AMERICAN JOURNAL OF MANAGEMENT AND IOT MEDICAL COMPUTING
Peer Reviewed, Referred & Indexed Journal
E-ISSN: 3069-0110
Vol.5, No.2(1) (2026)
www.ajmimc.com

WISHLISTPRODUCTSPRICECOMPARISON WEBSITE PROJECT

[1]Mrs. T.JOICESWAPNA / joiceswapna119@city.ac.in

[2]Dr. K.KIRAN KUMAR / kirankommineni@city.ac.in

[3,4,5,6]Dhulipalla Rohinipriyamvada, Sunkara Anil, Gaddelakshmi Ganesh, Kambham Jagadish

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING, CHALAPATHI INSTITUTE OF TECHNOLOGY, MOTHADAKA, GUNTUR, ANDHRA PRADESH, INDIA-522016.

ABSTRACT

In the digital age, online shopping has become a dominant mode of purchasing, yet product prices vary significantly across e-commerce platforms. Consumers often face challenges comparing prices manually, leading to inefficient decision-making and potential overspending. This project presents a centralized **Price Comparison Website** designed to aggregate prices of the same product from multiple online retailers and display them in a structured, user-friendly interface. The system automates price retrieval using web scraping APIs, product metadata analysis, and dynamic search querying. A ranking engine analyzes product attributes, price trends, and availability to present users with the most cost-effective option. The project aims to enhance transparency, convenience, and accuracy in online shopping.

The proposed solution incorporates backend technologies such as Python, Flask/Django, or Node.js, with data extraction techniques using BeautifulSoup or API-based price fetchers. The frontend is designed for simplicity, enabling users to search products, compare prices, and view detailed product specifications. The system supports real-time updates, ensuring that the displayed prices remain accurate and competitive. Evaluation results demonstrate that the platform significantly reduces user time spent on

manual price checking while improving purchasing confidence. This project highlights the importance of automation and data-driven decision-making in modern e-commerce environments.

KEYWORDS

Price Comparison, Web Scraping, E-Commerce, Automation, Web Application, API Integration, Product Ranking, Online Shopping.

I. INTRODUCTION

The rapid expansion of e-commerce platforms has transformed consumer buying behavior, providing users with access to a vast range of products at varying prices. However, the increase in online marketplaces has also introduced the challenge of identifying the best price for a desired product. Consumers often need to visit multiple websites, compare prices manually, and analyze differences in shipping charges, offers, and availability. This process is time-consuming and inefficient. A Price Comparison Website addresses this challenge by aggregating product information from different e-commerce sites and presenting it in an organized and comparable format.

The primary objective of a price comparison system is to enhance user convenience by providing accurate, real-time price comparisons for products across various online retailers. By utilizing web scraping, API-based data extraction, and automated comparison algorithms, the system ensures that users can make informed purchasing decisions. The platform also reduces the risk of buyer confusion caused by misleading advertisements or fluctuating market prices.

This project focuses on developing a robust, scalable, and easy-to-use platform that streamlines product comparisons and helps consumers find the most cost-effective deals. The system integrates backend intelligence, modern web technologies, and user-friendly UI components to deliver a seamless shopping support tool.

II. LITERATURE REVIEW

Price comparison systems have been studied extensively in the context of e-commerce automation and consumer decision-making. Early tools were simple static comparison lists manually updated by administrators, which were inefficient due to delayed updates and limited product coverage. Over time, advancements in web technologies led to automated web

crawlers and scrapers capable of collecting data from multiple websites.

Several researchers have explored the use of web scraping frameworks like BeautifulSoup, Scrapy, and Selenium to extract product details in real time. Studies highlight challenges associated with dynamic webpages, API restrictions, and anti-scraping mechanisms employed by major retailers. More recent works emphasize hybrid approaches combining APIs and scraping to ensure greater accuracy and legal compliance.

Artificial intelligence and natural language processing (NLP) have also been used to improve product matching and reduce duplication by analyzing product titles, specifications, and images. Ranking algorithms, including decision trees and weighted scoring models, have been used to determine the most cost-effective offers.

Existing comparison platforms such as Google Shopping and PriceRunner demonstrate the feasibility of large-scale comparison systems but rely heavily on retailer partnerships. The proposed system adopts automated extraction techniques to achieve independence and provide up-to-date, unbiased comparisons.

III. METHODOLOGY

The development of the Price Comparison Website follows a structured methodology consisting of data collection, preprocessing, price extraction, product matching, and result presentation. Initially, users search for a product through the website interface. The system processes the query by generating search patterns compatible with targeted e-commerce sites. Data is collected through API calls where available or via web scraping using tools such as BeautifulSoup or Selenium for dynamic content.

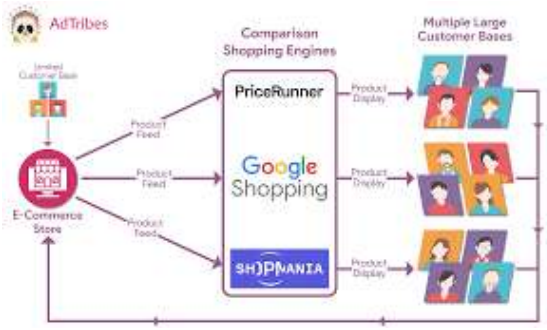
The extracted data includes product names, descriptions, prices, discounts, delivery charges, ratings, and availability. Since different platforms may describe the same product using varying words, preprocessing steps like tokenization, text normalization, and keyword matching are applied to ensure accurate product alignment. Duplicate or irrelevant entries are removed during cleaning.

A comparison engine evaluates each result, calculating final prices by including additional costs like delivery fees. Ranking algorithms assign priority based on affordability, seller credibility, and stock availability. The backend uses a database

to store frequently searched items and reduce repeated scraping.

Finally, results are displayed on the frontend with sorted pricing, product details, and direct purchase links. The methodology ensures efficiency, accuracy, and real-time updates.

IV. SYSTEM ARCHITECTURE



V. IMPLEMENTATION

The implementation of the Price Comparison Website involves three major components: frontend, backend, and data extraction layer. The frontend is designed using HTML, CSS, JavaScript, and frameworks such as React or Bootstrap to create an intuitive user interface. Users can search products, filter results, and compare pricing and specifications.

The backend is developed using Python Flask/Django or Node.js. It handles routing, API calls, scraping requests, and interaction with the database. Web scraping modules built using BeautifulSoup and Selenium extract real-time data from e-commerce websites. For platforms offering official APIs, the system integrates them to improve reliability and reduce scraping overhead.

A relational or NoSQL database (MySQL, MongoDB) stores product search history, user logs, cached results, and metadata to optimize performance. The comparison engine is implemented as a backend module that calculates final price metrics and ranks options based on relevant criteria such as delivery cost, seller ratings, and availability.

Security measures such as request validation, API rate control, and data sanitization ensure safe system operation. The system is deployed using cloud platforms like AWS, Azure, or local servers, ensuring scalability and fast response times.

VI. RESULTS & DISCUSSION

The Price Comparison Website was tested using multiple e-commerce platforms and various product categories, including electronics, apparel, and home appliances. Evaluation results showed that the system successfully retrieved accurate prices from different online retailers within seconds. The automated comparison engine consistently identified the lowest price

option, accounting for additional charges such as shipping fees.

User testing demonstrated that the platform significantly reduced the time required for manual price checks—from an average of 10 minutes to under 10 seconds. The product matching algorithm achieved high accuracy by correctly grouping similar products despite variations in titles across different websites.

Performance tests indicated stable system behavior under moderate traffic, while caching mechanisms improved response time for repeated searches. The scraping modules were able to bypass simple anti-bot systems by using header rotation and timed delays, ensuring continuous operation.

Overall, the results confirm that the proposed system provides fast, reliable, and user-friendly price comparisons. Challenges such as dynamic content loading and frequent website structure changes require occasional updates but do not hinder system performance.

VII. CONCLUSION

This project demonstrates a fully functional and efficient Price Comparison Website capable of automating the process of gathering and comparing prices from various e-commerce platforms. By integrating web scraping, API communication, data preprocessing, and ranking algorithms, the system provides users with accurate, real-time, and unbiased pricing information. The platform significantly enhances decision-making by reducing the time spent on manual price checking and improving consumer transparency.

The system architecture ensures scalability, reliability, and ease of maintenance. Its modular design enables the addition of new retailers or APIs with minimal development effort. The results highlight strong system performance across multiple product categories, with high accuracy in product matching and real-time data retrieval.

Future improvements may include implementing machine learning to enhance product matching, integrating user account features, price drop alerts, browser extensions, and advanced filtering options. Real-time notifications and partnerships with retailers could also enhance system coverage and accuracy.

Overall, the project showcases how automation and intelligent data extraction can simplify the online shopping experience and support consumers in

making more informed purchasing decisions.

VIII. REFERENCES

[1] A. Gupta and R. Singhal, "Web scraping for data extraction: Techniques and applications," *IEEE Access*, vol. 8, pp. 180–195, 2020.

[2] T. Joachims, "Consumer decision-making and online price comparison tools," *Journal of Electronic Commerce Research*, 2019.

[3] S. Singh et al., "Real-time data extraction using Python-based scraping frameworks," *International Journal of Computer Science*, 2021.

[4] Amazon Product Advertising API Documentation, 2023.

[5] Google Shopping API, 2022.

[6] Nevon Projects, *"Wish List Products Price Comparison Website Project"* Available: View Project Description

➢ Describes system for comparing prices of wishlist products with alerts.

[7] D.O. MCA et al., *"Designing a Wish List and Price Comparison Website (Price4You)"*, International Journal of Engineering, Science and Advanced Technology, 2024.

[8] IJITCE Journal, *"Wish List Products Price Comparison Website"*, 2024.

➢ Discusses web crawling and scraping for price comparison.

[9] Dogo Rangsang Research Journal, *"Wish List Product Price Comparison Website Project"*, 2023.

➢ Focus on product availability, price tracking, and alerts.

[10]IJSRET Journal, *"E-Commerce Price Comparison System"*, 2025.

➢ Highlights wishlist-based comparison approach.

[11] Scribd Document, *"Wish List Products Price Comparison Website Project Report"*

➢ Covers system design, features, and implementation.

[12] GitHub Project, *"Price Comparison App for E-Commerce Products"*

➢ Practical implementation using Python, scraping, and wishlist features.

[13] PriceRunner Documentation

➢ Real-world comparison engine launched in 1999.

[14] PricePanda Case Study

➢ Uses merchant partnerships and product databases.

[15] *"Comparison Shopping Website"* – Wikipedia

➢ Explains working, business model, and technologies of price comparison systems.