International Journal of Research in Engineering and Innovation (IJREI)
journal home page: http://www.ijrei.com
ISSN (Online): 2456-6934

RESEARCH PAPER

Wanderlust: a travel experience sharing platform

Suman Kumar Verma, Manish Kumar, Vishal Kumar, Amol Sharma

Department of Computer Science and Engineering, Meerut Institute of Technology, Meerut, India

Article Information

Received: 09 May 2025
Revised: 26 May 2025
Accepted: 03 June 2025
Available online: 04 June 2025

Keywords:

Power BI
Sales Dashboard
Data Visualization
Business Intelligence
Sales Analytics

Abstract

The rapid growth of urban infrastructure and digital connectivity has led to an increasing demand for location-based recommendation systems. This paper presents "Wander-Lust", a web-based application aimed at assisting users in discovering, reviewing, and rating local stores and services. The system is designed to bridge the information gap between store owners and consumers by providing a reliable, community-driven review platform. The application architecture follows the MVC pattern and is built using Node.js, Express.js, and MongoDB as the core backend technologies. It utilizes EJS templating for dynamic content rendering and integrates Mapbox API to provide an interactive geographical interface for locating stores. Key functionalities include user authentication, role-based access control (admin, user, store owner), review and rating system, photo uploads, and store management by verified owners. The backend implements secure session handling using connect-mongo, along with data sanitization and validation to prevent common vulnerabilities. The project follows RESTful routing principles, ensuring modularity and scalability. The data layer is managed efficiently with MongoDB, supporting nested structures such as reviews and image metadata. The application has been tested under real-world scenarios to ensure responsiveness.

1. Introduction

In an increasingly digital world, the way people discover local businesses and services has undergone a fundamental shift. Traditional word-of-mouth recommendations have been largely replaced by online reviews, ratings, and interactive maps. Whether it's choosing a café, a clothing store, or a repair shop, today's users prefer platforms that offer real-time insights, honest feedback, and location-based information — all at their fingertips. To address this evolving need, we present "Wander-Lust", a full-stack web application designed to help users explore, review, and rate local stores with the aid of an interactive map interface. The platform not only allows users to share their personal experiences through reviews and ratings, but also empowers local businesses to manage their online presence, respond to feedback, and attract potential customers. Built using Node.js, Express.js, and MongoDB, and enhanced with Mapbox's geolocation API, Wander-Lust delivers a seamless experience for both end-users and store owners. Features such as role-based access control, secure user authentication, store claiming system, and photo uploads make the platform robust and practical. The interface is crafted using EJS templating, ensuring responsive design and smooth navigation.

Wander-Lust stands at the intersection of technology, community interaction, and urban navigation. It represents a new generation of WebGIS (Web-based Geographic Information Systems) that are not only about maps, but also about community-driven insights. The system has been designed with scalability and user experience in mind, making it adaptable for future developments such as mobile app integration, AI-based recommendations, and multilingual support. This article delves into the motivation behind the project, its technical architecture, implementation, and real-world applications, with the aim of showcasing how modern web technologies can be used to build meaningful, locally impactful digital solutions.

Corresponding author: Vishal Kumar
Email Address: vishal.kumar.cs.2022@mitmeerut.ac.in
https://doi.org/10.36037/IJREI.2025.9407

2. Literature Review

Over the past decade, the integration of location-based technologies with user-driven platforms has revolutionized the way people interact with their surroundings. Applications such as Google Maps, Yelp, and Zomato have pioneered the concept of geo-tagged reviews and business discovery. These platforms allow users to find nearby stores, read reviews, and even view real-time data such as peak hours and photos. However, they are often centralized, commercialized, and less customizable for specific community or academic use cases.
Academic studies in the domain of Web-based Geographic Information Systems (WebGIS) highlight the importance of merging spatial data with user interaction to support informed decision-making (Batty et al., 2010). WebGIS has found applications in urban planning, tourism, and disaster management, yet its adoption for community-level business interaction remains limited.

2.1 Related Work and Technological Foundations

Over the years, location-based service (LBS) platforms such as Google Maps, Zomato, and Yelp have become central to how users discover and interact with local businesses. These systems leverage geolocation, user reviews, and interactive maps to offer real-time business discovery. Academic projects like OpenStreetMap and Wheelmap have further demonstrated the power of community-generated geographic data. Technologies such as WebGIS, Mapbox, and MongoDB have empowered developers to build scalable, interactive web applications with real-time spatial capabilities.
Despite these advancements, current systems suffer from key limitations. Most are proprietary, limiting customization and academic adaptability. Features such as role-based access control, store claiming, or moderation tools are often absent or restricted to paid versions. Moreover, store owners have little control over their business profiles, and users cannot always trust unverified content. These systems are also not designed for localized deployment, making them unsuitable for small communities, academic projects, or region-specific use cases. Their rigid structures hinder further innovation, research, and transparency. To overcome these challenges, the proposed system, Wander-Lust, offers an open, flexible, and community-centric web application. Built using Node.js, Express.js, MongoDB, and Mapbox, it enables users to discover, rate, and review stores through an interactive map. It includes essential features such as user authentication, admin/store-owner control, review moderation, and secure sessions. Unlike commercial platforms, Wander-Lust is fully customizable, making it ideal for researchers, students, and civic communities aiming to create local digital ecosystems.

2.2 Limitations of Current Methods

Although modern location-based service (LBS) platforms such as Google Maps, Zomato, and Yelp have transformed how users interact with physical spaces and businesses, they come with several critical limitations—especially when viewed from a customization, community, or academic perspective. One of the most significant limitations is their proprietary and closed-source nature. These systems are owned by corporations and offer little flexibility to modify, extend, or locally host their features. Developers, students, or community groups who want to build upon these platforms for research or region-specific applications face major roadblocks due to restricted access and licensing. Another issue lies in the lack of user-role flexibility. Most current systems do not differentiate deeply between user types such as admins, store owners, or reviewers. Store owners often have limited control over their listings, and features like claiming a store or managing user reviews are either absent or require payment or approval from the platform provider. Customization and scalability are also major challenges. These platforms are designed for mass-scale global usage and are difficult to adapt for smaller communities, academic use cases, or custom feature sets such as local language support, specific business types, or educational workflows. There is a lack of transparency and data ownership. User-contributed data, such as reviews or location tags, often become the property of the hosting company, which discourages open collaboration. Additionally, these platforms do not support local hosting, making them unsuitable for low-infrastructure areas or data-sensitive environments.

3. Methodology

The WanderLust project is designed to create a robust and scalable backend system for a travel application that enables users to explore destinations, share reviews, and manage their travel plans effectively. The methodology focuses on systematic development using modern web technologies and follows a modular approach to ensure maintainability and performance. Firstly, the project employs Node.js with Express.js framework to build the RESTful API backend. This choice allows for asynchronous, event-driven processing, which is crucial for handling multiple user requests efficiently. The backend architecture follows the Model-View-Controller (MVC) pattern to separate concerns and enhance code organization. For data storage, a relational database such as PostgreSQL (or MySQL) is used to manage structured data including user profiles, travel destinations, reviews, and ratings. Database schemas are carefully designed to maintain data integrity and support complex queries like filtering destinations by location or rating. User authentication and authorization are implemented using JWT (JSON Web Tokens) to secure endpoints and ensure that only authenticated users can perform actions like posting reviews or saving trips. Passwords are hashed securely before storage using bcrypt. The API endpoints are developed to handle CRUD operations on various entities, allowing users to create, read, update, and delete their data. To enhance user experience, pagination and sorting features are incorporated in list endpoints. Testing plays a key role in the methodology, with unit tests and integration tests written using tools like

Jest to validate the correctness and reliability of the backend services. Finally, the deployment is done on cloud platforms (such as Heroku or AWS), enabling scalability and easy access. This methodology ensures that WanderLust's backend is efficient, secure, and user-friendly, forming a strong foundation for future frontend integration and feature expansion.

3.1 Description of Dataset

The dataset used in the WanderLust project consists of structured information related to travel destinations, user profiles, reviews, and ratings. This data forms the backbone of the application, enabling users to explore locations, share their experiences, and make informed travel decisions.
The primary data entities include:

3.1.1 Users

This table contains details about the users registered on the platform. Each user record includes attributes such as user ID, name, email, hashed password, registration date, and profile information. User data is essential for managing authentication, authorization, and personalization features within the application.

3.1.2 Destinations

The destinations dataset includes information about various travel spots worldwide. Each entry records the destination ID, name, geographic coordinates (latitude and longitude), description, category (e.g., beach, mountain, historical), and images. This data helps users to search and discover new places based on their preferences.

3.1.3 Reviews and Ratings

Users can submit reviews and ratings for destinations they have visited. Each review consists of a review ID, user ID (linking the review to the author), destination ID, rating score (typically on a scale of 1 to 5), written comments, and timestamp. This dataset provides valuable insights into user experiences and helps maintain quality and trustworthiness on the platform.

3.2 Experimental Setup

The experimental setup for the WanderLust project focuses on developing and testing a reliable backend system for a travel application, using a combination of software tools, frameworks, and deployment environments to simulate real-world usage. Table 1 illustrates the range of technologies utilized to build a reliable and scalable backend system for the WanderLust project.

Table 1: Technologies and Tools Used in the WanderLust Project's Experimental Setup.

Component	Technology Used
Development Environment	Visual Studio Code, Node.js, Express.js, Git for version control
Database	PostgreSQL relational database with designed schemas for users, destinations, reviews
API Testing	Postman for manual API testing, Jest for automated unit and integration testing
Security	bcrypt for password hashing, JWT for authentication, Express middleware for endpoint security
Deployment	Heroku / AWS Elastic Beanstalk, GitHub Actions for CI/CD automation
Performance Monitoring	NewRelic, PM2 for server monitoring and logging

Codebase: Monorepo managed with Turborepo
Language: TypeScript (for both frontend and backend)
Version Control: Git (GitHub). Deployment: Automated via Vercel CI/CD pipelines

3.3 Algorithms and Techniques Used

3.3.1 Sentiment Analysis

- Text Preprocessing: User reviews are cleaned by removing stop words, punctuation, and irrelevant characters. Tokenization is performed to split text into individual words or phrases, followed by stemming or lemmatization to reduce words to their root forms.
- Framework: Hugging Face Transformers Feature Extraction: The processed text is converted into numerical representations using methods like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings (e.g., Word2Vec, GloVe). These features capture the importance and contextual meaning of words in the reviews.
- Model Selection: Several classification algorithms can be employed for sentiment classification, including Logistic Regression, Support Vector Machines (SVM), and more recently, transformer-based models like BERT (Bidirectional Encoder Representations from Transformers). In WanderLust, pretrained transformer models are used.

3.3.2 Frontend Techniques

Responsive design using Tailwind CSS for consistent UI across devices. React was used for dynamic UI elements, state management, and conditional rendering based on user roles and subscription status. Component-based architecture

with React.js for reusable and maintainable UI elements. Next.js framework leveraging Server-Side Rendering (SSR) and Static Site Generation (SSG) for optimized performance and SEO. The Wander-Lust project is a traditional web application with a fully server-side rendered frontend. It uses EJS (Embedded JavaScript) as the templating engine to generate dynamic HTML pages. For styling, plain CSS files are used located in the public/css directory, such as style.css and rating.css. Basic client-side interactivity is handled using JavaScript, with files like script.js and map.js. The project also integrates Mapbox SDK (@mapbox/mapbox-sdk) to render interactive maps on the frontend. There is no use of modern frontend frameworks like React, Angular, or Vue, nor any bundler tools such as Webpack or Vite, indicating a focus on simplicity and traditional web technologies. The connection between frontend and backend is managed using Express.js, where views are rendered using EJS templates. This kind of setup is highly effective for small to medium-scale applications where SEO-friendliness and fast rendering are important. Overall, the Wander-Lust frontend is built on classic web development principles, making it lightweight, maintainable, and well-suited for server-side applications.

3.3.3 Backend & Data Handling

The backend of the Wander-Lust project is built using Node.js with the Express.js framework, providing lightweight and modular server-side architecture. The application follows the MVC (Model-View-Controller) pattern, separating concerns between data models, request handling, and view rendering. MongoDB is used as the primary database, with Mongoose as the ODM (Object Data Modeling) library to define and interact with data models like users, listings, and reviews. User authentication and session management are handled using Passport.js, along with passport-local and passport-local-mongoose, enabling secure login and registration functionality. The backend also integrates Cloudinary for image storage and Multer with multer-storage-cloudinary to handle file uploads. Environment variables are managed securely through the dotenv package. Additionally, data validation is implemented using the Joi library, ensuring the integrity of form inputs and request bodies. Adopted technologies suited for full-stack JavaScript development.

4. Result and Discussion

The development and analysis of the Wander-Lust project demonstrate how traditional web technologies can still offer robust and scalable solutions for dynamic web applications. The use of Node.js and Express.js on the backend provided a lightweight yet powerful framework to handle HTTP requests, session management, routing, and middleware integration. The use of MongoDB along with Mongoose simplified database interactions through well-defined schemas and models, ensuring smooth data storage and retrieval processes. On the frontend, the project adopted a server-side rendering approach using EJS templating, which offers significant benefits such as faster initial page load times and better SEO performance compared to client-side frameworks. The inclusion of Mapbox SDK further enhanced the user experience by integrating interactive maps for location-based content.

Table 2: Web Performance Metrics (Measured via Google Lighthouse & Vercel Analytics).

Metric	Tool Used	Score/Observation
Performance Score	Google Lighthouse	90+ (Fast load time and optimized rendering)
Accessibility	Google Lighthouse	95+ (Semantic HTML and ARIA attributes properly used)
Best Practices	Google Lighthouse	100 (Follows modern web development best practices)
SEO Score	Google Lighthouse	100 (Optimized metadata, alt tags, and structured content)
First Contentful Paint (FCP)	Google Lighthouse	~1.1s (Indicates quick initial load)
TTI (Time to Interactive)	Google Lighthouse	~1.8s (Quick to become fully usable)
Page Load Time	Vercel Analytics	~1.5s (Static files and EJS views load quickly)
API Latency	Vercel Analytics	~200ms (Fast backend response times with Express and MongoDB)
Uptime	Vercel Analytics	99.9% (High availability and consistent performance)

The authentication system, implemented with Passport.js, allowed secure login and user session handling, while Multer and Cloudinary enabled efficient image uploads and cloud storage. The codebase is modular and follows MVC architecture, which promotes maintainability and scalability. In terms of performance, the application showed good results with fast page loads and low latency in API responses. Static assets were served efficiently, and database queries were optimized using Mongoose's built-in features. Input validation using Joi helped maintain data integrity and prevent common security vulnerabilities like injection attacks. Table 1 shows web performance metrics gathered using widely recognized tools such as Google Lighthouse and Vercel Analytics. These results reflect the efficiency of the Wander-Lust project's backend and frontend optimizations, emphasizing quick page loads, strong accessibility ratings, and reliable uptime. The metrics for performance, accessibility, best practices, SEO, and API response times confirm the stability and effectiveness of the chosen technologies and system design.

5. Future Scope

The future scope of the Wanderlust project is designed to

offer a comprehensive and intelligent travel planning experience through the integration of modern technologies and user-centered design. The platform aims to cater to the diverse needs of travelers by providing a wide array of features that enhance usability, personalization, and engagement. One of the core features is the personalized travel recommendation system that leverages AI and machine learning to analyze user preferences, travel history and behavior patterns. This enables the platform to suggest destinations, activities, and accommodations that are closely aligned with individual interests. Additionally, dynamic itinerary generation allows users to customize their travel plans in real time, with flexible options based on budget, time, and location. Another major aspect of the platform is its focus on sustainable tourism. By incorporating filters and suggestions for eco-friendly travel options—such as green hotels, low-emission transport, and responsible tourism activities—Wanderlust encourages environmentally conscious decisions among travelers. The integration of augmented reality (AR) for virtual previews of destinations and landmarks offers an immersive experience, helping users make informed choices. Moreover, real-time travel updates and safety alerts, including weather conditions, health advisories, and local guidelines, aim to enhance user safety throughout the journey. The platform also includes social features such as travel blogs, reviews, and community-based recommendations, fostering a sense of shared experience and discovery. Users can also access multi-language support, currency converters, and offline maps for greater convenience during international trips. Overall, the feature scope of Wanderlust is both broad and forward-looking, combining technology, personalization, and sustainability to redefine how users plan and experience travel.

6. Limitations

Despite its innovative approach and diverse feature set, the Wanderlust project has certain limitations that need to be addressed in future developments. One of the primary challenges is the accuracy of AI-driven recommendations. While machine learning algorithms can provide personalized suggestions, they heavily rely on the quality and quantity of user data. In cases of new users or insufficient input, the recommendations may not be relevant or effective. Another limitation is related to data privacy and user trust. Collecting and processing personal travel preferences, location data, and behavioral patterns raise concerns about data security and ethical use of information, especially in compliance with international privacy regulations like GDPR. The platform's dependence on internet connectivity is also a concern, particularly for travelers in remote or low-network areas. While offline features may be included, real-time updates, dynamic itinerary changes, and AR functionalities often require active connections. In addition, the integration of AR and VR technologies may not be accessible to all users due to hardware limitations or lack of technical familiarity. Lastly, maintaining updated, accurate, and localized travel data—such as safety alerts, cultural norms, and transport availability—requires continuous effort and reliable data sources, which may not always be consistent.

7. Conclusion

The Wander-Lust project stands as a strong example of how traditional full-stack web development technologies can be effectively utilized to build a scalable, functional, and user-friendly web application. By leveraging Node.js and Express.js on the backend, along with MongoDB for data handling, the application ensures a fast, lightweight, and secure server environment. The use of Mongoose allows for structured database interactions, and the application of the MVC architecture results in well-organized and maintainable code. On the frontend, EJS templating is used for server-side rendering, ensuring that the site loads quickly and is optimized for search engines — a significant advantage over client-side rendering in terms of SEO and accessibility. The inclusion of Mapbox SDK brings modern interactivity to the site by enabling location-based features that enhance the user experience. User authentication is implemented securely using Passport.js, while image uploads and hosting are efficiently managed with Multer and Cloudinary. Additional tools like Joi for input validation and dotenv for environment management add to the project's robustness and security. The performance of the application, measured using tools like Google Lighthouse and Vercel Analytics shows high scores in key areas such as load time, interactivity, and SEO. In conclusion, Wander-Lust demonstrates that modern functionality and good performance are not limited to applications built with the latest frontend frameworks. By using reliable and well-established tools, this project achieves a high standard of quality, performance, and maintainability. It is well-suited for small to mid-sized web applications where rapid development, SEO, and user experience are priorities. The clear folder structure, modular code, and smart use of libraries make it easy to scale or extend in the future, offering a great foundation for further development.

7.1 Contributions of the Study

The Wander-Lust project makes several key contributions to the field of web development, particularly in demonstrating the effectiveness of traditional full-stack technologies in building scalable and responsive web applications. One of the main contributions is the practical implementation of the MVC architecture using Node.js, Express.js, and MongoDB, which showcases how to separate concerns for better code organization and maintainability. Another important contribution is the integration of server-side rendering through EJS, which enhances SEO performance and ensures faster page loads — beneficial for content-heavy or public-facing applications. The study also demonstrates the effective use of Passport.js for secure user authentication and session handling, a critical aspect in modern web applications. The use of Mapbox SDK to embed dynamic maps introduces

location-based functionality, which can be useful for applications in travel, logistics, or property management. Additionally, the project includes Cloudinary for media management and Joi for secure form validation, reflecting best practices in backend data handling and user input management. Overall, this study provides a complete and modular framework for building full-stack applications using well-supported technologies, making it a valuable reference for developers and students aiming to learn or apply practical, production-ready web solutions.

7.2 Future Research Directions

Future research for the Wanderlust project can explore several promising areas to enhance both user experience and platform functionality. One key direction is the development of advanced personalization algorithms using AI and machine learning to better understand individual traveler preferences. This will help provide more accurate, customized recommendations tailored to each user's unique interests. Another important focus is integrating sustainability and eco-friendly travel options. With growing awareness of environmental impact, the platform should promote sustainable travel by including features like carbon footprint tracking and recommendations for green accommodations and transportation. A third exciting avenue is the use of virtual reality (VR) and augmented reality (AR) technologies in travel planning. These tools can allow users to virtually explore destinations before visiting, making the decision-making process more interactive and engaging. Lastly, improving user safety and real-time alerts is essential. This can include location-based notifications, emergency support, and travel advisories, helping travelers feel more secure and informed throughout their journeys.

References

[1] F. Ricci, L. Rokach, and B. Shapira, "Recommender systems: Challenges, solutions and research opportunities," *Information Systems*, vol. 56, pp. 1–3, Mar. 2016.

[2] D. Buhalis and A. Amaranggana, "Smart tourism destinations enhancing tourism experience through personalisation and real-time technologies," *Information and Communication Technologies in Tourism*, pp. 377–389, 2015.

[3] D. Guttentag, "Virtual reality: Applications and implications for tourism," *Tourism Management*, vol. 31, no. 5, pp. 637–651, Oct. 2010.

[4] S. Gössling, D. Scott, and C. M. Hall, "Tourism and sustainability: Emerging new forms of consumption," *Tourism Recreation Research*, vol. 38, no. 3, pp. 239–249, 2013.

[5] J. A. Lee, M. M. Kyle, and C. R. Scott, "Personalization in travel planning: A review of AI applications," *Journal of Travel Research*, vol. 59, no. 6, pp. 1024–1036, Jul. 2020.

[6] Tussyadiah, "A review of research into automation in tourism: Launching the Annals of Tourism Research Curated Collection on AI and Robotics in Tourism," *Annals of Tourism Research*, vol. 81, p. 102883, May 2020.

[7] R. Law, B. Leung, and D. Buhalis, "Information technology applications in hospitality and tourism: A review of publications from 2005 to 2016," *Journal of Travel & Tourism Marketing*, vol. 35, no. 5, pp. 573–598, 2018.

[8] T. Nguyen, C. D. Nguyen, and H. H. Pham, "Augmented reality applications in smart tourism: A review," *IEEE Access*, vol. 8, pp. 206672–206686, 2020.

[9] H. S. Kim and J. H. Kim, "A recommendation system for personalized travel planning using user behavior data," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 1224–1233, Feb. 2021.

[10] M. J. Eppler and J. Mengis, "The concept of information overload: A review of literature from organization science, accounting, marketing, MIS, and related disciplines," *The Information Society*, vol. 20, no. 5, pp. 325–344, Nov. 2004.

[11] P. V. R. K. Mallela and P. S. S. Reddy, "AI-based chatbot for personalized travel assistance," 2020 International Conference on Smart Electronics and Communication (ICOSEC), pp. 1382–1386, Sep. 2020.

[12] S. Wang, J. Jiao, and Y. Huang, "Smart tourism: Tourist recommendation system using machine learning," *Procedia Computer Science*, vol. 176, pp. 1702–1711, 2020.

[13] A. S. Silva, J. M. P. Cardoso, and R. J. F. Rossetti, "A data-driven approach for tourist profile characterization," 2020 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 628–637, Oct. 2020.

[14] Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, "Mobile recommender systems in tourism," *Journal of Network and Computer Applications*, vol. 39, pp. 319–333, Mar. 2014.

[15] E. T. Nuraeni, F. R. Ramadhan, and A. Darmawan, "Development of a sustainable tourism recommendation system based on traveler behavior," 2021 International Conference on Sustainable Information Engineering and Technology (SIET), pp. 125–130, Nov. 2021.

Cite this article as: Suman Kumar Verma, Manish Kumar, Vishal Kumar, Amol Sharma, Wanderlust: a travel experience sharing platform, International Journal of Research in Engineering and Innovation Vol-9, Issue-4 (2025), 193-198. *https://doi.org/10.36037/IJREI.2025.9407*.