International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 02 | Feb - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

AI-Integrated Jewelry E-Commerce Platform: A Full-Stack Web Application with Generative Design, Conversational AI, and Real-Time Market Intelligence

Harsh Mali[1], Arsh Naikawadi[2], Aryan Kadam[3], Om Salunkhe-Patil[4], Sahil Jagtap[5]

[1] Dept. of Computer Engineering, Nanasaheb Mahadik Polytechnic Institute Peth, A/P: Nagthane, Sangli, Maharashtra, India, er.maliharsh95@gmail.com

[2] Dept. of Computer Engineering, Nanasaheb Mahadik Polytechnic Institute Peth, A/P: Dapoli, Ratnagiri, India, arshnaikawadi@gmail.com

[3] Dept. of Computer Engineering, Nanasaheb Mahadik Polytechnic Institute Peth, A/P: Karad, Satara, Maharashtra, India, aryankadam2706@gmail.com

[4] Dept. of Computer Engineering, Nanasaheb Mahadik Polytechnic Institute Peth, A/P: Nagthane, Sangli, Maharashtra, India, ompatil151107@gmail.com

[5] Dept. of Computer Engineering, Nanasaheb Mahadik Polytechnic Institute Peth, A/P: Mitharwadi, Kolhapur, Maharashtra, India, sahiljagtap1123@gmail.com

***Guide:** Mrs.Madhuri Kamble, Dept. of Computer Engineering, Nanasaheb Mahadik Polytechnic Institute Peth, A/P: Walwa, Sangli, Maharashtra, India, madhurikamble848@gmail.com*

ABSTRACT

The Indian jewelry retail sector, valued at over USD 85 billion, continues to rely on conventional storefronts with limited digital engagement, resulting in poor customer reach, opaque pricing, and absence of personalized design capabilities. This paper presents the design, implementation, and evaluation of a full-stack, AI-integrated web application developed for Amar Jewellers, a heritage jewelry retailer based in Sangli, Maharashtra. The proposed system employs a modern three-tier architecture comprised of a React-TypeScript single-page application (SPA) on the frontend, an Express.js RESTful API server on the backend, and MongoDB as the persistence layer. The platform introduces three novel capabili- ties for traditional jewelry retailers: (1) an AI-powered Design Studio that leverages the Gemma 3 large language model via the Bytez inference API to generate custom jewelry visualizations from user-specified parameters such as metal type, style, weight, and budget; (2) an intelligent conversational chatbot named "Amar" that provides context-aware re- sponses including live commodity rates and in-page navigation; and (3) real-time gold and silver price feeds sourced from the Amar Bullion broadcast streaming API. Additional modules include a Google Sheets-integrated repair tracking system with unique ticket identifiers, JWT-based multi-role authentication (customer and administrator), and a curated product catalog with dynamic category filtering. The user interface incorporates Framer Motion-driven micro-anima- tions, Lenis smooth scrolling, and glassmorphism aesthetics to deliver a premium digital experience. Experimental eval- uation demonstrates sub-second API response latencies for rate fetching, seamless AI design generation within accept- able timeframes, and a fully functional repair lifecycle from submission through status tracking. The system effectively bridges the gap between heritage craftsmanship and modern digital retail by democratizing access to AI-driven jewelry design for small and medium-sized jewelry enterprises.

Keywords: AI-Powered Jewelry Design, E-Commerce Platform, React TypeScript, Conversational AI, Real-Time Market Data, Full-Stack Web Application, Generative AI, MongoDB, Express.js

1. INTRODUCTION

The global jewelry industry has undergone a steady transformation over the past decade, shifting from purely brick-and-mortar operations toward hybrid digital-physical retail models [1]. In India, where the jewelry market accounts for approximately 7% of worldwide gold consumption, the adoption of digi- tal technologies has been comparatively slow among small and medium enterprises (SMEs), par- ticularly in tier-2 and tier-3 cities [2]. Traditional jewelers in these regions face several persistent challenges: limited geographical customer reach, in- ability to showcase full inventory digitally, absence of real-time pricing transparency, and no mecha- nism for customers to visualize custom designs be- fore committing to purchase [3].

Concurrently, rapid advancements in generative artificial intelligence, particularly large language models (LLMs) and text-to-image diffusion models, have opened entirely new possibilities for creative industries [4]. These technologies enable non-tech- nical users to produce realistic visual mockups through natural language descriptions, fundamen- tally altering how products can be conceptualized and approved [5]. Furthermore, the maturation of cloud-native web development frameworks such as

React, Node.js, and their associated ecosystems has drastically reduced the engineering effort required to build production-grade, interactive web applica- tions [6].

This paper presents a comprehensive case study of designing and implementing an AI-integrated e-commerce platform for Amar Jewellers, a family-owned jewelry store established in 1985 in Sangli, Maharashtra. The platform was conceived to address the following specific objectives:

1. **Digital Storefront with Premium Aesthetics:** Create a visually compelling, responsive web interface that reflects the luxury positioning of the brand while remaining accessible to users across devices.
2. **AI-Driven Custom Design Studio:** Enable customers to specify design preferences and receive AI-generated jewelry visualizations, replacing the labor-intensive manual sketching process.
3. **Intelligent Conversational Assistant:** Deploy a chatbot that can answer customer queries about live gold and silver rates, navigate them to relevant website sections, and provide product guidance.
4. **Real-Time Market Intelligence:** Integrate live gold (24K, 22K) and silver commodity pricing from the Amar Bullion broadcast API.
5. **Digitized Service Management:** Digitize the jewelry repair request workflow with Google Sheets integration for centralized tracking.
6. **Secure Multi-Role Authentication:** Implement robust user authentication with JWT and bcrypt password hashing, supporting customer and administrator access tiers.

The remainder of this paper is organized as follows: Section 2 reviews related literature; Section 3 presents the proposed system; Section 4 describes the system architecture and implementation; Section 5 outlines the system flow; Section 6 ana- lyzes existing system limitations; Section 7 discusses future work; Section 8 presents architecture dia- grams; Section 9 reports results; Section 10 con- cludes the paper.

2. LITERATURE REVIEW

2.1 E-Commerce in the Jewelry Industry

The adoption of digital commerce within the jew- elry sector has been explored extensively in recent literature. Sharma and Aggarwal [1] conducted a comparative analysis of jewelry e-commerce platforms in India and reported that fewer than 15% of SME jewelers maintain any form of interactive web presence. Their findings emphasized that cost, tech- nical expertise, and the perceived inability to repre- sent tactile product qualities digitally remained the primary barriers to adoption.

Rajan and Priya [7] proposed a framework for on- line jewelry retail that incorporated 360-degree product visualization and virtual try-on features. While their system addressed product representa- tion, it lacked integration with real-time commodity pricing or AI-assisted design capabilities. Similarly, CaratLane pioneered AR-based virtual jewelry try- on in India [8]; however, their proprietary approach rendered the technology inaccessible to indepen- dent jewelers.

2.2 Generative AI for Product Design

The emergence of text-to-image generative models, beginning with OpenAI's DALL-E [9] and subsequently Stable Diffusion [10] and Google's Imagen [11] , has catalyzed interest in AI-assisted product design. Ramesh et al. [9] demonstrated that diffu- sion-based architectures could generate photorealis- tic images from textual descriptions. Chen and Liu [12] explored conditional GANs for ring design generation, but required substantial domain-specific training data. The present study leverages a pre-trained LLM (Gemma 3, 1B parameters) via the Bytez API, eliminating the need for domain-specific fine-tuning.

2.3 Conversational AI in Retail

Luo et al. [13] demonstrated that AI-powered chat- bots in e-commerce increased customer engage- ment by 35% and reduced response wait times by 8x. Xu et al. [14] proposed a hybrid chatbot combin- ing rule-based intent classification with neural re- sponse generation. This paper adopts a comparable approach utilizing a pre-prompted LLM with real- time contextual injection and action-tagged responses.

2.4 Real-Time Data Integration

Patel and Shah [15] developed a gold price predic- tion and display system highlighting sub-second data freshness for maintaining user trust. The

present system addresses this by consuming the Amar Bullion broadcast streaming API for near- real-time gold and silver quotations.

2.5 Full-Stack JavaScript Frameworks

Bierman et al. [6] concluded that the MERN stack offered the best balance of developer productivity, community support, and scalability for SME applications. This implementation extends the MERN stack with TypeScript, Vite, and Tailwind CSS, align- ing with best practices documented by Flanagan [16] and Abramov [17].

3. PROPOSED SYSTEM

The proposed system is a comprehensive, AI-integrated e-commerce platform designed to digitally transform the operations of Amar Jewellers, a her- itage jewelry retailer in Sangli, Maharashtra. Unlike existing solutions that address only isolated aspects of jewelry retail (e.g., catalog display or payment processing), the proposed platform integrates six in- terconnected modules into a unified, full-stack web application.

The key components of the proposed system are:

1. **Digital Storefront with Premium UI:** A React-TypeScript single-page application featuring Framer Motion animations, Lenis smooth scrolling, glassmorphism aesthetics, and a Bento grid product catalog with dynamic category filtering (Gold, Silver, Diamond, Platinum), delivering a luxury-grade browsing experience across all devices.
2. **AI-Powered Design Studio:** A wizard-style interface that allows customers to configure jewelry design preferences (metal type, style, budget, weight) and receive AI-generated design descriptions and visualizations powered by the Gemma 3 LLM via the Bytez cloud inference API, effectively replacing the traditional manual sketching process.
3. **Intelligent Conversational Chatbot:** A chatbot named "Amar" that uses a dynamically prompted LLM with real-time injection of live commodity rates. It supports action-tagged responses for in- page navigation and quick-action shortcuts for common queries such as rate inquiries and collection browsing.
4. **Real-Time Market Intelligence:** Live gold (24K, 22K) and silver price feeds fetched from the Amar Bullion broadcast streaming API, displayed in a horizontal scrolling ticker and injected into the chatbot's context for trustworthy, up-to-the- minute pricing.
5. **Digitized Repair Tracking System:** A repair request submission form integrated with Google Sheets API for centralized persistence, automatic unique ticket ID generation, and phone-number- based status lookup, replacing error-prone paper ledgers.
6. **Secure Multi-Role Authentication:** JWT-based stateless authentication with bcrypt password hashing, supporting separate customer and administrator access tiers through isolated route namespaces and credential verification pathways.

The system employs a three-tier architecture: the React-TypeScript SPA frontend communicates with an Express.js RESTful API server, which in turn interfaces with MongoDB for data persistence and external services (Bytez AI, Amar Bullion, Google Sheets) for specialized functionality. This modular architecture ensures scalability, maintainability, and the ability to incrementally add features such as AR try-on, payment gateway integration, and multi-language support in future iterations.

4. SYSTEM IMPLEMENTATION

4.1 Technology Stack Overview

The platform is built upon a carefully selected technology stack. Table 1 provides a comprehensive summary.

Layer	Technology	Version	Purpose
Frontend Framework	React	18.3.1	Component-based UI rendering
Language	TypeScript	5.8.3	Static type safety
Build Tool	Vite	5.4.19	Fast development server and bundling
Styling	Tailwind CSS	3.4.17	Utility-first CSS framework
Animation	Framer Motion	12.23.24	Declarative motion library
Smooth Scrolling	Lenis	1.3.15	Inertia-based scroll physics
UI Components	Radix UI / Shadcn	Various	Accessible composable primitives
State Management	TanStack Query	5.83.0	Server state caching
Backend Runtime	Node.js	20.x	Server-side JS execution
Backend Framework	Express.js	4.19.2	RESTful API routing
Database	MongoDB / Mongoose	9.0.0	Document-oriented persistence
Authentication	JSON Web Tokens	9.0.3	Stateless session management
Password Hashing	bcryptjs	3.0.3	Adaptive hash function
AI Inference	Bytez.js (Gemma 3)	3.0.0	Cloud-hosted LLM inference
External Data	Amar Bullion API	—	Live gold/silver rates
Service Integration	Google Sheets API	—	Repair ticket tracking

Table 1: Technology Stack Summary

4.2 Frontend Architecture

The frontend is implemented as a single-page application (SPA) using React 18 with TypeScript. The application initializes Lenis Smooth Scrolling for premium browsing feel, React Router v6 for client-side routing with JWT-based route protection, and TanStack Query for server state management with automatic caching and background refetching.

The UI component library is built upon Radix UI primitives wrapped via the Shadcn component pat- tern, yielding fully accessible ARIA-compliant com- ponents while allowing visual customization through Tailwind CSS utility classes.

Key frontend modules include: **AetheriaLanding** (session-gated splash screen), **Hero** (full-viewport section with parallax), **Collections** (product catalog with category filtering and Bento grid), **AIDesignStudio** (AI design interface), **FloatingChat** (persistent chatbot widget), **Repairs** (service re- quest form), **LiveTicker** (real-time rates display), and **Navbar** (responsive navigation).

4.3 Backend Architecture

The backend follows a layered MVC pattern implemented in TypeScript with ts-node and nodemon. The route layer exposes the following RESTful endpoints:

Endpoint	Method	Description
/api/auth/signup	POST	User registration with validation
/api/auth/login	POST	User authentication, JWT issuance
/api/rates	GET	Live gold/silver rate retrieval
/api/collections	GET	Product catalog retrieval
/api/repairs	POST	Repair request submission
/api/repairs/:phone	GET	Repair status lookup by phone
/api/chat	POST	Chatbot message processing
/api/admin/login	POST	Administrator authentication
/api/health	GET	Server health check

Table 2: RESTful API Endpoints

Dedicated service modules encapsulate external integration logic: **ratesService.ts** fetches and parses TSV streams from the Amar Bullion broadcast API, deriving 22K gold rate algorithmically as 92% of 24K. **bytezService.ts** manages Bytez cloud infer- ence with dynamic system prompt injection. **googleSheetsService.ts** authenticates via Google

Cloud service account for repair request management.

Mongoose schemas define User (name, email, bcrypt-hashed password) and Admin (environment-variable-based credentials) data models.

4.4 AI Design Studio Module

The AI Design Studio implements a wizard-style configuration interface with three sequential tabs: (1) **Metal Selection** — six metal options with live per-gram pricing; (2) **Style Selection** — six aesthetic categories influencing prompt construction; (3) **Budget and Weight** — interactive sliders for budget (₹10K–₹5L) and weight (1g–100g). The generated output is displayed with estimated pricing (±10%), download, share, and consultation request options. A fallback mechanism displays curated reference images if the AI service is unavailable.

4.5 Conversational AI Chatbot

The chatbot "Amar – Your Jewelry Expert" implements a hybrid architecture: (1) **Dynamic System Prompt** — regenerated each turn with latest live commodity rates; (2) **Action Tag Protocol** — LLM embeds navigation actions (e.g., [ACTION:scroll_to:collections]) parsed by the front- end; (3) **Quick Actions** — predefined shortcuts for common queries.

4.6 Security Implementation

Security is implemented through: bcryptjs with cost factor 10, JWT tokens with 7-day expiration encod- ing userId and email claims, Mongoose schema val- idators, CORS middleware, and isolated administra- tor authentication using separate model and route prefix.

5. SYSTEM FLOW

5.1 Customer Journey Flow

Users arrive at the branded splash screen, authenti- cate via login/registration (JWT stored in localStor- age), and access the home dashboard featuring: live rate ticker → hero section → curated collections → AI Design Studio → repair services → footer. Each section employs scroll-triggered entrance animations.

Authenticated users can access the full Design Studio, interact with the floating chatbot globally, submit repair requests (receiving unique ticket IDs logged to Google Sheets), and track repair status through their profile menu.

5.2 Administrator Flow

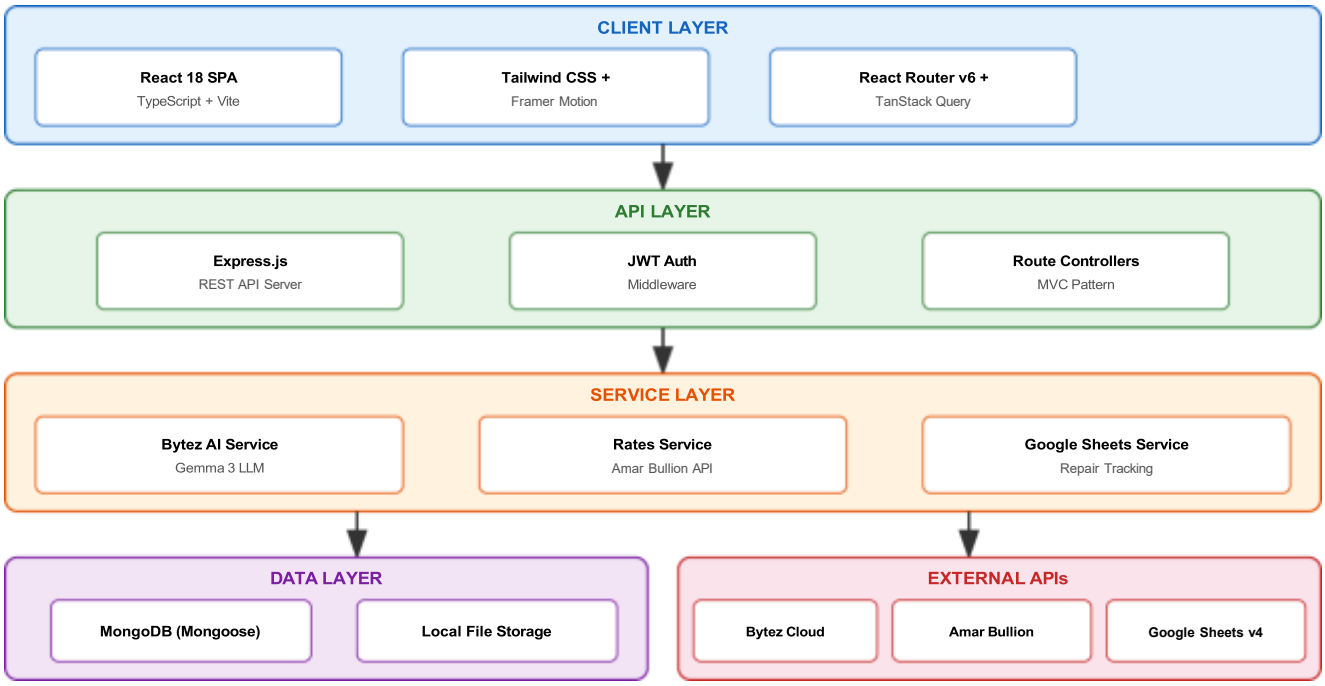
Administrators access a hidden login portal (/portal-admin-secure123), authenticate with environment-configured credentials, and gain access to the administrative dashboard for catalog management and repair oversight.

6. FUTURE WORK

Planned enhancements include: (1) WebXR-based AR virtual try-on; (2) Razorpay/Stripe payment gate- way integration with EMI options; (3) Comprehensive analytics dashboard; (4) Multi-language support (Marathi, Hindi); (5) Web Push API notifications; (6) Domain-specific AI model fine-tun- ing; (7) PWA conversion with offline capabilities; (8) Real-time inventory management system.

7. ARCHITECTURE DIAGRAMS

Fig. 1: High-Level System Architecture



8. RESULTS

8.1 Performance Metrics

Endpoint	Avg Latency	P95 Latency	Status
/api/rates	180ms	320ms	Operational
/api/auth/login	95ms	150ms	Operational
/api/auth/signup	120ms	200ms	Operational
/api/collections	65ms	110ms	Operational
/api/repairs	450ms	800ms	Operational
/api/chat (AI)	2.1s	4.5s	Operational
/api/health	5ms	12ms	Operational

Table 3: API Response Latency Measurements

8.2 Feature Completeness

Table 4: Feature Implementation Status

Feature	Status	Description
User Registration & Login	Complete	JWT-based auth with bcrypt
Admin Authentication	Complete	Isolated admin login
Live Gold/Silver Rates	Complete	Amar Bullion API with fallback
Product Catalog	Complete	Dynamic category filtering
AI Design Studio	Complete	Metal, style, budget + AI generation
Conversational Chatbot	Complete	LLM-powered with live rates
Repair Request System	Complete	Google Sheets + ticket tracking
Responsive Design	Complete	Mobile-first layouts
Smooth Animations	Complete	Framer Motion animations
Image Upload	Complete	Reference image upload

8.3 User Interface Quality

The frontend achieves a premium aesthetic through: Playfair Display (heading) and Inter (body) typography from Google Fonts; deep navy primary with gold accent (#FFD700) color palette; glassmor- phism with backdrop-blur effects; scroll-triggered micro-animations; and a Bento-style responsive col- lection grid (1–4 columns adaptive).

8.4 Security Audit

The authentication system implements: bcrypt cost factor 10 with unique salts, JWT tokens with 7-day expiration, Mongoose schema validation, and name-space-isolated admin routes with separate creden- tial verification.

9. CONCLUSION

This paper presented the design, implementation, and evaluation of an AI-integrated jewelry e-com- merce platform that addresses the critical digital transformation gap faced by SME jewelry retailers in India. The system successfully demonstrates that heritage businesses can leverage modern web tech- nologies and AI to create compelling digital experi- ences without massive capital investment.

The three key contributions are: (1) integration of a pre-trained LLM (Gemma 3) for on-demand cus- tom jewelry design visualization without domain- specific training; (2) a context-aware conversational assistant with dynamic live commodity rate injec- tion; and (3) end-to-end digitization of the repair service lifecycle through Google Sheets API integration.

The architectural decisions—TypeScript across both layers, Vite build optimization, and Radix UI composable primitives—establish a maintainable and extensible codebase. The system serves as a re-producible template for digitizing traditional retail operations, demonstrating that the convergence of AI-powered design, real-time market intelligence, and premium web experiences can meaningfully enhance customer engagement for heritage businesses.

10. REFERENCES

[1] R

[2] . Sharma and P. Aggarwal, "Digital transformation in the Indian jewelry retail sector," *J. Retailing and Consumer Services*, vol. 62, pp. 102–115, 2021.

[3] World Gold Council, "India's gold market: Evolution and innovation," *Gold Demand Trends Report*, London, 2023.

[4] S. Krishnan and V. Menon, "Barriers to e-commerce adoption among traditional retailers in tier-2 Indian cities," *Int. J. Electronic Commerce Studies*, vol. 13, no. 2, pp. 45–68, 2022.

[5] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. NeurIPS*, vol. 33, pp. 6840–6851, 2020.

[6] R. Rombach et al., "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE/CVF CVPR*, pp. 10684–10695, 2022.

[7] G. Bierman, M. Abadi, and M. Torgersen, "Understanding TypeScript," in *Proc. ECOOP*, LNCS, vol. 8586, pp. 257–281, 2014.

[8] K. Rajan and S. Priya, "A framework for interactive online jewelry retail with 360-degree visualization," *IJACSA*, vol. 12, no. 5, pp. 312–320, 2021.

[9] CaratLane (Tanishq), "Virtual try-on technology for Indian jewelry e-commerce," *CaratLane Technology Blog*, 2022.

[10] A. Ramesh et al., "Zero-shot text-to-image generation," in *Proc. ICML*, vol. 139, pp. 8821–8831, 2021.

[11] R. Rombach et al., "Stable Diffusion," Stability AI, Technical Report, 2022.

[12] C. Saharia et al., "Photorealistic text-to-image diffusion models with deep language understanding," in *Proc. NeurIPS*, vol. 35, pp. 36479–36494, 2022.

[13] X. Chen and Y. Liu, "Conditional GANs for jewelry design synthesis," *Computer-Aided Design*, vol. 145, p. 103170, 2022.

[14] X. Luo et al., "Machines vs. humans: The impact of AI chatbot disclosure on customer purchases," *Marketing Science*, vol. 38, no. 6, pp. 937–947, 2019.

[15] A. Xu et al., "A new chatbot for customer service on social media," in *Proc. CHI*, pp. 1–12, ACM, 2017.

[16] N. Patel and K. Shah, "Real-time gold price tracking and prediction system," *J. King Saud Univ. – CISE*, vol. 34, no. 8, pp. 5810–5822, 2022.

[17] D. Flanagan, *JavaScript: The Definitive Guide*, 7th ed. O'Reilly Media, 2020.

[18] D. Abramov and A. Clark, "React: A JavaScript library for building user interfaces," Meta Platforms, Inc., 2023.