ***Purpose:*** Facilitates connections between components without soldering.
- ***Specifications:***
    - Types: Male-to-Male and Male-to-Female
    - Application: Breadboard and module interfacing
    - Quantity: 10–15 male-to-male, 5–10 male-to-female
    - Features: Flexible, insulated, and reusable

7. ***Wi-Fi Router (for Cloud Communication)***

- ***Purpose:*** Provides internet access to upload location data to cloud services.
- ***Stability Consideration:*** A stable internet connection is essential to ensure uninterrupted data transmission and real-time updates on the frontend interface.
- ***Specifications:***
    - Frequency Band: 2.4GHz (compatible with NodeMCU ESP8266)
    - Requirement: Internet-enabled router with access to a cloud database or server

3.4. SOFTWARE REQUIREMENTS

This project requires a combination of embedded programming, backend processing, and frontend visualization. It involves real-time data acquisition from hardware sensors, cloud-based communication via IoT protocols, and seamless user interaction through a web interface. The system also emphasizes low-power operation and portability, making it suitable for both indoor and outdoor tracking scenarios. The following software tools and languages are used:

1. Embedded Software for NodeMCU (Firmware)

- ***Programming Language****:* Arduino C/C++ (using the Arduino IDE)
- ***Libraries Required****:*
  - SoftwareSerial: For communication between NodeMCU and the GPS module
  - Wi-Fi: For connecting NodeMCU to Wi-Fi networks
  - HTTP Client: For transmitting data to a cloud service or web server
  - TinyGPS++: To parse GPS data and extract relevant information (e.g., latitude, longitude, altitude)
- ***Functionality****:*
  - Initialize GPS module
  - Read GPS coordinates periodically
  - Send location data to a cloud platform (e.g., Firebase, ThingSpeak, custom server)
  - Trigger piezo buzzer when required (via app command or geofence breach)
  - Manage Wi-Fi connection with auto-reconnect
  - Optimize battery usage with sleep/power management logic
  - Log GPS data locally (optional): Store recent GPS coordinates in onboard memory for redundancy in case of temporary internet loss.

2. *Mobile App or Web Interface (Frontend)*

- ***Technology****:* HTML/CSS/JavaScript (for web interface)
- ***Functionality****:*
  - Display real-time location on an interactive map
  - Allow users to set geofences around specific locations (e.g., home, office)
  - Provide a "Locate" button to remotely trigger the buzzer
  - Show notifications on location updates or geofence events
  - User-friendly UI for seamless interaction.
  - Enable historical tracking: View previously recorded paths or visited locations for better context and movement history.

3. ***Cloud Server or Database***

- ***Platform****:*

  Firebase, AWS IoT, ThingSpeak, or custom cloud-based backend
- ***Functionality****:*
  - Store real-time GPS data (latitude, longitude, timestamp)
  - Provide historical location data for analysis
  - Support real-time sync with web/mobile interface
  - Ensure user authentication and secure data access (OAuth/Firebase Auth)
  - Push event-driven updates (e.g., geofence crossing)

4. ***Notification System***

- ***Technology****:* Firebase Cloud Messaging (FCM) or equivalent push notification services
- ***Functionality****:*
  - Send notifications when device exits or enters geofenced area
  - Alert user when the buzzer is triggered remotely
  - Include location coordinates in the notification message
  - Ensure timely delivery of alerts to improve responsiveness

5. Geofencing Logic

- ***Algorithm****:* Distance-based geofencing implemented in mobile app or frontend
- ***Functionality:***
  - Allow users to define a virtual boundary with a set radius (e.g., 100 meters)
  - Continuously compare live GPS location with the defined boundary
  - Trigger alerts or buzzer on geofence entry/exit events
  - Log geofence breach events in cloud database

CHAPTER 4
METHODS AND IMPLEMENTATIONS

4.1. SYSTEM OVERVIEW:

The proposed system is a GPS-based portable tracking device designed to track the location of personal belongings, vehicles, or even vulnerable individuals. It uses GPS and IoT technologies (Wi-Fi) to offer real-time location tracking and includes a piezo buzzer that can be manually activated to assist the user in locating the device. This system is intended to provide a reliable and affordable alternative to commercial tracking devices (such as Apple AirTag), especially in environments with limited Bluetooth or cellular connectivity. It features a user-friendly web interface that displays live coordinates and map visuals, enhancing accessibility. The device is compact, cost-effective, and easily customizable, making it suitable for various everyday tracking applications.

4.2. WORKING PRINCIPLE:

The working principle of the GPS-based Portable Tracking Device with Piezo Buzzer is centred around real-time geolocation tracking using GPS technology and IoT communication via Wi-Fi. Here’s how the system operates:

1. ***GPS Data Acquisition:***
   The Neo-6M GPS module is continuously gathering GPS data (latitude, longitude, and time) at regular intervals. The GPS data is transmitted to the NodeMCU ESP8266 microcontroller via serial communication.

2. ***Microcontroller Processing:***
   The NodeMCU ESP8266 microcontroller processes the incoming GPS data and prepares it for transmission to the cloud server. The microcontroller also handles user input for triggering the piezo buzzer and geofence monitoring.

3. ***Cloud Communication:***

   The NodeMCU establishes a Wi-Fi connection and sends the processed location data to a cloud server (such as Firebase, ThingSpeak, or a custom server). The cloud server stores and provides real-time location data to the mobile/web app. **Geofencing:** The system continuously checks if the device is inside or outside the predefined geofenced area. If a violation occurs (i.e., the device enters/exits the area), an alert is sent.

4. ***User Interaction and Alert Mechanism:***

   The user can access the mobile/web app to view the device's real-time location on a map interface. Locate Feature: When the user wants to locate the device, they press a button, which sends a signal to the NodeMCU to activate the piezo buzzer. Geofence Alerts: When the device enters or exits a geofence, the app sends a push notification to the user, and optionally, triggers the piezo buzzer.

5. ***Manual Buzzer Trigger:***

   The piezo buzzer emits an audible sound (high-pitched) for a set period when activated, helping the user locate the device if it is nearby or misplaced.

6. ***Location History and Notifications:***

   The system logs the device's location history in the cloud for the user to access later. The app notifies the user if the device is moved out of the geofenced area or if there is a new location update.

4.3. KEY FEATURES OF THE PROPOSED SYSTEM

1. ***Real-Time Location Tracking:***

   The system provides real-time GPS-based location updates for tracking personal belongings, vehicles, or vulnerable individuals. The location is transmitted to the cloud and made accessible via a mobile/web app.

2. ***Manual Buzzer Trigger:***

   A piezo buzzer is included to help users locate the device when it is nearby. The buzzer can be triggered remotely via the app.

3. ***Geofencing:***

   Users can define a geofenced area around a specific location. The system automatically triggers alerts if the device enters or exits the geofenced area.

4. ***Cloud Integration:***

   The system store location data on the cloud, allowing users to access historical tracking information and location logs. Push notifications are sent to the user's mobile/web app for events like entering/exiting the geofence.

5. ***Portable and Battery-Powered:***

   The device is designed to be lightweight and powered by a rechargeable battery, making it highly portable and ideal for tracking items on the go. Battery efficiency ensures it lasts for up to 12-24 hours on a single charge.

6. ***Mobile/Web App Interface:***

   The app displays the device's real-time location on an interactive map and provides functionalities like locating the device, setting up geofences, and viewing historical data. The app is designed to be user-friendly, with simple controls for interacting with the system.

7. ***User Notifications:***

   Push notifications alert the user when the device crosses a geofence or when an important event occurs (e.g., buzzer trigger). The app provides instant updates on the device's status, ensuring the user is always informed.

8. ***Security and Authentication:***

   The system ensures data privacy and user authentication through secure communication (e.g., HTTPS) between the device, cloud, and app. Only authorized users can access location data and manage device settings.

Advantages Over Existing Solutions:

TABLE 4.1. Advantages over existing solutions

Advantage	***Description***
Real-Time GPS Tracking	Provides direct GPS-to-cloud tracking, unlike BLE-based systems that rely on nearby devices.
Geofencing and Notifications	Allows users to set custom virtual boundaries and receive alerts when the device enters/exits these areas.
Manual Locator with Buzzer	Users can remotely activate a piezo buzzer through the app for immediate device location.
Affordable and Scalable	Utilizes cost-effective, open-source components (e.g., NodeMCU, GPS module) and supports tracking of multiple devices.
Enhanced Privacy	No dependency on third-party networks like Apple's, ensuring user-controlled data and privacy.
Battery Efficiency	Optimized for low power usage, offering 12–24 hours of operation without frequent charging.

CHAPTER 5
SYSTEM ARCHITECTURE

The system architecture of the GPS-based Portable Tracking Device involves multiple components working together to provide a seamless tracking experience. The architecture includes hardware components, software components, and the communication between them. The integration of cloud connectivity through the Blynk IoT platform ensures real-time data exchange and remote control capabilities from any location with internet access. The following diagram provides a visual representation of the system architecture, followed by a detailed explanation of each component.

5.1. SYSTEM ARCHITECTURE DIAGRAM:

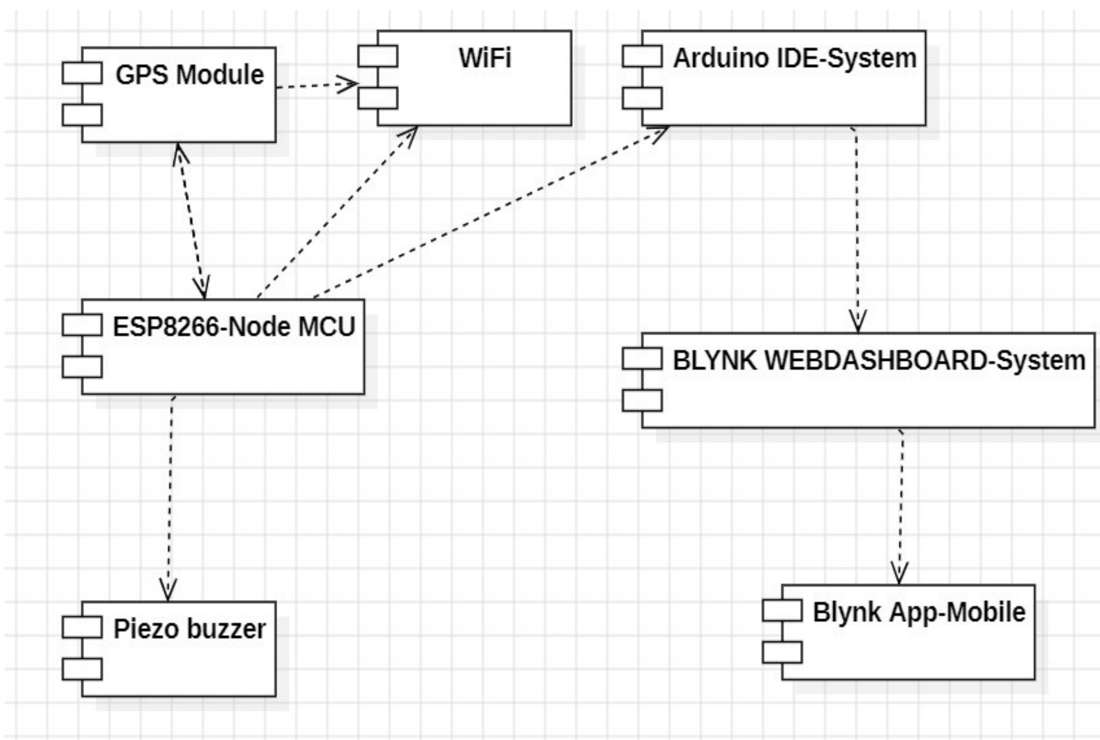

Figure 5.1. System Architecture Diagram

5.2. WORKING DESCRIPTION:

- ***Initialization:*** ESP8266 initializes and connects to the Wi-Fi network.
- ***GPS Tracking****:* The GPS module continuously receives satellite signals to determine current coordinates.
- ***Data Transmission****:* ESP8266 collects this data and uploads it to a cloud server or sends directly to a mobile app via HTTP/MQTT.
- ***User Interaction****:*
    - The user can see the location on the mobile app.
    - When needed (e.g., finding a lost item nearby), the user can activate the buzzer from the app.
- ***Buzzer Activation****:* ESP8266 receives the control signal and powers the buzzer, aiding in physical detection.

5.3. CIRCUIT DESIGN:

The circuit design of the GPS tracking system is centered around the ESP8266 NodeMCU, a Wi-Fi-enabled microcontroller that serves as the main control unit. It interfaces with the NEO-6M GPS module, which continuously receives satellite signals to determine the device's real-time geographical location. The GPS data is processed by the NodeMCU and transmitted to the cloud via Wi-Fi. To aid in manual localization, a piezo buzzer is connected to the system, which can be remotely triggered through a mobile application to produce sound. The mobile app interface, such as Blynk or a custom-built app, acts as the user control panel, allowing users to monitor the device's location, trigger the buzzer, and receive geofencing notifications. The overall circuit is designed to be compact, power-efficient, and reliable for real-time tracking applications. All components are powered through a single 5V input, making the system highly portable and suitable for battery-powered deployment in outdoor too. Jumper wires and breadboard are used during prototyping to ensure flexible and non-permanent connections between modules. The use of libraries like TinyGPS++ and BlynkSimpleEsp8266 simplifies code integration and supports fast, customizable development.

Connections:

TABLE 5.1. Connections

Component	***ESP8266 Pin***	***Description***
GPS Module TX	*D7 (GPIO5)*	*GPS → ESP8266 (UART RX)*
GPS Module RX	*D8 (GPIO4)*	*ESP8266 → GPS (optional)*
Piezo Buzzer (+)	*D5 (GPIO14)*	*Digital Output*
Piezo Buzzer (–)	*GND*	*Ground*
GPS VCC	*Vin*	*Power Supply*
GPS GND	*GND*	*Ground*

5.4. POWER CONSIDERATIONS:

- ***ESP8266:*** 3.3V (can be powered by 5V USB via onboard regulator)
- ***GPS Module:*** Typically supports both 3.3V and 5V
- ***Buzzer:*** Works between 3V and 5V
- ***Battery Options:***
    - Lithium Polymer (LiPo) 3.7V 1000mAh+
    - 18650 Rechargeable Battery with Boost Converter (to 5V)

5.5. USE CASE APPLICATIONS:

- Pet trackers
- Vehicle keys or luggage
- Child tracking
- Tool or equipment tracking
- Wallets or small valuable items

CHAPTER 6
RESULTS AND DISCUSSIONS

This chapter presents the outcomes observed during the implementation and testing of the GPS-based tracking system. The results highlight the effectiveness, accuracy, and practicality of the proposed system in various real-world conditions.

6.1. KEY BENEFITS OF THE PROPOSED SYSTEM:

1. ***Real-Time Location Tracking:***
   The integrated GPS module provides continuous and accurate location updates, allowing users to track the exact position of the tagged object in real-time. This is particularly useful for tracking moving objects such as pets, bicycles, or luggage during transit.

2. ***Remote Monitoring and Control:***
   Through the use of the ESP8266's Wi-Fi capability, the system can transmit data over the internet to a mobile device. Users can access the location information remotely, eliminating the need for physical proximity to the device.

3. ***Physical Identification with Buzzer Alert:***

   A built-in piezo buzzer can be activated remotely, helping users locate their item even when GPS data may not be precise (e.g., indoors). This feature is especially beneficial for locating misplaced keys, wallets, or bags in confined areas.

4. ***Cost-Effective Design:***
   The system is developed using low-cost, readily available electronic components. This makes it affordable and easy to replicate or mass-produce, which is ideal for students, DIY developers, or small businesses.

5. ***Portability and Compact Size:***
   The lightweight and compact design allows the device to be easily attached to any object without affecting its usability. This enables a wide range of applications from daily use to professional tracking systems.

6. ***Low Power Consumption:***
   The device is optimized for energy efficiency, offering several hours of operation on a standard lithium polymer (LiPo) battery. Power-saving techniques such as sleep modes and intermittent GPS readings can further extend battery life.

7. ***Scalability and Flexibility:***
   The system can be easily extended with additional features like motion sensors, geofencing alerts, cloud integration, or GSM fallback modules. This makes it suitable for both personal and industrial applications.

8. ***Multi-Scenario Compatibility:***
   The AirTag system can be used in various scenarios, including child safety, pet tracking, vehicle anti-theft systems, and tool or equipment management in industrial environments.

6.2. USE CASE TESTING:

The system was tested under various scenarios to assess its performance in different environments and use cases.

1. *Urban Outdoor Environment*

In this scenario, the device was placed in a moving vehicle in an urban outdoor environment. The system provided real-time GPS updates with accurate geofencing alerts. Despite this, the system performed reliably for real-time tracking and alert notifications.

2. *Indoor Environment*

When tested inside a building with poor satellite visibility, the device experienced a reduction in location accuracy, and occasional delays were noted in the GPS updates.

3. Manual Location Retrieval

The manual location retrieval feature was tested by triggering the piezo buzzer through the app in a crowded area. The buzzer responded instantly, providing effective device localization in environments like parking lots or within the home. This feature proved highly useful for small-radius searches where immediate attention is required.

4. *Multi-Device Scalability*

Multiple trackers were operated simultaneously through the same mobile application. The system handled multi-device tracking with no data loss or application lag, confirming that the solution is scalable for tracking multiple devices at once. This feature is valuable for larger deployments, where several assets or individuals need to be monitored concurrently.

6.3. CODE SNIPPETS:

Importing libraries and connecting with blynk template id:

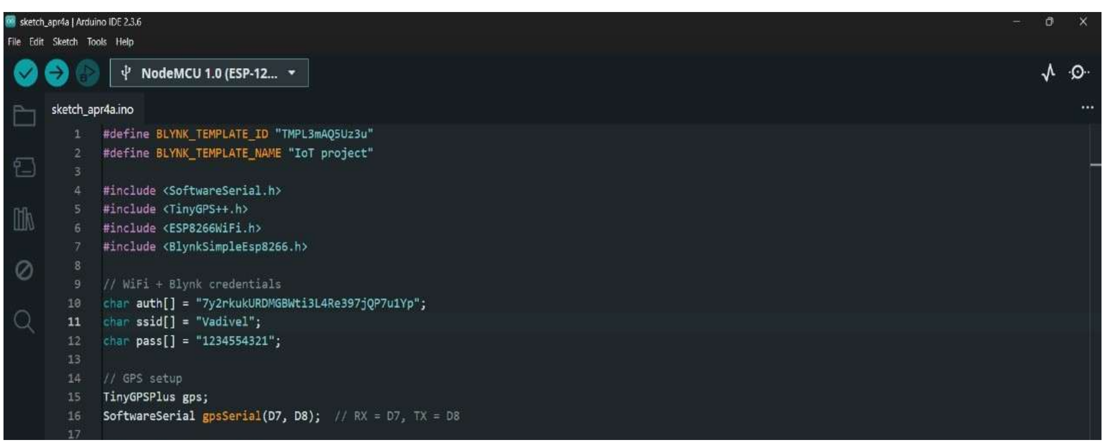

Figure 6.1. Code Snippet 1

To turn on and turn off the buzzer:

Figure 6.2. Code Snippet 2

Searching the coordinates and buzzing the device:

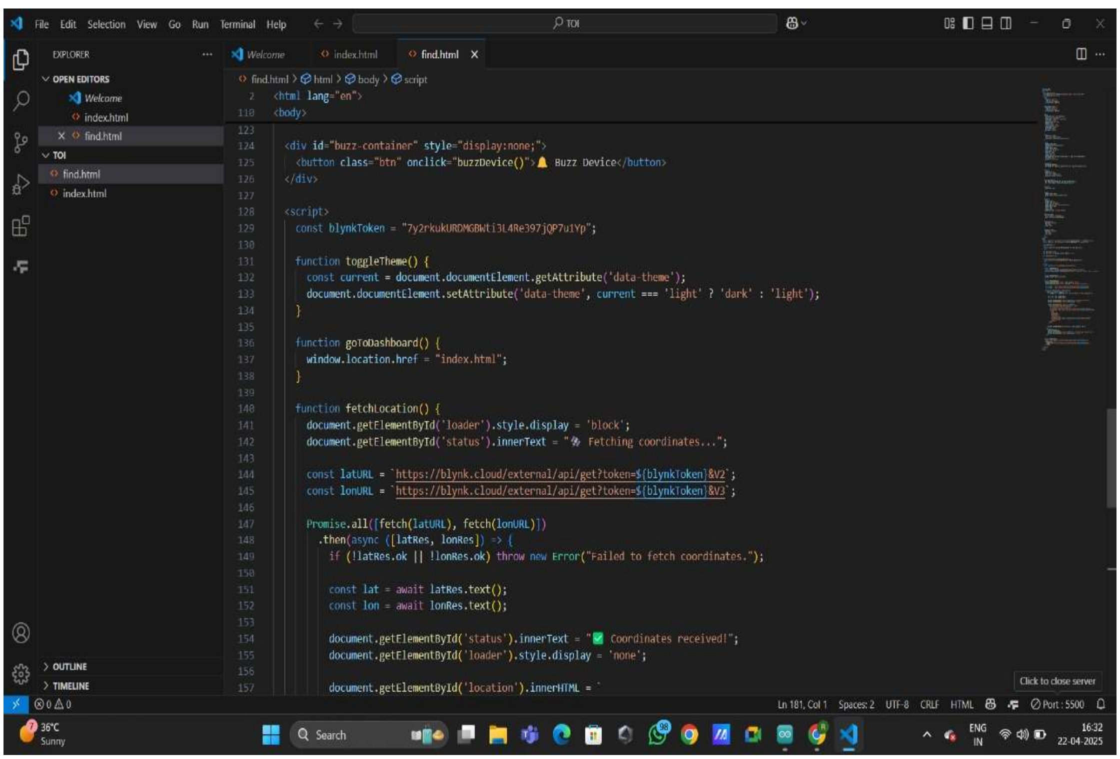

Figure 6.3. Code Snippet 3

6.4. RESULTS

Frontend of buzzNfind:

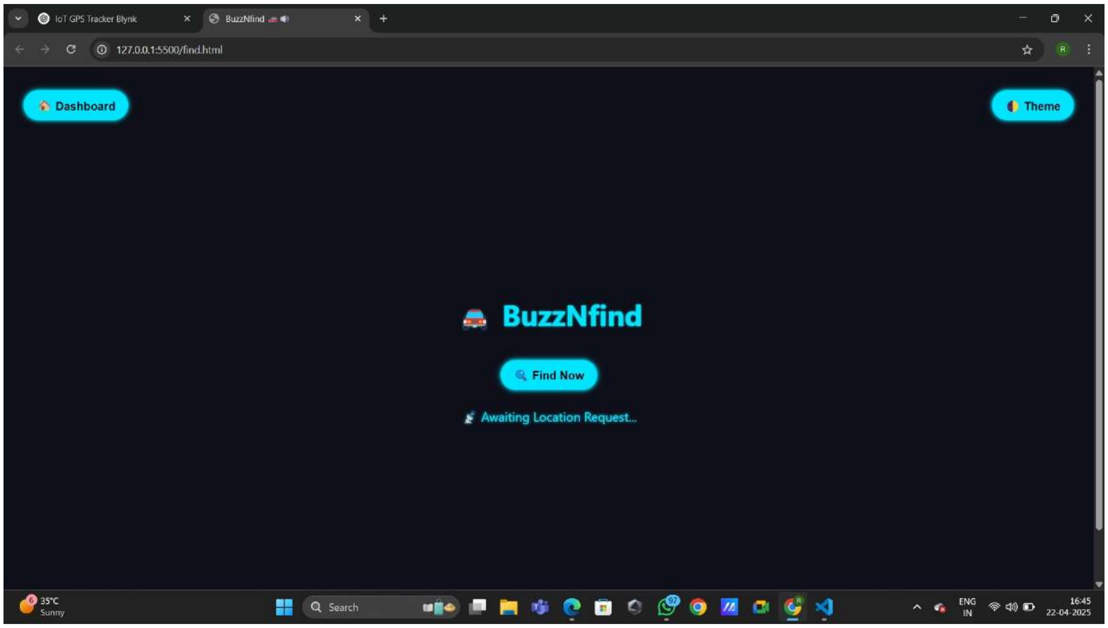

Figure 6.4. Frontend page

Current latitude and longitude mapped:

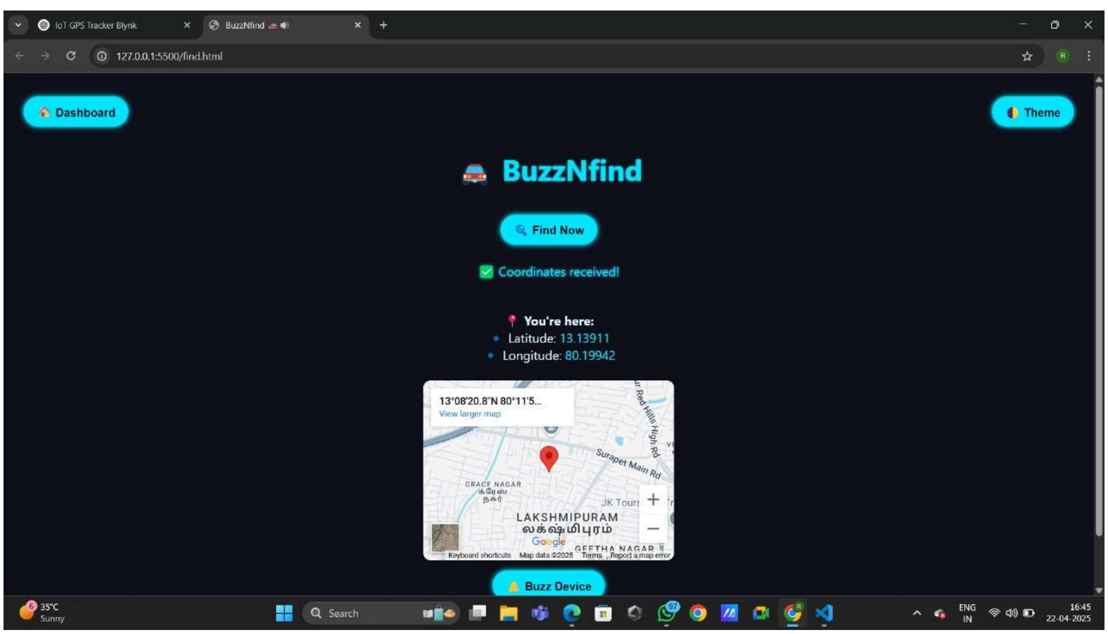

Figure 6.5. Frontend page after fetching coordinates

Output from GPS Tracking:

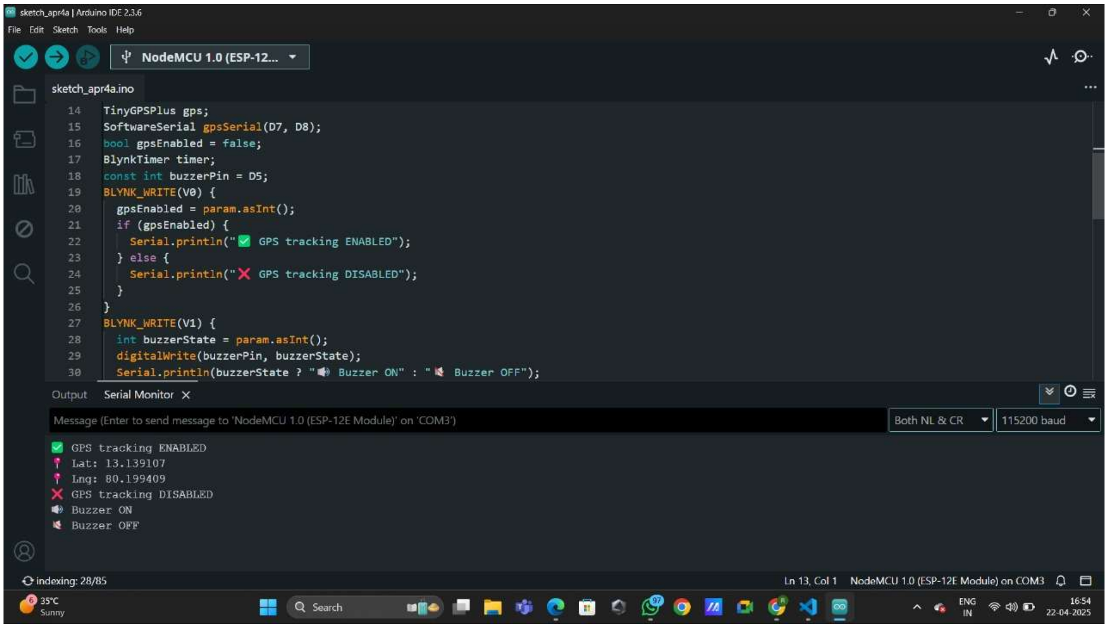

Figure 6.6. Sketch and serial monitor

Blynk web dashboard:

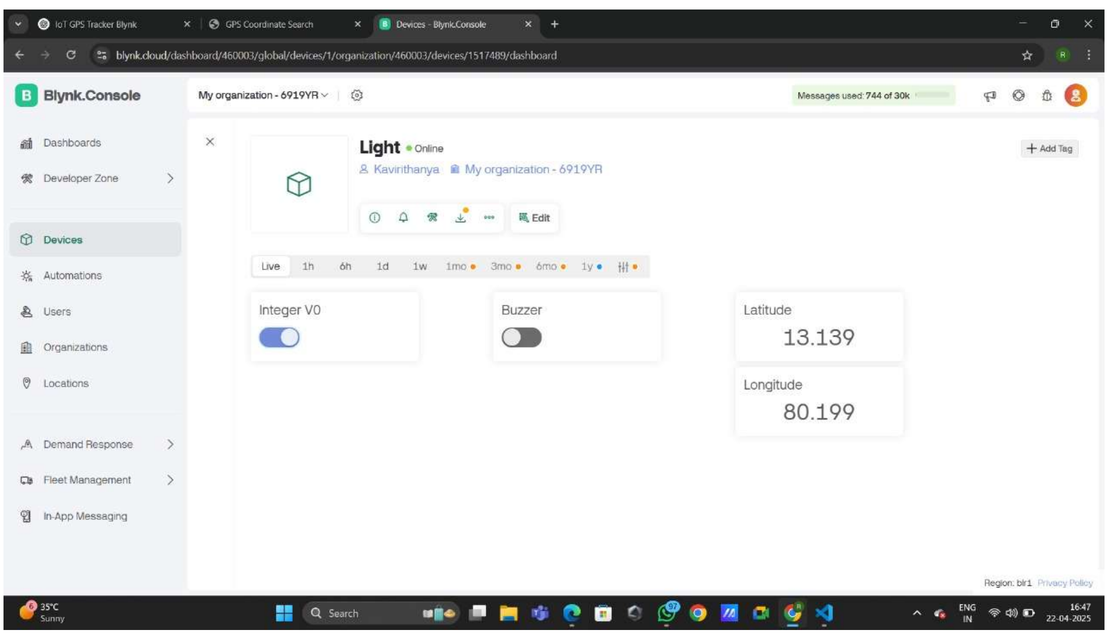

Figure 6.7. Working of blynk webdashboard

Blynk IOT Mobile App:

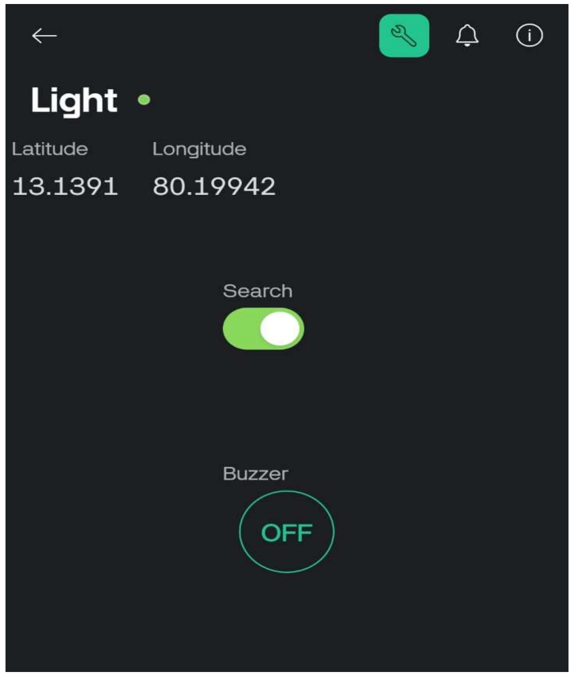

Figure 6.8. Mobile App Dashboard interface

Serial Monitor Output

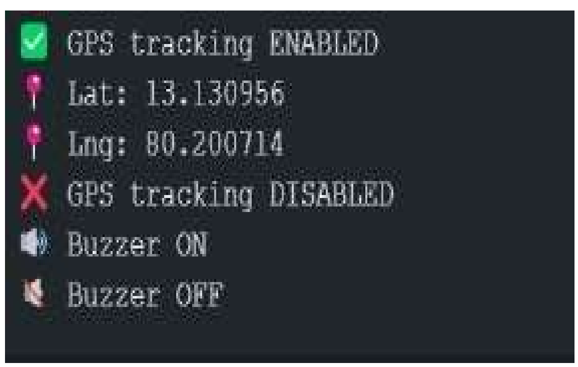

Figure 6.9. Serial monitor output

CHAPTER 7
CONCLUSION AND FUTURE ENHANCEMENTS

7.1. CONCLUSION:

This project successfully demonstrates the design and implementation of a functional, low-cost GPS-based tracking system powered by IoT technologies. At its core, the system utilizes the ESP8266 NodeMCU microcontroller and NEO-6M GPS module to retrieve precise real-time geographic coordinates. These include the device’s latitude and longitude, which are parsed and transmitted wirelessly via Wi-Fi using the Blynk IoT platform. The retrieved data is made accessible through a custom-built web interface, allowing users to view the device’s live location on an embedded Google Map. With a single click on the “Find Now” button, the system initiates tracking and updates the position visually, providing a seamless user experience. To enhance practical utility, the system also includes a remotely triggered piezo buzzer, activated through the “Buzz Device” button on the frontend. This enables users to audibly locate the device when it's within close physical range, such as inside a bag or drawer. The entire communication and control process is wireless, removing the need for physical connections between modules. The hardware is programmed using the Arduino IDE, making the system both highly customizable and accessible to developers and students. The use of open-source libraries like TinyGPS++ and Blynk enhances flexibility and ensures future scalability. Due to its compact form factor, affordable components, and cross-platform accessibility (via web or mobile), this solution proves suitable for a wide range of real-world applications — including asset monitoring, child or elderly tracking, vehicle safety, and personal security.

In conclusion, BuzzNfind offers a responsive, scalable, and intuitive alternative to commercial tracking devices, combining reliability, accessibility, and customization in one cohesive solution.

7.2. FUTURE SCOPE:

This GPS-based tracking system has strong potential for future enhancements that can significantly increase its versatility and efficiency. One key addition could be geofencing, which would allow users to set virtual boundaries and receive alerts if the device moves beyond them. Integrating data logging capabilities would enable storage and analysis of historical location data for trend tracking and reporting. A dedicated mobile application could further improve user accessibility and real-time interaction. To extend its usability in remote or rural areas, upgrading to LoRa or 4G communication modules would ensure better connectivity and longer transmission ranges. Incorporating power optimization techniques, such as sleep modes and energy-efficient components, can reduce power consumption. Adding solar charging capabilities would support sustainable, long-term outdoor deployment. Enhancing the system’s security with encrypted data transfer and user authentication can protect sensitive information from unauthorized access. These developments would not only make the system more robust but also suitable for industrial, commercial, and safety-critical applications.

REFERENCES

1. Adafruit, Piezo Buzzer – Basic. Adafruit Industries, 2020. [Online]. Available: https://www.adafruit.com/product/160
2. Arduino, TinyGPS++ Library Documentation. Arduino Project Hub, 2021. [Online]. Available: https://github.com/mikalhart/TinyGPSPlus
3. Blynk, Blynk HTTP RESTful API Documentation. Blynk, 2022. [Online]. Available: https://docs.blynk.io/en/blynk.cloud/http-restful-api
4. Blynk, Getting Started with Blynk. Blynk, 2021. [Online]. Available: https://docs.blynk.io/en/getting-started/
5. Espressif Systems, Getting Started with ESP8266 on Arduino IDE. Espressif, 2021. [Online]. Available: https://docs.espressif.com/projects/arduino-esp8266/en/latest/
6. G. C. Martins et al., "Energy-Efficient Low-Power Circuit Techniques for Wireless Energy and Data Transfer in IoT Sensor Nodes," *arXiv preprint*, arXiv:1704.08910, 2017. [Online]. Available: https://arxiv.org/abs/1704.08910
7. H. Okuhara et al., "An Energy-Efficient Low-Voltage Swing Transceiver for mW-Range IoT End-Nodes," *arXiv preprint*, arXiv:2010.04566, 2020. [Online]. Available: https://arxiv.org/abs/2010.04566
8. L. Zhao, "Battery Power Optimization Techniques for IoT Systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 3244–3251, May 2019, doi: 10.1109/TII.2018.2879293
9. M. Mohan et al., "Design and Implementation of Real Time Vehicle Tracking System Using GPS and GSM," *International Journal of Engineering Research & Technology*, vol. 9, no. 6, pp. 120–123, Jun. 2020
10. R. C. Lee et al., "Low Power Design for Internet of Things Devices," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 35–45, Jan. 2020, doi: 10.1109/JIOT.2019.2932555
11. SparkFun Electronics, Piezo Buzzer Hookup Guide. SparkFun, 2020. [Online]. Available: https://learn.sparkfun.com/tutorials/piezo-buzzer-hookup-guide
12. Ublox, NEO-6M GPS Module User Manual. Ublox, 2020. [Online]. Available: https://www.u-blox.com/en/product/neo-6-series

13. W. Wang et al., “Enabling Low-Power OFDM for IoT by Exploiting Asymmetric Clock Rates,” *arXiv preprint*, arXiv:2002.07155, 2020. [Online]. Available: https://arxiv.org/abs/2002.07155
14. Y. Sharma et al., “IoT-Based Smart Luggage Tracker Using GPS and GSM,” in *Proc. of the 2021 International Conference on Communication and Electronics Systems (ICCES)*, pp. 135–140, 2021, doi: 10.1109/ICCES51350.2021.9489149