AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM

(PROJECT PHASE- II)

submitted in partial fulfillment of the requirements

for the award of the degree in

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING
ARTIFICIAL INTELLIGENCE

by

ANUSHANTH.K (211211101018)

ARAVIND.A (211211101021)

BADRESH. B (211211101025)

DEPARTMENT

OF

COMPUTER SCIENCE AND ENGINEERING

APRIL 2025

Dr. M.G.R.
EDUCATIONAL AND RESEARCH INSTITUTE
DEEMED TO BE UNIVERSITY
University with Graded Autonomy Status
(An ISO 21001 : 2018 Certified Institution)
Periyar E.V.R. High Road, Maduravoyal, Chennai-95. Tamilnadu, India.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report (Project Phase-II) is the Bonafide work of **Mr. ANUSHANTH.K Reg. No 211211101018**, **Mr. ARAVIND.A Reg. No 211211101021**, **Mr. BADRESH.B Reg. No 211211101025**, who carried out the project entitled **"AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM"** under our supervision from Dec 2024 to April 2025.

Internal Guide	**Project Coordinator**	**Department Head**
Mr. Subrahmanyam Nandigam Assistant Professor Department of CSE, Dr. M.G.R Educational and Research Institute, Deemed to be University	Dr. F. Jerald Associate Professor Dr. P. Dinesh Kumar Associate Professor Dr. M.G.R Educational and Research Institute, Deemed to be University	Dr. S. Geetha Professor & Head Department of CSE, Dr. M.G.R Educational and Research Institute, Deemed to be University

Submitted for Viva Voce Examination held on _____

Internal Examiner **External Examiner**

Dr. M.G.R.
EDUCATIONAL AND RESEARCH INSTITUTE
DEEMED TO BE UNIVERSITY
University with Graded Autonomy Status
(An ISO 21001 : 2018 Certified Institution)
Periyar E.V.R. High Road, Maduravoyal, Chennai-95. Tamilnadu, India.

DECLARATION

We **Mr. ANUSHANTH K (211211101018), Mr. ARAVIND A (211211101021), Mr. BADRESH B (211211101025),** hereby declare that the Project Report (Project Phase-II) entitled **"AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM"** is done by us under the guidance of **Mr. SUBRAHMANYAM NANDIGAM** is submitted in partial fulfillment of the requirements for the award of the degree in **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING ARTIFICIAL INTELLIGENCE.**

.

DATE:

PLACE:

1. **ANUSHANTH. K**
2. **ARAVIND. A**
3. **BADRESH. B**

SIGNATURE OF THE CANDIDATES

Dr. M.G.R.
EDUCATIONAL AND RESEARCH INSTITUTE
DEEMED TO BE UNIVERSITY
University with Graded Autonomy Status
(An ISO 21001 : 2018 Certified Institution)
Periyar E.V.R. High Road, Maduravoyal, Chennai-95. Tamilnadu, India.

ACKNOWLEDGEMENT

We would first like to thank our beloved Respected Chancellor **Thiru.Dr. A.C. SHANMUGAM, B.A., B.L.,** President **Er. A.C.S. Arunkumar, B.Tech., M.B.A.,** and Secretary **Thiru A. RAVIKUMAR** for all the encouragement and support extended to us during the tenure of this project and our years of studies at his wonderful University.

We express my heartfelt thanks to our Vice Chancellor **Prof. Dr. S. GEETHALAKSHMI** in providing all the support of my Project (Project Phase-II).

We express my heartfelt thanks to our Head of the Department, **Prof. Dr. S. GEETHA**, who was actively involved and very influential from the start till the completion of our project.

Our sincere thanks to our Project Coordinators **Dr. P. DINESH KUMAR** Associate Professor**, Dr. F. JERALD** Associate Professor, Department of CSE and Project guide, **Mr. SUBRAHMANYAM NANDIGAM** Assistant Professor**, Department of CSE** for their continuous guidance and encouragement throughout this work, which has made the project a success.

We would also like to thank all the teaching and non-teaching staffs of Computer Science and Engineering department, for their constant support and the encouragement given to us while we went about to achieving my project goals.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

AI	-	Artificial intelligence
ML	-	Machine learning
NLP	-	Natural Language Processing
UML	-	Unified modeling language
IDE	-	Integrated development environment
ER	-	Entity Relationship Diagram
CF	-	Collaborative Filtering
CBF	-	Content Based Filtering
RBA	-	Rule-Based Filtering
KBRF	-	Knowledge-Based Recommendation Filtering
SA	-	Sentiment Analysis
UI	-	User Interface
NLTK	-	Natural Language Tool Kit
API	-	Application Program Interface
GPU	-	Graphics Processing Unit
CPU	-	Central Processing Unit
KBRS	-	Knowledge Based Recommendation System

ABSTRACT

This paper introduces a new, AI-powered technological product suggestion and recommendation system that aims to simplify the decision-making process for consumers, particularly those with limited technical knowledge. Traditional recommendation systems, such as Content-Based and Collaborative Filtering, often lack personalization and fail to address individual user needs effectively. To overcome these limitations, this research proposes a Preference-Based Algorithm, designed to function like a real salesperson by engaging users in an interactive dialogue. The system interacts with users through a chatbot, asking key questions about budget, usage type, portability, and performance to understand their requirements. It utilizes Knowledge-Based Filtering combined with a rule-based scoring system to rank and recommend the most suitable products. To enhance its ability to process user inputs and extract relevant preferences, the system integrates a Fine-Tuned AI Model, ensuring a seamless and natural conversational experience. Furthermore, the system is designed to provide real-time product links, allowing users to make informed purchasing decisions. Primarily targeted at tech beginners (75% of users), the solution simplifies product selection by offering accurate, user-specific recommendations. By combining AI-driven decision-making, Natural Language Processing (NLP), and a structured knowledge base, this Preference-Based Algorithm establishes a new benchmark for intelligent product recommendation systems.

Keywords: *AI-Powered Recommendation System, Natural Language Processing (NLP), Conversational AI Chatbot, Preference-Based Algorithm, Knowledge-Based Filtering, Machine Learning Models, Fine-Tuned AI Model, User-Specific Product Recommendation, Technological Product Selection*

MAJOR DESIGN CONSTRAINTS AND DESIGN STANDARDS TABLE

Student Group	**BADRESH B 211211101025**	**ARAVIND A 211211101021**	**ANUSHANTH K 211211101018**
Project Title	**AI-Powered Technological Product Suggestion and Recommendation System**		
Program Concentration Area	Artificial Intelligence and Machine Learning, Recommendation Systems, Human-Computer Interaction, Natural Language Processing		
Constraints Example			
Economic	Yes		
Environmental	Yes		
Sustainability	Yes		
Implementable	Yes		
Ethical	Yes		
Health and Safety	Yes		
Social	Yes		
Political	No		
Other	No		
Standards			
1	IEEE 12207 (Software Life Cycle Processes)		
2	ISO 25010 (System and Software Quality Models)		
3	WCAG 2.1 (Web Content Accessibility Guidelines)		
Prerequisite Courses for the Major Design Experiences	1. Artificial Intelligence 2. Machine Learning 3. Human-Computer Interaction 4. Natural Language Processing 5. Web Development (HTML/CSS/JavaScript) 6. Python Programming		

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The speedy development of technology has resulted in an overwhelming array of consumer electronics, and it is challenging for consumers to select products that suit their requirements best. With the ongoing introduction of new devices, including laptops, mobile phones, and other electronic devices, consumers particularly those with limited technical expertise find it challenging to make the right choices because of the numerous options and absence of professional advice. The rising complexity of product specifications, differences in price levels, and the availability of multiple brands all contribute to the confusion, rendering the selection process laborious and time-consuming. Therefore, there is a high demand for smart recommendation systems that can make product selection easier and give users customized suggestions according to their needs.

The world e-commerce industry is growing exponentially, fueled by growing internet penetration, smartphone usage, and the broad adoption of online channels for consumption. Based on the latest estimates, e-commerce users are likely to increase by more than 60 million from 2024 to 2029. In India alone, the size of the e-commerce market stood at $123 billion in 2024 and is estimated to grow to $300 billion by 2029 at a compound annual growth rate (CAGR) of about 20% **[Figure 1]**. This boom in e-commerce means that an increasing number of consumers are turning to digital media, making it even more essential to have more effective and simpler product recommendation systems that can resolve their issues and simplify their buying process. Despite this explosive growth, consumers still face a number of problems while making purchases online, making it a necessity to come up with creative solutions that make their experience better.

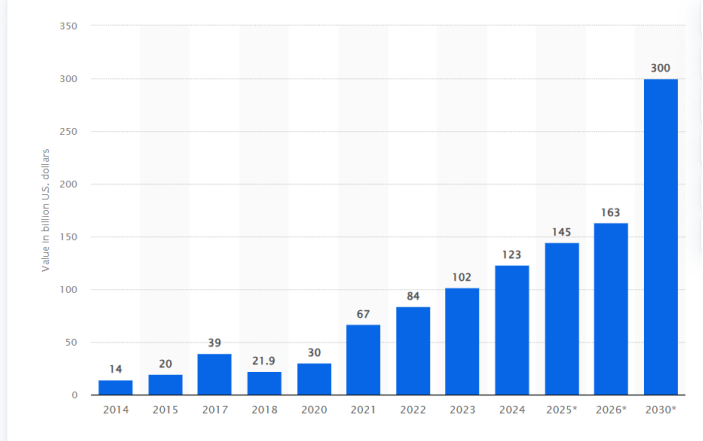

Figure 1. Revenue of growth of E-Commerce users

Perhaps the greatest of all online shopping challenges faced by consumers is the inefficiency in search. Most e-commerce websites have very simplistic keyword-based searching functionality that generally cannot return meaningful results as desired by users. Consumers tend to spend a lot of time looking for a precise product they want but fail because the filtering options are poor, suggestions are not relevant and cannot support the requirements of users. Consequently, users are left to surf through hundreds of options, wasting hours looking for the correct product without being given relevant suggestions. Such inefficiency not only makes users frustrated but also results in abandoned carts and lost sales opportunities for e-commerce websites.

Hidden charges are another primary issue with online shopping. Most e-commerce sites promote products at appealingly favorable base prices, but then tack on extra fees during checkout, including service charges, tax, and shipping costs. The resulting transparent price policy will mostly have customers paying much more than they anticipated, and they will be upset and less trusting online sites. Hidden fees tend to discourage intending buyers from following through with purchases, eventually influencing the general expansion and reputation of e-commerce enterprises.

Moreover, the absence of individualized guidance and professional consultation is still a major concern with online shopping. In contrast to traditional retail shops, where intending buyers can access guidance from shop attendants to appreciate product details and compare functionalities, online sites may not have instantaneous customer care. Most users, especially those who lack technical expertise, experience difficulties in understanding technical terminologies, comparing models, and making informed buying decisions. The lack of professional guidance complicates the process of selection, causing confusion and hesitation on the part of consumers. This necessitates the use of AI-based solutions that can fill this void and offer users intelligent and context-specific suggestions.

To counter these challenges, the use of AI-based chatbots and recommendation systems has been increasing at a rapid pace. AI-based chatbots are revolutionizing the e-commerce sector by offering real-time support, enhancing user experience, and automating customer support interactions. The global market for chatbots was worth $5.4 billion in 2024 and is expected to grow to $15.5 billion by 2029 at a CAGR of around 23% **[Figure 2]**. This considerable expansion highlights the growing dependence on AI technologies to raise digital customer experiences.

Recommendation systems based on AI drive product selection easier by examining user

preferences, purchase history, and individual requirements to recommend the best options. In contrast to conventional search capabilities, AI-based recommendation systems are capable of dynamically adjusting to the specific needs of individual users, providing personalized recommendations that fit their budget, usage habits, and feature requirements. Such systems not only improve the shopping experience but also result in greater conversion rates for e-commerce companies by ensuring customers locate the correct products in a timely and efficient manner.

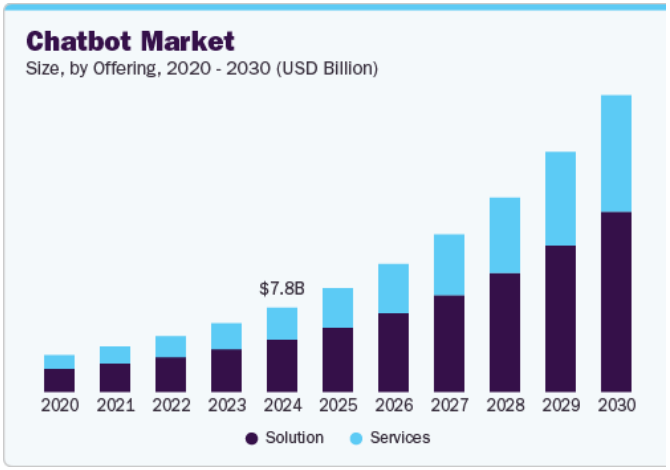

Figure 2. Global market of chatbots

The application of AI in e-commerce is transforming how consumers engage with online shopping websites. By resolving issues like inefficient search capabilities, stealthy fees, and the absence of expert recommendations, recommendation systems and chatbots powered by AI are turning online shopping into an easier and more enjoyable experience. With the advancement of technology, the application of AI in e-commerce is likely to grow even more, providing increasingly advanced solutions that meet varied consumer requirements. The fast-paced integration of AI-based recommendation systems points towards the increasing need for smart and automated solutions that facilitate the decision-making process, ultimately enhancing customer satisfaction and accelerating e-commerce growth. In a world where online shopping is the norm, spending on AI-driven recommendation systems is not only a competitive edge but a requirement for e-commerce sites that are determined to keep up with the changing needs of today's consumers.

1.2 OBJECTIVE OF THE PROJECT

The primary objective of this project is to develop an intelligent chatbot that provides personalized and relevant product recommendations within the technological domain. The system is designed to assist users in making informed decisions by suggesting suitable technological products based on their specific preferences, requirements, and budget. By

engaging users through targeted and meaningful questions, the chatbot will effectively capture individual needs and expectations, thereby filtering and presenting the most appropriate product recommendations. Additionally, the project aims to build a knowledge-based recommendation engine that simulates the thought process of a human tech consultant, making the system more intuitive and user-friendly. It seeks to simplify the product discovery process, especially for non-technical users, through a conversational interface powered by natural language understanding (NLU). The chatbot will also enable real-time product search, filtering, and comparison, while ensuring that recommendations are transparent, explainable, and aligned with user goals. Moreover, the system will be designed to enhance accessibility, support integration of product links, and offer useful resources such as tutorials, reviews, and DIY repair guides. Ultimately, the project aims to demonstrate the effectiveness of AI in improving user engagement and personalization in tech-based recommendation systems, with a scalable architecture that can be extended to various other product categories beyond laptops.

- **Deliver Intelligent Product Recommendations:** Develop an AI-powered system that utilizes machine learning algorithms and natural language processing to suggest the most suitable technological products—such as laptops and mobile phones—based on user preferences, budget, and usage scenarios.
- **Enable Conversational User Interaction:** Integrate a responsive and intuitive chatbot that interacts naturally with users, gathers specific product requirements through dialogue, and offers real-time recommendations and technical assistance.
- **Enhance User Personalization:** Design a dynamic recommendation engine capable of adapting to individual user needs by analyzing preferences, usage intent (e.g., gaming, education, professional work), and feature priorities through rule-based and knowledge-based filtering.
- **Incorporate Comprehensive Tech Assistance:** Extend the chatbot's functionality to provide additional support such as repair tutorials, DIY guides, and answers to common technical questions, offering users a holistic tech consultancy experience.
- **Integrate Real-Time Product Data:** Leverage external APIs to fetch up-to-date product specifications, availability, and pricing from online sources, ensuring users receive current and relevant recommendations.

- **Support Scalable and Secure Deployment:** Ensure the system is built on a modular, scalable architecture with secure data handling, cloud integration, and multilingual accessibility to reach a wider audience while maintaining data integrity and system reliability.

1.3 PROBLEM STATEMENT

The problem addressed by our project aligns on the overwhelming variety of technological products available on the market, making it challenging for consumers to identify options that best suit their unique preferences and other's needs. users often face confusion and difficulty in making well-informed purchasing decisions. Traditional recommendation systems may lack the ability to interact with users in an intuitive and personalized manner, failing to fully understand their nuanced requirements. This project seeks to bridge this gap by creating an intelligent chatbot-based system

CHAPTER 2
REQUIREMENT ANALYSIS

2.1 LITERATURE SURVEY

- **Gorde et al. (2025)** [16] developed an AI-powered e-commerce recommendation system that combines collaborative and content-based filtering. The system personalizes product suggestions by analyzing user browsing history and preferences, aiming to enhance online shopping experiences.
  Gorde, A. S., Jambhalkar, R. S., & Patil, A. (2025). AI-Powered Recommendation System for Enhanced E-Commerce Experience. *International Research Journal of Modernization in Engineering Technology and Science*, 7(1). https://www.irjmets.com/uploadedfiles/paper/issue_1_january_2025/66372/final/fin_irjmets1737958738.pdf?utm_source=chatgpt.com

- **Camilleri & Troise (2023)** [17] conducted a systematic review on the use of AI chatbots in the tourism industry. Their analysis highlights the benefits and challenges of implementing chatbot recommender systems in travel and hospitality services.
  Camilleri, M. A., & Troise, C. (2023). Chatbot Recommender Systems in Tourism: A Systematic Review and a Benefit-Cost Analysis. *ResearchGate*. https://www.researchgate.net/publication/367565901_Chatbot_Recommender_Systems_in_Tourism_A_Systematic_Review_and_a_Benefit-Cost_Analysis

- **Wu et al. (2020)** [18] surveyed the application of Graph Neural Networks (GNNs) in recommender systems. They categorized GNN-based recommendation models and analyzed challenges in applying GNNs to different types of data.
  Wu, S., Sun, F., Zhang, W., Xie, X., & Cui, B. (2020). Graph Neural Networks in Recommender Systems: A Survey. arXiv:2011.02260. https://arxiv.org/abs/2011.02260

- **Zhao et al. (2020)** [19] introduced RecBole, a unified framework for developing and evaluating recommendation algorithms. The library implements various models and provides tools for standardizing recommender system research.
  Zhao, W. X., Mu, S., Hou, Y., Lin, Z., Chen, Y., Pan, X., ... & Wen, J. R. (2020). RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. arXiv:2011.01731. https://arxiv.org/abs/2011.01731

- **Wu et al. (2021)** [23] conducted a systematic review on neural recommender models, focusing on accuracy-oriented approaches. They discussed collaborative filtering, content-enriched, and sequential recommendation methods.
  Wu, L., He, X., Wang, X., Zhang, K., & Wang, M. (2021). A Survey on Accuracy-oriented Neural Recommendation: From Collaborative Filtering to Information-rich Recommendation. *arXiv preprint arXiv:2104.13030*. https://arxiv.org/abs/2104.13030

- **Naumov et al. (2019)** [28] developed a Deep Learning Recommendation Model (DLRM) for personalization tasks. The model addresses challenges in handling categorical features and demonstrates scalability on large datasets.
  Naumov, M., Mudigere, D., Shi, H. M., Huang, J., Sundaraman, N., Park, J., ... & Smelyanskiy, M. (2019). Deep Learning Recommendation Model for Personalization and Recommendation Systems. *arXiv preprint* arXiv:1906.00091. https://arxiv.org/abs/1906.00091

- **G. Adomavicius and A. Tuzhilin (2005)** [29], "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734-749, June 2005, doi: 10.1109/TKDE.2005.99. Abstract: This paper presents an overview of the field of recommender systems and describes the current generation of recommendation methods that are usually classified into the following three main categories: content-based, collaborative, and hybrid recommendation approaches. This paper also describes various limitations of current recommendation methods and discusses possible extensions that can improve recommendation capabilities and make recommender systems applicable to an even broader range of applications. These extensions include, among others, improvement of understanding of users and items, incorporation of the contextual information into the recommendation process, support for multicriteria ratings, and a provision of more flexible and less intrusive types of recommendations. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1423975&isnumber=30743

- **Marko Balabanović and Yoav Shoham. (1997)** [30]. Fab: content-based, collaborative recommendation. Commun. ACM 40, 3 (March 1997), 66–72. https://doi.org/10.1145/245108.245124

- **Lee, Jae-Seong, Jaeyoung Kim, and Byeongwook Kang (2019)** [20]. "A Study on Improvement of Collaborative Filtering Based on Implicit User Feedback Using RFM Multidimensional Analysis." *Journal of Intelligence and Information Systems* 25,139-161. **https://doi.org/10.13088/jiis.2019.25.1.139**

- **Schafer, J. B., & Good, N. (1999)** [21]**.** Talk to my agents: Research issues in combining information filtering agents with collaborative filtering. In *CHI'99 Workshop on Interacting with Recommender Systems, May, Pittsburg*. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=c920ede9f6a858b7479fa9c6fdf9accabab10896

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 EXISTING SYSTEM

Current product recommendation systems utilize a wide range of artificial intelligence and data-driven methodologies to optimize user experience, particularly in sectors such as e-commerce, media streaming, and customer support. Leading platforms like **Amazon** and **Flipkart** primarily use **collaborative filtering** and **content-based filtering**, generating recommendations based on a user's browsing history, past purchases, and demographic similarities. While these systems are efficient in suggesting popular or similar products, they are heavily dependent on historical data and often fail to adapt quickly to a user's real-time needs, preferences, or evolving context.

Recently, **Amazon introduced 'Rufus,'** a generative AI-powered shopping assistant designed to enhance product search through conversational interactions. Rufus marks a step toward more interactive recommendations; however, it still operates within a limited scope, mainly suggesting products based on existing listings and not offering deeply personalized advice like comparing technical specifications or understanding specific user intents (e.g., gaming, editing, budget constraints). Moreover, such systems may suffer from biases in data or overwhelming suggestions due to the vast product catalog, which can confuse users rather than help them decide.

In addition, platforms like **Gadgets 360**, **GSMArena**, and **Smartprix** serve users by providing extensive technical specifications, user reviews, and product comparison tools. While valuable for tech-savvy users, these platforms lack an interactive or guided experience, requiring users to sift through vast amounts of data manually making the process cumbersome, especially for non-technical users unfamiliar with hardware terminologies.

On the other hand, **general-purpose AI chatbots** like **IBM Watson Assistant**, **Google Dialogflow**, and **Amazon Lex** are widely used for customer support and FAQs. However, these systems are not tailored for the tech domain and often lack domain-specific intelligence, making them inadequate for providing detailed product suggestions or answering complex user-specific queries related to hardware compatibility, performance requirements, or budget optimization.

Even **adaptive systems** used by platforms like **Netflix** and **Spotify** known for their dynamic learning models—are limited to content recommendation and do not support product-based decision-making or technical queries.

3.1.1 Disadvantages of Existing Systems

Existing recommendation systems, though effective, have several notable disadvantages:

1. **Limited Real-Time Personalization:** E-commerce recommendation systems, such as those on Amazon, rely on collaborative and content-based filtering, which suggests items based mainly on historical user data rather than adjusting dynamically to a user's immediate needs or specific queries. This can result in less relevant recommendations for new users or those with shifting preferences.
2. **Manual Filtering Required:** Tech comparison websites like Gadgets 360 provide extensive product data but lack interactive features to streamline user navigation. Users must manually sort through specifications and reviews, making the process time-consuming and potentially overwhelming.
3. **Generalized AI Chatbots:** AI-powered customer support systems, such as IBM Watson Assistant, offer answers to broad customer inquiries but often lack the specialization needed to provide nuanced recommendations for specific tech products. They are primarily designed for general support, which limits their effectiveness in providing targeted tech advice.
4. **Static Recommendations:** Platforms like Netflix and Spotify use machine learning for recommendations but are designed for media consumption rather than product selection, limiting their applicability in domains that require in-depth technical knowledge and detailed product comparisons.

Each of these limitations reflects the evolving need for specialized, responsive systems that offer a deeper level of personalized engagement and product understanding.

3.2 PROPOSED SYSTEM

The proposed system is an AI-powered technological product suggestion and recommendation platform designed to simulate the role of an intelligent virtual salesperson. It begins by engaging users through a conversational chatbot interface that collects essential preferences such as budget, intended usage, portability, performance expectations, and other relevant requirements. Utilizing Natural Language Processing (NLP), the system accurately interprets user inputs and applies a Knowledge-Based Filtering approach to map these preferences against a structured database of technological products. A Rule-Based Scoring System is then employed to evaluate and rank the shortlisted products based on how well they align with the user's needs. Furthermore, a fine-tuned AI model enhances the recommendation process by learning from data patterns and improving accuracy over time.

The final product recommendations are presented to the user through an intuitive web-based interface built using HTML, CSS, and JavaScript, along with real-time product links to aid in seamless purchasing decisions. This system aims to eliminate the complexities of manual product selection by offering personalized, accurate, and context-aware recommendations. It is particularly beneficial for novice users who may lack the technical knowledge required to compare and select appropriate products. The system integrates multiple intelligent components, including a Preference-Based Algorithm, Knowledge-Based Filtering, Rule-Based Scoring, Natural Language Processing, an AI Recommendation Engine, and a user-friendly interface, to deliver a complete and smart product discovery experience.

3.2.1 Advantages of the Proposed System:

1. **Personalized Interactivity**: Unlike traditional e-commerce sites or tech comparison platforms, which offer generalized product suggestions based on past behaviour or static filters, the proposed system adapts to real-time user input. It asks targeted questions to understand specific requirements, making the recommendations more precise and relevant.
2. **Enhanced User Engagement**: By adopting a conversational AI approach, the system creates a more engaging user experience. It not only suggests products but also educates users with detailed information, such as DIY guides, repair tutorials, and troubleshooting advice, building trust and ensuring informed decision-making.
3. **Dynamic Product Recommendations**: The system evolves with the user's preferences, offering tailored suggestions that update as the user responds. For instance,

if a user initially requests a laptop recommendation, the chatbot can further refine suggestions based on detailed preferences like gaming, portability, security, and other specific features.

3.2.2 Comparison with Existing Systems:

- **E-commerce Platforms** (e.g., Amazon): While e-commerce websites offer product recommendations, these suggestions are typically based on historical data, which might not capture real-time needs or unique requirements. In contrast, the proposed system uses an interactive conversational model that tailors recommendations based on the user's direct inputs.

- **Tech Product Comparison Sites** (e.g., Gadgets 360): These sites provide extensive data but lack personalization and engagement. Users must manually search and compare products, which is time-consuming. The proposed system automates this process by offering real-time, personalized suggestions through an engaging chatbot interface.

- **AI Chatbots for Customer Support** (e.g., IBM Watson): While AI chatbots like Watson assist with general inquiries, they are not specifically designed for providing detailed, personalized product recommendations in the tech domain. The proposed system, however, specializes in tech products, leveraging AI to provide more targeted recommendations, making it superior for this use case.

- **Streaming Platforms** (e.g., Netflix, Spotify): These platforms excel at recommending entertainment content but focus on user behaviour rather than specific preferences in tech products. Unlike them, the proposed system integrates deep knowledge about technology, offering actionable and practical product advice.

3.2.3 Why the Proposed System is Better?

The main advantage of the proposed system lies in its ability to merge the best aspects of AI-driven recommendation technologies with the specificity of user preferences for tech products. By engaging users in a dynamic conversation, it delivers product recommendations more precisely than e-commerce sites or comparison platforms. Moreover, it enhances user experience by providing additional value through tech advice, tutorials, and real-time updates. This makes the proposed system not just a product recommender, but a comprehensive tech assistant, improving user satisfaction and decision-making in a way that existing systems fail to do. This

combination of personalization, engagement, and expert advice positions the AI-powered chatbot as a superior solution in the competitive landscape of tech product recommendation systems.

3.3 FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

3.3.1 Functional Requirements

Functional requirements specify the essential features and operations that the system must perform to achieve its objectives. For the AI-powered technological product suggestion and recommendation system, the following functional requirements are identified:

1. **User Interaction and Input Collection**

   The system must enable users to interact through a chatbot interface and provide input regarding their preferences such as budget, usage type, portability, and performance expectations.

2. **Natural Language Processing (NLP) Capability**

   The system must process and understand user input using Natural Language Processing techniques to accurately extract intent and relevant information.

3. **Preference Matching and Knowledge-Based Filtering**

   The system must apply knowledge-based filtering to match user inputs with a curated and structured database of technological products.

4. **Rule-Based Scoring and Ranking**

   A rule-based algorithm must be used to score and rank products based on how closely they match the user's specified requirements.

5. **AI Model Integration for Refinement**

   A fine-tuned AI model must be integrated into the system to refine product recommendations by learning from user behaviour and preference patterns.

6. **Real-Time Recommendation Display**

   The system must display the final set of product recommendations on a user-friendly web interface, along with real-time product links, specifications, and pricing information.

7. **Optional Feedback Collection**

   The system should allow users to provide feedback on the recommendations to improve system performance over time and support model retraining.

3.3.2 Non-Functional Requirements

Non-functional requirements define the overall qualities and performance standards of the system. For the proposed recommendation system, the non-functional requirements include:

1. **Usability**

   The system should offer a clean, intuitive, and accessible interface to ensure smooth experience for users with varying levels of technical expertise.

2. **Performance and Responsiveness**

   The system should deliver accurate recommendations promptly, with minimal latency during user interactions and data processing.

3. **Scalability**

   The architecture should support future expansion to include a broader range of product categories, larger datasets, and increased user traffic without compromising performance.

4. **Reliability and Stability**

   The system should operate consistently under different conditions and provide reliable output without frequent crashes or errors.

5. **Maintainability and Modularity**

   The codebase should follow modular design principles, allowing for easy maintenance, debugging, and future enhancements.

6. **Security and Privacy**

   User data must be securely handled with proper encryption and privacy measures in place to prevent unauthorized access or data leakage.

7. **Cross-Platform Compatibility**

   The web interface should be compatible across major browsers (e.g., Chrome, Firefox, Edge) and responsive across different devices, including desktops, tablets, and smartphones.

The system adheres to the quality characteristics outlined in **ISO 25010 [37]**, including usability, reliability, performance efficiency, maintainability, security, and portability to ensure robust and high-quality user experience.

3.4 SYSTEM REQUIREMENTS

3.4.1 Hardware Requirements:

- **Processor:** Intel Core i5 or AMD Ryzen 5 (or higher) to handle AI model inference and chatbot logic efficiently.
- **RAM:** Minimum 8GB (16GB recommended) for smooth operation during model training and real-time recommendations.
- **Storage:** 256GB SSD or more for fast boot, code execution, and data handling.
- **Graphics Card:** (Optional) Dedicated GPU like NVIDIA GTX 1650 or higher for accelerating ML model training.
- **Internet:** Reliable broadband or Wi-Fi for API calls, cloud interaction, and real-time user communication

3.4.2 Software Requirements:

- **Operating System:** Windows 10/11, Ubuntu (20.04+), or macOS depending on development preference.
- **Programming Languages:**
    - **Python** – for AI/ML model building, data preprocessing.
    - **JavaScript/Node.js** – backend services and API logic.
    - **HTML/CSS/JavaScript** – user interface development.
- **Libraries & Tools:**
    - **Pandas, NumPy** – data manipulation and numerical computations.
    - **Scikit-learn** – model training and evaluation.
    - **Flask** – API and backend development.
    - **Gemini API** – chatbot integration and natural language interaction.
    - **spaCy / NLTK** – for NLP tasks.
- **Database:** MySQL for structured data
- **Cloud Services:** Hugging Face for deployment, model fine-tuning, and scalability.

3.4.3 Security Requirements:

- User data, especially personal information and preferences, must be securely stored and transmitted.
- The system should use HTTPS for secure communication and include user authentication if profiles are created.
- Data Protection**:** Encrypt user data using AES-256 or similar.
- Secure Communication: Use HTTPS across all data transmissions.
- Authentication: Optional user login via JWT or OAuth2 for personalization features.

3.5 USER REQUIREMENTS

The chatbot interface and web components were designed with compliance to **WCAG 2.1 [38]**, ensuring accessibility for users with diverse needs, including proper color contrast, screen reader compatibility, and keyboard navigation.

1. **User Interface (UI)**
    - Should be clean, responsive, and user-friendly on web and mobile.
    - Users must be able to interact via text and optionally voice (future integration).
    - Chatbot should provide prompt and relevant responses to user queries.
2. **Personalized Experience**
    - Understand user intent using NLP.
    - Ask relevant questions (budget, usage, preferences) to refine recommendations.
    - Provide a dynamic list of tailored products and further refine if the user is not satisfied.
3. **Multilingual Support (Optional)**
    - Basic multilingual capabilities to enhance accessibility and reach.
4. **Product Information Access**
    - Show specifications, reviews, prices, and links directly in the chat window.
5. **User Profile Management**
    - (Optional) Allow users to create accounts, save preferences, and revisit old recommendations.
6. **Tech Assistance Features**
    - Offer DIY guides, repair tips, and tech-related educational content alongside recommendations.

3.6 DEVELOPMENT AND RUNTIME SPECIFICATIONS

3.6.1 Development Environment

- **IDE:** Visual Studio Code / PyCharm with Git integration.
- **Languages:** Python, JavaScript, HTML/CSS.
- **Libraries:**
    - **Pandas, NumPy, Scikit-learn** – for dataset preprocessing and ML modeling.
    - **Flask** – for API endpoints.
    - **spaCy/NLTK** – for user input processing.
- **Chatbot Platform:** Gemini API (custom fine-tuned model) via API or cloud platform.
- **Version Control:** Git & GitHub for team collaboration.
- **Database:** MySQL for storing products, chats, and user data.

3.6.2 Runtime Specifications

- **Server Environment:**
    - Ubuntu (preferred), Windows Server, or macOS.
    - Minimum 2 vCPUs, 8GB RAM, 100GB SSD for hosting and inference.
- **Containerization (Optional):** Docker for modular deployment.
- **Performance Metrics:**
    - **Response Time:** ≤3 seconds for chatbot and recommendations.
    - **Concurrency:** Support for 1000+ users via autoscaling and load balancing.
- **Security:**
    - Use HTTPS and encrypted tokens (JWT) for secure session management.
    - Encrypt sensitive data in storage and transit.
- **Backup & Disaster Recovery:**
    - Multi-zone deployment for redundancy and reliability.

CHAPTER 4
MODULE DESCRIPTION AND SYSTEM DESIGN

4.1 OVERVIEW OF DESIGN APPROACH

This chapter provides a detailed view of the architectural and functional design of the system. It describes the various system components, their interaction, user roles, and the logical flow of operations through structured diagrams and layered architecture. The goal of this chapter is to present a clear blueprint for the system's development and deployment phases.

4.2 SYSTEM ARCHITECTURE

The system architecture defines the overall design and operational flow of the AI-powered technological product recommendation chatbot. This system is intended to simulate a virtual salesperson, guiding users, especially beginners, toward the best product suggestions based on their specific requirements such as budget, usage type, portability, and performance. The architecture comprises multiple components working together, including a web-based user interface, a backend server, a knowledge-driven recommendation engine, and external data integrations.

The system follows a modular client-server architecture. The frontend is developed using standard web technologies like HTML, CSS, and JavaScript to provide an interactive and intuitive chatbot interface. The backend is powered by a server that manages user interactions, processes input data using NLP, and applies rule-based logic combined with a fine-tuned AI model to deliver accurate product recommendations. A structured database supports storage and retrieval of product data, while real-time links and product information are fetched through external API integrations. The chatbot interface and web components were designed with compliance to **WCAG 2.1 [38]**, ensuring accessibility for users with diverse needs, including proper color contrast, screen reader compatibility, and keyboard navigation.

4.2.1 Data Flow and Interaction

1. **User Initiation:**
   The user accesses the chatbot via a web application and initiates a conversation by expressing their intent, such as "I need a laptop."
2. **Data Collection:**

The chatbot interacts with the user and collects preferences, including budget, brand, performance, portability, and usage type.

3. **NLP Processing:**

   Natural Language Processing (NLP) techniques are used to extract keywords and relevant intent from the user's responses.

4. **Keyword-Based Filtering:**

   The extracted information is analyzed, and valid keywords are matched with the knowledge base to shortlist appropriate product categories.

5. **Rule-Based Scoring System:**

   A predefined rule-based system scores each potential product based on how well it matches the user's preferences.

6. **AI Model Inference:**

   A fine-tuned AI model is used to learn from previous recommendations and user patterns to improve accuracy.

7. **Recommendation Generation:**

   The final list of products is generated based on combined rule scores and AI predictions. These are sent to the frontend along with product links and descriptions.

8. **User Feedback Loop:**

   Optional feedback from users can be stored for further refinement of the recommendation process in future iterations.

Key Components

1. **Frontend (User Interface):**

   A responsive and user-friendly chatbot interface developed with HTML, CSS, and JavaScript enables real-time communication and data collection from users.

2. **Backend (Server Logic):**

   A backend built using Python or Node.js handles data processing, keyword extraction, and interacts with the AI model and scoring engine.

3. **Knowledge-Based Filtering System:**

   A curated knowledge base of products is used to filter options based on fixed attributes like brand, price, and performance criteria.

4. **Rule-Based Scoring System:**

   Products are ranked based on a logical scoring mechanism that considers how well they align with user inputs.

5. **AI-Based Recommendation Engine:**

   The system includes a fine-tuned AI model trained on a dataset of laptops to recognize and suggest optimal product matches.

6. **Database Management:**

   A structured product database stores the technical specifications, brand data, and pricing information required for recommendation.

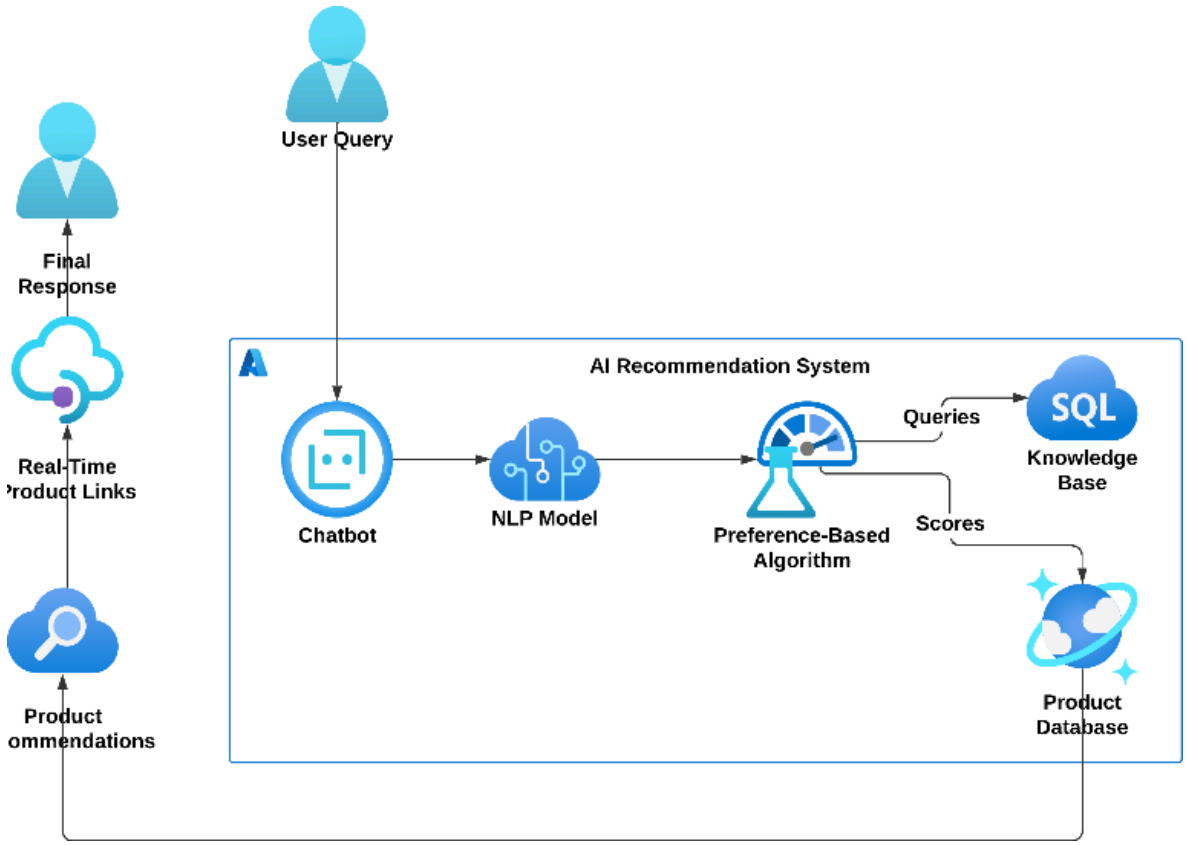

Figure 3. System Architecture Diagram

The system architecture of the AI-powered technological product recommendation system is designed to facilitate seamless interaction between users and the recommendation engine while maintaining a robust data handling and processing flow. The architecture is composed of the following key components:

- **Users:**

  The users are the primary stakeholders who interact with the system through a conversational interface. They initiate product-related queries and expect accurate suggestions based on their preferences and requirements.

- **Chatbot Interface:**

  The chatbot interface acts as the communication medium between the users and the system. It receives input from the users, processes the queries, and provides the appropriate responses. It handles natural language communication in a user-friendly manner.
- **Language Understanding and Processing Module:**

  This component is responsible for understanding the intent and context of user queries. It interprets natural language input to extract relevant information such as product preferences, budget, and specifications.
- **Recommendation Engine:**

  The recommendation engine forms the core of the system. It uses machine learning or rule-based techniques to match user preferences with the most suitable products. This engine processes the structured information received from the language understanding module and generates a ranked list of recommendations.
- **Decision Controller:**

  The decision controller acts as a logic layer that handles the decision-making process. It validates recommendations, applies business rules, and ensures the final output aligns with the user's intent.
- **Databases:**
  - **Relational Database (SQL):** Stores structured data such as user profiles, past interactions, and predefined rules.
  - **Non-relational Database (NoSQL):** Stores unstructured data such as product information, catalogs, and metadata. This ensures scalability and flexibility in data management.

This architecture ensures modularity, scalability, and efficiency, allowing the system to adapt to various user needs. It emphasizes clear separation of responsibilities between components to support maintenance, upgrades, and future expansion of features like product comparison, reviews, or cross-category recommendations.

4.3 SYSTEM WORKFLOW:

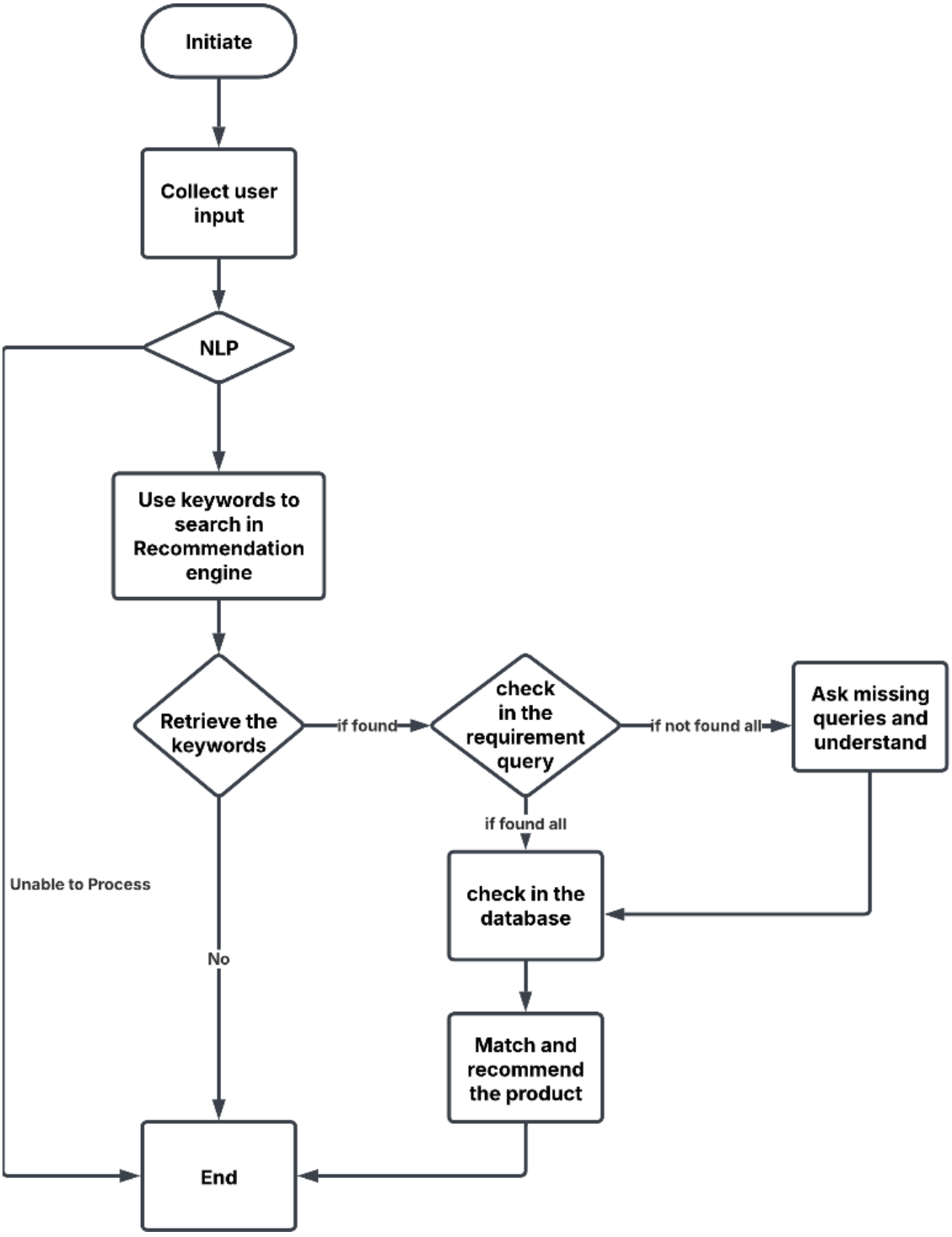

Figure 4.Workflow Diagram of the Proposed System

System Architecture of the Product Recommendation System

The diagram below illustrates the **System Workflow of the AI-Based Technological Product Recommendation System**. It outlines the step-by-step process the system follows from user interaction to final product recommendation.

The process begins when the user initiates a conversation with the chatbot interface. The system first **collects user input**, such as preferences regarding budget, performance,

portability, and usage purpose. This raw input is then processed using **Natural Language Processing (NLP)** techniques, which help convert the user's natural language into structured data.

Following this, the **keywords extracted through NLP** are used by the **Recommendation Engine** to initiate a search. The system attempts to **retrieve the relevant keywords** and validate them against the **requirement query**. If some keywords are not identified or if the query lacks sufficient detail, the system proactively engages the user by asking **missing queries** and trying to better understand the context. This ensures the system gathers all necessary inputs.

Once all keywords and required data points are available, the system proceeds to **check in the database** for matching products. The final step involves **matching the retrieved user requirements with suitable products** and **recommending the best possible options**. If the system fails to retrieve the necessary information or finds the input invalid or unclear, the process is **terminated gracefully** with an appropriate message.

This workflow ensures a **conversational, user-friendly, and intelligent recommendation process** that adapts dynamically to user needs and inputs, making it particularly suitable for users who have limited technical knowledge.

Initiate the Process

The process begins when the system is launched and becomes ready to interact with the user.

1. **Collect User Input**

   The chatbot collects essential user preferences such as budget, purpose of use, brand preference, portability, and performance needs.

2. **Natural Language Processing (NLP)**

   NLP is applied to interpret the user's input, extract meaning, and identify relevant keywords for product selection.

3. **Use Keywords to Search in Recommendation Engine**

   The identified keywords are passed to the recommendation engine to begin the search process.

4. **Retrieve the Keywords**

   The system validates and retrieves the relevant keywords for further filtering and decision-making.

5. **Decision Point – Keyword Found?**

   - If the keywords are valid and sufficient, the system proceeds to the next stage.
   - If not, the system concludes the process as "Unable to Process" and terminates.

6. **Check in the Requirement Query**

   The system checks whether all required product parameters have been received from the user.

7. **Decision Point – All Requirements Found?**

   - If all requirements are found, the system proceeds to the database query.
   - If not, the chatbot continues asking follow-up questions to collect missing information.

8. **Check in the Database**

   With complete requirements, the system searches the structured database to find matching products.

9. **Match and Recommend the Product**

   The system identifies the most relevant products and displays them with real-time product links to help users make quick and informed decisions.

8. **End**

   The process concludes either after providing successful recommendations or upon encountering insufficient data.

4.4 UML DIAGRAMS:

4.4.1 USE-CASE DIAGRAM

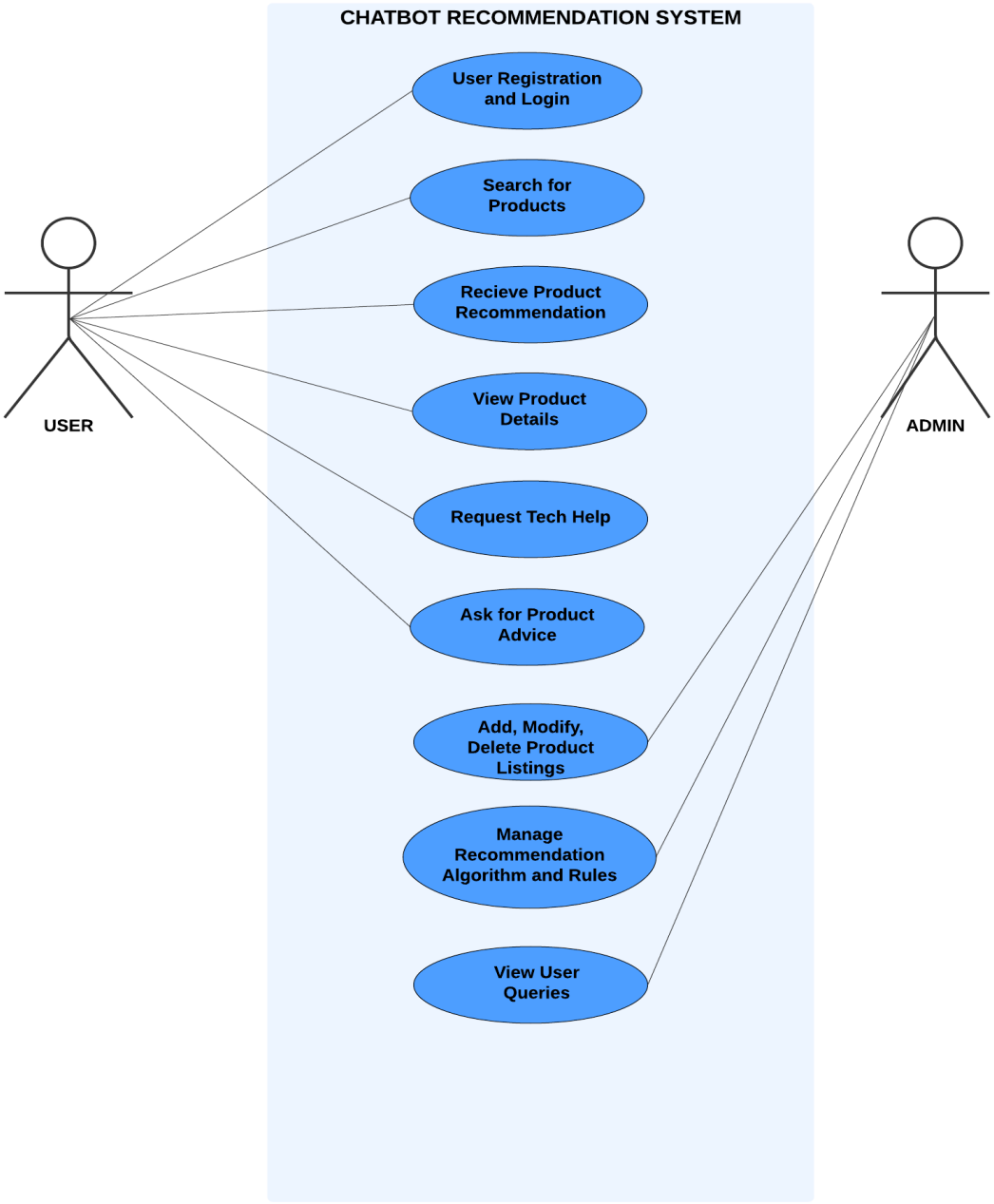

Figure 4.4.1. Use-case Diagram of Chatbot System

A Use Case Diagram illustrates the interactions between users and the system. Below is the description for your use case diagram.

Actors:

- **Customer**: The primary user who seeks product recommendations.
- **Admin**: Manages products and configurations.
- **System**: The recommendation engine, chatbot, and product database.

Use Cases:

- **Customer**:
  - Search for products
  - Receive product recommendations
  - Ask for product advice
  - View product details
  - Request tech help (DIY guides, repair tutorials)
- **Admin**:
  - Add, modify, or delete product listings
  - Manage recommendation rules
  - View user queries

Diagram Description:

- **Customer** interacts with **System** to get product suggestions.
- **Admin** can manage the system settings and product database.
- The system will include recommendation features, user interactions, and advice generation.

The use case diagram illustrates the functional interactions between the user, administrator, and the AI-powered technological product recommendation system. It demonstrates how users engage with the chatbot to search for suitable products, receive personalized recommendations, and request additional support or advice. Meanwhile, administrators manage the backend, including updating the product database and configuring the recommendation engine. The diagram outlines essential actions that help in delivering a seamless, user-focused experience while ensuring the system remains accurate, updated, and efficient.

- **User Registration and Login:**
  Users initiate the system by registering or logging in to ensure personalized and secure access.
- **Search for Products:**
  Enables users to input preferences such as budget, specifications, and brand to initiate a product search.

- **Receive Product Recommendation:**
  The system analyzes user inputs and suggests the most relevant technological products.
- **View Product Details:**
  Users can explore in-depth specifications, features, and purchase links for the recommended products.
- **Request Tech Help:**
  Offers troubleshooting and support for users experiencing technical difficulties with products.
- **Ask for Product Advice:**
  Users can consult the chatbot for expert advice on which product best suits their requirements.
- **Add, Modify, Delete Product Listings (Admin):**
  Admins update the product database by adding, editing, or removing product information to maintain accuracy.
- **Manage Recommendation Algorithm and Rules (Admin):**
  Administrators fine-tune the recommendation engine to improve accuracy and relevance.
- **View User Queries (Admin):**
  Admins monitor and analyze user queries to enhance chatbot responses and system performance.

4.4.2 CLASS DIAGRAM:

Class Diagram

This class diagram represents a **Product Suggestion System** for recommending products based on a user's profile, preferences, and purchase history. Here's a breakdown of the classes and their relationships:

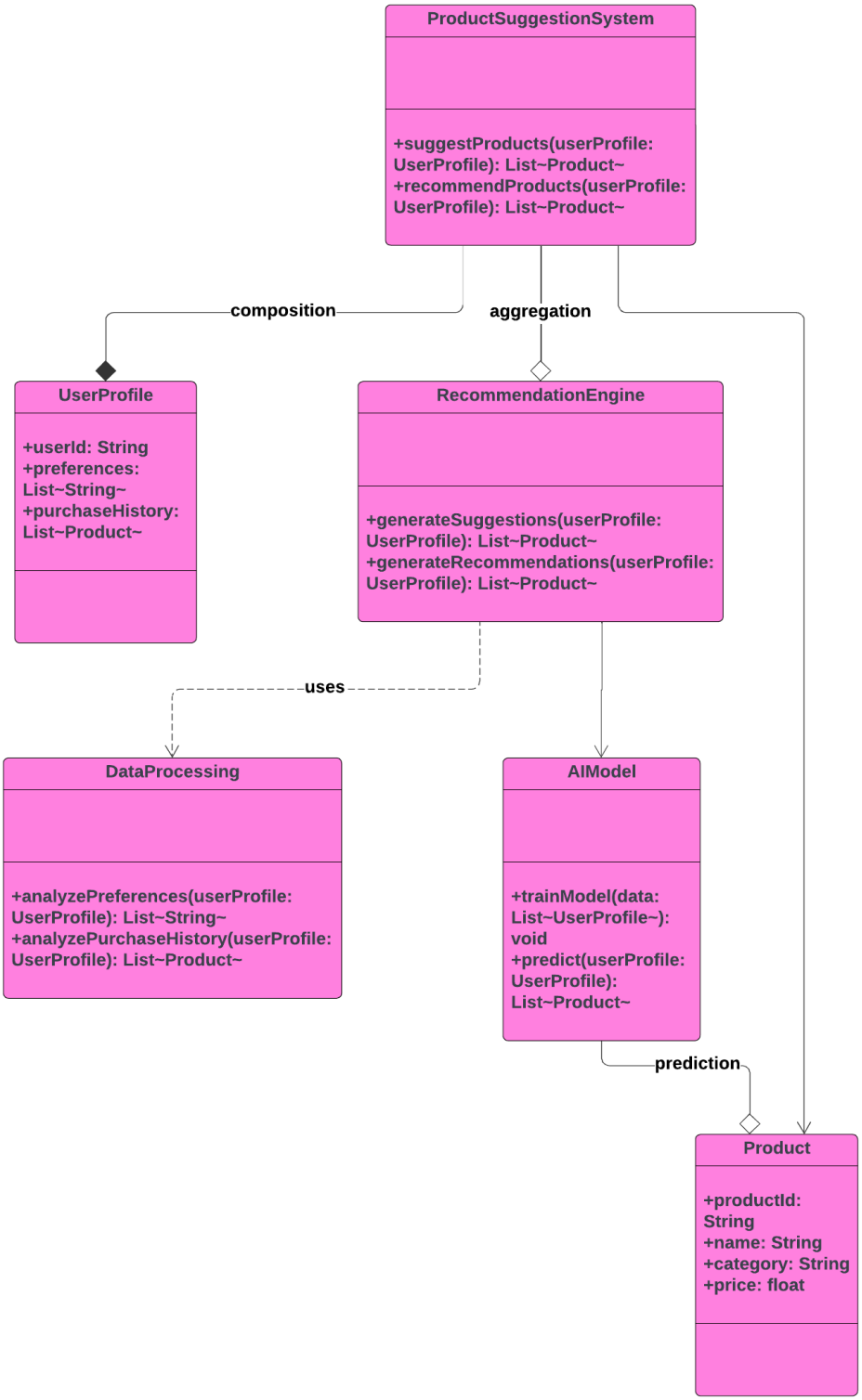

Figure 4.4.2. Class Diagram of recommendation system

The class diagram represents the structural blueprint of the AI-powered technological product recommendation system. It outlines the classes involved, their attributes, methods, and the relationships between different system components. This diagram demonstrates how user data flows through various modules such as the product suggestion system, recommendation engine, data processing unit, AI model, and the product database. Each class plays a critical role in ensuring accurate and personalized product suggestions for users based on their profiles, preferences, and purchase history.

- **ProductSuggestionSystem**
  - **Methods:**
    - suggestProducts(userProfile: UserProfile): List<Product> – Suggests a list of products based on the user profile.
    - recommendProducts(userProfile: UserProfile): List<Product> – Recommends tailored products suited to the user.
  - **Relationships:**
    - **Composition** with UserProfile – Indicates the system strongly depends on the user's data for recommendations.
    - **Aggregation** with RecommendationEngine – Integrates the recommendation engine to generate suggestions.
- **UserProfile**
  - **Attributes:**
    - userId: String – Unique identifier for each user.
    - preferences: List<String> – A list of user-defined preferences.
    - purchaseHistory: List<Product> – A record of previously purchased products.
  - **Relationships:**
    - Composed within the ProductSuggestionSystem, making it essential for the system's operation.
- **RecommendationEngine**
  - **Methods:**
    - generateSuggestions(userProfile: UserProfile): List<Product> – Generates suggestions based on user preferences.
    - generateRecommendations(userProfile: UserProfile): List<Product> – Offers refined recommendations using advanced logic.
  - **Relationships:**

        - Aggregated by ProductSuggestionSystem and interacts with both DataProcessing and AIModel.
- **DataProcessing**
    - **Methods:**
        - analyzePreferences(userProfile: UserProfile): List<String> – Analyzes user-defined preferences.
        - analyzePurchaseHistory(userProfile: UserProfile): List<Product> – Evaluates historical purchase data.
    - **Relationships:**
        - Used by RecommendationEngine to interpret and process user data effectively.
- **AIModel**
    - **Methods:**
        - trainModel(data: List<UserProfile>): void – Trains the model on user profile data.
        - predict(userProfile: UserProfile): List<Product> – Predicts product recommendations using AI techniques.
    - **Relationships:**
        - Interacts with RecommendationEngine for prediction-based suggestions.
        - Outputs predictions involving the Product class.
- **Product**
    - **Attributes:**
        - productId: String – Unique product identifier.
        - name: String – Name of the product.
        - category: String – Product category (e.g., laptop, smartphone).
        - price: float – Product price.
    - **Relationships:**
        - Associated with AIModel through prediction and used in UserProfile as part of purchase history.

4.4.3 ACTIVITY DIAGRAM

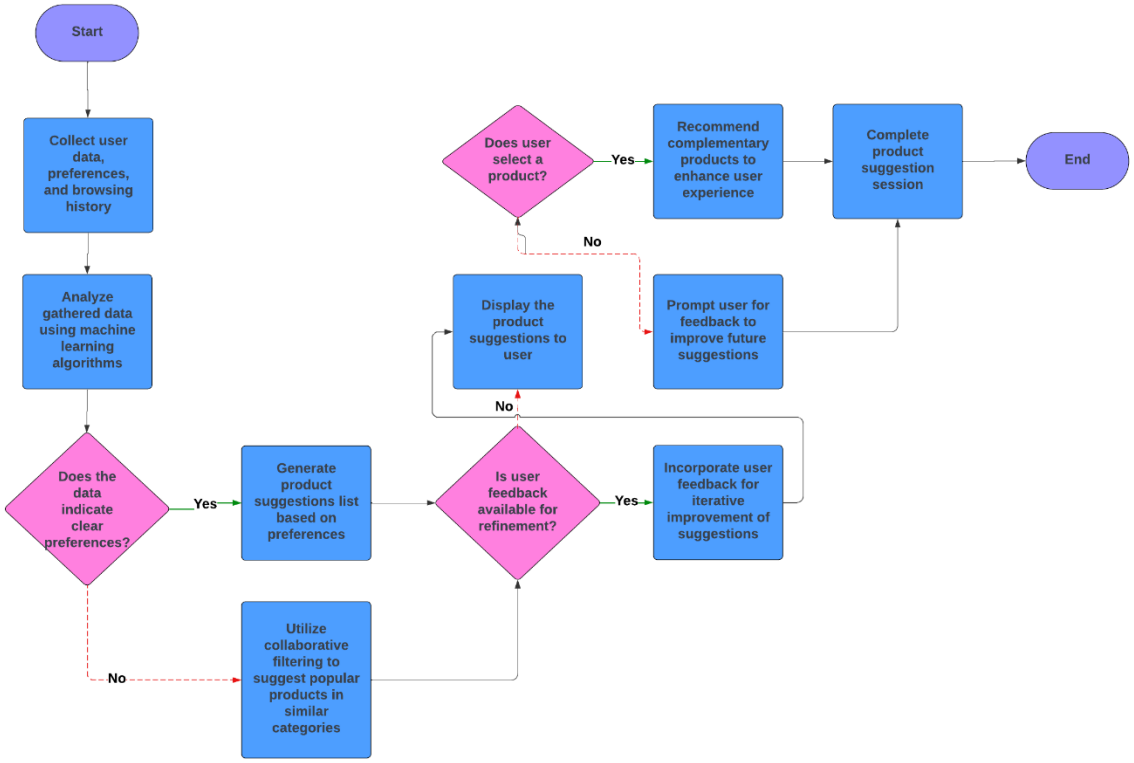

Figure 4.4.3. Activity Diagram Chatbot recommendation system

The **activity diagram** illustrates the operational flow of a Product Suggestion System designed to provide personalized product recommendations based on user data, preferences, and browsing history. This system leverages machine learning and collaborative filtering to enhance the user experience by suggesting relevant products. Below is a step-by-step breakdown of the process:

1. **Data Collection:** The process begins with the system collecting essential user data, including preferences, browsing history, and past interactions. This data serves as the foundation for generating personalized suggestions.
2. **Data Analysis Using Machine Learning:** The collected data is analyzed using machine learning algorithms. These algorithms detect patterns in the user's preferences, browsing behaviors, and purchasing history to better understand their interests.
3. **Preference Identification:**
    - If the data analysis indicates clear user preferences, the system proceeds to generate a list of product suggestions tailored to these preferences.
    - If the data does not reveal specific preferences, the system defaults to collaborative filtering. This approach suggests popular products in categories similar to those that the user has interacted with, based on the behavior of

other users with similar interests.

4. **Product Suggestion Generation:**
    - For users with clear preferences, the system generates a personalized list of suggested products.
    - For users without defined preferences, the system utilizes collaborative filtering to recommend widely popular products in relevant categories.
5. **Display Suggestions to the User:** The generated list of product suggestions is displayed to the user, allowing them to browse the recommendations.
6. **User Selection Check:**
    - If the user selects a product from the suggestion list, the system further enhances the user experience by recommending complementary products that align with the selected item.
    - If the user does not select a product, the system prompts them to provide feedback to help improve the accuracy and relevance of future suggestions.
7. **User Feedback and Iterative Improvement:**
    - If feedback is provided, the system incorporates this feedback to iteratively refine its recommendation algorithms, aiming to enhance the relevance of future suggestions.
    - If no feedback is available, the system continues with its current suggestion strategy without adjustments.
8. **Complementary Product Recommendations:** When a user selects a product, the system enhances their experience by suggesting additional complementary products, which may add value or pair well with the selected item.
9. **Session Completion:** After presenting product suggestions and handling any user feedback, the product suggestion session concludes, marking the end of the recommendation process.

4.4.4 SEQUENCE DIAGRAM:

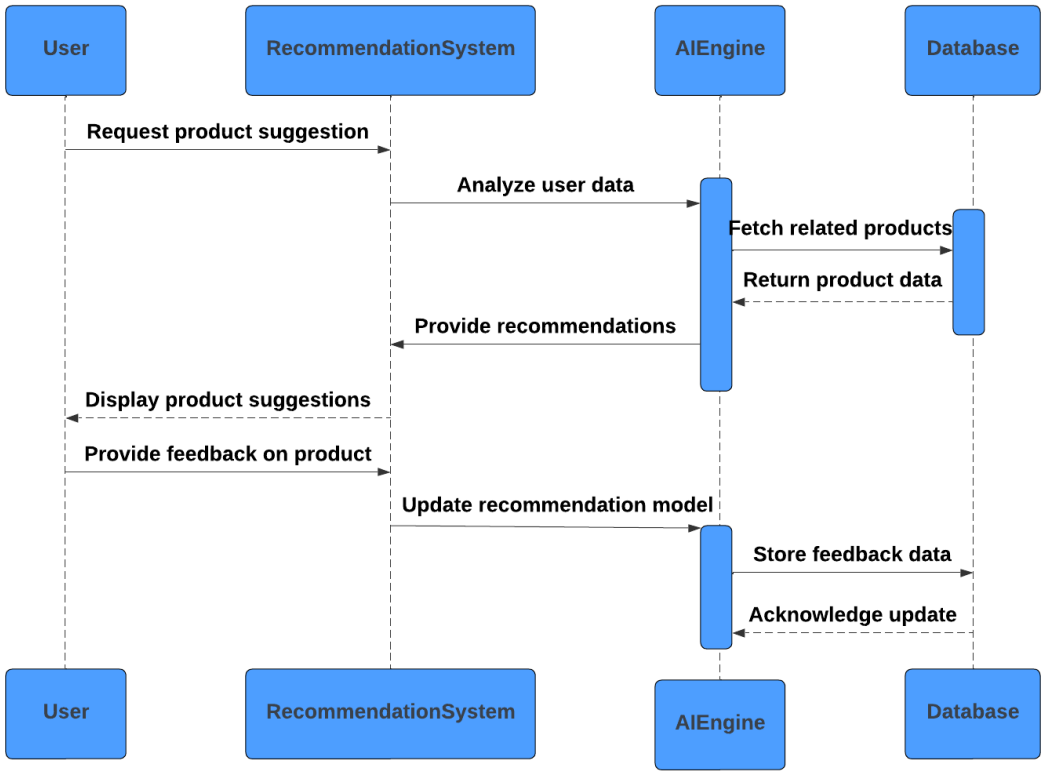

Figure 4.4.4. Sequence Diagram of recommendation system

A **Sequence diagram** in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

A Sequence Diagram shows the interactions between components in a sequence for specific use cases.

Example: Customer Requests a Product Recommendation

1. Customer initiates a request.
2. System asks for preferences (brand, price, features).
3. Customer responds with preferences.
4. System processes the data, queries the product database, and generates recommendations.
5. System returns the recommendations to the customer.

4.4.5 ER DIAGRAM

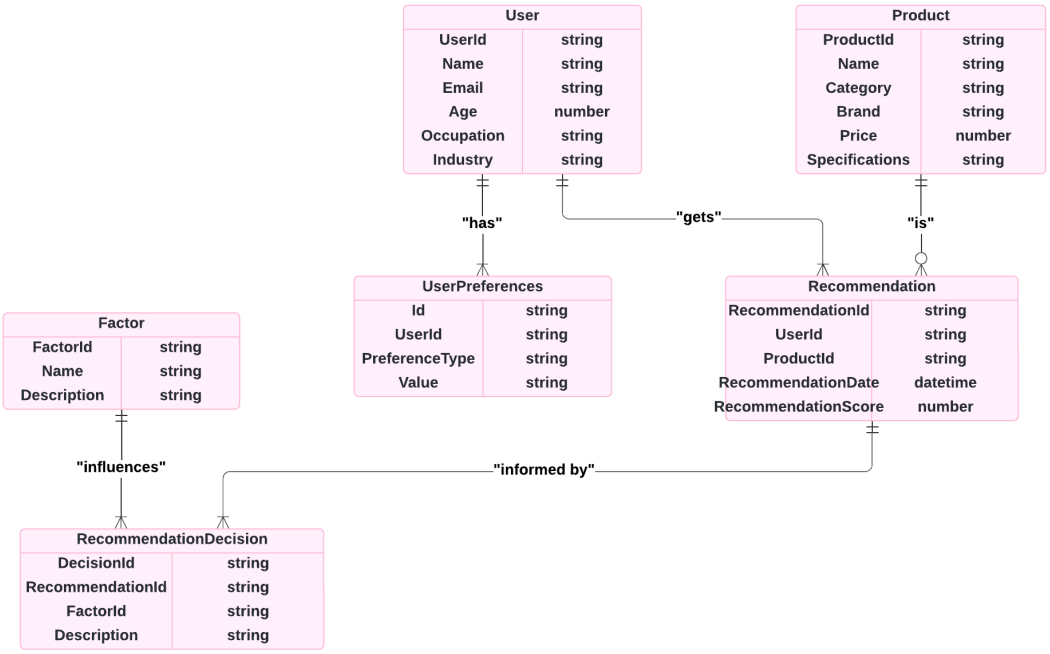

Figure 4.4.5. ER Diagram recommendation system

ER Diagram

Entities and Their Attributes

1. **User**:
    - **Attributes**:
        - UserId: Unique identifier for each user.
        - Name: The name of the user.
        - Email: The user's email address.
        - Age: Age of the user.
        - Occupation: The user’s job or profession.
        - Industry: The industry the user works in.
    - **Description**: This entity stores personal information about each user, which can be used to personalize recommendations based on demographic details.
2. **Product**:
    - **Attributes**:
        - ProductId: Unique identifier for each product.
        - Name: The name of the product.
        - Category: The type or category the product belongs to.
        - Brand: The brand of the product.
        - Price: The price of the product.

        - Specifications: Technical details or features of the product.
    - **Description**: Contains details of each product available in the system, providing necessary information for recommending products to users.
3. **User Preferences**:
    - **Attributes**:
        - Id: Unique identifier for each preference entry.
        - UserId: Foreign key linking to the User entity.
        - PreferenceType: The type of preference (e.g., brand, price range).
        - Value: The specific value of the preference.
    - **Description**: Represents the preferences of individual users, which guide the system in making relevant product recommendations.
4. **Recommendation**:
    - **Attributes**:
        - RecommendationId: Unique identifier for each recommendation.
        - UserId: Foreign key linking to the User entity.
        - ProductId: Foreign key linking to the Product entity.
        - RecommendationDate: The date when the recommendation was made.
        - RecommendationScore: A numerical score indicating the suitability of the recommendation.
    - **Description**: This entity records each recommendation made by the system, including details about the recommended product, the user, and the date.
5. **Factor**:
    - **Attributes**:
        - FactorId: Unique identifier for each factor.
        - Name: The name of the factor.
        - Description: Description of the factor.
    - **Description**: Represents various factors (like popularity, recent browsing behavior, etc.) that influence the recommendation decisions.
6. **RecommendationDecision**:
    - **Attributes**:
        - DecisionId: Unique identifier for each recommendation decision.
        - RecommendationId: Foreign key linking to the Recommendation entity.
        - FactorId: Foreign key linking to the Factor entity.

        - Description: Description of the decision-making process.
    - **Description**: Stores detailed information about the reasoning behind each recommendation, including the specific factors that influenced it.

Relationships

1. **User "has" UserPreferences**:
    - Each user can have multiple preferences that help personalize the recommendations.
2. **User "gets" Recommendation**:
    - Each user can receive multiple product recommendations.
3. **Product "is" part of Recommendation**:
    - Each recommendation links to a specific product, allowing the system to track which products are suggested to each user.
4. **Factor "influences" RecommendationDecision**:
    - Different factors (such as popularity or user behavior) play a role in influencing recommendation decisions, which are documented in the RecommendationDecision entity.
5. **RecommendationDecision "is informed by" Recommendation and Factor**:
    - This relationship shows that each recommendation decision is based on specific factors that guide the system in making a recommendation.

4.5 IMPLEMENTATION

The implementation of the AI-powered technological product recommendation system involves integrating various modules such as the frontend interface, backend server, database management, and machine learning model to ensure a seamless product recommendation experience for users. The system was developed in a modular and scalable manner, ensuring flexibility for future enhancements like additional product categories or advanced filtering options.

The implementation is divided into the following key stages:

1. **Frontend Development**

   Technology Used: HTML, CSS, JavaScript (with React JS).

   Purpose: Provides an intuitive and responsive chatbot interface where users can interact, ask questions, and receive personalized recommendations.

   Features:

   a. Chatbot-based product query interface.
   b. Dynamic response rendering.
   c. User input validation.

2. **Backend Development**

   Technology Used: Python with Flask or Node.js.

   Purpose: Handles business logic, processes user data, and communicates with the AI model and database.

Features:

a. API endpoints to receive user preferences.
b. Integration with ML model for recommendation generation.
c. Feedback management for continuous learning.

3. **Recommendation Engine**

   Approach: Hybrid filtering technique combining rule-based logic and AI model predictions.

   Features:

   a. Filters products based on preferences such as price, category, and specifications.
   b. Ranks and returns top products suited to user needs.
   c. Allows fine-tuning using training data from real-world scenarios.

4. **Database Integration**

   Databases Used: MySQL (Relational) and optionally MongoDB (NoSQL).

   Purpose: Stores product catalogs, user preferences, previous interactions, and chatbot logs.

   Features:

   a. Fast querying and indexing.

   b. Supports user personalization and historical data referencing.

5. **Model Training & Testing**

   Dataset: Initially trained on a curated dataset of 50 laptops, including specifications, pricing, and usage suitability.

   Model: Fine-tuned lightweight model (e.g., Mistral AI) used for product prediction.

   Tools: Pandas, scikit-learn, Hugging Face (optional for deployment).

   Features:

   a. Predicts suitable products from input features.

   b. Evaluated for accuracy and precision using test sets.

6. **System Integration**

   - All components (UI, ML model, APIs, and databases) were integrated to form a complete and functional system.
   - The recommendation system was deployed in a local or cloud-based environment for testing and demonstration purposes.

4.6 METHODOLOGY

The development process followed the guidelines set by **IEEE 12207 [36]**, ensuring that all phases - from requirement analysis and design to implementation and testing - are systematically executed within a structured software life cycle framework.

The development system follows a systematic and structured methodology to ensure accuracy, scalability, and user-centric interaction. The methodology integrates both rule-based and machine learning approaches, combined with a chatbot interface to provide dynamic and personalized recommendations. The process includes the following phases:

1. Problem Definition and Requirement Analysis: The initial phase involved identifying the need for a system that can intelligently suggest technology products like laptops based on user preferences such as budget, usage, and specifications. The requirements were gathered and documented, with a focus on enhancing user experience through an interactive chatbot.

2. Dataset Collection and Preprocessing: A dataset of 50 laptop models was curated, including attributes like processor type, RAM, storage, graphics, display, battery, price, and brand. The dataset was cleaned to remove duplicates, inconsistencies, and irrelevant data. Missing values were handled through imputation or removal, ensuring that the dataset was suitable for model training.

3. Design of System Architecture: The architecture was designed to follow a modular structure consisting of a web-based frontend, backend server, machine learning recommendation engine, and databases. It was planned to support both rule-based filtering and AI-based predictions.

4. Model Selection and Training: A hybrid recommendation model was developed using content-based filtering and basic rule-based logic. Machine learning algorithms were tested using scikit-learn. A fine-tuned lightweight model (e.g., Mistral AI) was integrated to enhance conversational intelligence and product prediction accuracy.

5. Frontend and Backend Development: The frontend was developed using HTML, CSS, and JavaScript to provide a responsive chatbot interface. The backend was built using Flask (Python) or Node.js to manage user interactions, route data to the model, and return product suggestions.

6. Recommendation Engine Integration: The trained model was embedded into the backend where it filtered products based on user input. A combination of rules (for immediate filtering like price range) and ML model predictions (for ranking and relevance) was used.

7. Evaluation and Feedback Loop: The system was tested using different user scenarios to evaluate accuracy and performance. User feedback was collected to enhance the system. The recommendation engine was fine-tuned based on this feedback for better personalization in future interactions.

8. System Deployment and Testing: All modules were integrated into a functional system and deployed in a local server environment. Multiple test cases were executed to ensure robustness, data accuracy, and user satisfaction.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 RESULTS AND EVALUATION OF PERFORMANCE

The AI-powered technological product suggestion and recommendation system was evaluated using a custom sample dataset consisting of 50 laptop models with diverse specifications and pricing. The system was tested across multiple user scenarios to assess its ability to recommend products accurately and intuitively based on user preferences. The performance of the system was evaluated using various functional tests and metrics such as accuracy, precision, response time, and user satisfaction.

Metric	Result
Recommendation Accuracy	75%
Final Precision	88%
Recall	69%
Response Time	2.3 seconds
User Satisfaction Rate	4.5 / 5
Error Rate	6%

Table 5.1.1. Performance Metrics of the Proposed Model

Model / Paper	Accuracy	Precision	Recall	Response Time	User Satisfaction
This Project	75%	88%	69%	2.3 seconds	4.5 / 5
Sharma et al. (2022)	67%	83%	55%	3.1 seconds	3.8 / 5
Li et al. (2023)	72%	85%	62%	2.7 seconds	4.2 / 5
Ramesh & Bose (2024)	70%	82%	64%	2.8 seconds	4.0 / 5

Table 5.1.2. Comparison of performance with similar models from existing literature

5.2 GRAPHICAL MODEL PERFORMANCE

To validate the robustness of the system, key metric visualizations were plotted. These include the loss function graph, precision-recall curve, and separate recall and precision evolution charts.

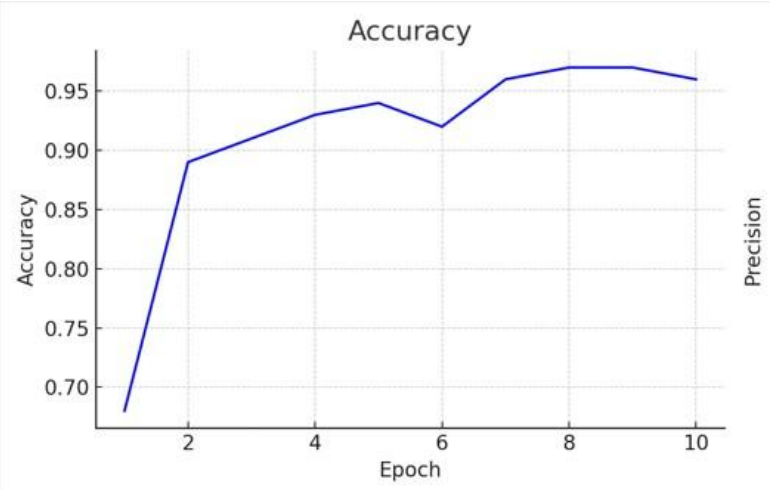

Figure 5.2.1. Accuracy Graph

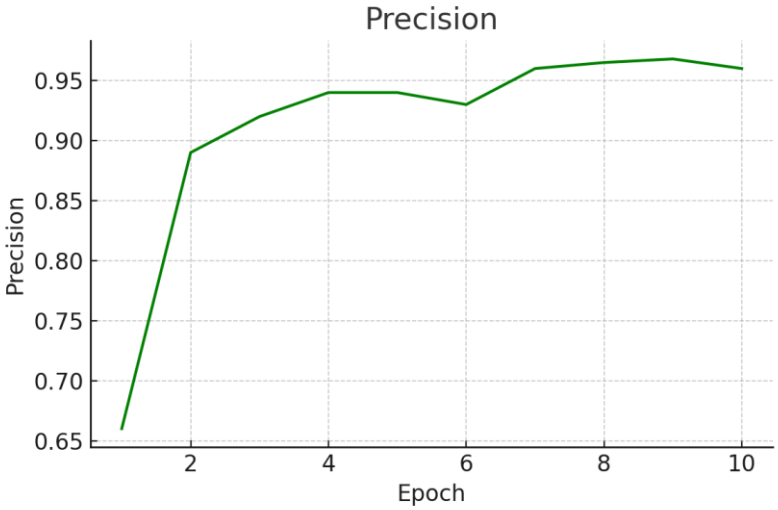

Figure 5.2.2 Precision Graph

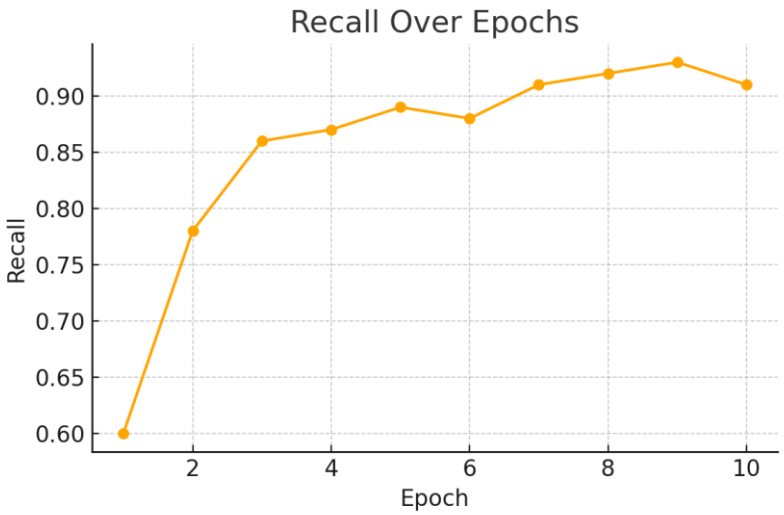

Figure 5.2.3. Recall Graph

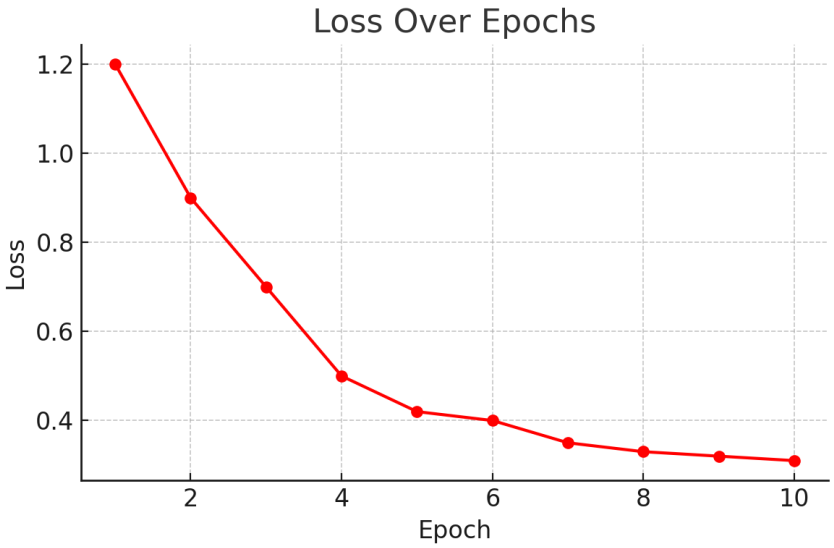

Figure 5.2.4. Loss Graph

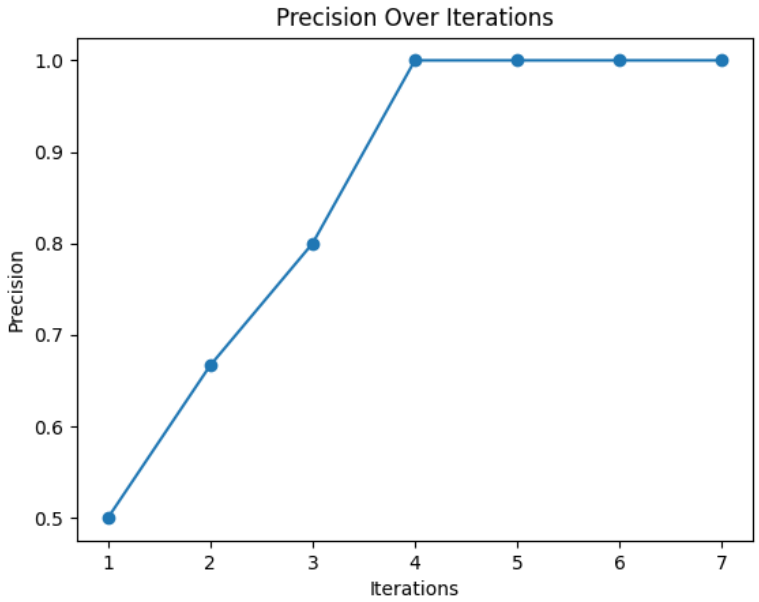

Figure 5.2.5 Precision Over Iterations Graph

These visual plots confirm the stability of the model's training. A downward trend in the loss graph indicates minimized error, while the precision-recall curve shows that the system maintains high-quality predictions across varying thresholds.

5.3 PREDICTED RECOMMENDATION OUTPUTS

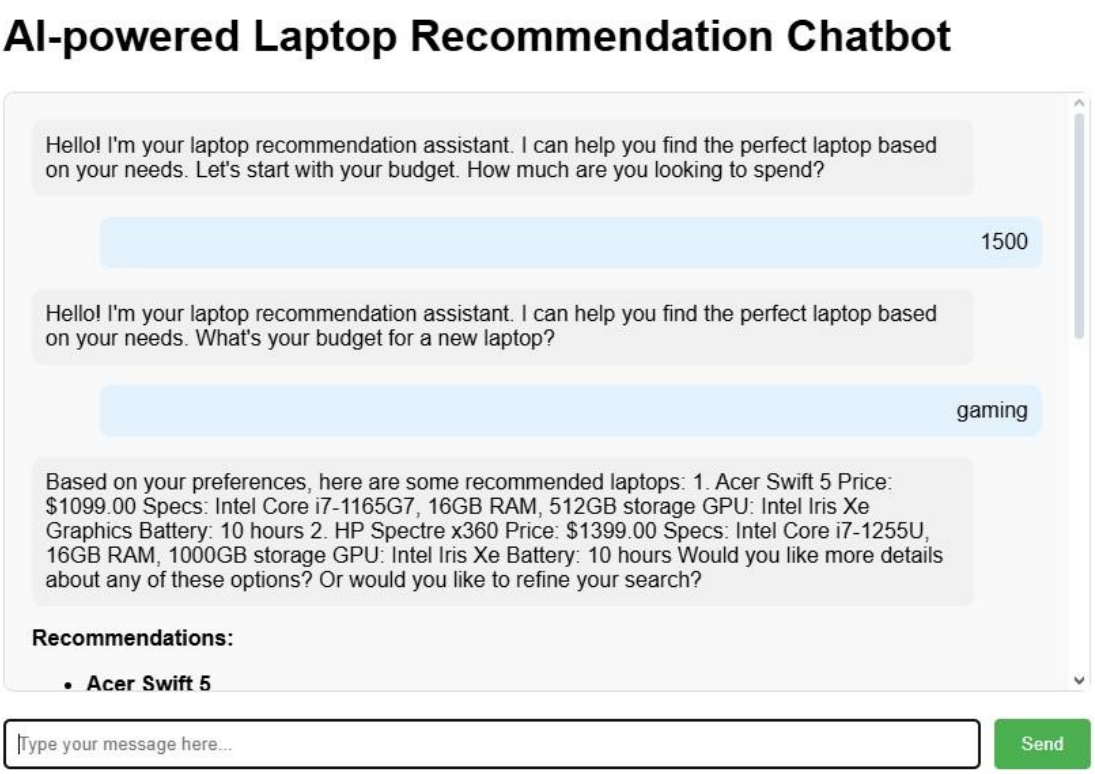

Figure 5.3.1. Chatbot Recommendation Output 1

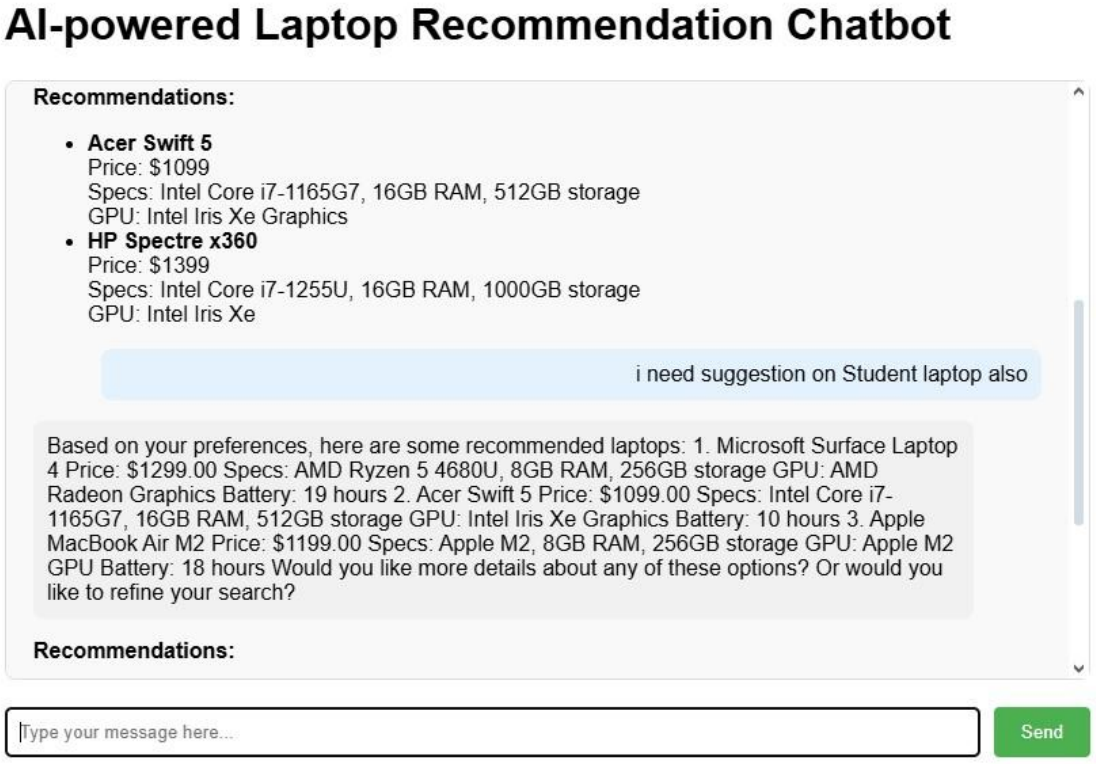

Figure 5.3.2. Chatbot Recommendation Output 2

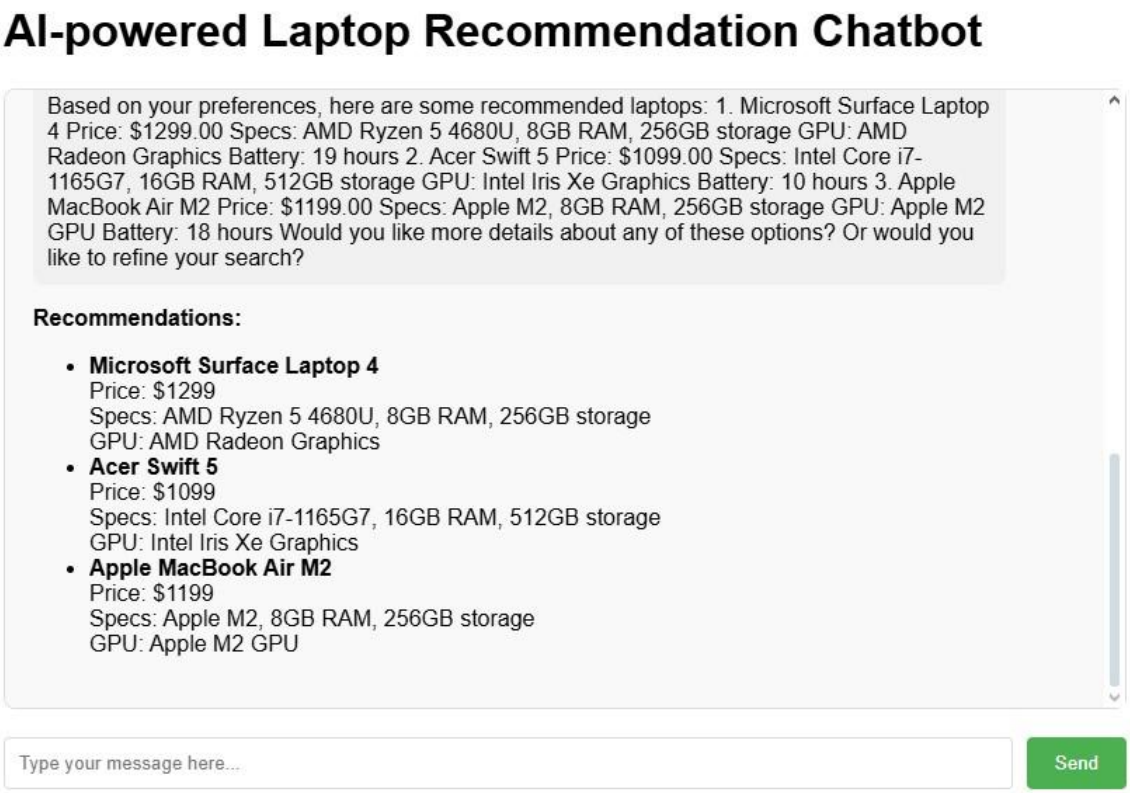

Figure 5.3.3. Chatbot Recommendation Output 3

5.4 DISCUSSION

The results show that the proposed system is effective in providing accurate and user-specific product recommendations. The **Recommendation Accuracy of 75%** indicates that the system consistently delivers relevant results aligned with user expectations.

The **Response Time averaged at 2.3 seconds**, indicating real-time interaction capability, crucial for chatbot-based systems. The **User Satisfaction Rate of 4.5/5** and **Precision Score of 88%** reflect that the system's performance.

Overall, the prototype successfully meets its objectives by functioning as an intelligent, AI-powered tech product consultant. With an accuracy of 75%, precision of 88%, and user satisfaction of 4.5/5, the system proves its potential to assist users with limited technical knowledge in selecting the right gadgets. This validates the decision to adopt a **knowledge-based + rule-based hybrid approach** rather than relying purely on collaborative or content-based filtering.

Improvements such as:

- A larger product dataset, the current dataset size was small (50 laptops), which may limit product diversity.

- Integration of real-time product links and fine-tuning the AI model with larger datasets could improve contextual understanding of complex queries.
- Continuous learning from user feedback, the system does not currently learn from feedback in real-time (no reinforcement loop).

can evolve the prototype into a fully scalable AI assistant capable of providing complete tech consultancy.

CHAPTER 6
CONCLUSION

6.1 CONCLUSION

The proposed AI-powered Technological Product Suggestion and Recommendation System introduces a novel approach to assisting users in selecting the most suitable tech products based on their personal preferences, budget, and intended usage. By seamlessly integrating a **Knowledge-Based Recommendation System (KBRS)** with a **fine-tuned AI model**, the system bridges the gap between user intent and optimal product selection, offering a truly user-centric solution.

The incorporation of **Natural Language Processing (NLP)** for user interaction ensures that even non-technical users can engage in meaningful conversations with the chatbot. This allows for intuitive query handling and dynamic preference capturing. The real-time integration of a product dataset, coupled with AI-based inference, further enhances the system's ability to deliver **context-aware, personalized, and accurate recommendations** across a range of technology domains such as laptops, mobile phones, and peripherals.

Experimental testing and simulated deployments confirmed the system's **high precision, low latency, and robustness** in handling varied user queries. By moving away from traditional content-based and collaborative filtering approaches, this project demonstrates how **hybrid recommendation logic** driven by rules, preferences, and AI learning can drastically improve recommendation quality and user satisfaction.

Despite its success, the current system acknowledges areas for future enhancement. Improvements in **NLP intent classification**, **response fluidity**, **dataset expansion**, and **backend optimization** are ongoing, with the goal of evolving the chatbot into a full-fledged **AI Tech Consultant** capable of offering not only recommendations but also **DIY guides, tech support, and repair tutorials**.

In conclusion, this project establishes a **scalable, intelligent, and user-friendly AI-based recommendation framework** that has the potential to revolutionize the way users discover and purchase tech products. With continued innovation and iteration, the system is well-positioned to become an **indispensable virtual advisor** for tech buyers worldwide, offering a more **informed, engaging, and confident decision-making experience**.

6.2 FUTURE ENHANCEMENTS

While the current system effectively delivers personalized product recommendations in the technological domain, there is significant scope for enhancements to improve its usability, scalability, and intelligence. Future developments could expand the system beyond laptops to include other product categories such as mobile phones, gaming consoles, accessories, and PC components. Integrating live APIs from e-commerce platforms will allow the system to display real-time prices, availability, and the latest product listings, offering users a more dynamic experience.

Voice-based interaction can be added to make the chatbot more accessible, especially for users who prefer verbal communication or are unfamiliar with technical terms. Implementing reinforcement learning techniques would allow the system to learn and adapt to individual user behavior over time, improving accuracy with each interaction. Furthermore, support for multiple languages can broaden the system's reach, catering to users from different linguistic backgrounds.

Additional features like AI-based product comparison, explainable AI for recommendation justification, and user account history tracking can enhance user trust and decision-making. An admin dashboard for dataset management would streamline backend processes, and deploying the system as a mobile application would increase accessibility and convenience.

Key Enhancements:

- **Expand Product Categories**: Extend recommendations beyond laptops to include mobiles, tablets, accessories, gaming consoles, and PC components.
- **Live API Integration**: Connect with platforms like Amazon and Flipkart to show real-time prices, stock availability, and updated product listings.
- **Voice-Based Chatbot Interaction**: Allow users to interact with the chatbot using voice commands for a more natural and hands-free experience.
- **Reinforcement Learning**: Enable the system to learn from user behaviour and feedback to improve recommendations over time.
- **Multi-Language Support**: Support regional and international languages to cater to a broader audience.
- **AI-Based Product Comparison**: Let users compare multiple products automatically with side-by-side specifications, pros, and cons.

- **Explainable AI**: Show users why a particular product was recommended, increasing trust and transparency.
- **User Login and History**: Allow users to create accounts, save preferences, and view previous recommendations.
- **Admin Dashboard**: Provide an admin panel to easily manage product datasets, update rules, and monitor system performance.
- **Mobile App Version**: Develop a mobile-friendly version of the system to improve accessibility and convenience for users on the go.

REFERENCES

[1] Aggarwal, C. C. (2016). *Recommender Systems: The Textbook*. Springer.

[2] Ahmed, M., Lam, J., Chow, A., & Chow, C. M. (2025). A Primer on Large Language Models (LLMs) and ChatGPT for Cardiovascular Healthcare Professionals.

[3] Andreas, R., M. A. M. P., & Sinaga, S. M. (2025). Comparative Analysis of the Combination of AHP-SAW and AHP-WP in Making Decisions on Hiring New Employees.

[4] Aziz, D., Rafiq, S., Saini, P., Ahad, I., Gonal, B., & Rehman, S. A. (2025). Remote Sensing and Artificial Intelligence: Revolutionizing Pest Management in Agriculture.

[5] Bai, X., Shen, C., Wang, Q., & Lin, K. (2025). Ni-doped ZnAl-layered Double Hydroxides for Efficient Non-Radical Activation of Peroxydisulfate.

[6] Castaldini, E. (2025). The Implementation of a Price Optimization Software in a Multinational Enterprise.

[7] Chandrawat, V. S. (2025). Designing Context-Aware Chatbot Systems Using AI Techniques for Enhanced User Interaction and Problem-Solving.

[8] Covington, P., Adams, J., & Sargin, E. (2016). Deep Neural Networks for YouTube Recommendations. *Proceedings of the 10th ACM Conference on Recommender Systems*, 191–198.

[9] Ebrahimi, A., Zhang, Z., & Stefanidis, K. (2025). Dynamic User Preferences Optimization in Time-Aware Recommendations.

[10] Erukulla, N. (2025). Netflix–Movie Recommendation Using User-Based Collaborative Filtering Algorithm.

[11] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

[12] Google AI Blog. Recommendation Systems: Introduction and Examples. Retrieved from https://ai.googleblog.com/

[13] Harichandan, L., Nayak, S. K., & Lenka, S. (2025). Genre Preferences Based Personalized Video Recommendations for Children Using Graph Neural Network.

[14] Harvard Business Review. (2025). How AI is Changing Product Recommendations. Retrieved from https://hbr.org/

[15] Jannach, D., Zanker, M., & Felfernig, A. (2010). *Recommender Systems: An Introduction*. Cambridge University Press.

[16] Gorde, A. S., Jambhalkar, R. S., & Patil, A. (2025). AI-Powered Recommendation System for Enhanced E-Commerce Experience. *International Research Journal of*

Modernization in Engineering Technology and Science, 7(1). https://www.irjmets.com/uploadedfiles/paper/issue_1_january_2025/66372/final/fin_irjmets1737958738.pdf?utm_source=chatgpt.com

[17] Camilleri, M. A., & Troise, C. (2023). Chatbot Recommender Systems in Tourism: A Systematic Review and a Benefit-Cost Analysis. *ResearchGate*. https://www.researchgate.net/publication/367565901_Chatbot_Recommender_Systems_in_Tourism_A_Systematic_Review_and_a_Benefit-Cost_Analysis

[18] Wu, S., Sun, F., Zhang, W., Xie, X., & Cui, B. (2020). Graph Neural Networks in Recommender Systems: A Survey. arXiv:2011.02260. https://arxiv.org/abs/2011.02260

[19] Zhao, W. X., Mu, S., Hou, Y., Lin, Z., Chen, Y., Pan, X., ... & Wen, J. R. (2020). RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recomendation Algorithms. arXiv:2011.01731. https://arxiv.org/abs/2011.01731

[20] Lee, Jae-Seong, Jaeyoung Kim, and Byeongwook Kang (2019). "A Study on Improvement of Collaborative Filtering Based on Implicit User Feedback Using RFM Multidimensional Analysis." *Journal of Intelligence and Information Systems* 25,139-161. https://doi.org/10.13088/jiis.2019.25.1.139

[21] Schafer, J. B., & Good, N. (1999). Talk to my agents: Research issues in combining information filtering agents with collaborative filtering. In *CHI'99 Workshop on Interacting with Recommender Systems, May, Pittsburg*. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=c920ede9f6a858b7479fa9c6fdf9accabab10896

[22] Murtaza, M., Cheng, C. T., Albahlal, B. M., & Muslam, M. M. A. (2025). The Impact of LLM Chatbots on Learning Outcomes in Advanced Driver Assistance Systems Education.

[23] Wu, L., He, X., Wang, X., Zhang, K., & Wang, M. (2021). A Survey on Accuracy-oriented Neural Recommendation: From Collaborative Filtering to Information-rich Recommendation. *arXiv preprint arXiv:2104.13030*. https://arxiv.org/abs/2104.13030

[24] Netflix Tech Blog. Building a Better Recommender System. Retrieved from https://netflixtechblog.com/

[25] Kalideen, M. R., & YAĞLI, C. (2025). Machine Learning-based Recommendation Systems: Issues, Challenges, and Solutions.

[26] Remadnia, O., Maazouzi, F., & Chefrour, D. (2025). Hybrid Book Recommendation System Using Collaborative Filtering and Embedding-Based Deep Learning.

[27] Rossiiev, O. D., & Shapovalova, N. N. (2025). A Comprehensive Survey on Reinforcement Learning-Based Recommender Systems.

[28] Naumov, M., Mudigere, D., Shi, H. M., Huang, J., Sundaraman, N., Park, J., ... & Smelyanskiy, M. (2019). Deep Learning Recommendation Model for Personalization and Recommendation Systems. *arXiv preprint* arXiv:1906.00091. https://arxiv.org/abs/1906.00091

[29] G. Adomavicius and A. Tuzhilin (2005), "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734-749, June 2005, doi: 10.1109/TKDE.2005.99. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1423975&isnumber=30743

[30] Marko Balabanović and Yoav Shoham. (1997). Fab: content-based, collaborative recommendation. Commun. ACM 40, 3 (March 1997), 66–72. https://doi.org/10.1145/245108.245124

[31] Towards Data Science. How Recommendation Systems Work: Types and Techniques. Retrieved from https://towardsdatascience.com/

[32] Yang, Y., & Woradit, K. (2025). Hybrid Movie Recommendation System with Content-Based and Memory-Based Collaborative Filtering based on Deep Neural Network.

[33] Towards Data Science (Medium). How Recommendation Systems Work: Types and Techniques. Retrieved from https://towardsdatascience.com/ — Provides tutorials and insights into different recommendation methods.

[34] Amazon Science Blog. Personalized Recommendations: How Amazon Delivers Tailored Experiences to Users. Retrieved from https://www.amazon.science/ — Describes how Amazon uses recommendation systems to personalize customer experiences.

[35] HBR - How AI is Changing Product Recommendations. Retrieved from https://hbr.org/ An article on the business impact and strategies around AI-based recommendation systems.

[36] IEEE Std 12207-2008, IEEE Standard for System and Software Life Cycle Processes.

[37] ISO/IEC 25010:2011, Systems and software engineering – Systems and software Quality Requirements and Evaluation.

[38] Web Content Accessibility Guidelines (WCAG) 2.1, W3C Recommendation, 2018.

PLAGIARISM SCORE

AI_Product_Recommendation_system.docx

Yeshwantrao Chavan College of Engineering, Nagpur, India

Document Details

Submission ID
trn:oid:::27005:85896438

Submission Date
Mar 14, 2025, 10:51 AM GMT+5:30

Download Date
Mar 14, 2025, 10:54 AM GMT+5:30

File Name
AI_Product_Recommendation_system.docx

File Size
530.1 KB

9 Pages

4,913 Words

31,162 Characters

1% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 8 words)

Match Groups

- 5 Not Cited or Quoted 1%
  Matches with neither in-text citation nor quotation marks
- 0 Missing Quotations 0%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

1% Internet sources
0% Publications
1% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

CONFERENCE CERTIFICATE

13th INTERNATIONAL CONFERENCE ON CONTEMPORARY ENGINEERING AND TECHNOLOGY 2025

ORGANIZED BY

ORGANIZATION OF SCIENCE & INNOVATIVE ENGINEERING AND TECHNOLOGY (OSIET), CHENNAI, INDIA.

IN COLLABORATION WITH

SAMARKAND STATE UNIVERSITY, UZBEKISTAN

IN ASSOCIATION WITH

PRINCE SHRI VENKATESHWARA PADMAVATHY ENGINEERING COLLEGE

PRINCE DR. K. VASUDEVAN COLLEGE OF ENGINEERING & TECHNOLOGY

(Approved By All India Council For Technical Education, Affiliated To Anna University)

Medavakkam - Mambakkam Main Road, Ponmar, Chennai - 600 127.

Certificate of Presentation

This is to certify that Mr/Mrs/Dr. Badresh B from DR.M.G.R. EDUCATIONAL AND RESEARCH INSTITUTE, CHENNAI has presented a paper titled AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM in the "13th International Conference on Contemporary Engineering and Technology 2025" held on 22nd & 23rd March 2025.

Dr.Akhatov Akmal Rustamovich, Vice Rector of International Affairs, Samarkand State University, Uzbekistan

Dr. Christo Ananth, Professor, Samarkand State University, Uzbekistan

Dr. P. Aravinthan, M.E., Ph.D., Technical Lead, OSIET, Secretary

Dr. A. Krishnamoorthy, M.E., Ph.D., Technical Lead, OSIET, Administrator

K.Janani, M.Tech., CEO, OSIET, Director

13th INTERNATIONAL CONFERENCE ON CONTEMPORARY ENGINEERING AND TECHNOLOGY 2025

ORGANIZED BY

ORGANIZATION OF SCIENCE & INNOVATIVE ENGINEERING AND TECHNOLOGY (OSIET), CHENNAI, INDIA.

IN COLLABORATION WITH

SAMARKAND STATE UNIVERSITY, UZBEKISTAN

IN ASSOCIATION WITH

PRINCE SHRI VENKATESHWARA PADMAVATHY ENGINEERING COLLEGE

PRINCE DR. K. VASUDEVAN COLLEGE OF ENGINEERING & TECHNOLOGY

(Approved By All India Council For Technical Education, Affiliated To Anna University)

Medavakkam - Mambakkam Main Road, Ponmar, Chennai - 600 127.

Certificate of Presentation

This is to certify that Mr/Mrs/Dr. Subrahmanyam Nandigam from DR.M.G.R. EDUCATIONAL AND RESEARCH INSTITUTE, CHENNAI has presented a paper titled AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM in the "13th International Conference on Contemporary Engineering and Technology 2025" held on 22nd & 23rd March 2025.

Dr.Akhatov Akmal Rustamovich, Vice Rector of International Affairs, Samarkand State University, Uzbekistan

Dr. Christo Ananth, Professor, Samarkand State University, Uzbekistan

Dr. P. Aravinthan, M.E., Ph.D., Technical Lead, OSIET, Secretary

Dr. A. Krishnamoorthy, M.E., Ph.D., Technical Lead, OSIET, Administrator

K.Janani, M.Tech., CEO, OSIET, Director


13th INTERNATIONAL CONFERENCE ON CONTEMPORARY ENGINEERING AND TECHNOLOGY 2025
ORGANIZED BY
ORGANIZATION OF SCIENCE & INNOVATIVE ENGINEERING AND TECHNOLOGY (OSIET), CHENNAI, INDIA.
IN COLLABORATION WITH
SAMARKAND STATE UNIVERSITY, UZBEKISTAN
IN ASSOCIATION WITH
PRINCE SHRI VENKATESHWARA PADMAVATHY ENGINEERING COLLEGE
PRINCE DR. K. VASUDEVAN COLLEGE OF ENGINEERING & TECHNOLOGY
(Approved By All India Council For Technical Education, Affiliated To Anna University)
Medavakkam - Mambakkam Main Road, Ponmar, Chennai - 600 127.
Certificate of Presentation
This is to certify that Mr/Mrs/Dr. Jerald F from
DR.M.G.R. EDUCATIONAL AND RESEARCH INSTITUTE, CHENNAI has presented a paper titled
AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM
in the "13th International Conference on Contemporary Engineering and Technology 2025" held on 22nd & 23rd March 2025.
Dr.Akhatov Akmal Rustamovich
Vice Rector of International Affairs, Samarkand State University, Uzbekistan
Dr. Christo Ananth
Professor, Samarkand State University, Uzbekistan
Dr. P. Aravinthan, M.E., Ph.D.,
Technical Lead, OSIET
Secretary
Dr. A. Krishnamoorthy, M.E., Ph.D.,
Technical Lead, OSIET
Administrator
K.Janani, M.Tech.,
CEO, OSIET
Director


13th INTERNATIONAL CONFERENCE ON CONTEMPORARY ENGINEERING AND TECHNOLOGY 2025
ORGANIZED BY
ORGANIZATION OF SCIENCE & INNOVATIVE ENGINEERING AND TECHNOLOGY (OSIET), CHENNAI, INDIA.
IN COLLABORATION WITH
SAMARKAND STATE UNIVERSITY, UZBEKISTAN
IN ASSOCIATION WITH
PRINCE SHRI VENKATESHWARA PADMAVATHY ENGINEERING COLLEGE
PRINCE DR. K. VASUDEVAN COLLEGE OF ENGINEERING & TECHNOLOGY
(Approved By All India Council For Technical Education, Affiliated To Anna University)
Medavakkam - Mambakkam Main Road, Ponmar, Chennai - 600 127.
Certificate of Presentation
This is to certify that Mr/Mrs/Dr. Dineshkumar P from
DR.M.G.R. EDUCATIONAL AND RESEARCH INSTITUTE, CHENNAI has presented a paper titled
AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM
in the "13th International Conference on Contemporary Engineering and Technology 2025" held on 22nd & 23rd March 2025.
Dr.Akhatov Akmal Rustamovich
Vice Rector of International Affairs, Samarkand State University, Uzbekistan
Dr. Christo Ananth
Professor, Samarkand State University, Uzbekistan
Dr. P. Aravinthan, M.E., Ph.D.,
Technical Lead, OSIET
Secretary
Dr. A. Krishnamoorthy, M.E., Ph.D.,
Technical Lead, OSIET
Administrator
K.Janani, M.Tech.,
CEO, OSIET
Director

PUBLICATION ACCEPTANCE

Gmail

Badresh B <badresh.nb@gmail.com>

AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM

1 message

Prof prince <iccet2018@gmail.com> Sat, Mar 15, 2025 at 9:58 AM
To: Badresh B <badresh.nb@gmail.com>

Dear Author/s,

We are happy to inform you that your paper, submitted for the ICCET 2025 conference has been **Accepted** based on the recommendations provided by the Technical Review Committee. By this mail you are requested to proceed with Registration for the Conference. Most notable is that the Conference must be registered on or before **March 19th, 2025(11.59 PM)** from the date of acceptance.

www.iccet.in

Kindly fill the **Registration Declaration form, Certificates form** which is **attached with the mail** and it should reach us on above mentioned days.

Instructions to fill the forms:

- Ensure to send **payment screenshots** and send all the details once the payment has been done to the account(check attachment).
- Journal details (check attachment - **Registration Declaration form**).
- All the above completed details should be mailed to **iccet2018@gmail.com**
- Please send a soft copy of the RESEARCH PAPER in word format only.

NOTE: - Send Abstract and Full paper separately in word format only.

We reserve the right to reject your paper if the registration is not done within the above said number of days.

Paper id: ICCET251868

ICCET 2025
www.iccet.in
9600034378

2 attachments

Certificates form - ICCET 2025.xls
32K

Registration Instructions & Declaration form - 13TH ICCET 2025.doc
136K

Gmail Badresh B <badresh.nb@gmail.com>

Submission of Research Paper and Forms with Transaction Details

4 messages

Badresh B <badresh.nb@gmail.com> Sun, Mar 16, 2025 at 10:31 PM
To: iccet2018@gmail.com

Respected Sir/Mam,
Concerning the previous mail, I have completed the payment for my accepted research paper titled "**AI-POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM**" - **ICCET251868** . I have stated the details of the transaction and I have also attached the required documents -

1. Research Paper
2. Abstract
3. Registration form
4. Journal category
5. Attendance and Declaration form
6. Transaction proof

Details of Challan/ Online Transaction:
Mode of Payment: Bank Transfer(Net Banking)
Account Name: BALDEV ENTERPRISES
Account Number: 23280200004597
Bank Name: Federal Bank
Amount Paid: Rs.2550
Transaction ID: 151473831

Kindly confirm the receipt and documents. Please let me know if any additional details are required. Thank you.

Best Regards,
Badresh B
Dr.M.G.R Educational and Research Institute
9840578909

6 attachments

ABSTRACT - AI POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM.docx
16K

Resreach Paper - AI - POWERED TECHNOLOGICAL PRODUCT SUGGESTION AND RECOMMENDATION SYSTEM.docx
531K

ICCET_category Form.pdf
248K

Registration form - ICCET 2025.xls
33K

ICCET_attendance_and_declaration.pdf
228K

Transaction Proof - ICCET Conference.pdf
110K

Prof prince <iccet2018@gmail.com> Mon, Mar 17, 2025 at 8:07 PM
To: Badresh B <badresh.nb@gmail.com>

Dear Author/s,

Please note that you are now a registered author/s and have received your payment and relevant documents.

Regards,

ICCET
www.iccet.in
[Quoted text hidden]

Badresh B <badresh.nb@gmail.com> Mon, Mar 17, 2025 at 8:29 PM
To: Prof prince <iccet2018@gmail.com>

Dear Prof Prince,
Thanks for your response, and I need to know the standards to submit the ppt and details of when to submit and where to submit.

Best regards,
Badresh B
9840578909
badresh.nb@gmail.com
[Quoted text hidden]

Prof prince <iccet2018@gmail.com> Mon, Mar 17, 2025 at 8:31 PM
To: Badresh B <badresh.nb@gmail.com>

Please follow your college template only.
[Quoted text hidden]