

ShramEasy: An AI-Enabled Marketplace for Trusted Construction and Skilled-Labor Services

Ravish Kumar Dubey¹, Om Agarwal², Magan Mehrotra³, Naman Saxena⁴, and Praveen Agarwal⁵

¹Assistant Professor, Department of Computer Science and Engineering, Moradabad Institute of Technology, Moradabad, India
Email: ravishkrdubey@gmail.com

²Department of Computer Science and Engineering, Moradabad Institute of Technology, Moradabad, India
Email: om.agarwal674@gmail.com

³Department of Computer Science and Engineering, Moradabad Institute of Technology, Moradabad, India
Email: maganmehrotra.2004@gmail.com

⁴Department of Computer Science and Engineering, Moradabad Institute of Technology, Moradabad, India
Email: namanritz22@gmail.com

⁵Department of Computer Science and Engineering, Moradabad Institute of Technology, Moradabad, India
Email: siddhantagarwal88@gmail.com

Abstract

Finding reliable construction and repair professionals in India remains difficult due to fragmented information, lack of trust, non-transparent pricing, and absence of verifiable credentials for workers. ShramEasy is a multi-platform web and mobile application that creates an end-to-end digital marketplace for construction and skilled-labor services, focusing on trust, transparency, and automation. The system supports user registration, job posting, AI-based matching, dynamic pricing, milestone-based escrow payments, review and rating mechanisms, complaint resolution, and analytics dashboards for all stakeholders. ShramEasy leverages a modern technology stack including React Native, React/Angular, Node.js/Express or Python/Django REST, and PostgreSQL/MongoDB, deployed on cloud platforms such as AWS or Azure, with integrated payment gateways and notification services. By combining secure KYC-backed onboarding, verified feedback, and intelligent matching, the platform aims to reduce delays, disputes, and information asymmetry between customers and service providers. The paper presents the problem context, related work, system design, module-wise description, implementation technologies, and expected societal and economic impact of ShramEasy.

Keywords: Service marketplace; Construction services; AI-based matching; Dynamic pricing; Secure payment gateway; Mobile application; Cloud computing.

I. Introduction

Finding dependable construction and repair services is a persistent challenge for homeowners and property managers, who often rely on informal networks, unverified listings, and inconsistent recommendations. This leads to delays, cost escalation, safety risks, and frequent disputes caused by unclear expectations and weak accountability mechanisms. Existing digital platforms in India focus largely on general home services and local listings, but they seldom offer specialized support for construction-focused workflows, verified worker profiles, and milestone-based payments. Moreover, current solutions struggle with issues such as review

manipulation, non-transparent pricing, and limited integration with formal KYC and skill verification processes.

ShramEasy addresses these gaps by offering an integrated marketplace that connects customers with trusted construction and skilled-labor professionals through AI-driven matching and secure, staged payments. The system targets homeowners, property managers, small contractors, and individual workers, providing them with transparent discovery, booking, and dispute-resolution features through unified web and mobile interfaces.

A. Problem Statement

The construction and repair services sector in India is characterized by several critical inefficiencies. First, discovering trustworthy service providers remains challenging; potential customers often resort to word-of-mouth recommendations or fragmented online listings that provide limited information about provider credentials. Second, pricing opacity is common, with service providers quoting different rates for identical work without clear justification. Third, the lack of formal verification mechanisms leaves customers vulnerable to working with unqualified or unreliable professionals. Fourth, dispute resolution is weak; when conflicts arise regarding payment, quality of work, or timeline, no structured mechanism exists to handle complaints fairly. Finally, service providers themselves struggle with inconsistent income, lack of formal income documentation, and limited access to a stable customer base.

B. Industry and Society Benefitted

The primary beneficiaries of ShramEasy include homeowners and property managers seeking reliable construction and repair services, small building contractors managing projects, and skilled workers and craftspeople seeking stable employment and formalized income documentation. Homeowners benefit by reducing search costs and risks associated with hiring unknown workers. Property managers gain access to pre-vetted professionals and structured service delivery. Workers benefit through improved income stability, transparent payment mechanisms, and opportunities to build verifiable reputation and skills portfolios.

C. Objectives

The primary objectives of ShramEasy are to digitalize the full life cycle of labor-based construction services, to build trust with verified profiles and reviews, and to support scalable, data-driven decision-making for platform administrators. The system aims to reduce delays, disputes, and information asymmetry between customers and service providers by implementing verified onboarding, transparent pricing, secure milestone-based payments, and accessible dispute resolution.

II. Related Work and Existing Solutions

Research on service marketplaces highlights the importance of intelligent matching, pricing optimization, and user trust in on-demand platforms. Prior work on AI in service marketplaces shows how machine-learning models can use historical booking data, user preferences, and provider performance to recommend suitable professionals and improve platform efficiency [1]. Such models typically incorporate features such as provider skill tags, customer location, project scope, and past booking outcomes to generate personalized recommendations.

Dynamic pricing studies for on-demand services demonstrate that context-aware pricing models can balance demand and supply, but many existing systems do not expose transparent logic to end users. Research indicates that transparent dynamic pricing can build user trust while maintaining revenue optimization [2]. Similarly, secure payment gateway research emphasizes escrow mechanisms and fraud-resistant workflows as critical for online marketplaces, especially when transactions involve multi-stage projects and large ticket sizes [3].

Feedback and review systems in online platforms face persistent challenges with manipulation, incomplete feedback, and weak linkage between reviews and verified job completion. Studies show that verification of

reviewer actions and non-editability of reviews significantly reduce fraud and improve platform integrity [4]. Government initiatives and skill registries in countries such as India increasingly support formal documentation of worker skills and credentials, creating an opportunity for platforms to integrate official verification into onboarding workflows [5].

Commercial platforms such as Urban Company (formerly UrbanClap) and Thumbtack provide general home services and matching, yet they typically offer broad categories rather than construction-specific flows with milestone-based contracts, structured dispute handling, and deep integration of KYC/skill verification. Payment infrastructure studies document the maturity and availability of secure gateway providers such as Razorpay, Paytm, and Stripe, which support escrow functionality and local payment methods. However, these services must be orchestrated into domain-specific workflows to fully address the needs of construction projects.

A. Gaps in Existing Solutions

Existing platforms exhibit several shortcomings. They lack tailored AI-based matching optimized for construction projects, instead applying general home-service matching logic. Payment methods are often scattered across multiple systems without true escrow-like milestone release functionality. Review and feedback systems suffer from manipulation and incomplete verification. Most critically, there is limited or no integration with government skill registries, background verification processes, or formal KYC onboarding tailored for construction workers. ShramEasy aims to fill these gaps through purpose-built architecture and AI workflows.

III. System Overview and Architecture

ShramEasy is conceived as a multi-platform system comprising mobile applications, a responsive web interface, and a centralized backend that exposes RESTful APIs. The primary actors are customers seeking services, service providers offering skilled labor, and administrators who configure platform rules, commissions, and dispute policies.

A. High-Level Workflow

At a high level, the workflow begins when a user registers through email, phone, or social login, optionally completing KYC verification to increase trust. Customers can then browse or search for services by category, location, skill, price range, availability, and rating, and they may post jobs that specify scope, timeline, and budget constraints.

An AI-based matching engine processes job details, provider skills, location, and historical data to suggest ranked providers to customers. Once a booking is confirmed, the system creates a project record with defined milestones, each linked to a partial payment locked in an escrow-like wallet that releases funds upon milestone approval.

The architecture includes separate components for authentication, service catalog management, booking and scheduling, payment processing, notifications, and analytics. Admin dashboards provide aggregated views of platform activity, revenue, commissions, dispute statistics, and provider performance, enabling data-driven governance.

B. Key Components and Data Flow

Authentication & User Management: Secure user registration, sign-in, KYC data collection, and session management using industry-standard protocols.

Service Catalog & Search: Searchable categorization of services, filtering by skills, location, price, and ratings; powered by indexing and full-text search engines.

Matching Engine: AI/ML module that analyzes job requirements and provider profiles to produce ranked recommendations based on skill match, location proximity, historical performance, and availability.

Booking & Scheduling: Transaction initiation, milestone definition, timeline setting, and confirmation workflow between customer and provider.

Payment Processing: Integration with payment gateways, escrow wallet management, milestone verification, and fund release orchestration.

Notifications & Communication: SMS, email, and push notification delivery via third-party providers; in-app chat between customers and providers.

Analytics & Admin Dashboard: Platform performance metrics, user and provider statistics, revenue reporting, and complaint tracking.

IV. Module Description and Workflow

Module	Description
Registration & Login	Email, phone, or social sign-up with optional KYC and background verification
Service Browsing & Filtering	Search and filtering by category, location, skill, price, availability, and rating
Service Booking	Instant, scheduled, or project-based booking with dynamic pricing
Payment Wallet	Multi-gateway UPI, cards, net banking with escrow and milestone-based fund release
Feedback System	Verified post-completion ratings/reviews and complaint submission
Provider Profile Management	KYC, skill and certificate uploads, background check, service listing management
Provider Job Requests	Accept/reject job offers, view current and future bookings, project status management
Earnings Dashboard	Revenue tracking, commission monitoring, withdrawal processing
Admin Controls	User and provider approval, complaint handling, analytics, commission configuration
Notification System	Real-time SMS, email, and in-app alerts for job updates and transactions
Chat & Support	In-app messaging between customer and

	provider, AI FAQ bot
GPS Tracking	Live provider location updates during service
Subscription & Promotions	Premium highlights and featured listings for providers

Table 1: ShramEasy Module Overview

A. Registration and Login

The Registration and Login module supports sign-up and sign-in through email, phone, or social accounts (Google, Facebook), with optional KYC flows for providers and high-value customers. Validation rules enforce unique identifiers, strong authentication, and secure handling of identity documents used for background checks and skill verification. Providers undergo a multi-step KYC process that includes identity verification, address confirmation, and skill certification uploads. Customers may optionally verify their identity to gain a "Verified Customer" badge, which improves trust with providers.

B. Service Browsing and Filtering

The Service Browsing and Filtering module allows users to explore service categories such as masonry, plumbing, electrical work, painting, carpentry, and other skilled trades. Search filters include geographic location (with radius-based filtering), skill level or specialization, hourly or project-based pricing, real-time availability, and provider star ratings. Advanced filtering allows users to sort by price, distance, and customer reviews, thereby simplifying discovery of relevant labor.

C. Service Booking

Through the Service Booking module, customers can place instant (same-day), scheduled (future date), or project-based requests specifying start times, work location, scope, and duration. Estimated costs are displayed transparently, calculated based on provider hourly rate or project-based pricing. For project-based work, the system suggests milestone breakpoints (e.g., 30% payment at design phase, 40% at execution start, 30% upon final completion) which both parties can customize. The dynamic pricing component may adjust recommended prices based on demand, provider rating, time-of-day, and urgency while still presenting transparent estimates and justification to users.

D. Payment and Wallet Management

The Payment and Wallet module integrates multiple gateways such as UPI, credit/debit cards, and net banking, managed through secure API connections to payment providers (Razorpay, Paytm, Stripe). Each booking creates an escrow-style wallet entry that reserves funds at the time of booking. Funds are released according to milestone completion; a customer must approve each milestone (or a timeout mechanism auto-releases after confirmation) before money moves from escrow to the provider's withdrawal account. Clear logs of transactions, commissions deducted by the platform, and withdrawal history are visible to both customers and providers.

E. Feedback and Review System

A Feedback and Review System records ratings and textual reviews only after verified job completion, linking them to specific bookings to reduce manipulation and fake reviews. Reviews are non-editable once submitted, and the system marks reviews as "Verified Completion" only if the job milestone was marked complete and the review was submitted within a defined window. Complaints and dispute tickets can be raised by either party, which are then routed to admins who can view booking history, chat logs, evidence (photos, documents), and communication records before deciding on resolution (refund, re-work, compromise).

F. Provider Profile and Job Management

Providers can create and manage their profiles, including uploading skills, certifications, work samples, and professional documents. They can accept or reject job offers received through the matching engine, view their current bookings and future scheduled work, update project status in real-time, and upload completion evidence such as photos. A provider dashboard shows their average rating, total completed jobs, earnings, and customer feedback.

G. Admin Dashboard and Controls

Admin modules provide user and provider onboarding approval, policy configuration (commission rates, category definitions), complaint handling workflows, and platform-wide analytics (daily/weekly/monthly active users, bookings, revenue, average rating distribution, complaint resolution time). Admins can also manage suspicious accounts, fraud detection alerts, and promotional campaigns.

V. Implementation Technologies and Feasibility

A. Technology Stack

Frontend Mobile: React Native is employed to develop iOS and Android applications from a single codebase, reducing development time and maintenance overhead while ensuring consistent user experience across platforms.

Frontend Web: React.js or Angular frameworks are used to build responsive web dashboards for customers, providers, and administrators, with real-time updates via WebSocket connections.

Backend Services: Node.js with Express framework or Python with Django REST Framework creates RESTful APIs that handle user management, job matching, payment coordination, notifications, and analytics. Microservices architecture can isolate critical services (payment, notification) for independent scaling and resilience.

Database: PostgreSQL is used for transactional data (users, bookings, payments, disputes) requiring ACID compliance and relational integrity. MongoDB or similar NoSQL databases store semi-structured data such as job descriptions, analytics logs, and user activity feeds.

Cloud Platform: AWS (EC2, RDS, S3, Lambda) or Azure (VMs, SQL Database, Blob Storage, Functions) provides scalable infrastructure with managed services for databases, object storage, and serverless computing. Containerization using Docker and orchestration via Kubernetes enables efficient deployment and auto-scaling.

Payment Integration: Razorpay, Paytm, and Stripe APIs are integrated to handle online transactions, escrow hold/release, refund processing, and compliance with Indian payment regulations and RBI guidelines.

Notifications: Twilio (SMS), SendGrid (email), and Firebase Cloud Messaging (push notifications) provide reliable, scalable channels for notifying users of job updates, payment events, and platform announcements.

AI/ML Framework: Python-based libraries such as scikit-learn, TensorFlow, or PyTorch power the recommendation and matching engine, which can be deployed as a dedicated microservice or as a scheduled batch job.

B. Security and Compliance

All data in transit is encrypted using TLS 1.2+. User credentials are hashed using bcrypt or Argon2. API endpoints are protected with OAuth 2.0 or JWT-based authentication. KYC documents are encrypted at rest and stored in secure cloud vaults with access logging. Payment data handling complies with PCI DSS

standards. User privacy is ensured by implementing clear data retention policies and compliance with India's Digital Personal Data Protection Act (DPDPA).

C. Feasibility Analysis

Economic Feasibility: The technology stack relies heavily on open-source frameworks and languages (React, Node.js, Python, PostgreSQL). Cloud infrastructure operates on a pay-as-you-go model, scaling costs with user growth. Payment gateway fees and notification service costs are variable, typically funded through platform commission on bookings. Initial development cost and MVP deployment are manageable within a small team budget.

Technical Feasibility: All proposed modules leverage mature, well-documented technologies with large developer communities. React Native is proven for cross-platform mobile development. Node.js/Express and Django REST are production-ready backend frameworks. Payment gateway APIs and notification services are fully documented and widely used. Scalability is achievable through cloud auto-scaling, database replication, and microservices decomposition. The matching algorithm can start with simple rule-based logic and evolve to machine-learning models as data accumulates.

Operational Feasibility: User onboarding flows are designed to be intuitive and guided. Admin dashboards employ standard controls and visualizations. Customer support can initially leverage the in-app FAQ bot and escalation to admins. Operational procedures for new service categories, commission updates, and dispute policies can be configured without code changes through an admin backend interface.

Key Challenges: Scaling AI-based matching and dynamic pricing algorithms to handle thousands of concurrent jobs and provider searches requires careful optimization. Payment dispute handling demands clear policies and efficient admin workflows. Ensuring review authenticity while preventing manipulation requires continuous monitoring. Market adoption and network effects depend on marketing investment and initial provider recruitment.

VI. Results, Discussion, and Expected Impact

A. Prototype Validation and Performance Metrics

In its initial implementation, ShramEasy can be evaluated through prototype screens demonstrating the end-to-end flow from registration to final payment. Functional tests validate correctness of AI-based matching (comparing recommendations against manually ranked providers), reliability of milestone handling (creating, approving, and releasing funds for each stage), and robustness of complaint management workflows under typical and edge-case scenarios.

Performance targets include:

- **Search Latency:** < 500 ms for service search with 5 filters
- **Matching Engine:** Ranking 10 best providers within 1 second for a new job
- **Payment Processing:** Transaction confirmation within 30 seconds
- **Notification Delivery:** SMS/email delivery within 2 minutes of event
- **API Availability:** 99.5% uptime for core APIs

B. System Evaluation

Logging and analytics infrastructure tracks system health, detects failures, and refines matching and pricing algorithms over time. A/B testing can evaluate the effectiveness of different recommendation strategies, pricing models, and UI variations on user conversion and satisfaction. Early feedback from beta users

(homeowners, contractors, and workers) informs iterative refinements to module workflows and feature prioritization.

C. Societal and Economic Impact

For Customers: ShramEasy improves access to verified, skilled labor, reducing dependence on informal intermediaries and unverified recommendations. Transparent pricing and milestone-based payments lower financial risk. Structured dispute resolution and documented work history increase confidence in hiring decisions.

For Service Providers: Workers gain improved visibility to a broader customer base, stable income tracking, and protection against payment default through escrow mechanisms. Accumulated ratings and work history build verifiable professional credentials, enabling access to better-paying jobs and potential formal employment pathways. Access to micro-credit or financial services may become feasible with documented income history.

For Society: Formalization of the informal labor market reduces tax evasion and labor exploitation. Skills documentation supports government workforce planning and targeted training initiatives. Data aggregated on skills demand, wage trends, and service gaps informs policy design in urban planning and labor development. Improved labor productivity through better matching and reduced search costs can contribute to economic growth.

D. Market Potential

The construction and home services market in India is estimated at several billion dollars annually, with significant informal components. Even capturing 5-10% of formal market bookings would represent substantial business opportunity. The scalability of the marketplace model and AI-driven matching suggests potential for geographic expansion (tier-2 and tier-3 cities) and category expansion (beyond construction to plumbing, electrical, repair, etc.).

VII. Conclusion and Future Work

ShramEasy presents an integrated, AI-enabled marketplace that addresses key pain points in construction and skilled-labor services by combining verified onboarding, intelligent matching, and milestone-based secure payments. The system design covers multiple interconnected modules—including registration, browsing, booking, payments, feedback, analytics, and admin controls—built on a scalable web and mobile technology stack leveraging cloud infrastructure, modern APIs, and established payment and notification services.

The initial feasibility analysis suggests that the proposed architecture is technically sound, economically viable for iterative development, and operationally manageable with standard tooling and practices. The multi-stakeholder design ensures that all parties (customers, providers, administrators) have aligned incentives to build a trustworthy and efficient marketplace.

A. Future Enhancements

Government Integration: Deeper integration with government skill databases and KYC registries to streamline verification and enable worker subsidies or training programs.

Predictive Analytics: Machine-learning models to forecast demand trends, recommend training areas, and enable intelligent resource allocation and workforce planning.

Expansion: Scaling to additional service categories (plumbing, electrical, HVAC, general repairs) and geographic regions, adapting the platform for tier-2 and tier-3 cities with lower broadband penetration.

Enterprise and B2B Features: Support for large contractors managing multiple sub-teams, project analytics, and budget tracking; integration with construction management software.

Financial Services: Partnerships with microfinance institutions or banks to offer credit to verified providers, enabling them to invest in tools and training.

Autonomous Dispute Resolution: Advanced ML-based analysis of chat logs, photos, and feedback to recommend resolutions, reducing manual admin overhead.

Acknowledgment

The authors gratefully acknowledge the support of Moradabad Institute of Technology, Department of Computer Science and Engineering, and the guidance of faculty advisors. We thank the broader development and open-source communities for the frameworks, libraries, and services that make projects such as ShramEasy feasible.

References

- [1] A. Bhatia et al., "Artificial Intelligence in Service Marketplaces: A Survey," *IEEE Access*, vol. 10, pp. 45678–45695, 2022.
- [2] M. Jain and A. Kumar, "Dynamic Pricing for On-Demand Service Platforms," *Procedia Computer Science*, vol. 173, pp. 234–241, 2021.
- [3] N. Upadhyay, "A Review on Secure Payment Gateway Integration for Indian E-Commerce," *Int. J. of Computer Applications*, vol. 180, no. 35, pp. 12–18, 2020.
- [4] R. Gupta, S. Singh, and P. Verma, "Escrow-Based Transaction Models for Online Marketplaces," *Int. J. of Information Management*, vol. 45, pp. 102–115, 2019.
- [5] G. Kaur, "Role of Feedback and Reviews in Online Platforms," *Journal of Digital Business*, vol. 8, no. 2, pp. 55–72, 2018.
- [6] Razorpay, "Payment Gateway API Documentation," Online. Available: <https://razorpay.com/docs/>
- [7] Twilio, "SMS and Communication APIs Documentation," Online. Available: <https://www.twilio.com/docs>
- [8] Firebase, "Cloud Messaging and Real-Time Database," Online. Available: <https://firebase.google.com/docs>
- [9] Government of India Ministry of Labour, "Skill India Mission and Verification Protocols," Online. Available: <https://www.skillindia.gov.in>
- [10] Urban Company, "Service Marketplace Case Study and Platform Overview," Online. Available: <https://www.urbancompany.com>