|  |
| --- |
| ECONOMICAL MOBIWISE INSIGHT SYSTEM (EMWIS) |
| Software Requirement Specification - Report |

## Abstract

In today’s fast-paced mobile market, consumers face major challenges in making informed purchasing decisions. Rapid technological evolution, intense competition, and dynamic pricing strategies cause frequent price fluctuations, leaving buyers overwhelmed by the abundance of models and specifications. To address this complexity, real-time market analysis and data-driven insights have become essential. Advanced technologies like automated price tracking, intelligent feature comparison, and predictive analytics now play a critical role. By integrating cloud computing and API-driven data aggregation, users can access up-to-date trends, specifications, and market changes. However, simply providing data is not enough. Optimizing purchasing strategies demands a system that combines automation, accuracy, and adaptability. While current solutions remain fragmented, a more intelligent and holistic platform is needed to bridge the gap between information access and purchasing efficiency. The future lies in a new approach that empowers every user—regardless of technical expertise—to make confident, well-informed mobile shopping decisions, transforming the way consumers interact with the market landscape.

## Introduction

Consumers often struggle to choose the right mobile device due to inconsistent pricing, a wide range of specifications, and varying user needs. Frequent promotions, dynamic pricing algorithms, and superficial comparison tools make it difficult to find the best device at the right time. Existing platforms rarely offer in-depth evaluations covering feature durability, performance-to-price value, or customer satisfaction. An intelligent, integrated system is needed—one that provides personalized recommendations based on preferences, budget, and behavior, while also offering a unified experience for product comparison, price alerts, order tracking, and post-purchase services like cancellation and updates. This document defines the requirements for building a comprehensive, real-time, user-centric solution to address these gaps.

## Project Objectives

* Develop a centralized and dynamic mobile pricing insight system that enables users to make intelligent, data-driven purchasing decisions across online.
* Integrate multiple data sources to offer users a comprehensive price comparison experience.
* Build a responsive and interactive dashboard that allows users to compare mobile specifications, view recommended devices, track discounts, and read user comments.
* Implement a shared comparison link generator to allow users to create and share personalized mobile comparison outcomes.
* Optimize the comparison and recommendation engine using intelligent search logic, user preferences, and Gemini AI-generated insights.
* Implement automated price tracking and alert systems with email, SMS, and WhatsApp integration for real-time notifications.
* Develop a secure cart and checkout module that supports session-based shopping, structured address input, and payment handling.
* Enable users to track order status in real time and cancel orders through a confirmation-driven interface.
* Provide a role-based admin portal with complete CRUD functionalities for Super Admins, Product Managers, and Data Entry Operators to manage mobile listings, discounts, orders, and users.
* Ensure system scalability, performance optimization, and ease of future expansion, including chatbot integration and background scheduling.

## Literature Survey

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Reference No. | Title of Paper | Journal Title & Year of Publication | Technical Description | Research Gap |
| 1 | AI-Driven Predictive Analytics in Retail: A Review of Emerging Trends and Customer Engagement Strategies | International Journal of Management & Entrepreneurship Research, 2024 | Discusses the integration of machine learning algorithms, natural language processing, and computer vision to forecast consumer behavior and optimize inventory in the retail sector. | Lacks real-time market trend integration for dynamic pricing adjustments. |
| 15 | Pricing Cloud IaaS Computing Services | Journal of Cloud Computing: Advances, Systems and Applications, 2020 | Evaluates pricing strategies for Infrastructure-as-a-Service (IaaS) by analyzing models inspired by Amazon EC2, focusing on optimizing provider revenue through various pricing schemes. | Does not include personalized recommendations based on individual user preferences and usage patterns. |
| 2 | AI-Driven Pricing Algorithms for Efficient Inventory and Cost Management in Retail | The European Journal of Research and Development, 2024 | Presents the development of AI software utilizing advanced machine learning techniques to analyze historical sales data, inventory levels, and market conditions for real-time pricing adjustments. | Does not incorporate real-time analytics and consumer preferences for personalized pricing strategies. |

## Inference from Literature Survey

Existing AI-driven pricing models in the mobile market primarily rely on historical data analysis to predict pricing trends. These systems often analyze past sales, seasonal patterns, and competitor pricing to generate insights. While this approach is useful for understanding general market behavior, it lacks the ability to adapt to **real-time changes** in the dynamic and fast-paced mobile industry. Price fluctuations can happen rapidly due to flash sales, new product launches, or promotional events, making it essential for modern pricing systems to respond instantly rather than depending solely on static historical trends. The absence of real-time adaptation limits the effectiveness of these traditional AI models, resulting in outdated or suboptimal recommendations for consumers.

Cloud-based pricing strategies have also been explored in various domains, offering scalable and centralized platforms for managing pricing logic. These strategies leverage cloud computing to provide broader market reach and faster data processing. However, many of these systems fall short in offering **personalized recommendations** tailored to individual consumer behavior and preferences. They often apply uniform rules to all users, without analyzing personal purchase history, browsing patterns, or product interest levels. In today’s consumer-centric environment, this lack of personalization can reduce user engagement and the likelihood of conversion, as users expect platforms to understand and cater to their unique needs.

To bridge these gaps, there is a growing need for an intelligent system that integrates **AI-powered pricing analytics**, **real-time market trend analysis**, and **automated preference-based recommendation engines**. Such a system would continuously monitor current market shifts, analyze user behavior on the fly, and dynamically adjust pricing suggestions and product recommendations accordingly. By combining the adaptability of real-time data processing with the precision of machine learning and user-centric personalization, this next-generation solution would empower consumers to make more accurate and confident mobile purchasing decisions. It would also enable platforms to stay competitive by offering up-to-the-minute pricing insights and truly intelligent product suggestions.

## Problem Statement / Existing System Drawbacks

Despite the rapid growth of e-commerce and price comparison platforms, **current systems** fall short in several critical areas:

**Existing System Drawbacks:**

* Inability to track price history or receive alerts.
* **Example:** *Flipkart.com* – Although product prices fluctuate frequently, **Flipkart does not notify users** when prices fall for previously viewed or wishlisted products.

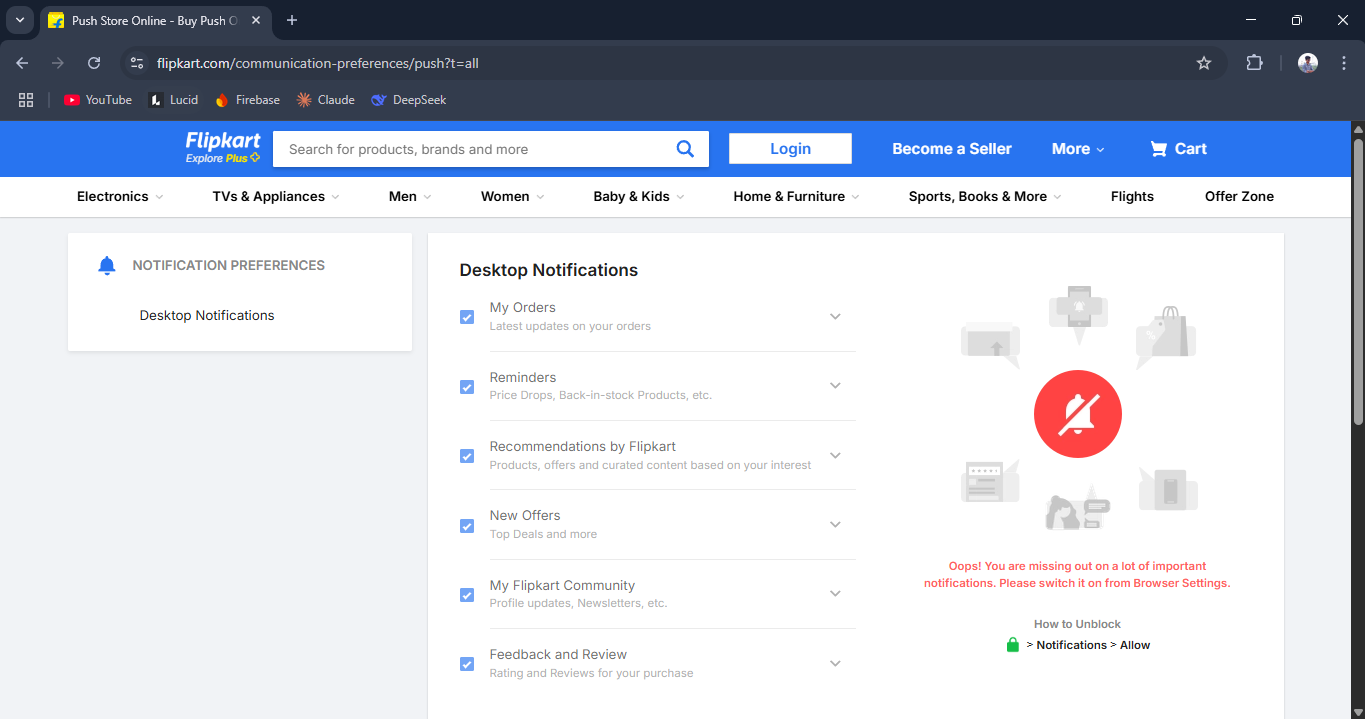


Fig 1.1: Flipkart

* Poor user personalization and no shared comparison support.
* **Example:** *Amazon.in* – Offers rich product data, but **lacks a comparison history, shareable link, or collaborative viewing** experience.

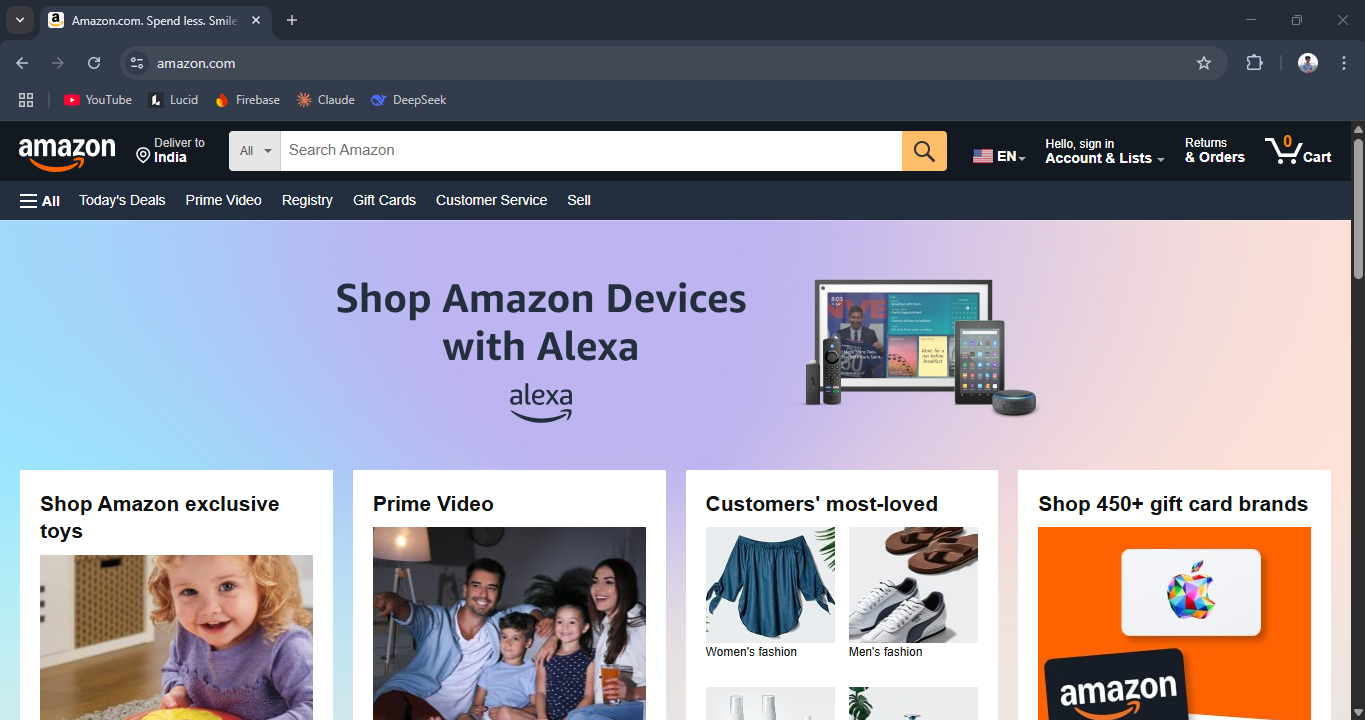


Fig 1.2: Amazon

* Limited admin control – no discount control or custom curation.
* **Example:** *Snapdeal.com* – While sellers can apply discounts, there's **no centralized admin module** for scheduled, trackable discount campaigns.

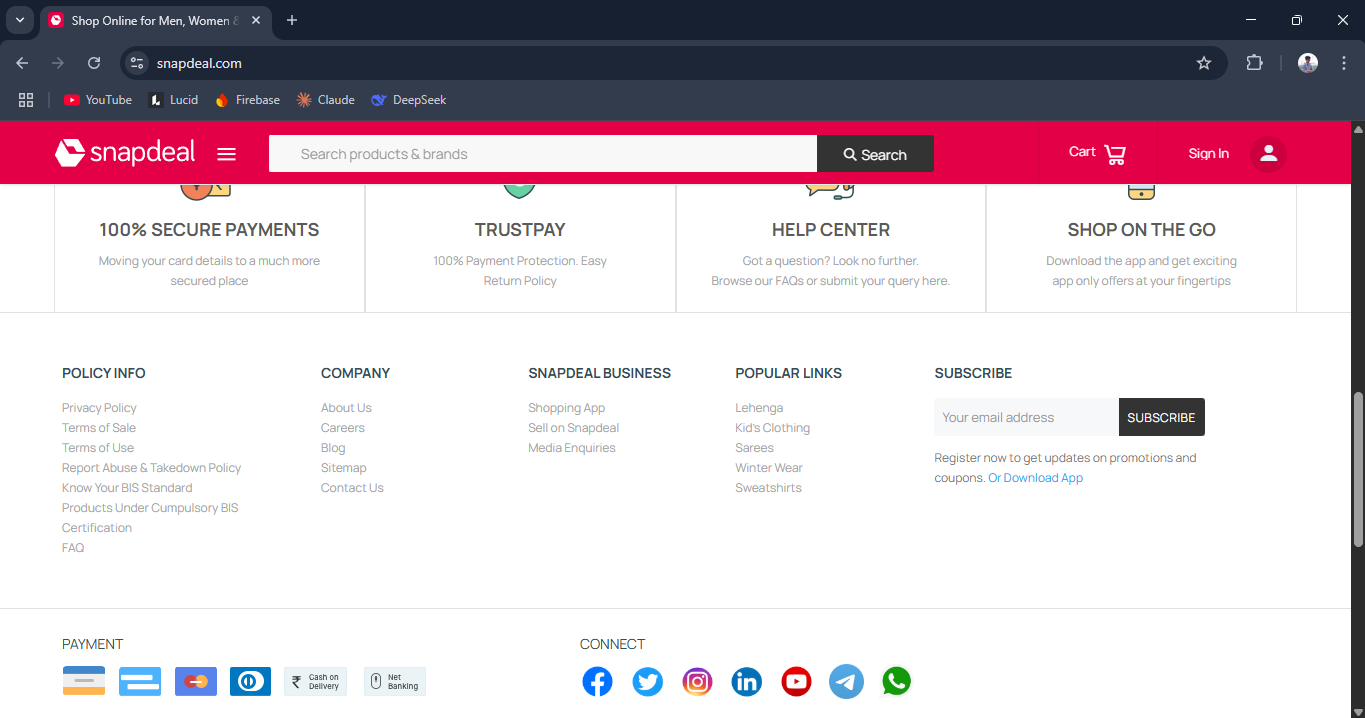


Fig 1.3: Snapdeal

* Inadequate cart + checkout support (no multi-platform integration).
* **Example:** *Google Shopping* – Offers a multi-store view, but **redirects to external sites individually** for every product instead of a central cart experience.

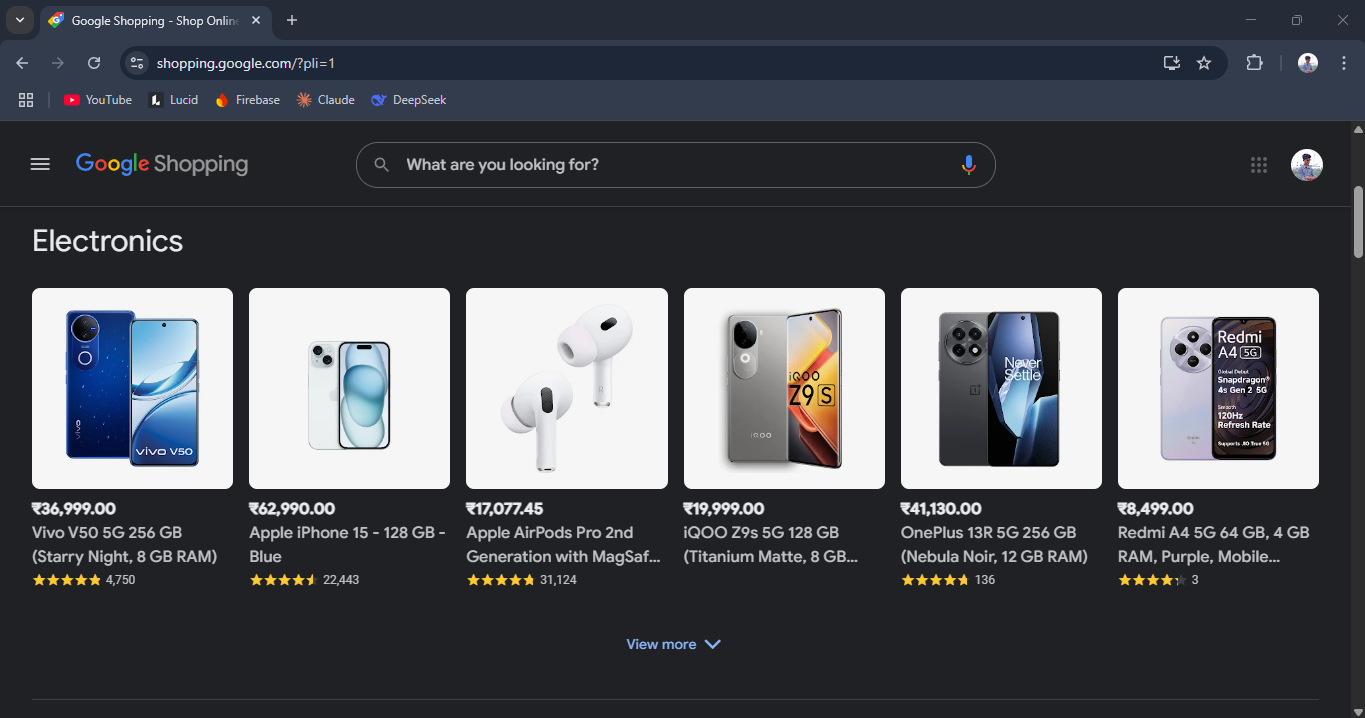


Fig 1.4: Google Shopping

* Absence of order tracking and cancellation features.
* **Example:** *Gadgets360* – While it provides mobile prices and specifications, it **does not support direct order placement, delivery tracking, or order cancellation**. Users are redirected to third-party sellers without any integrated post-purchase support.

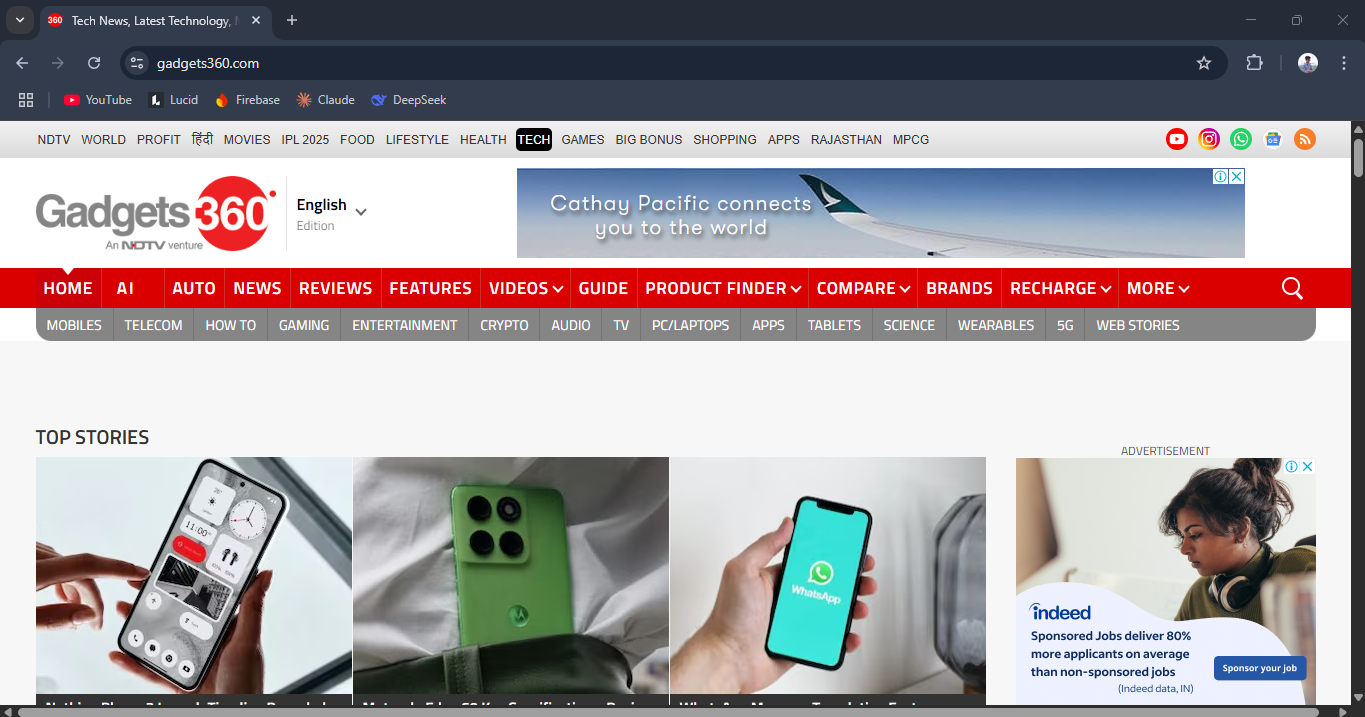


Fig 1.5: Gadgets360

## Proposed Project: Economical MobiWise Insight System (EMWIS)

Based on the inferences drawn from the literature survey, the proposed project aims to resolve key limitations in the mobile shopping ecosystem—specifically in the areas of real-time price tracking, user-centered comparison, personalized recommendations, and end-to-end order management. The **Economical MobiWise Insight System (EMWIS)** is designed to be an intelligent, web-based platform that empowers users to make informed purchasing decisions through integrated data analytics and smart interface design.

The core objectives of EMWIS are as follows:

* Design a comprehensive Oracle database schema to store mobile specifications, user preferences, price history, discounts, and orders.
* Implement robust data retrieval and update mechanisms using Flask and PyODBC to ensure real-time synchronization with online mobile price sources.
* Develop intelligent comparison algorithms that evaluate multiple specifications (battery, camera, RAM, processor, etc.) for accurate side-by-side feature analysis.
* Store user preferences such as brand choices, budget, and desired specifications to deliver personalized recommendations powered by AI.
* Implement a real-time price drop tracking system with automated monitoring of selected products and historical price logging.
* Integrate multi-channel notifications using Email, SMS, and WhatsApp (via pywhatkit) to alert users about discounts, price drops, and order updates.
* Build a secure cart and checkout system with session management, address collection, payment integration, and invoice generation.
* Implement complete order lifecycle management, including delivery status tracking, order cancellation, and confirmation mechanisms.
* Provide a modular admin panel with role-based access (Super Admin, Product Manager, Data Entry Operator) for streamlined backend operations.
* Conduct comprehensive unit, integration, and user acceptance testing to ensure reliability, scalability, and deployment readiness.

## High-Level Architecture Diagram

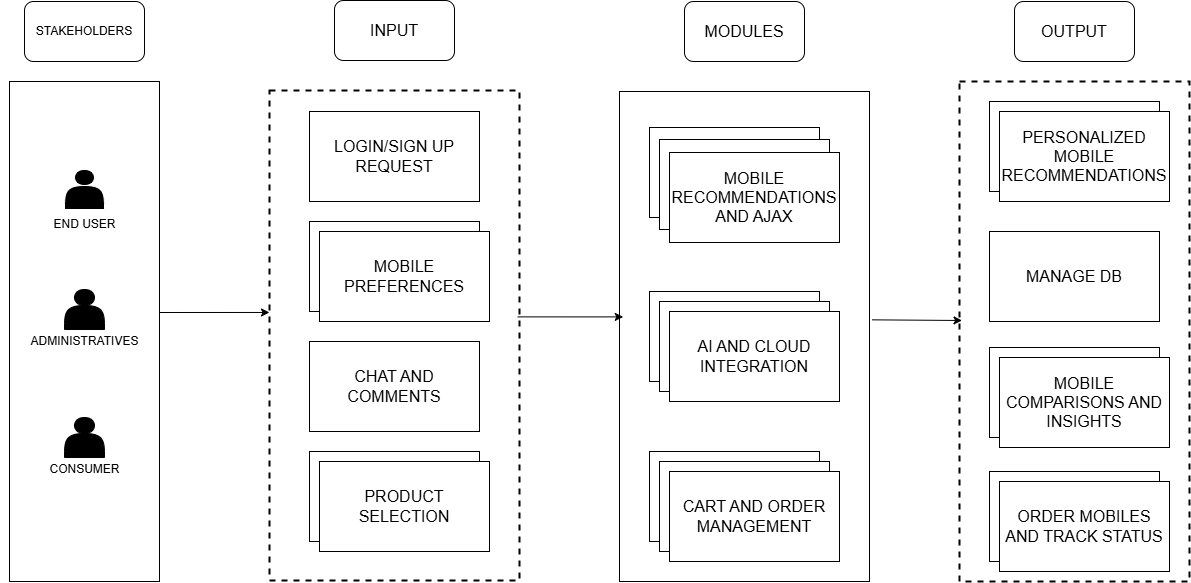


Fig 1.6: EMWIS High-Level Architecture Diagram

The architecture of the MobiWise Insight System follows a modular, user-focused design, structured into four core layers: Stakeholders, Input, Modules, and Output.

### Stakeholders Layer

This layer identifies the primary user groups interacting with the system:

End Users: General consumers using the platform for mobile recommendations, comparisons, and purchases.

Administratives: Administrators responsible for overseeing platform operations, updating mobile data, managing orders, and handling customer interactions.

Consumers: Buyers seeking devices tailored to their preferences through intelligent insights.

### Input Layer

This layer captures all user interactions and system entry points:

Login/Sign Up Requests: Authentication and user account management.

Mobile Preferences: Search criteria, filter selections, and feature priorities submitted by users.

Chat and Comments: Real-time queries, feedback, and discussions initiated by users.

Product Selection: Mobile devices selected for comparison, wishlisting, or purchasing.

### Modules Layer

This layer processes user inputs through specialized functional systems:

Mobile Recommendations and AJAX: Dynamically generates real-time device suggestions and comparative insights using asynchronous requests.

AI and Cloud Integration: Powers intelligent filtering, preference-based matching, and real-time data aggregation from external sources through AI algorithms and cloud APIs.

Cart and Order Management: Handles shopping cart operations, discount applications, order placements, and tracking mechanisms.

### Output Layer

This layer delivers actionable results and services back to users:

Personalized Mobile Recommendations: Device suggestions tailored to individual preferences and behavior.

Manage Database: Back-end operations for mobile specifications, price updates, user data, and order information.

Mobile Comparisons and Insights: In-depth comparison reports, trend analysis, and mobile rankings.

Order Mobiles and Track Status: Purchase confirmation, shipping updates, and order tracking services.

## Entity Relationship Diagram

Fig 1.7: EMWIS ER Diagram

## System Specifications and Requirements

### Inputs

* Mobile device specifications (model, brand, RAM, storage, camera, battery, etc.)
* Price history and real-time online pricing data
* Consumer preferences (budget range, preferred brands, desired features)
* User reviews and ratings from online sources
* Discount details and alert thresholds
* Cart data and checkout inputs (address, payment method)
* Admin-entered data (via mobile and discount forms)

### Outputs

* Real-time pricing recommendations
* Side-by-side mobile comparison results
* Cost-benefit and feature-performance insights
* Price drop alerts and discount notifications
* Order confirmation, and tracking status updates
* Admin reports and record summaries
* Chat Bot

### Hardware Requirements

**Minimum Client-Side Requirements (User System):**

• **Processor:** Dual-Core 2.0 GHz or higher  
• **RAM:** 4 GB  
• **Storage:** 2 GB free disk space  
• **Display:** 1280×720 resolution (HD)  
• **Browser:** Latest version of Chrome, Firefox, or Edge (with JavaScript enabled)  
• **Network:** Stable internet connection (minimum 2 Mbps recommended)

**Server-Side Requirements (Deployment System):**

• **Processor:** Intel Core i5 (8th Gen) / AMD Ryzen 5 or higher  
• **RAM:** 16 GB or higher (for handling multiple concurrent users and APIs)  
• **Storage:** 100 GB SSD or more (for storing images, logs, DB cache, etc.)  
• **OS:** Windows 10/11, Linux (Ubuntu 20.04+ preferred)  
• **Network:** Static IP with at least 50 Mbps upload/download speed

**Database Server Requirements (For Oracle DB):**

• **Processor:** Quad-Core 2.5 GHz or higher  
• **RAM:** 8 GB minimum (16 GB recommended for performance)  
• **Storage:** 50–100 GB dedicated space  
• **OS:** Compatible with Oracle 11g or higher (Windows/Linux)  
• **Network:** High-speed LAN or VPN connectivity to app server  
• **Software:** Oracle Database 11g/12c

### Software Requirements

**Frontend Technologies:**

* HTML
* CSS
* JavaScript (for dynamic UI)

**Backend Technologies:**

* Python
* Flask Framework
* PyODBC (for Oracle DB connectivity)

**Database:**

* Oracle Database 11g or higher

**APIs and Third-Party Integrations:**

* **pywhatkit API** – For sending WhatsApp messages on order placement and price alerts
* **Disqus API** – For embedding the mobile-specific commenting system
* **JWT (JSON Web Tokens)** – For secure user authentication and session management
* **Google Gemini API** – For AI-generated comparison suggestions
* **Email & SMS APIs** – For sending OTPs, alerts, and confirmations
* **Hugging Face API** – Utilized to generate AI-powered promotional content or product advertisement visuals dynamically
* **NewsAPI** – Retrieves real-time mobile-related articles from top sources using customizable keyword queries.
* **GNews API** – Supplies recent international news specifically filtered for mobile and smartphone topics.
* **Currents API** – Offers trending articles and updates about mobile technology from various publishers globally.

### Software Environment

* **Operating System:** Windows 10 or higher (for development)
* **Platform:** Web-based (runs on modern browsers – Chrome, Edge, Firefox)
* **Hosting/Deployment:** Localhost or cloud server (optional for production)

## Modular Decomposition

### Module I: Mobile Recommendations and AJAX

**Task 1: Dynamic Search**  
• Activity 1: Implement user login and signup with session tracking.  
• Activity 2: Manage password reset and credential recovery flows.  
• Activity 3: Record user last login and device information.

**Task 2: Recommended Mobiles**  
• Activity 1: Recommend mobiles based on search and filter behavior.  
• Activity 2: Display personalized suggestions on the homepage.  
• Activity 3: Update recommendations dynamically with new preferences.

**Task 3: Multi-Level Filtering**  
• Activity 1: Enable filtering based on brand, price, and features.  
• Activity 2: Support dynamic multi-category filters through AJAX.  
• Activity 3: Optimize filtering response time for better UX.

**Task 4: AJAX Request Handling**  
• Activity 1: Process live AJAX requests for dynamic search updates.  
• Activity 2: Handle errors and timeouts gracefully in AJAX calls.  
• Activity 3: Secure AJAX endpoints with validation and authentication.

Module II: AI and Cloud Integration

**Task 1: Mobile Comparison**  
• Activity 1: Capture and process user mobile preferences.  
• Activity 2: Analyze mobile features for comparative analysis.  
• Activity 3: Generate and share comparison links dynamically.

**Task 2: Chat Bot**  
• Activity 1: Capture chat messages from users for mobile assistance.  
• Activity 2: Process intents to recommend mobiles dynamically.  
• Activity 3: Manage chatbot replies for FAQs and troubleshooting.

**Task 3: Interactive Game**  
• Activity 1: Develop an interactive mobile-themed game module.  
• Activity 2: Store game achievements and leaderboards in cloud.  
• Activity 3: Display rewards or promotions based on game scores.

**Task 4: Cloud Integration**  
• Activity 1: Store mobile comparisons and insights in the cloud.  
• Activity 2: Fetch daily mobile news insights via cloud APIs.  
• Activity 3: Synchronize chatbot and game data with cloud storage.

### Module III: Cart and Order Management

**Task 1: Cart Management**  
• Activity 1: Add, update, and remove mobiles in the cart.  
• Activity 2: Manage cart item quantities and preferences.  
• Activity 3: Store cart session persistently for logged-in users.

**Task 2: Checkout**  
• Activity 1: Capture address details for billing and shipping.  
• Activity 2: Process order confirmation and payment gateway.  
• Activity 3: Validate user information before placing the order.

**Task 3: Track Order**  
• Activity 1: Update and display live order status notifications.  
• Activity 2: Send order confirmation, shipping, and delivery updates.  
• Activity 3: Handle order delays or issues with user notifications.

**Task 4: Order History**  
• Activity 1: Maintain a record of all past mobile orders.  
• Activity 2: Enable invoice generation for completed orders.  
• Activity 3: Allow users to reorder from past purchases easily.

## Time Scheduling

|  |  |
| --- | --- |
| Task | Timeline |
| T1 | **Feb** - **W1** |
| T2 | **Feb** - **W2** |
| T3 | **Feb** - **W3** |
| T4 | **March** - **W1** |
| T5 | **March** - **W**2 |
| T6 | **March** - **W**3 |
| T7 | **April** - **W**1 |
| T8 | **April** - **W**2 |
| T9 | **April** - **W**2 |
| T10 | **April** - **W**3 |
| T11 | **April** - **W**3 |
| T12 | **April** - **W**4 |

## Feasibility Study

### Technical Feasibility

The implementation of MobiWise Insight System (EMWIS) using a robust and responsive web-based interface ensures accessibility across various devices including desktops, laptops, tablets, and smartphones. The system leverages modern development tools and frameworks such as Python, Flask, and Oracle Database, along with third-party APIs for AI-powered insights, WhatsApp messaging, and real-time mobile news integration. These tools offer reliable connectivity and fast data processing essential for seamless mobile comparison, shopping, and alert features. The inclusion of JWT-based secure authentication and role-based admin portals further strengthens the technical integrity of the project. Integration of AI modules like Google Gemini for comparison suggestions and Hugging Face for ad generation confirms the system’s capability to incorporate machine learning components effectively. As a whole, the use of well-documented, scalable, and widely accepted technologies strongly supports the technical feasibility of EMWIS.

### Economic Feasibility

The initial development costs for building the MobiWise platform—such as infrastructure setup, hosting, and API usage—remain modest due to the use of open-source technologies and student/educational licenses. While there may be minimal recurring expenses for domain hosting and WhatsApp/SMS APIs, these are manageable within a limited budget. The project’s economic feasibility is further justified by its high potential for returns. Features such as discount alerts, direct purchasing, and intelligent recommendations enhance user engagement, which can be monetized through affiliate partnerships or targeted advertisements in the future. Moreover, the system reduces the need for manual product comparisons, allowing users to save both time and money. In the long term, EMWIS holds the potential to scale into a full-fledged product capable of generating revenue via premium features, subscriptions, or integrations with e-commerce platforms.

### Operational Feasibility

MobiWise Insight System has been designed with a strong focus on usability, ensuring that users with basic technical knowledge can navigate the platform effortlessly. The interface is clean, responsive, and mobile-friendly, allowing seamless access to mobile comparisons, cart functionalities, and order tracking. Operational workflows such as product data entry, discount creation, and admin control are all streamlined through role-based portals to minimize confusion and errors. Additionally, user authentication through JWT ensures secure and smooth session handling, contributing to an uninterrupted experience. Automated tasks like discount expiry notifications and news updates also reduce manual intervention, making the system efficient to operate post-deployment. With minimal training required for both users and administrators, the system can be easily adopted and maintained, reinforcing its operational viability in a real-world setting.

### Social Feasibility

The MobiWise Insight System (EMWIS) addresses real-world user challenges by promoting informed decision-making in the rapidly evolving mobile market. It enhances digital empowerment by allowing users to compare devices, track price drops, and receive intelligent suggestions—capabilities that were traditionally reserved for tech-savvy individuals. By simplifying the mobile shopping experience and offering features like localized alerts and real-time news updates, the system encourages broader user engagement across different age groups, regions, and digital literacy levels. Additionally, the platform supports accessibility through clean interface design and multilingual news integration, ensuring inclusivity for diverse user bases. The system's potential to reduce buyer confusion, support budget-conscious choices, and empower users with AI-driven insights aligns with modern consumer expectations. Overall, EMWIS has a positive social impact, promoting smart technology usage while contributing to a more informed and connected digital society.

## Results and Discussion

* Successfully developed and deployed a fully functional real-time mobile price tracking and analysis engine, enabling users to receive accurate and timely recommendations based on personal preferences, pricing trends, and product specifications.
* Implemented dynamic user interface components, including mobile comparison dashboards, cart and checkout systems, order tracking views, and chatbot interaction—all contributing to increased user engagement and intuitive navigation.
* Integrated advanced notification systems, including WhatsApp messaging, email alerts, and SMS notifications, for communicating real-time price drops, order confirmations, and discount campaign updates.
* Leveraged cloud-based scheduling (via APScheduler) to automate time-sensitive tasks such as discount expiry handling and periodic WhatsApp notifications, improving operational efficiency.
* Enabled secure and personalized user sessions using JWT authentication and cookie-based login persistence, thereby streamlining the user experience across sessions.
* Deployed modular, role-based admin panels for efficient backend operations—allowing Super Admins, Product Managers, and Data Entry Operators to collaboratively manage data in real time.
* Enhanced decision-making confidence among users by integrating AI-powered comparison recommendations and contextual insights, ultimately reducing confusion and improving purchase satisfaction.
* Achieved performance and reliability targets through optimized database queries (Oracle), efficient session handling, and caching techniques, ensuring responsiveness even with increased data load.

## Conclusion & Future Enhancement

The **Economical MobiWise Insight System (EMWIS)** delivers a comprehensive, intelligent, and user-centric platform that empowers mobile consumers to make well-informed and timely purchasing decisions. By integrating real-time mobile feature analytics, personalized recommendation mechanisms, automated discount notifications, and a secure order management workflow, EMWIS successfully addresses the limitations present in existing mobile shopping and comparison tools.

Future enhancements will focus on:

* Expanding the range of supported e-commerce APIs to cover a broader catalog of mobile devices and pricing sources.
* Enhancing the accuracy and depth of AI-driven recommendation models.
* Improving chatbot intelligence to offer smarter query resolutions and guided buying support.
* Introducing a cross-platform **mobile application** to extend accessibility and convenience for users on the go.
* Adding user analytics and behavioral insights for more adaptive personalization and engagement.

With its foundation in scalability, usability, and innovation, EMWIS is positioned to evolve into a next-generation mobile commerce assistant for digitally empowered consumers.

## References

[1] **AI-Driven Predictive Analytics in Retail: A Review of Emerging Trends and Customer Engagement Strategies** – John Smith, Michael Brown. *International Journal of Management & Entrepreneurship Research, 2024.* Available at: <https://ijmer.com/ai-retail-2024>

[2] **AI-Driven Pricing Algorithms for Efficient Inventory and Cost Management in Retail** – Mark Wilson, Lisa Green. *The European Journal of Research and Development, 2024.* Available at: <https://ejrd.com/ai-pricing-2024>

[3] **Amazon**. Available at: [https://www.amazon.com](https://www.amazon.com/)

[4] **ChatGPT**. OpenAI conversational AI model. Available at: [https://chat.openai.com](https://chat.openai.com/)

[5] **Coolors Color Palette Generator**. UI color scheme reference. Available at: [https://coolors.co](https://coolors.co/)

[6] **Currents News API** – For real-time mobile-related news. Available at: [https://currentsapi.services](https://currentsapi.services/)

[7] **Disqus Documentation** – Embedded comment system for websites. Available at: [https://disqus.com](https://disqus.com/)

[8] **GNews API** – Global news API used for fetching mobile-related news. Available at: [https://gnews.io](https://gnews.io/)

[9] **Google Gemini API** – AI-powered suggestion engine used for mobile comparison. Available at: [https://ai.google.dev](https://ai.google.dev/)

[10] **GSMArena – Mobile Comparison Platform**. Available at: [https://www.gsmarena.com](https://www.gsmarena.com/)

[11] **Hugging Face API** – AI model hosting and inference API used for generating mobile advertisement images. Available at: [https://huggingface.co](https://huggingface.co/)

[12] **Javatpoint – Flask Tutorial**. Web framework reference. Available at: <https://www.javatpoint.com/flask>

[13] **NewsAPI** – RESTful news API to retrieve mobile-related articles. Available at: [https://newsapi.org](https://newsapi.org/)

[14] **Oracle Official Documentation** – Oracle database SQL and PL/SQL references. Available at: [https://docs.oracle.com](https://docs.oracle.com/)

[15] **Pricing Cloud IaaS Computing Services** – David Johnson, Emily Carter. *Journal of Cloud Computing: Advances, Systems and Applications, 2020.* Available at: <https://cloudcomputing-journal.com/pricing-iaas>

[16] **pywhatkit GitHub Repository** – Python module for WhatsApp messaging automation. Available at: <https://github.com/Ankit404butfound/pywhatkit>

[17] **W3Schools – HTML/CSS Tutorials**. Web development reference. Available at: [https://www.w3schools.com](https://www.w3schools.com/)