Smart College Bus Tracking System

Aditi Nikam, Shruti Balekundri, Amna Bijapuri, Vishal Kadalagi
Department of Computer Science and Engineering
KLE Dr. M.S.Sheshgiri College of Engineering and Technology, Belagavi - 590 001
Belagavi, Karnataka, India
Email: {02fe23bcs416, 02fe23bcs425, 02fe23bcs427, 02fe23bcs431}@kletech.ac.in
Under Guidance of Miss. Madhurani B S - madhuranishiddibhavi.mss@kletech.ac.in

***Abstract*—The Smart College Bus Tracking System is a cutting-edge solution meant to enhance the transportation experience for students and college administrators. It enables students to locate their buses in real-time on a mobile app, allowing them to schedule their lives more efficiently and minimize waiting times. Administrators can track bus movements, facilitating smoother operations and more efficient route management. The system is meant to make the daily commute convenient and more reliable.**

The project employs GPS technology to give precise bus location updates and is integrated with Google Maps for real-time route visualization. Developed using Flutter for the frontend and Firebase for the backend, the system provides a clean, user-friendly interface that guarantees smooth interaction for everyone. Students are able to see their buses' current location, and administrators are able to make sure buses stick to their routes.

This system also resolves typical transport issues, like delays and route uncertainties, through the provision of timely and precise information. With its user-friendly design and real-time tracking functionality, it is a vital aid for colleges to enhance convenience as well as general transport management.

***Index Terms*—Smart Bus Tracking, Flutter, Firebase, Real-Time Tracking, API Limitations**

I. Introduction

Effective and efficient transportation is an important part of any educational facility. The Smart College Bus Tracking System is aimed at transforming college transportation. With real-time bus tracking, this system does away with the uncertainty of bus arrival times and routes, providing students with a hassle-free commuting experience. Through a simple-to-use mobile app, students can track the live location of their bus, saving time on unnecessary waiting and enhancing their daily routine.

This system makes use of advanced GPS technology and is connected with Google Maps to provide precise route and location information. It was built with Flutter for the front end and Firebase for the back end, giving it a stylish, responsive, and interactive UI. The bus routes and operations can be efficiently managed by the administrators, and the entire efficiency of the college transport system increases.

Aside from enhancing convenience, this system maximizes safety and transparency in college transport. Through real-time tracking, administrators are assured that buses operate on assigned routes, and students are assured of their travels as their transport is being monitored. The system also maximizes resource use through the insights it provides regarding bus usage patterns, and scheduling and routing planning can thus be maximized. Through the integration of technology with daily travel, this solution elevates conventional college transport to a technological, efficient, and user-friendly mode.

II. System Architecture

Key Components of the Architecture

- **User Interfaces (Frontend):**
  Student Module: Mobile application (Flutter) for bus location tracking, viewing schedules, and getting updates.
  Driver Module: Mobile application (Flutter) for drivers to send real-time location through GPS.
  Admin Module: Web dashboard for route planning, bus tracking, and schedule updating.

- **GPS and Data Collection**:
  A GPS-capable mobile device (driver's phone) is installed in each bus.
  The mobile application collects real-time location data and sends it to the cloud.

- **Backend System (Firebase Cloud Services)**:
  Firebase Realtime Database stores and handles bus locations, schedules, and user information.
  Google Maps API is used for route visualization and geolocation tracking.
  Authentication is managed through Firebase for secure access control (students, admins, and drivers).

- **Data Processing Communication**:
  Driver's mobile app sends GPS data to Firebase in real-time.
  Student's app retrieves bus location updates from Firebase every few seconds.
  Admin dashboard offers bus tracking and operational insights.

- **Security Data Privacy**:
  End-to-End Encryption for bus locations and user profiles, which are sensitive data.
  Role-Based Access Control (RBAC) provides drivers, students, and admins with varying levels of access.
  Adherence to Privacy Laws to ensure user information from unauthorized access.

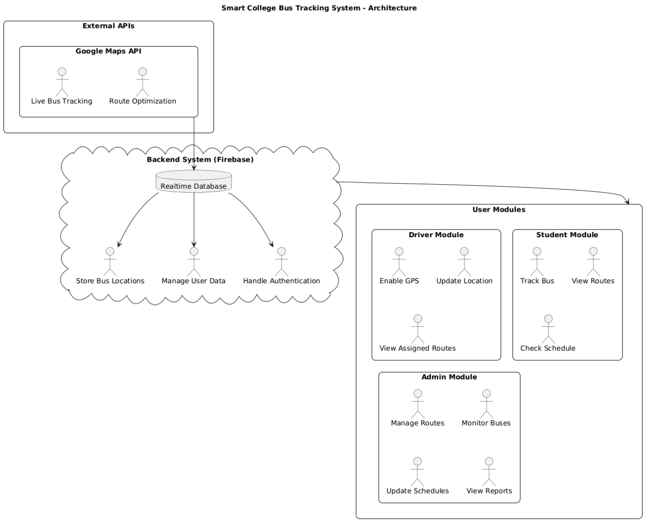

Fig. 1: System Architecture

Workflow:

1. Student opens the Mobile App (Flutter) → Logs in and uses bus tracking features.
2. Student chooses "Track Bus" → Requests real-time bus location from the Backend System (Firebase).
3. Backend retrieves the current GPS location → Reads data from the Realtime Database.
4. Driver turns on GPS updates location → The app takes the real-time position of the bus and uploads it to Firebase.
5. Backend updates the location in the database → Maintains real-time sync with the student's app and admin dashboard.
6. Google Maps API retrieves the live route → Shows the bus location best route based on traffic.
7. Student sees real-time bus tracking on the app → Location is updated dynamically as the bus travels.
8. Admin views the Dashboard (Web) → Tracks bus locations, allocates routes, and schedules.
9. Admin updates routes or schedules if necessary → Reflects in the database and gets pushed to students.
10. Internet Cloud Communication guarantees smooth data flow → Syncs Firebase, Google Maps API, Admin Dashboard, and Mobile App.

III. Implementation

A. Frontend Development - Flutter

The frontend of the Smart College Bus Tracking System is developed with Flutter, an open-source Google UI software development toolkit. Flutter provides cross-platform development for Android and iOS, providing a unified user experience.

Key Features

- **Real-time Bus Tracking**: Retrieves and shows current GPS positions of buses.
- **Google Maps Integration**: Indicates bus routes, route of arrival, and alternative routes.
- **User Authentication**: Uses Firebase authentication for students, drivers, and admins.
- **Responsive UI**: Maintains Material Design principles for usability across various screen sizes.
- **Dynamic Scheduling**: Enables students to see bus schedules and route updates in real time.

Techical Implementation:

1. State Management: Employs Provider/Riverpod for effective state management.
2. Google Maps SDK: Integrated for real-time bus tracking and route visualization.
3. REST API Calls: Retrieves live bus information from the Firebase backend.
4. Location Services: Utilizes Flutter Geolocator package to monitor GPS coordinates.

B. Backend Development - Firebase

The backend infrastructure is serverless, driven by Firebase, providing real-time data synchronization with low latency.

Key Features

- **Realtime Database**: Maintains live GPS coordinates, routes, and user information.
- **Authentication**:Secure student, driver, and admin login through Firebase Auth.
- **Cloud Functions**: Automates route updates, geofencing notifications, and data synchronization.
- **Storage**:: Maintains logs of bus movements.

Techical Implementation:

1. Firebase Realtime Database: Provides instant data updates.
2. Cloud Firestore: Manages other metadata, such as student reviews and reports.

C. API Integration

The system connects with multiple APIs to enhance efficiency and accuracy.

API's Used

- **Google Maps API**: Offers real-time location tracking and route optimization.
- **Geolocation API**: Retrieves accurate GPS coordinates of buses.
- **Firebase SDK**: Manages authentication, database management, and cloud storage.

Techical Implementation:

1. REST API Requests: Retrieve live GPS data and update Firebase.
2. WebSockets: Facilitates real-time updating of bus locations.
3. GeoFencing API: Fires alerts when the bus arrives at a particular stop.

IV. Limitations

A. API Key Limitations

- The system currently employs a sample API key for testing instead of an API key that was bought or purchased.
- Therefore, the polyline formed between stops is a straight line and not an actual road route.
- The validate path functionality is not available since there are limitations imposed by APIs.

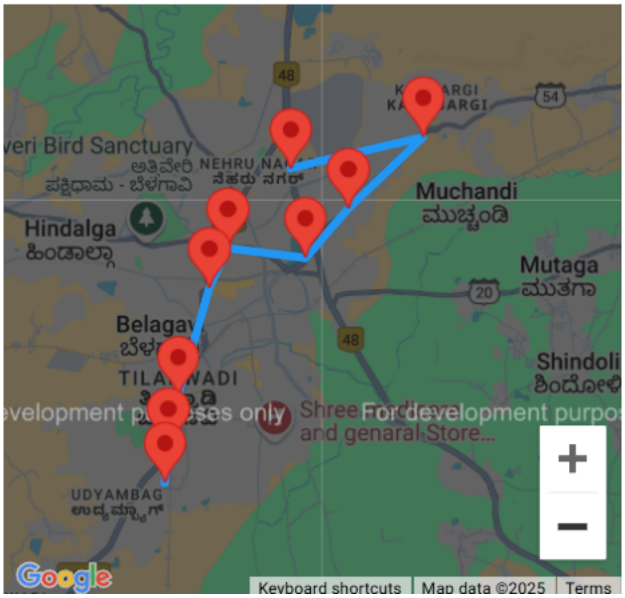

Fig. 2: Polyline Generated Using Sample API Key

B. *Current Deployment Restrictions*

- The system cannot presently deploy on a mobile application because of API constraints and unchecked configurations.
- Currently, the application is only available through localhost and web-based applications, so mobile users have to depend on the web version for tracking and updates.

V. Future Work

- Integration of a Purchased API Key: Upgrade from a sample API key to a licensed Google Maps API to enable accurate route mapping, real-time traffic information, and path validation.
- Mobile Application Deployment: Adjust configurations and fix API issues to activate the mobile application on Android and iOS platforms for real-time bus tracking.
- Automated Route Optimization: Apply AI-driven route optimization to modify bus routes in real-time considering traffic, road closures, and user demand.
- Push Notifications Alerts: Enable a real-time notification system for informing users regarding bus delays, schedule updates, and route diversion.
- Offline Mode Support: Create an offline mode support feature enabling the display of last known bus position and schedules without internet connectivity.
- User Feedback Analytics: Incorporate a student and driver feedback system to report problems and suggest improvements, as well as usage analytics for performance enhancement.
- College ID System Integration: Integrate QR code scanning for secure boarding of buses by students, with proper authentication and attendance tracking.

VI. Conclusion

The Smart College Bus Tracking System has revolutionized the interactions between college transportation services, students, drivers, and administrators. By offering real-time tracking, route planning, and feedback collection all on a single, easy-to-use platform, the system facilitates an easy and seamless traveling experience. Students enhance daily schedules with accurate bus locations, drivers experience smooth route assignments and positive feedback, and administrators have more control over operations, significantly reducing labor effort and maximizing overall efficiency. The system not only maximizes convenience but also fosters a safer, more integrated campus lifestyle, setting the bar for future transportation solutions.

Through the provision of real-time access to transit data, the system minimizes waiting and uncertainty, enabling improved time management and productivity for students. Route optimization and organized scheduling for drivers minimize wasteful delays and enhance overall service dependability. Administrators are able to track and analyze transportation trends, facilitating informed data-driven decisions to improve fleet efficiency and resource utilization. These elements combined provide an enhanced, more dependable, and more efficient means of transportation, further securing the system's status as a vital resource for campus transport.

Caution

Note: The API key employed here is a test key and not bought. Consequently, the polyline between stops will be displayed as a straight line rather than taking actual routes, and the validate path option is not available.

Moreover, because of API constraints, the system can't be implemented on a mobile app and can only be accessed currently through web or localhost. An active API key is necessary to facilitate precise route mapping, real-time traffic information, path verification, and complete mobile support for smooth functioning.

References

[1] Google Maps Platform, "Routes API," Google Developers: https://developers.google.com/maps/documentation/routes

[2] Firebase, "Realtime Database," Google Firebase Documentation https://firebase.google.com/docs/database

[3] Flutter, "Flutter Documentation," Google Developers https://flutter.dev/docs

[4] Geolocation API, "Geolocation and Positioning," MDN Web Docs https://developer.mozilla.org/en-US/docs/Web/API/Geolocation_API

[5] K. N. Sivaraman, "IoT-Based Real-Time Bus Tracking System," International Journal of Computer Applications, vol. 180, no. 30, pp. 25-30, 2018.

[6] M. Patel and S. Gupta, "A GPS-based Bus Tracking System Using Mobile Application," International Journal of Advanced Research in Computer Science, vol. 9, no. 2, pp. 45-50, 2019.

[7] H. Kim, J. Lee, and S. Park, "Smart Bus Tracking System Using GPS and Cloud Computing," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 5, pp. 3125-3135, 2021.

[8] A. Sharma and P. Mishra, "Enhancing Public Transportation with AI and IoT-Based Bus Tracking," IEEE Access, vol. 10, pp. 56745-56758, 2022.

[9] J. R. Smith, "A Review of Location-Based Services and Their Impact on Public Transport Efficiency," IEEE Internet of Things Journal, vol. 6, no. 8, pp. 1523-1535, 2020.