



International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 15, Issue 3, March 2026

FixServe: An On-Demand Electronics Repair Service Management System

Ms.G.V. Badgujar¹, Nachiket Narwelkar², Atharv Sawant³, Onkar Mandale⁴, Pranav Kumbhar⁵

Department of Information Technology Engineering, Pimpri Chinchwad Education Trust's, Pimpri Chinchwad
Polytechnic, Pradhikaran, Nigdi, Pune, India^{1,2,3,4,5}

ABSTRACT: FixServe is a full-stack mobile-based platform designed to simplify and modernize the process of booking electronic device repair services. Traditional repair services often involve delays, high service costs, and inconvenient manual booking procedures. To address these challenges, this research presents the design and implementation of an On-Demand Electronics Repair Service Management System that connects users with verified technicians through a real-time mobile application. The system is developed using Flutter for the frontend, Flask for the backend API services, and MongoDB for data storage. Real-time communication between users and technicians is achieved using Socket.IO, enabling live booking notifications, job status updates, and messaging functionality. The platform supports role-based access for customers and technicians, allowing users to create repair requests with device details and issue images, while technicians can receive and accept service requests based on availability. The system also incorporates a simulated real-time tracking mechanism for demonstration purposes and an offline intent-based support chatbot embedded within the mobile application to assist users with booking and service-related queries. Experimental implementation results demonstrate that the system improves service accessibility, reduces response time, and provides a scalable architecture for on-demand repair service platforms.

KEYWORDS: On-Demand Services, Electronics Repair Platform, Flutter Mobile Application, Flask Backend, MongoDB Database, Role-Based Access, Real-Time Communication, Socket.IO, Service Booking System, Intent-Based Chatbot.

I. INTRODUCTION

In our earlier work [9], the conceptual design and architecture of an on-demand electronics repair service management system were presented. The present study extends that work by implementing the FixServe platform as a fully functional mobile-based system with real-time communication and technician workflow management.

Electronic devices such as smartphones, laptops, televisions, and home appliances have become essential in modern daily life. As the use of these devices increases, the demand for reliable and efficient repair services has also grown. However, traditional repair systems often involve several challenges such as delayed technician response, inconvenient booking processes, and lack of transparency in service progress. Customers are typically required to visit service centers or contact technicians manually, which can lead to delays and inefficient service management.

With the advancement of mobile technologies and internet connectivity, on-demand service platforms have become increasingly popular for connecting users with service providers through digital applications. These platforms demonstrate how mobile-based systems can simplify service booking, improve communication, and reduce service response time. However, many existing platforms focus on general home services and do not specifically address the needs of electronic device repair services.

To address these challenges, this research presents FixServe, an On-Demand Electronics Repair Service Management System designed to streamline the repair service process through a mobile application. The system is implemented using a Flutter-based frontend, a Flask-based backend server, and MongoDB for data storage. Real-time communication between users and technicians is supported through Socket.IO, enabling instant service request notifications and job status updates.

The FixServe platform supports two primary roles: users and technicians. Users can create repair bookings by providing device details and issue descriptions, while technicians receive service requests and manage repair tasks through a dedicated dashboard. The system also includes additional features such as simulated technician tracking and an integrated offline chatbot that assists users with common service-related queries.

The proposed system demonstrates how a mobile-based platform can improve service accessibility, communication, and efficiency in managing electronics repair services.

II. LITERATURE REVIEW

With the increasing use of mobile technologies and internet-based services, several digital platforms have been developed to simplify service management and improve accessibility for users. On-demand service applications such as ride-hailing, food delivery, and home service platforms demonstrate how mobile applications can efficiently connect customers with service providers. These platforms reduce service delays, improve communication, and enhance operational efficiency through real-time updates and automated service request handling.

Recent research in mobile service management systems focuses on integrating mobile applications with backend servers and cloud-based databases to manage service requests, user information, and service provider interactions. Many systems implement role-based access mechanisms where users can request services and service providers can manage tasks through dedicated dashboards.

However, most existing platforms are designed for general home services and do not specifically address the challenges associated with electronic device repair services. Additionally, some systems lack integrated real-time communication and intelligent assistance features for users during the booking process. To address these limitations, the proposed **FixServe system** provides a full-stack mobile-based platform that enables users to request electronic repair services while allowing technicians to manage service tasks efficiently through real-time updates and an integrated support chatbot.

III. METHODOLOGY

The FixServe platform is implemented using a full-stack architecture that integrates a mobile frontend, a backend application server, and a locally hosted database system. The system provides a unified environment where users can request electronic device repair services while technicians can manage repair tasks efficiently through a mobile application. The architecture follows a layered design consisting of the User Interface Layer, Application Logic Layer, and Database Layer, ensuring modularity, scalability, and efficient communication between system components.

The **User Interface Layer** is developed using the Flutter framework, enabling the creation of a responsive cross-platform mobile application. The application supports two primary roles: users and technicians. Users can register and log in to the system, manage their profile information, create repair bookings by providing device details and issue descriptions, and optionally upload images of faulty components. Once a booking is created, users can view booking history, monitor service status, communicate with technicians through a live chat interface, and track technician location through a demonstration tracking interface. Technicians can register on the platform by submitting required personal details and verification documents. These documents are verified before technicians are allowed to receive repair requests. After successful registration and authentication, technicians can toggle their availability status, receive incoming repair requests in real time, accept jobs, update service progress, and monitor their earnings. The mobile application also uses local persistence mechanisms such as SharedPreferences to maintain user sessions and store authentication tokens.

The **Application Logic Layer** is implemented using the Flask web framework, which provides RESTful APIs for managing authentication, booking operations, technician interactions, and service status updates. User authentication is secured using JSON Web Tokens (JWT), enabling session persistence and secure communication between the mobile application and backend services. Email verification is performed during account registration to ensure authenticity of users and technicians. Real-time communication between users and technicians is implemented using Socket.IO, allowing instant booking notifications, job acceptance updates, status synchronization, and live chat functionality. The

backend also implements several business logic rules, including limiting one active job per technician, restricting users to a single active booking at a time, and managing service cancellation conditions. For navigation assistance, the system includes a simulated technician tracking mechanism implemented using a polling-based approach with predefined latitude and longitude values. The mobile application periodically retrieves location updates from the backend, allowing the interface to display technician movement toward the user location. The system provides a demonstration-based technician tracking interface where technicians can view user location and move toward the destination until they are within approximately 120 meters, after which the repair process can begin. The backend additionally includes a **beyond-repair reporting mechanism** for situations where a device cannot be repaired. In such cases, technicians can submit a report by uploading an image and providing a description explaining why the device is not repairable. Once verified, the system automatically cancels the booking and allows the technician to receive new service requests. The backend also manages pricing calculations, platform commission, technician earnings, and service completion workflows. The **Database Layer** is implemented using MongoDB, where all system data is stored in structured collections including users, technicians, bookings, chat messages, service reports, and payment records. This document-based structure allows efficient handling of dynamic service data and simplifies the management of repair service transactions. In addition to the core service workflow, the system also incorporates an offline intent-based chatbot integrated directly within the mobile application. The chatbot assists users by responding to common queries related to booking creation, cancellation procedures, technician delays, payment issues, and service support, thereby improving user experience without requiring external cloud-based AI services.

Overall, the FixServe system integrates mobile application technologies, secure authentication mechanisms, real-time communication frameworks, and scalable backend services to provide an efficient and user-friendly platform for managing on-demand electronics repair services.

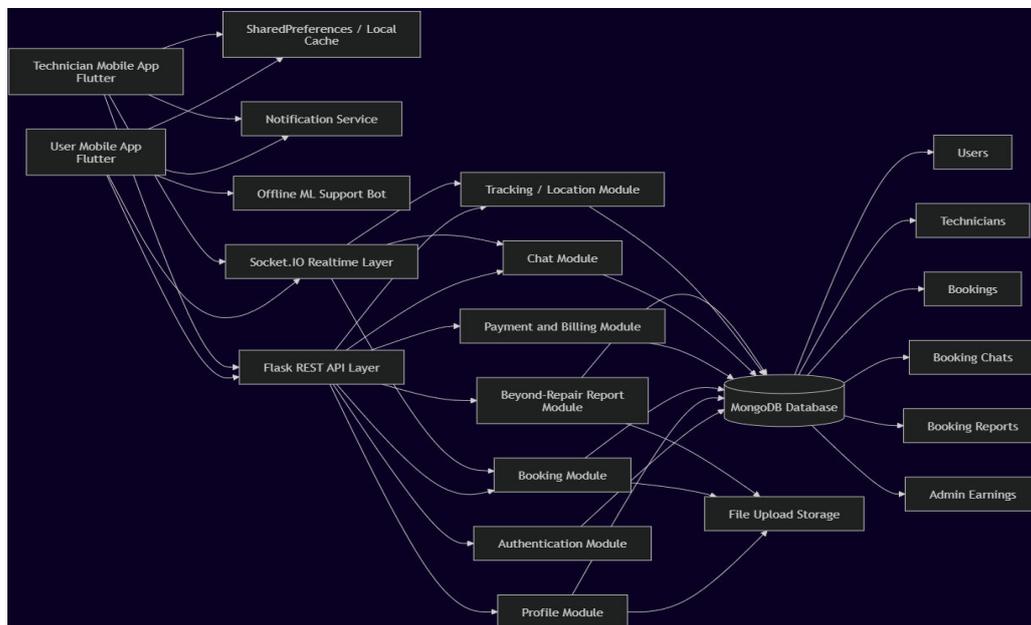


Fig.1 System Architecture

IV. EXPERIMENTAL RESULTS



Fig. 2 User Dashboard

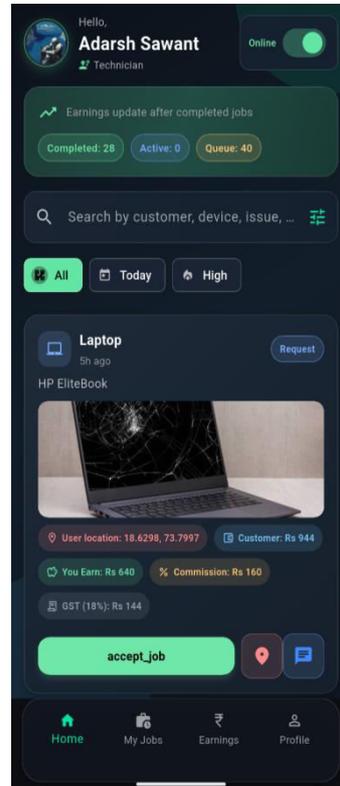


Fig. 3 Technician Dashboard

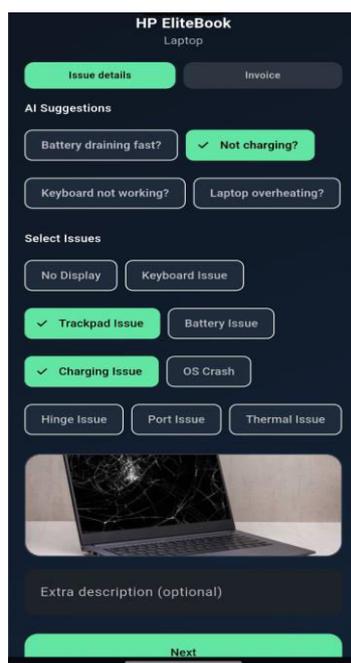


Fig. 4 Booking Creation Screen

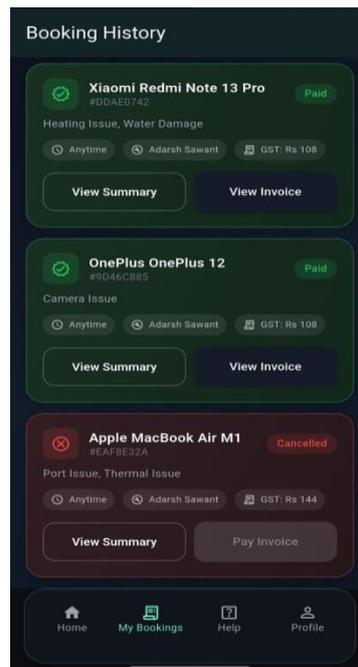


Fig. 5 Booking History Screen

The experimental implementation confirms that the FixServe platform successfully integrates user booking management, technician task handling, and real-time communication within a single mobile-based system. The presented interfaces demonstrate how the platform simplifies the repair service process while maintaining efficient coordination between users and technicians. The user dashboard and booking creation interface illustrated in **Fig. 2** and **Fig. 4** demonstrate how users can easily create and manage repair requests, while the technician dashboard shown in **Fig. 3** enables efficient job management. The booking history interface presented in **Fig. 5** allows users to review past service records and track completed repair activities.

The developed FixServe platform improves service accessibility and coordination between users and technicians compared to traditional repair booking methods. The use of a mobile-based interface simplifies service request creation, while real-time communication using Socket.IO enables instant updates and interaction between system participants. The role-based architecture ensures organized service management, allowing technicians to efficiently handle incoming repair requests and track their earnings. These features demonstrate that the proposed platform provides a more structured and efficient approach for managing electronics repair services.

V. CONCLUSION

This research presented **FixServe**, an On-Demand Electronics Repair Service Management System designed to simplify the process of booking and managing electronic device repair services through a mobile platform. The system integrates a Flutter-based mobile application, a Flask backend server, and a MongoDB database to provide a scalable and efficient service management environment. Real-time communication between users and technicians is achieved using Socket.IO, enabling instant service requests, job status updates, and live interaction.

The platform supports role-based access for users and technicians, allowing customers to create repair bookings while technicians can accept jobs, update service progress, and manage their earnings through a dedicated dashboard. Additional features such as simulated technician tracking, booking history management, and an offline intent-based chatbot further enhance the usability of the platform. The experimental implementation demonstrates that the FixServe system effectively supports repair service coordination and improves service accessibility through a mobile-based architecture.

Future improvements may include integration with real GPS-based navigation services, advanced analytics for service optimization, and deployment of cloud-based infrastructure for large-scale service management.

REFERENCES

- [1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image Inpainting," Proceedings of SIGGRAPH, pp. 417–424, 2000.
- [2] A. Criminisi, P. Perez, and K. Toyama, "Region Filling and Object Removal by Exemplar-Based Image Inpainting," IEEE Transactions on Image Processing, vol. 13, no. 9, pp. 1200–1212, 2004.
- [3] Google Developers, "Flutter Documentation," Available: <https://flutter.dev>
- [4] Flask Documentation, "Flask Web Framework," Available: <https://flask.palletsprojects.com>
- [5] MongoDB Inc., "MongoDB Atlas Documentation," Available: <https://www.mongodb.com>
- [6] P. Sharma and R. Gupta, "A Study on On-Demand Service Platforms," International Journal of Computer Applications, vol. 183, no. 25, pp. 1–8, 2021.
- [7] K. Verma and M. Patel, "Design and Implementation of On-Demand Service Mobile Application," International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 2, pp. 54–61, 2020.
- [8] S. Kumar, A. Singh, and P. Sharma, "IoT Enabled Service Automation in Urban Areas," IEEE Access, vol. 8, pp. 145672–145683, 2020.
- [9] G. V. Badgajar, N. Narwelkar, A. Sawant, O. Mandale, and P. Kumbhar, "On-Demand Electronics Repair Service Management System," International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET), vol. 14, issue 10, Oct. 2025.



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details