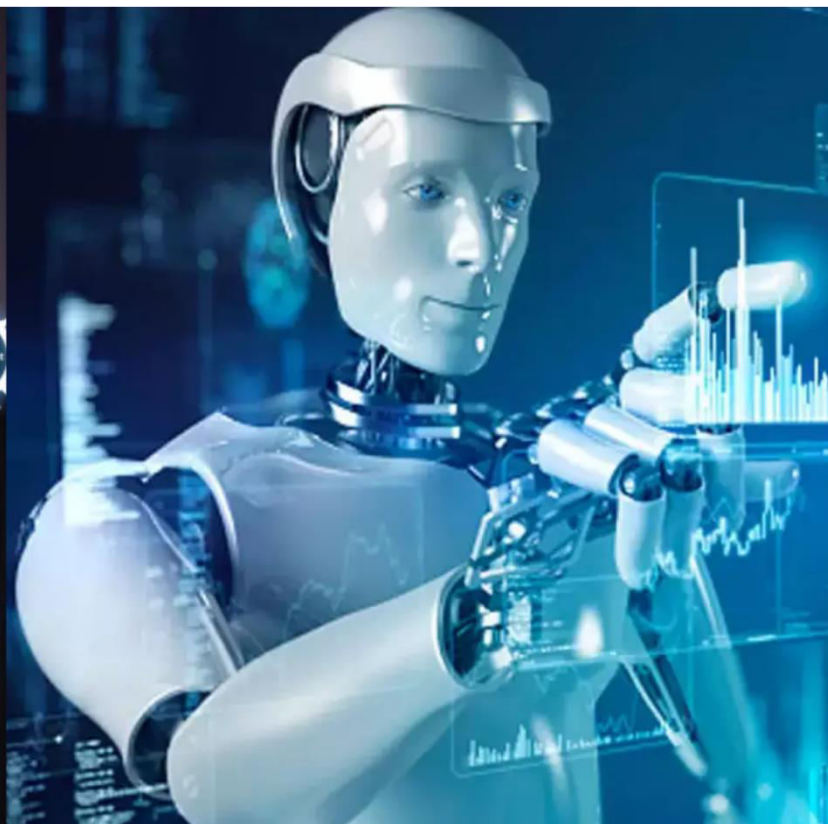




International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

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



Impact Factor: 8.699

Volume 14, Issue 10, October 2025

On-Demand Electronics Repair Service Management System

G.V. Badgujar, Nachiket Narwelkar, Atharv Sawant, Onkar Mandale, Pranav Kumbhar

Department of Information Technology Engineering, Pimpri Chinchwad Education Trust's

Pimpri Chinchwad Polytechnic, Pradhikaran, Nigdi, Pune, India

ABSTRACT: The present research paper details the design and development plan of an On-Demand Electronics Repair Service Management System (O-DERMS), which is intended to address common challenges encountered in repairing electronic devices such as mobile phones, televisions, washing machines, and other household appliances. Users frequently experience high service costs, delayed technician response, and cumbersome booking procedures when approaching official service centres. In order to overcome these challenges, the proposed system provides a unified mobile platform that allows users to submit repair requests along with device details and photographs of the faulty components. Once a request is made, the system automatically assigns a nearby technician based on location and expertise. Furthermore, the platform implements dual interfaces for customers and technicians, ensuring that technicians are verified before accessing repair requests, thereby maintaining trust and accountability. Payments are processed only after repair confirmation, ensuring transparency for both users and service providers. This paper primarily focuses on the proposed system architecture, workflow, and frontend design, demonstrating how this platform has the potential to improve convenience, service transparency, and overall efficiency.

KEYWORDS: On-Demand Services, Electronics Repair, Role-Based Access, Flutter, Python, MongoDB, Mobile Application, Technician Verification

I. INTRODUCTION

In contemporary life, electronic devices have become integral to daily functioning, yet repairing them remains a time-consuming and costly process. Official service centres often respond slowly, charge substantial fees, and require users to follow inconvenient manual booking procedures. Recognizing these persistent challenges, our team proposes a digital platform designed to streamline electronics repair services through an on-demand mobile application. The platform enables users to submit repair requests along with photographs of the faulty device components, ensuring accurate diagnosis and repair. It also incorporates a location-based allocation system that assigns the nearest verified technician to the request, thus reducing waiting time and enhancing service reliability. The application introduces role-based access, whereby users must select either a Customer Login or Technician Login upon opening the app. Depending on the selected role, the interface dynamically adapts to provide functionalities specific to customers or technicians while maintaining a consistent overall theme. Technicians undergo a verification process prior to gaining access to repair requests, which prevents misuse and ensures that only skilled personnel perform repairs. This approach guarantees a secure, efficient, and user-friendly repair ecosystem, offering convenience, trust, and quality service.

II. LITERATURE REVIEW

Several existing on-demand service platforms, such as Urban Company and delivery applications like Zepto, employ role-based verification and user-friendly interfaces to enhance service reliability and customer satisfaction. Research studies indicate that verified service providers significantly improve user trust and satisfaction, while role-specific interfaces reduce operational errors and confusion during service execution. Additionally, the integration of photo-based service requests improves diagnostic accuracy and expedites repair workflows. Despite these advances, few systems focus specifically on the electronics repair domain. Our proposed platform leverages these proven principles but integrates both customer and technician roles within a single mobile application, creating a seamless and intuitive user experience. By consolidating service requests, technician allocation, verification, and payment workflows into one platform, the system addresses the limitations of existing service solutions.

III. PROPOSED METHODOLOGY

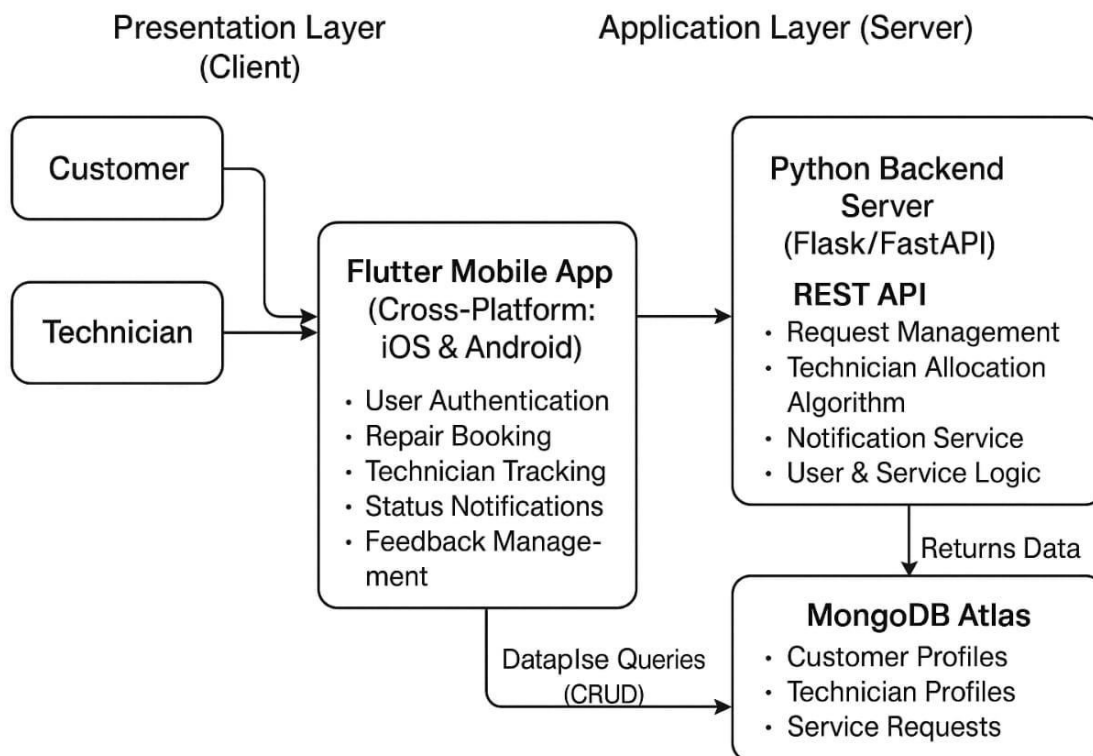
The proposed system adopts a three-layer architecture comprising the User Interface Layer, Application Logic Layer, and Database Layer. The User Interface Layer is developed using Flutter, enabling cross-platform compatibility and a responsive design. Through a single application, customers can submit repair requests along with device photographs, track technician locations, confirm repair completion, and process payments. Technicians, on the other hand, can view assigned jobs, manage schedules, accept or reject requests, and monitor earnings.

The Application Logic Layer, developed using Python Flask, handles the workflow of repair request submission, technician allocation based on location and expertise, real-time notifications, and secure payment processing. Importantly, this layer enforces that only verified technicians gain access to repair requests, ensuring service integrity.

The Database Layer, built on MongoDB, stores user profiles, device details, repair requests, technician records, verification statuses, and payment transactions. The system workflow is as follows: after selecting their role, a customer submits a repair request along with device details and photos. Verified technicians are then notified and can accept or reject the job. Once the repair is performed, the customer confirms completion, triggering payment processing. Both parties have the option to provide ratings and feedback, fostering accountability and service quality.

IV. SYSTEM ARCHITECTURE

The proposed On-Demand Electronics Repair Service Management System (O-DERMS) follows a client-server architecture to ensure scalability, modularity, and efficient communication between users and the backend system. The client side is developed using Flutter, which enables a unified cross-platform mobile application for both Android and iOS. This single app provides distinct interfaces for customers and technicians while maintaining a consistent design theme. Customers can log in to submit repair requests, upload device images, track technician locations, confirm repair completion, and process payments securely. Technicians, on the other hand, can view assigned jobs, manage schedules, accept or reject repair requests, and track their earnings. The system also enforces role-based access control (RBAC), ensuring that users interact only with features relevant to their role. The backend or application layer is built using Python Flask, which serves as the bridge between the client and the database through RESTful APIs. It handles business logic, repair request management, technician allocation based on location and expertise, authentication and verification, real-time notifications, and payment workflows. Flask ensures secure and smooth data exchange while maintaining low latency and reliability. The database layer, implemented using MongoDB Atlas, acts as the cloud-based data repository for storing all relevant information including user profiles, technician details, repair requests, images, feedback, and payment transactions. Its NoSQL document-oriented structure offers flexibility and scalability, allowing efficient handling of large volumes of semi-structured data. The communication between these layers follows a structured request-response cycle, where data from the mobile app is processed by the Flask server, stored or retrieved from MongoDB, and returned to the client for real-time updates. This architecture not only ensures fast and secure operations but also promotes easy maintenance and future scalability. By integrating both customer and technician functionalities within a single application, the system reduces complexity and operational cost. Moreover, its modular design supports future upgrades such as AI-based fault prediction, chatbot-driven troubleshooting, and multi-language support. Overall, this architecture provides a robust, reliable, and extensible foundation for delivering efficient and trustworthy on-demand electronic repair services.



V. IMPLEMENTATION

The implementation phase of the On-Demand Electronics Repair Service Management System (O-DERMS) focuses primarily on developing an interactive and user-friendly mobile interface, establishing seamless communication with the backend, and ensuring secure data handling between the client and server. The frontend design serves as the user's entry point, offering two distinct yet integrated interfaces within a single Flutter-based application: one for customers and the other for technicians. Upon launching the application, users are presented with a role selection screen—allowing them to choose either “Customer” or “Technician.” Once a selection is made, the user is directed to a role-specific dashboard tailored to their needs and permissions, while maintaining a uniform aesthetic design throughout the app for consistency and brand identity.

For customers, the interface allows effortless navigation to submit repair requests, attach photos of damaged or malfunctioning devices, provide additional notes, and select the category of the device—such as mobile, television, or washing machine. The customer dashboard displays a real-time technician tracking system, enabling users to monitor the location and status of their assigned technician. Furthermore, customers can view repair history, receive service notifications, and confirm the completion of repairs before proceeding with payment. Once the repair is verified, the integrated payment process ensures secure transaction handling using verified payment gateways.

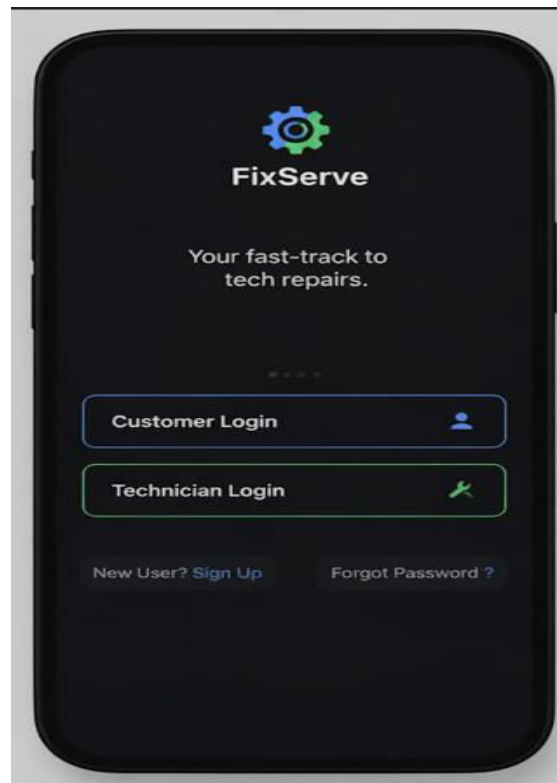
On the technician side, the interface provides a dashboard for managing job requests, schedules, and earnings. Technicians can view newly available repair requests, accept or reject jobs based on proximity and workload, and update the service status in real time. Additionally, technicians have access to features such as job tracking, customer communication, and feedback management, allowing them to maintain transparency and accountability. The system ensures that only verified technicians—those who have successfully completed the verification and approval process—can access and manage service requests, thus preventing misuse and ensuring reliability.

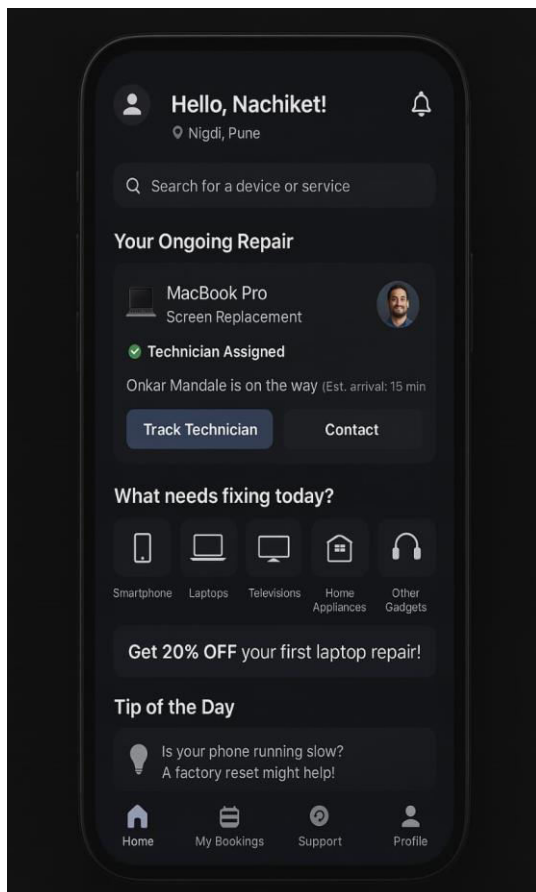
From the backend perspective, the frontend interacts with Python Flask-based RESTful APIs that process all service requests, store data in MongoDB, and handle authentication, verification, and notification mechanisms. The Flutter app communicates via HTTPS protocols, ensuring encrypted and secure data transmission between the client and server.

The integration with MongoDB Atlas allows the system to store user details, technician profiles, repair logs, feedback, and payment records efficiently in a NoSQL environment. The design supports scalability and high availability, ensuring the system remains responsive even under heavy load conditions.

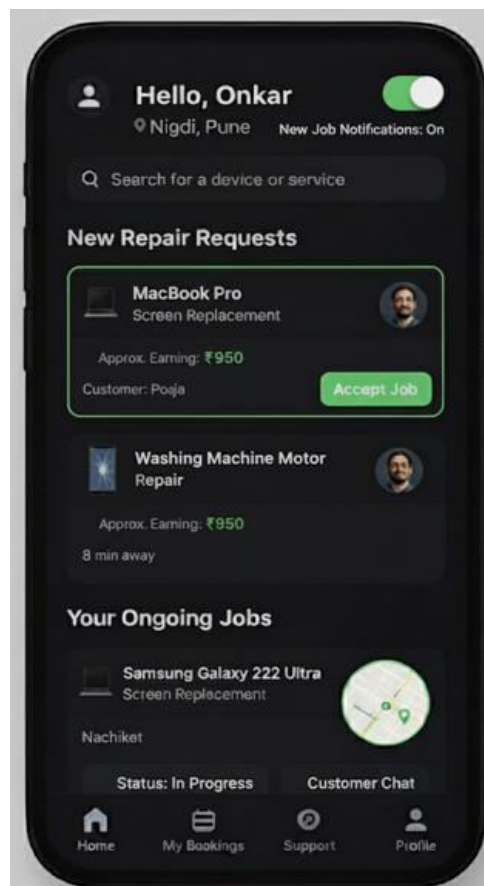
During the implementation phase, several mock-up designs and screenshots are prepared to visually demonstrate the user flow and layout. These include the login/role selection screen, customer dashboard, technician dashboard, and service booking pages. Each mock-up emphasizes simplicity, clarity, and user-centered design principles, ensuring accessibility across devices with different screen sizes. The implementation strategy also focuses on modular code organization, allowing easy updates, testing, and future feature integration such as AI-driven diagnostic assistance or chatbot-based customer support.

Overall, the implementation demonstrates how a single, well-structured application can effectively serve two user categories—customers and technicians—without requiring separate installations or complex user switching mechanisms. This unified design approach not only simplifies maintenance but also provides a cohesive, efficient, and intuitive user experience, setting the foundation for a scalable and production-ready system. Dashboard Image:





Customer Interface



Technician Interface

VI. RESULTS AND DISCUSSION

Since the full backend implementation and live testing are not yet completed, the expected results are described based on the proposed design and workflow. The automated technician allocation system is anticipated to reduce delays compared to traditional service centers. Verified technicians enhance trust and reliability, preventing unauthorized access. The dual interface design improves usability for both customers and technicians, ensuring a user-friendly experience. Payment upon repair confirmation ensures transparency and accountability. Once implemented, this system is expected to significantly improve service quality, efficiency, and customer satisfaction.

VII. CONCLUSION AND FUTURE SCOPE

The On-Demand Electronics Repair Service Management System offers a proposed solution to the common challenges faced in electronics repair services. The system incorporates a role-based dual interface in a single mobile application, ensures technician verification, allows photo-based repair requests, and enables payment only after confirmation.

Future enhancements include backend integration and real-time testing, AI-based predictive maintenance suggestions, a chatbot for instant troubleshooting support, multi-payment gateway integration, and an analytics dashboard with multi-language support. This proposed platform demonstrates a scalable, user-friendly, and efficient solution for managing electronics repair services in an on-demand framework.

REFERENCES

1. S. Kumar, A. Singh, and P. Sharma, "IoT-Enabled Service Automation in Urban Areas," IEEE Access, vol. 8, pp. 145672–145683, 2020.
2. 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.
3. Google Developers, "Flutter Documentation," [Online]. Available: <https://flutter.dev>
4. MongoDB Inc., "MongoDB Atlas Overview," [Online]. Available: <https://www.mongodb.com/cloud/atlas>
5. K. Verma and M. Patel, "Design and Implementation of On-Demand Service Mobile Application," International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 9, no. 2, pp. 54–61, 2020.
6. M. Joshi, "Mobile-Based Electronics Repair Service Management System," Journal of Mobile Computing, vol. 12, pp. 23–31, 2021.



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