Leveraging GIS and Mobile Technology for Efficient Road Data Collection: A Case Study with GetMap

M.S. Saran[1*], Nisha Radhakrishnan[2], and P.P. Anjana[1]
[1]KSCSTE - National Transportation Planning and Research Centre, Trivandrum, Kerala,
[2]National Institute of Technology (Tiruchirapalli), Tamil Nadu
*Corresponding author email: mssaran6200243@kerala.gov.in

(Received on 29 January 2025; In final form on 14 August 2025)

DOI: https://doi.org/10.58825/jog.2025.19.2.221

Abstract: Data is the basic requirement in information derivation across all fields of study. And this is especially true for transportation engineering, where road and traffic-related data are essential for getting meaningful results. The traditional methods of road data collection involve extensive preparation, manual works, and use of paper forms. These methods are time-consuming and expensive. With the growth of information technology and the widespread availability of the internet, we have entered an era of real-time data capture and sharing. This study introduces a user-friendly mobile application, GetMap, designed for real-time road data collection and sharing. The Android based app records users travel path and also collects road related data and uploads the information to Firebase cloud server. App's key functionalities include recording travel tracks, adding road inventory and cross-sectional details, and marking points of interest with photographs. The app outputs are generated as KML files and Excel sheets, facilitating facile integration with GIS platforms. Getmap will be an effective tool for road data collection agencies like Public Works Departments (PWD), transportation planners, and road safety authorities.

Keywords: Android application; GetMap; location-based service, Geographic Information Systems (GIS), Firebase Cloud Storage

1. Introduction

Access to accurate data is vital for analysis and well-informed decision making. The ubiquity of internet enabled smartphones with built-in GPS capabilities has unleashed enormous potential for collecting geographically referenced data. This is especially relevant to transportation research. The accelerated growth of mobile navigation and urban services is one aspect of the digital transformation in contemporary cities (Seliverstov et al., 2017). As a result, integration with mobile technologies allows real-time monitoring and evaluation of citizen satisfaction.

Location based services (LBS) have been thought of as a highly intensive research and development field; especially after the US government ended its selective availability in May 2000. LBS, by and large are mobile computing systems that provides location and context specific information to users (Raper et al., 2007). Geographical Information System (GIS) plays an important role by supporting Location Based Services (Sadoun et al., 2007). Leveraging the capabilities of Geoinformatics and LBS, we have developed an Android-based mobile application designed for real-time, paperless field data collection. This app automates the creation of digital files required for further processing, addressing the challenges of traditional data collection methods. Developed specifically for research and academic purposes within our organization, the application streamlines the extensive process of road data collection, offering a practical and efficient solution to a complex problem.

Related work: The mobile information era is profoundly transforming both society and science. Location-Based Services (LBS), which utilize the location of mobile devices to provide a wide range of functionalities, have become increasingly prevalent. These applications range from ride-sharing platforms like Uber to entertainment experiences such as Pokémon Go and safety-oriented solutions like emergency alert systems. Introduced in the early 1990s, LBS gained prominence as a research focus in the early 2000s (Huang et al., 2018). While navigation and mobile guide applications remain dominant, the scope of LBS has expanded significantly to include real-time traffic updates and landmark-based navigation (Krisp and Keler, 2015). Crowdsourced platforms such as Waze have demonstrated the potential of LBS by providing real-time navigation and road condition updates (https://www.waze.com). Emerging trends in LBS include safety warning applications (Li et al., 2015), on-street parking availability checkers like Parkbob, and multimodal routing solutions (Bucher et al., 2017). In the educational domain, LBS applications are utilized for field-based learning (Sailer et al., 2016), while in public safety, they aid in disaster management and crime mapping (Choy et al., 2016; Jadhav et al., 2014 and Toledo et a1., 2017). Smartphones, with their built-in sensors, have enabled innovative research in transportation including travel survey applications (Abdulazim et a1., 2014 and Vlassenroot et al., 2015). For example, accelerometers have been used to measure road roughness, providing data analogous to bump integrators (Allaire and Hanson, 2017). These devices estimate roughness indices by analysing vehicle vibrations, contributing valuable data for transportation planning (Silyanov et al., 2020).

Building on these advancements, the GetMap application was developed as a tool to simplify and enhance field data collection. GetMap integrates seamlessly with GIS platforms, enabling efficient data analysis and decision-making. The addition of a web application for data visualization and sharing further distinguishes it from traditional methods and other LBS tools. This innovation fills a critical gap by offering an all-encompassing platform for real-time data collection, management, and utilization.

2. Methodology

Field data collection in transportation studies has traditionally relied on extensive manual efforts, including paper-based forms for recording and storing data. This approach is not only time-consuming and expensive but also prone to errors, such as mis-recording or misplacing critical information. The limitations of these methods highlight the need for innovative solutions that leverage modern technology to streamline data collection processes. The methodology involved in the development of the mobile app is given in Figure 1.

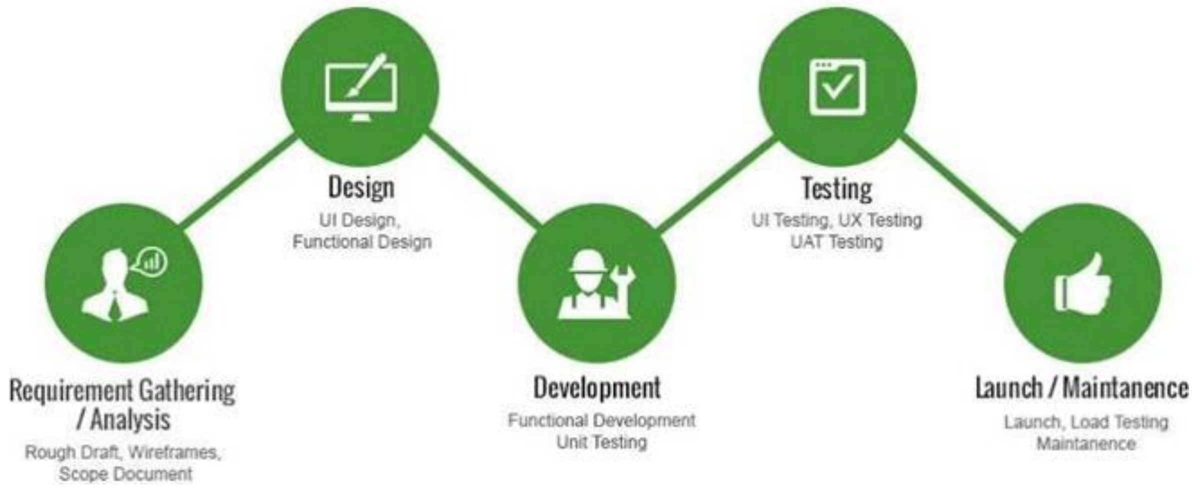


Figure 1. Flow Chart of Methodology

2.1 Development Process

The methodology for developing the **GetMap** mobile application involved several stages, as outlined below:

1. Requirement Gathering and Analysis
   a) Objective: Identify the key features and functionalities required for effective data collection.
   b) Process: Surveys and interviews with stakeholders, including transportation researchers, field survey teams, and public works officials, were conducted to understand user needs.
   c) Outcome: Key requirements identified included real-time geo-tagging, intuitive user interfaces, automated data storage, and compatibility with GIS platforms.

2. Design and App Development
   a) Platform Selection: The app was developed using Android Studio, chosen for its robust development environment and widespread adoption among smartphone users.
   b) User Interface (UI): A simple, user-friendly UI was designed to ensure accessibility for users with varying levels of technical expertise.
   c) Features:
      - Real-time path tracking using OpenStreetMap (OSM).
      - Data entry forms for inputting road inventory and cross-sectional details.
      - Capability to add points of interest with photographs and descriptions.
      - Storage of collected data in KML and Excel formats.
   d) Integration: The app uses the *Osmdroid* library for map interactions and *Firebase* for cloud storage and authentication.

3. Testing and Validation
   a) Initial Testing: The app was tested on various Android devices to evaluate functionality, user experience, and performance under different conditions.
   b) Field Testing: Data collection trials were conducted by staff and students on diverse road types to validate the app's accuracy and usability.
   c) Feedback Loop: Feedback from users was collected via surveys and used to refine the app, addressing any identified issues or usability concerns

2.2 Workflow and Outputs

The GetMap application follows a streamlined workflow:

1. Data Collection: Users record travel paths, road inventory, and cross-sectional details using the app.
2. Data Upload: Collected data is automatically uploaded to the *Firebase* cloud server.

3. Data Access: Administrators and stakeholders access the data via the web application for analysis and decision-making.
4. Integration with GIS: KML files can be imported into GIS platforms like QGIS or ArcGIS for advanced spatial analysis.

3.Overview

Getmap, an android based mobile application is introduced in this paper. The application enable accurate and efficient collection of road related data. Taking advantage of modern smartphones' capabilities, GetMap allows users to capture, store, and share road-related data such that field works based on traditional human labour are no more necessary. The application tracks the user's location and displays it in real-time over an Open Street Map (OSM). The main functions of GetMap are as follows: Tracking path recording: the entry of the exact routes including, but not limited to, where we visited while in the field surveying for transportation data; Road inventory giving includes variables such as road type, width, pavement material type and asset relating on track; Road cross-sectional data, which records measurement, dimensions and structure of the road at specific locations. The application can also be utilized to add points of interest along with pictures on the map, used as visual reference; and to export data in KML file or Excel. KML files being recorded on the mobile device are simply transferred to a computer and analyzed using compatible explanatory software, adding flexibility and performance to field data collection.

3.1 Developing the Android mobile app architecture

The general app architecture for gathering, processing, and storing GPS data using an Android smartphone is illustrated in Figure 2. The application retrieves GPS data from satellites and integrates it with maps provided by OpenStreetMap (OSM). Processed data is stored locally on the smartphone in KML and Excel formats for immediate access. The data is then transferred to a *Firebase* cloud server via the internet, ensuring secure storage. Users can manage and view the data through a web application, making it an efficient solution for road-related data collection for both public and professional use.

3.2 Android-Based Smart Phones

In the rapidly digitizing world, smartphones have become ubiquitous, with their usage and functionality expanding rapidly. The developed application was specifically designed for deployment on Android-based smartphones, one of the most widely used mobile operating systems for devices like smartphones, tablets, and other touch-enabled devices. Android is built on a modified version of the Linux kernel, offering a user interface that supports touch gestures such as tapping, swiping, pinching, and reverse pinching, along with a virtual keyboard for input. Users can easily download and install third-party applications through platforms like Google Play or Amazon Appstore. Figure 3 illustrates the layered architecture of the Android operating system. The Linux kernel forms the foundation, with APIs written in C, middleware, and libraries built on top. Above these layers is the application framework, which includes Java-compatible libraries essential for running application software. Within this framework, the Location Manager class provides access to the system's location services. These services periodically determine the device's geographic location and trigger application-specified intents when the device enters the proximity of predefined geographic locations. This functionality is utilized by the app to accurately track travel paths and record location details.

3.3 OpenStreetMap

OpenStreetMap (OSM) is a collaborative initiative created by a global community of volunteer mappers who contribute and maintain geographic data, including roads, traffic, railway stations, and more. It serves as a free, editable map, with its primary output being the geographic data that underpins the map. OSM is largely built using affordable mobile satellite navigation devices and leverages the widespread availability of global map data. This service was utilized as the base map for the developed application, offering users a reliable and versatile mapping platform. OSM empowers individuals to access and utilize map data in innovative and productive ways. It imposes no restrictions on the type of data submitted to its database, provided the information is accurate and respects copyright laws.

3.4 Android Studio

Android Studio, the official integrated development environment (IDE) for Google's Android operating system, is widely used for Android development. Its built-in emulators simplify the application testing process, eliminating the need for physical Android devices. The **GetMap** application was developed using Android Studio, utilizing its advanced tools and features to create an intuitive and visually appealing user interface (UI). The IDE's powerful editor enabled the design of a user-friendly UI that enhances the app's functionality. A screenshot of the development environment is presented in Figure 4.

3.5 Java Android Programming

Java is a preferred technology for building Android applications using managed code, which can be executed on mobile devices. For our app development, we utilized the Java programming language and the *Osmdroid* library on the Android Studio platform. To track user location changes, we employed the Android interface called *Location Listener*, which receives notifications from the Location Manager when the user's location changes. The on Location Changed method is automatically triggered when a location update occurs. Through this method, the app dynamically records the travel path as the user moves. Additionally, enhancements to the travel and image features will be implemented as part of the app development process, with similar functionality carried over to other applications

3.6 Firebase as Cloud Server

The app integrates Firebase Cloud as the backend database, utilizing several of its services including Authentication, Realtime Database, Cloud Firestore, and Cloud Storage. To facilitate user registration, we implemented Cloud Firestore to store new user details. Users can sign up by providing essential information like Name, Email, Mobile number, and Password, which is then stored in the Firestore Database. The login system uses Firebase Authentication, allowing registered users to sign in with their Email and Password. Additionally, users collecting field data can simultaneously upload and download various files, including KML files, Excel files, and photos, to and from Firebase Cloud Storage..

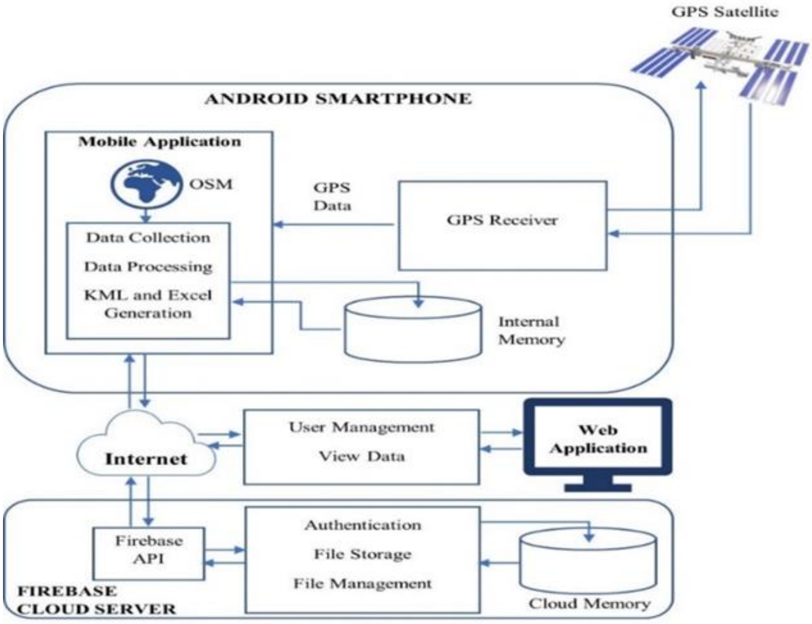


Figure 2. System Architectural development

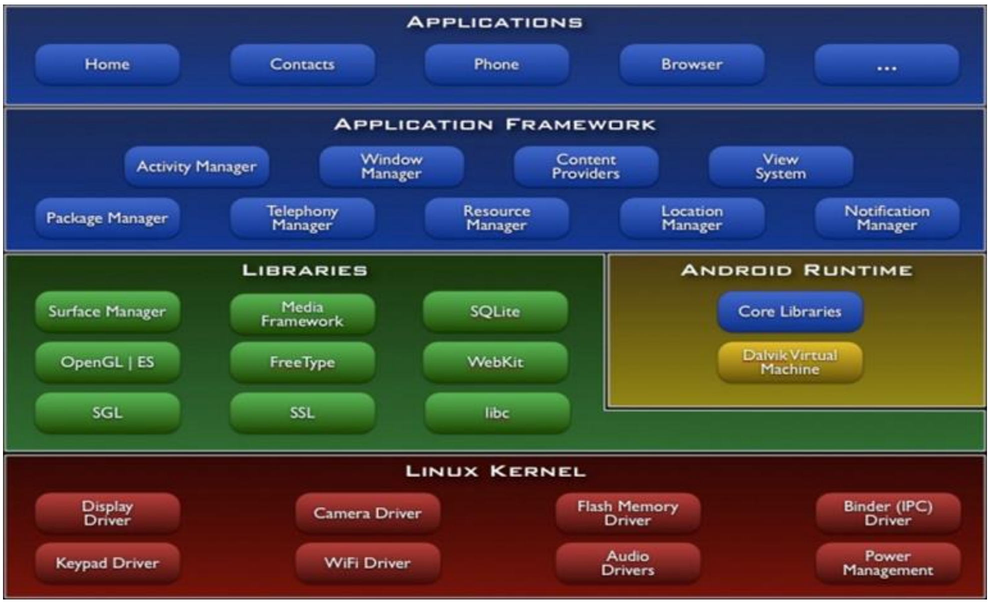


Figure 3. Android operating system layer
(Source: Smieh - Anatomy Physiology of an Android, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=20067152

4. Key Features and Functionalities of the App

The **GetMap** application was developed to streamline the process of road-related data collection and management. It offers a range of features designed to simplify tasks such as recording travel paths, capturing road inventory details, and documenting cross-sectional data. These functions make it an invaluable tool for transportation surveys and infrastructure projects, enabling efficient, accurate, and user-friendly data handling. Below is an overview of the app's primary functions:

4.1 App Interface

The GetMap application provides a user-friendly interface designed for efficient road data collection and travel path recording. It is easily downloadable on smartphones, targeting professionals in traffic and transportation projects while it will be made accessible to general users.

Upon first use, the app prompts for permissions to access location services and device storage. It features two primary screens: The Welcome Screen and the Map Screen. The Welcome Screen acts as a splash screen during app loading, transitioning to the login screen and then to the Map Screen. The Map Screen serves as the core interface, automatically displaying the user's current location on the map and project details. The app has six interactive buttons for various functions: (1) Global Track, (2) Start Record, (3) Stop Record, (4) My Location, (5) Add Details, and (6) My Project, along with (7) My Location Marker that indicates the user's live movement (Figure 5). These interactive buttons and the location marker are discussed in detail in the following sections.

A navigation bar displays the current project name as a subtitle, while additional fields provide real-time data such as track name, speed, chainage, distance travelled, and the number of available satellites. This intuitive design ensures users can seamlessly navigate and utilize the app's features for road data collection and management.

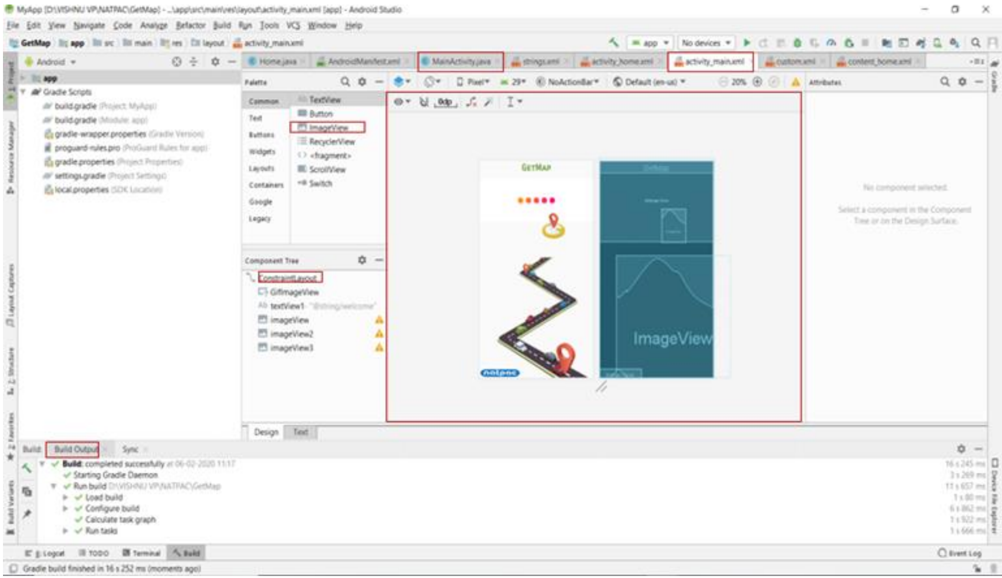


Figure 4. Android App development using Android Studio

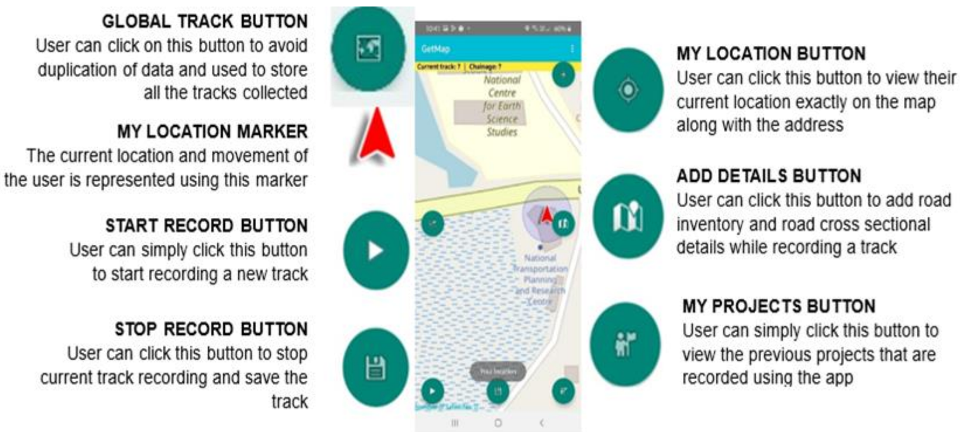


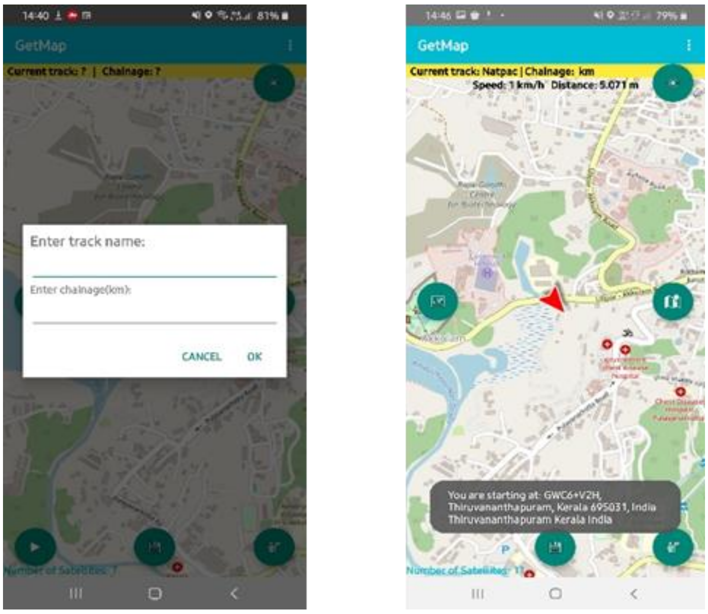
Figure 5. Map Screen of the app with the description of buttons

4.2 Record Track

To initiate track recording, the user clicks the "START RECORD" button, prompting the app to request the track name and chainage. Upon entering this information, the app displays the address of the starting point and begins recording the track. The track name is shown at the top of the map screen, alongside the speed and distance travelled from the starting point. The recorded track is visually represented as a coloured line on the map, with distinct colours assigned to different tracks. Recording can be stopped by clicking the "STOP RECORD" button, after which the track is saved to the current project. Figure 6 illustrates the app screen for entering the track name and the interface used during track recording.

"Global Track" feature has been added to prevent duplication of data. This option saves all tracks uploaded by users and are presented as a comprehensive dataset that anyone is free to view within the app. Users can select the track of their interest and request the same to the administrator. Upon approval by the administrator, the selected track can be downloaded and accessed within the app.

The "MY PROJECTS" button in the app facilitates users to access saved projects. By selecting browse option, files stored on their phone memory or on a cloud server can be searched by the users. Then the chosen file can be opened within the app. To provide smooth operation, the app keeps a check on internet connectivity and location services of the smartphone at all times. In case any network or location service issues arise, alert messages are shown. Once these problems are resolved, the chosen files can be accessed within the app.

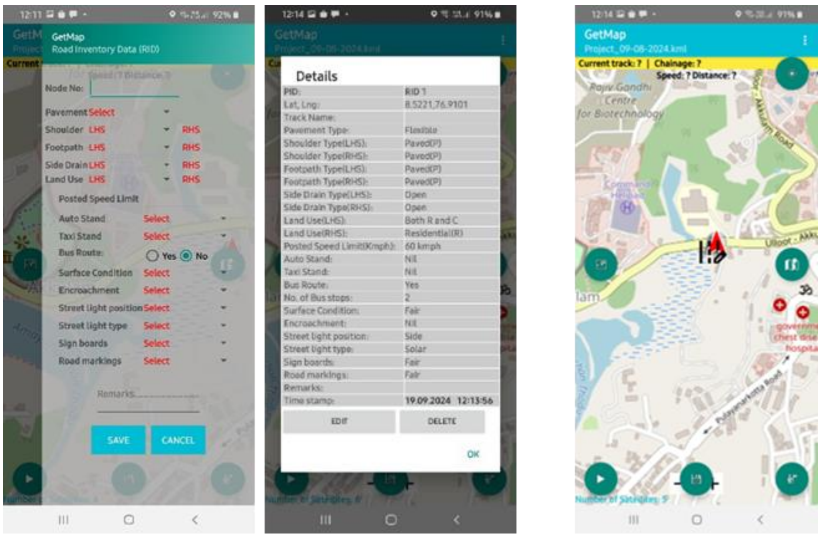


(a) **(b)**

Figure 6. App screen to (a) Enter track name (b) Recording the track

4.3 Add Road Inventory Details

Road inventory details get added to the current track by clicking the ADD DETAILS button. The app prompts the user for confirmation before proceeding to enter road inventory information. Upon confirmation, a sliding navigation drawer appears over the map screen, allowing users to input details such as pavement type, shoulder type, speed limit, footpath presence, and land use. The interface includes textboxes and dropdown menus to simplify data entry and minimize typographical errors, improving upon the previous application's level-based system. The entered road inventory details are saved to the current project as point entities, represented by markers on the OpenStreetMap (OSM). When a marker is clicked, the app displays the associated road inventory details in a user-friendly tabular format, along with the date and time of entry. Figure 7 illustrates the app interface for entering road inventory details via the sliding navigation drawer and the tabular display of recorded information. The app provides options to edit or delete road inventory details. When a user selects a marker, an "EDIT" button appears alongside the road inventory information. Upon confirmation, the update screen (Figure 8) displays the current details, allowing users to modify specific fields and save changes by clicking the "SAVE" button.



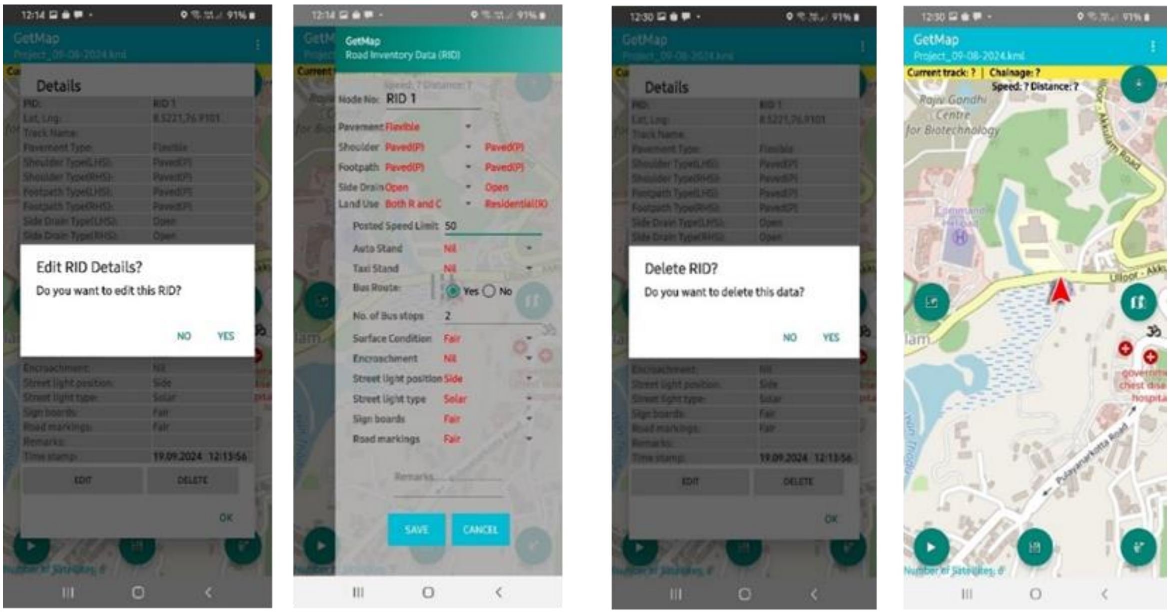
(a) **(b)**

Figure 7. App screen displaying with the (a) Sliding navigation drawer to enter the Road Inventory Details (b) Road Inventory Details as marker

"DELETE" button is available to remove previously saved road inventory details. Clicking this button prompts a confirmation, and upon approval, the marker is removed from the map.

4.4 Record Road Cross-Sectional Data

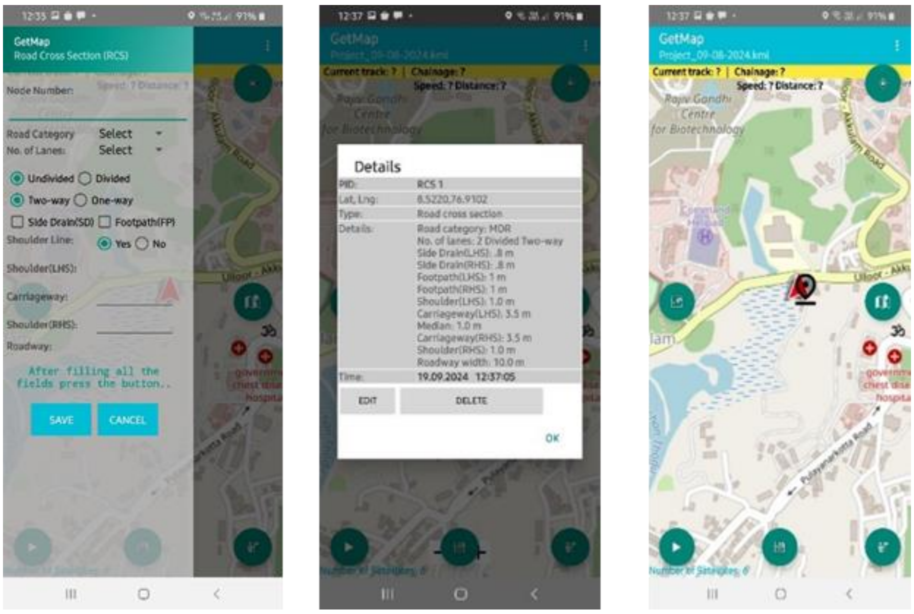
The "ADD DETAILS" icon also enables the recording of road cross-sectional details. When road inventory details are saved, the app prompts the user to confirm if the road cross-sectional data should be recorded. Upon confirmation, a sliding navigation drawer appears over the map screen, allowing the user to input details such as *road category, number of lanes, shoulder width, carriageway and median widths, and the dimensions of side drains and footpath.*



(a) **(b)**

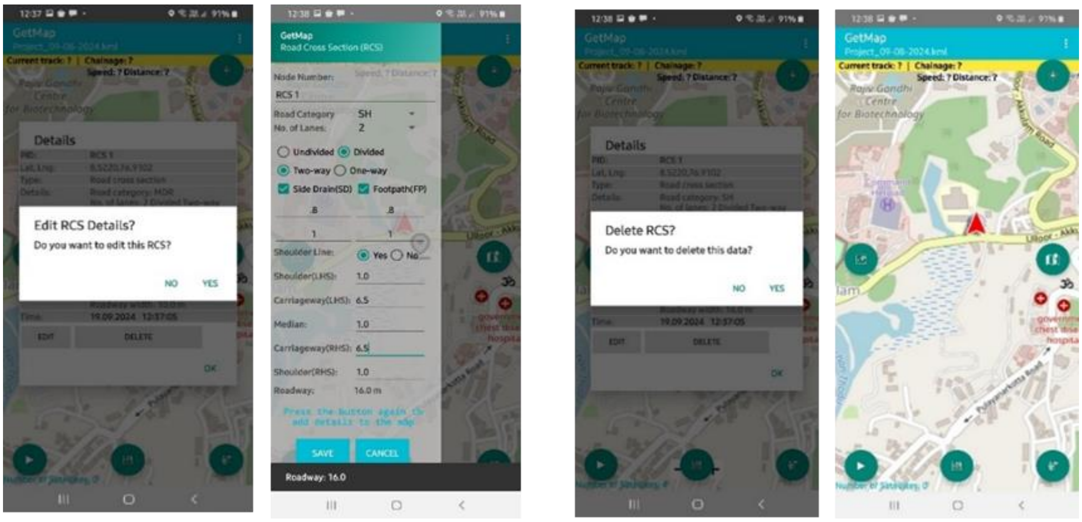
Figure 8. App Screens to (a) Edit Road Inventory Details (b) Delete Road Inventory Details

The roadway width is automatically calculated based on the entered shoulder, carriageway, and median values, with the units of measurement displayed during data entry. Similar to the road inventory data, the recorded road cross-sectional details are saved as point markers on the map (Figure 9). Clicking on the marker displays the entered road cross-sectional details. The app is provided with editing and deleting functionality (Figure 10)



(a) **(b)**

Figure 9. App screen displaying with the (a) Sliding navigation drawer to enter the road cross-sectional details (b) Road Cross-Section details as marker

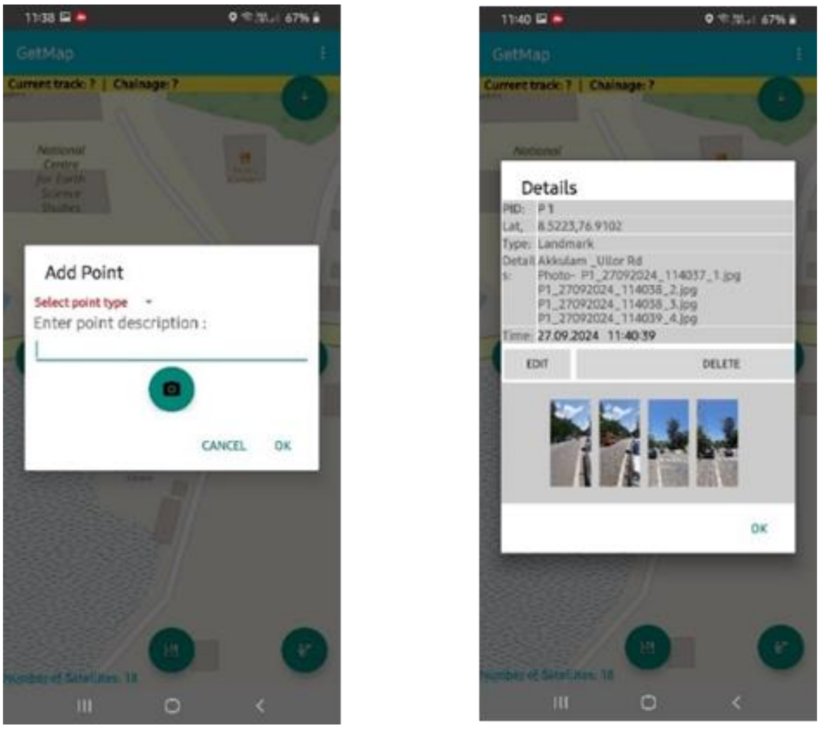


(a) **(b)**

Figure 10. App Screens to (a) Edit Road Cross Sectional Details (b) Delete Road Cross Sectional details

4.5 Logging Location Information

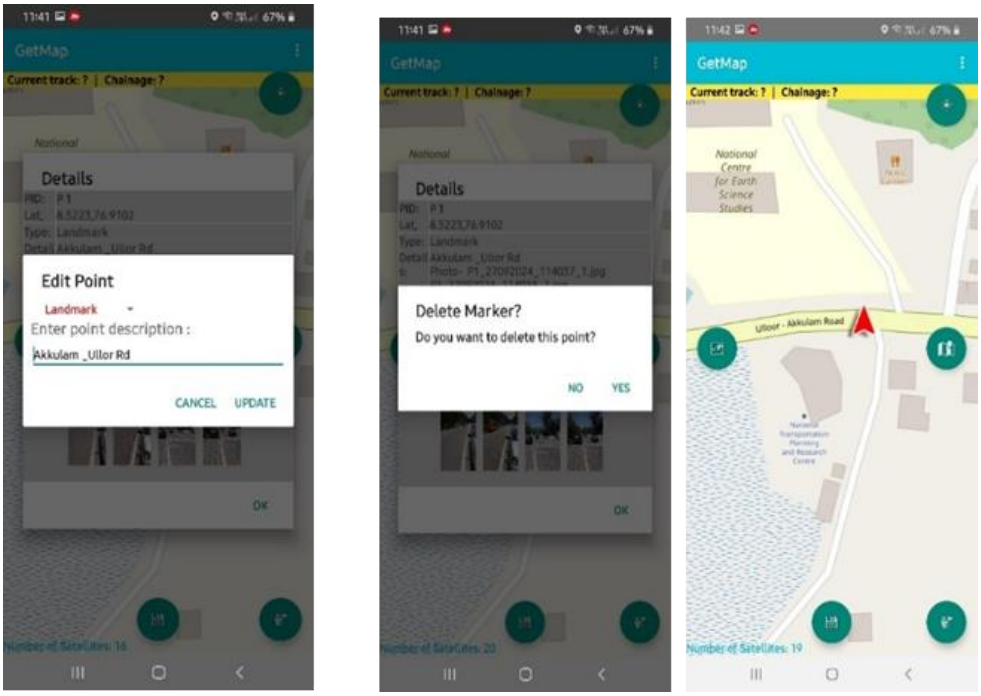
Point features along the road, such as *bus stops, bridges, landmarks, and junctions*, are recorded along the travel path by long-pressing the desired location on the map. The app allows users to provide a description for each point, selected from a pre-defined drop-down list. Different icons are used to represent various point types on the map, ensuring easy identification. Each added point is automatically assigned a sequential number, allowing users to track the order in which the markers were created. An optional feature enables users to upload up to five photos related to the point, which can be previewed within the app (Figure 11). Uploaded photos are stored in a folder within the phone memory alongside the KML file. Each point is also saved to cloud storage by updating the associated file.



(a) **(b)**

Figure 11. (a) App screen used to add a new point (b) App screen displaying the details of the point

The app includes an option to edit the details of previously added points. To access this feature, the user selects a marker and clicks the "EDIT" button. The app prompts for confirmation before proceeding, and upon approval, a screen (Figure 12) is displayed, showing the current details of the point. The user can update the type and description of the point, make any necessary changes, and save them by pressing the "UPDATE" button. Additionally, the app allows users to delete points. By selecting a marker, "DELETE" button appears alongside the point details. Upon confirmation, the marker and its associated details are removed from the map.



(a) (b)

Figure 12. App Screens showing (a) Edit previously added points (b) Delete previously added points

4.6 Output Generation

The traditional process of data collection involves manually gathering field data and subsequently entering it into a database, typically in Excel, for pre-processing and analysis. The app streamlines this labour-intensive task by automatically generating an Excel file with three worksheets for points, road inventory, and cross-section details (Figure 13). Each entry is assigned a unique ID and timestamp for easy identification and retrieval. Captured photo filenames are encoded as hyperlinks within the Excel file, allowing users to directly access the images by clicking the link. Additionally, users can edit or delete the details in the Excel file through the app, with changes automatically reflected in both the phone memory and the cloud server. Geographic features collected using the app are also saved as KML files for further use.

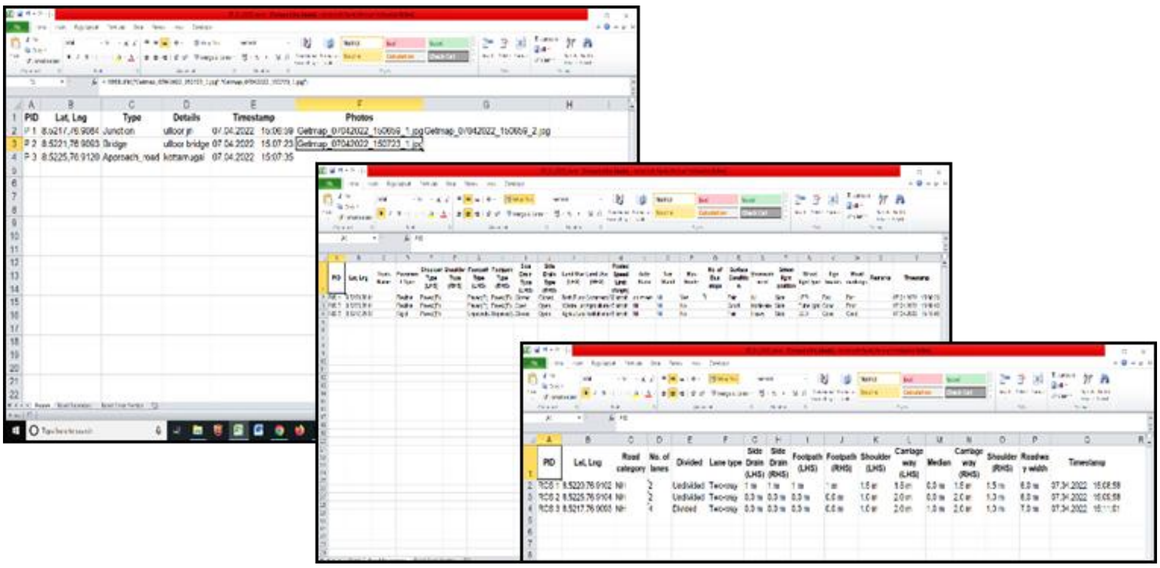


Figure 13. Excel file showing the details of Points, Road inventory, and Road cross-section

5. Admin Interface for data analysis

The admin interface is designed to facilitate efficient data visualization and streamlined project management. This tool allows administrators to quickly review and approve or reject project submissions, ensuring quality control and compliance with predefined criteria. Additionally, the dashboard provides access to registered users' contact information, such as phone numbers and email addresses, simplifying user management and project tracking (Figure 14). By consolidating user data and project oversight functions, the admin interface significantly reduces manual workload and enhances overall productivity.

6. File Management in Cloud Server

The KML/Excel files and photos collected through the app are seamlessly uploaded to the cloud server. To ensure effective file management, the system automatically creates separate folders for each user during the signup process. These folders are named after the user's unique identification number (UID) for easy organization and retrieval (Figure 15).

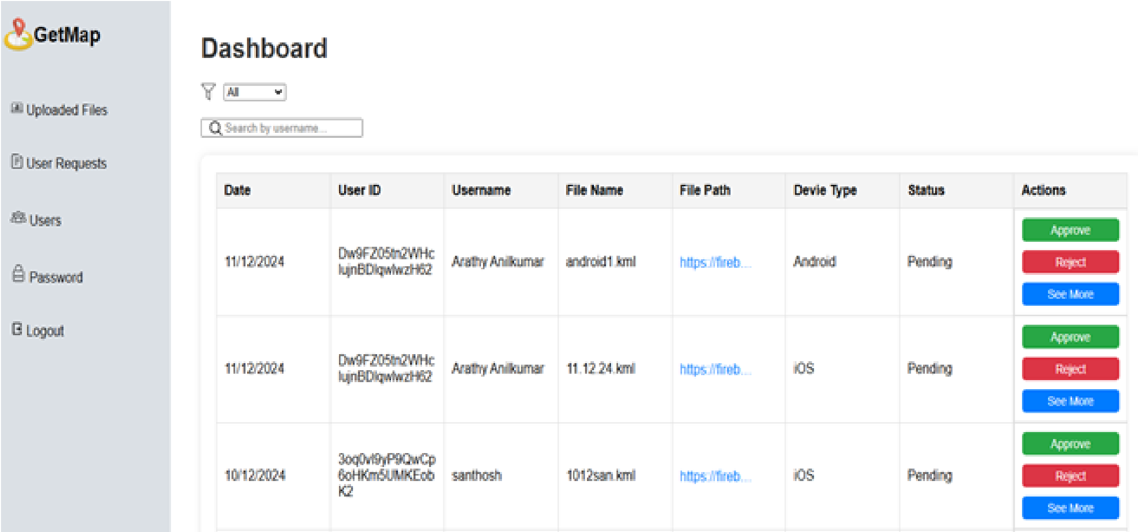


Figure 14. Admin Interface for data analysis

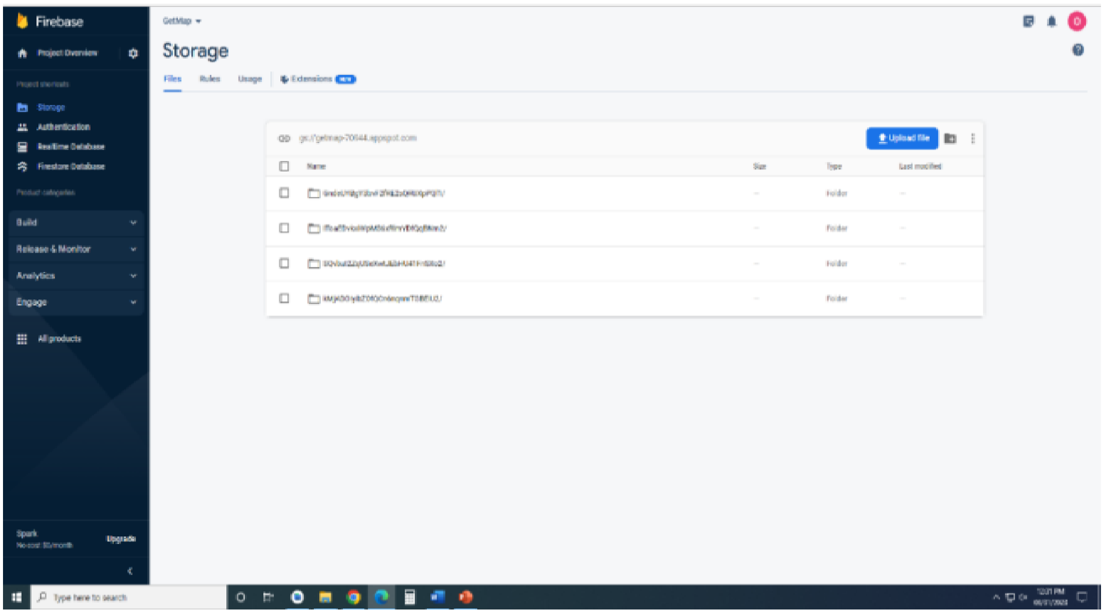


Figure 15. Firebase Cloud Storage Folders

7. Application of the Outcome

The GetMap mobile application was tested for collecting road data by installing it on Android smartphones. The app enables users to record travel paths, points, and other road-related information, which can be saved in both KML and Excel formats. By eliminating the need for manual field surveys, GetMap significantly enhances the efficiency of road data collection. KML files stored on smartphones or cloud servers can be easily transferred to a PC for visualization in Google Earth software (Figure 16). Additionally, KML files can be imported as vector data into QGIS or exported as shapefiles in ArcGIS, expanding possibilities for GIS-based analysis. The automatic export of data into Excel ensures immediate availability, reducing reliance on paper forms and manual data entry. Field users collecting data can simultaneously upload it to the server, ensuring real-time updates of all field-collected data. A web-based admin portal is developed to efficiently manage this data and oversee various levels of authorized users. This setup enhances data security by allowing role-based access control, assigning specific permissions to each user. Officials associated with the survey can verify live data remotely from their office through the web application, streamlining the verification process. The results are depicted in Figure 17.

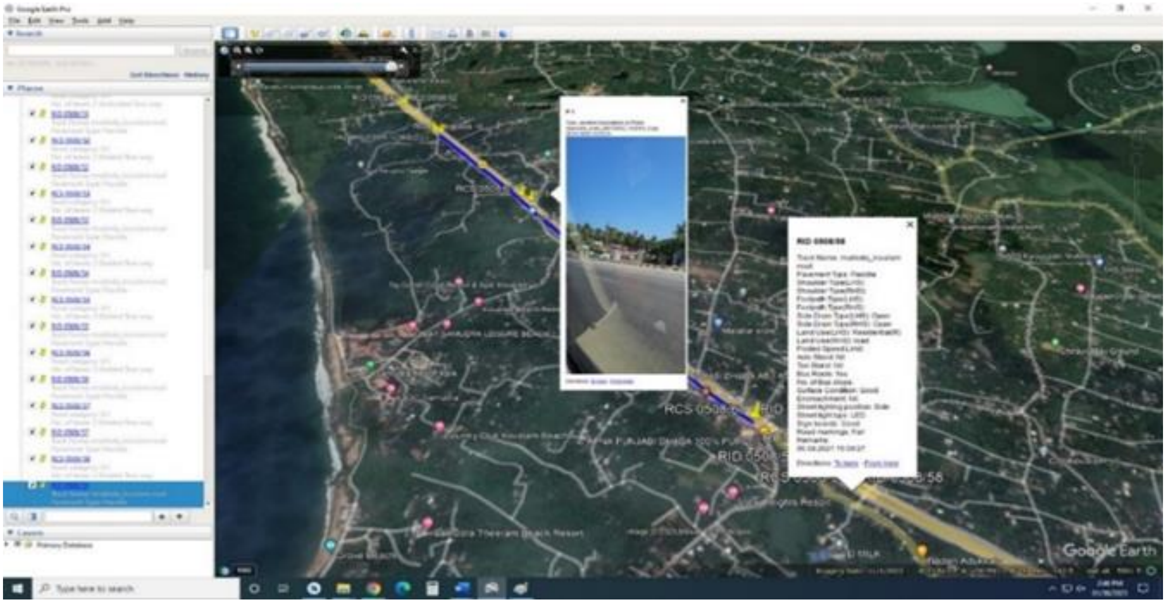


Figure 16. KML file opened in Google Earth

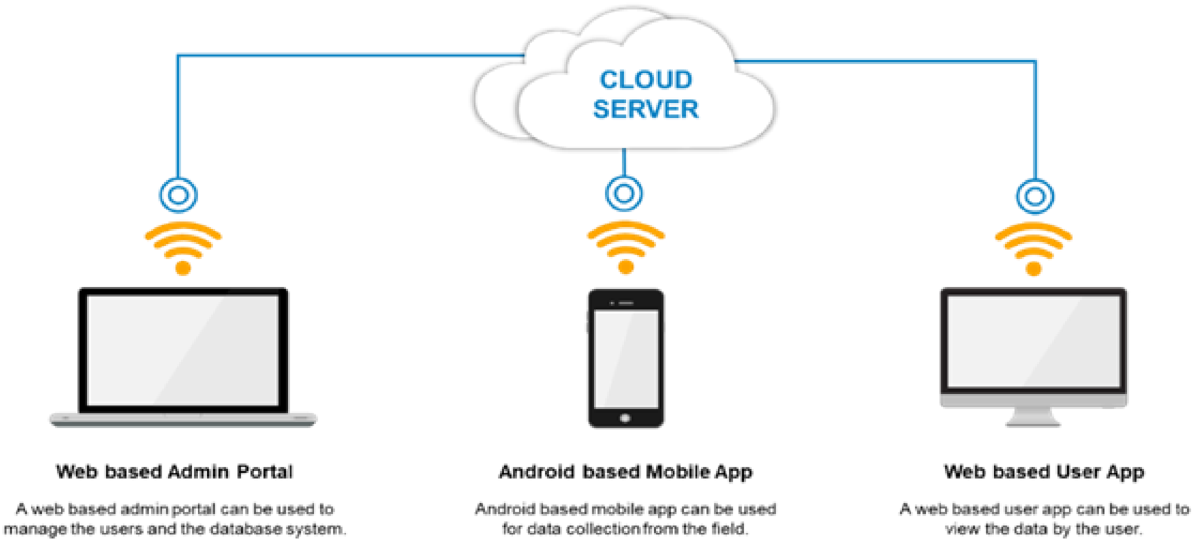


Figure 17. Final Outcome of the Project

8. Validation in the field

After achieving a satisfactory level of development, the **GetMap** application was subjected to rigorous field validation to ensure its functionality and reliability under real-world conditions. The app was installed on various Android smartphones and tested on diverse road types, including urban, rural, and highway environments, to simulate different field scenarios. The validation process involved using the app to record travel paths, add road inventory and cross-sectional details, and mark significant points of interest. The app's ability to handle real-time data collection, GPS accuracy, and seamless interaction with OpenStreetMap (OSM) was closely monitored. This testing phase also included simultaneous uploads to the *Firebase* cloud server to verify data synchronization and integrity. To assess its usability and performance, the app was shared with students from engineering colleges and staff members of the NATPAC office. These participants collected field data, and their feedback was gathered through an online survey conducted via Google Forms. The survey focused on the app's ease of use, functionality, and potential areas for improvement.

8.1 Key Findings

1. User Satisfaction: The majority of users expressed satisfaction with the app's intuitive interface and its capability to streamline road data collection. The

automated generation of KML and Excel files was particularly appreciated for its efficiency in eliminating manual data entry.Feedback Highlights: While the app performed well overall, users indicated that reducing the time required for data entry and optimizing the app's workflow would enhance its practicality further. Suggestions included minimizing the number of steps for recording data and enhancing the app's speed when handling large datasets.

2. Adaptation Challenges: Despite its advantages, a notable challenge was motivating users to transition from traditional paper-based methods to the mobile app. This shift requires changes in mindset and training to maximize the app's potential benefits.
3. Reliability: The app demonstrated robust GPS tracking, reliable data uploads, and accurate visualizations on OSM. However, minor bugs, such as occasional delays in rendering map updates, were identified and addressed in subsequent updates.
4. Outcome: The survey results, summarized in Figure 18, validated the app's utility for field data collection. It highlighted areas for refinement, emphasizing the need for continued updates to improve user experience and efficiency. The field validation process established **GetMap** as a practical tool for road-related data collection. Its performance and feedback from diverse user groups underscored its potential to revolutionize how road data is collected and managed in transportation research and planning.

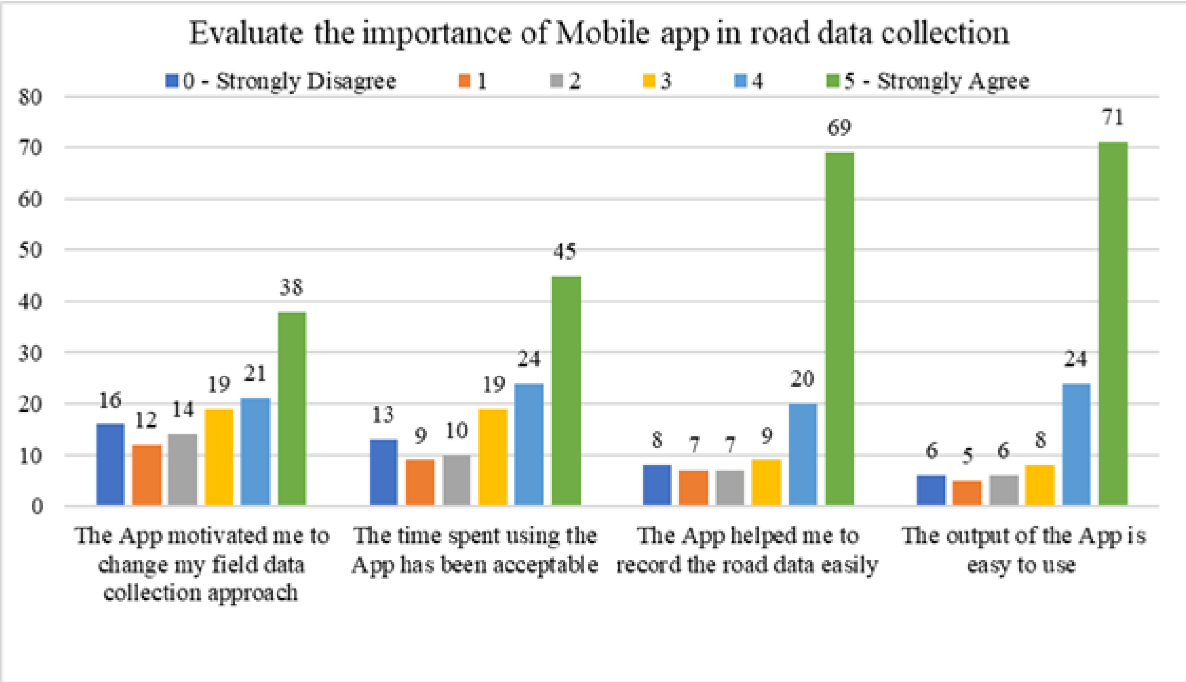


Figure 18. Graphical representation of the data obtained from the Google form

9. Conclusions

The app that has been developed offers a cost-effective solution for gathering road survey data, allowing users to capture travel routes and related road information. Tracks, points, road inventory, and cross-section data are essential road data that can be captured directly with the app. Smartphone's location driver is used by the app to track the user's location using Android location services. Data collected in the app can be directly exported in the form of Excel and KML files, which can be opened directly into Google Earth software or exported for use with any GIS software. As compared to other mobile applications, this app is unique due to its organized and easy to use interface. The application is built to support field data collection activities for any agencies involved in road surveying.

The app currently run on the free Firebase server plan, with further attempts to make the transition to a paid plan and list the app on the Google Play Store. The app currently supports Android. Future development plans include iOS and Windows Phone OS. Additionally, a web-based admin portal is developed to manage data securely and allow for role- based user access. This portal lets survey officials to verify live data remotely, improving efficiency and data security.

Declarations

Disclosure statement

No potential conflict of interest was reported by the authors.

Acknowledgment

We would like to express our sincere gratitude to Kerala State Council for Science, Technology and Environment (KSCSTE) - National Transportation Planning & Research Centre (NATPAC) for providing technology materials and financial support to this study. The authors also acknowledge the limited use of AI-assisted language refinement tools, which were employed solely to improve

grammar and readability. The scientific content, originality, and responsibility for the manuscript remain entirely with the authors.

Notes

1. Android https://en.wikipedia.org/wiki/Android_(operating_system) (accessed: 10 June 2021).
2. OpenStreetMap https://www.openstreetmap.org/about (accessed: 7 June 2021).

References

Abdulazim, T., H. Abdelgawad, K. M. N. Habib and B. Abdulhai (2013). Using smartphones and sensor technologies to automate the collection of travel data. Transportation Research Record: Journal of the Transportation Research Board, 2383(1), 44–52.

Allaire, F. and T. Hanson (2017). Potential of road roughness data from smartphones as an input to spring weight restriction decision-making. Innovations in Pavement Management, Engineering and Technologies, Transportation Association of Canada, Canada, 24–27 September 2017.

Bucher, D., D. Jonietz and M. Raubal (2017). A heuristic for multi-modal route planning. In Gartner, G. and H. Huang (eds.) Progress in Location-Based Services 2016, 211–229.

Choy, S., J. Handmer, J. Whittaker, Y. Shinohara, T. Hatori and N. Kohtake (2016). Application of satellite navigation system for emergency warning and alerting. Computers, Environment and Urban Systems, 58, 12–18.

Huang, H., G. Gartner, J. Krisp, M. Raubal and N. Van de Weghe (2018). Location based services: Ongoing evolution and research agenda. Journal of Location Based Services, 12, 63–93.

Jadhav, R., J. Patel, D. Jain and S. Phadhtare (2014). Emergency management system using Android application. International Journal of Computer Science and Information Technologies (IJCSIT), 5(3), 2803–2805.

Krisp, J. M. and A. Keler (2015). Car navigation – computing routes that avoid complicated crossings. International Journal of Geographical Information Science, 29(11), 1988–2000.

Li, Y., J. Wang and L. Zhang (2015). LBS-based dilemma zone warning system at signalized intersection. In Gartner, G. and H. Huang (eds.) Progress in Location-Based Services 2014, 223–237, Cham, Springer. https://doi.org/10.1007/978-3-319-11879-6_16.

Raper, J., G. Gartner, H. Karimi and C. Rizos (2007). A critical evaluation of location based services and their potential. Journal of Location Based Services, 1(1), 5–45.

Sadoun, B. and O. Al-Bayari (2007). LBS and GIS technology combination and applications. IEEE/ACS Computer Systems and Applications, Jordan, 13–14 May 2007.

Sailer, C., P. Kiefer, J. Schito and M. Raubal (2016). Map-based visual analytics of moving learners. International Journal of Mobile Human Computer Interaction, 8(4), 1–28.

Seliverstov, S. A., I. G. Malygin, A. L. Starichenkov, R. R. Muksimova, V. A. Grigoriev and A. N. Asaul (2017). Modeling of megalopolis traffic flows with the introduction of a new line of water intercity passenger transport. Soft Computing and Measurements (SCM), Russia, 24–26 May 2017.

Silyanov, V., J. Sodikov, R. Kiran and I. Sadikov (2020). An overview of road data collection, visualization, and analysis from the perspective of developing countries. IOP Conference Series: Materials Science and Engineering, 832(1), 012056.

Toledo, M. P., J. C. Y. Sarvida, B. K. S. Patiten, D. C. R. Mitamura and R. R. H. Guadaña (2017). SakunAPP: A framework for mobile application development in disaster awareness, preparedness and response. TENCON 2017 – IEEE Region 10 Conference, Penang, Malaysia, 5–8 November 2017.

Vlassenroot, S., D. Gillis and R. Bellens (2015). The use of smartphone applications in the collection of travel behaviour data. International Journal of Intelligent Transportation Systems Research, 13(1), 17–27.