iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79654

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

LocalSeva: A Web-Based Local ServiceAggregation Platform

Asst. Prof. Ashish Trivedi[1], Palak Kumbhare[2], Apeksha Bagde[3], Pranali Raut[4], Riddhi Bansod[5], Riya Chaudhari[6], Ruchika Dhengre[7]
Department of Computer Science and Engineering, G.H.Raisoni University Amravati,Maharashtra, India

Abstract: ***The growing use of smart devices and increasing demand for on-demand home services have accelerated the development of digital service platforms. LocalSevaplatform is a web-based solution that connects users with nearby service providers.Also, the platform provides workers with potential employment opportunities. The platform supports a wide range of services, like electrical, plumbing, carpentry, cleaning, appliance repair and home maintenance.***
LocalSeva platform is developed as a full-stack web application using React.js (TypeScript) for the frontend and Node.js with Express.js for the backend and MongoDB as database for data storage. It includes separate dashboards for users and service providers, along with features such as geolocation-based service discovery, time-slot booking, booking management, earnings tracking and review systems.
This paper presents the system design, architecture, key featuresand social impact of LocalSeva platform as a scalable and user-friendly solution for modern local service discovery and delivery.
Keywords: Local Service Platform, On-Demand Services, Service Aggregation, Web Application,Booking System, Geolocation, React.js, Node.js, Express.js, MongoDB

I. INTRODUCTION

The rapid growth of on-demand digital platforms has increased the demand rate among people how they access everyday services. Applications for food delivery and ride-booking have made services more convenient and accessible. However, in developing countries like India, the home service sector remains largely unexplored. Despite of having skilled professionals such as electricians, plumbersand carpenters, users often rely on word-of-mouth, local ads or informal channels to find services.

To fulfil this gap, LocalSeva is introduced as a local service booking platform. Inspired by the Hindi word *"Seva"* (meaning service), the platform focuses on providing reliable and simpler solutions. It enables users to find verified professionals nearby, view ratings and book services through a simple web interface.

Unlike platforms such as Urban Company and TaskRabbit, LocalSeva platform is tailored for Tier-2 and Tier-3 cities, where digital adoption is still evolving. The platform prioritizes ease of use, ensuring accessibility even for first-time users of online services.

LocalSevaplatform also supports service providers by offering tools to manage bookings, track earningsand set availability,helping them build a digital presence. To maintain trust, the platform contains features like user reviews, ratings, verification systems, etc. It is also designed to scale-up with scope for future requirements like AI-based recommendations and real-time tracking.

The rest of this paper is organized as follows: Section II reviews related work, Section III explains the proposed system, Section IV discusses the architecture and technologies used and the remaining sections cover system features, workflow, implementation, impact and conclusion.

II. RELATED WORK

The growth of digital technologies has significantly improved the service industry, especially in on-demand home service platforms. Traditional methods like manual booking and phone-based coordination are often inefficient and time-consuming. Patel and Vaghela showed that online booking systems improve scheduling efficiency and reduce administrative effort [9].

With the rise of mobile and web technologies, several solutions have been developed. Kumar et al. introduced an Android-based service booking app that allows users to select services and schedule appointments easily [8]. However, such applications often face limitations in scalability and cross-platform compatibility.

Web-based systems have further enhanced service platforms. Pandey et al.developed a Django-based system with features like service categorization, provider verification, and payment integration [4]. Similarly, Mishra et al.proposed a centralized platform using MVC architecture and Role-Based Access Control (RBAC) to improve system security and maintainability [1].

©IJRASET: All Rights are Reserved

Digital platforms also play an important role in supporting local service providers. Pawar et al. highlighted that these platforms improve visibility and efficiency but also identified challenges such as low digital literacy and limited infrastructure in non-metro areas [5].

From a hyperlocal perspective, Ganapathy emphasized the importance of location-based services for delivering fast and area-specific solutions, though issues like scalability and logistics remain [6]. Georgiadis et al. demonstrated that Location-Based Services (LBS) enhance personalization by using geographical data [10]. Similarly, Lokithan and Hemamalini developed a MERN-based platform integrating geolocation with service booking [2].

Recent research also focuses on intelligent systems. Kambale et al. applied machine learning techniques to improve service recommendations [3], while Singh et al. highlighted the importance of secure payment systems and user-friendly interfaces in modern platforms [7].

Despite these developments, most existing systems target urban users or involve complex designs. Many lack strong localization and are not well-suited for Tier-2 and Tier-3 cities.

Therefore, there is a clear need for a simple, scalable and accessible local service booking platform. The proposed system, LocalSeva, addresses this gap by combining geolocation-based discovery, interactive interfaces and transparent booking workflows into a unified solution.

TABLE I Comparative Analysis Table

System	Key Features	Limitations	LocalSeva Advantage
MVC Platform [1]	RBAC, structured architecture	Limited personalization	Hyperlocal + user-centric
ServeSpot [2]	Geolocation + booking	Complex system	Simplified UX
ML-Based System [3]	Smart recommendations	High complexity	Lightweight + scalable
Django System [4]	Tracking, payments, verification	Complex, urban-focused	Simple & accessible design
Digital Platforms Study [5]	Business growth enablement	Requires digital literacy	Simplified onboarding
Hyperlocal Model [6]	Fast localized services	Logistics challenges	Structured booking system
Payment Platform [7]	Secure transactions	Limited localization	Hyperlocal + integrated
Android Booking App [8]	Multi-service booking	Limited scalability	Web-based scalable system
Online Appointment System [9]	Scheduling automation	No service marketplace	Full-service ecosystem
LBS Model [10]	Location-aware services	Privacy concerns	Controlled geolocation

III. PROPOSED SYSTEM

The proposed system, LocalSevais a web-based platform designed to connect users with nearby service providers for various home services. As a local service aggregator, it allows users to discover localservice providers based on location, view verified profiles, compare ratings and book services through a simple time-slot-based system. The platform supports ease of use, making it accessible even for first-time users.

Technically, LocalSevaplatform is a full-stack web application built using React.js and TypeScript for the frontend, Node.js with Express.js for the backend and MongoDB as database for data storage. The backend handles core functionalities such as authentication, service listings, bookings, data processing, etc.

The system includes separate dashboards for both users and service providers. Users can manage bookings, track services, provide feedback, etc., while providers can manage availability, handle requests, monitor earnings, etc.

Additionally, geolocation-based service discovery ensures relevant and nearby results, which improves convenience and reduces search time. Features like ratings and reviews provides transparency and trust.

Overall, LocalSevaplatform provides a scalable, user-friendly solution for local service delivery, particularly aimed at non-metropolitanregions.

IV. SYSTEM ARCHITECTURE AND TECHNOLOGY STACK

A. Overall Architecture

LocalSevaplatform is developed using a Single-Page Application (SPA) architecture, ensuring reliable access and continuous availability. The frontend is developed using React.js with TypeScript and Vite, enabling fast development throughmodularapproach. Navigation is handled using React Router DOM, allowing smooth transitions without page reloads.

The system follows a full-stack architecture, where the frontend interacts with the backend through RESTful APIs. The backend is implemented using Node.js and Express.js, managing core functionalities like authentication, service listings, bookingsand data processing. While MongoDB database is used for data storage, providing efficient handling of user and service-related information. Geolocation features are supported through the Nominatim API for accurate location-based service discovery.

To improve scalability and maintainability, the platform uses a component-based design with reusable UI components. State management is handled using React hooks, ensuring efficient data flow and responsive interactions. Performance is enhanced through techniques like lazy loading and code splitting.

The system also follows a modular structure with clear separation of concerns, simplifying development and maintenance. Error handling and fallback interfaces improve reliability, while the setup is suitable for continuous deployment whenever needed and the updates can be made easily. The architecture is designed to fit future requirements such as real-time features, analytics and secure payment integration.

B. Technology Stack

TABLE II Technology Stack of LocalSeva

Category	Technology / Library	Purpose
UI Designing	Figma	Used for well structuring of platform
UI Framework	React.js 18.3 (TypeScript)	Component-based SPA development
Build Tool	Vite 6.3	Fast dev server & optimized bundling
Styling	Tailwind CSS 4.1	Utility-first responsive design
Component Library	Radix UI + shadcn/ui	Accessible UI primitives
Routing	React Router DOM v7	Client-side navigation
Animation	Framer Motion (motion 12)	Scroll-triggered animations
Forms	React Hook Form 7.55	Efficient form handling
Charts / Analytics	Recharts 2.15	Data visualization
Icons	Lucide React 0.487	Icon system
Geolocation	Nominatim OSM API	Location detection
Notifications	Sonner 2.0	Alerts & notifications
Date Handling	React Day Picker + date-fns	Calendar & scheduling
Backend	Node.js + Express.js	API and business logic layer
Database	MongoDB (Mongoose)	Persistent data storage

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

C. System Architecture Diagram

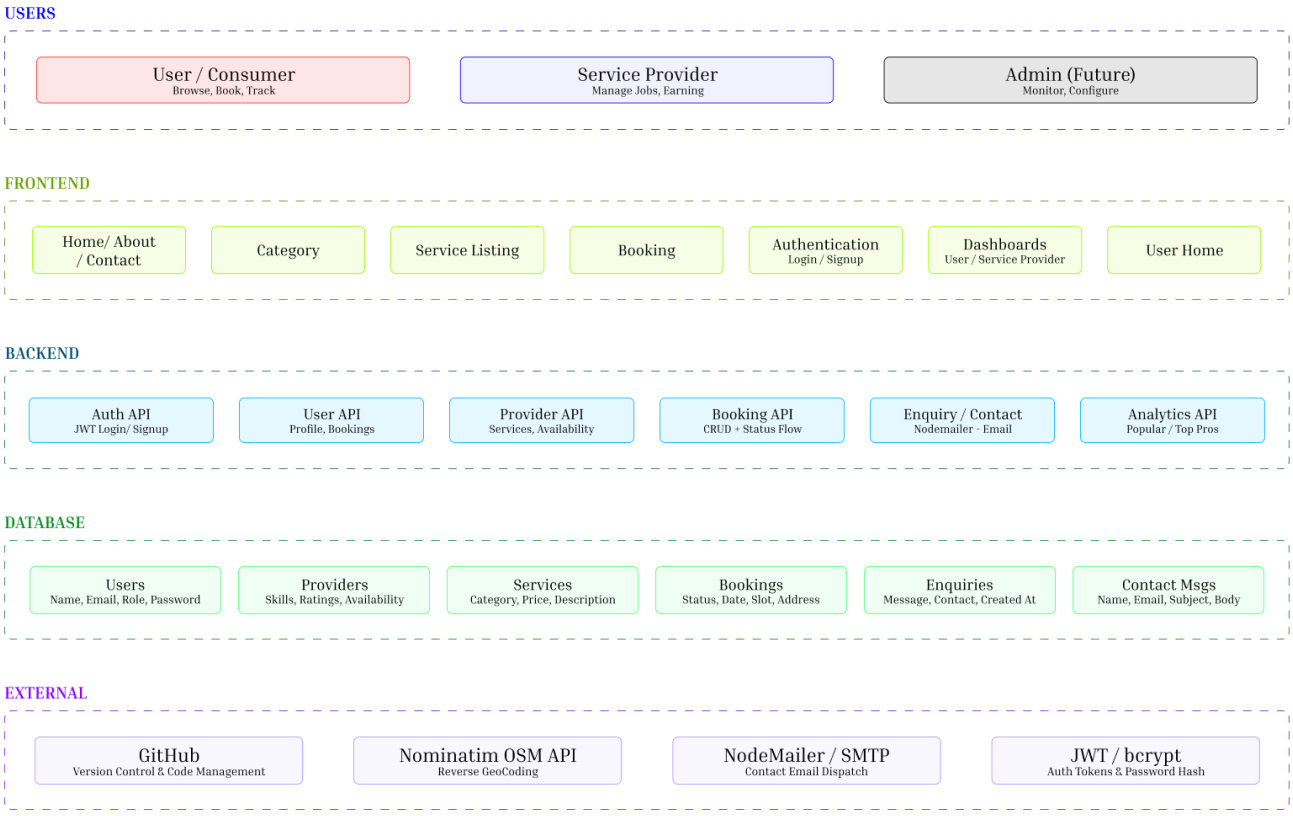


Fig. 1 LocalSeva Full-Stack System Architecture

V. SYSTEM MODULES AND KEY FEATURES

A. Landing and Home Page

The Home Pageis the main page where thenew users visit the website for the firsttime.This page's mainpurpose it to provide the information about the website using images and information. It contains sections like mission and vision, feature highlights, service previews, testimonialsand a call-to-action allowing the users to explore or register.

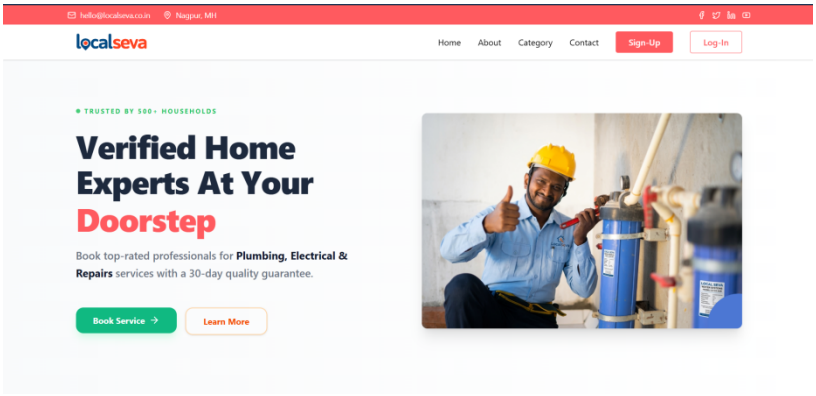


Fig. 2 LocalSeva Landing Page

B. Service Category Discovery

This Page shows different services like electricians, plumbers, cleaning, etc. Users can choose any service available under it. It alsohas the search barto make it easy to find specific service quickly. To move ahead, users need to Login / Signup in their accounts.

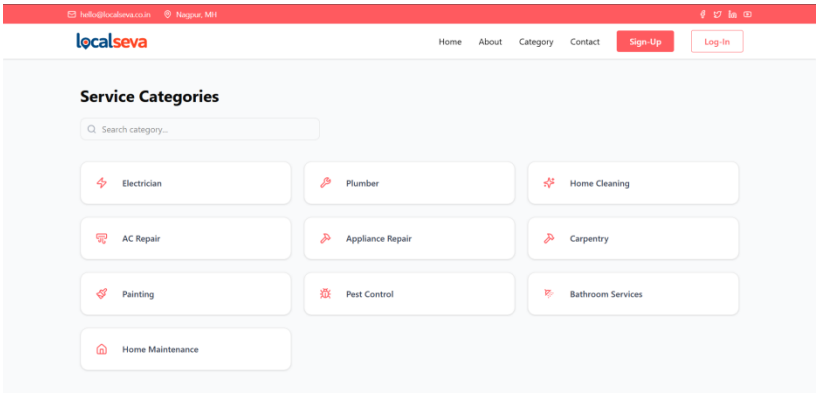


Fig. 3 LocalSeva Service Category Discovery Page

C. Service Provider Listings

This Page shows the list of service providers as per the category selected by user. Each listing card contains details like rating, location, availability, etc. Users can shortlist service providers using filter options like sort by, near me, top rated, etc. and has profile or booking page to know more or continue ahead with the booking.

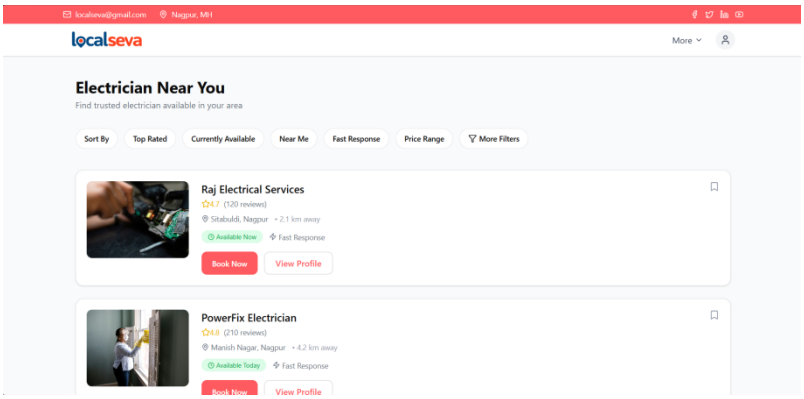


Fig. 4LocalSevaService Providers Listing Page

D. Booking and Geolocation Module

The Booking Module allows users to book services using date and time selection. It also has geolocation and the Nominatim API to detect user's location automatically. The booking summary section shows details like cost, service name, date and time. After successful booking users receives confirmation notifications via email and on manage notification panel in user dashboard to stay updated.

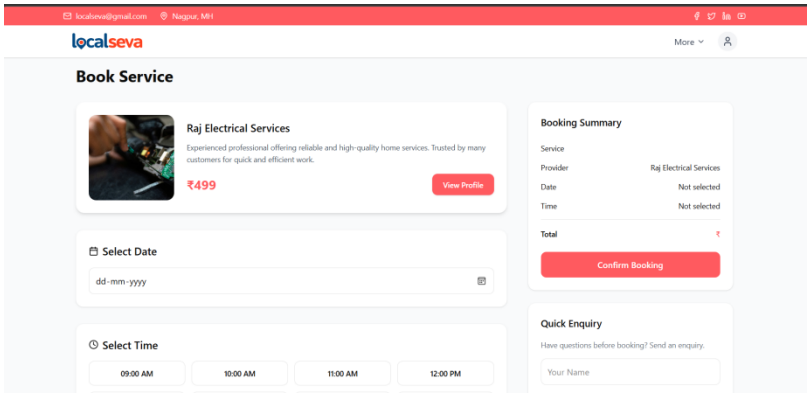


Fig. 5LocalSeva Booking Module

E. User Dashboard

The User Dashboard provides a track of services through panels like manage bookings, view service history, update profiles,notifications, etc. Users can track, reschedule, request revisit or cancel bookings and submit reviews after service completion.

F. Service Provider Dashboard

This Dashboard allows service providers to manage bookings, track earnings, update servicesand set availability. It also let the service provider access a structured booking workflows and revisit requests, improving their service efficiency.

G. Authentication System

The Platform contains a basic authentication system with login, registrationand password recovery features. Session management ensures controlled access to services and dashboardsfor secure management.

H. Static Information Pages

The Platform has utility pages like About, Contact, Privacy Policy and Terms& Conditionsto maintain transparency and user trust. And thereis a header and a footer that provides easy navigation across important pages and sections.

VI. USER ROLES AND SYSTEM WORKFLOW

A. User Roles

The platformhas two main user roles:

1) *Consumer / Customer:*Users who search for services, view provider profiles, book services as per the availability and give ratingsafter service completion.
2) *Service Provider:*Professionals who register, manage services, set availability, handle bookings and track earnings through a dashboard.

B. Consumer Workflow

The consumer's workflow is given as follows:

1) Users visit the platform and explore service categories.
2) Login or registration is required to access detailed listings.
3) A service category is selected.
4) Available providers are displayed with key details like ratings, location, and availability.
5) The user selects a provider and proceeds to booking.
6) Date, time, and address can be entered manually or via geolocation.
7) The booking is reviewed and confirmed.
8) Users manage bookings through the dashboard (track, reschedule, or cancel).
9) After service completion, users can provide ratings and feedback.

C. Service Provider Workflow

The service provider's workflow is given as follows:

1) Registration and profile setup.
2) Adding services, pricing, and availability.
3) Receiving booking requests in the dashboard.
4) Accepting, rejecting, or completing bookings.
5) Managing revisit or complaint requests.
6) Tracking earnings through dashboard insights.
7) Updating services and profile information as needed.
8) Viewing customer details and service history for better understanding.
9) Getting notified about new bookings and updates.
10) Manages daily schedule to avoid complexities in bookings.
11) Viewing customers' queries or revisit requests if needed.
12) Keep track about ratings and feedbacksfor improvement in service quality.

©IJRASET: All Rights are Reserved

D. Service Booking Flowchart

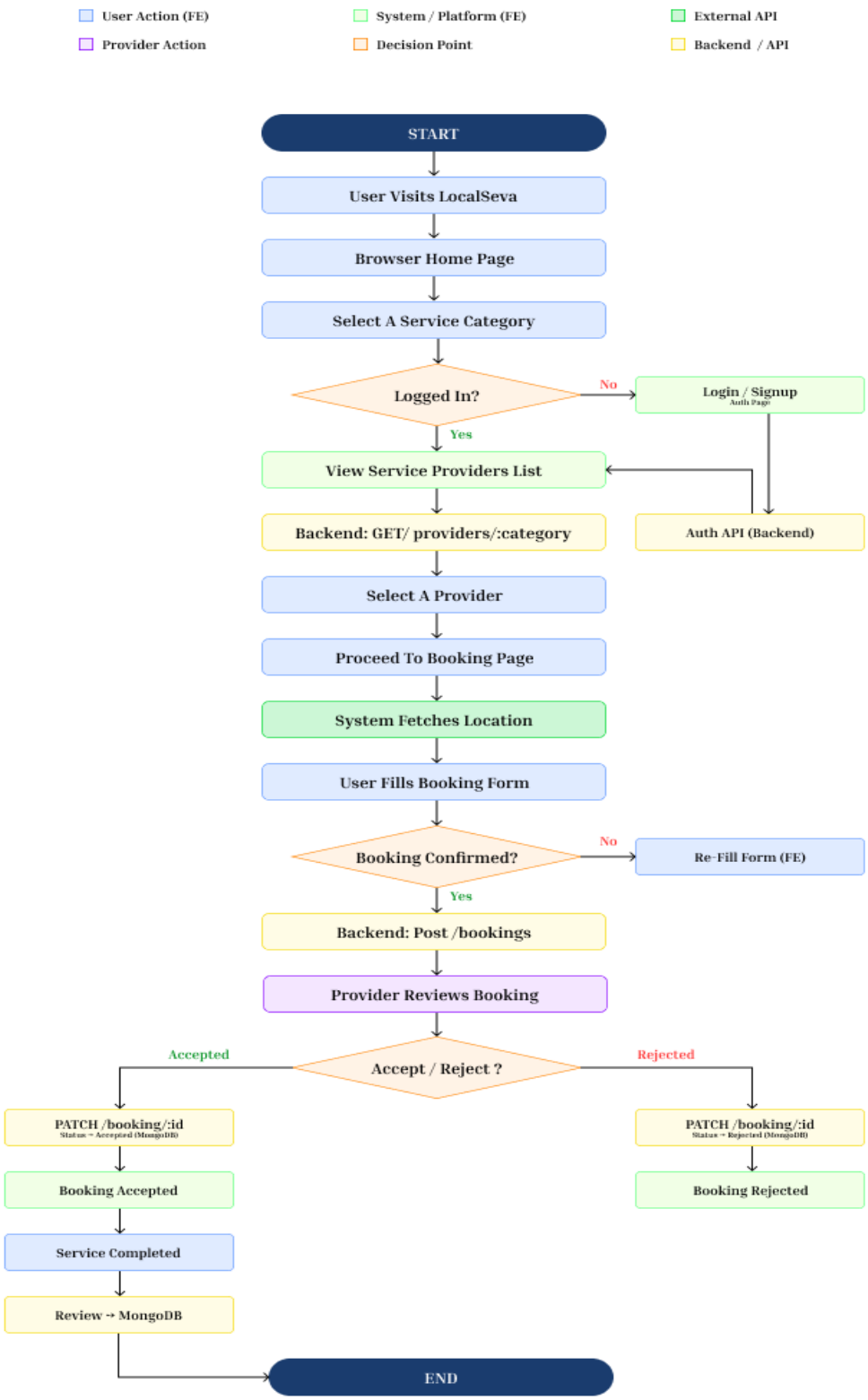


Fig. 6 Full-Stack Service Booking Flowchart

VII. SYSTEM DESIGN AND BACKEND ARCHITECTURE

A. Routing Architecture

The platform uses a dynamic routing system developedusing React Router DOM. Routes are used to manageconditional renderingof components like headers and footers. Dashboard pages use a simplified layout for better focus. The routing structure has pages like home, categories, provider profiles, booking, authentication, dashboards, static pages, etc.allowing smooth navigation without page reloads. The defined routes are mentioned below:

TABLE III Application Routes and Corresponding Components

Route Path	Component / Description
/	Home – Landing page
/about	About – Company information & FAQ
/contact	Contact – Service inquiry form
/category	CategoryPage – Service category grid
/services/:serviceName	ServiceProvidersPage – Dynamic provider listing
/services/home-service	HomeService – Home service detail page
/provider/:id	ViewProfile – Provider public profile
/booking/:id	BookingPage – Appointment booking form
/login	LoginPage – User authentication
/signup	SignupPage – New user registration
/forgot	ForgotPasswordPage – Password recovery
/userdashboard	UserDashboard – Customer management portal
/userhome	UserHome – Logged-in user home/dashboard landing
/serviceproviderdashboard	ServiceProviderDashboard – Provider portal
/privacy	PrivacyPolicy – Data privacy page
/terms	TermsConditions – Terms of service

B. Component Architecture

The frontend has component-based structure which includes core layout components like pageviews, feature modulesand reusable UI elements. This structure helpsto improvemaintainability, consistency and reusability of all the components and pages across platform.

C. State Management

State is managed using React hooks to maintain data flow and easy handling of user interactions across components. This makes it easier to maintain the interface smoothness whenever users perform actions such as booking a service or navigating between different pages.

D. Responsive Design

The platform was developed to be accessible across different devices to ensure compatibility. The Tailwind CSS is used formaking layouts responsive, while dashboards are flexible for smaller screens with adaptive navigation patterns like hamburger.

E. Backend Overview

The backend is developed using Node.js and Express.js. It handles core functionalities like authentication, service management, bookings, analytics, etc. It also processes user data to identify popular services and top-providers. It also manages secured data handling and API integration to maintain a reliable system.

F. Database Design

The MongoDB database is used to store collections for users, providers, services, bookingsand enquiries. Thebackend has a modular structure (models, controllers, routes, middleware) that maintain scalability and efficiency of data management.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

G.REST API Structure

The system uses RESTful APIs for connecting frontend and backend. They are used to manage authentication, services, bookings,user operations, etc., allowing a structured and maintainable workflow.

H. Authentication and Security

Security is providedwith the help of JWT authentication. User data is protected by hashingand middleware enforces role-based access control to secure sensitive operations.

I. Email Notification System

An email system is added to manage user queries and notifications. Requests are forwarded to database and trigger automated email responses to users.

J. Analytics: Popular Services and Top Professionals

The platform contains analytics systemthat highlights popular services (based on bookings) and top providers (based on ratings and reviews), this helps user with decision-making and engagement.

VIII. SYSTEM DEVELOPMENT

TheLocalSeva is developed to provide structured and simpler approachwith the aim to give a scalable and user-friendly platform. The process started with requirement analysis, where key features like service discovery, booking, user roles and backend processing were analysed. Based on requirements as per analysis, a modular system architecture was designed to ensure flexibility and ease of future updates required.

The development of frontend is done using React.js and TypeScript, allowing components' reusability and maintenance. Responsive design was achieved using Tailwind CSS and custom modular CSS to make the platform device responsive. The backend isdeveloped using Node.js and Express.js, with RESTful APIs handling authentication, service management, bookingsand database interactions.

Development was carried out in stages, where individual modules were developed and tested before integrating them. MongoDB is used for data storage, efficiently managing user, provider, booking data, etc. Also, continuous testing and debuggingis done for system reliability, performanceand smooth user experience. Version control tools that is GitHub supported efficient code management.

The final system is developed to meet real user's need, ensuring real-world usability and availability.

Overall, the development was majorly focused upon scalability, reliability and maintainability. The platform is designed to support future improvementslike real-time features, secure payments and advanced analytics.

IX. SOCIAL IMPACT AND FUTURE SCOPE

A. Social Impact

LocalSevaplatform creates a positive impact by connecting service providers and customers in practical ways. For local service providers from informal sectors, it gives them opportunity to come online, reach more people and earn regularly. It also helps bring some structure to their work and allows them to build trust through ratings and past service records.

For customers from Urbancities, it makes it much easier to find dependable professionals without spending too much time searching. It lowers the chances of choosing unverified services and improves overall quality through clear pricing and reviews, making services more reliable and easier to access. It also saves time by offering multiple services in one place, so users don't have to look elsewhere. At the same time, it helps improve communication between customers and providers, making the whole service process smoother.

B. Future Scope

Future enhancements of platform include:

1) *Real-time Communication:*Instant updates and messaging using technologies like Socket.io.
2) *Payment Gateway Integration:*Secure payments via platforms such as Razorpay or Stripe.
3) *AI-Based Recommendations:*Personalized service suggestions based on user behavior and location.
4) *Mobile Application:*Expansion to mobile apps using Flutter or React Native.

5) *Multi-language Support:*Inclusion of Hindi and regional languages for wider accessibility.
6) *Provider Verification:*Advanced identity and certification validation for improved trust.
7) *Advanced Analytics:*Insights on demand trends, service performance, and earnings.
8) *Community Features*: Forums and knowledge-sharing tools to enhance user engagement.

X. CONCLUSIONS

This paper presented LocalSeva platform, a web-based service platform designed to connect users with service providers even from the non-metro regions of India. It is developed using React.js, Node.js, Express.jsand MongoDB.The platform provides a complete and scalable solution for service discovery via listings and booking.

LocalSeva platform focuses upon simplesolution, that offers a user-friendly and device-responsive interface. It allows consumers to easily find reliable local professionals, whileservice providers are provided with the tools to manage bookings, services, earnings, etc. There are features like verified profiles, transparent pricing and user reviews which helps to build trust among users and improve service quality.The platform has scalable architecture thatsupports future improvements, including real-time communication, secure payments, AI-based recommendations, etc. By connecting local service providers with users, LocalSevaplatform contributes to digital inclusion and creates new economic opportunities.

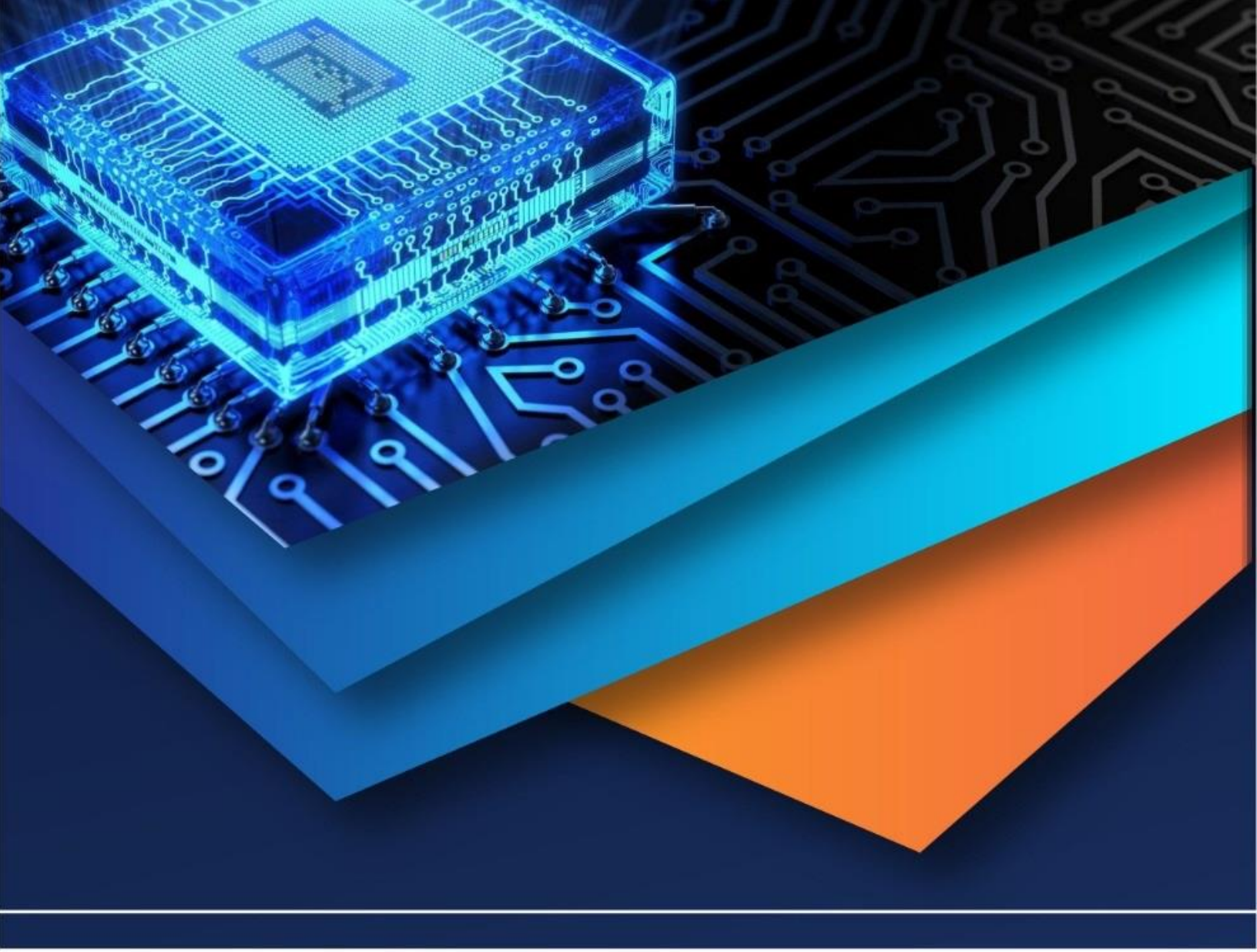
Overall, LocalSeva platform demonstrates how technology can help modernize traditional service sectors, with the flexibility of adapting future needs through advanced features and wider accessibility.

XI. ACKNOWLEDGEMENT

This project required significant time, effortand teamwork, and we are grateful to everyone who supported us throughout the process. Our guide was always there to help us when we faced difficulties and provided practical suggestions that guided us in the right direction. There were several instances where we had to rethink our approach and correct our mistakes and those experiences helped us gain a deeper understanding. Working together as a team came with its own challenges, but through open discussions and proper division of tasks, we were able to manage everything effectively. We would also like to thank our college for providing the necessary resources and environment to carry out this work. Lastly, we sincerely appreciate the support of our friends and family, who encouraged us and stood by us during demanding and stressful times.

REFERENCES

[1] V. Mishra et al., “Service Booking Platform,” International Journal of Progressive Research in Engineering Management and Science, vol. 6, no. 1, 2026.
[2] G. R. Lokithan and Hemamalini, “ServeSpot: A Location-Based Service Discovery and Booking Platform,” International Journal of Current Science, vol. 16, no. 1, 2026.
[3] S. N. Kambale et al., “Home Service Providing System Using Machine Learning,” International Journal of Science, Engineering and Technology, vol. 13, no. 3, 2025.
[4] S. K. Pandey et al., “A Comprehensive Approach to Home Service Booking System Development in a Django Web Application,” International Journal of Research - GRANTHAALAYAH, vol. 13, no. 4, 2025.
[5] C. Pawar et al., “The Role of Digital Platforms in Scaling Local Service Businesses in India,” International Journal of Scientific Research in Engineering and Management, vol. 9, no. 11, 2025.
[6] V. Ganapathy, “Hyperlocal Delivery: Leveraging Technology to Serve Local Communities,” Journal of Management and Entrepreneurship, vol. 18, no. 4, 2024.
[7] Singh et al., “Online Service Booking Platform with Payment Integration,” International Journal of Information Technology, Research and Applications, vol. 2, no. 2, 2023.
[8] K. Kumar et al., “Development of Online Service Booking Mobile Application Using Android Studio,” International Journal of Innovative Research in Engineering & Management, vol. 9, no. 3, 2022.
[9] T. Patel and S. Vaghela, “Online Appointment Booking System,” International Journal for Research in Engineering Application & Management, vol. 4, no. 11, 2019.
[10] C. K. Georgiadis, G. Melas, and P. Tzamtzis, “Location Based Services for Mobile Commerce Applications,” Proceedings of the Second International Conference on e-Business, 2007.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)