Project 3

“Analysis of Application Architecture on the Amazon E-commerce Website”

Group 4

Names :

I Dewa Nyoman Rama Putera Sadiartha

Muhammad Fakhri Amir

Zahwa Aprilia

Class:

1CC5

CEP-CCIT

FACULTY OF ENGINEERING UNIVERSITY OF INDONESIA

2024

PREFACE

We wish to express our profound gratitude to Almighty God for His boundless mercy and grace, which have been instrumental in the successful completion of this manuscript entitled "Analysis of Application Architecture on the Amazon Website." This manuscript represents the culmination of extensive research and academic effort, undertaken as part of the Center of Excellence Program (CEP) at the Center for Computing and Information Technology (CCIT), Faculty of Engineering, University of Indonesia.

This paper provides an in-depth look at the application architecture of the Amazon website, focusing on two important areas: Search and Query Optimization and Scalability and Performance. As one of the biggest names in e-commerce, Amazon is known for its ability to deliver a smooth and reliable experience for millions of users worldwide. By exploring these areas, this study aims to understand the techniques and strategies Amazon uses to maintain its leading position in the industry.

Search and Query Optimization examines how Amazon processes millions of search requests every second, ensuring users quickly find relevant and personalized results from its vast catalog of products. This involves advanced tools and systems like Elasticsearch, as well as methods to improve speed and accuracy. Scalability and Performance, on the other hand, looks at how Amazon handles billions of daily interactions by using techniques like load balancing, distributed systems, and real-time monitoring to keep its platform running smoothly, even during busy times like Black Friday.

The goal of this paper is to share useful insights into Amazon’s approach to managing such a large and complex platform. By analyzing real-world examples and data, this study also hopes to provide practical ideas for others who want to create systems that are efficient, reliable, and able to handle growth effectively. We hope this work will be helpful to students, professionals, and organizations interested in learning more about successful application design at a large scale.

CEP-CCIT Faculty of Engineering

Universitas Indonesia

TABLE OF CONTENTS

TABLE OF FIGURES

CHAPTER I
INTRODUCTION

1.1 Background

As one of the world's leading e-commerce platforms, Amazon relies on a sophisticated application architecture to support its extensive operations. Search and query optimization are integral components, enabling users to efficiently locate relevant products among millions of listings. By employing advanced techniques such as indexing, caching, and distributed systems like Elasticsearch, Amazon achieves the capability to process millions of queries per second with remarkable precision and reliability.

Scalability and performance are equally paramount to Amazon's architectural framework. The platform accommodates billions of daily interactions by leveraging distributed architectures, load-balancing mechanisms, and horizontal scaling to efficiently manage traffic demands. These measures ensure Amazon maintains consistent performance, even during peak periods such as Black Friday and Prime Day, where surges in user activity present significant challenges.

To uphold system reliability, Amazon implements comprehensive performance monitoring strategies, utilizing key metrics such as latency and throughput. This real-time oversight, combined with predictive analytics, enables proactive issue resolution and ensures the platform delivers a seamless and high-quality user experience across its global operations.

1.2 Writing Objective

The objective of this script is to analyze the application architecture of the Amazon website by examining its approaches to Search and Query Optimization and Scalability and Performance.

- **Analyze Amazon's Architecture**: Examine how the Amazon website handles real-time search and query operations efficiently, leveraging technologies like indexing, caching, and predictive analytics.
- **Explore Scalability**: Highlight Amazon's strategies for scaling its infrastructure, including distributed systems, load balancing, and microservices, to support billions of transactions daily.
- **Search and Query Optimization**: Analyze how Amazon implements advanced indexing, caching, and full-text search engines to deliver accurate and fast search

results, alongside techniques like autocomplete and query suggestions to enhance user experience and reduce query load on the system.

1.3 Problem Domain

This paper explores Amazon's approach to search and query optimization, focusing on processing millions of queries per second through indexing techniques like inverted indexes and distributed systems such as Elasticsearch. It also examines the scalability and performance strategies Amazon uses to handle massive traffic and continuously monitor system efficiency.

Search and Query Optimization

- **Core Challenges**: Discuss how Amazon manages to process millions of search queries per second, ensuring that results are relevant, fast, and personalized.
- **Search Engine**: Highlight Amazon's use of Elasticsearch or OpenSearch for full-text search and how these tools enable distributed querying across vast datasets.

Scalability and Performance

- **Massive Traffic**: Amazon handles billions of requests daily, requiring a highly scalable and distributed system.
- **Performance Monitoring**: Highlight how Amazon uses tools and metrics (e.g., latency, throughput) to continuously monitor and improve performance.

1.4 Writing Methodology

This study employs a descriptive-analytical methodology to explore how Amazon optimizes its search engine and ensures scalability and performance across its global platform. The descriptive approach is used to detail the key components of search and query optimization, including indexing strategies, caching mechanisms, and the architecture of Amazon’s search engine, such as Elasticsearch/OpenSearch, as well as advanced techniques like predictive analytics and machine learning for personalization and relevance ranking. Additionally, the study examines the scalability strategies employed by Amazon, such as distributed systems, sharding, microservices, and load balancing, which enable it to manage billions of daily transactions while maintaining high performance and reliability. The analytical approach evaluates the effectiveness, efficiency, and trade-offs of these components,

addressing challenges such as maintaining consistency across vast datasets, handling dynamic traffic patterns, and delivering low-latency experiences globally. By combining a critical analysis of secondary data from reliable sources, the paper aims to provide in-depth technical insights and practical recommendations for implementing similar strategies in other large-scale, high-demand systems.

1.5 Writing Framework

Chapter 1: Introduction

Include background of Application Architecture on the Amazon E-commerce Website, writing objective, problem domain, and writing methodology.

Chapter 2: Basic Theory

Contains brief description of Application Architecture on the Amazon E-commerce Website.

Chapter 3: Problem Analysis

Discussion of Search and Query Optimization and Scalability and Performance of Application Architecture on the Amazon E-commerce Website.

Chapter 4: Conclusion and Suggestion

Conclusion and suggestion to take from it.

CHAPTER II

BASIC THEORY

2.1 What is Amazon E-Commerce Website

Figure 2.1 Amazon (REF: https://www.nj.com/resizer/FG8ZVUTaIK-Z55BWu9Fg-Vk4l-w=/1280x0/smart/cloudfront-us-east-1.images.arcpublishing.com/advancelocal/OOOYRA5AQVCCPFYYQXJOJXT7UY.JPG **)**

Amazon was once intended to be an online book store when Jeff Bezos founded it in a Washington garage in 1994. The ecosystem swiftly grew to provide a greater variety of goods, such as electronics, video games, food, furniture, and more. In terms of market capitalization, Amazon overtook Walmart as the most valuable US retailer by 2015.

Amazon is well-known for upending the e-commerce industry, and it keeps investing in innovative strategies to alter the dynamics between customers and sellers, such as Amazon Prime or Amazon FBA. Customers may now quickly and easily locate the things they desire in a digital environment thanks to the solution. At the same time, those who want to sell their own goods online may use Amazon to instantly reach a vast client base.

Amazon claims that its sellers have all they need to market their goods, control feedback, track performance, and increase sales. In actuality, third-party merchants now make up over 58% of all Amazon sales, having begun selling on the platform in 1999. Furthermore, Amazon's third-party sales increase at a pace of about 52% annually, while Amazon's own sales only rise at a rate of 25% annually.

2.2 What is Application Architecture

A structural diagram of how a software program is put together and how it communicates with other programs to satisfy user or business needs is called an application architecture. An application architecture helps businesses find functional gaps and guarantees that programs are scalable and dependable.

Generally speaking, an application architecture outlines how programs communicate with databases, middleware, and other programs. Although they might not have official industry standards, application architectures often adhere to well recognized software design principles.

When developing a structure, the application architecture might be compared to architectural blueprints. The building's layout and the locations of utilities like plumbing and electricity are specified in the drawings. These designs may then be used by the builders to construct the structure, and they can be consulted later if the building needs to be expanded or renovated. Without a carefully thought-out application architecture, creating a contemporary software application or service stack would be as challenging as attempting to build a structure without blueprints.

2.3 Benefit of an Application Architecture

All things considered, an application architecture facilitates collaboration between business strategists and IT, ensuring that the appropriate technology is accessible to achieve the goals of the company. An application architecture especially provides the following advantages:

- Identifying redundancies, such using two separate databases that can be replaced by one, lowers expenses.
- Increases productivity by finding gaps, including necessary services that are unavailable through mobile apps.
- Establishes a corporate platform for third-party integration and application accessibility.
- Makes it possible for modular, interoperable systems that are simpler to use and manage.

- Aids in helping architects "see the big picture" and match software plans with the overarching commercial goals of the company.

2.4 Common Application Architecture Patterns

These patterns define how a single application is designed and functions.

- Monolithic Architecture

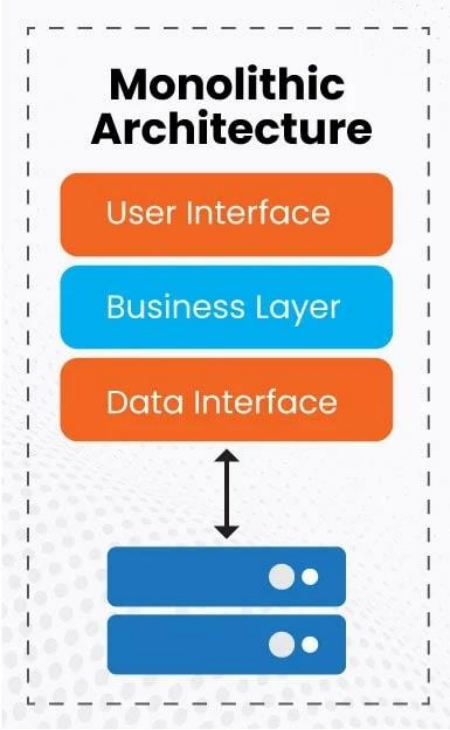


Figure 2.4.1 Monolithic Architecture (REF:
https://www.clariontech.com/hubfs/Monolithic%20Architecture%20Vs.%20Microservices.jpg **)**

Which all operations are managed by a single application running on a single codebase. The application's monolithic design manages data processing, storage, and user input. Many earlier commercial applications employed monolithic architecture. Because they are difficult to maintain and update, many apps are no longer preferred. For really tiny apps run by small teams, they can nevertheless be helpful.

- N-tier architecture

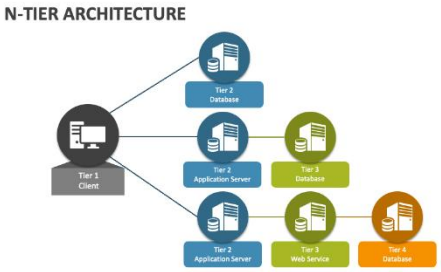


Figure 2.4.2 N-tier Architecture (REF:

https://www.collidu.com/media/catalog/product/img/1/7/17372ce2c0dd9cdbc51bedfbe12ca57970b9aaf04eb23ff5d2d9da25ac97f537/n-tier-architecture-slide5.png **)**

Which each layer of the program handles a distinct aspect of the functionality and the functions are separated into hierarchical levels. The three-tier is a popular design for online applications, whereas the two-tier client-server architecture is the most popular and straightforward.

- Microservices Architecture

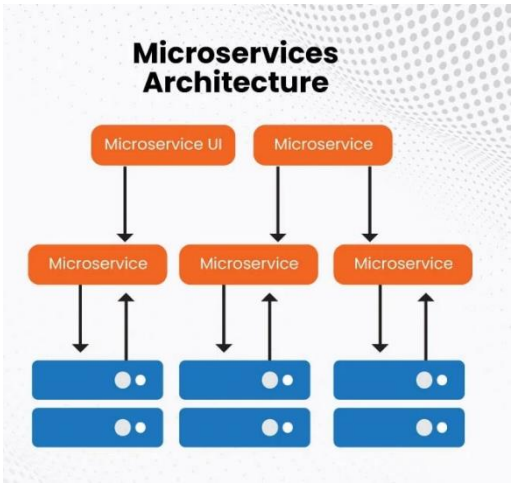


Figure 2.4.3 Microservices Architecture (REF:

https://www.clariontech.com/hubfs/Monolithic%20Architecture%20Vs.%20Microservices.jpg)

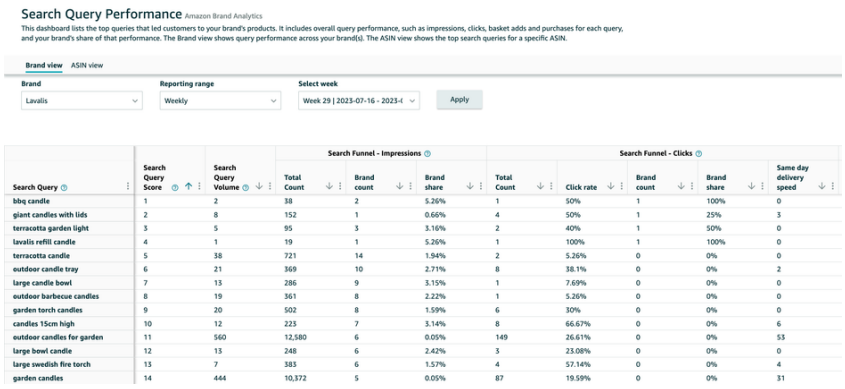
Is a design in which every application function is separated into a separate, independent service or component. Application programming interfaces are a common means of communication between microservices. Applications may be updated and scaled fast with microservices, but this can lead to design silos.

CHAPTER III
PROBLEM ANALYSIS

3.1 Search and Query Optimization

Search and query optimization amazon uses advanced technologies like machine learning and natural language processing to deliver fast and personalized search results. By integrating Elasticsearch and OpenSearch, Amazon efficiently handles large datasets and high query volumes, ensuring scalability and real-time data processing. These powerful tools enable sophisticated querying features and tailored user experiences, enhancing both seller performance and customer satisfaction.

1. Core Challenges



Search Query	Search Query Score	Search Query Volume	Search Funnel - Impressions: Total Count	Brand count	Brand share	Search Funnel - Clicks: Total Count	Click rate	Brand count	Brand share	Same day delivery speed
bbq candle	1	2	38	2	5.26%	1	50%	1	100%	0
giant candles with lids	2	8	152	1	0.66%	4	50%	1	25%	3
terracotta garden light	3	5	95	3	3.16%	2	40%	1	50%	0
lavalis refill candle	4	1	19	1	5.26%	1	100%	1	100%	0
terracotta candle	5	38	721	14	1.94%	2	5.26%	0	0%	0
outdoor candle tray	6	21	369	10	2.71%	8	38.1%	0	0%	2
large candle bowl	7	13	286	9	3.15%	1	7.69%	0	0%	0
outdoor barbecue candles	8	19	361	8	2.22%	1	5.26%	0	0%	0
garden torch candles	9	20	502	8	1.59%	6	30%	0	0%	0
candles 15cm high	10	12	223	7	3.14%	8	66.67%	0	0%	6
outdoor candles for garden	11	560	12,580	6	0.05%	149	26.61%	0	0%	53
large bowl candle	12	13	248	6	2.42%	3	23.08%	0	0%	0
large swedish fire torch	13	7	383	6	1.57%	4	57.14%	0	0%	4
garden candles	14	444	10,372	5	0.05%	87	19.59%	0	0%	31

Figure 3.1.1 Overview of the Search Query Performance Report (REF: https://www.amalytix.com/en/knowledge/controlling/amazon-search-query-performance-report/)

Amazon uses cutting-edge technology and a strong infrastructure to efficiently process millions of search queries every second. The business uses advanced search algorithms that properly decipher user intent from short search queries by utilizing machine learning and natural language processing. Because of this skill, Amazon is able to extract important characteristics from user inputs, guaranteeing that search results are pertinent and customized to each user's demands. Furthermore, real-time data processing is made possible by Amazon's scalable cloud architecture, which enables the platform to react quickly to user requests even during times of heavy demand.

In 2022, Amazon launched a Search Query Performance Dashboard to boost seller performance and consumer satisfaction. This dashboard, which shows important metrics including impressions, clicks, add-to-cart actions, and transactions, gives vendors a thorough understanding of how customers search. Based on real user interactions, these metrics assist merchants in honing their marketing strategy and keyword selection. Additionally, by

examining user activity, Amazon tailors search results according to search history and previous purchases, personalizing the user experience. Amazon's competitive advantage in the e-commerce market is maintained by this combination of sophisticated algorithms, real-time processing, and customized experiences.

2. Search Engine

Search Engine: Amazon utilizes Elasticsearch and OpenSearch, powerful tools for full-text search, enabling efficient distributed querying across vast datasets. This implementation is crucial for managing the enormous scale of data and search queries that Amazon processes.

Elasticsearch is a distributed search engine built on Apache Lucene, designed to efficiently handle large data volumes and deliver fast search results. Its distributed architecture divides indices into shards, allowing each shard to operate independently. This enables parallel query processing across multiple servers, significantly enhancing performance and scalability. Such a setup is ideal for applications requiring real-time data access and analysis. The architecture supports horizontal scalability, meaning it can expand easily by adding more nodes to the cluster, which helps manage high query loads effective.

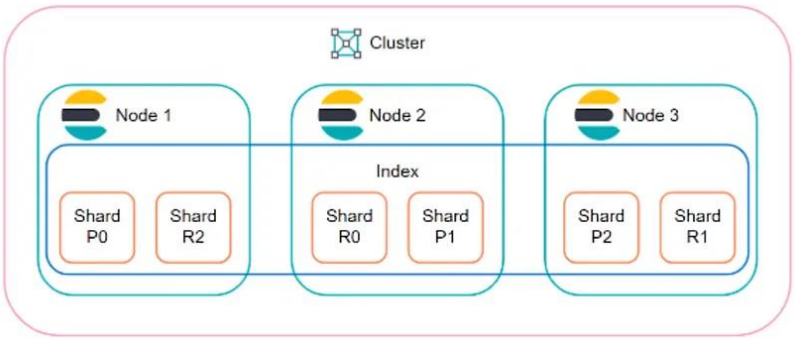


Figure 3.1.2 Elasticsearch Setup (REF: https://www.google.com/url?sa=i&url=https%3A%2F%2Fnidhig631.medium.com%2Fprimary-shards-replica-shards-in-elasticsearch-269343324f86&psig=AOvVaw1o94TIrJa1Ieyb_Iwg-jLz&ust=1733402681719000&source=images&cd=vfe&opi=89978449&ved=0CBQQjRxqFwoTCMiT8qqSjooDFQAAAAAdAAAAABAE **)**

OpenSearch provides powerful full-text search capabilities that enable efficient retrieval of relevant information from vast datasets. By leveraging advanced querying features such as full-text queries, filtering, sorting, and aggregations, OpenSearch enhances the overall search experience across various applications. Its scalable architecture ensures that it can handle increasing data volumes without compromising performance or accuracy. Open search

can be used in a E-commerce platform, customer can quickly find products through natural language searches with results filtered by categories or price ranges.

OpenSearch function:

A. Full-Text Search Capabilities

OpenSearch's full-text search functionality allows users to search through unstructured text data effectively.

- Query DSL (Domain-Specific Language): OpenSearch utilizes a powerful Query DSL that enables users to construct complex search queries. The DSL supports various query types:
- Match Query: Searches for documents containing specific terms or phrases. It can handle variations in word forms and synonyms, enhancing the search experience.
- Multi-Match Query: Similar to the match query but can be applied across multiple fields simultaneously, allowing broader searches.
- Phrase Match Query: Searches for an exact sequence of words within documents, ensuring that the words appear in the specified order.
- Fuzzy Matching: This feature allows for approximate matches based on edit distance, which is particularly useful for handling typos or variations in spelling.
- Term-Level Queries: These queries focus on exact matches within specified fields.
- Term Query: Matches documents containing a specific term in a particular field.
- Range Query: Finds documents where a field value falls within a specified range (e.g., numerical values or dates).

B. Complex Query Structures

OpenSearch supports complex query structures that enhance its search capabilities:

- Filtering: Users can apply filters to narrow down results based on specific criteria without affecting the scoring of documents. Filtering is essential for refining searches and ensuring that users receive the most relevant results based on their needs.
- Sorting: OpenSearch allows users to sort results based on various fields, such as relevance scores or specific attributes (e.g., price or date). This feature enhances user experience by presenting the most pertinent information at the top of the results list.

C. Aggregations

One of OpenSearch's standout features is its ability to perform aggregations on search results. Aggregations allow users to summarize data and gain insights into trends without needing separate analytical tools. They can be categorized into two main types:

- Bucket Aggregations: These group documents into buckets based on specified criteria. For example:
- Filters Aggregation: Allows users to create multiple buckets based on different filter criteria. This is useful for analyzing subsets of data within a larger dataset.
- Terms Aggregation: Groups documents by unique terms in a particular field, providing insights into the distribution of values.
- Metric Aggregations: These compute statistics over a set of documents within a bucket. Common metric aggregations include:
    1. Average: Computes the average value of a specified field across all documents in a bucket.
    2. Sum: Calculates the total value of a specified field across all documents.
    3. Count: Returns the number of documents in each bucket.

3.2 Scalability and Perfomance

Scalability is the capacity of a system to expand without sacrificing performance or to manage growing workloads. Applications that need to adapt to changes in user demand must have it. Performance, on the other hand, includes measurements like reaction time, throughput, and resource use and refers to how well a system performs under various conditions. Performance and scalability go hand in hand; a system that is properly scaled must also continue to operate at high levels. Here are 2 important points about scalability and performance:

1. Massive Traffic

    Massive Traffic refers to a significant volume of users or data attempting to access a system or network simultaneously, often leading to congestion and performance challenges. This concept is particularly relevant in contexts such as e-commerce platforms, social media, and online services, where spikes in user activity can occur during events like sales promotions or product launches.

Figure 3.2 Amazon US Sales Volume During Events (REF: https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.roadtovr.com%2Fquest-black-friday-sales-volume-amazon-us-2024%2F&psig=AOvVaw1r42oMPPMx28gzlzAMfmfX&ust=1733467877995000&source=images&cd=vfe&opi=89978449&ved=0CBUQjRxqFwoTCKjE6p2FkIoDFQAAAAAdAAAAABAK **)**

Applications like Amazon.com have a difficult time managing large volumes of visitors, particularly during periods of high demand for purchases or special occasions. Several tactics are used to efficiently handle this demand:

- Load balancing: This refers to the process of distributing incoming traffic across a number of servers so that none become overwhelmed, thereby spreading requests out evenly and ensuring responsiveness, which improves general system reliability during periods of increased traffic.
- Horizontal Scaling-Scale-Out: This is about adding other servers, not using more powerful ones. Normally, such a technique is referred to as vertical scaling. In the case of horizontal scaling, very huge, virtually unlimited growth can be achieved. Besides more possibilities regarding the volume of data being processed, this approach will enable effective workload distribution over various nodes and increase the system's fault tolerance.
- Content Delivery Networks: These will cache the information closer to the user geographically, hence reducing latency and accelerating content delivery. In such cases, even when the demand for information is high, it reaches the users quickly.
- Auto Scaling: Auto scaling in cloud computing is a mechanism of scaling resources dynamically by applications themselves as per the real-time traffic demand. For instance, at instances of sudden traffic spikes due to user activities, auto-scaling may be performed by automatically scaling server instances.

2. Perfomance Monitoring

Performance monitoring is a critical practice in IT operations that involves systematically tracking and assessing the performance of applications, infrastructure, and networks. This process ensures that systems operate efficiently and meet established service-level objectives (SLOs). Below is a detailed explanation of performance monitoring, including its definition, components, and significance.

Steady performance under heavy loads could be obtained by continuous monitoring. Performance monitoring could be properly done by observing the following:

- Real-time Analytics: Continuous monitoring of system performance metrics will yield critical data on response times, error rates, and server loads to predict impending problems before users are affected. Real-time analytics enable proactive application health management.
- APM: Application Performance Management extends visibility into application behavior, crossing a wide range of loads. This is done by finding, in real time, the problems behind slow transactions or errors and thus helping in their quick resolution.
- Load Testing: Performing the load tests from time to time simulates the times of high traffic in order to understand how well the system performs under stress. This proactive measure prevents defects in architecture during peak times.
- Resource Usage Monitoring: It would analyze the resources-CPU, memory, network bandwidth-that give an organization insight into the limits of any particular system. Hence, valuable inputs are provided for developing scaling strategies and performance optimization by making informed decisions.

CHAPTER IV
CONCLUSION AND SUGGESTION

4.1 Conclusion

The dominance of Amazon in the e-commerce landscape has been achieved through its novel use of application architecture, scalable infrastructure, and advanced search technologies. Using Elasticsearch and OpenSearch, for example, enables Amazon to process millions of queries with precision and speed, returning results that are relevant and timely for the user. Employing scalable strategies like load balancing, horizontal scaling, and content delivery networks ensures the performance of Amazon does not falter during periods of peak traffic. Further, the commitment to performance monitoring and real-time analytics gives the platform a clear look into the potential issues well in advance of them occurring.

Furthermore, the ability of Amazon to adapt and increase the scale of its architecture raised the bar for the entire industry in terms of reliability, scalability, and customer experience. By continually fine-tuning its system with advanced AI and machine learning technologies, Amazon has managed to remain at the forefront of innovation. This optimization, combined with smoothly integrating third-party services and seller tools, has kept the company ahead of competition and created a strong ecosystem both for customers and merchants.

4.2 Suggestion

Amazon should invest in the development of a hybrid search system that combines semantic search with real-time behavioral analysis. Amazon can provide even more intuitive and context-aware search results by integrating NLU with the insights from user interactions in real time. This will help not only in enhancing the accuracy and personalization of the search outcomes but also in customer engagement, reduced bounce rates, and conversions, while further solidifying its dominance in the e-commerce space.

BIBILIOGRAPHY

Reference:

[1] Carter, R. (2023, March 11). What is Amazon? Everything You Need to Know. Ecommerce Platforms. https://ecommerce-platforms.com/glossary/amazon [4/12/2024]

[2] Fries, T. (2023, July 27). Explaining the Amazon Search Query Performance Report in Detail. AMALYTIX. https://www.amalytix.com/en/knowledge/controlling/amazon-search-query-performance-report/ [4/12/2024]

[3] iCreativeLabs. (2024, January 10). Apa itu Performance Testing? Dan Apa Saja Toolsnya? https://icreativelabs.com/blog/apa-itu-performance-testing-dan-apa-saja-toolsnya [4/12/2024]

[4] Profil Kinerja Skalabilitas | AppMaster. (2023. November 6). AppMaster - Ultimate All-in No-code Platform. https://appmaster.io/id/glossary/profil-kinerja-skalabilitas

[5] Wright, G., & Ferguson, K. (2024, November 22). What is an application architecture? Search App Architecture. https://www.techtarget.com/searchapparchitecture/definition/application-architecture [3/12/2024]