BANGABANDHU SHEIKH MUJIBUR RAHMAN SCIENCE AND TECHNOLOGY UNIVERSITY, GOPALGANJ.

Department of Electrical and Electronic Engineering.

Project Report On

TracePoint

An Ultimate Solution of BSMRSTU Transportation

A report is submitted to complete the EEE309 course of 3rd year 1st semester.

SUBMITTED TO

Priyankar Biswas

Lecturer,

Department of EEE, BSMRSTU.

SUBMITTED BY

Md. Moniruzzaman Sojol
Student Id: 19EEE019
Session: 2019-20
Dept. of EEE, BSMRSTU.

Md. Aukil Ahmed
Student Id: 19EEE038
Session: 2019-20
Dept. of EEE, BSMRSTU.

Acknowledgment

First and foremost, we express our deepest gratitude to Almighty Allah for giving us the strength and perseverance to complete this project. We are incredibly grateful to our project supervisor, **Priyankar Biswas Sir**, Lecturer, Department of Electrical and Electronic Engineering at BSMRSTU, for his invaluable guidance, encouragement, and support throughout the project. His insightful suggestions and continuous feedback were instrumental in the successful completion of this work.

Abstract

"TracePoint" is a GPS-based tracking technology that aims to improve the efficiency and safety of BSMRSTU's transportation services. The project combines modern GPS technology to provide real-time position tracking, route planning, and data visualization for university buses, meeting the important demand for accurate and trustworthy location data. The system combines hardware and software tools, such as **NodeMCU ESP8266**, **Neo 6-M GPS** modules, and the **MERN** stack, to provide an easy-to-use user interface and reliable backend support.

TracePoint's major purpose is to deliver accurate, up-to-date positioning information that enhances university transportation management while also assuring student and staff safety and convenience. This study discusses the system's design, essential features, and prospective applications while also investigating its performance in real-world circumstances.

TracePoint adds substantial value to the campus by providing a scalable and effective solution to mobilit -y management concerns.

Table of Contents

	Page
Chapter 1	
1.1 Introduction	
1.2 Problem Definition	5-6
1.3 Objective	
Chapter 2	
2.1 Hardware Components	
2.2 Software Tools	7-11
Chapter 3	
3.1 Key Features	12
3.2 System Workflow	
Chapter 4	
4.1 Circuit Diagram	
4.2 TracePoint WebApp Overview	13-15
Chapter 5	
5.1 Further Development Scope	
5.2 Conclusion	16
5.3 Reference	

Chapter 1

1.1 Introduction

The "Trace Point" project is an innovative solution designed to leverage GPS technology for advanced tracking and location services. In today's fast-paced and interconnected world, the need for precise and real-time location data has become increasingly essential across various industries. "Trace Point" addresses this demand by providing a comprehensive GPS tracking system that offers reliable, accurate, and up-to-date positional information.

This project report outlines the development and implementation of the "Trace Point" GPS tracking system, detailing its architecture, functionalities, and potential applications. The system is engineered to cater to diverse needs, including fleet management, asset tracking, personal safety, and logistics optimization, making it a versatile tool for enhancing operational efficiency and security.

By utilizing state-of-the-art GPS technology, "Trace Point" aims to set a new standard in tracking solutions, delivering unparalleled accuracy and user experience. This report will discuss the technical specifications, key features, and potential benefits of the system, alongside an analysis of its performance and reliability in real-world scenarios.

1.2 Problem Definition

In the modern world, the ability to track and monitor the location of assets, vehicles, and individuals in real time is critical for various sectors, including logistics, transportation, security, and personal safety. However, many existing GPS tracking systems are limited by several challenges, such as inaccurate location data, poor connectivity in remote areas, high operational costs, and lack of integration with other systems. These issues can lead to inefficiencies, increased risks, and a lack of reliable data for decision-making.

1.3 Objective

The "Trace Point" project aims to develop a state-of-the-art GPS tracking system that provides reliable, accurate, and real-time location data for a variety of applications. The specific objectives of this project are:

1. **Develop a Tracking System**: To design a GPS tracking system that offers high precision and accuracy in location data, ensuring reliable tracking of assets, vehicles, and individuals across different environments, including urban, rural, and remote areas.
2. **Enhance Real-Time Monitoring Capabilities**: To implement real-time data transmission and monitoring features, allowing users to receive up-to-date information on the location and movement of tracked entities, thus enabling proactive decision-making and response.
3. **Ensure Wide Coverage and Connectivity**: To create a system that maintains consistent connectivity and coverage, even in challenging or remote geographical locations, by utilizing advanced GPS technology and supplementary communication methods.
4. **Integrate with Existing Systems and Platforms**: To ensure that the "Trace Point" GPS tracking system can easily integrate with other software platforms and systems, providing seamless interoperability and enhancing the utility of the data collected.

5. **Optimize for Cost-Effectiveness and Efficiency**: To design a solution that is not only technologically advanced but also cost-effective, reducing the overall expense of implementing and maintaining the GPS tracking system for users.
6. **Improve Security and Data Privacy**: To implement robust security measures to protect sensitive location data and ensure user privacy, complying with relevant data protection regulations and standards.

2.1 Hardware Components

1) **NodeMCU ESP8266 WiFi Module:** The central unit that interfaces with the Neo GPS 6-M modules.

Figure: NodeMCU ESP8266 WiFi Module

The NodeMCU ESP8266 WiFi module is a low-cost microcontroller board with built-in WiFi capabilities, based on the ESP8266 chip. It's popular among hobbyists and developers for IoT (Internet of Things) projects due to its affordability, ease of use, and versatility. Here's how it works:

a) Microcontroller Core

- **ESP8266 Chip**: At the heart of the NodeMCU is the ESP8266 microcontroller, which integrates a full TCP/IP stack and a microcontroller unit (MCU) that can be programmed using various programming environments such as Arduino IDE, MicroPython, and Lua.
- **Processor**: It uses a 32-bit Tensilica L106 running at 80 MHz (or overclocked to 160 MHz), making it quite powerful for its size.

b) WiFi Connectivity

- **Integrated WiFi**: The ESP8266 has built-in WiFi, which means it can connect to WiFi networks without needing an external module. It supports 2.4 GHz WiFi (802.11 b/g/n).
- **Modes**: It can operate in three modes:
  - **Station Mode**: The ESP8266 connects to an existing WiFi network (like your home router) to access the internet or communicate with other devices.
  - **Access Point Mode**: It acts as a WiFi access point (hotspot) that other devices can connect to.
  - **Station + Access Point Mode**: It can act as both an access point and a station simultaneously, allowing it to connect to a network while also allowing other devices to connect to it.

c) GPIO (General Purpose Input/Output) Pins
The NodeMCU board provides several GPIO pins that can be used to connect sensors, actuators, and other peripherals. These pins can be programmed to act as digital inputs or outputs and support various communication protocols (I2C, SPI, UART).

d) Programming Environment

- **Arduino IDE**: The NodeMCU can be programmed using the Arduino IDE, which provides an easy-to-use interface for writing code. The ESP8266 core for Arduino provides libraries and functions to make it easy to write code for networking, GPIO control, etc.
- **MicroPython**: A lightweight version of Python designed for microcontrollers. It's an alternative to the Arduino IDE, suitable for users familiar with Python.
- **Lua**: The NodeMCU firmware was originally developed for Lua scripting language, and many users still use Lua to write code on the NodeMCU.

e) Power Supply
The NodeMCU board can be powered via a micro-USB connection or through the VIN pin with an external power source (typically 5V). The onboard voltage regulator converts this to the 3.3V needed by the ESP8266 chip.

f) USB-to-Serial Converter
The board includes a USB-to-serial converter chip (usually CH340 or CP2102), which allows it to be connected to a computer for programming and debugging. This makes it easy to upload code to the microcontroller and monitor serial output for debugging purposes.

g) Flash Memory
The ESP8266 has built-in flash memory, typically ranging from 512 KB to 4 MB, used to store the firmware and user programs. The size of the flash memory can vary based on the specific module or version of the NodeMCU.

2) GPS Module (Neo 6-M): Sends the captured GPS data to the backend server via HTTP requests

Figure: GPS Module (Neo 6-M)

The GPS module (Neo 6-M) is a compact and affordable GPS receiver that provides accurate positioning and time information. It is widely used in various applications, such as navigation, vehicle tracking, and personal projects involving location-based data. Here's how the Neo 6-M GPS module works:

a) Core Components of the Neo 6-M GPS Module

- **GPS Receiver**: The Neo 6-M module is a GPS receiver, which means it receives signals from GPS satellites to determine its location on Earth.
- **GPS Antenna**: Typically, a ceramic patch antenna is integrated into the module, optimized for receiving GPS signals. An external antenna can also be connected for better signal reception in areas with poor visibility.
- **Microcontroller Unit (MCU)**: Processes the satellite signals to compute the position (latitude, longitude, altitude), velocity, and precise time (UTC).
- **Backup Battery**: The module may come with a small battery to power an internal real-time clock (RTC) and keep the GPS data up to date, allowing for quicker GPS fixes on startup (known as hot starts).

b) How GPS Works

- **Global Positioning System (GPS)**: GPS is a network of about 24-30 satellites orbiting the Earth, constantly transmitting signals with their location and time. The Neo 6-M GPS module receives these signals to determine its location.
- **Triangulation/Trilateration**: To calculate its position, the Neo 6-M module needs signals from at least four satellites. The module uses the timing information from these signals to determine how far away each satellite is (based on how long the signal took to arrive). By knowing the distance from multiple satellites, it can calculate its position on the Earth's surface through a process called trilateration.

c) Signal Reception and Processing

- **Signal Reception**: The GPS antenna on the Neo 6-M receives signals from GPS satellites. Each signal contains information about the satellite's location and the time the signal was sent.
- **Time Difference Calculation**: The Neo 6-M GPS module calculates the time difference between when the signal was sent and when it was received. Since the speed of light is constant, this time difference helps determine the distance to each satellite.
- **Position Calculation**:

    1. Using the distance from at least four satellites, the Neo 6-M module calculates its exact position (latitude, longitude, and altitude) by solving a set of equations related to the position of the satellites and the receiver.
    2. The process involves using the known positions of the satellites (broadcast in their signals) and the calculated distances to the receiver.

- **Data Output**: The module processes the received signals and outputs data in the form of NMEA sentences (National Marine Electronics Association protocol), which are standardized text-based data sentences that include information like:

    1. **GGA** (Global Positioning System Fix Data): Provides the time, position, and fix-related data.
    2. **RMC** (Recommended Minimum Specific GNSS Data): Contains recommended minimum data for GPS, including time, date, position, course, and speed.
    3. **GSV** (Satellites in View): Information about the satellites currently visible to the receiver.

d) Connectivity and Communication

- **Serial Communication (UART)**: The Neo 6-M communicates with other devices, like microcontrollers (e.g., Arduino, Raspberry Pi), using serial communication (UART protocol). It transmits NMEA sentences at a configurable baud rate, commonly set to 9600 bps by default.
- **Interfaces**: Some modules may also support other communication interfaces like I2C or SPI for integrating with various host systems.

e) Power Supply

- **Operating Voltage**: The Neo 6-M GPS module typically operates at 3.3V to 5V. It's often powered via the VCC pin from a microcontroller or an external power supply.

f) Performance Features

- **Cold, Warm, and Hot Start**:

    - **Cold Start**: When the module is powered on with no previous data, it takes longer to acquire the initial fix because it needs to download satellite information (almanac and ephemeris data).
    - **Warm Start**: If the module has a valid almanac (satellite location data) but not an ephemeris (precise satellite location data), it can get a GPS fix faster than a cold start.

- **Hot Start**: When the module has both valid almanac and ephemeris data (such as from the backup battery), it can acquire a fix very quickly.

- **Accuracy**: The Neo 6-M GPS module can typically provide an accuracy of about 2.5 meters in ideal conditions.
- **Update Rate**: It usually provides updates at a rate of 1 Hz (once per second), though some configurations may allow higher update rates.

2.2 Software Tools

1) **Backend (Node.js with Express):** Handles incoming data from the client side and stores it in the MongoDB database.
2) **Database (MongoDB and Firebase):** Stores the tracking data from tracing device, ensuring persistence and easy retrieval.
3) **Frontend (React.js):** Provides a user-friendly interface for viewing the real-time location of buses on a map and other information on user side.

Chapter 3

3.1 Key Features

1. Real-time Location Tracking:

- Provides live tracking of devices and assets with high accuracy.
- Utilizes advanced GPS algorithms to ensure precise positioning.

2. Bus driver information

- TracePoint will provide admin-controlled bus driver information.

3. User friendly interface

- The web app interface of TracePoint is designed in such a way that anyone can easily use it.

3.2 System Workflow

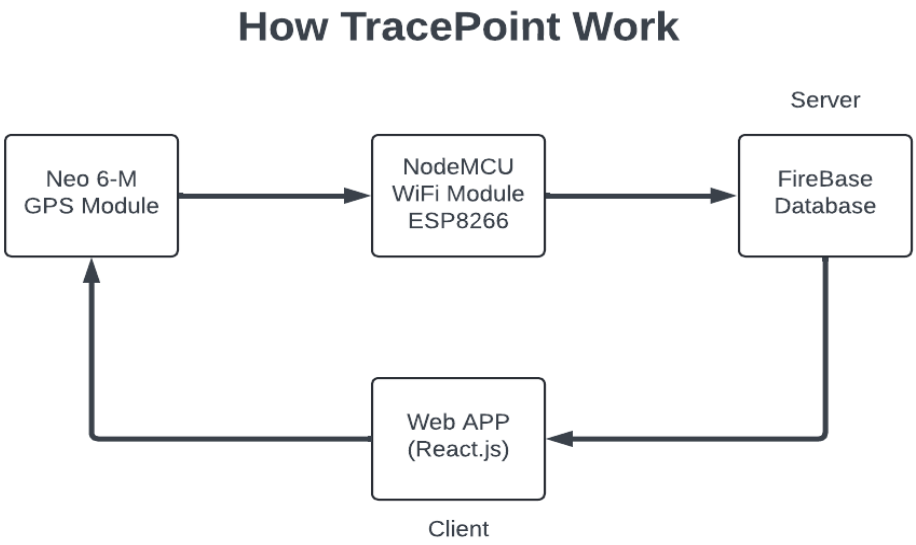

Figure: System workflow

4.1 Circuit Diagram and Code

Circuit Diagram:

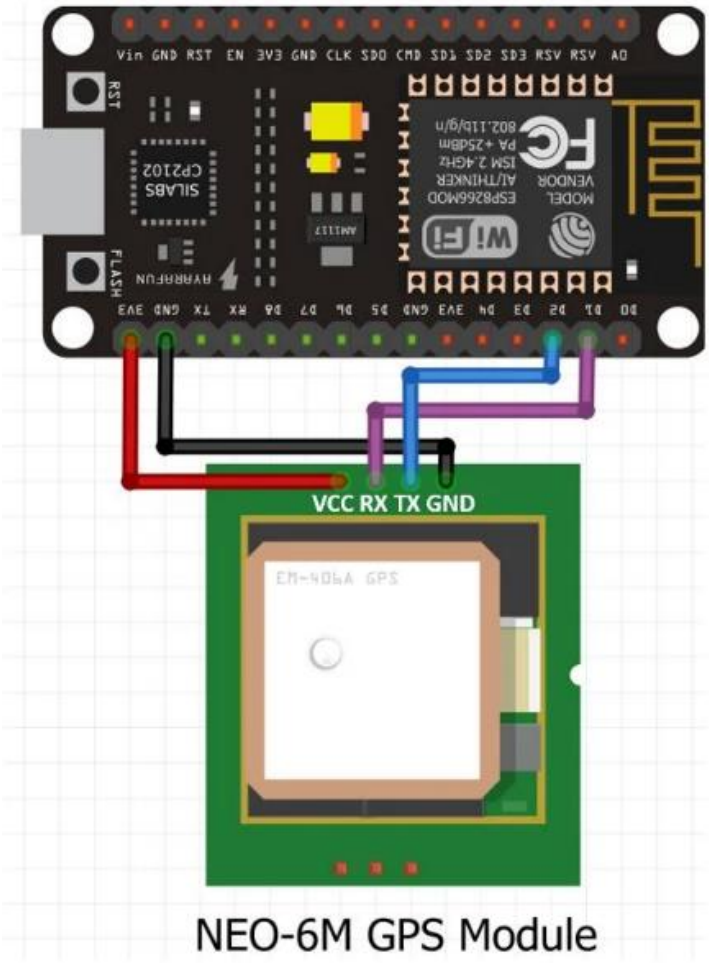

Figure: Circuit Diagram of Device

Code:

Please visit this link given bellow:

https://github.com/sojol4242/TracePoint-Web-APP-IoT-Based-GPS-tracker-

4.2 TracePoint WebApp Overview

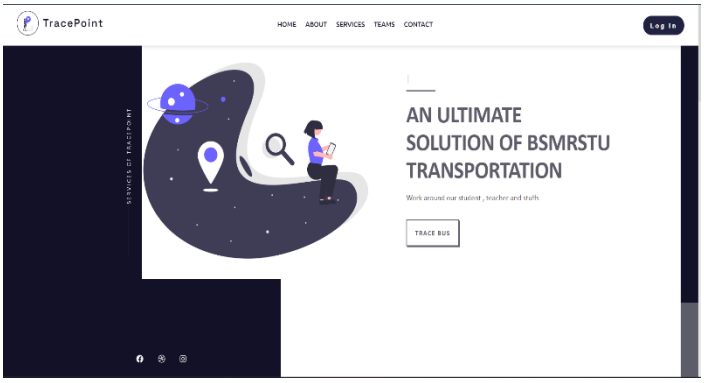

Figure: Home Page

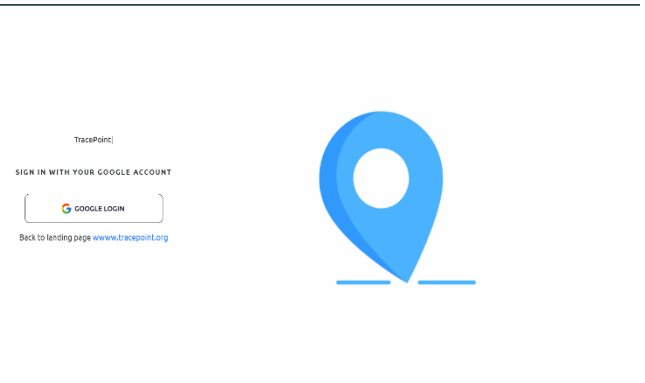

Figure: login page

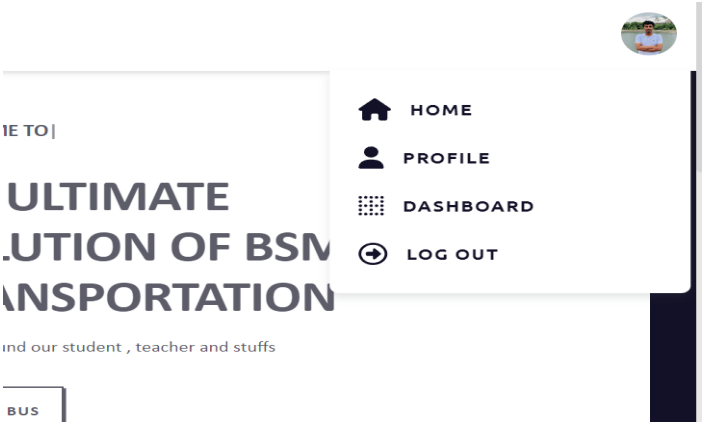

Figure: User logged in

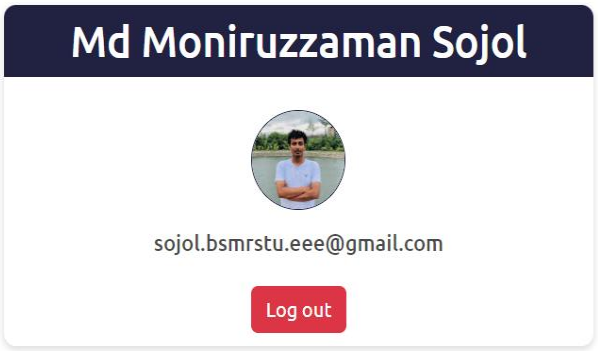

Figure: User profile

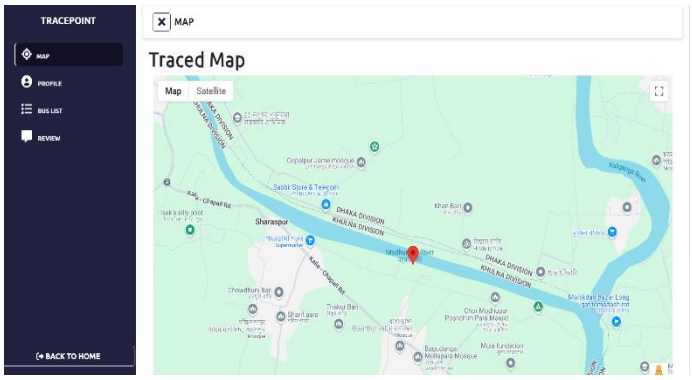

Figure: User's traced location

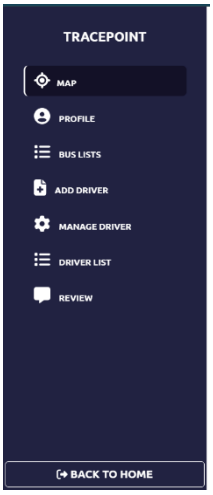

Figure: Admin Logged in

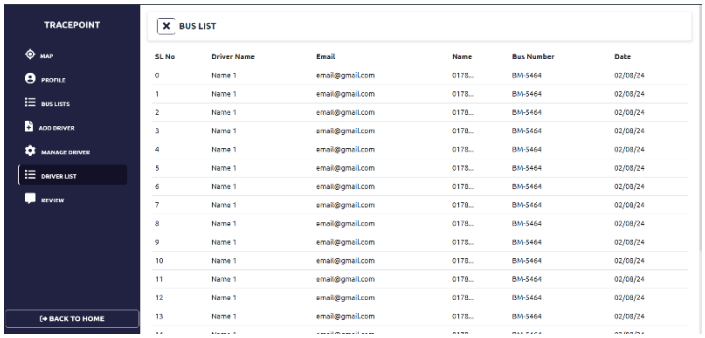

Figure: Bus driver and bus information

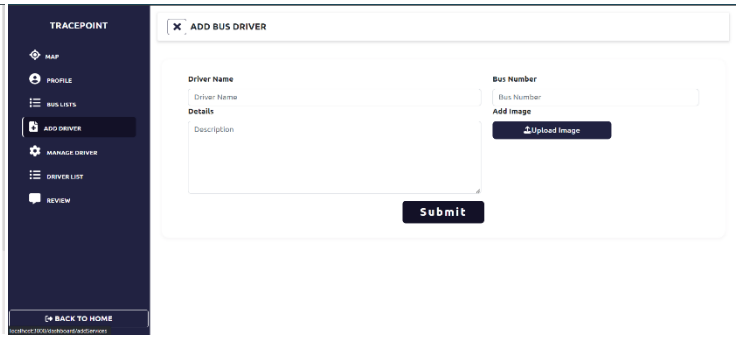

Figure: Admin can add new driver

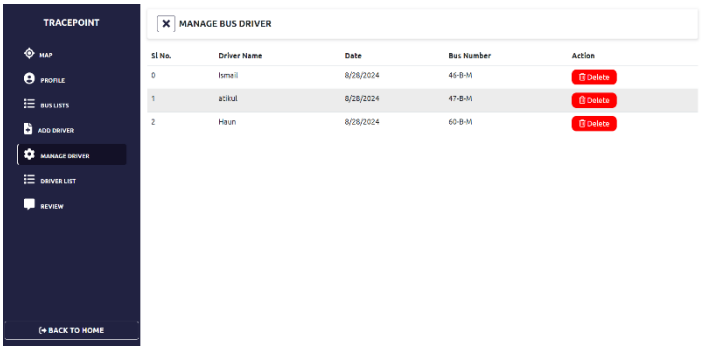

Figure: Admin Control Bus Driver Information

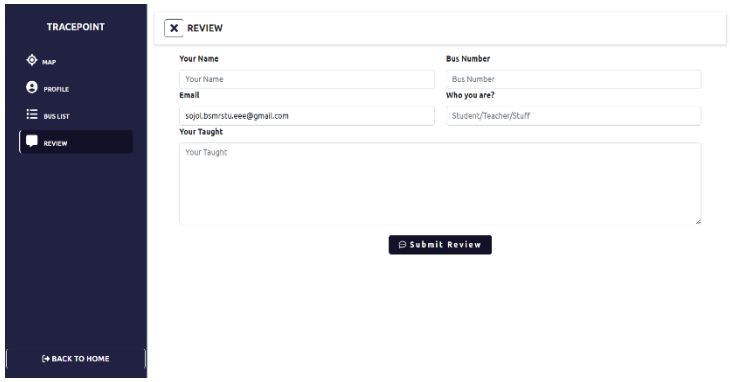

Figure: User can do review the app

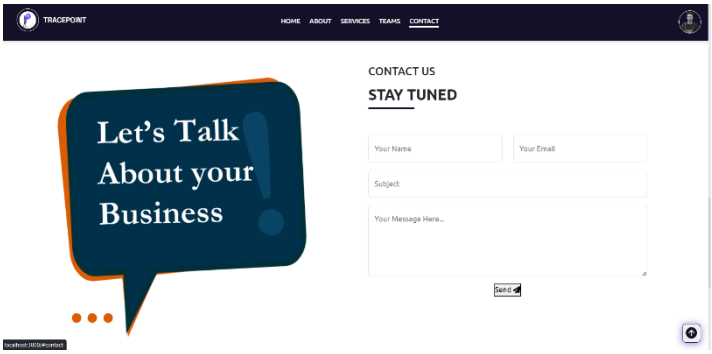

Figure: Contact us

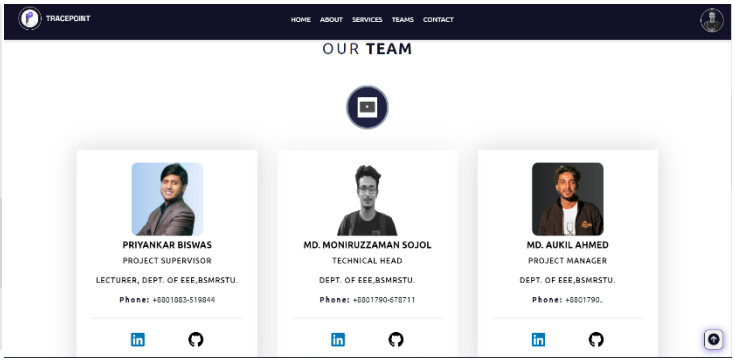

Figure: TracePoint team

5.1 Further Development Scope

1) Bus speed measurement.
2) Approximate trip time calculation.
3) Trip schedule alarm.
4) Fuel monitoring.
5) Store Data Schedule Based.

5.2 Conclusion

The "TracePoint" project provides a comprehensive answer to BSMRSTU's transportation difficulties, notably the real-time tracking of university vehicles. The project effectively offers exact position data using GPS technology and the MERN stack, increasing user safety and convenience. The system's capacity to deliver reliable, real-time updates on bus whereabouts marks a substantial advance in campus transportation management.

Further development might focus on adding functions such as bus speed measurement, travel duration computation, and fuel monitoring to make the system more robust and user-friendly. Overall, "TracePoint" is a promising instrument that has the potential to dramatically improve transportation services, with opportunities for future innovation and development.

5.3 References:

1) *Ashokan, M. (2023). Campus Connect - A College Bus Tracking System. International Journal for Research in Applied Science and Engineering Technology, 11(5), 7435–7444. https://doi.org/10.22214/ijraset.2023.53417*
2) *Google maps API implementation on IOT platform for tracking an object using GPS*
3) *Arduino Documentation - "ESP8266 NodeMCU WiFi Module." Available at: https://www.arduino.cc/en/Guide/ESP8266*
4) *u-blox - "NEO-6M GPS Module Data Sheet." Available at: https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_(GPS.G6-HW-09005).pdf*
5) *MongoDB Documentation - "Introduction to MongoDB." Available at: https://docs.mongodb.com/manual/introduction/*
6) *React.js Documentation - "React: A JavaScript Library for Building User Interfaces." Available at: https://reactjs.org/docs/getting-started.html*
7) *Express.js Documentation - "Express - Node.js Web Application Framework." Available at: https://expressjs.com/*
8) *GPS.gov - "Global Positioning System (GPS)." Available at: https://www.gps.gov/systems/gps/*
9) *Firebase Documentation - "Firebase Realtime Database." Available at: https://firebase.google.com/docs/database This reference covers the Firebase Realtime Database.*