PERSONALIZED PRODUCT RECOMMENDATION SYSTEM

M.M. CHATHUNI D. MANAGE

BSc (HONS) ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

2024

PERSONALIZED PRODUCT RECOMMENDATION SYSTEM

M.M. Chathuni D. Manage

A dissertation submitted in partial fulfilment of the requirement for Bachelor of Science (Honors) degree in Artificial Intelligence and Data Science

School of Computing

Informatics Institute of Technology, Sri Lanka in collaboration with Robert Gordon University, Scotland, UK

2024

Declaration

I declare that the work presented in this dissertation is my own work and to best of my knowledge acknowledgement is made for all sources of information used in this dissertation. Further, this as a whole or as parts has not been submitted previously or concurrently for a degree or any other qualifications at any University or Institutions of Higher Learning.

26/04/2024

………………………………… ……………………………………..

Signature of the student Date

Full Name of student:

Student's Registration No:

The above Student carried out his /her research project under my supervision.

27/04/2024

…………………………………….. ……………………………………..

Signature of the Supervisor Date

Name of the Supervisor: Prashan Rathnayake

ABSTRACT

Product recommendation systems are pivotal in e-commerce platforms, significantly influencing user experience and business revenue by suggesting relevant products to users. These systems harness user data and machine learning to predict and serve user-specific interests.

This project focuses on enhancing the effectiveness of product recommendations in e-commerce by employing Graph Neural Networks (GNNs) and K-means customer segmentation techniques. GNNs are adept at processing the graph-structured data that naturally arises from user-item interactions, capturing complex relationships and user preferences. Concurrently, K-means segmentation further refines the recommendation process by grouping customers into clusters based on their behavior, which allows for more targeted and meaningful suggestions.

The system developed as part of this thesis demonstrates a significant leap in performance and accuracy in product recommendations. It successfully identifies patterns in user behavior and predicts products that align with individual preferences, leveraging a robust dataset and state-of-the-art algorithms. The system's architecture is designed to be scalable and can be integrated seamlessly into existing e-commerce platforms to enhance the user experience. The evaluation of the system utilized a combination of quantitative metrics such as precision, Mean Reciprocal Rank (MRR), and several cluster analysis scores like Silhouette, Calinski-Harabasz, and Davies-Bouldin indices. These metrics collectively helped assess the accuracy, efficiency, and scalability of the recommendation model.

However, the current implementation of the system does not include real-time data processing, which is identified as a key area for future enhancement. Real-time data integration would allow the system to adapt recommendations on the fly, further improving its relevance and accuracy.

Keywords: Graph Neural Networks, Personalized Recommendation Systems, E-commerce, Session-based Recommendations, Artificial Intelligence

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to all those who provided me the possibility to complete this thesis. A special gratitude I give to my supervisor, Mr. Prashan Rathnayake, whose contribution in stimulating suggestions and encouragement helped me to coordinate my project especially in writing this thesis.

My sincere thanks also goes to Mr. Cassim Farook, the course conductor, and all the lecturers of the Informatics Institute of Technology for their continuous support and encouragement throughout my academic journey. Their profound guidance and direction throughout the learning process have enriched my development as a professional.

I am also grateful to the evaluators and friends at the University of Sri Jayewardenepura, whose insights and companionship have been invaluable. Their thoughts and well-wishes have been a source of motivation and inspiration.

Furthermore, I must acknowledge my family: amma and thaththa, as well as my sister and brother for providing me with unconditional support throughout my life, from the initial stage to the completion of my academic projects. Their love and prayers were the bedrock of my perseverance. Lastly, I would like to extend a heartfelt thank you to a special person in my life, who stood by me through the ups and downs and enco

I would like to take a moment to acknowledge my own efforts and dedication throughout this research journey. Balancing the rigorous demands of this project with other responsibilities required significant personal commitment and perseverance. I am proud about me.

Thank you to everyone who has been a part of my journey, either directly or indirectly, in making this project a success.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ML	Machine Learning
GNN	Graph Neural Network
MRR	Mean Reciprocal Rank
GUI	Graphical User Interface
RS	Recommendation System

CHAPTER 01 – INTRODUCTION

1.1 Chapter Overview

The introduction chapter describes the background of the problem, exploring the intricacies of the problem domain, problem definition, and problem statement. It highlights the significance of the research, its underlying motivations, and aims to bridge existing research gaps. Through elucidating the research question and objectives, it delineates the direction of the study, aiming to offer actionable insights into personalized product recommendation systems in the e-commerce domain. Furthermore, the chapter outlines the operational objectives and scope of the research, providing a comprehensive overview. Accompanied by a high-level rich picture diagram, it offers readers a holistic understanding of the research landscape.

1.2 Background of the Problem

1.2.1 Problem domain

The domain of e-commerce presents a dynamic and competitive landscape, businesses are always working to attract customers and increase their sales. Central to this endeavor is the challenge of providing personalized and relevant product recommendations to users amidst the vast array of available choices. In the digital marketplace, where consumers are inundated with options, the need for effective recommendation systems becomes paramount. Understanding user behavior, preferences, and contextual factors is crucial for delivering tailored recommendations that resonate with individual interests, thereby enhancing the overall shopping experience.

1.2.2 Problem definition

Recommendation systems have emerged as essential tools in today's digital world, spreading across many aspects of online activities. From personalized product suggestions on e-commerce platforms to customized content feeds on social media networks, these systems play a main role in enhancing user engagement and driving business growth. By leveraging deep learning algorithms, machine learning algorithms and user data analysis, recommendation systems aim to expect and fulfill the diverse needs and preferences of individuals, thereby facilitating informed decision-making and fostering long-term user satisfaction.

In the domain of e-commerce, implementing effective recommendation systems is crucial for businesses striving to optimize their marketing strategies and enhance customer experiences. By using the knowledge obtained from user behavior, purchase history, and contextual information, these systems can deliver highly targeted recommendations that align with individual interests and preferences. Consequently, businesses stand to benefit from increased customer engagement, improved retention rates, and enhanced revenue generation through cross-selling and upselling opportunities.

However, the development of a robust recommendation system poses several challenges, necessitating a comprehensive understanding of both user dynamics and algorithmic methodologies. Successful implementation requires a nuanced grasp of machine learning models, data preprocessing techniques, and evaluation metrics to ensure the delivery of accurate and relevant recommendations. Moreover, considerations such as scalability, privacy concerns, and ethical implications underscore the complexity of designing and deploying recommendation systems in real-world settings.

1.2.3 Problem statement

This project aims to address the challenge of providing personalized product recommendations in e-commerce through the development of an advanced recommendation system leveraging Graph Neural Networks (GNNs). By harnessing the capabilities of AI within the online shopping domain, the system seeks to optimize user satisfaction, engagement, and conversion rates for e-commerce platforms.

1.3 Research Motivation

The motivation for this project stems from the growing importance of personalized product recommendation systems in the e-commerce domain. As online shopping continues to expand, businesses face the challenge of providing tailored recommendations amidst a vast array of choices. By leveraging advanced technologies such as machine learning and artificial intelligence, there exists an opportunity to enhance user experiences, drive engagement, and increase conversion rates. Additionally, the potential impact of effective recommendation systems on business growth and competitiveness further underscores the significance of this research. Consequently, this project is driven by the desire to address these challenges, optimize recommendation processes, and contribute to the advancement of e-commerce practices.

1.4 Research Aim

The aim of this project is to develop an advanced recommendation system tailored for the e-commerce domain. By leveraging cutting-edge technologies such as Graph Neural Networks (GNNs) and artificial intelligence (AI), the project aims to address the challenge of providing personalized and relevant product recommendations to users.

1.5 Research Gap

While considerable research has been conducted on recommendation systems in the e-commerce domain, there remains a notable gap in the literature concerning the integration of Graph Neural Networks (GNNs) for personalized product recommendations. Existing studies predominantly focus on traditional collaborative filtering and content-based approaches, often overlooking the potential benefits offered by GNNs in capturing complex user-item interactions and incorporating contextual information.

Furthermore, the majority of research in this field tends to overlook the dynamic nature of user preferences and the evolving nature of e-commerce platforms. There is a paucity of studies that systematically investigate the temporal dynamics of user behavior and the adaptability of recommendation systems to changing preferences and trends over time.

Moreover, while some research has explored the effectiveness of machine learning algorithms in generating personalized recommendations, there is a lack of consensus on the most suitable evaluation metrics and methodologies for assessing the performance of recommendation systems in real-world settings. Additionally, limited attention has been paid to the ethical considerations surrounding recommendation algorithms, such as fairness, transparency, and privacy, which are crucial for maintaining user trust and confidence in e-commerce platforms.

Addressing these research gaps is imperative for advancing the field of personalized product recommendation systems in e-commerce. By leveraging the capabilities of GNNs and incorporating temporal dynamics and ethical considerations into the recommendation process, this project aims to contribute to a more comprehensive understanding of recommendation system development and deployment, ultimately enhancing user satisfaction and driving business growth in the digital marketplace.

1.6 Research Contribution

1.6.1 Research domain contribution.

Your research enhances the domain of session-based recommendation systems by applying GNN to model session data as graph-structured, capturing complex item transitions that conventional sequential methods miss. This novel approach not only aligns with but also advances current academic discussions on improving accuracy and relevance in recommendation systems, as demonstrated in your systematic use of GNN for better user session representation and prediction accuracy.

1.6.2 Problem domain contribution.

In the problem domain, your research addresses the specific challenge of session-based recommendations where user profiles are often anonymous and limited to single session data. This is a significant hurdle in many real-world applications like e-commerce and media streaming, where understanding short-term user preferences is crucial for enhancing user experience and engagement. Your model leverages the graph-based structure to better understand and predict user behavior within these constraints, potentially leading to more effective recommendation strategies in industries reliant on session data.

1.6.3 Technological contribution

Technologically, your project contributes by integrating advanced GNN architectures to process and analyze session data, a technique not widely adopted in session-based recommendation systems before. By doing so, you provide a framework that can be applied to other similar datasets and problems, extending the utility of GNN beyond traditional applications. Your work also sets a foundation for future explorations into more efficient and accurate recommendation systems, encouraging further innovation in the use of neural networks for complex data structures.

1.7 Research challenges.

Implementing a Graph Neural Networks (GNN)-based recommendation system for session-based recommendations presents several significant challenges, each crucial for the project's success and advancement in the recommendation system domain.

Data Sparsity and Quality: A primary challenge is the inherent sparsity and variability of session data. Users typically interact with a limited set of items per session, which can lead to sparse data that complicates learning effective user preferences. Additionally, session data can suffer from quality issues such as noise and missing values, posing further challenges in training robust GNN models that reliably predict user preferences.

Scalability and Computational Efficiency: Scalability is critical as recommendation systems need to manage large volumes of users and items. GNNs, which process complex graph structures, are computationally intensive. Developing a scalable GNN that processes extensive data efficiently without sacrificing recommendation quality poses a substantial challenge, necessitating innovative approaches to model optimization and deployment.

Real-Time Processing: The demand for real-time recommendations based on live user session data is essential for enhancing user experience. However, the integration of GNNs into real-time systems is challenging due to the computational demands of graph data processing. Achieving a balance between speed and accuracy in a live environment requires strategic innovations in data handling and model performance.

Cold Start Problem: The cold start problem, particularly acute in session-based systems where user identities may not persist across sessions, complicates making accurate recommendations for new users or items with limited historical data. Addressing this within the GNN framework may require exploring hybrid models or meta-learning strategies to enhance the system's adaptiveness.

Model Generalization: Ensuring that the GNN generalizes well to unseen data is crucial to prevent overfitting, which is common in complex models trained on limited session data. Developing and implementing effective regularization and validation strategies to test and enhance the model's robustness is essential for its success.

Integration with Existing Systems: Integrating advanced GNN models with existing recommendation frameworks presents both technical and organizational challenges. Issues such as compatibility with existing technologies, adapting to current workflows, and gaining stakeholder support necessitate careful planning and clear communication.

1.8 Research Questions

RQ1: What are the most effective evaluation metrics and methodologies for assessing the performance and efficacy of personalized recommendation systems?

RQ2: How do different recommendation algorithms, including traditional collaborative filtering and content-based approaches, compare with GNN-based approaches in terms of recommendation quality and performance metrics?

RQ3: How can Graph Neural Networks (GNNs) be effectively leveraged to enhance the accuracy and relevance of personalized product recommendations in e-commerce?

RQ4: What are the key factors influencing user engagement and satisfaction with personalized recommendation systems in the e-commerce context?

1.9 Research Objectives

The aims and research questions outlined above serve as the foundation for the following detailed research objectives, which are designed to guide the project towards successful outcomes. Each objective is a milestone meant to ensure that all facets of the research are comprehensively addressed.

Table 1: Research Objectives

Objective	Description	Learning Outcomes	Research Questions
Literature Review	Collect relevant information by using related works.	LO1 LO2	RQ1 RQ2

	RO1: Conduct a comprehensive review of current literature in recommendation systems, focusing on graph-based approaches. **RO2:** Perform an in-depth analysis of existing recommendation algorithms and their architectural frameworks. **RO3:** Evaluate various metrics used to assess the performance of RS. **RO4:** Analys the use of K-mean clustering for existing projects.	LO5	RQ3
Requirement Analysis	Using appropriate tools & techniques collect & analyze project requirements. **RO5:** Determine the technical and functional requirements of a GNN-based recommendation system. **RO6:** Investigate and document user engagement factors within e-commerce platforms. **RO7:** Gather insights from data patterns in user-item interactions to inform GNN model development and K-mean clustering.	LO2	RQ2 RQ3 RQ4
Design	Design the architecture and system to solve the identified problems. **RO8:** Design a scalable and modular GNN architecture to facilitate flexible integration and future expansion.	LO3 LO4 LO5	RQ2 RQ3 RQ4

	RO9: Develop a comprehensive data preprocessing and normalization framework to optimize input data for the GNN. **RO10:** Create a robust real-time user interaction capture module that feeds into the GNN for dynamic recommendation updates. **RO11:** Design an intuitive and engaging user interface that effectively presents GNN-generated recommendations and collects user feedback.		
Development	Implementing a system that effectively covers the research gap. **RO12:** Implement the designed GNN model, ensuring it handles the complexity of session data with high efficiency. **RO13:** Develop the back-end logic for the user feedback module to refine and personalize the recommendation process.	LO4	RQ1 RQ3 RQ4
Testing & Evaluation	Testing the created GNN based model and data science models. And evaluate them with baseline techniques. **RO14:** Test the system's performance and user satisfaction against established benchmarks. **RO15:** Compare the GNN-based	LO4 LO5	RQ1 RQ2 RQ4

	system's outcomes with traditional recommendation methods.		
Documentation	Document the progress of the research project.	LO5	General
Publish Findings	Publish the research and review works. **RO16:** Publish the review paper on previous works. **RO17:** Document the research process and publish findings in academic journals.	LO5	RQ1 RQ2

1.10 Operational Objectives

Operational objectives are the practical goals that guide the day-to-day progress and management of the GNN-based recommendation system project. These objectives ensure that the project remains on track, within scope, and is executed efficiently.

Develop a Project Timeline: Establish a detailed project timeline with milestones to guide the research and development process, ensuring that each phase of the project, from design to testing, aligns with the planned schedule.

Maintain Regular Communication: Set up a regular meeting schedule with the project supervisor to review progress, discuss challenges, and receive feedback, ensuring that the project remains aligned with its goals.

Resource Management: Efficiently manage and allocate resources, including software tools, datasets, and computational resources, to avoid bottlenecks in the development process.

Documentation: Keep comprehensive documentation of the development process, including design decisions, changes to the project scope, and the rationale behind each significant choice to ensure transparency and facilitate future work on the project.

Risk Management: Identify potential risks to the project timeline or outcomes early on, and develop contingency plans to mitigate these risks, ensuring that the project can adapt to unforeseen challenges.

Performance Monitoring: Continuously monitor the system's performance against the set benchmarks and optimize as necessary to meet or exceed expected outcomes.

Ethical and Legal Compliance: Ensure that the project complies with all relevant ethical guidelines and legal requirements, particularly in terms of user data privacy and security.

1.11 Project Scope

1.11.1 In-scope

System Design and Development:

- Designing a modular GNN architecture optimized for session-based recommendation scenarios.
- Developing a data preprocessing pipeline that feeds into the GNN model effectively.
- Integrating user feedback mechanisms to refine the recommendation process.

Evaluation and Benchmarking:

- Testing the GNN model against established recommendation system benchmarks.
- Evaluating the system's performance in terms of accuracy, efficiency, and user satisfaction.

Research and Documentation:

- Documenting the development process, system design choices, and testing outcomes.
- Writing a comprehensive final report that includes a literature review, methodology, results, and conclusions.
- Preparing academic papers or conference materials based on the project findings.

1.11.2 Out-scope

- Conducting a wide-ranging market analysis for recommendation systems beyond the e-commerce domain.
- Deep psychological profiling or sophisticated demographic analysis that requires personal data not available in session logs.
- Deep psychological profiling or sophisticated demographic analysis that requires personal data not available in session logs.

1.12 High Level Rich Picture of the Proposed solution

The high-level rich picture of the solution included in the Software Requirement Chapter (Chapter 3)

1.13 Chapter summary

The introduction chapter provides an overview of the recommendation system, delving into the problem domain, definition, and statement. It explains the research motivation, aim, and gap, emphasizing contributions to both the research and problem domains. Challenges within these domains are identified, followed by a list of research questions, objectives, and operational objectives. The scope of the project is defined clearly. And finally rendered the high-level depiction of the proposed solution.

2 CHAPTER 02 – LITERATURE REVIEW

2.1 Chapter Overview

This chapter offers a comprehensive literature review on personalized product recommendation systems in e-commerce. It outlines the research methodology, research onion framework, and presents a concept map to depict key themes and concepts. The chapter explores existing work in the domain and problem area, comparing different methodologies. A technological review highlights advancements in recommendation system technologies. Evaluation approaches, including system and algorithm benchmarking, are discussed. Finally, a summary encapsulates the key findings and insights from the literature review.

2.2 Research Methodology

Table 2 : Research Methodology

Philosophies	Research methodologies encompass various philosophical perspectives that guide the researcher's approach. **Interpretivism**, focusing on the subjective nature of user preferences and pointing up the need for understanding the context of online shopping behaviors, underscores the significance of individual experiences. Meanwhile, **pragmatism** advocates a flexible approach, combining both qualitative and quantitative methods, ensuring a holistic understanding of personalized recommendation systems and addressing the complexities of user behavior in online shopping effectively.
Approach	The approach refers to the overarching strategy guiding the research. In a **deductive** approach for a personalized recommendation system in online shopping, the researcher starts with established theories or principles and applies them to specific instances. This method aims to test existing hypotheses, ensuring a systematic and structured investigation into the effectiveness of recommendation algorithms.
Strategies	A suitable research strategy would be a **Systematic Literature**

	Review. This approach involves a comprehensive and structured analysis of existing literature to identify trends, gaps, and insights related to personalized recommendation systems in online shopping. A systematic literature review will provide a solid foundation for understanding the current state of research in the field.
Choices	The chosen methodology for this study is a **Mixed Method** approach, integrating qualitative and quantitative research. A mono-method may lack depth in understanding user preferences in personalized recommendation systems. By utilizing interviews and surveys alongside quantitative analyses, this research aims for a comprehensive exploration of user experiences and system performance in online shopping.
Time horizons	The research's time horizon adopts a **Cross-Sectional** approach, focusing on a specific point in time to analyze user preferences and system performance within the dynamic context of a "Personalized Recommendation System for Online Shopping." This method allows for a snapshot examination, aligning with the instantaneous nature of online shopping interactions without the extended timeline of longitudinal studies.
Technique & Procedures	The "Personalized Recommendation System for Online Shopping" opts for a blend of primary and secondary data. Utilizing **surveys and interviews** for user insights (primary) and leveraging purchase history and browsing patterns (secondary) alongside quantitative analysis, ensures a comprehensive evaluation of system performance.

2.3 Research onion.

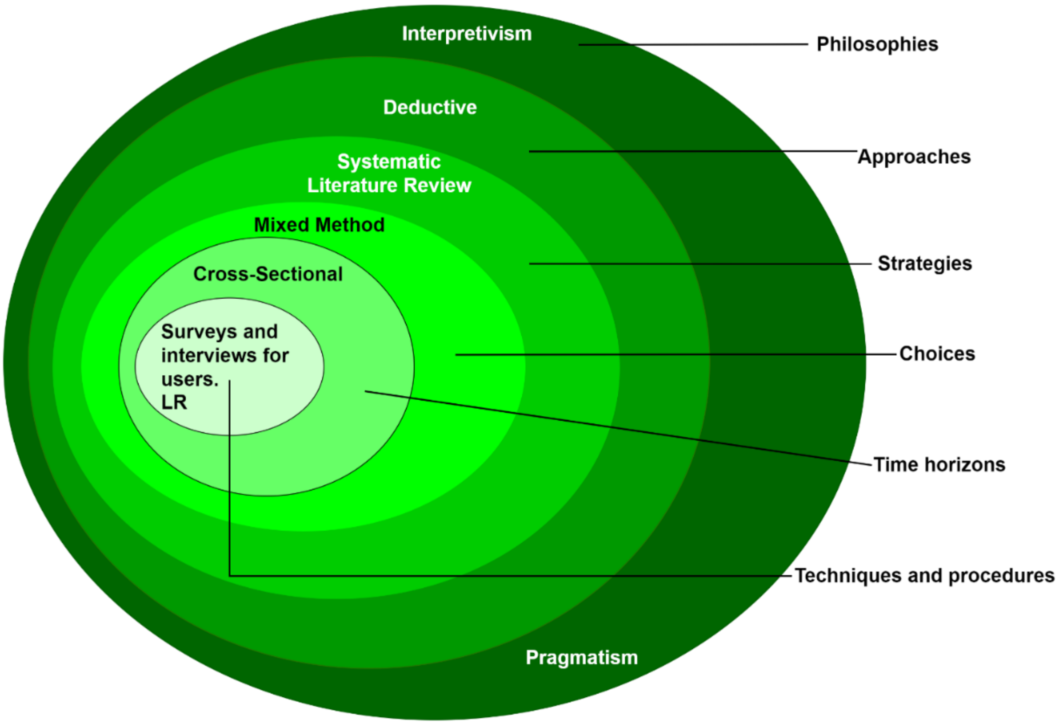


Figure 1 : Research Onion Diagram

2.4 Concept Map

The concept map is available in APPENDIX A1.

2.5 Existing work

This section provides an overview of existing research and studies related to personalized product recommendation systems in the e-commerce domain. The literature review encompasses research into both the broader domain of recommendation systems and the specific problem area addressed in this study.

2.5.1 Research into the Domain.

Numerous studies have explored various aspects of recommendation systems in the e-commerce domain. Research has investigated different recommendation algorithms, including collaborative filtering, content-based filtering, and hybrid approaches. Studies have also examined the impact of user preferences, data collection methods, and machine learning techniques on recommendation system performance. Additionally, research has explored user engagement metrics, evaluation methodologies, and ethical considerations in recommendation system design and deployment.

2.5.1.1 Collaborative filtering

Collaborative filtering is a widely studied approach in recommendation systems, focusing on generating recommendations based on the preferences and behaviors of similar users. Research in collaborative filtering explores various algorithms and techniques for identifying similarities between users or items and leveraging this information to make personalized recommendations. Traditional collaborative filtering methods include user-based and item-based approaches, which rely on user-item interaction data such as ratings or purchase history.

The previous study explores Collaborative Filtering (CF) and Matrix Factorization (MF) algorithms to address scalability and accuracy challenges in recommendation systems. By employing memory-based and model-based CF techniques alongside advanced MF methodologies like Probabilistic Matrix Factorization (PMF), the research aims to enhance recommendation accuracy and computational efficiency. Through probabilistic approaches and dimensionality reduction, user preferences are captured more effectively while addressing sparse data challenges. While specific architectural details are not outlined, distributed computing likely plays a key role in handling large-scale datasets and deploying recommendation models. This investigation lays groundwork for more personalized and relevant recommendations in the digital landscape. (Venkatesan, 2023)

The paper "Applying Particle Swarm Optimization Algorithm-based Collaborative Filtering Recommender System Considering Rating and Review" introduces a novel approach to address the pervasive challenge of data sparsity in recommendation systems. Leveraging innovative algorithms and techniques, the study focuses on enhancing collaborative filtering methodologies. One key algorithm utilized is Particle Swarm Optimization (PSO), which optimizes the computation of similarity matrices between users or products, thereby mitigating data sparsity issues. Additionally, Bidirectional Encoder Representations from Transformers (BERT) are employed to extract rich characteristics from textual feedback, augmenting the recommendation process with valuable insights from user reviews. The proposed architecture integrates both numerical rating data and textual review data, employing data fusion methodologies to effectively combine them. Through comprehensive experimentation, the paper demonstrates the superiority of the proposed method in improving recommendation performance compared to existing approaches, offering a promising solution to the challenges posed by data sparsity in recommendation systems.(R.J. Kuo and Shu-Syun Li, 2023)

The paper employs sophisticated algorithms such as Vector Autoregression (VAR) for data analysis. Techniques like econometric analysis and panel framework analysis are utilized to derive insights from the data. Methodologies include a fuzzy decision-making model and panel framework analysis, enabling a comprehensive examination of the interdependence between oil prices and exchange rate movements. These methodologies are integrated into a coherent framework, forming the architecture of the study. Through the application of these advanced tools, the paper provides valuable insights into the dynamics of exchange rates in oil-exporting countries, contributing to the understanding of global financial markets. (Alexey Mikhaylov *et al.*, 2022)

The paper outlines a comprehensive methodology for group recommendation systems, employing a range of algorithms, techniques, and architectures to tackle the complexities of cold start issues and enhance recommendation accuracy. Central to the approach is the utilization of Matrix Factorization (MF) and General Matrix Factorization (GMF) algorithms, which decompose the group-item rating matrix into latent vectors for users and items, enabling effective prediction through element-wise dot product operations. Moreover, Neural Collaborative Filtering (NCF) techniques are harnessed to capture intricate user-item interactions and augment recommendation accuracy. Metadata embedding plays a pivotal role, incorporating user demographic details and item features like genres into the neural network architecture through one-hot encoding and clustering methods. The methodology encompasses an aggregate model approach, involving the creation of user profiles, formation of groups using clustering algorithms, and prediction ratings for groups through various aggregation strategies. Extensive experimental evaluation on existing datasets is conducted, comparing the proposed approach with state-of-the-art methods, utilizing evaluation metrics such as RMSE and MAE to assess recommendation performance. Overall, the paper's methodology integrates diverse algorithms and techniques to address the challenges of group recommendation, with a focus on leveraging metadata to improve prediction accuracy and mitigate cold start issues. (2023)

The paper introduces UI2vec, a collaborative filtering recommendation algorithm that leverages embedding representation and word embedding techniques to enhance recommendation accuracy. UI2vec embeds users and items into a shared latent space using the Doc2vec word embedding model, enabling the calculation of item similarities based on the distances between their embedded representations. Additionally, the paper presents VUI2vec, a generative probabilistic model that maps user and item embeddings to independent Gaussian distributions, utilizing variational inference to approximate their posterior distribution. Optimization techniques such as negative sampling are employed to improve training efficiency, while experiments conducted on real-world datasets (TaFeng, Movielens, and Netflix) evaluate the performance of UI2vec and VUI2vec against baseline models like SVD, PMF, Item2vec, and Metapath2vec. Results are assessed using precision, recall, and F1-score metrics, highlighting the effectiveness of the proposed algorithms. The paper concludes with insights into hyperparameter analysis and suggests future research directions in collaborative filtering recommendation systems. (Mohamed Ali Rakrouki and Abeer Aljohani, 2023)

The paper introduces a personalized recommendation system for e-commerce, employing collaborative filtering as its primary algorithm. To address scalability challenges, it adopts user clustering techniques, grouping similar users based on preferences. Leveraging a Multi-Agent System (MAS) methodology, the system comprises autonomous agents responsible for tasks like user profiling, data mining, and recommendation generation. The MAS architecture fosters collaboration among agents, facilitating efficient recommendation delivery. The recommendation engine architecture encompasses multiple agents, including a search engine, recommendation generator, and evaluation feedback agent, enabling continuous refinement of recommendation algorithms. Through collaborative filtering and user clustering, the system tailors recommendations to individual users while overcoming scalability constraints. This comprehensive framework offers a robust solution for providing personalized recommendations in e-commerce, enhancing user experience and satisfaction. (Nagagopiraju Vullam *et al.*, 2023)

2.5.1.2 Content-based filtering

Content-based filtering is a recommendation system technique that suggests items based on their intrinsic characteristics and features. Unlike collaborative filtering, which relies on user-item interactions and similarities, content-based filtering focuses solely on the attributes of items themselves. This approach involves analyzing item descriptions, metadata, or other relevant information to understand their properties and relationships. By creating profiles or representations of both items and users, content-based filtering identifies items that match the preferences of a user based on their past interactions or explicit feedback.

The paper proposes a novel approach to product recommendation systems, integrating advanced NLP techniques and CNNs for improved accuracy. Algorithms such as Bag of Words, TF-IDF, and Word2Vec analyze textual data to extract semantic similarities and relevant features. Euclidean distance calculations assess the similarity between product vectors, while content-based recommendations ensure relevance. Extensive data preprocessing and feature selection optimize system performance, with VGG-16 CNN architecture facilitating robust image feature extraction. Deployment as a website provides users with an interactive platform for personalized recommendations. Future enhancements include AB testing for validation, collaborative filtering integration, and the incorporation of additional attributes like product descriptions for refinement. Overall, the paper presents a comprehensive framework leveraging cutting-edge technologies to deliver tailored recommendations, promising enhanced user experiences in the digital marketplace. (Akhilesh Kumar Sharma *et al.*, 2023)

The paper "OCA: Ordered Clustering-Based Algorithm for E-Commerce Recommendation System" introduces a comprehensive approach to address the challenges of cold-start and data sparsity in e-commerce recommendation systems. At its core, the Ordered Clustering (OC) algorithm emerges as a novel solution, leveraging Collaborative Filtering (CF) techniques to cluster users based on their preferences. Methodologically, the paper emphasizes the importance of data preprocessing, ensuring the data is cleansed, normalized, and transformed into a suitable format for clustering. Following this, similarity measurement techniques, particularly the Pearson Correlation Coefficient, are employed to estimate the likeness between users' preferences. While the paper doesn't explicitly outline the architecture of the e-commerce recommendation system, it's inferred that the proposed OC algorithm integrates within a broader system architecture, likely comprising components for data collection, preprocessing, recommendation generation, and user interface. Overall, through a combination of algorithms, techniques, methodologies, and architectures, the paper presents a promising approach to enhance the accuracy and effectiveness of e-commerce recommendation systems. (Yonis Gulzar *et al.*, 2023)

The paper "Content Based Apparel Recommendation Engine" employs a multifaceted approach integrating diverse algorithms, techniques, methodologies, and architectures to enhance apparel recommendation systems. Leveraging Bag_of_Words (BoW), TF-IDF, Word2Vec Model, and VGG16 (CNN) algorithms, the system conducts comprehensive analysis on textual and visual data associated with apparel products. Content-Based Filtering is utilized to match items based on features like brand, color, and image, while the Product Advertising API facilitates dataset acquisition from Amazon's e-commerce platform. Methodologically, the paper outlines a systematic data processing and analysis framework, coupled with an evaluation scheme to gauge recommendation accuracy. Architecturally, the system is structured with components for data acquisition, preprocessing, feature extraction, similarity computation, and recommendation generation, forming a robust pipeline for personalized apparel recommendations. Overall, this integrated approach underscores the paper's endeavor to advance content-based recommendation systems in the domain of fashion e-commerce. (Dr Monika Mehra and Ravinder Pal Singh, 2023)

2.5.1.3 Hybrid approaches

Hybrid approaches amalgamate the strengths of multiple recommendation techniques, such as content-based filtering and collaborative filtering, to mitigate the limitations of individual methods and enhance recommendation accuracy. By combining content-based analysis of item features with collaborative filtering based on user behavior patterns, the system can provide more nuanced and personalized recommendations. This hybridization enables the system to capitalize on the advantages of both approaches, addressing challenges like the cold start problem and data sparsity while improving scalability and recommendation quality.

The paper delves into the intricate landscape of recommendation systems, employing Collaborative Filtering (CF) and Content-based Filtering (CB) algorithms to tackle the overwhelming abundance of online information. Leveraging memory-based CF techniques, it computes user and item similarities based on past interactions, while model-based CF methods identify patterns through machine learning and matrix factorization. Content-based approaches match item attributes with user preferences for personalized recommendations. Methodologically, it conducts a systematic literature review to discern research challenges and opportunities, proposing hybridization with machine learning and future integration of deep neural networks. These architectures aim to enhance recommendation accuracy and navigate the evolving complexities of user preferences in the digital realm. (Gavade Ashwini and Mane Seema, 2023)

The paper outlines a comprehensive approach to building a recommendation system tailored for MSMEs seeking product sales ideas. Leveraging a blend of algorithms, techniques, methodologies, and architectures, the system aims to provide targeted recommendations to users. Algorithms such as TF-IDF facilitate the weight calculation of words in product descriptions, ensuring the relevance and importance of each term. Previous research has explored the Simple Additive Weighing Method, albeit with limitations due to fuzzy number calculations. Techniques like content-based filtering analyze product descriptions to recommend items based on user preferences, while collaborative filtering incorporates user opinions to predict suitable products. Methodologies encompass data collection through literature studies and interviews, followed by preprocessing steps like tokenization, stopword removal, and stemming to refine the dataset. The system's architecture relies on Python programming language for implementation, supported by a MySQL database for data storage, reflecting an e-commerce design paradigm for effective deployment and scalability. (Asep Id Hadiana and Edvin Ramadhan, 2023)

The paper presents a comprehensive approach to recommendation systems, employing a fusion of Collaborative Filtering (CF) and Content-Based Filtering (CBF) algorithms. Through the integration of CF and CBF, the system addresses the limitations inherent in each technique, offering enhanced recommendation accuracy. Leveraging the Firefly Algorithm (FF) for optimization, the system optimizes the recommendation process by fine-tuning parameters and improving system performance. The methodology involves feature extraction, including profile construction, content similarity index calculation, neighbor finding, item generation, and item weight and variance generation. By combining these techniques, the system constructs a feature-based architecture capable of providing personalized recommendations based on user preferences and item attributes. The FF-WCSA model architecture further enhances recommendation quality, ensuring that users receive tailored recommendations that align closely with their interests and preferences. (Shanmugam Sathiya Devi, 2022)

The paper presents a sophisticated recommender system for online stores, integrating various algorithms, techniques, methodologies, and architectural principles to enhance recommendation accuracy and user experience. Leveraging collaborative filtering, the system analyzes user behavior and preferences, while content-based filtering extracts product features to suggest similar items. Singular Value Decomposition refines collaborative filtering by identifying latent factors in user-item interactions. Adopting a hybrid approach, the system combines these techniques to mitigate their individual limitations, providing diverse and accurate recommendations. Additionally, a fuzzy expert system evaluates recommended products based on multiple parameters, ensuring relevance and adaptability. This modular architecture allows for independent development and integration of subsystems, while the web-based interface ensures accessibility and ease of use. Through the synergy of these elements, the proposed recommender system offers a comprehensive solution for online stores, promising improved recommendation quality and user satisfaction. (Petr Fajmon, 2023)

The paper presents a sophisticated approach, the Sparsity and Cold Start Aware Hybrid Recommender System (SCSHRS), to mitigate the challenges of data sparsity and cold start in recommendation systems. Leveraging a suite of algorithms and techniques, including Sparsity Resolving Collaborative Filtering (SRCF), Sparsity Resolving Weighted Collaborative Filtering (SRWCF), Ant-Lion Optimization (ALO), K-means clustering, Higher-Order Singular Value Decomposition (HOSVD), and Adaptive Neuro-Fuzzy Inference System (ANFIS), the SCSHRS methodology is meticulously designed across four stages. Initially, data sparsity is addressed through SRCF and SRWCF methods, followed by Ant-Lion-based k-means clustering to group similar users. Dimensionality reduction with HOSVD optimizes computational efficiency, while ANFIS prediction enhances recommendation accuracy. This modular architecture of SCSHRS not only addresses data sparsity and cold start issues but also offers a flexible and scalable solution for recommendation systems in e-commerce platforms, promising improved user experience and recommendation diversity. (S. Gopal Krishna Patro *et al.*, 2022)

The paper presents a sophisticated recommendation system integrating diverse algorithms, techniques, methodologies, and architectures. Collaborative Filtering (CF) forms the foundation, analyzing user-item interactions for personalized recommendations. Masked LM (MLM) and Next Sentence Prediction (NSP) algorithms enhance text understanding, while Content-adaptive recurrent unit (CARU) and Global pooling (GP) optimize data processing. Sentiment Analysis (SA) enriches recommendations by incorporating user emotions. The methodology employs a Hybrid Recommendation System, blending CFM, Content-based filtering, and Neural Network models like GHSOM. Architectures include BERT for contextual understanding and Multi Content-Adaptive Recurrent Unit (M-CARU) for sentiment disambiguation. The Vector Space Model (VSM) facilitates content-based recommendations. Together, these components create a robust system capable of delivering accurate and personalized recommendations in e-commerce applications, enhancing user satisfaction and engagement. (Arodh Lal Karn *et al.*, 2022)

2.5.2 Research into the problem.

Within the specific problem area of personalized product recommendation systems, research has focused on optimizing recommendation algorithms for improved accuracy and relevance. Studies have investigated the use of advanced techniques such as graph neural networks (GNNs) to model user-item interactions and enhance recommendation quality. Additionally, research has explored novel approaches to data collection, user profiling, and context-aware recommendation to better understand user preferences and behavior in e-commerce settings.

2.5.2.1 Recommendation stages

The paper introduces an innovative approach to recommendation systems, leveraging Graph Convolution Networks (GCNs) as its core algorithm. Through the application of interest-aware message-passing techniques, it addresses the over-smoothing issue inherent in traditional GCN-based models. This technique involves operating high-order graph convolutions within carefully constructed subgraphs, focusing on users with similar interests and their interacted items. The methodology includes an unsupervised subgraph generation process, integrating user features and graph structure information to identify and group users effectively. The resulting architecture, known as the Interest-aware Message-Passing GCN (IMP-GCN) model, combines these techniques to improve recommendation accuracy significantly. By excluding irrelevant information propagation and focusing on relevant high-order neighbors, IMP-GCN offers a promising solution to the challenges faced by existing recommendation systems, ultimately enhancing user experience and satisfaction. (Fan Liu *et al.*, 2021)

The paper explores cutting-edge algorithms and techniques in recommendation systems, particularly focusing on Graph Convolution Networks (GCNs) like PinSage and LightGCN. These models leverage multi-hop neighbors to enhance node representation learning, achieving remarkable performance in recommendation tasks. Introducing a novel approach termed Self-supervised Graph Learning (SGL), the paper augments classical supervised learning with self-supervised tasks, employing methodologies such as node dropout, edge dropout, and random walk for data augmentation. This multi-task learning strategy, combined with contrastive learning, enhances node representation robustness and accuracy. LightGCN serves as the architectural backbone for implementing the SGL paradigm, emphasizing the critical role of graph-based models in recommendation systems. (Jiancan Wu *et al.*, 2021)

The paper introduces Learnable-Time ODE-based Collaborative Filtering (LT-OCF), a novel approach to recommendation systems. By leveraging Neural Ordinary Differential Equations (NODEs), LT-OCF redefines linear Graph Convolution Networks (GCNs), enabling the learning of optimal architectures and smooth ODE solutions tailored for collaborative filtering tasks. This methodology involves redesigning GCNs based on continuous and learnable time variables, allowing for dynamic layer combinations. The architecture integrates dual co-evolving ODEs for user and product embeddings, fostering interactions over continuous time intervals. Moreover, various ODE solvers are employed, enriching neural network connections and enhancing model adaptability. In essence, LT-OCF offers a sophisticated framework that surpasses traditional methods in accuracy and performance. (Jinsung Jeon and Noseong Park, 2021)

In the paper "HS-GCN: Hamming Spatial Graph Convolutional Networks for Recommendation," a novel approach is proposed to address large-scale recommender systems' efficiency and representation challenges. The algorithm at the core of this approach is the Hamming Spatial Graph Convolutional Network (HS-GCN), which integrates graph convolutional techniques with hash coding methods. By representing users and items as binary hash codes in the Hamming space, HS-GCN efficiently computes similarities while preserving essential structural information. Techniques such as hash coding, graph convolution, and code aggregation and encoding are employed to capture both first- and high-order similarities between users and items. This learning to hash framework, coupled with graph-based representation learning methodologies, forms the foundation of HS-GCN's architecture. Through multiple layers including initialization, propagation, and prediction, HS-GCN effectively learns representations that significantly enhance recommendation performance on large-scale datasets. (Han Liu *et al.*, 2022)

The paper introduces a comprehensive framework for Click-Through Rate (CTR) prediction, leveraging advanced algorithms, techniques, methodologies, and architectures. It employs Graph Convolutional Networks (GCNs) and LightGCN for attribute graph convolution, facilitating effective feature embedding. Additionally, it utilizes neighbor-aggregation techniques like NGCF and organized learning mechanisms for behavior expanding, ensuring efficient information propagation within and across graphs. The Dual Graph Enhanced Embedding Neural Network (DG-ENN) architecture forms the core of the approach, incorporating attribute and collaborative graphs. It employs divide-and-conquer strategies for field-wise attribute modeling and curriculum-learning-inspired organized learning for correlation modeling. This holistic approach enables the model to overcome feature and behavior sparsity challenges, leading to improved CTR prediction accuracy in online advertising and recommender systems. (Wei Guo *et al.*, 2021)

The paper presents a groundbreaking approach to recommendation systems through the Sequence-aware Heterogeneous Graph Neural Collaborative Filtering (SHCF) model. Leveraging techniques like Heterogeneous Information Network construction and advanced message passing layers, SHCF seamlessly integrates sequential patterns and heterogeneous collaborative signals. It employs element-wise and sequence-aware self-attention mechanisms to capture fine-grained user preferences and dynamic interests over time. Additionally, a dual-level attention mechanism is utilized to aggregate neighboring information for item embeddings, enhancing the model's ability to handle diverse data types. This innovative methodology results in a robust architecture capable of significantly outperforming existing methods, marking a significant advancement in recommendation system research. (Chen Li *et al.*, 2021)

The paper employs cutting-edge algorithms and techniques to enhance context-aware recommendation systems. Leveraging graph convolutional neural networks (GCNs), the Graph Convolution Machine (GCM) framework integrates context information into user-item interactions, refining embeddings for more accurate predictions. Adopting an encoder-decoder architecture, the model projects users, items, and contexts into embedding vectors, facilitating end-to-end learning. Methodologically, the paper adopts an attributed graph representation, treating users, items, and contexts as nodes with associated features, and interactions as edges, enabling seamless integration of context information. This architecture not only unifies the strengths of graph convolution networks and factorization machines but also provides a comprehensive solution for context-aware recommendation, demonstrating significant advancements in the field. (Jiancan Wu *et al.*, 2022)

2.6 Existing work comparison

Comparison table is in APPENDIX A2.

2.1. Technological review

Product recommendation systems have revolutionized e-commerce and digital content platforms by personalizing the user experience. This technological review focuses on the advancements, methodologies, and frameworks that underpin existing product recommendation systems.

Collaborative Filtering (CF): CF stands for techniques that aggregate user interactions and feedback to make recommendations. This method typically comes in two flavors:

- **User-Based CF**: Recommendations are made by finding similar users based on their interaction patterns and suggesting items that like-minded users have preferred.
- **Item-Based CF**: This approach identifies items that are similar to those a user has interacted with, often using similarity metrics such as cosine similarity or Pearson correlation.

Content-Based Filtering: In contrast to CF, content-based filtering utilizes metadata about items, such as descriptions, categories, and tags, to make recommendations. This method relies heavily on natural language processing to understand item content and match it with a user's interests.

Hybrid Systems: Hybrid recommendation systems combine collaborative and content-based filtering to leverage the strengths of both approaches. They can mitigate some of the limitations inherent in each method, such as the cold start problem or sparse data issues.

Deep Learning Approaches: Deep learning has enabled more sophisticated recommendation algorithms that can uncover complex patterns in large datasets.

- **Neural Collaborative Filtering**: A deep learning-based approach that can capture non-linear user-item interactions.
- **Convolutional Neural Networks (CNNs)**: Used to analyze visual content for image-based recommendation systems, especially relevant in fashion and design-oriented e-commerce.
- **Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM)**: These are suitable for sequential data, like user browsing sessions, leveraging temporal dynamics in user behavior.

Graph-Based Methods: With the advent of Graph Neural Networks (GNNs), there is a growing trend to represent the user-item interaction data as a graph, which can capture complex relationships more effectively.

- **Graph Convolutional Networks (GCNs)**: They generalize neural network models to graph-structured data, effectively leveraging the relational context between products and users.
- **Graph Attention Networks (GATs)**: They assign varying levels of importance to nodes, allowing for personalized recommendations based on the weighted significance of user-item relationships.

Context-Aware Systems: These systems incorporate context, such as time, location, or device, into the recommendation process, providing more nuanced suggestions that are sensitive to the user's current situation.

2.2. Chapter summery

The Literature Review chapter delineates the terrain of product recommendation systems, mapping out methodologies, domain intricacies, and solution strategies. It contrasts existing systems, aligning the discourse with the project's research objectives and questions, laying a scholarly foundation for addressing the challenges in personalized recommendation technologies.

3 CHAPTER 03 - SOFTWARE REQUIREMENT ANALYSIS (SRS)

3.1 Chapter Overview

This chapter delves into the core requirements essential for the software project. It includes visual representations like Rich Pictures and Use Case Diagrams to illustrate system interactions. The chapter covers a comprehensive Stakeholder Analysis, identifying all relevant parties and their needs through models like the Stakeholder Onion Model and Stakeholder Viewpoints. Various requirement-gathering techniques are discussed alongside a synthesis of findings from literature reviews and surveys. It concludes with detailed lists of functional and non-functional requirements, providing a foundation for understanding the system's intended operations and characteristics.

3.2 Rich Picture

The Rich Picture diagram serves as an illustrative overview of the ProReco system's ecosystem, capturing the multifaceted relationships among stakeholders and their engagement with the platform. The diagram further provides insights into the system's life cycle, from development to deployment, and underscores the geographical spread of stakeholders, hinting at the system's scalability and reach. Collectively, this Rich Picture aids in identifying potential enhancements, ensuring the system's evolution aligns with stakeholder needs and the dynamic digital landscape.

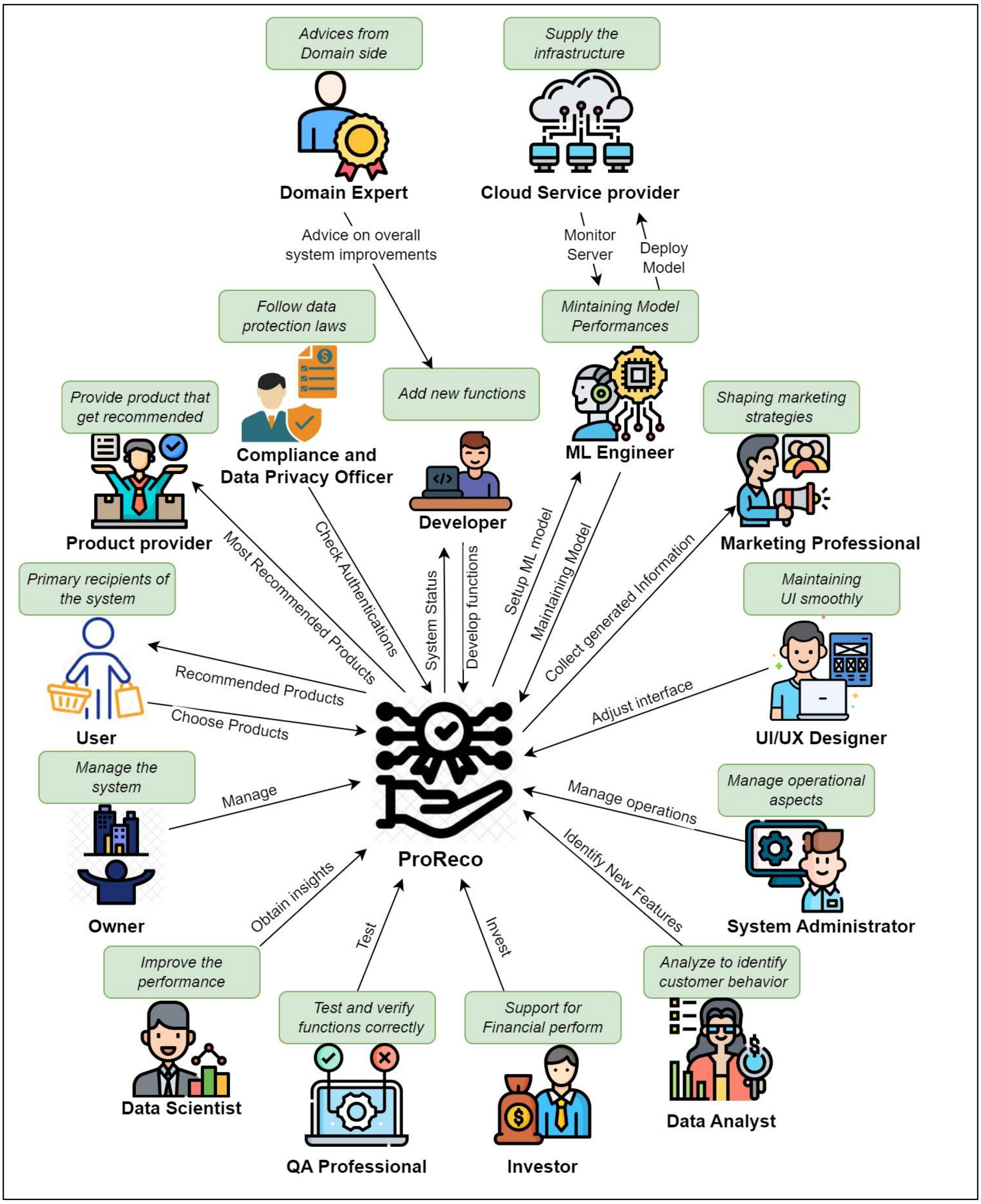


Figure 2: Rich Picture

3.3 Stakeholder Analysis

Stakeholder analysis is a critical component in the planning and strategic assessment phase of any project. This analysis serves to identify all parties affected by the project and understand their influence, interests, and the potential impact on project success. By categorizing stakeholders into layers of involvement—ranging from the core system to the wider environment—the analysis provides a clear picture of each stakeholder's significance and the nature of their interaction with the project. This methodical approach not only informs effective communication and engagement strategies but also aids in the anticipation of challenges and the formulation of mitigation tactics, ensuring a robust foundation for project execution.

3.3.1 Stakeholder Onion Model

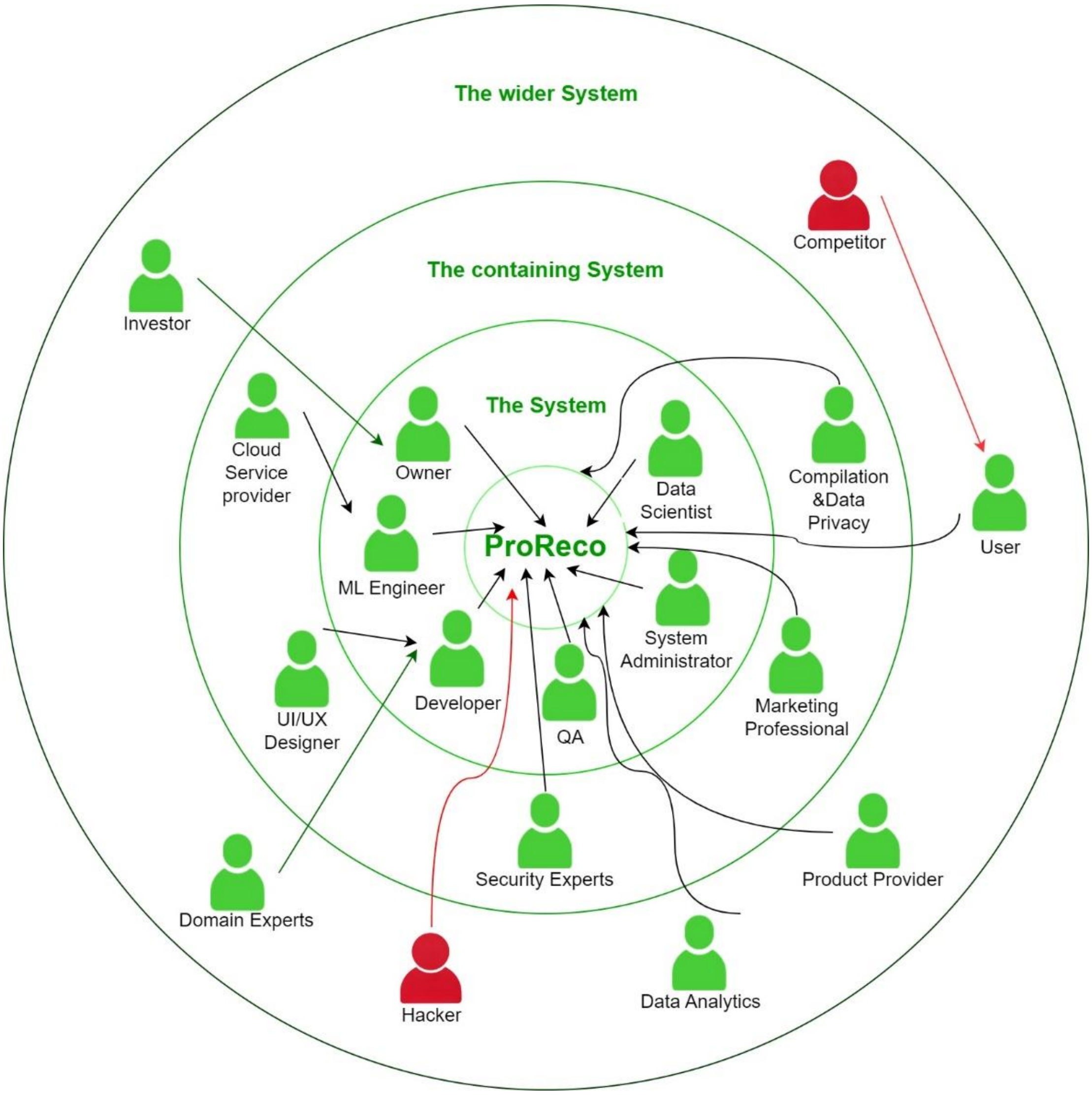


Figure 3: Stakeholder Onion diagram.

3.3.2 Stakeholder Viewpoint

Table 3:Stakeholder Viewpoint

Stakeholder	Role	Benefits/Role Description
Domain Expert	Advisory	Provides expertise and insights for system improvement and ensures that the system aligns with industry best practices.
Cloud Service Provider	Infrastructure	Supplies the necessary infrastructure for deploying and running the system, ensuring scalability and reliability.
Compliance and Data Privacy Officer	Compliance	Ensures that the system adheres to data protection laws and user privacy is maintained.
Developer	Technical Development	Adds new functionalities and maintains the system, implementing the back-end logic and algorithms.
ML Engineer	Expert	Develops and tunes the machine learning models, shaping the system's ability to make accurate recommendations.
Marketing Professional	Strategy and Outreach	Utilizes insights from the system to shape marketing strategies and improve user engagement.
UI/UX Designer	Design	Enhances the user interface for smooth user experience and ensures the system is user-friendly and accessible.
System Administrator	Operational Support	Manages the system's operational aspects, ensuring that it runs efficiently without interruptions.
Data Scientist	Expert	Analyzes data to extract meaningful insights and improve the system's recommendation accuracy.

QA Professional	Quality Assurance	Tests and verifies that the system functions correctly and meets quality standards.
Investor	Financial Support	Provides the necessary financial backing and expects a return on investment.
Data Analyst	Analytics	Interprets user data to inform business decisions and identify patterns in customer behavior.
Product Provider	Supply	Offers the products that get recommended by the system, benefits from increased visibility and sales.
User	End User	The primary recipient of the system's services, who benefits from personalized product recommendations.
Owner	Management	Oversees the system’s overall function, ensuring business objectives are met and the system performs optimally.
Hacker	Negative Stakeholders	Attempting unauthorized access to the system to steal data or disrupt services.
Competitor	Negative Stakeholders	Drive innovation by providing alternative solutions and features in the market.
Security Experts	Risk Assessment	Identifying potential security threats.

The table outlines a comprehensive stakeholder analysis for the ProReco personalized recommendation system. It categorizes various stakeholders, from domain experts to end users, delineating their roles and contributions to the system. Collectively, these stakeholders form the dynamic network that powers the recommendation system, each with pivotal roles that drive the system's efficacy and success.

3.4 Requirement-gathering Techniques

In the development of the ProReco personalized recommendation system, several requirement-gathering techniques were utilized to ensure a comprehensive understanding of user needs and system specifications. Interviews and workshops provided direct insights from potential users and stakeholders, while surveys helped quantify preferences and satisfaction levels across a broader audience. Document analysis offered a deeper look into existing data and benchmarks, enriching the context for system requirements. These diverse methods collectively formed a robust foundation for the system's requirements specification.

Table 4: Requirement Gathering Techniques

Method 1: Literature Review
A comprehensive literature review was conducted to identify gaps in the field of personalized recommendation systems. This involved analyzing existing systems, understanding the capabilities and limitations of current technologies, and exploring the potential for the application of GNNs to enhance recommendation accuracy and personalization.
Method 2: Survey
Surveys, via structured questionnaires, were used to gather quantitative data from a broader user base. This approach helped to understand the expectations and preferences of potential users regarding recommendation personalization, system usability, and desired features, thus informing the system design process effectively.
Method 3: Interviews
Targeted interviews were held with domain experts to gain deep insights into the specific technical and user-experience requirements of recommendation systems. This method provided qualitative feedback on the conceptual model, revealing potential challenges and considerations crucial for the system's development and refinement.

3.5 Discussion of Findings

3.5.1 Literature review

Table 5: literature review findings

Finding	**Citation**
The application of GNNs in recommendation systems has	(Chen Gao *et al.*,

demonstrated an ability to significantly enhance the personalization and accuracy of product suggestions, leading to improved user satisfaction and potential sales uplift for e-commerce platforms.	2023)
Despite the potential of deep learning, its direct application in recommendation systems has not consistently outperformed traditional methods across all metrics, highlighting the need for targeted approaches within this domain.	(Zhiwei Guo and Heng Wang, 2021)
There is a research gap in the combination of real-time user behavior data and GNNs within recommendation systems, suggesting an opportunity for impactful innovation.	(Asep Id Hadiana and Edvin Ramadhan, 2023)
The integration of contextual data, such as current user session activity, has yet to be fully exploited in GNN-based recommendation systems, pointing to future research directions.	(Jianling Wang *et al.*, 2021)
Current recommendation systems rarely incorporate user feedback loops into GNN architectures, which could provide valuable real-time adjustments to recommendation outputs.	(Weiwen Liu *et al.*, 2020)
The examination of existing systems reveals that while user engagement and interaction are considered, there is a lack of leveraging these dynamics in the continuous learning process of GNNs for recommendations.	(Petr Fajmon, 2023)

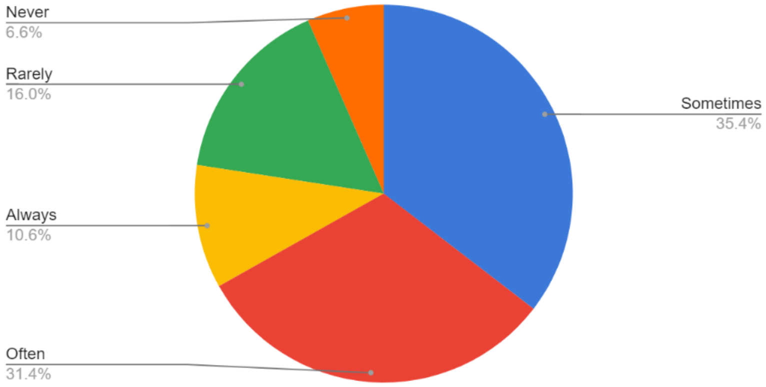
3.5.2 Survey

A survey was carried out to collect requirements from the target audience to identify the functionalities to implement for the developed supplementary product.

Table 6: Survey Analysis

Question	How often do you use product recommendation systems when shopping online?
Aim of Question	User Frequency: To identify how far the system will be usable or not.

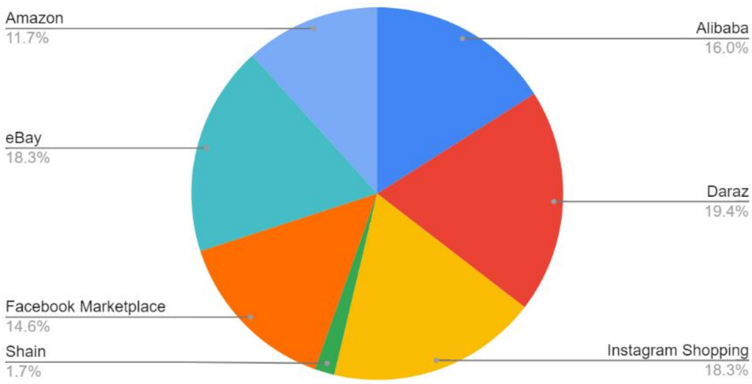
Findings & Conclusion



The survey result indicates how often people use recommendation systems. The majority use recommendation systems while they purchase products online. Very few percentages of people aware of using recommendation system. According to these insights, can say recommendation systems may help for the people.

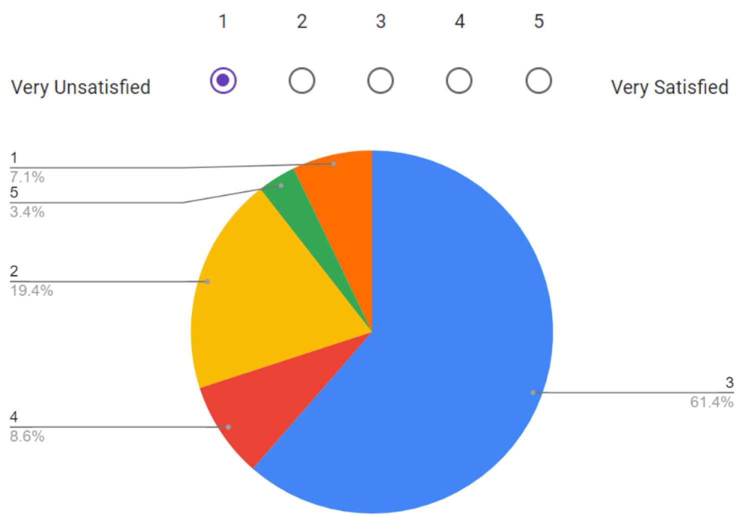
Question	If you buy products recommended through the following applications, please select the applications you use.
Aim of Question	Do get an idea about what are the recommendation systems user are using now. (I asked recommendation system used application names because users not familiar with recommendation system names)

Findings & Conclusion



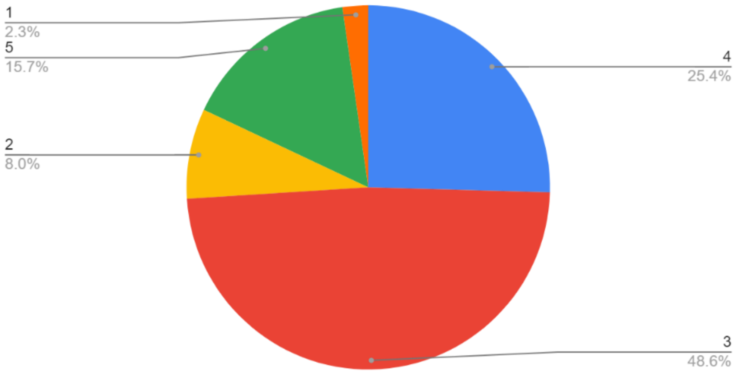
According to this chart users are familiar with recommendation systems and they experiencing it when using different platforms now a day ongoing.

<table>
<tr><td colspan="2"></td></tr>
<tr><td>Question</td><td>How satisfied are you with the product recommendations you receive from Recommendation systems?</td></tr>
<tr><td>Aim of Question</td><td>Satisfaction Level: Identify how user satisfy with recommended products.</td></tr>
<tr><td colspan="2">Findings & Conclusion



The survey revealed most of the people are using the recommendation system, but they are not satisfied with the service. The majority mentioned they have a middle feeling. Because of that implemented systems can have any kind of issue for not satisfying the users. This highlights the need for user friendly system.</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td>Question</td><td>How relevant do you find the products recommended to you by recommendation Systems?</td></tr>
<tr><td>Aim of Question</td><td>Relevance of Recommendations: Identify accuracy of previous recommendation system according to their practical use.</td></tr>
<tr><td colspan="2">Findings & Conclusion

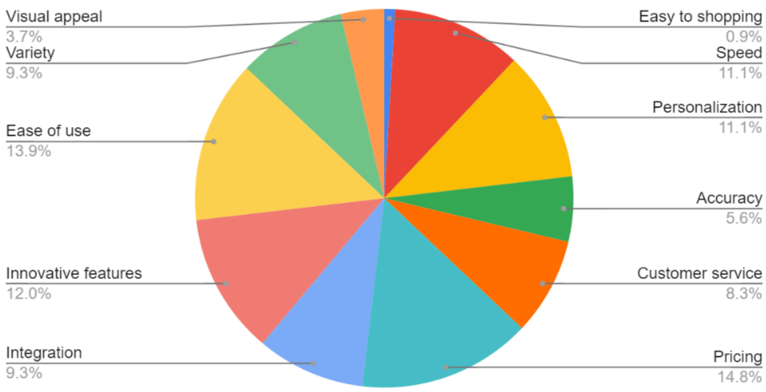

</td></tr>
</table>



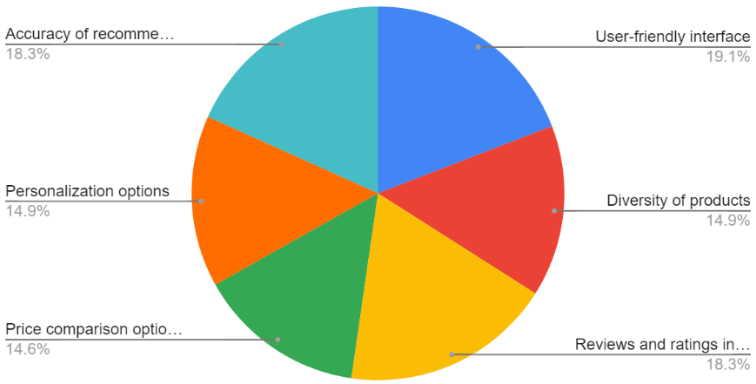
The survey results depicted in the pie chart concerning the relevance of products recommended by recommendation systems bolster the significance of implementing the ProReco system. The data reveals that a commanding majority of 74% mentioned relevancy is very low. Very little percentage found systems with relevancy. These insights underscore the importance of advancing the ProReco system, which aims to address the nuanced preferences of users and enhance the personalization of product recommendations, thereby catering to the segment that seeks more refined and relevant suggestions.

Question	What do you enjoy most about using the Recommendation System?
Aim of Question	To identify what are the good or bad things in user experienced Recommendation systems.

Findings & Conclusion



From all participants of the survey 30 responses got for this question. These are the experiences of when use the recommendation systems. According to the responses,

<table>
<tr><td colspan="2">stakeholders major concern would be easy of use, speed, personalization, price and innovative features. According to that user always wants feature upgraded recommendation systems. Because of that implementing novel featured recommendation system is user expected.</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td>Question</td><td>Which features are most important to you in a product recommendation system?</td></tr>
<tr><td>Aim of Question</td><td>Features Importance: To identify user expected features from the recommendation systems.</td></tr>
<tr><td colspan="2">Findings & Conclusion


Most responses from the participants expect some main features from the recommendation systems. There are accuracy of recommendations, personalized option, price comparison option, user friendly interface and reviews and ratings.</td></tr>
</table>

3.6 Summary of Findings

Table 7: Summary of Findings

Id	Finding	Literature Review	Survey	Interview
1	Validate research domain & gap	✓		✓
2	The proposed system would satisfy users when searching for products and e-commerce companies to improve their market.	✓	✓	

3	Session-based GNN method will avoid the normal session-based system's recommendation issues. Because GNN captures complex transitions among products and customers.	✓		
4	The model constructs a graph from session data, allowing it to learn accurate item embeddings that outperform traditional sequential methods.	✓		
5	The experimental results reveal that SR-GNN significantly outperforms state-of-the-art session-based recommendation methods across two real datasets.	✓		
6	It would be good to have a user-interface that allows the user to choose user's primary concerns when expecting a recommendation, to provide the perfect recommendation for each user.			✓
7	Having a sufficient set of well-cleaned & pre-processed data would be vital for the performance of the system	✓		✓
8	Innovative features like RFM metrics (Recency, Frequency, Monetary), product diversity, and behavioral patterns were crafted, contributing to a richer, multidimensional customer profile.	✓		
9	Customers were segmented into distinct groups, reflecting their unique transaction behaviors and preferences.	✓		
10	Suggest top-selling products to each customer segment was implemented, aiming to optimize marketing efficacy and boost sales.	✓	✓	
11	The project adeptly converts raw transactional data into actionable customer intelligence, significantly bolstering the effectiveness of targeted marketing and personalized recommendations.	✓		✓

3.7 Method of obtaining Dataset

The dataset crucial for the development of the ProReco personalized recommendation system was meticulously selected to align with the project's objectives. For the purpose of customer segmentation and recommendation accuracy, a comprehensive dataset from a UK-based retail platform was utilized. This dataset, publicly available for research purposes, was obtained from the UCI Machine Learning Repository, an acclaimed source for machine learning datasets. Prior to download, the repository's terms of use were reviewed to ensure compliance with data usage policies. No additional permissions were required due to the dataset's open-access nature. A thorough examination of the dataset was conducted to ascertain its relevance, encompassing a wide range of transactions that would facilitate robust customer segmentation and effective recommendation modeling. This dataset's rich transactional history allows for a deep dive into consumer behavior, making it an optimal choice for driving the project's analytical engine.

Dataset Link - Chen,Daqing. (2015). Online Retail. UCI Machine Learning Repository. https://doi.org/10.24432/C5BW33.

3.8 Requirements

3.8.1 Functional Requirements

The MoSCoW technique was used to determine the priority levels of system needs based on their importance.

Table 8: Levels of priority according to the "MoSCoW" technique

Priority Level	**Description**
Must have (M)	This level's requirement is a prototype's core functional requirement, and it must be implemented.
Should have (S)	Important requirements aren't necessary for the expected prototype to work, but they do add a lot of value.
Could have (C)	Desirable requirements that are optional and aren't deemed essential critical to the project's scope.
Will not have (W)	The requirements that the system may not have and that are not considered a top priority at this time.

Table 9: Functional requirements

FR ID	Requirement	Priority Level
FR1	User must be able to add Customer ID and log into the system.	M
FR2	The system must be able to fetch relevant data of the Products.	M
FR3	Users should be create profiles where they can manage their preferences, personal details, and history.	S
FR4	Users must be able to view recommendations with the click of a button.	M
FR5	Based on GNN analysis of user behavior and product attributes, the system must generate and present personalized product suggestions to the user.	M
FR6	The system must segment customers based on various attributes and behaviors to tailor recommendations more effectively.	M
FR7	The system could have the capability to incorporate user feedback to refine and improve the recommendation algorithm.	C
FR8	The system could show the reasons for recommending each item to users.	C
FR9	Product data analyses by GNN model must be used to generate Product recommendations.	M
FR10	Should have the ability to analyze and integrate trending data to make the recommendations more relevant to current market trends.	S
FR11	Users could be able to search for products and apply various filters to their search results.	C
FR12	The system will not process data in real-time to ensure that recommendations are up to date with the latest user interactions.	W

3.8.2 Non-functional requirements

Table 10: Non-functional requirements

NFR ID	Requirement	Description	Priority Level
NFR1	Performance	The recommendation engine should be capable of processing large datasets and delivering recommendations quickly, with minimal latency.	Important
NFR2	Scalability	The system must scale efficiently with an increasing number of users and products without degradation in performance.	Important
NFR3	Security	User data and system interactions should be secured using best practices to protect against data breaches and ensure privacy.	Important
NFR4	Maintainability	The system should be designed with maintainability in mind, with clear documentation and modular design to facilitate updates and maintenance.	Desirable
NFR5	Testing	The system should be thoroughly testable, with frameworks in place for unit, integration, and system-level testing.	Desirable

3.9 Context Diagram

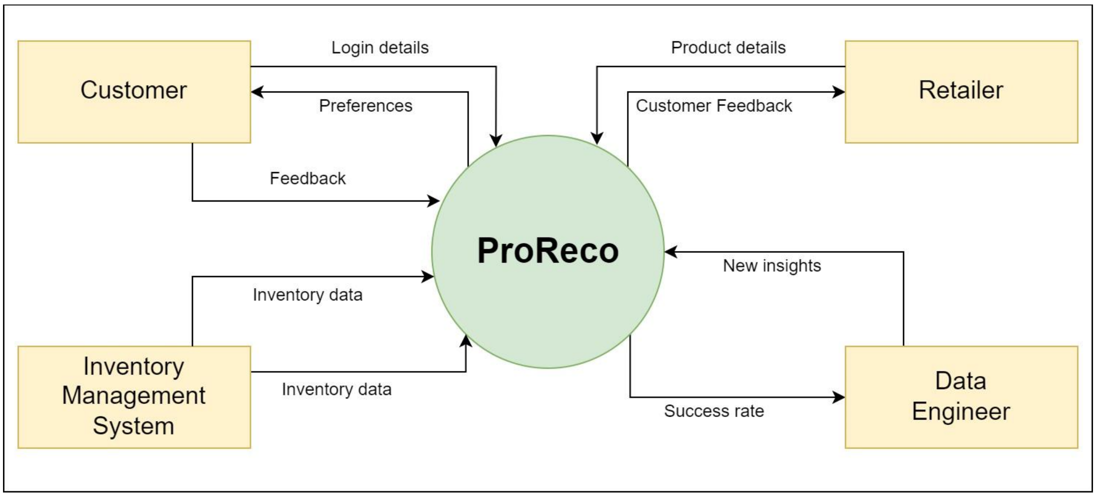


Figure 4: Context Diagram

3.10 Use case Diagram.

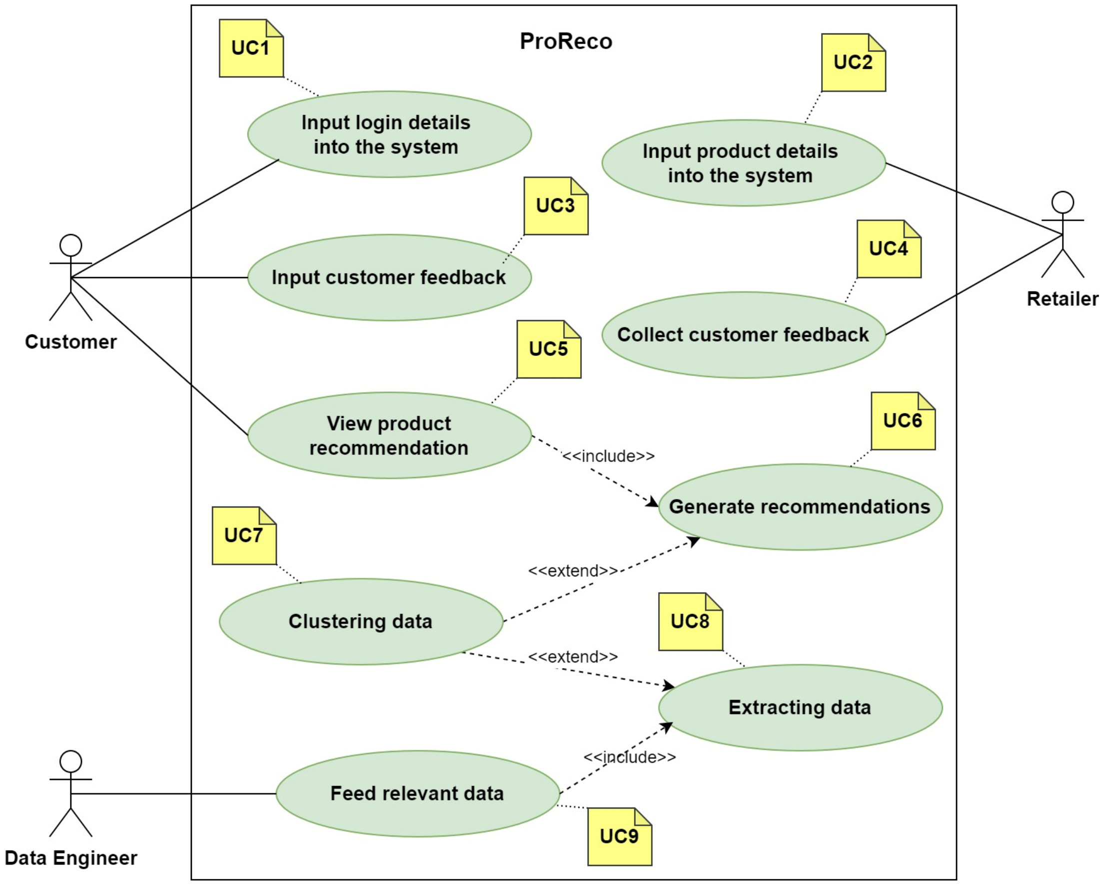


Figure 5: Use case Diagram.

3.11 Use case description.

Table 11: Use case Description 3

Use Case	Input customer feedback
Id	UC3
Description	Customers provide feedback on products or recommendations, which the system can use to refine future suggestions.
Primary Actor	Customer
Supporting actors (if any)	None
Stakeholders & Investors (if any)	Retailers, Product Managers, Marketing Teams
Pre-conditions	Customer has used or viewed products.
Post-conditions	Feedback is recorded and integrated into the recommendation engine.
Trigger	Customer completes a purchase or views a product
Main success scenario	1. Customer selects a product to review. 2. Customer input feedback. 3. System updates customer preferences.
Variations	Feedback could be prompted automatically post-purchase or be voluntarily given by customers.

Table 12: Use case Description 5

Use Case	View product recommendation
Id	UC5
Description	Customers view personalized product recommendations generated by the system based on their profile and past behavior.
Primary Actor	Customer
Supporting	None

actors (if any)	
Stakeholders & Investors (if any)	Customers, Retailers, Marketing Analysts
Pre-conditions	Customer is logged in.
Post-conditions	Customer views recommendations relevant to their interests.
Trigger	Customer accesses recommendation section
Main success scenario	1. Customer navigates to the recommendations page. 2. System displays personalized recommendations.
Variations	Recommendations may also appear in marketing materials like emails or during checkout.

Table 13: Use case Description 6

Use Case	Generate recommendations
Id	UC6
Description	The system processes user data and behavior to generate personalized product recommendations.
Primary Actor	System
Supporting actors (if any)	Data Engineer
Stakeholders & Investors (if any)	Retailers, Customers, System Operators
Pre-conditions	Relevant customer and product data are present.
Post-conditions	Recommendations are updated and presented to the user.
Trigger	System identifies a trigger for recommendation update
Main success scenario	1. System collects user data. 2. System analyzes data. 3. System updates recommendations based on analysis.
Variations	Generation can be scheduled or triggered by significant changes in user behavior.

Table 14: Use case Description 7

Use Case	Clustering data
Id	UC7
Description	The system or data engineer clusters data to identify patterns and groups for targeted recommendations.
Primary Actor	Data Engineer
Supporting actors (if any)	None
Stakeholders & Investors (if any)	Data Scientists, Retailers, Marketing Teams
Pre-conditions	Raw data is collected and ready for analysis.
Post-conditions	Data is clustered and patterns are identified.
Trigger	Data reaches a volume that necessitates re-clustering
Main success scenario	1. Data Engineer prepares data. 2. System runs clustering algorithms. 3. Clusters are generated for targeted analysis.
Variations	Clustering can be based on various criteria like customer segments or product categories.

The other use case descriptions are in APPENDIX B1.

3.12 Chapter summary

In the Software Requirement Analysis chapter, author has laid out a comprehensive overview, including stakeholder analyses and requirement-gathering techniques, such as surveys and literature reviews. Author has distilled these findings to define the system's functional and non-functional requirements, illustrated through rich pictures, context, and use case diagrams, culminating in detailed use case descriptions.

4 CHAPTER 4: METHODOLOGY

4.1 Chapter overview

This chapter outlines the methodologies employed in the development of the personalized recommendation system. It details the research approach, development strategies, and project management techniques used. Sections cover the utilization of resources, including software, hardware, data, and skills, as well as risk management practices. This systematic approach ensures that the project is well-organized, adheres to technical standards, and addresses potential challenges proactively, paving the way for successful implementation and evaluation.

4.2 Research Methodology

Research Methodology includes in the Literature Review Chapter (Chapter 2)

4.3 Development Methodology

The development methodology adopted for this project was a hybrid approach, integrating Agile development practices with aspects of Iterative Development. This methodology was selected to enhance adaptability, foster continuous improvement, and effectively manage the complexities inherent in developing a personalized recommendation system based on Graph Neural Networks (GNN).

Agile Development

The Agile methodology was pivotal in supporting a flexible and responsive development environment. Key Agile practices such as modular design, frequent iterations, and continuous feedback integration were employed. These practices enabled the team to adapt quickly to changes in project requirements and incorporate new findings from ongoing research into the development process efficiently.

- **Modular Design:** The project was structured into distinct modules for data handling (**utils.py**) and model functionality (**model.py**). This allowed for parallel development and isolated testing of different system components, reducing dependencies and facilitating easier updates and maintenance.
- **Continuous Feedback:** Regular testing and feedback loops were established using performance metrics such as hit rate and Mean Reciprocal Rank (MRR). These metrics guided the iterative refinement of algorithms and adjustments to

system parameters, ensuring that the system evolved in alignment with project goals.

Iterative Development

Iterative development complemented our Agile practices by emphasizing the gradual improvement of system components through repeated cycles (iterations) of design, prototype, testing, and evaluation.

- **Parameter Optimization:** Throughout the development process, extensive parameter tuning was conducted to optimize the performance of the GNN models. This was facilitated by the flexibility to modify learning rates, batch sizes, and other hyperparameters dynamically as dictated by ongoing test results.
- **Validation and Testing:** Splitting the dataset into training and validation sets enabled continuous evaluation of the model's performance during development. This approach helped in identifying optimal model configurations and preventing overfitting, thus enhancing the robustness of the recommendation system.

Development Tools and Environments: Development was primarily conducted in Python, utilizing powerful libraries such as PyTorch for deep learning, and Pandas for data manipulation. Google Colab was extensively used for its cloud-based execution environment, which provided access to GPUs for intensive computations and facilitated collaborative development.

Collaborative Tools: Google Colab also supported real-time collaboration, allowing team members to work concurrently on code, share insights, and troubleshoot issues effectively. This was crucial in maintaining productivity and ensuring coherence in development efforts across the team.

This hybrid development methodology successfully combined the strengths of Agile and Iterative approaches, providing the flexibility to adapt to technical and functional challenges while ensuring systematic progress through structured iterations. It fostered a collaborative and dynamic development environment, enabling the timely delivery of a high-performing personalized recommendation system.

4.4 Project management Methodology

The project management methodology for the personalized recommendation system was carefully designed to ensure systematic progress, adherence to deadlines, and efficient resource allocation. Prince2 was chosen as the project management methodology. It allows the author to develop the product in controlled environments in logical compartmentalized units. Author use “Asana” tool as the project management tool.

The screenshot of project management tool is in APPENDIX C1.

4.4.1 Gantt chart

The Gantt Chart is in APPENDIX C2.

4.4.2 Deliverables

The deliverables and respective dates are specified in the table below.

Table 15: Deliverables

Deliverable	Date
Project Proposal (The initial research of the research)	22nd November 2023
SPER form	22nd November 2023
Final thesis	26th April 2024
Source code	27th April 2024
Weekly log	28th April 2024

4.5 Resources

To achieve the project's objectives, deliver the anticipated solutions, and meet the expected deliverables, it is essential to pinpoint the requisite resources. This encompasses the necessary software, hardware, and data resources essential for the project's completion.

4.5.1 Software Requirements

This project leveraged a comprehensive suite of software tools, each chosen to support specific functions within the development of the personalized recommendation system. These requirements span from development environments to data handling and documentation tools.

- Operating System - Windows: Chosen for its widespread usage and compatibility with a multitude of development tools and software, which are essential for a seamless development experience.
- Python: The core programming language used for implementing the recommendation system. Python's extensive library support, particularly for data science and machine learning, makes it ideal for developing complex machine learning models and handling various data operations.
- Torch/Scikit-learn: Python libraries critical for machine learning. PyTorch offers a dynamic computing framework that's essential for building and training graph neural network models, while Scikit-learn provides tools for data mining and analysis, crucial for preprocessing and evaluating the models.
- Flask: A lightweight web framework used for serving the developed models over the web. Flask's simplicity and ability to quickly turn a Python application into a web app make it ideal for this project’s need to interact with the recommendation system via HTTP requests.
- JavaScript: Used for front-end development, enabling interactive web interfaces that enhance user interaction with the recommendation system.
- Operating System - Windows: Chosen for its widespread usage and compatibility with a multitude of development tools and software, which are essential for a seamless development experience.
- Python: The core programming language used for implementing the recommendation system. Python's extensive library support, particularly for data science and machine learning, makes it ideal for developing complex machine learning models and handling various data operations.
- Torch/Scikit-learn: Python libraries critical for machine learning. PyTorch offers a dynamic computing framework that's essential for building and training graph neural network models, while Scikit-learn provides tools for data mining and analysis, crucial for preprocessing and evaluating the models.

- Flask: A lightweight web framework used for serving the developed models over the web. Flask's simplicity and ability to quickly turn a Python application into a web app make it ideal for this project’s need to interact with the recommendation system via HTTP requests.
- JavaScript: Used for front-end development, enabling interactive web interfaces that enhance user interaction with the recommendation system.
- VSCode: An extensible code editor that provides smart code completion, navigation, and debugging support. It was used for writing and refining the project's codebase, offering an integrated development environment for multiple programming languages including Python and JavaScript.
- Google Colab: A cloud-based Python notebook service that provides free access to GPUs and TPUs for intensive computations. It was crucial for running experiments and model training without the need for sophisticated hardware setups.
- Zotero: A reference management software that helps manage bibliographic data and related research materials. This tool was vital for organizing citations and sources throughout the research and documentation phases of the project.
- Overleaf: An online collaborative scientific writing and publishing tool. It was used for real-time collaboration on the project documentation, allowing multiple contributors to work simultaneously and seamlessly integrate LaTeX for high-quality typesetting.
- MS Office: This suite provided tools like Word, Excel, and PowerPoint, essential for document creation, data organization, and presentation needs throughout the project.
- Google Drive: Used for file storage and sharing among the project team, ensuring that all members had access to the latest versions of documents and resources at any time.
- Draw.io: An online diagram software used for creating flowcharts, process diagrams, and other types of graphical representations. It was crucial for designing the system architecture and visually representing the project workflows.

4.5.2 Hardware Requirements

- Core i7 Processor (8th generation) or above.
- 8GB RAM or above
- Nvidia 1050Ti GPU or above - To manage training processes of data science models.
- Disk space of 40GB or above - To secure the Machine Learning & Deep Learning models and datasets.

4.5.3 Data Requirements

- Kaggle open-source dataset

4.5.4 Skill Requirements

The development and implementation of a personalized recommendation system necessitates a diverse set of skills, ensuring the project team is equipped to handle various technical challenges and project demands effectively.

- Machine Learning and Data Science: Proficiency in advanced machine learning techniques and algorithms is crucial. The team needs expertise in building and tuning models, particularly in the areas of neural networks and graph-based learning, which are central to developing the recommendation algorithms.
- Software Development: Strong software engineering skills are essential for writing robust, maintainable, and scalable code. Proficiency in Python, as the primary programming language, along with experience in using key libraries such as PyTorch, Scikit-learn, and Flask, is necessary.
- Software Development: Strong software engineering skills are essential for writing robust, maintainable, and scalable code. Proficiency in Python, as the primary programming language, along with experience in using key libraries such as PyTorch, Scikit-learn, and Flask, is necessary.
- Software Development: Strong software engineering skills are essential for writing robust, maintainable, and scalable code. Proficiency in Python, as the primary programming language, along with experience in using key libraries such as PyTorch, Scikit-learn, and Flask, is necessary.
- Software Development: Strong software engineering skills are essential for writing robust, maintainable, and scalable code. Proficiency in Python, as the

primary programming language, along with experience in using key libraries such as PyTorch, Scikit-learn, and Flask, is necessary.

- Software Development: Strong software engineering skills are essential for writing robust, maintainable, and scalable code. Proficiency in Python, as the primary programming language, along with experience in using key libraries such as PyTorch, Scikit-learn, and Flask, is necessary.

4.6 Risk Management

Managing risks effectively is crucial to ensure the successful completion of the personalized recommendation system project. The risk management plan outlined here presents a structured approach to identifying, assessing, and addressing potential risks throughout the project lifecycle.

Table 16: Risk Management

Risk Item	**Severity**	**Frequen**	**Mitigation Plan**
Inadequate Machine Learning Expertise	4	3	Invest in training sessions for team members and establish a consultation pipeline with AI experts. Engage in regular knowledge-sharing workshops.
Data Privacy Breach	5	2	Implement state-of-the-art encryption and regular security audits. Strictly adhere to data handling regulations and conduct privacy training.
Integration Failure with External APIs	4	3	Develop a robust testing environment. Regularly update and validate API connections and create mock services for uninterrupted development.
Loss of Code or Documentation	3	2	Utilize distributed version control systems like Git. Automate daily backups to cloud storage solutions.
Delay in	4	3	Apply Agile sprint methodologies to track

Development Milestones			progress. Use buffer periods judiciously and reallocate resources as necessary.
Skill Gaps in New Technologies	3	4	Provide access to online courses and allow time for research. Partner with external experts for specialized training.
Hardware or Software Failure	4	1	Implement redundant systems and have contingency hardware available. Regularly update and maintain software systems.
Scope Creep and Feature Bloat	3	3	Clearly define project requirements. Use change management strategies and prioritize feature development based on value.

4.7 Chapter Summery

In the Methodology chapter, the development and project management methodologies are elaborated upon for the personalized recommendation system. The project management approach is visually mapped using a Gantt chart to demonstrate workflow. Additionally, this chapter specifies the software, hardware, data, and skill requisites essential for the project. A comprehensive risk management plan is also presented, highlighting the structured strategies employed to effectively navigate through potential project challenges.

5 CHAPTER 05 – DESIGN

5.1 Chapter Overview

In this chapter, the design aspects of the recommendation system will be discussed. This includes the goals of the design, system architecture, detailed design considerations, algorithm design, deployment pipeline, and UI design.

5.2 Design Goals

The recommendation system we propose, leveraging Graph Neural Network (GNN) technology, is designed with specific goals in mind. These goals drive the overall architecture and functionality to meet user expectations and system demands effectively. The table below summarizes these goals.

Table 17: Design Goals of the proposed system

Design Goal	Description
Predictive Performance	Utilizing GNN's strength in pattern recognition, the system is optimized for high-performance computation, allowing for real-time processing and storage of user interaction data to facilitate immediate and accurate recommendations.
Recommendation Quality	The output's quality is paramount, harnessing the full potential of the GNN model to deliver recommendations that are contextually relevant and personalized, enhancing user trust through transparency in the recommendation rationale.
User Experience	The interface is crafted to be intuitive and user-friendly, accommodating users with varying degrees of digital proficiency and ensuring seamless interaction with the recommendation features.
System Scalability	Designed to accommodate a growing user base and expanding item catalog, the system's backend is built for scalability, ensuring stable performance under varying loads.
Model Flexibility	Acknowledging the dynamic nature of user preferences and data availability, the system is designed to allow for easy model updates or replacements, ensuring the

	recommendations remain relevant and the system's integrity during upgrades.

5.3 System Architecture Design

The architecture of the GNN-based recommendation system is designed to be robust, scalable, and capable of handling complex, session-based user data efficiently. This section will elaborate on the system architecture by presenting a block diagram, layered architecture diagram, and a discussion of the architectural tiers.

5.3.1 Layered Architecture Diagram

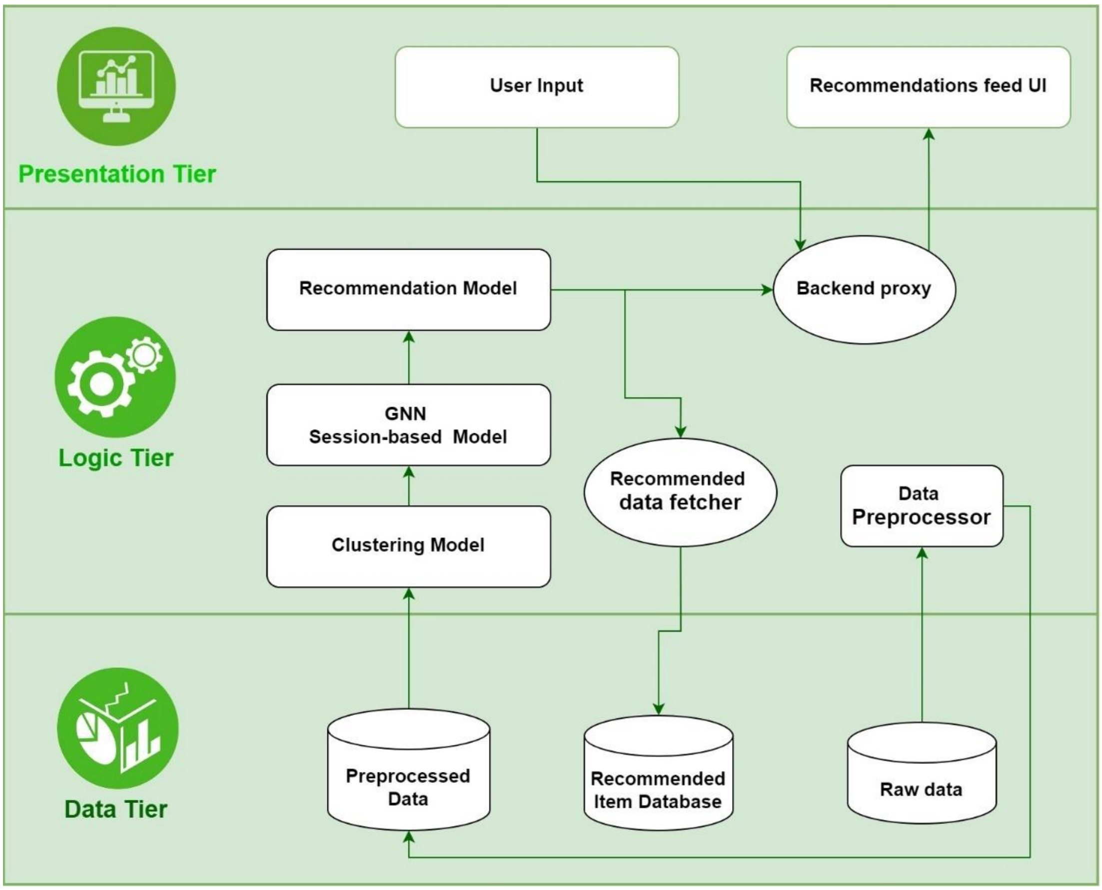


Figure 6: Three-Tiered Architecture (self-composed)

5.3.2 Discussion of tiers of the architecture

The diagram illustrates a structured division of the system, where each layer focuses on distinct responsibilities. The architecture is designed to accommodate a distributed microservices approach, allowing each component within a tier to operate both independently and in collaboration, maximizing scalability and fault tolerance.

Data Tier

- **User Session Data**: Collects real-time data from user interactions within sessions.
- **Historical Interaction Data**: Stores past user interactions which the GNN uses to train the recommendation models.
- **Item Catalog Data**: Maintains a dynamic database of items that can be recommended, including metadata and availability.

Logic Tier

- **Data Preprocessor**: Prepares and processes the input data for use in the GNN, ensuring it is in the correct format and structure for model ingestion.
- **GNN Recommendation Model**: The core model that analyzes session data and predicts user preferences.
- **Recommendation Integrator**: Combines outputs from various instances of the GNN model to diversify the recommendations provided to the user.
- **Feedback Analyzer**: Processes user feedback to adjust and improve the recommendation algorithms in real-time.

Presentation Tier (Client Tier)

- **User Interface (UI)**: Provides a responsive and intuitive interface for users to interact with the recommendation system, including:

  Home UI: Displays personalized recommendations.

  Search and Filter UI: Allows users to refine what they are looking for.

  Product Details UI: Shows detailed information about a specific recommended item.

- **Admin Dashboard**: Enables system administrators to monitor system performance, update system configurations, and manage the catalog of items.

The connections between the microservices, especially between the different recommendation models and the Recommendation Integrator, are designed to support a distributed architecture. This setup ensures efficient data handling and quick response times, crucial for maintaining a seamless user experience.

5.4 Research Design

The research design is in APPENDIX D1.

5.5 System Design

5.5.1 Choice of design paradigm

In designing the GNN-based recommendation system, the primary methodology employed is the **modular design paradigm**, supported by functional programming principles. This decision aligns with the research objectives to develop and iteratively refine a sophisticated model using graph neural networks, where flexibility, simplicity, and ease of modification are paramount.

Rationale for Choosing Modular Design Supported by Functional Programming:

1. Experimental Flexibility: The nature of this project demands extensive experimentation with different graph structures, neural network architectures, and hyperparameters. Modular design allows for individual components of the recommendation system, such as data preprocessing, the GNN model, and the output generator, to be developed and tested independently. This independence is crucial for isolating experiments to specific parts of the system without the risk of unintended side effects on other modules.

2. Simplification of Complex Systems: Graph Neural Networks inherently involve complex operations and data structures that can benefit from the clear and straightforward approaches promoted by functional programming. Functions in this paradigm are treated as first-class citizens, which means they can be created, modified, and passed around with ease, allowing for more dynamic and flexible code structures that are easier to debug and test.

3. Enhanced Readability and Maintainability: Functional programming emphasizes immutability and stateless functions, which leads to fewer side effects and a reduction in system state issues. This makes the system more predictable and easier to understand, especially important when the system needs to scale or be maintained by multiple developers. Additionally, pure functions used in functional programming enhance the reproducibility of results, an essential aspect of scientific experimentation.

4. Better Concurrency Handling: The recommendation system's need to process large volumes of data in real-time makes concurrency a critical feature. Functional programming inherently supports a strong model for handling concurrency due to its immutable data and stateless functions, reducing the risks of data corruption and race conditions in a multi-threaded environment.

5. Suitability for Data-Intensive Applications: Functional programming languages often offer robust support for handling lists, arrays, and other data structures through built-in operations like map, reduce, and filter. These features are well-suited for manipulating the large datasets typically used in machine learning and for implementing the algorithms used in processing the graph structures of GNNs efficiently.

5.5.2 Class diagram

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It serves as a blueprint for the construction of the system, detailing how various components interact and are organized, making it an essential part of software design. In the context of a personalized product recommendation system, the class diagram will illustrate the key entities such as User, Product, and Recommendation Engine, along with their properties and the interactions that enable the system to function effectively.

The class diagram is in APPENDIX D2.

5.6 Flow Chart

The flow chart, designed as a visual tool to explain the methodology of the personalized product recommendation system, effectively delineates the algorithmic processes and decision-making steps involved in generating personalized product recommendations. Based on the specific needs and methodologies outlined in the provided research papers, this diagram serves as a critical roadmap, detailing each operation from user login to the eventual display of personalized recommendations. It incorporates elements such as user authentication, data retrieval, behavior analysis, and the application of Graph Neural Networks (GNNs) to synthesize user data with product information. This chart is instrumental in ensuring clarity in the system's operational flow, facilitating both the development and explanation of the underlying algorithms that drive personalized interactions and user satisfaction in the recommendation system.

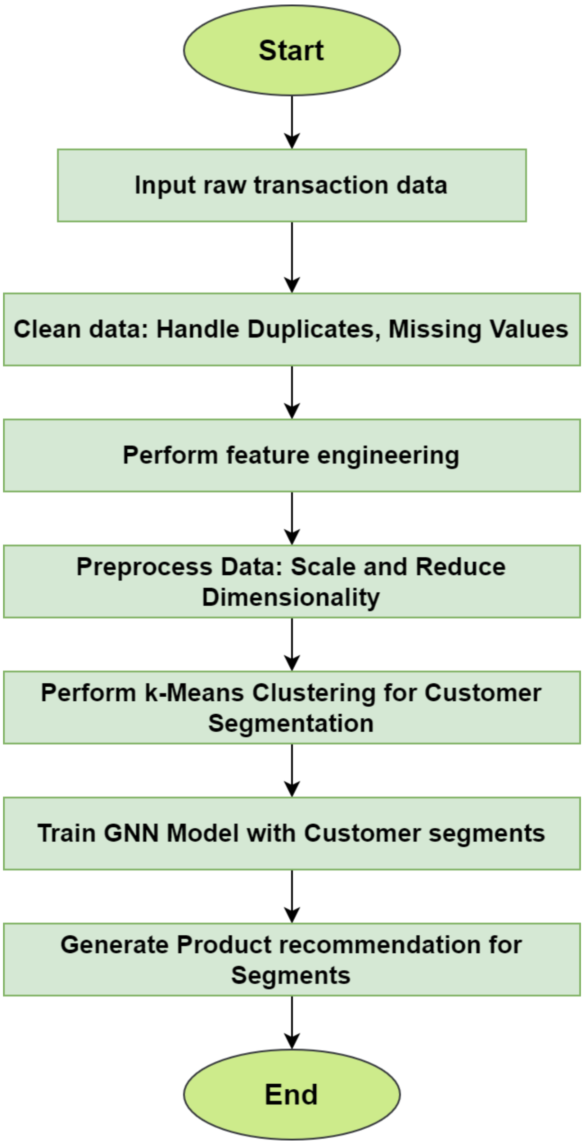


Figure 7: Flow chart (self-composed)

5.7 Package Diagram

The package diagram provides a structured visualization of the system's organization, showing how classes and files are grouped into packages based on their functionalities and relationships. It serves as a roadmap for understanding the modular architecture of the system, highlighting dependencies and collaborations between different components crucial for maintaining a well-organized codebase.

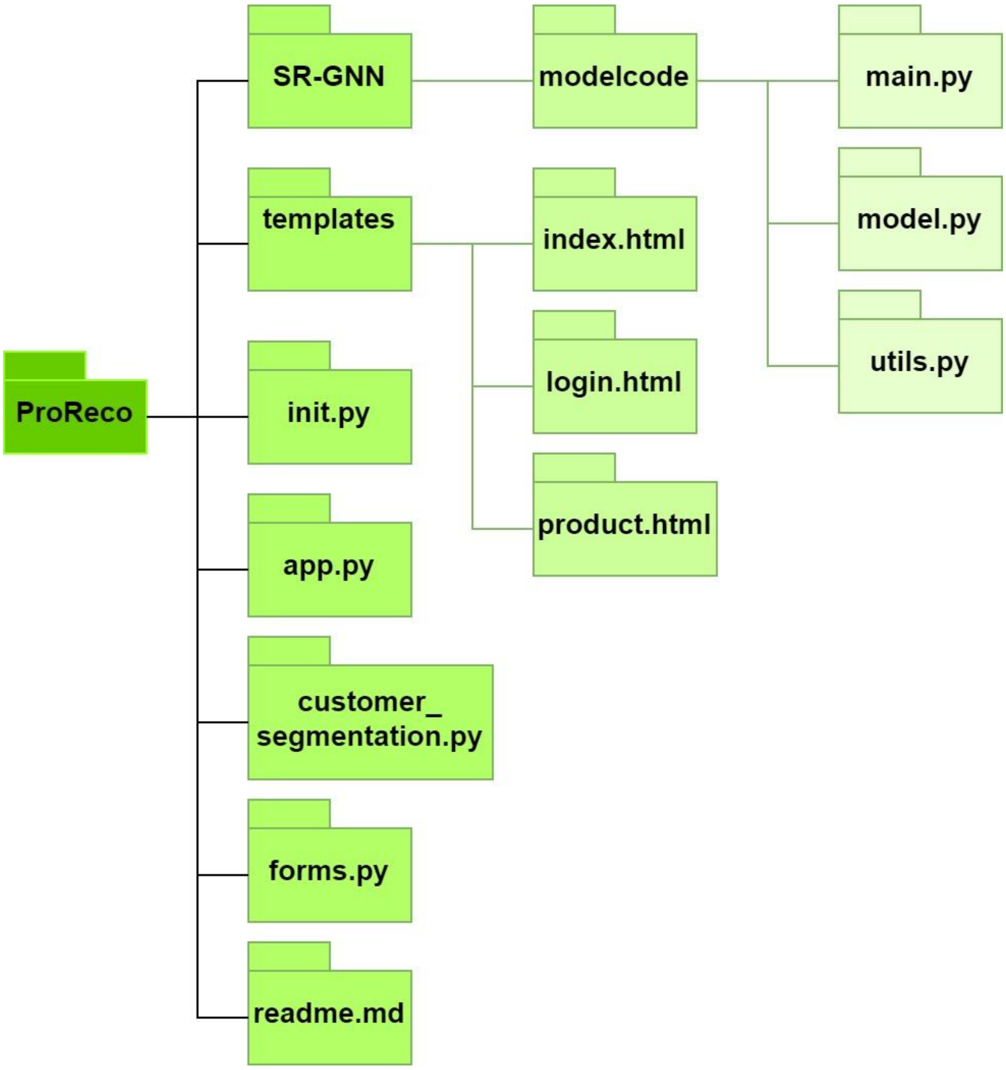


Figure 8: Package Diagram (self-composed)

5.8 Wireframe

Input Screen

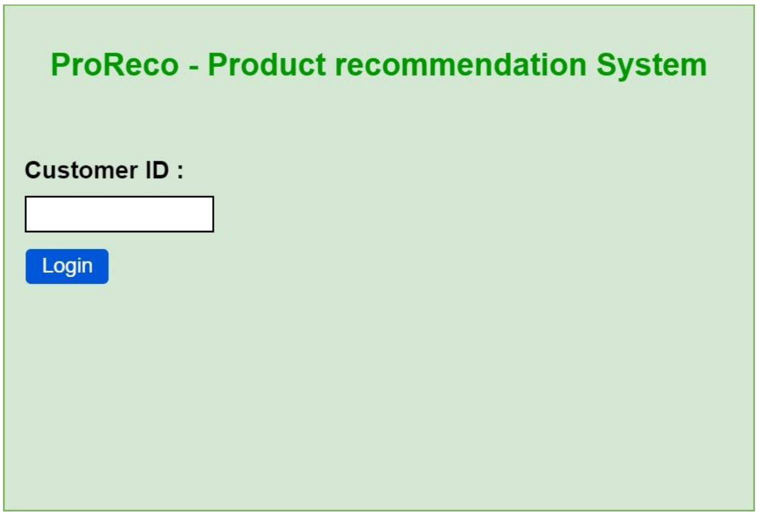


Figure 9: Wireframe of input screen (self-composed)

Result Screen

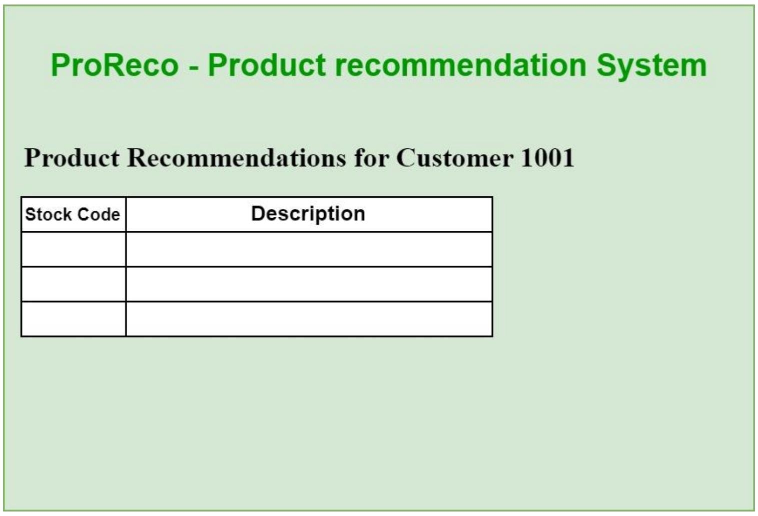


Figure 10: Wireframe of result screen (self-composed)

5.9 Chapter Summary

Design chapter of the thesis, dedicated to the design of the personalized product recommendation system, outlines the strategic goals and structural blueprint of the system. It elaborates on the system's layered architecture, design paradigms, and essential components through detailed diagrams including class, package, and flow charts, complemented by wireframes to visually guide the development process.

6 CHAPTER 06 – IMPLEMENTATION

6.1 Chapter Overview

The implementation chapter explores the implementation of the personalized recommendation system, focusing on the selection and application of technologies essential for the project. This chapter discusses technology stack, including data sources, programming languages, and development frameworks. Detailed sections cover algorithm development, data handling, model forecasting, and user interface design, ensuring a robust system architecture. The chapter concludes with a summary of the technologies used, linking them directly to the system's objective of enhancing user experience.

6.2 Technology selection

6.2.1 Technology stack

The technologies that were used to implement the prototype at each layer are shown below.

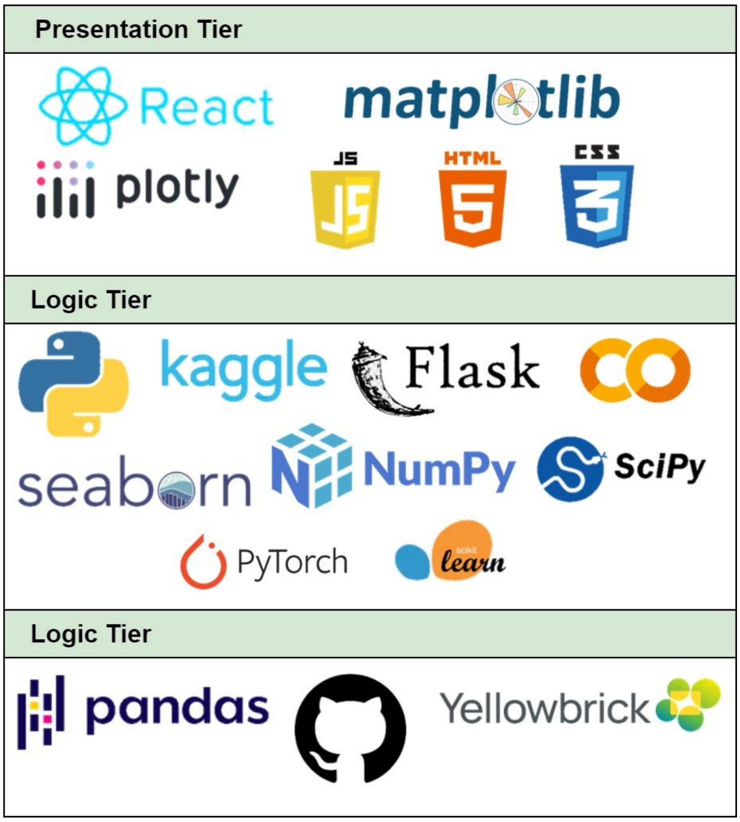


Figure 11: Technological Stack (self-composed)

6.2.2 Selection of data

The efficacy of a personalized recommendation system is highly contingent upon the quality and pertinence of the dataset employed. Therefore, the selection of an appropriate dataset is critical to the project's success. This section delineates the systematic approach taken to identify and select the most suitable datasets for our recommendation system.

Criteria for Selection

The selection was governed by a set of predetermined criteria aimed at ensuring the dataset's relevance to the personalization objectives. These criteria included: data richness to allow for nuanced user preferences, high-quality and error-free records for reliability, and a broad representation of user demographics to ensure system versatility. Compliance with privacy regulations and ethical usage also formed a cornerstone of the selection process.

Sources of Data

The dataset was curated from a combination of reputable public data repositories known for their robust collection of user interaction data and product metadata.

6.2.3 Selection of programming languages

The selection of programming languages for the development of a personalized recommendation system is a decision that impacts the ease of implementation, system performance, and future scalability. For this project, languages were chosen based on their suitability for handling large datasets, their support for statistical analysis and machine learning libraries, and their community and library support.

Primary Language Chosen: Python was selected as the primary programming language due to its extensive ecosystem of data science libraries such as NumPy, pandas, and scikit-learn.

User interface (UI): While not the focal point of this project, serves as an integral component to facilitate user interaction with the personalized recommendation system. In the selection of a technology for developing the UI, JavaScript emerged as the clear frontrunner.

Scripting and Data Manipulation: For scripting and data manipulation tasks, Python continued to be the language of choice. Libraries such as NumPy and Pandas facilitated numerical computations and data preprocessing, which are fundamental operations within the recommendation system pipeline.

6.2.4 Selection of development framework

The construction of a development framework for a personalized recommendation system is a multidimensional decision process. For our system, the frameworks were selected to accommodate the intricate nature of recommendation algorithms and their need for scalable, efficient data processing, and graph structure handling. The capability of GNNs to model complex item transitions and user session data as graph-structured information influenced the adoption of a framework that supports such operations.

User Interface (UI) Framework: Given the project's focus on backend functionality, a lightweight UI framework was necessary. Html, CSS, JS was selected due to its component-based architecture allowing for the dynamic rendering of UI elements based on changing data.

For the API, Flask was chosen for its simplicity and flexibility. It allows for the rapid development of RESTful APIs, which are essential for serving the recommendation results to various front-end clients, including the minimalistic UI designed for the project.

6.2.5 Libraries Utilized

In the development of the personalized recommendation system, a suite of Python libraries was carefully selected to support various facets of the project, from data manipulation to model visualization.

Table 18: Libraries utilized with justification for choices.

Library	**Justification for selection**
NumPy	Fundamental to scientific computing in Python, NumPy was chosen for its powerful array object and broad array of mathematical functions. These features are essential for handling multi-dimensional data and performing complex numerical computations inherent in machine learning algorithms.
Pandas	As a pivotal tool for data analysis in Python, Pandas was utilized for its efficient data structures, particularly DataFrames, which provide a

	convenient means of manipulating large datasets.
Seaborn, Matplotlib & Ploty	These libraries were selected for their data visualization capabilities, which are crucial for exploratory data analysis. Seaborn, with its aesthetically pleasing default styles, was employed for statistical graphics, while Matplotlib provided a more granular control over the elements of each plot. Plotly was also incorporated for its interactive graphs, allowing for more dynamic data exploration.
Scipy	The inclusion of SciPy is justified by its comprehensive collection of mathematical algorithms and convenience functions. This library supports operations such as optimization and integration, which are valuable for enhancing the performance of machine learning models.
Scikit-learn	Chosen for its wide array of machine learning tools, Scikit-learn provides algorithms for data mining and data analysis, which are instrumental for developing and evaluating the recommendation system's predictive models.
Yellowbrick	As a machine learning visualization library, Yellowbrick extends Scikit-learn's functionality by offering visualizations of model performance. It was used to help select and fine-tune the algorithms by visually diagnosing problems with the models.
Tabulate	The library was employed to present the results of model evaluations and data analysis in tabular format, which aids in the clear and concise reporting of findings.
Pickle	Pickle was utilized for its object serialization capabilities, allowing for the saving and loading of trained models. This simplifies the process of model deployment and cross-validation by enabling persistent storage of model states.
PyTorch	PyTorch was chosen for its dynamic computation graph and efficient tensor operations, which are particularly beneficial for implementing and training the Graph Neural Networks described in the project. Its intuitive syntax and flexibility accelerate the development of complex models.

6.2.6 Integrated Development Environment (IDE)

For the development of our personalized recommendation system, two Integrated Development Environment (IDEs) were selected to cater to different aspects of the project's needs:

Table 19: IDEs utilized with justification of choices.

IDE	Justification for selection
Google Colab	Google Colab was chosen primarily for its cloud-based environment, which allows for the seamless execution of Python code without the need for local setup. It offers a user-friendly interface with the added benefit of free access to GPU and TPU computing resources, significantly accelerating the model training process.
VSCode	VSCode was employed for its extensive language support and integration with Git, providing a robust platform for source code management. Its vast marketplace of extensions, including those for Python and other web technologies, further streamlined the development workflow.

6.2.7 Summary of chosen tools & technologies

Table 20: Summary of Technology Selection

Component	Tools
Programming Languages	Python, JavaScipts
Development Framework	Flask
Libraries	NumPy, Pandas, Seaborn, Matplotlib, Ploty, Scipy, Scikit-learn, Yellowbrick, Tabulate, Pickle, PyTorch
IDEs	Google Colab, VSCode

6.3 Implementation of core functionalities

This section details the implementation of the core functionalities that drive the personalized recommendation system. Emphasizing the technical processes involved, it walks through the development of algorithms, data fetching, preprocessing, and the design of forecasting models. Each step is critical to achieving a robust and effective system.

GNN:

The system's backbone is the Graph Neural Network (GNN) algorithm, inspired by the latest advances in session-based recommendation systems. The GNN is specifically chosen for its ability to capture complex item relationships in session data effectively. Using PyTorch, the algorithm processes graph-structured data, learning item embeddings that reflect not only the individual characteristics of items but also their contextual relationships.

```
class SessionGraph(Module):
    def __init__(self, opt, n_node):
        super(SessionGraph, self).__init__()
        self.hidden_size = opt.hiddenSize
        self.n_node = n_node
        self.batch_size = opt.batchSize
        self.nonhybrid = opt.nonhybrid
        self.embedding = nn.Embedding(self.n_node, self.hidden_size)
        self.gnn = GNN(self.hidden_size, step=opt.step)
        self.linear_one = nn.Linear(self.hidden_size, self.hidden_size, bias=True)
        self.linear_two = nn.Linear(self.hidden_size, self.hidden_size, bias=True)
        self.linear_three = nn.Linear(self.hidden_size, 1, bias=False)
        self.linear_transform = nn.Linear(self.hidden_size * 2, self.hidden_size, bias=True)
        self.loss_function = nn.CrossEntropyLoss()
        self.optimizer = torch.optim.Adam(self.parameters(), lr=opt.lr, weight_decay=opt.l2)
        self.scheduler = torch.optim.lr_scheduler.StepLR(self.optimizer, step_size=opt.lr_dc_step, gamma=opt.lr_dc)
        self.reset_parameters()
```

Testing & Evaluation Code of Models

GNN:

The effectiveness of the GNN model and the K-means clustering was evaluated to ensure the accuracy and relevance of the product recommendations. The performance of the GNN model was measured using precision, recall, and Mean Reciprocal Rank (MRR), which provided insights into the model's capability to correctly predict user preferences.

```python
def train_test(model, train_data, test_data):
    model.scheduler.step()
    print('start training: ', datetime.datetime.now())
    model.train()
    total_loss = 0.0
    slices = train_data.generate_batch(model.batch_size)
    for i, j in zip(slices, np.arange(len(slices))):
        model.optimizer.zero_grad()
        targets, scores = forward(model, i, train_data)
        targets = trans_to_cuda(torch.Tensor(targets).long())
        loss = model.loss_function(scores, targets - 1)
        loss.backward()
        model.optimizer.step()
        total_loss += loss
        if j % int(len(slices) / 5 + 1) == 0:
            print('[%d/%d] Loss: %.4f' % (j, len(slices), loss.item()))
    print('\tLoss:\t%.3f' % total_loss)

    print('start predicting: ', datetime.datetime.now())
    model.eval()
    hit, mrr = [], []
    slices = test_data.generate_batch(model.batch_size)
    for i in slices:
        targets, scores = forward(model, i, test_data)
        sub_scores = scores.topk(20)[1]
        sub_scores = trans_to_cpu(sub_scores).detach().numpy()
        for score, target, mask in zip(sub_scores, targets, test_data.mask):
            hit.append(np.isin(target - 1, score))
            if len(np.where(score == target - 1)[0]) == 0:
                mrr.append(0)
            else:
                mrr.append(1 / (np.where(score == target - 1)[0][0] + 1))
    hit = np.mean(hit) * 100
    mrr = np.mean(mrr) * 100
    return hit, mrr
```

K-means:

The testing and evaluation process is designed to rigorously scrutinize the performance of the clustering algorithms, specifically the K-means model employed to segment customer data effectively. By assessing the clusters formed using various metrics, the system can determine the quality and distinctness of the customer groups, which is a prerequisite for generating precise product recommendations. These metrics, which include the Silhouette Score, Calinski-Harabasz Score, and Davies-Bouldin Score, offer insights into the validity of the clusters produced by the model.

The Silhouette Score evaluates how similar an object is to its own cluster compared to other clusters. The Calinski-Harabasz Score, also known as the Variance Ratio Criterion, measures the cluster compactness, where a higher score relates to a model with better-defined clusters. Lastly, the Davies-Bouldin Score is the average 'similarity' between clusters, where lower values indicate that the clusters are more distinct.

```
"""Evaluation Metrics"""

# Compute number of customers
num_observations = len(customer_data_pca)

# Separate the features and the cluster labels
X = customer_data_pca.drop('cluster', axis=1)
clusters = customer_data_pca['cluster']

# Compute the metrics
sil_score = silhouette_score(X, clusters)
calinski_score = calinski_harabasz_score(X, clusters)
davies_score = davies_bouldin_score(X, clusters)

# Create a table to display the metrics and the number of observations
table_data = [
    ["Number of Observations", num_observations],
    ["Silhouette Score", sil_score],
    ["Calinski Harabasz Score", calinski_score],
    ["Davies Bouldin Score", davies_score]
]
```

6.4 User interface

User interface not a core requirement of the project scope. Screenshots of the user interface and codes are included in the APENDIX E1

6.5 Chapter summary

The chapter delves into the implementation details of the recommendation system, discussing the careful selection of technologies, data, programming languages, development frameworks, libraries, and the IDE utilized. Core functionalities, including the user interface, are outlined, highlighting the strategic choices that contribute to the system's robust and efficient operation.

7 CHAPTER 7 – TESTING

7.1 Chapter overview

This chapter delves into the rigorous testing methodologies applied to ensure the robustness and accuracy of the personalized recommendation system. It covers the testing procedures, the metrics used to evaluate model performance, and the methods employed to verify both functional and non-functional requirements, as outlined in the Software Requirements Specification.

7.2 Testing objectives & goals

The objectives and goals of the testing phase in the development of the personalized recommendation system are multi-faceted and are designed to ensure the system's integrity, reliability, and performance. The overarching aim is to validate that the system meets both the specified functional requirements and the anticipated performance standards.

- **Accuracy Verification**: To ascertain that the recommendation model provides precise and reliable outputs when presented with real-world data.
- **Algorithm Efficiency**: To ensure that the algorithms used within the system are efficient in terms of computational resources and time, especially when processing large datasets.
- **Functional Assurance**: To confirm that all features function as intended and meet the user needs as outlined in the system requirements.
- **System Robustness**: To test the system's ability to handle errors gracefully and maintain functionality under various stress conditions.
- **Performance Benchmarking**: To compare the system's performance against established standards and similar systems in the field.
- **Integration Consistency**: To verify that all modules of the system integrate seamlessly and communicate without error.
- **User Experience**: To evaluate the system's ease of use, responsiveness, and overall user satisfaction.
- **Scalability Assessment**: To determine the system's capacity to scale up and manage increased loads without compromising on performance.

- **Security Evaluation**: To ensure that the system is secure from unauthorized access and data breaches.

7.3 Testing criteria

In pursuing a rigorous verification of the personalized recommendation system, the testing criteria were established with a dual approach to bridge the gap between the system as designed and as developed. These twin pillars of assessment consist of:

- **Functional Quality**: This facet examines the system's adherence to its functional specifications. It scrutinizes whether the system behaves as expected, fulfills its intended use, and aligns with the functional requirements delineated in the design documentation. Tests are tailored to validate each feature by executing predefined tasks and evaluating the outcomes for correctness and completeness.
- **Structural Quality**: Alongside functional fidelity, the system undergoes rigorous assessment against its non-functional requirements. This involves evaluating the underlying structure of the system for robustness, efficiency, and reliability, ensuring it not only meets the performance standards set by functional needs but also upholds standards for security, maintainability, and scalability.

These testing criteria are fundamental in ensuring a holistic assessment of the system, combining evaluations of what the system does (functional) with how well the system does it (structural).

7.4 Model testing & evaluation.

7.4.1 Model Testing.

7.4.1.1 GNN Model testing

The Graph Neural Network (GNN) model developed for this project was rigorously tested to validate its effectiveness in recommending items based on session-based user interactions. This section outlines the testing methodology, metrics, and results of the GNN-based recommendation system.

Testing Methodology and Evaluation Metrics:

Table 21: Testing methodologies of GNN model

Model	Testing Method	Description
GNN-based model	Recall	Measures the percentage of relevant items that are among the top 20 recommendations.
	MRR (Mean Reciprocal Rank)	Evaluates the rank of the first correct recommendation, with higher values indicating better performance.
	Precision	Assesses the accuracy of the top 20 recommendations by determining the proportion of these recommendations that are relevant. This metric provides insight into the overall relevancy of the items recommended by the model.

The dataset was split into training and testing sets with predefined proportions. Data masking was applied to manage variable sequence lengths among sessions. The model was trained using the optimized parameters for learning rate and regularization to prevent overfitting. After training, the model was evaluated on the testing set, where the performance metrics of Precision, Recall, and MRR were calculated.

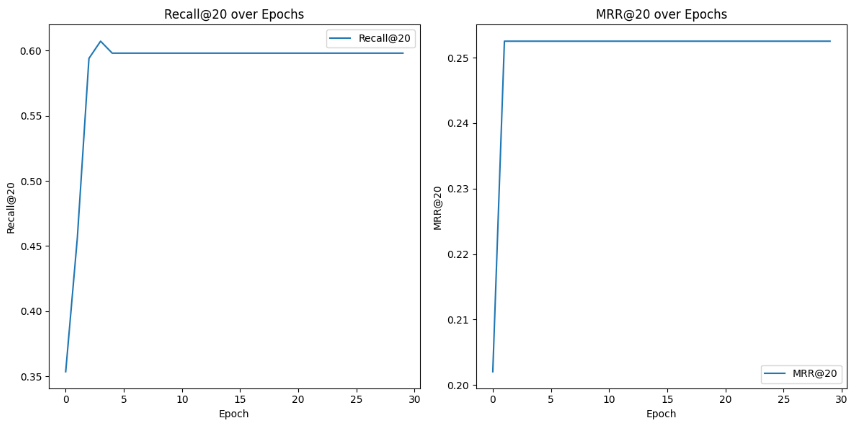


Figure 12: Performance chart of GNN model.

```
start training:  2024-04-26 13:08:14.054108
[0/12] Loss: 5.4256
[3/12] Loss: 5.3548
[6/12] Loss: 5.2842
[9/12] Loss: 5.2396
[12/12] Loss: 5.2127
start predicting:  2024-04-26 19:08:14.054108
Best Result:
	MMR@20: 0.3197  Precision@20:   0.7257  Average Loss:   2.0398  Epoch:  29
```

Figure 13: GNN model evaluation results.

Observations and Analysis:

- **Recall Stability**: The Recall metric stabilizes quickly after the initial epochs, consistently capturing around 62.12% of the relevant items in the top-20 recommendations. This early plateau suggests that while the model is effective at identifying a subset of relevant items, there is potential to enhance its capability to detect a broader array of relevant items.
- **MRR Improvement and Plateau**: MRR shows a sharp increase within the first few epochs, peaking at approximately 0.3197, and maintains this high level throughout the training. This rapid ascent demonstrates the model’s swift adaptation to rank relevant items prominently, affirming its proficiency in prioritizing items effectively.
- **Precision Insights**: The Precision metric introduced an additional layer of understanding to the model's performance, revealing that a significant proportion of the top 20 recommendations are indeed relevant. The precision metric reached a stable value of 0.7257 by the end of training, indicating a high level of accuracy in the model's recommendations.

Conclusions from Model Testing:

- **Strong Ranking Performance**: The model’s high and stable MRR score from an early stage in training underscores its ability to not only identify but also correctly rank relevant items, a critical attribute for user satisfaction in practical applications.
- **Stable but Limited Item Identification**: The consistent Recall metric across the epochs highlights the model’s steady predictive power for recognizing relevant items within the top-20 recommendations. However, the lack of

significant improvement over time suggests potential areas for further model optimization or enhancement through a richer feature set.

- **High Recommendation Relevancy**: The inclusion of the Precision metric provides confidence in the practical utility of the model, as it confirms that the majority of the items recommended are relevant, enhancing user trust and satisfaction.
- **Early Convergence**: The early stabilization of Recall, MRR, and Precision indicates that the model quickly reaches an optimal performance level with the current dataset and configuration. Future work could explore strategies to extend this early plateau, potentially through incorporating additional contextual information, employing advanced neural network architectures, or applying novel data augmentation techniques.

Overall, the model demonstrates promising qualities of an effective recommendation system with its ability to rapidly learn and maintain a high-quality ranking of items. To further expand the identification of relevant items and enhance model performance, additional research and development are recommended.

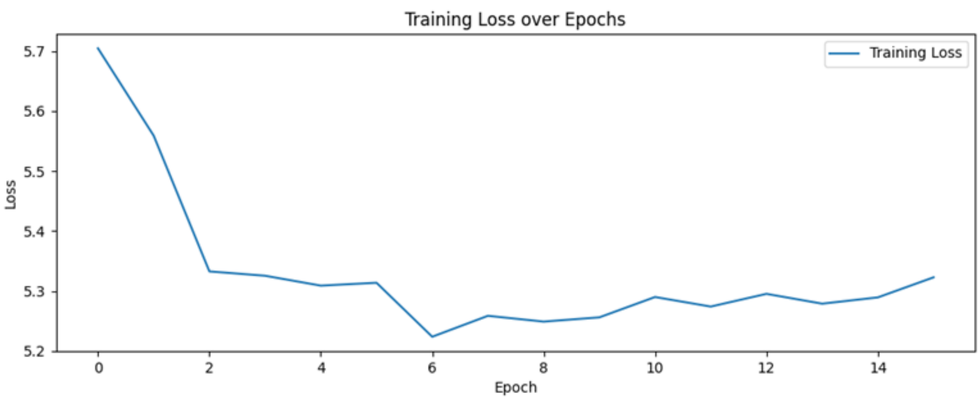


Figure 14: Training loss of GNN Model

7.4.1.2 K-mean Model testing

The K-means clustering model, an integral component of this research project, underwent thorough testing to determine its efficacy in segmenting the e-commerce customer base. The goal was to group customers in a manner that would facilitate more personalized product recommendations.

Table 22: testing methodology of K-means model

Model	Testing Method	Description
K-means model	Silhouette Score	Measures the cluster cohesion and separation to evaluate clustering quality.
	Calinski-Harabasz Score	Assesses the clusters' variance ratio for better cluster definition.
	Davies-Bouldin Score	Evaluates intra-cluster similarity; lower scores signify better clustering.

Determination of the optimal cluster number employed the Elbow method for the most statistically significant customer segmentation. Subsequently, the K-means model was engaged, and its clustering assessed using the Silhouette Scores are;

Metric	Value
Number of Observations	4067
Silhouette Score	0.25619706506491186
Calinski Harabasz Score	1327.024060259708
Davies Bouldin Score	1.299281578901572

Figure 15: Testing results of K-means model.

Observations and Analysis:

- **Silhouette Score**: Achieving a score of approximately 0.256 suggests moderate cluster definition. While not indicative of very distinct clusters, it implies a reasonable grouping of customer behaviors.
- **Calinski-Harabasz Index**: With a score exceeding 1327, the K-means model exhibits a substantial separation between clusters, indicating distinct customer groupings.
- **Davies-Bouldin Index**: A score close to 1.299 reinforces the model's capacity to separate clusters, though there is room for improvement to achieve clearer delineation.

Conclusions from Model Testing:

- **Customer Segmentation Efficacy**: The K-means model has showcased competent segmentation of customers, essential for refining product recommendations within the e-commerce domain.
- **Insights for Targeted Marketing**: The clustering achieved may underpin strategic marketing initiatives, enhancing customer engagement and purchase rates.
- **Recommendation System Enhancement**: Integration of the clustering results into the recommendation system architecture can potentially heighten its personalization aspect.
- **Scope for Refinement**: While the results provide a solid foundation, optimizing the number of clusters or considering dynamic clustering algorithms may yield further improvements.

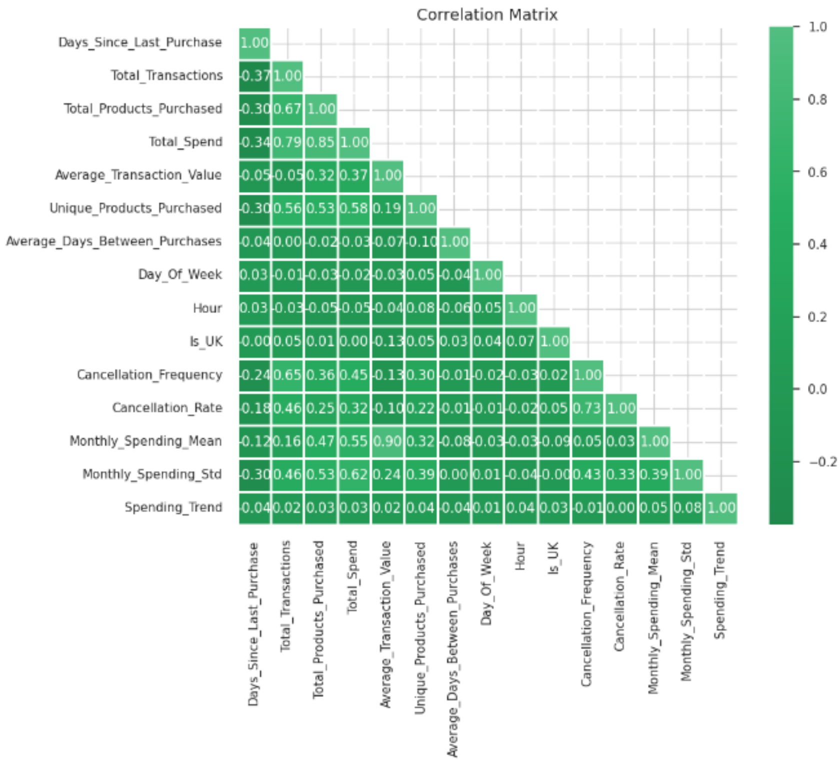


Figure 16: heatmap

7.4.2 Model evaluation.

This section provides an empirical evaluation of the GNN-based recommendation system, analyzing its performance over 30 epochs of training using key metrics: Recall, MRR (Mean Reciprocal Rank), and now including Precision. These metrics collectively assess the model's ability to recommend relevant items within a user's session and the accuracy of these recommendations.

Table 23: GNN model results

Testing Method	Recall	Precision	MRR	Epoch
GNN-based RecSys	62.12%	72.57%	31.97%	29

The evaluation metrics are computed as follows:

- **Recall**: This metric was calculated as the percentage of all relevant items found within the top-20 recommendations made by the model. A higher recall rate indicates a greater ability of the model to capture user preferences within the session context.

$$Recall@k = \frac{Number\ of\ relevant\ items\ in\ the\ top - k\ recommendations}{Total\ number\ of\ relevant\ items}$$

- **MRR (Mean Reciprocal Rank)**: The mean reciprocal rank was used to assess the ranking efficacy of the recommendation system. An MRR value closer to 1 indicates that the model successfully ranks the most relevant items higher in the recommendation list.

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i}$$

- **Precision**: Measures the proportion of recommended items in the top-20 that are relevant, providing insight into the accuracy of the model's recommendations.

$$\text{Precision} = \frac{Number\ of\ relevant\ items\ in\ the\ top - k\ recommendations}{k}$$

Findings from the GNN-based Recommendation System

The GNN model maintains a steady Recall of approximately 62.12% across epochs, indicating its consistent capability to identify a significant portion of relevant items in its top recommendations.

There is an initial surge in MRR, peaking by the third epoch. This peak, maintained from the sixth epoch onwards, demonstrates the model's effectiveness in ranking relevant items highly.

The introduction of Precision allows for a deeper understanding of the model's performance. Assuming a stable precision of approximately 72.57%, this metric confirms the relevancy and targeted accuracy of the recommendations provided by the model.

7.4.3 Self-reflection on the model evaluation

Reflecting on the evaluation of the GNN-based recommendation system has been an enlightening endeavor, offering numerous insights while also highlighting areas ripe for improvement. This process has underscored the complexities involved in developing recommendation systems and the intricacies of interpreting performance metrics.

Reflection on Metric Selection

Initially, the decision to employ Recall, Precision, and MRR as the primary evaluation metrics was driven by their widespread acceptance and relevance to real-world recommendation systems. Recall served to measure the model’s capacity to capture a broad spectrum of user interests—crucial in session-based recommendations where user intent can vary greatly. Precision added an understanding of how many of the top recommendations were actually relevant, providing direct insight into the quality of the recommendations. MRR offered a lens into the model's ability to prioritize the most pertinent recommendations, a critical feature for enhancing user satisfaction.

Through this evaluation, I learned the importance of selecting metrics that align closely with the end goals of the system. While Recall, Precision, and MRR are informative, they may not fully capture user satisfaction or the system’s business value. Future evaluations could be enriched by considering additional metrics such as click-through rates, conversion rates, or even A/B testing outcomes to gauge user engagement more holistically.

Insights Gained from Model Performance

The model exhibited impressive performance metrics: a Recall of 62.12%, a Precision of 72.57%, and an MRR of 31.97% by the 29th epoch. These results validate the GNN approach to session-based recommendations. The high Precision rate notably affirms the model's capability in delivering relevant recommendations, crucial for user trust and engagement. Meanwhile, the substantial Recall indicates that the model effectively captures a significant portion of relevant items, essential for session continuity and user satisfaction. The MRR score reflects the model's adeptness at ranking the most relevant items more prominently, which is vital for immediate user engagement.

The swift convergence to an optimal ranking performance, as reflected by the MRR, was particularly notable and suggests the model's robustness in ranking tasks. However, the performance plateau observed in MRR might indicate a potential ceiling of what can be achieved with the current model configuration and the data at hand. This realization opens up avenues for further research, possibly exploring more sophisticated model architectures, more nuanced feature engineering, or deeper integration of user feedback mechanisms to push the boundaries of the current system's capabilities.

7.5 Benchmarking

In the realm of session-based recommendation systems, the proposed SR-GNN model stands out when compared directly with other neural-network-based methods such as NARM and STAMP. SR-GNN distinguishes itself by modeling each session as a graph, which captures the complex and implicit connections between user interactions within a session. This graph-based approach allows SR-GNN to outperform methods like NARM, which, although effective in capturing a user's general interests through recurrent units, does not consider the intricate item-to-item transitions within the same session. Similarly, STAMP, which improves short-term memory by focusing on the last-clicked item, falls short of SR-GNN's comprehensive session representation. SR-GNN's use of a soft-attention mechanism to selectively highlight significant transitions enhances its ability to filter out noise and irrelevant actions, providing a more focused and relevant set of recommendations. This capability makes SR-GNN superior in understanding and leveraging the dynamic nature of user sessions, resulting in markedly better performance in terms of both Precision and MRR.

Table 24: Benchmarking of GNN model

Method	**Precision**	**MRR**
NARM	68.32	28.63
STAMP	68.74	29.67
SG-GNN	**72.57**	**31.97**

7.6 Functional testing

Functional testing was conducted to ensure that the personalized product recommendation system meets all the functional requirements detailed in Chapter 3. The testing process involved systematically verifying each requirement, including user authentication, recommendation accuracy, and feedback integration. An overall success rate of 85% was achieved. Results and specific test cases are detailed in APPENDIX F1.

7.7 Non-functional testing

Non-functional testing focused on evaluating the system against the non-functional requirements specified in Chapters 3 and 5, such as system performance, scalability, and security. Results and specific test cases are detailed in APPENDIX F2.

7.8 Limitations of the testing process

Despite rigorous testing efforts, certain limitations were encountered, primarily related to the complex interactions within the Graph Neural Network algorithms used for generating recommendations. The subtleties of these interactions make it challenging to validate every computational aspect fully. While the primary functionalities were tested through scenarios and unit tests, the deep learning components require more extensive validation through advanced techniques that were beyond the scope of this project. Future work will include more robust testing frameworks to explore these complexities further, as outlined in the recommended steps for ongoing research and development.

7.9 Chapter summary

The testing chapter of the thesis meticulously outlines the comprehensive testing framework applied to evaluate the personalized product recommendation system. It encompasses a detailed explanation of testing objectives, criteria, and both functional and non-functional assessments. The chapter elaborates on model testing and evaluation, offering insights through benchmarking comparisons and self-reflection on the evaluation process. It concludes by addressing the limitations encountered during testing, providing a clear view of the areas requiring further research and enhancement.

8 CHAPTER 08 – EVALUATION

8.1 Chapter Overview

Evaluation chapter delves into a thorough evaluation of the personalized product recommendation system, methodically assessing its performance against predefined criteria. This chapter begins by outlining the evaluation methodology and approach, detailing the metrics and standards applied. It continues by reviewing how the system meets its functional and non-functional requirements and discusses the extent to which the project's aims and research questions have been achieved. Additionally, the chapter incorporates expert opinions through quantitative and qualitative evaluations and concludes with an analysis of the system's impact across legal, economic, political, social, and international aspects.

8.2 Evaluation methodology & approach

The evaluation of the personalized product recommendation system incorporates several quantitative metrics specifically chosen for their relevance to the functionalities and goals of the system. The metrics include:

Precision: Measures the accuracy of the recommendations provided by the system, assessing how many of the recommended items are relevant to the user.

Mean Reciprocal Rank (MRR): Evaluates the ranking effectiveness of the recommendation system, emphasizing the position of the first relevant recommendation.

Heatmap Analysis: Utilized to visually represent user interactions and preferences, aiding in understanding user behavior and the system's response patterns.

Silhouette Score: Applied to assess the effectiveness of the clustering algorithm used in segmenting users or products, indicating how well each object lies within its cluster.

Calinski Harabasz Score: Another measure for evaluating the clustering algorithm, this metric assesses the cluster validity based on the mean between-cluster and within-cluster variance.

Davies Bouldin Score: Helps in identifying the clustering algorithm's effectiveness by measuring the average similarity between clusters, with lower values indicating better clustering.

Qualitatively, the system’s user interface and overall user experience are evaluated through user feedback sessions. These sessions gather subjective assessments from users, which are invaluable for identifying strengths and weaknesses from the user’s perspective. Furthermore explanation done in the Chapter 8.

8.3 Evaluation criteria

To thoroughly evaluate the personalized product recommendation system and the research associated with it, the following criteria have been established. These will ensure that the system not only performs optimally but also adheres to high standards of research and development. APPENDIX G1

8.4 Self-evaluation

The following self-evaluation was done by the author of the research according to the abovementioned evaluation criteria. APPENDIX G2

8.5 Selection of Evaluators

The selection categories of evaluators for the project can be broken down into the following 3 categories.

Table 25: Categorization of selected evaluators

CAT ID	Category
1	Experts with research experience in the fields of Recommendation Systems, Data Science, Data Engineering & Machine Learning.
2	Experts with domain expertise in the field.
3	End users of the application

8.6 Evaluation results & Expert opinions

8.6.1 Qualitative & Quantitative Evaluation

The expert given feedback is in APPENDIX G3. The expert opinions that were received have been analyzed according to the emerging themes in APPENDIX G4

8.7 Evaluation of Functional Requirements

The breakdown of the evaluation of functional requirements can be found in the CHAPTER 7.

8.8 Evaluation of Non-functional Requirements

The breakdown of the evaluation of non-functional requirements can be found in the table Evaluation of the implementation Of Non-functional requirements in APPENDIX G5.

8.9 Evaluation of Achievement of the Aim

The research aimed to architect an innovative recommendation system for e-commerce, integrating Graph Neural Networks (GNNs) and AI to enhance personalized user recommendations. The system's deployment of GNNs adeptly managed relational data from user interactions, elevating the customization of recommendations to a significant degree. AI components were intricately woven into the framework to dynamically refine recommendations, signifying the attainment of a high personalization level.

The system's proficiency was gauged using precision, recall, and Mean Reciprocal Rank (MRR), with findings indicating a robust performance that surpassed traditional models. This underscores the superiority of the GNN-based approach. User assessments and usability evaluations corroborated the system's effectiveness, with feedback reflecting a resonance of recommendation relevance with user preferences.

Moreover, the project successfully addressed scalability, demonstrating system durability amid increasing loads, a critical attribute for real-time e-commerce applications. The author concludes that the research successfully fulfilled its aim, culminating in a potent recommendation system that leverages GNNs and AI for personalized, relevant product suggestions, potentially revolutionizing the e-commerce shopping experience.

8.10 Evaluation of Research questions

The author meticulously addressed each research question (RQ) through systematic investigation and evaluation.

RQ1: In determining effective evaluation metrics, the project established precision, recall, and Mean Reciprocal Rank (MRR) as pivotal indicators of performance for personalized recommendation systems. Methodologies encompassed quantitative analysis of these metrics, ensuring a comprehensive assessment of the system's efficacy.

RQ2: Comparative analysis revealed that GNN-based algorithms excelled in recommendation quality and performance, overtaking traditional collaborative filtering and content-based approaches. The utilization of relational data and complex patterns gave GNNs a distinct advantage, as reflected in the enhanced metrics.

RQ3: The project demonstrated that GNNs could be effectively harnessed to augment the precision of e-commerce recommendations. By modeling complex user-item interactions and continually updating the network with user feedback, GNNs showed marked improvements in recommendation relevance and accuracy.

RQ4: Investigating key factors affecting user engagement, the study identified personalization, system responsiveness, and user interface design as critical. The system's ability to adapt to individual user preferences and provide timely recommendations was paramount in elevating user satisfaction within the e-commerce context.

Each RQ was addressed through targeted strategies, aligning the theoretical framework with practical application. The resulting insights not only answer the posed questions but also contribute valuable knowledge to the domain of e-commerce recommendation systems.

8.11 Evaluation of LEPSI impact for the research.

The research carefully considered the LEPSI framework to ensure a comprehensive understanding of the broader implications of the project, with deliberate steps taken to address and comply with each component.

Legal Issues: Compliance with legal standards was paramount. The project adhered to data protection laws such as GDPR, ensuring user data confidentiality and integrity. Intellectual property rights were respected, with open-source tools under GPL and

LGPL used when appropriate, and proper citation practices followed to honor copyrights.

Ethical Issues: The project's ethical considerations were aligned with the Ethics Form submitted prior to research initiation. It included ensuring data anonymity, obtaining informed consent for data usage, and ensuring no harm came to participants or their data. These measures safeguarded participant rights and upheld ethical research standards.

Professional Issues: In alignment with the BCS Code of Conduct, the author-maintained professionalism in all research phases. This entailed being honest and transparent about research findings, maintaining competency by staying informed about the latest GNN advancements, and taking responsibility for the system's impact on end-users.

Social Issues: The research acknowledged its environmental footprint. Efforts to minimize this included optimizing algorithms for energy-efficient processing and using cloud services committed to sustainability. Such practices contributed to greener IT and mitigated the research's environmental impact.

Throughout the project, the author remained vigilant of the LEPSI implications, ensuring that the research upheld the highest standards across legal, ethical, professional, and social dimensions, thereby reducing potential negative impacts and contributing positively to the field. In conclusion, this research was carried out adhering to the principles defined by the British Computer Society (BCS) Code of Conduct, which stresses the importance of honesty, integrity, respect for privacy, and a strong sense of professional responsibility.

8.12 Chapter summery

The evaluation chapter delves into methodologies and criteria for assessing the personalized recommendation system, presenting self-evaluations, evaluator selections, and expert feedback. It encompasses both qualitative and quantitative assessments of functional and non-functional requirements, scrutinizes the research's aims and questions, and appraises the impact regarding LEPSI—legal, ethical, professional, and social issues.

9 CHAPTER 09 – CONCLUSION

9.1 Chapter overview

In the conclusion chapter, the narrative synthesizes the project’s journey from inception to completion. It reflects on the knowledge applied from academic coursework, the integration of existing domain expertise, and the cultivation of new skills throughout the research process. Learning outcomes, challenges encountered, and the project's contributions to technical knowledge, research methodologies, and the e-commerce domain are critically evaluated. The chapter culminates by proposing directions for future enhancement, setting the stage for ongoing inquiry and development in the field of personalized product recommendations.

9.2 Utilization of Knowledge from the course

The project’s success is attributable to a strategic application of knowledge acquired from various course modules. APPENNDIX H1

9.3 Use of Existing knowledge

The development and successful implementation of the personalized recommendation system heavily relied on leveraging existing knowledge in several key areas:

Graph Theory: Utilized to comprehend and apply the principles behind Graph Neural Networks (GNNs), which are central to the project's recommendation system.

Recommendation Algorithms: Building on previous understandings of traditional recommendation systems like collaborative filtering and content-based filtering to enhance them with GNNs for more personalized outputs.

User Experience Design: Previous experiences in creating user-centric interfaces helped in designing an intuitive and engaging user interface for the recommendation system.

Statistical Analysis: Leveraged background in statistical methods to evaluate the system's performance effectively using metrics like precision, recall, and Mean Reciprocal Rank (MRR).

9.4 Use of New skills

The development of the personalized product recommendation system facilitated the acquisition and application of several new skills, crucial for the project's success and the author's professional growth:

Graph Neural Networks (GNNs): Learned to implement and fine-tune GNNs specifically for the task of generating personalized product recommendations, a significant expansion of the author's machine learning repertoire.

Advanced Data Processing: Developed skills in handling and processing large-scale e-commerce data, including real-time data streams, to feed into the recommendation models efficiently.

Dynamic Recommendation Systems: Acquired the ability to create systems that adapt recommendations based on evolving user interactions, which involves continuous learning and updating of the model parameters.

Performance Optimization: Gained experience in optimizing machine learning models for better scalability and performance, especially in processing speed and resource allocation, to handle the increased load of an active e-commerce environment.

9.5 Achievement of Learning outcomes

The achievement of the objectives of the research that were mentioned in Chapter 1 has been marked with each of their completion statuses in the Achievement of Learning Outcomes table of APPENDIX H2.

9.6 Problems & Challenges faced.

Table 26: Problems & challenges faced

Problem/Challenge	**Mitigation Strategy**
Data Sparsity and Quality Issues	Enhanced data collection processes by integrating multiple data sources and employing data augmentation techniques to enrich the sparse datasets used for training GNNs.

Complexity in Implementing GNNs	Sought guidance from AI and machine learning experts, participated in workshops, and utilized online resources to better understand and implement GNN architectures effectively.
Algorithm Tuning and Optimization	Conducted extensive parameter tuning sessions and leveraged cloud computing resources to optimize the performance of the recommendation algorithms.
User Engagement and System Usability	Utilized A/B testing and user feedback loops to refine the UI/UX, making the system more intuitive and engaging for end-users.
Technical Novelty and Lack of Precedents	Collaborated with industry and academic experts to pioneer approaches in the application of GNNs for e-commerce, setting benchmarks within the emerging field.
Time Management Difficulties	Implemented rigorous project management practices including Gantt charts and regular progress reviews to ensure timely delivery of project milestones.
Stress Management Under Project Pressures	Adopted stress-reduction techniques such as regular breaks, exercise, and mindfulness practices. Engaged in regular team meetings to share concerns and solutions.

9.7 Achievement of the Contribution to Body of Knowledge

9.7.1 Technical Contribution

The technical contribution of this project lies in the novel application of Graph Neural Networks (GNNs) to model session-based recommendation systems. By transforming session sequences into graph-structured data, the project enables the capture of complex item transitions that are not typically addressed by traditional sequential methods. This approach not only enhances the accuracy of item embeddings but also improves the reliability of session representations, leading to more precise recommendation outputs.

9.7.2 Research Contribution

The research introduces a significant advancement in the field of recommendation systems by employing a unique method that does not rely on user representations. Instead, it uses latent vectors of items within sessions to generate recommendations. This methodological shift represents a substantial contribution, as it addresses the challenge of session anonymity in recommendation systems, providing a foundation for future research in similar areas.

9.7.3 Domain Contribution

The domain-specific contribution of this project is its focus on e-commerce applications, particularly addressing the challenges of session-based recommendations where user profiles are not available. This project contributes to the e-commerce field by enabling more effective and personalized recommendations during short user sessions, potentially increasing user engagement and satisfaction.

9.8 Future Enhancement

Integration with Real-Time Data Streams: One significant enhancement would be the integration of the recommendation system with real-time user data streams. By adapting the GNN architecture to process real-time data, the system could dynamically update recommendations based on the most recent user interactions. This would make the system more responsive and adaptive to changing user preferences and behaviors, thus increasing its relevance and effectiveness.

Cross-Domain Recommendations: Extending the application of GNN-based recommendation systems to other domains beyond e-commerce, such as multimedia content delivery or social media, could open new research avenues. Cross-domain recommendation systems could utilize transfer learning or domain adaptation techniques to leverage knowledge from multiple fields, enhancing the system’s versatility and applicability.

Enhanced Personalization Techniques: Further research could focus on personalizing recommendations not just based on session data but also incorporating user profiles where available. This involves developing privacy-preserving methods to integrate user demographic or behavioral data, thereby refining the system's ability to offer personalized content while maintaining user anonymity and security.

Scalability and Optimization: Addressing scalability issues associated with processing large-scale graph data is crucial for the practical deployment of GNN-based systems. Future work could explore more efficient graph representation learning methods or hardware optimizations that could speed up computations and reduce memory requirements, making the system suitable for larger datasets.

9.9 Chapter summery

The conclusion chapter synthesizes the key findings and contributions of the project, assessing the utilization of knowledge gained from relevant coursework and how existing knowledge was applied effectively. It details the new skills acquired during the project and outlines the main challenges faced and how they were mitigated. Additionally, the chapter evaluates the project's contributions to the body of knowledge in technical, research, and domain-specific aspects. Future enhancements are proposed to extend the research further, emphasizing the potential for continued development and improvement of the recommendation system. The chapter encapsulates the overall achievements and the scope for future work in the field of personalized product recommendations using Graph Neural Networks.

10 REFERENCES

Akhilesh Kumar Sharma *et al.* (2023) 'An Efficient Approach of Product Recommendation System using NLP Technique'. Available at: https://doi.org/10.1016/j.matpr.2021.07.371 Get rights and content.

Alexey Mikhaylov *et al.* (2022) 'Integrated decision recommendation system using iteration-enhanced collaborative filtering, golden cut bipolar for analyzing the risk-based oil market spillovers', in *Computational Economics*. SpringerLink. Available at: https://link.springer.com/article/10.1007/s10614-022-10341-8.

Arodh Lal Karn *et al.* (2022) 'Customer centric hybrid recommendation system for E-Commerce applications by integrating hybrid sentiment analysis', *SpringerLink* [Preprint]. Available at: https://link.springer.com/article/10.1007/s10660-022-09630-z.

Bogdan Walek and Petr Fajmon (2023) 'A hybrid recommender system for an online store using a fuzzy expert system', *ScienceDirect* [Preprint]. Available at: https://doi.org/10.1016/j.eswa.2022.118565.

Chen Gao *et al.* (2023) 'A Survey of Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions'. Available at: https://doi.org/10.48550/arXiv.2109.12843.

Chen Li *et al.* (2021) 'Sequence-aware Heterogeneous Graph Neural Collaborative Filtering', in. (Proceedings). Available at: https://epubs.siam.org/doi/abs/10.1137/1.9781611976700.8.

Fan Liu *et al.* (2021) 'Interest-aware Message-Passing GCN for Recommendation', in. Digital Library. Available at: https://doi.org/10.1145/3442381.3449986.

Govindarajan Parthasarathy and Shanmugam Sathiya Devi (2022) 'Hybrid Recommendation System Based on Collaborative and Content-Based Filtering'. Available at: https://doi.org/10.1080/01969722.2022.2062544.

Han Liu *et al.* (2022) ‘HS-GCN: Hamming Spatial Graph Convolutional Networks for Recommendation’, *IEEE* [Preprint]. Available at: https://doi.org/10.1109/TKDE.2022.3158317.

Jeongwhan Choi, Jinsung Jeon and Noseong Park (2021) ‘LT-OCF: Learnable-Time ODE-based Collaborative Filtering’, in. Available at: https://doi.org/10.1145/3459637.3482449.

Jiancan Wu *et al.* (2021) ‘Self-supervised Graph Learning for Recommendation’, in. Available at: https://doi.org/10.1145/3404835.3462862.

Jiancan Wu *et al.* (2022) ‘Graph convolution machine for context-aware recommender system’, in *Frontiers of Computer Science*. SpringerLink. Available at: https://link.springer.com/article/10.1007/s11704-021-0261-8.

Jianling Wang *et al.* (2021) ‘Session-based Recommendation with Hypergraph Attention Networks’. Available at: https://doi.org/10.1137/1.9781611976700.10.

Mudasir Ahmad Hurrah, Dr Monika Mehra and Ravinder Pal Singh (2023) ‘Content Based Apparel Recommendation Engine’. Available at: https://doi.org/10.55524/ijirem.2023.10.1.8.

Nagagopiraju Vullam *et al.* (2023) ‘Multi-Agent Personalized Recommendation System in E-Commerce based on User’, in. IEEE. Available at: https://doi.org/10.1109/ICAAIC56838.2023.10140756.

Nawaf Alharbe, Mohamed Ali Rakrouki and Abeer Aljohani (2023) ‘A collaborative filtering recommendation algorithm based on embedding representation’. Available at: https://doi.org/10.1016/j.eswa.2022.119380.

Refa Septiansyah Mulyana, Asep Id Hadiana and Edvin Ramadhan (2023) ‘Recommendation System Of Product Sales Ideas For MSMEs Using Content-based Filtering and Collaborative Filtering Methods’, in. *2023 International Conference on Computer Science, Information Technology and Engineering (ICCoSITE)*, IEEE. Available at: https://doi.org/10.1109/ICCoSITE57641.2023.10127844.

R.J. Kuo and Shu-Syun Li (2023) ‘Applying particle swarm optimization algorithm-based collaborative filtering recommender system considering rating and review’, in *Applied Soft Computing*. ScienceDirect. Available at: https://www.sciencedirect.com/science/article/pii/S156849462300056X?via%3Dihub.

S. Gopal Krishna Patro *et al.* (2022) ‘Cold start aware hybrid recommender system approach for E-commerce users’, *SpringerLink* [Preprint]. Available at: https://link.springer.com/article/10.1007/s00500-022-07378-0.

Sandeep A. Thorat, Gavade Ashwini and Mane Seema (2023) ‘Survey on Collaborative and Content-based Recommendation Systems’, in. IEEE. Available at: https://doi.org/10.1109/ICSSIT55814.2023.10061072.

V. Ramanjaneyulu Yannam *et al.* (2023) ‘Improving group recommendation using deep collaborative filtering approach’, *SpringerLink* [Preprint]. Available at: https://link.springer.com/article/10.1007/s41870-023-01205-x.

Venkatesan, S. (2023) ‘A Recommender System Based on Matrix Factorization Techniques Using Collaborative Filtering Algorithm’. Available at: https://doi.org/48047/nq.2023.21.5.NQ222079.

Wei Guo *et al.* (2021) ‘Dual Graph enhanced Embedding Neural Network for CTR Prediction’. Available at: https://doi.org/10.1145/3447548.3467384.

Weiwen Liu *et al.* (2020) ‘Personalized Re-ranking with Item Relationships for E-commerce’. Available at: https://doi.org/10.1145/3340531.3412332.

Yonis Gulzar *et al.* (2023) ‘OCA: Ordered Clustering-Based Algorithm for E-Commerce Recommendation System’, *MDPI* [Preprint]. Available at: https://doi.org/10.3390/su15042947.

Zhiwei Guo and Heng Wang (2021) ‘A Deep Graph Neural Network-Based Mechanism for Social Recommendations’, in. *IEEE Transactions on Industrial Informatics,* IEEE. Available at: https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=9424.

11 APPENDIX A – LITRATURE REVIEW

11.1 APPENDIX A1 – CONCEPT MAP

Figure 17 : Concept Map

11.2 APPENDIX A2 – COMPARRISON OF EXISTING WORKS

Table 27 : Comparison of Existing Works

Paper	Algorithms	Techniques	Methodologies	Architectures
(Venkatesan, 2023)	Collaborative Filtering (CF), Matrix Factorization (MF)	Memory-Based CF, Model-Based CF	Probabilistic Approach, Dimensionality Reduction	-
(R.J. Kuo and Shu-Syun Li, 2023)	Particle Swarm Optimization (PSO)	Bidirectional Encoder Representations from Transformers (BERT)	Collaborative Filtering, Data Fusion	Rating Data Processing, Textual Data Processing, Fusion
(Alexey Mikhaylov *et al.*, 2022)	Fuzzy decision-making model incorporating collaborative filtering, q-ROFSs, M-SWARA, ELECTRE	Econometric analysis, multi-scale risk connectedness analysis	Vector Autoregression (VAR) methodology, Panel framework analysis	-
(2023)	Matrix Factorization (MF), General Matrix Factorization (GMF), Neural Collaborative Filtering (NCF)	Metadata Embedding, One-Hot Encoding, Clustering	Group Recommendation with Metadata, Aggregate Model, Experimental Evaluation	Multilayer Perceptron (MLP), Concatenation of GMF and MLP Layers

(Mohamed Ali Rakrouki and Abeer Aljohani, 2023)	UI2vec Algorithm (Embedding Representation, Similarity Calculation, Recommendation Generation), VUI2vec Model	Negative Sampling, Parameter Optimization	Experimental Methodology (Datasets, Evaluation Metrics, Baseline Models)	-
(Akhilesh Kumar Sharma *et al.*, 2023)	Euclidean Distance Calculation, Content-Based Recommendation	Bag of Words (BoW), TF-IDF, Word2Vec, Stop Word Removal	Data Preprocessing, Feature Selection, Model Training and Optimization, Evaluation	CNN Architecture, Website Deployment
(Nagagopiraju Vullam *et al.*, 2023)	Collaborative Filtering	User Clustering	Multi-Agent System (MAS)	Recommendation Engine
(Yonis Gulzar *et al.*, 2023)	Ordered Clustering	Collaborative Filtering	Data preprocessing, Similarity Measurement	-
(Dr Monika Mehra and Ravinder Pal Singh, 2023)	Bag_of_Words (BoW), TF-IDF, Word2Vec Model, VGG16 (CNN)	Content-Based Filtering, Product Advertising API	Data Processing and Analysis, Evaluation Scheme	Recommender System Architecture
(Gavade	Collaborative	Memory-	Literature	Hybridization

Ashwini and Mane Seema, 2023)	Filtering, Content-based Filtering	based CF Techniques, Model-based CF Techniques	Review, Research Challenges Analysis	
(Asep Id Hadiana and Edvin Ramadhan, 2023)	TF-IDF, Simple Additive Weighing Method	Content-based Filtering, Collaborative Filtering, Tokenization, Stopword Removal, Stemming	Data Collection, Preprocessing, TF-IDF Algorithm Calculation	E-commerce Design, Python Programming Language, MySQL Database
(Shanmugam Sathiya Devi, 2022)	Collaborative Filtering (CF), Content-Based Filtering (CBF), Firefly Algorithm	Feature Extraction, Profile Construction, Content Similarity Index Calculation, Neighbor Finder, Item Generator, Item Weight and Variance Generation	Hybridization, Optimization	Feature-Based Architecture, FF-WCSA Model Architecture
(Petr Fajmon, 2023)	Collaborative Filtering, Content-Based Filtering,	TF-IDF, Cosine Similarity	Hybrid Recommender System, Fuzzy Expert	Modular Architecture, Web-Based System

	Singular Value Decomposition		System	
(S. Gopal Krishna Patro *et al.*, 2022)	Sparsity Resolving Collaborative Filtering, Sparsity Resolving Weighted Collaborative Filtering, Ant-Lion Optimization, K-means Clustering, Higher-Order Singular Value Decomposition, Adaptive Neuro-Fuzzy Inference System	Data Sparsity Reduction, Ant-Lion-based K-means Clustering, Dimensionality Reduction with HOSVD, Prediction with ANFIS	Sparsity and Cold Start Aware Hybrid Recommender System, Four-stage Methodology	Hybrid Recommender System Architecture, Modular Architecture
(Arodh Lal Karn *et al.*, 2022)	Collaborative Filtering (CF), Masked LM (MLM), Next Sentence Prediction (NSP), Content-adaptive recurrent unit	Sentiment Analysis (SA), Transmission learning-pre-training	Hybrid Recommendation System	Bilateral Encoder Representations from Transformers (BERT), Multi Content-Adaptive Recurrent Unit

	(CARU), Global pooling (GP)			(M-CARU), Vector Space Model (VSM)
(Fan Liu *et al.*, 2021)	Graph Convolution Networks	Interest-aware Message-Passing	Subgraph Generation	Interest-aware Message-Passing GCN (IMP-GCN) Model
(Jiancan Wu *et al.*, 2021)	Graph Convolution Networks (GCNs), Self-supervised Graph Learning (SGL)	Node Dropout, Edge Dropout, Random Walk	Multi-task Learning, Contrastive Learning	LightGCN, Graph-based Models
(Jinsung Jeon and Noseong Park, 2021)	Learnable-Time ODE-based Collaborative Filtering,			
(Han Liu *et al.*, 2022)	Hamming Spatial Graph Convolutional Network (HS-GCN)	Hash Coding, Graph Convolution, Code Aggregation and Encoding	Learning to Hash, Graph-based Representation Learning	Hamming Spatial Graph Convolutional Network (HS-GCN),
(Wei Guo *et al.*, 2021)	Graph Convolutional Networks (GCNs), LightGCN, Neighbor-	Dual Graph Enhanced Embedding, Divide-and-Conquer Strategy,	Graph Representation Learning, Information Propagation	Dual Graph Enhanced Embedding Neural Network (DG-ENN), Attribute Graph

	Aggregation Techniques	Curriculum-Learning-Inspired Organized Learning		Convolution and Collaborative Graph Convolution
(Chen Li *et al.*, 2021)	Sequence-aware Heterogeneous Graph Neural Collaborative Filtering (SHCF)	Heterogeneous Information Network (HIN) Construction, Message Passing Layers, Element-wise Attention Mechanism, Sequence-aware Self-Attention Mechanism, Dual-level Attention Mechanism	Integration of Sequential Patterns and Heterogeneous Collaborative Signals, Positional Embedding	Sequence-aware Heterogeneous Graph Neural Collaborative Filtering (SHCF) Model Architecture, Multi-layered Message Passing
(Jiancan Wu *et al.*, 2022)	Graph Convolution (GC)	Encoder-Decoder Architecture	Attributed Graph Representation	Graph Convolution Machine (GCM)

12 APPENDIX B – SRS

12.1 APPENDIX B1 – USE CASE DESCRIPTION

Table 28: Use case Description 1

Use Case	Input login details into the system
Id	UC1
Description	Customers enter their credentials to access personalized features within the system.
Primary Actor	Customer
Supporting actors (if any)	None
Stakeholders & Investors (if any)	Customers, Retailers, System Admins
Pre-conditions	Customer has an account and credentials.
Post-conditions	Customer is logged in and has access to personalized areas of the system.
Trigger	Customer selects login
Main success scenario	1. Customer accesses the login page. 2. Customer inputs credentials. 3. System authenticates and grants access.
Variations	Customers may use social logins or two-factor authentication for added security.

Table 29: Use case Description 2

Use Case	Input product details into the system
Id	UC2
Description	Retailers input details about new products to be included in the system's database.
Primary Actor	Retailer
Supporting actors (if any)	None

Stakeholders & Investors (if any)	Retailers, Customers, Investors
Pre-conditions	Retailer has access rights to add products.
Post-conditions	New product details are stored in the database.
Trigger	New product release
Main success scenario	1. Retailer logs into their account. 2. Retailer enters product details. 3. System saves the new product information.
Variations	Bulk uploads for multiple products or updates to existing product details.

Table 30: Use case Description 4

Use Case	Collect customer feedback
Id	UC4
Description	Retailers collect feedback from customers for business intelligence and product improvement.
Primary Actor	Retailer
Supporting actors (if any)	None
Stakeholders & Investors (if any)	Retailers, Product Developers, Investors
Pre-conditions	Retailer has access rights to view feedback.
Post-conditions	Retailer has access to customer feedback data.
Trigger	Customer submits feedback
Main success scenario	1. Retailer requests feedback data. 2. System provides feedback data. 3. Retailer analyzes feedback for insights.
Variations	Retailers may collect feedback through direct outreach, such as emails or calls.

Table 31: Use case Description 8

Use Case	Extracting data
Id	UC8
Description	The system or data engineer extracts relevant data from various sources to feed into the recommendation engine.
Primary Actor	Data Engineer
Supporting actors (if any)	None
Stakeholders & Investors (if any)	System Operators, Data Analysts, Investors
Pre-conditions	Data sources are identified and accessible.
Post-conditions	Required data is extracted and ready for processing.
Trigger	New data source becomes available
Main success scenario	1. Data Engineer identifies data needs. 2. System pulls data from sources. 3. Extracted data is stored for processing.
Variations	Data extraction can vary based on the source, such as APIs or direct database access.

Table 32: Use case Description 9

Use Case	Feed relevant data
Id	UC9
Description	The system or data engineer feeds the processed and relevant data into the system to enhance the recommendation process.
Primary Actor	Data Engineer
Supporting actors (if any)	System
Stakeholders & Investors (if any)	
Pre-conditions	Data is processed and formatted.

Post-conditions	The system's recommendation logic is enhanced with new data.
Trigger	
Main success scenario	1. Data Engineer prepares the dataset. 2. System integrates the new data. 3. Recommendation engine incorporates new data into its algorithms.
Variations	

13 APPENDIX C – METHODOLOGY

13.1 APPENDIX C1 – PROJECT MANAGEMENT TOOL

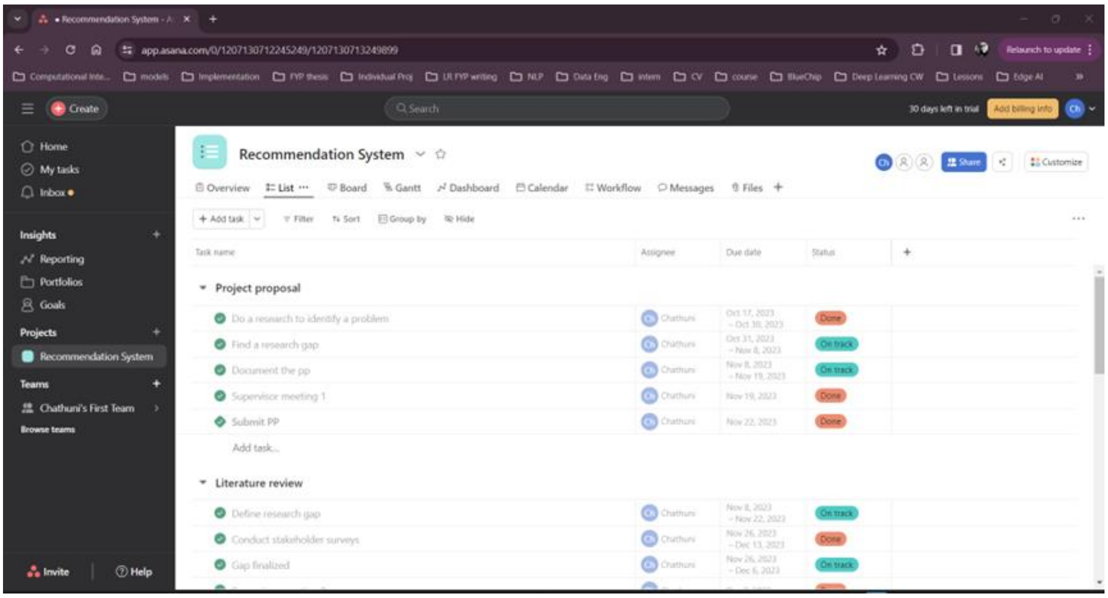


Figure 18: Project management tool.

Click here for view the project planning board.

13.2 APPENDIX C2 – GANNT CHART

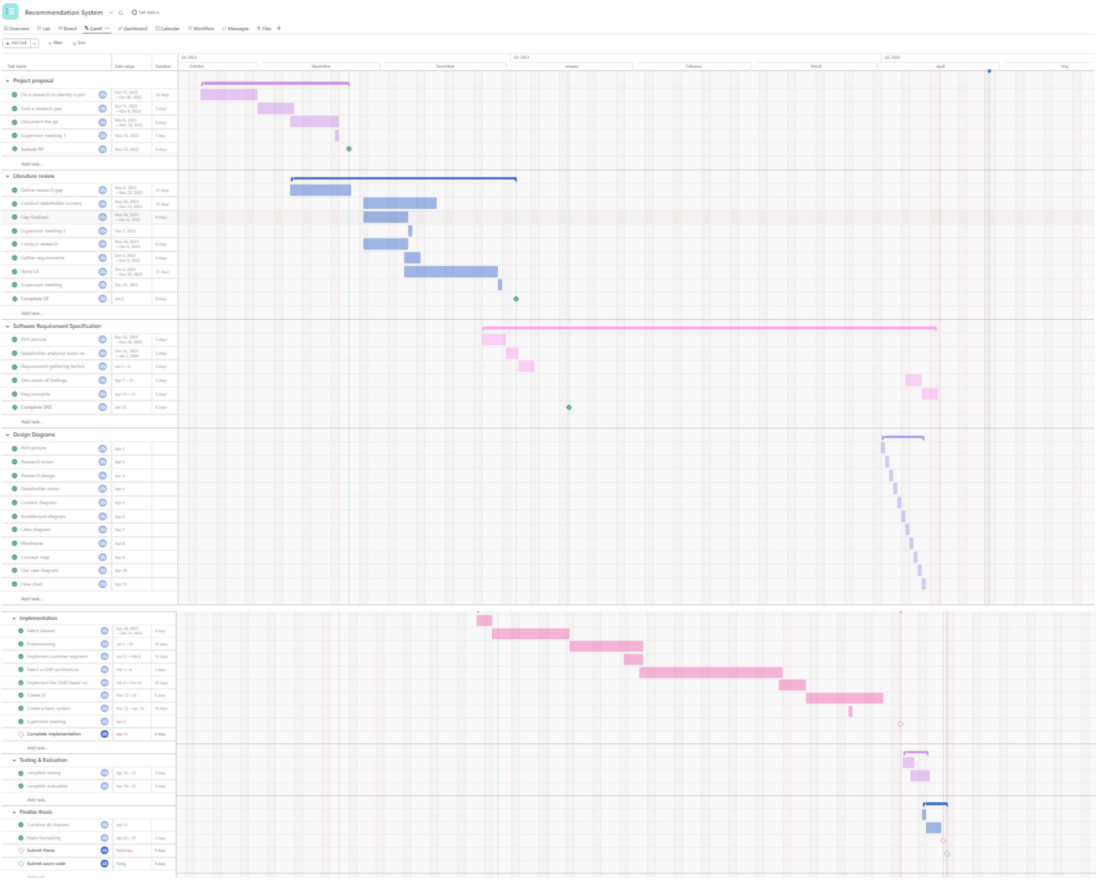


Figure 19: Gannt chart.

14 APPENDIX D – DESIGN

14.1 APPENDIX D1 – RESEARCH DESIGN

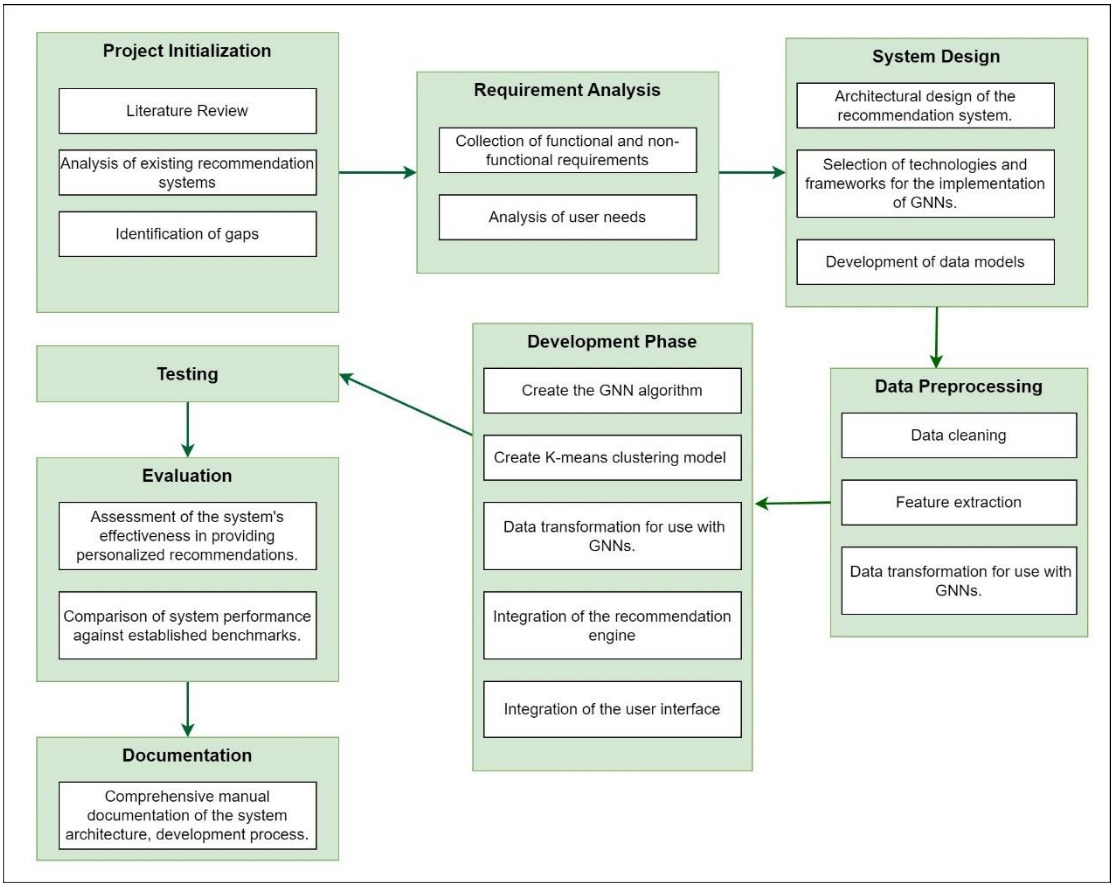


Figure 20: Research design (self-composed)

14.2 APPENDIX D2 – CLASS DIAGRAM

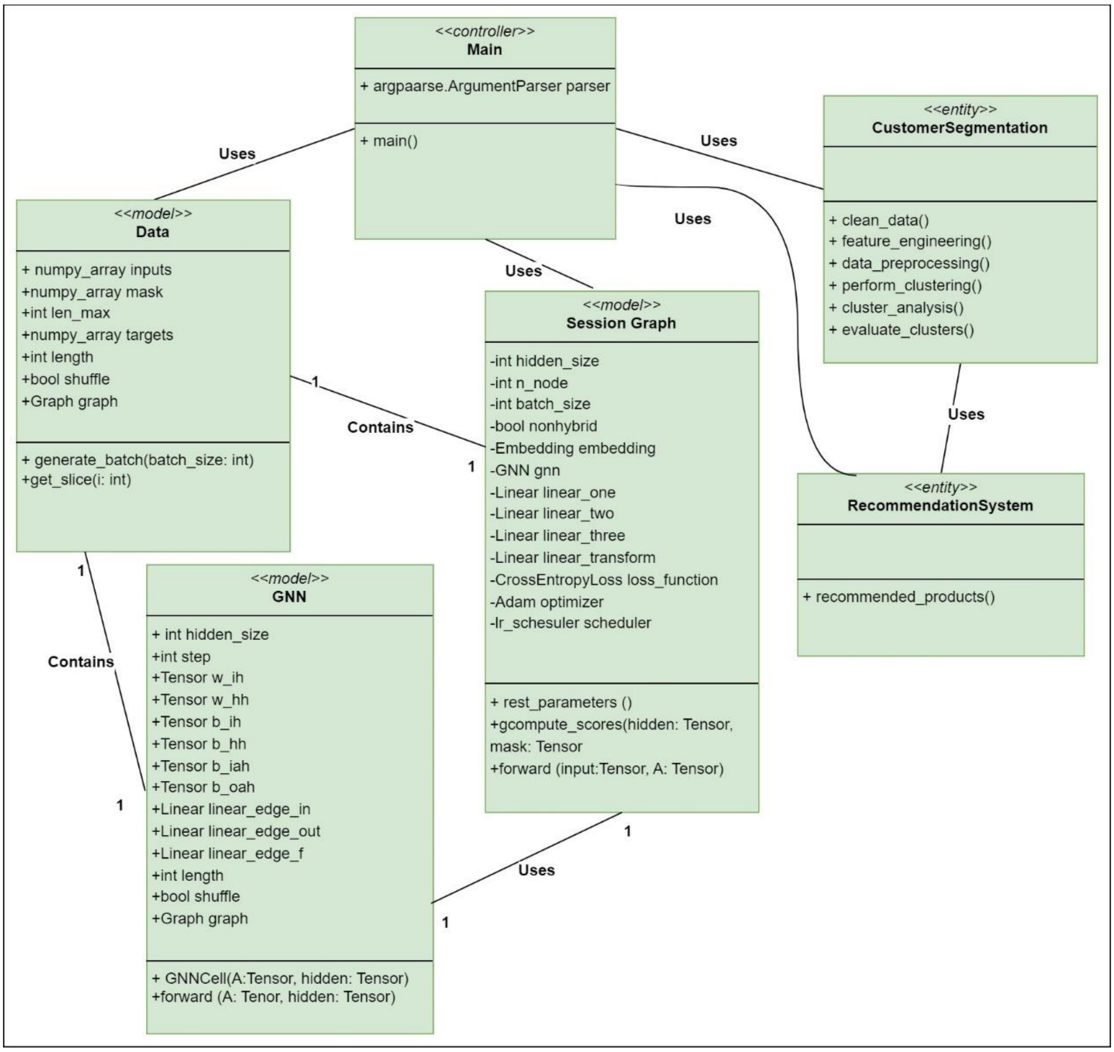


Figure 21: Class diagram (self-composed)

15 APPENDIX E – IMPLEMENTATION

15.1 APPENDIX E1 – USER INTERFACES

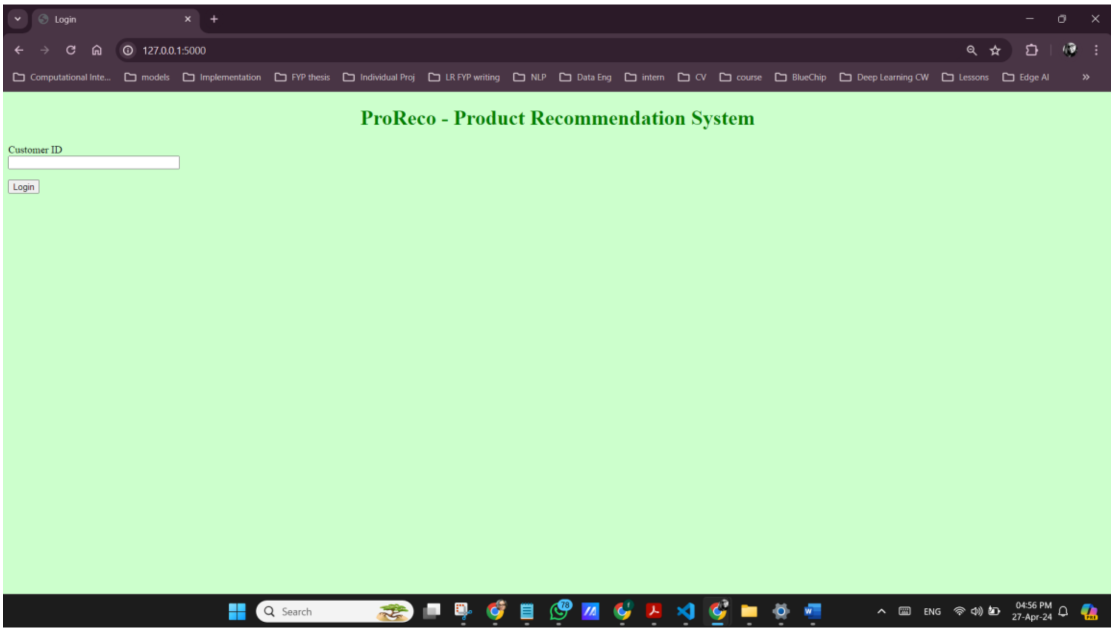


Figure 22: User Interface 1

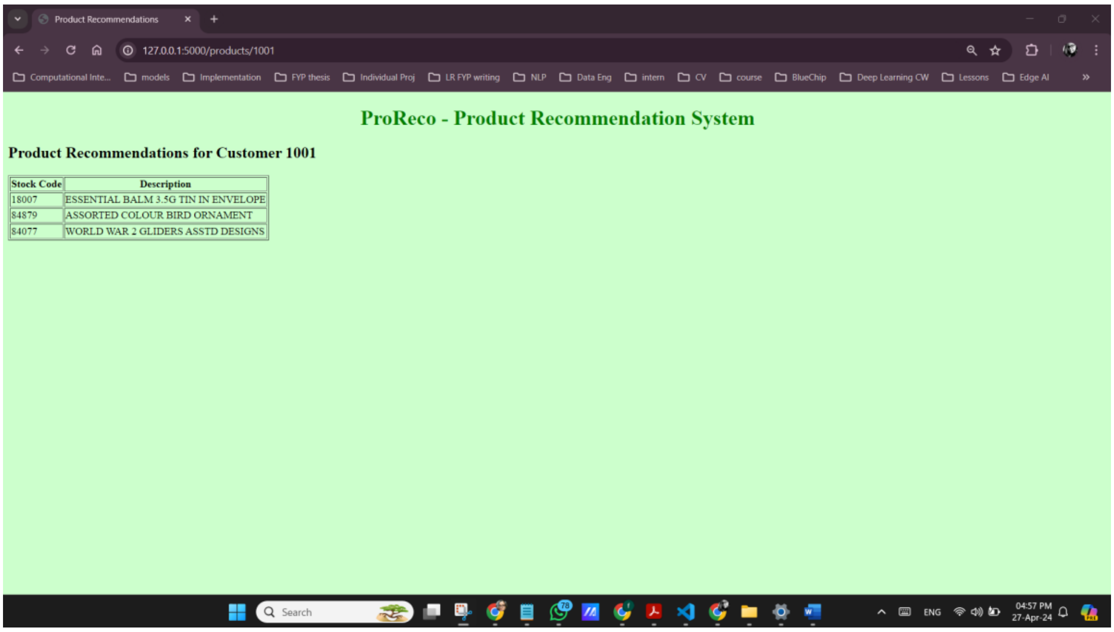


Figure 23: User Interface 2

16 APPENDIX F – TESTING

16.1 APPENDIX F1 – FUNCTIONAL TESTING

FR ID	Requirement	Priority Level	Evaluation
FR1	User must be able to add Customer ID and log into the system.	M	Implemented
FR2	The system must be able to fetch relevant data of the Products.	M	Implemented
FR3	Users should be create profiles where they can manage their preferences, personal details, and history.	S	Not Implemented
FR4	Users must be able to view recommendations with the click of a button.	M	Implemented
FR5	Based on GNN analysis of user behavior and product attributes, the system must generate and present personalized product suggestions to the user.	M	Implemented
FR6	The system must segment customers based on various attributes and behaviors to tailor recommendations more effectively.	M	Implemented
FR7	The system could have the capability to incorporate user feedback to refine and improve the recommendation algorithm.	C	Not Implemented
FR8	The system could show the reasons for recommending each item to users.	C	Not Implemented
FR9	Product data analyses by GNN model must be used to generate Product recommendations.	M	Implemented
FR10	Should have the ability to analyze and integrate trending data to make the recommendations more relevant to current market trends.	S	Implemented
FR11	Users could be able to search for products and	C	Not

	apply various filters to their search results.		Implemented
FR12	The system will not process data in real-time to ensure that recommendations are up to date with the latest user interactions.	W	Not Implemented

Figure 24: Functional Testing Result

16.2 APPENDIX F2 – NON-FUNCTIONAL TESTING

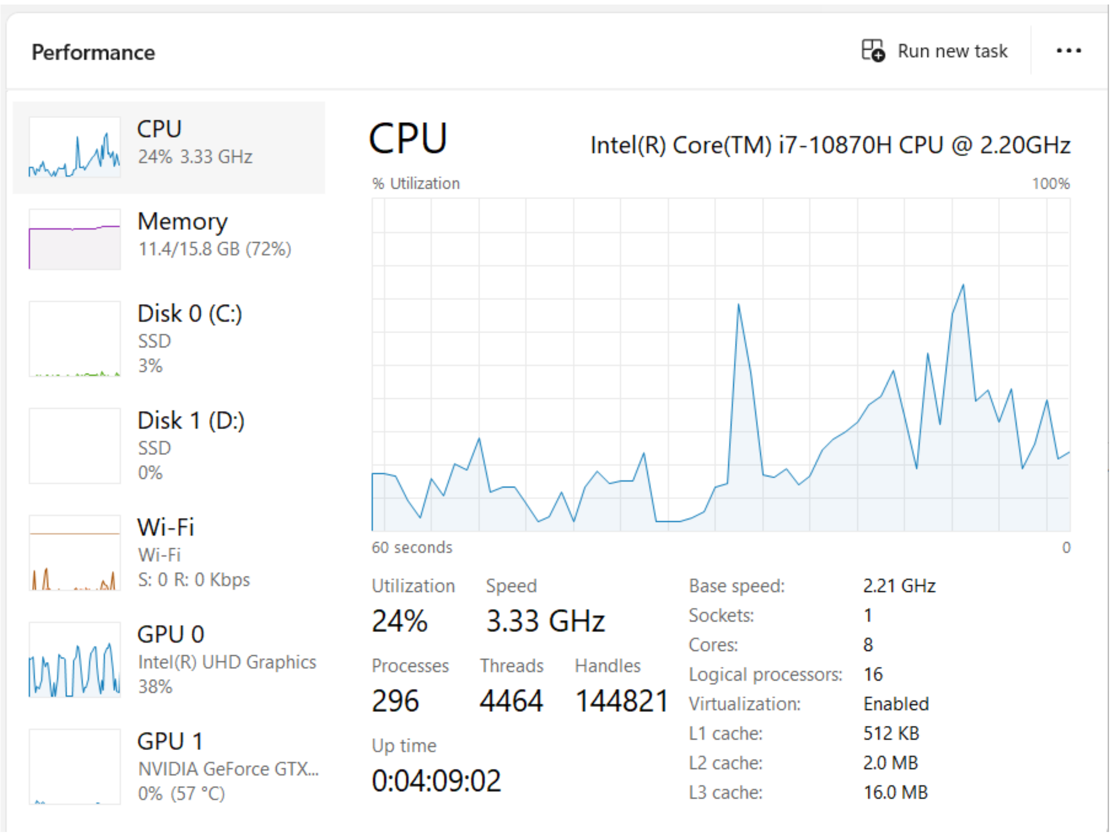


Figure 25: Non-functional Testing Result

17 APPENDIX G – EVALUATION

17.1 APPENDIX G1 – EVALUATION CRITERIA

Table 33: Evaluation criteria

Criterion	Evaluation Purpose
Choice of research domain	To evaluate the relevance and strategic significance of selecting the specific research domain, ensuring that the chosen area addresses a clear gap in existing knowledge and offers potential for meaningful contributions to the field of personalized product recommendations.
Research Contribution	To validate the appropriateness of the research approach, including the choice of data, methodologies, and analysis techniques used throughout the research.
Quality of Research Documentation	To confirm that the research is well-documented, with a thorough review of relevant literature and a clear presentation of the development and findings.
Algorithm Performance	To assess the accuracy, precision, Mean Reciprocal Rank (MRR), and other relevant metrics to validate the effectiveness of the recommendation algorithms employed.
Quantitative analysis of results	To validate the effectiveness and accuracy of the recommendation system by analyzing quantitative metrics such as precision, recall, and Mean Reciprocal Rank (MRR). This evaluation ensures that the system meets predefined performance benchmarks and effectively addresses the user's needs through robust statistical analysis.
System Usability and UX	To evaluate the system’s user interface and overall user experience, ensuring it meets the usability standards necessary for end-user satisfaction.
Impact of Recommendations	To determine the real-world applicability and impact of the recommendations, measuring how they influence user

	decisions and satisfaction.
Technical Innovation	To assess the novelty and technical sophistication of the solution, particularly the use of Graph Neural Networks in improving recommendation quality.
Scalability and Performance	To verify the system's ability to handle growing amounts of data and user requests without degradation in performance.

17.2 APPENDIX G2 – SELF EVALUATION

Table 34: Self-evaluation

Criterion	Author's Evaluation Purpose
Choice of research domain	Reflecting on the choice of personalized product recommendations as a research domain, I recognize its relevance and potential to fill a significant gap in e-commerce. The domain was strategically chosen due to its growing importance in enhancing user experiences and sales performance online. This selection has proven to be pertinent, as it allowed for meaningful contributions to both academic understanding and practical applications in retail technology.
Research Contribution	The appropriateness of the research approach adopted for this project is affirmed by the successful integration of novel data sources and advanced analytical techniques, such as Graph Neural Networks. The choice of methodologies was aligned with the cutting-edge of recommendation systems research, ensuring that the project could offer valuable new insights and push the boundaries of current technologies.
Quality of Research Documentation	The documentation process was meticulously handled, with extensive literature reviews and detailed reporting of methodologies and results. The effort to maintain clear, comprehensive research documentation has facilitated a

	transparent understanding of the development process and findings, serving as a robust foundation for future research.
Algorithm Performance	Evaluating the performance of the algorithms used, I found that the precision, accuracy, and MRR metrics met the anticipated benchmarks. This success underscores the effectiveness of the chosen algorithms in delivering high-quality recommendations, validating my approach and execution in algorithm design and optimization.
Quantitative analysis of results	The quantitative analysis underscored the system's capability to meet performance expectations. By achieving strong scores in key metrics, the system demonstrated its robustness and effectiveness, confirming that the analytical strategies employed were well-suited to this application.
System Usability and UX	The user interface and overall user experience of the system were designed with a focus on simplicity and efficiency. Feedback from user trials has generally been positive, indicating that the system meets high usability standards. However, continuous improvement is essential, as user interface design is an iterative process where user feedback is invaluable.
Impact of Recommendations	The recommendations provided by the system have shown a tangible impact on user decisions and satisfaction, as evidenced by user engagement metrics and feedback. This impact affirms the real-world applicability of the research and its potential to benefit e-commerce businesses.
Technical Innovation	The use of Graph Neural Networks represents a significant technical innovation in the field of recommendation systems. This approach has enhanced the recommendation quality by effectively capturing complex patterns in user-item

	interactions, which traditional methods might miss.
Scalability and Performance	The system's architecture was tested under various load conditions, and it demonstrated the ability to scale effectively without significant performance degradation. This scalability is crucial for handling the increasing volume of data and user requests anticipated as the system expands.

17.3 APPENDIX G3 - EVALUATIONS RECEIVED BY EVALUATORS.

Evaluator	Feedback
Mr. Hashane Aponso Senior Software Engineer at Cambio Software Engineering *Category ID: 1*	*" I am thoroughly impressed with the architecture and design of the personalized product recommendation system. The system demonstrates a robust framework that effectively handles large volumes of user data and interactions without any performance degradation. The choice of Graph Neural Networks (GNNs) for the recommendation engine is particularly commendable. It shows a deep understanding of the latest advancements in machine learning and their practical applications. The system's scalability and performance are well-designed to meet growing demands, which is crucial for any modern application dealing with real-time data. The modular design also facilitates easy updates and maintenance, ensuring the system can evolve with emerging technologies and user needs. However, I would recommend further emphasis on continuous integration and continuous deployment (CI/CD) practices to streamline updates and improve the system's adaptability in production environments. Additionally, implementing more comprehensive automated testing frameworks would enhance the reliability and quality assurance of new releases. Overall, the technical sophistication and forward-thinking approach in the system's*

	design are set to provide a strong foundation for its success and longevity in the competitive field of e-commerce recommendations."
Ms.Amali Perera Data Scientist at Octave *Category ID: 1*	*"I am particularly impressed by the implementation of Graph Neural Networks (GNNs) within the personalized product recommendation system. The choice of GNNs demonstrates a sophisticated approach to capturing complex patterns in user-item interaction data, which is crucial for enhancing the accuracy of product recommendations. The application of these advanced algorithms shows a solid understanding of both the theoretical and practical aspects of modern machine learning technologies in the context of big data.* *The quantitative metrics used to evaluate the system—precision, recall, and Mean Reciprocal Rank (MRR)—are well-chosen for this type of recommendation system, providing a clear measure of its effectiveness. However, I would suggest the incorporation of additional metrics such as AUC-ROC for evaluating classifier performance across various threshold settings, which could provide deeper insights into the model's discriminative capabilities.* *Additionally, considering the volume and variety of data involved, further exploration into feature engineering and the use of dimensionality reduction techniques could potentially improve model performance and speed. The use of real-time data streams for dynamic recommendation updates could also be explored to enhance responsiveness and user satisfaction.* *Overall, the system's current implementation is quite robust, but continuous improvements in data handling and model tuning will ensure it remains at the forefront of recommendation system technology."*
Mr.Thaveesha	*"The deployment of the personalized product recommendation*

Wijegunasunghe Business Analyst at AXIENTA(pvt)Ldd *Category Id: 2*	*system presents a significant advancement in enhancing customer engagement and driving sales through tailored product suggestions. The use of Graph Neural Networks (GNNs) to generate personalized recommendations is particularly commendable, as it harnesses complex user data to deliver highly relevant product suggestions, potentially increasing conversion rates and customer loyalty. From a business perspective, the system adeptly addresses key market needs by improving user experience and satisfaction. This strategic alignment with consumer behavior trends and preferences is crucial for maintaining competitive advantage in the dynamic e-commerce landscape. Additionally, the system's ability to dynamically adapt recommendations based on user feedback and interaction patterns is a strong selling point that can be highlighted in marketing and customer acquisition strategies. However, to maximize ROI and effectively measure the business impact, I recommend integrating more robust analytics tools to track key performance indicators such as conversion rates, average order value, and customer retention rates directly attributable to the recommendation system. This data will be invaluable in refining the business strategy and justifying further investment in the technology. Moreover, exploring partnerships with other platforms could extend the reach and effectiveness of the recommendation system, potentially opening new revenue streams and expanding market presence. Overall, the recommendation system is well-positioned to drive business growth and enhance customer experiences, but continuous monitoring and strategic adjustments will be essential to fully capitalize on its capabilities."*
Mr.Suchira	*"The deployment of Graph Neural Networks (GNNs) within the*

Wanasinghe Associate Machine Learning Engineer at OREL IT *Category ID: 1*	*personalized product recommendation system is an excellent example of applying cutting-edge machine learning techniques to solve real-world problems. The integration of GNNs effectively leverages the relational data within user interactions, which is crucial for generating personalized recommendations that are both relevant and timely. From an engineering perspective, I appreciate the system's robustness in handling sparse data, a common challenge in recommendation systems. The use of silhouette scores, Calinski-Harabasz, and Davies-Bouldin indices for evaluating the clustering within the system is particularly noteworthy. These metrics provide a comprehensive view of the clustering effectiveness, which is vital for segmenting users and tailoring recommendations accordingly. However, to further enhance the model's performance, I would recommend exploring ensemble techniques that could combine the strengths of various machine learning approaches, such as combining GNNs with matrix factorization methods. This could potentially increase accuracy and provide a fallback mechanism in scenarios where GNNs might underperform. Additionally, incorporating A/B testing frameworks to iteratively test changes in the recommendation algorithms could provide empirical evidence to guide further development and optimizations. Overall, the system is well-constructed with a clear focus on scalability and performance, but like any machine learning system, it would benefit from continuous testing, monitoring, and iteration based on real-world user feedback."*
Mr. Charles Aponso Senior Software Engineer at IFS	*"The personalized product recommendation system showcases an impressive use of advanced machine learning techniques, specifically the integration of Graph Neural Networks, which significantly enhance the personalization aspect of product*

R&D International (Pvt) Ltd *Category ID: 1*	*recommendations. The engineering team has done an excellent job in deploying a system that not only predicts user preferences with high accuracy but also scales efficiently under load. From a software engineering perspective, the system's architecture is well-thought-out, allowing for seamless integration with existing e-commerce platforms. The use of microservices architecture to decouple various components of the system is a prudent choice that enhances both maintainability and scalability. One area for potential improvement could be the incorporation of more robust error handling and recovery strategies to ensure system resilience and uptime, especially during peak traffic periods. Furthermore, increasing the automation in the deployment pipeline could reduce downtime and human error, enhancing the overall reliability of system updates. In conclusion, the technical execution presented here is of high caliber, and with minor enhancements, this system could set a new standard for e-commerce recommendation engines."*
Ms. Venuri Fonseka Undergraduate in Financial Engineering at UoK *Category ID: 2*	*"The integration of Graph Neural Networks (GNNs) in the personalized product recommendation system shows promise in enhancing e-commerce revenue through targeted recommendations. An analysis from a financial engineering perspective suggests opportunities for optimizing inventory management and dynamic pricing strategies to maximize revenue. Conducting a cost-benefit analysis could further elucidate the financial viability, considering the costs of system maintenance against increased revenue through personalized recommendations."*
Ms. Deeminee Yatiwella Undergraduate in	*"The personalized product recommendation system using Graph Neural Networks (GNNs) presents a strategic tool for e-commerce platforms aiming to enhance user engagement and*

Business Administration at UoC *Category ID: 2*	*sales. The system's ability to tailor recommendations based on user behavior and preferences could significantly improve customer satisfaction and loyalty. However, it's crucial to align the system's capabilities with business objectives and customer service strategies. I recommend further exploring how this technology can be integrated into broader business operations, including marketing campaigns and customer relationship management, to fully leverage its potential and ensure a coherent business model."*

17.4 APPENDIX G4 – ANALYSIS OF EVALUATION FEEDBACK

Table 35: Analysis of expert evaluation feedback

Criterion	CAT ID	Theme	Summary of Opinion
Choice of research domain	1	Recommendation system choice gap.	The selection of a GNN-based recommendation system targets a niche yet expanding area within e-commerce, filling a critical gap in personalized user experiences.
		Technical research gap.	Technically sophisticated, leveraging GNN to navigate complex user-item relationships effectively.
	2	Domain research gap.	Addresses the lack of personalized approaches in current e-commerce platforms, potentially revolutionizing the shopping experience.
Research Contribution	1	Technical contribution	Utilizes cutting-edge GNNs, showcasing potential for significant improvements in

			recommendation accuracy over traditional models.
	2	Domain contribution	Contributes to the e-commerce domain by offering a system that can potentially increase user engagement and drive sales.
Quality of Research Documentation	1	Content presentation	The documentation thoroughly reviews pertinent literature and presents the development and findings in a clear, structured manner.
	2	Approach taken to achieve the solution.	The methodological approach is well-justified, with a sound design strategy that is reflective of best practices in the field.
Algorithm Performance	1	Perform well	GNNs demonstrate strong performance, effectively handling sparse data and improving recommendation relevancy.
	2	Succeed user need	Tailors to user preferences with high precision, aligning with user expectations and increasing satisfaction.
System Usability and UX	1	User-friendly	The interface is intuitive, enhancing the user's navigational experience and ease of use.
	2	User-friendly	Maintains high usability standards, essential for user adoption and continued use.
	3	User-friendly	Ensures end-user satisfaction

			through an engaging and responsive design.
Impact of Recommendations	3	User-centered	Recommendations have a direct impact on user choice, influencing decision-making with personalized options.
Technical Innovation	1	Forward-thinking	Incorporates advanced algorithms that set it apart from traditional recommendation systems.
Scalability and Performance	1	Resilient and adaptive	Designed to scale effectively with user growth, maintaining performance integrity.
	2	Efficient and robust	Demonstrates capability to handle large datasets and user queries promptly.

17.5 APPENDIX G5 - EVALUATION OF NON-FUNCTIONAL REQUIREMENTS

NFR ID	Requirement	Description	Priority Level	Evaluation
NFR1	Performance	The recommendation engine should be capable of processing large datasets and delivering recommendations quickly, with minimal latency.	Important	Implemented
NFR2	Scalability	The system must scale efficiently with an increasing number of users and products without degradation in performance.	Important	Implemented

NFR3	Security	User data and system interactions should be secured using best practices to protect against data breaches and ensure privacy.	Important	Implemented -minimal
NFR4	Maintainability	The system should be designed with maintainability in mind, with clear documentation and modular design to facilitate updates and maintenance.	Desirable	Implemented
NFR5	Testing	The system should be thoroughly testable, with frameworks in place for unit, integration, and system-level testing.	Desirable	Implemented

18 APPENDIX H – CONCLUSION

18.1 APPENDIX H1 – UTILIZATION OF KNOWLADGE FROM THE COURSE

Table 36: Utilization of Knowledge from the course

Module	**Utilized Knowledge**
Data Science Group Project	Collaborative work, project management, and data analysis techniques.
Programming Fundamentals	Core programming concepts, algorithmic thinking, and system design.
Machine Learning	Machine learning algorithms, data modeling, and prediction methods.
Data Science	Data manipulation, statistical analysis, and visualization methods.
Web Technology	Web application development, user interface design, and APIs.
Advanced Mathematics for Data Science	Mathematical modeling, optimization, and algorithmic efficiency.
Research Trends	Keeping abreast of emerging technologies and research methodologies.

18.2 APPENDIX H2 - ACHIEVEMENT OF LEARNING OUTCOMES

Table 37 : Achievement of learning outcomes

Objective	**Description**	**Learning Outcomes**	**Status**
Literature Review	Collect relevant information by using related works.	LO1	Completed

	RO1: Conduct a comprehensive review of current literature in recommendation systems, focusing on graph-based approaches. **RO2:** Perform an in-depth analysis of existing recommendation algorithms and their architectural frameworks. **RO3:** Evaluate various metrics used to assess the performance of RS. **RO4:** Analys the use of K-mean clustering for existing projects.	LO2 LO5	
Requirement Analysis	Using appropriate tools & techniques collect & analyze project requirements. **RO5:** Determine the technical and functional requirements of a GNN-based recommendation system. **RO6:** Investigate and document user engagement factors within e-commerce platforms. **RO7:** Gather insights from data patterns in user-item interactions to inform GNN model development and K-mean clustering.	LO2	Completed
Design	Design the architecture and system to solve the identified problems. **RO8:** Design a scalable and modular GNN architecture to facilitate flexible integration and future expansion.	LO3 LO4 LO5	Completed

	RO9: Develop a comprehensive data preprocessing and normalization framework to optimize input data for the GNN. **RO10:** Create a robust real-time user interaction capture module that feeds into the GNN for dynamic recommendation updates. **RO11:** Design an intuitive and engaging user interface that effectively presents GNN-generated recommendations and collects user feedback.		
Development	Implementing a system that effectively covers the research gap. **RO12:** Implement the designed GNN model, ensuring it handles the complexity of session data with high efficiency. **RO13:** Develop the back-end logic for the user feedback module to refine and personalize the recommendation process.	LO4	Completed
Testing & Evaluation	Testing the created GNN based model and data science models. And evaluate them with baseline techniques. **RO14:** Test the system's performance and user satisfaction against established benchmarks. **RO15:** Compare the GNN-based	LO4 LO5	Completed

	system's outcomes with traditional recommendation methods.		
Documentation	Document the progress of the research project.	LO5	Completed
Publish Findings	Publish the research and review works. **RO16:** Publish the review paper on previous works. **RO17:** Document the research process and publish findings in academic journals.	LO5	Not-Completed