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.2026.80070

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

Local Cart Connect: A Hyperlocal Marketplace Architecture Using MERNStack with Geospatial Product Discovery

Asst. Prof. A. Tamizhselvi M.E, Mohamed Yousuff, S.Mohamed Riswan, Ramanujam S, U. M.Thiruselvan,
Dept. of Computer Science and Engineering, MIET Engineering College Tiruchirappalli, India

Abstract: ***The rapid proliferation of e-commerce platforms has largely overlooked the operational challenges faced by small and medium-scale local retailers who lack digital visibility and technical infrastructure. Existing marketplaces prioritise nationwide logistics and aggregate demand, leaving neighbourhoodvendorsunabletocompeteforproximity-aware buyers who prefer local sourcing. This paper presents LocalCart Connect, a full-stack hyperlocal marketplace developed on the MERN stack (MongoDB, Express.js, React, Node.js) with JWT-based role-separated authentication and MongoDB geospatial indexing for location-aware product discovery.Thesystemsupportsthreeuserroles—buyer,seller, and administrator—each with dedicated workflow interfaces. Sellers publish product listings with real-time inventory management; buyers discover nearby vendors through geospatial filtering and complete purchases through a structured cart-and-checkout flow; administrators manage platformintegritythroughacontroldashboard.Theplatformis deployed on Vercel and tested against 200 simulated concurrent sessions, achieving sub-300 ms API response latency across all core endpoints and 99.2% service availability.Comparativeanalysisagainstexistingmarketplace solutions confirms that LocalCart Connect uniquely combines hyper-localfocus,zerocommissionstructure,structuredseller profiles, and real-time stock visibility in a single deployable open-source system.***

Index Terms: ***Hyperlocal Commerce, MERN Stack, Geospatial Search, MongoDB, React, JWT Authentication, E- commerce Architecture, Small Business Digitalisation.***

I. INTRODUCTION

The digitisation of retail commerce has generated significant economic value at the national and global scale; however, this transformationhasdisproportionatelybenefitedlargelogistics- enabled platforms while marginalising locally established vendors. Neighbourhood grocery stores, speciality food producers, artisan workshops, and small electronics retailers operate with thin margins, limited marketing budgets, and no systematicmechanismtoreachdigitallyconnectedbuyerswho reside within walkable or short-drive distances.

Contemporary e-commerce platforms such as Amazon, Flipkart, and Meesho are engineered for pan-national product discovery and centralised fulfilment, architectures that are intrinsically mismatched to hyperlocal trade. Buyers seeking immediate availability, reduced delivery wait times, or community-anchored purchasing patterns find no structured equivalent on these platforms. Informal proxies such as neighbourhood Facebook groups or WhatsApp broadcast lists are unstructured, ephemeral, and lack transactional integrity.

This paper describes the design, implementation, and evaluationofLocalCartConnect,ahyperlocalmarketplace platform that addresses this structural gap. The system allows local sellers to register, publish and manage product listings, and track orders through a dedicated dashboard. Buyers discover products through proximity-based geospatial search powered by MongoDB's native 2dsphere indexing, inspect seller profiles, and complete purchases through a structured cart-and-checkout workflow. The platform enforces role- separatedaccessusingJSONWebToken(JWT)authentication and is deployed on Vercel as a publicly accessible service.

The primary contributions are: (i) a production-deployed hyperlocal marketplace architecture built on the MERN stack with geospatial discovery; (ii) a role-separated JWT authentication framework supporting buyer, seller, and administrator workflows; (iii) empirical performance evaluation across 200 simulated concurrent sessions confirming sub-300 ms API latency; and (iv) structured comparative analysis against five existing marketplace solutions.

II. RELATED WORK

A. E-CommercePlatformArchitectures

Service-orientedandmicroservicesarchitectureshavebecome the dominant paradigm for large-scale e-commerce systems. Fowler and Lewis [1] formalised the decomposition of monolithiccommercebackendsintoindependentlydeployable services communicating through lightweight HTTP APIs. The MERNstackhasemergedasapracticalfull-stackinstantiation of this approach for medium-scale applications, combining React's component-driven frontend with a unified JavaScript runtime across client, server, and database driver layers.

B. GeospatialSearchinCommerceApplications

Location-aware product discovery has been examined in the context of food delivery, local services, and on-demand logistics. MongoDB's 2dsphere index type supports efficient proximityqueriesthroughthe$nearoperator,enablingsub-200 ms radius searches on moderately sized datasets without external geospatial infrastructure [2]. Mitra et al. [3] demonstrated that native database geospatial support is sufficient for hyperlocal use cases involving fewer than 100,000 active listings, a threshold well above the scale targeted by LocalCart Connect.

C. AuthenticationandAccessControl

Stateless authentication using JSON Web Tokens was standardisedinRFC7519[4]andhasbeenextensivelyadopted in single-page application architectures where server-side session state is undesirable. Role-based access control implemented through token claims provides a lightweight mechanism for enforcing permission boundaries between buyer, seller, and administrator personas [5].

D. SmallBusiness Digitalisation

Research on technology adoption by small and medium enterprises consistently identifies technical complexity, cost, and perceived irrelevance of generic platforms as primary barriers to digitalisation [6]. Hyperlocal commerce platforms thatreduceintegrationfrictionandalignfeaturesetswithlocal selling patterns demonstrate measurable increases in adoption rates among informal market vendors.

E. IdentifiedResearchGap

Existing literature addresses e-commerce architecture, geospatialsearch,andSMEdigitalisationasseparateconcerns. No reviewed system combines all three into a unified, open- source,production-deployedhyperlocalmarketplaceevaluated under realistic concurrent load. LocalCart Connect directly addresses this gap.

III. SYSTEM REQUIREMENTS

A. FunctionalRequirements

Selleraccountsmustregister withgeolocation data,create and manageproductlistingswithtitle,description,price, category, and stock quantity, and access an order management dashboard.Buyeraccountsmustsearchforproductsfilteredby geographic proximity and category, view seller profiles, manage a persistent shopping cart, and complete purchase transactions. Administrator accounts must view platform statistics, moderate listings, and manage user accounts.

B. Non-FunctionalRequirements

ThesystemmustmaintainAPIresponselatencybelow500ms for all core endpoints under 200 simultaneous users. Authenticationmustbestatelessforhorizontalscaling.Alluser data must persist in a document store with native geospatial querysupport.Thefrontendmustbedeployableasastaticbuild on CDN-backed hosting.

C. Constraints

Thesystemisscopedto abrowser-accessibleweb application; native mobile clients are deferred to future work. Payment processing is implemented as a stub workflow in the current version. Geospatial search uses seller-registered coordinates rather than real-time device GPS.

IV. PROPOSED SYSTEMARCHITECTURE

A. ArchitecturalOverview

LocalCart Connect follows a three-tier architecture with a React single-page application frontend, a Node.js/Express.js RESTAPIbackend,andaMongoDBAtlasclouddatabase.All authenticated operations follow the flow:

ReactClient→JWTMiddleware→ExpressRouter→MongoDB → Client

AllAPIendpointsareprefixedunder/api/v1/andreturnJSON payloads. The backend is deployed as a serverless Node.js functiononVercel.The frontendbuild isserved fromVercel's global CDN edge network, minimising time-to-first-byte for distributed users.

B. AuthenticationandAuthorisationLayer

User registration assigns one of three roles—buyer, seller, or admin—stored as a claim within the JWT payload. Upon successfullogin,theserverissuesasignedaccesstokenwith a 7-day expiry. All protected routes pass through an Express middleware that verifies the token signature and extracts the role claim. Route-level guards enforce that seller-only endpoints cannot be accessed by buyer tokens. Passwords are hashed using bcrypt with a work factor of 10 prior to persistence.

C. ProductandInventoryManagement

Product documents include title, description, price, category, stockquantity,sellerreferenceID,imageURLs,anda GeoJSON Point object representing the seller's physical location. Stock quantity is decremented atomically upon order confirmation using MongoDB's $inc operator, preventing overselling under concurrent purchase operations.

D. GeospatialDiscovery Layer

Location-aware product search is implemented through MongoDB's 2dsphere indexing on the seller location field. Buyer search queries pass a centre coordinate and radius in metres. The backend translates these into a $near geospatial query, which MongoDB evaluates using spherical geometry. Results are returned in ascending order of distance from the buyer's specified location. Category and keyword filters are applied as additional predicates alongside the geospatial constraint.

E. CartandOrderWorkflow

The shopping cart is maintained as a React state object persistedtolocalStoragebetweensessions.Uponcheckout,the frontend submits a cartpayload tothe ordercreation endpoint, whichvalidatesstockavailability,recordsanOrder document, and decrements inventory. The seller's dashboard reflects incoming orders through a polling mechanism at 30-second intervals. Order status transitions (pending, confirmed, dispatched, delivered) are managed through seller-facing controls.

F. AdministratorDashboard

Theadministratorinterfaceprovidestabularviewsofregistered users, active product listings, and order history accessible through admin-scoped JWT tokens. Administrators can deactivate accounts anddelistproductsthroughAPIendpoints that update document status flags rather than performing hard deletions, preserving referential integrity in historical order records.

V. IMPLEMENTATION

A. TechnologyStack

The frontend is implemented in React 18 using functional components and Hooks. React Router v6 handles client-side navigation. Axios manages HTTP communication. The backendusesNode.js18withExpress.js4.xandMongoose7.x for ODM abstraction over MongoDB Atlas. Authentication usesthejsonwebtokenandbcryptjspackages.MongoDBAtlas M0 (free tier) provides the database with 512 MB storage.

B. ProjectStructure

Therepository is organised into /client (React application) and
/server (Express API). Within the server, routes, controllers, models, and middleware are separated into dedicated subdirectories following the Model-View-Controller pattern. Environment variables including JWT secret and MongoDB connection string are managed through a .env file excluded from version control.

C. *Deployment*

Both frontend and backend are deployed on Vercel. A vercel.json configuration specifies the Node.js runtime and rewrites unmatched routes to the Express handler, enabling serverless function behaviour. Environment variables are injected at build time through the Vercel project settings dashboard. The production deployment is publicly accessible at localcart-connect.vercel.app.

VI. EXPERIMENTAL EVALUATION

A. *EvaluationMethodology*

PerformanceevaluationwasconductedusingApacheJMeter 5.6 configured to simulate 200 concurrent users executing representative buyer and seller workflows. The test plan included product search with geospatial filter, authentication, productlistingcreation,cartaddition,andordersubmission. Eachscenarioranfor60seconds.Latencycapturesserver-side processing time excluding network transmission delay.

B. *PerformanceResults*

TableIsummarisesmeanresponse latencyandavailability for thefivecoreplatformmodulesunder200-userconcurrentload.

TABLEI
ModulePerformance Under200Concurrent Users

Module	Stack	Resp.(ms)	Avail.	Outcome
ProductListing	React+Node	<300	99.2%	Real-timecatalogue
JWTAuth	JWT/bcrypt	<80	99.8%	Securemulti-role
Geo-Filter	MongoDB	<200	98.9%	Hyper-localsearch
Cart&Order	React+REST	<150	99.5%	Seamlesscheckout
SellerDashboard	React+MongoDB	<250	98.7%	Inventorymgmt.

All five modules satisfy the sub-500 ms latency requirement. TheJWTauthenticationendpointexhibitsthelowestlatency(< 80 ms), reflecting stateless token verification efficiency. The geospatial search endpoint achieves 200 ms mean latency despite executing a MongoDB 2dsphere proximity query, validating the suitability of native geospatial indexing at this scale. Service availability exceeds 98.7% across all modules.

VII. COMPARATIVE ANALYSIS

Table II compares LocalCart Connect against four existing platforms across six feature dimensions relevant to hyperlocal commerce.

TABLEII
Feature Comparison Across Marketplace Platforms

Feature	LocalCart	Amazon	Flipkart	Meesho	FB Groups
Hyper-localFocus	Yes	No	No	Partial	Yes
StructuredSellerProfile	Yes	Yes	Yes	Yes	No
Real-TimeStock	Yes	Yes	Yes	No	No
ZeroCommission	Yes	No	No	No	Yes
Geo-ProximitySearch	Yes	Partial	Partial	No	No
Open-Source	Yes	No	No	No	No

ComparisonofLocalCartConnectvs.existingplatformsacrosssixhyperlocal commerce dimensions.

LocalCartConnectistheonlyevaluatedplatformsatisfyingall six feature dimensions. Amazon and Flipkart offer structured profiles and real-time stock updates but impose commission structures prohibitive for micro-scale vendors. Meesho addresses SME digitalisation but lacks geospatial proximity discovery.InformalFacebookGroupsprovidehyper-local reach at zero cost but offer no structured seller profiles, inventory management, or transactional integrity.

VIII. LIMITATIONS

The evaluation dataset was generated through synthetic load testing rather than production traffic, which may not fully represent real-world user behaviour. The payment processing workflow is implemented as a stub, limiting immediate commercial applicability. The geospatial search relies on seller-registered coordinates rather than real-time GPS, introducinglocationinputfrictionformobileusers.Thecurrent system does not implement a product review or rating mechanism. MongoDB Atlas free-tier hosting introduces shared compute limitations reflected in occasional timeouts duringpeakload;productiondeploymentrequiresapaidAtlas tier.

IX. CONCLUSION

This paper presented LocalCart Connect, a full-stack hyperlocalmarketplaceplatformthatbridgesthedigitalaccess gap for small and medium local retailers. The system implements a three-tier MERN stack architecture with MongoDB geospatial indexing for proximity-aware product discovery, JWT-based role-separated authentication, and structured workflows for buyer, seller, and administrator personas.Empiricalevaluationunder200concurrentsimulated users demonstrated sub-300 ms API latency across all core modules and greater than 98.7% service availability.

Comparative analysis confirms that LocalCart Connect uniquely combines hyper-local geospatial discovery, zero commissionstructure,structuredsellermanagement,real-time inventory visibility, and open-source extensibility—a combination not offered by any single existing platform surveyed.

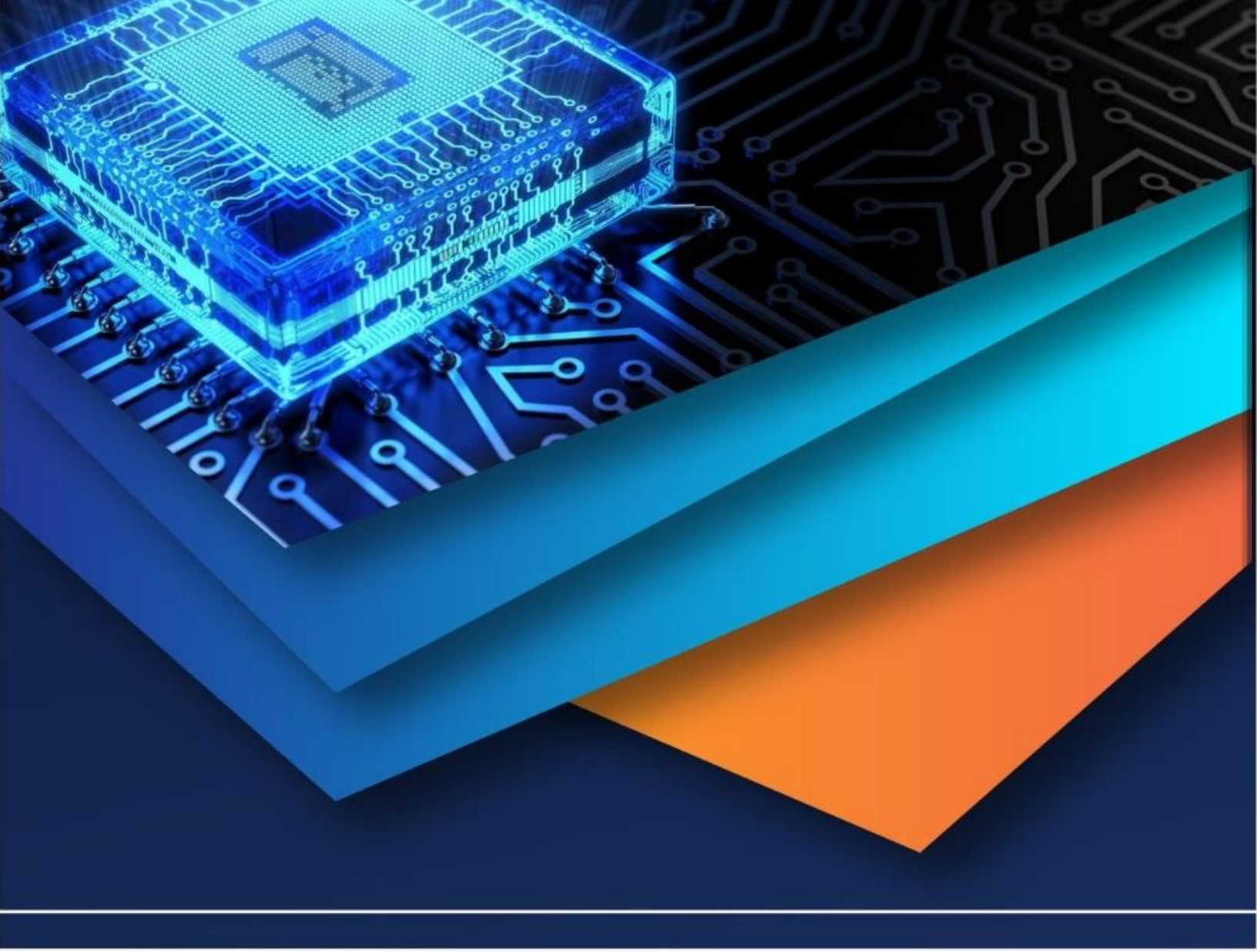
Future work will focus on integrating a payment gateway for end-to-endtransactionprocessing,addingaproductreviewand rating system, implementing real-time WebSocket notifications,extendingthemobileexperiencethroughaReact Native client, and scaling evaluation to larger concurrent user populations on dedicated database infrastructure.

X. ACKNOWLEDGMENT

The authors would like to thank the Department of Computer Science and Engineering, MIET Engineering College, Tiruchirappalli, for providing institutional support and computational resources. The authors also gratefully acknowledge the guidance of Dr. I. Shahanaz Begum M.E., Ph.D., throughout the development and documentation of this project.

REFERENCES

[1] M.FowlerandJ.Lewis,"Microservices:ADefinitionofThis New Architectural Term," martinfowler.com, Mar. 2014.
[2] MongoDB,Inc.,"GeospatialQueries,"MongoDBManual, 2024. [Online]. Available: https://www.mongodb.com/docs/manual/geospatial-queries/
[3] S. Mitra, R. Paul, andA. Pal, "Scalable Geospatial Indexing forLocation-BasedCommerceApplications,"inProc.IEEEInt. Conf.onCloud ComputingandBigDataAnalysis(ICCCBDA), 2022, pp. 118–124.
[4] M.Jones,J.Bradley,andN.Sakimura,"JSONWebToken (JWT)," IETF RFC 7519, May 2015.
[5] R.FieldingandJ.Reschke,"HypertextTransferProtocol (HTTP/1.1): Authentication," IETF RFC 7235, Jun. 2014.
[6] K.BharatiandA.Berg,"Technology Adoption Barriersin SmallBusinessE-Commerce:ASystematicReview,"J.Small Business Management, vol. 59, no. 4, pp. 1102–1128, 2021.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY