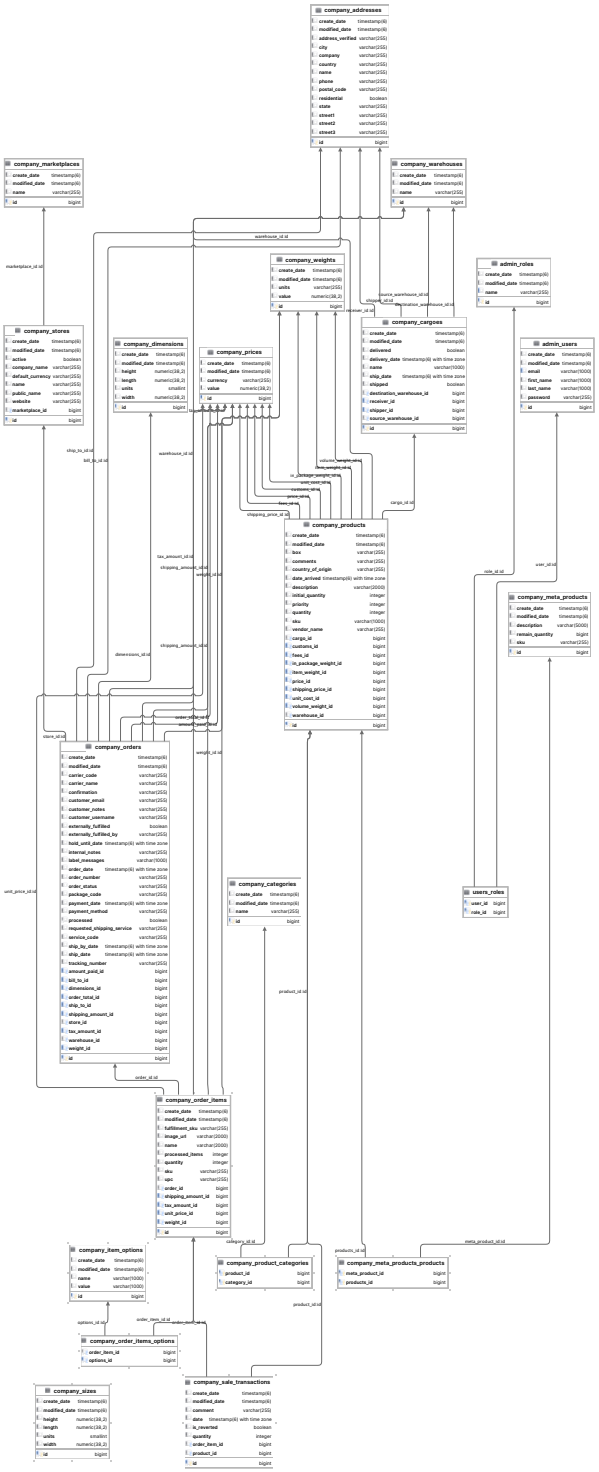
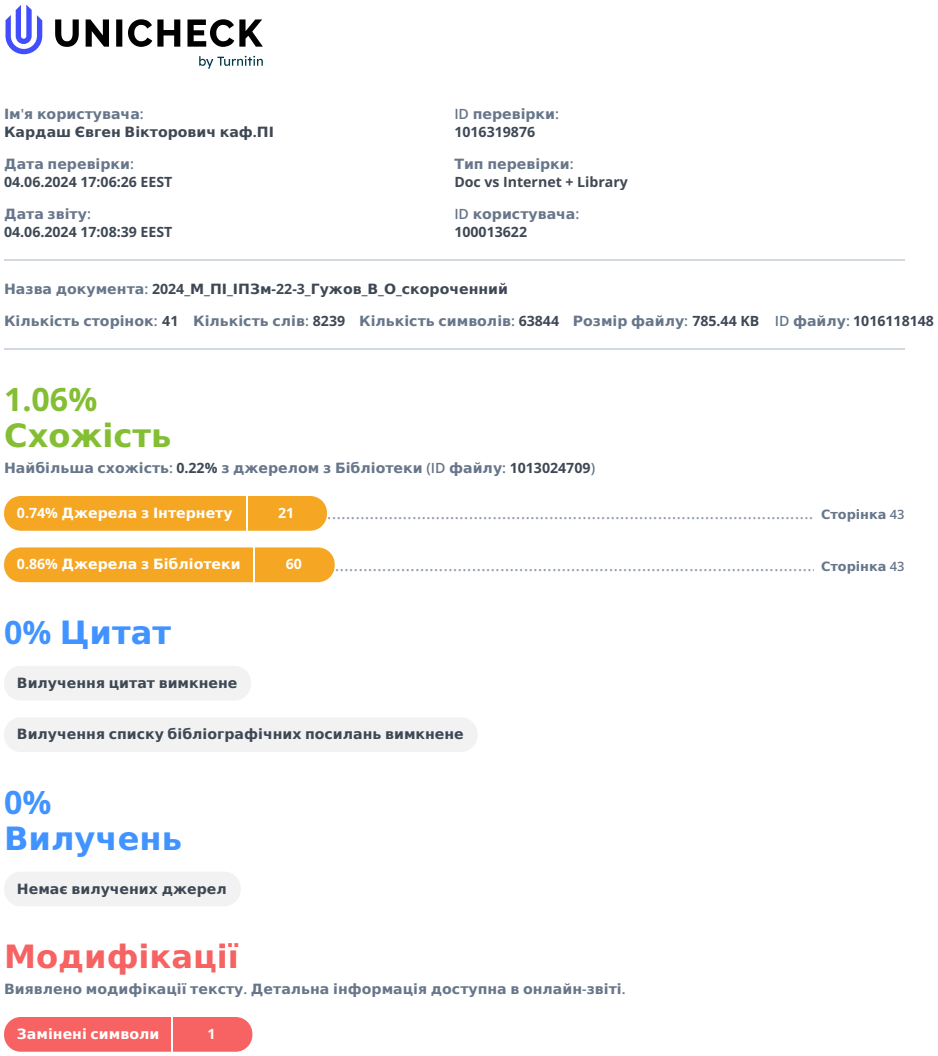
ДОДАТОК А

Схема бази даних



ДОДАТОК Б

Результати перевірки на унікальність тексту в базі ХНУРЕ



UNICHECK
by Turnitin

Ім'я користувача:
Кардаш Євген Вікторович каф.ПІ

ID перевірки:
1016319876

Дата перевірки:
04.06.2024 17:06:26 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
04.06.2024 17:08:39 EEST

ID користувача:
100013622

Назва документа: 2024_М_ПІ_ІПЗм-22-3_Гужов_В_О_скороченний

Кількість сторінок: 41 Кількість слів: 8239 Кількість символів: 63844 Розмір файлу: 785.44 KB ID файлу: 1016118148

1.06%
Схожість

Найбільша схожість: 0.22% з джерелом з Бібліотеки (ID файлу: 1013024709)

0.74% Джерела з Інтернету 21 Сторінка 43

0.86% Джерела з Бібліотеки 60 Сторінка 43

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 1

Рисунок Б.1 – Титульний аркуш звіту результатів перевірки на унікальність тексту в базі ХНУРЕ

ДОДАТОК В

Слайди презентації

Рисунок В.1 – Перший слайд презентації

Рисунок В.2 –Другий слайд презентації

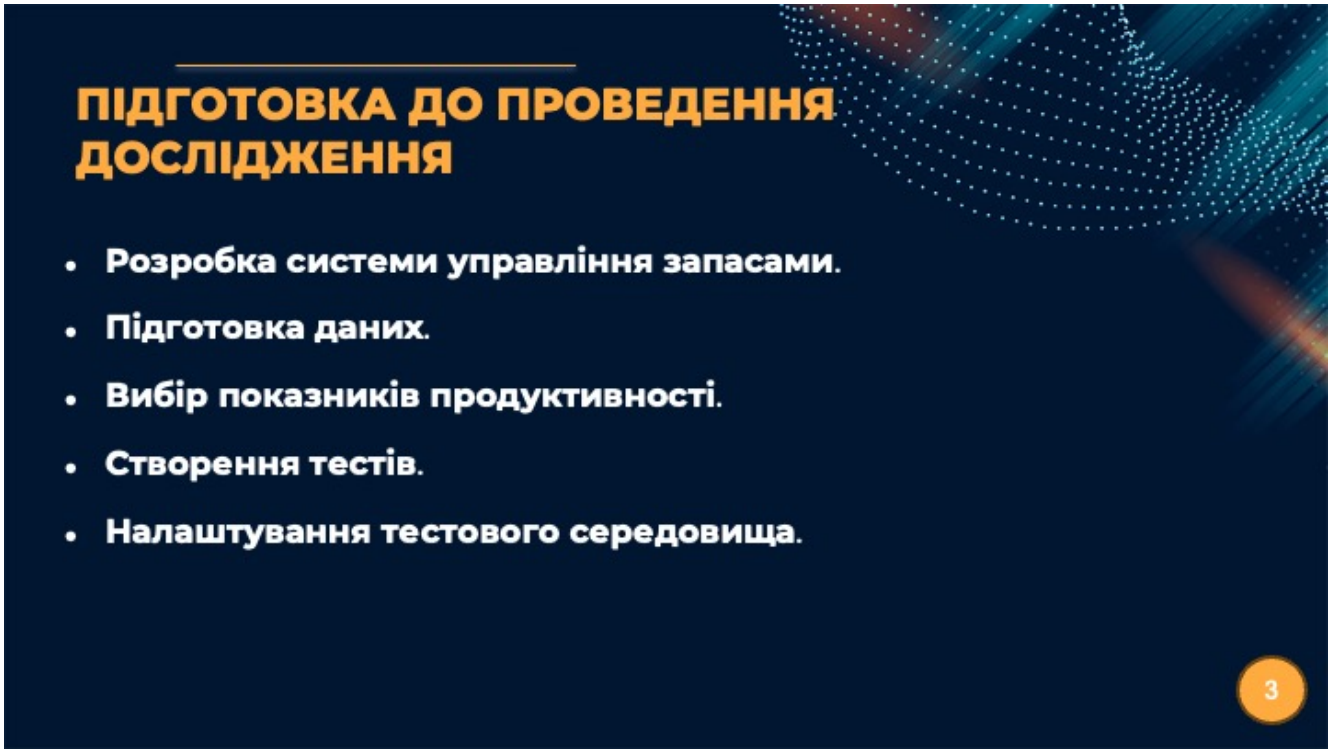


Рисунок В.3 – Третій слайд презентації

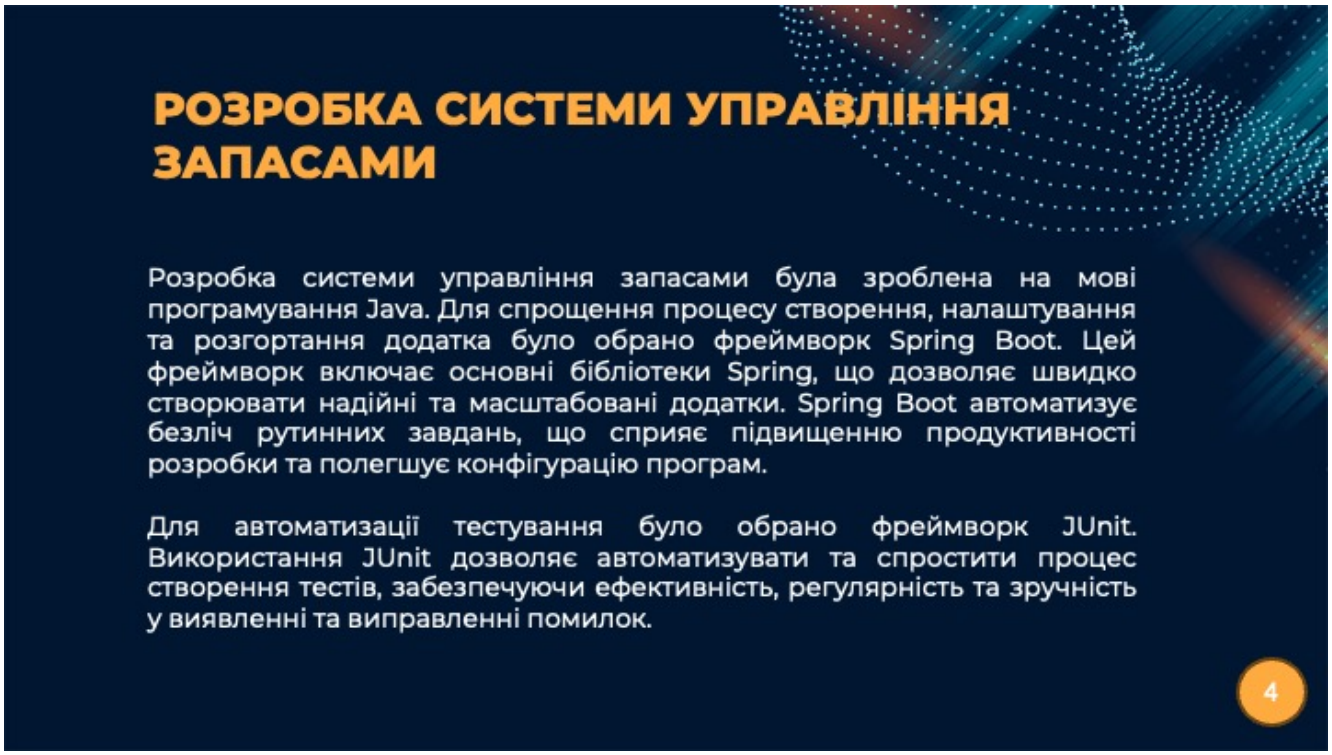


Рисунок В.4 – Четвертий слайд презентації

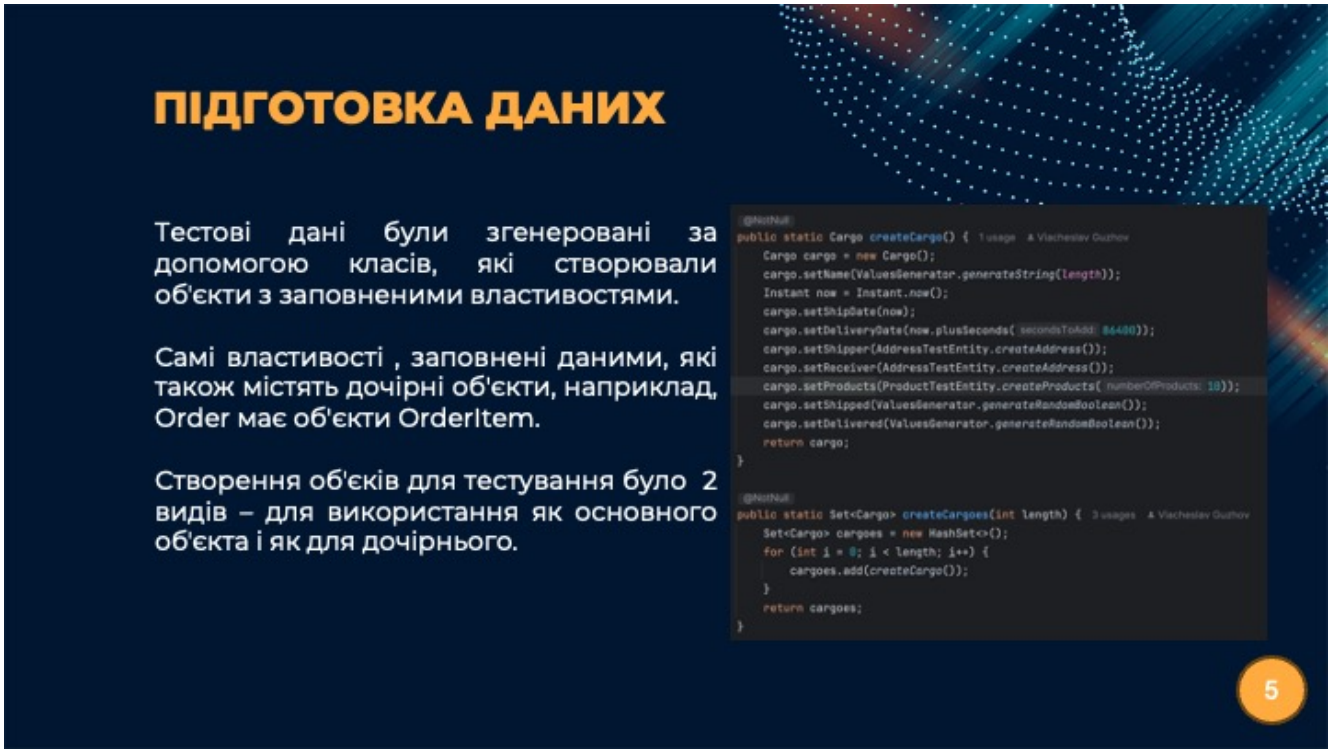


Рисунок В.5 – П’ятий слайд презентації

Рисунок В.6 – Шостий слайд презентації

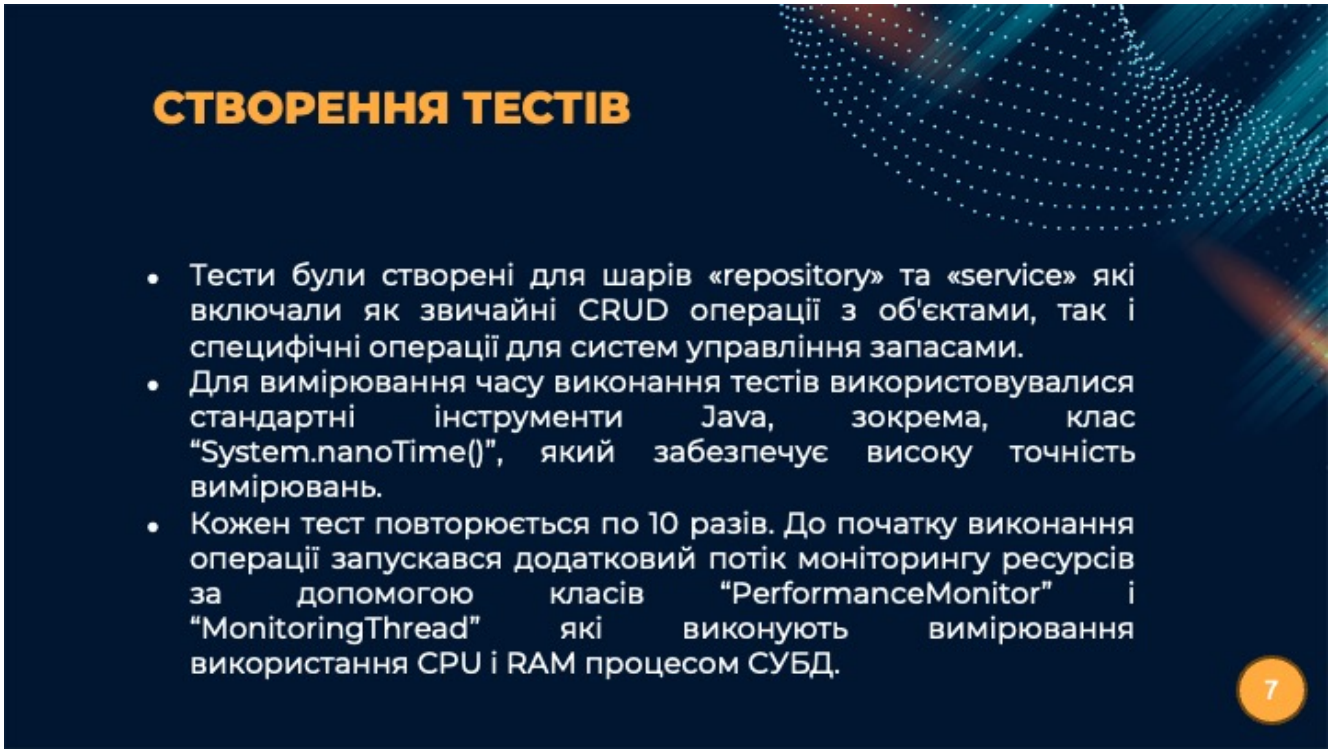


Рисунок В.7 – Сьомий слайд презентації

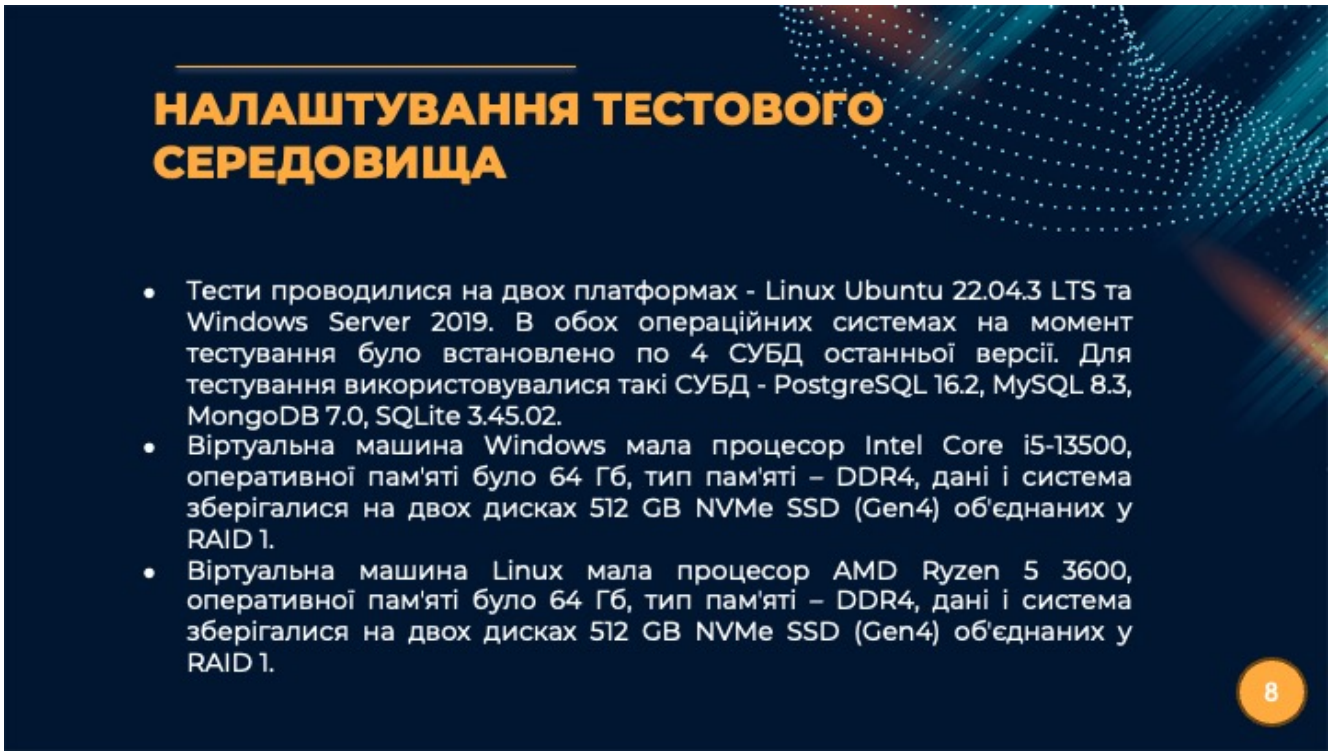


Рисунок В.8 – Восьмий слайд презентації

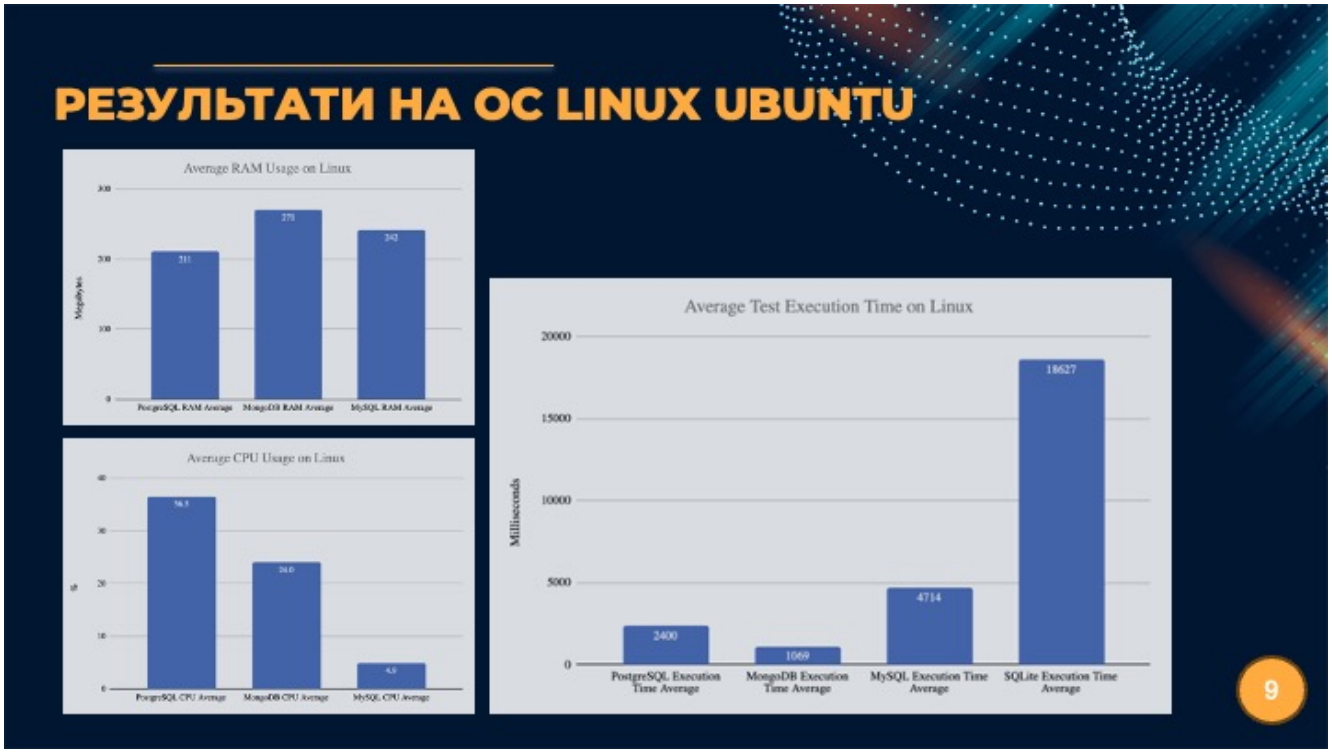


Рисунок В.9 – Дев’ятий слайд презентації

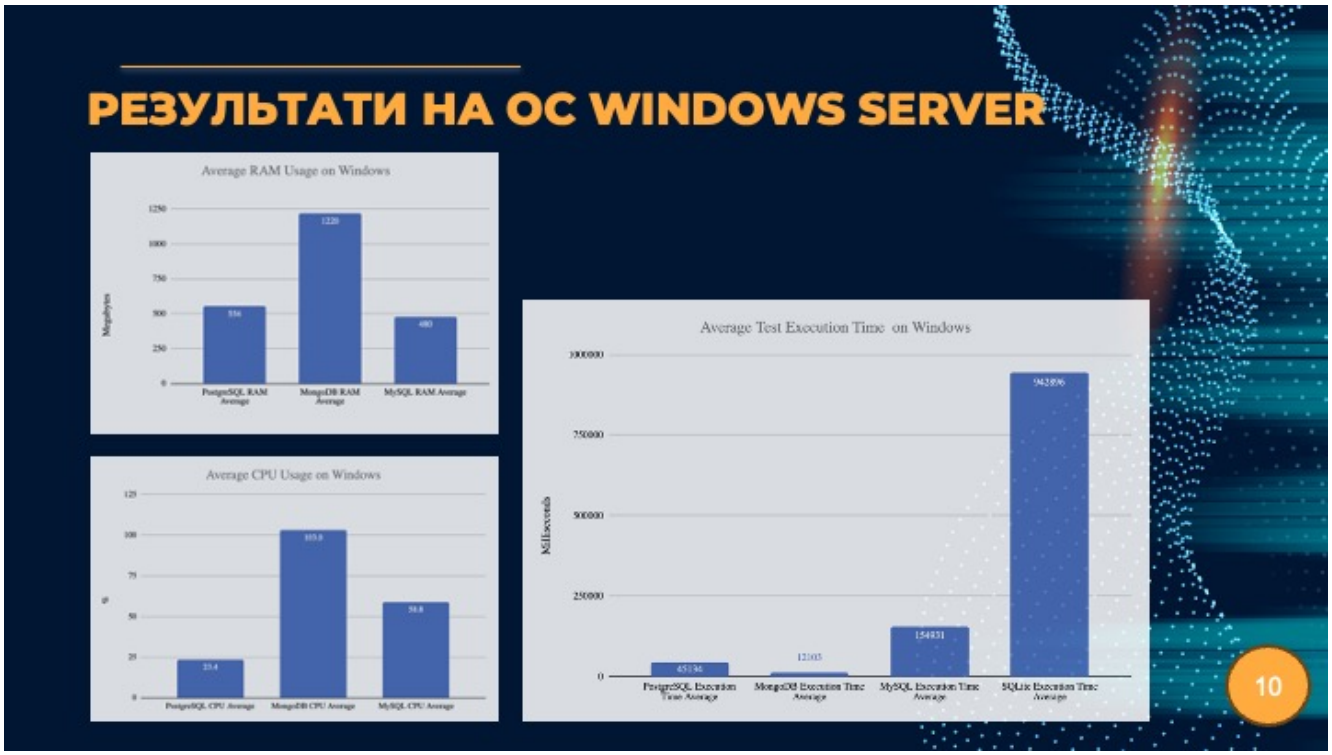


Рисунок В.10 –Десятий слайд презентації

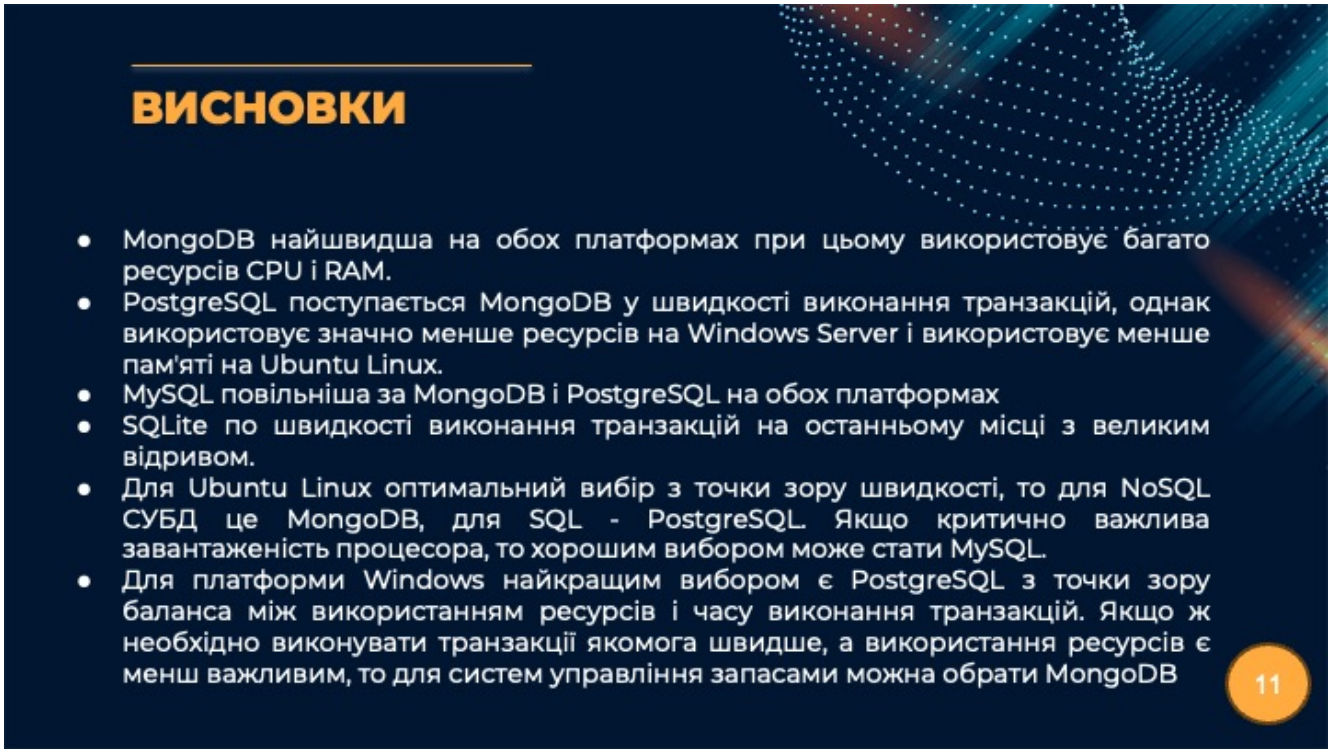


Рисунок В.11 – Одинадцятий слайд презентації

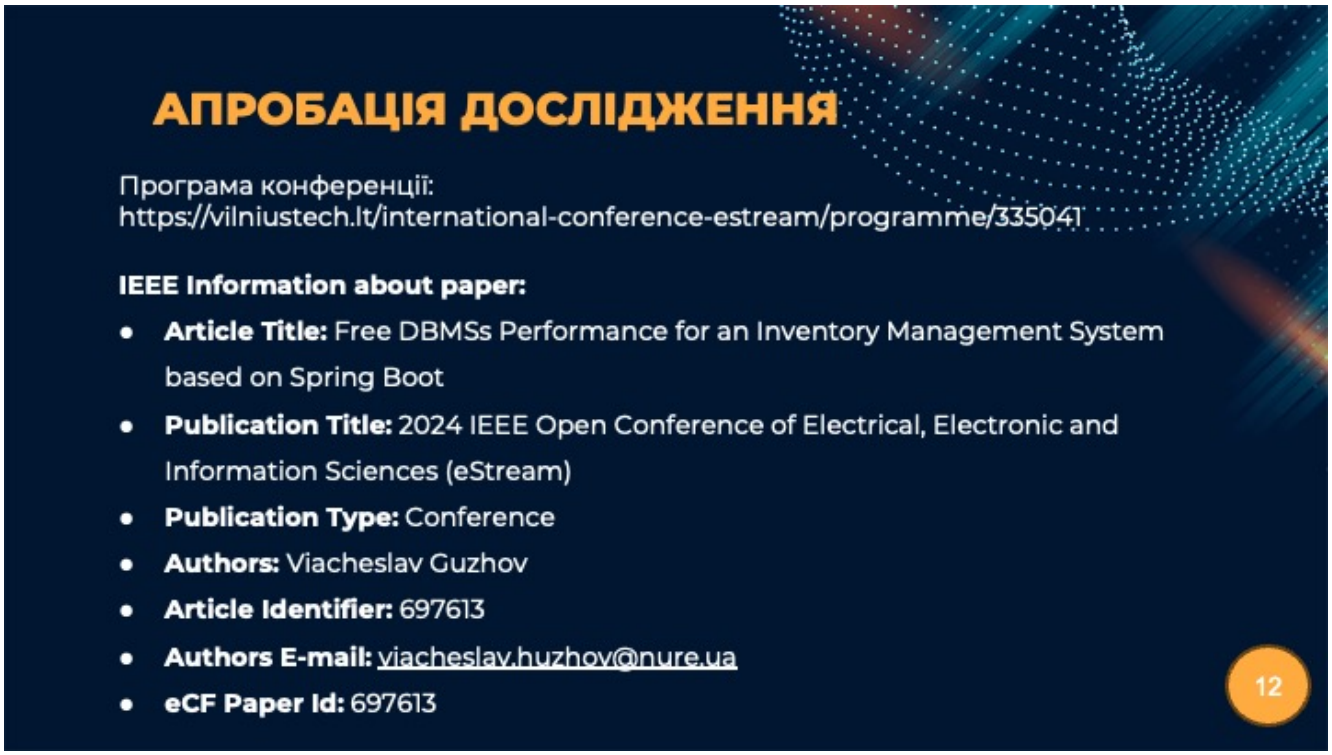


Рисунок В.12 – Дванадцятий слайд презентації

Рисунок В.13 – Тринадцятий слайд презентації

ДОДАТОК Г

Текст наукової публікації за темою кваліфікаційної роботи

Free DBMSs Performance for an Inventory Management System based on Spring Boot

Viacheslav Guzhov
Software Engineering Department
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
viacheslav.huzhov@nure.ua

Kyrylo Smelyakov
Software Engineering Department
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
kyrylo.smelyakov@nure.ua

***Abstract* – This research examines the integration of free Database Management Systems (DBMSs) with a Spring Boot-driven Inventory Management System. Evaluating MySQL, PostgreSQL, MongoDB, and SQLite, the study explores performance metrics, benchmarks, and practical scenarios, unveiling the strengths and weaknesses of each DBMS. Tailored for developers, architects, and decision-makers, the results aid in selecting an optimal free DBMS for improved system performance and informed decision-making within organizations adopting open-source solutions.)**

Keywords—Inventory Management System, DBMS, MySQL, PostgreSQL, MongoDB, SQLite, performance.

I. Introduction

In today's business landscape, where cost-effectiveness and flexibility are paramount, organizations are increasingly turning to free and open-source technologies. This paper explores the performance of free DBMSs with Spring Boot application which is also free and open-source and makes it possible to build enterprise-ready applications [1, 2]. Spring Boot also greatly simplifies the testing process [3]. Modern technologies even allow for the identification of goods by image [4] or the identification of employees using facial identification [5], which can be incorporated into modern inventory management systems.

As companies seek cost-effective solutions, the choice of an appropriate DBMS becomes critical. This study evaluates and compares the efficiency and reliability of popular free DBMSs — MySQL, PostgreSQL, MongoDB, and SQLite — in the context of an Inventory Management System. In this study, a NoSQL database is specifically added because this type of database has its advantages over SQL databases [6, 7]. It is known that MongoDB can execute transactions faster than SQL databases [8, 9] but besides the speed of transaction execution, there are other criteria for choosing a DBMS that will be considered in this study. The significance of this research lies in its potential to guide decision-makers in making informed choices aligned with their organization's specific needs and goals.

In the realm of inventory management, organizations often aim to develop in-house solutions to optimize resource utilization. Inventory Management Systems solve many of the problems faced by e-commerce companies [10, 11, 12, 13]. Leveraging the versatility of the Spring Boot framework, this research focuses on performance metrics with integrating free DBMSs into an Inventory Management System.

II. Methods and Materials

The next subsections in this section describe the creating application, prepare data for testing and evaluation metrics which have been used for experiments.

A. Creating an application

To test the DBMS's, an application was created in the Java programming language using the Spring Boot framework. The application includes all the basic entities for the inventory management system. Class diagram presented on Appendix 1.

The application has several layers of abstraction - a layer of entities that are database objects, a repository layer for interacting with the DBMS, and a service layer where business logic is written. The database schema is shown in Appendix 2.

Tests were written for the Repository and Service layers to verify the correctness of their operation.

B. Prepare data for testing

To prepare the test data, special classes were written to generate objects, the data of the object fields were filled with random data, and objects were created that had connections to the test objects, for example, for the Order object, OrderItem objects were created, and so on. The number of objects to be tested for each repository and service is shown in Table 1.

TABLE 1

NUMBER OF TEST ENTITIES

Entity Name	Quantity
Role	5000
User	5000
Address	20000
Cargo	300
Cartegory	5000
Dimensions	10000
ItemOption	20000
Marketplace	5
OrderItem	2000
Order	1000
Product	5000
SaleTransaction	10000
Store	10
Warehouse	5
Weight	20000

Рисунок Г.1 – Тези магістерського дослідження з конференції "2024 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)". Перша сторінка

C. Creating tests

For the service and repository layers, tests have been created that check the operation of functions that are necessary for the inventory management system. In each test function there is the part of the function in which the key calls take place, the performance of which we measure. These key calls is repeated 1000 times in test function to avoid anomalies in the results. For the same purpose, each test is run 10 times in both the repository and service layers.

D. Prepare an environment for running tests in Linux and Windows virtual machines

The tests were conducted on two platforms - Linux Ubuntu 22.04.3 LTS and Windows Server 2019. Both operating systems had 4 DBMSs of the latest version installed at the time of testing. The following DBMSs were used for testing – PostgreSQL 16.2, MySQL 8.3, MongoDB 7.0, SQLite 3.45.02.

E. Define performace metrics

To evaluate the performance of each DBMS, it was chosen to collect data on the execution time of operations and the CPU and RAM load. Monitoring of CPU and RAM utilization was built into the application in such a way that when the test is started, a separate thread is launched in which data on CPU and RAM utilization is read.

III. Experiment

The experiment consists of three parts.

1. Prepare an environment for running tests in Linux and Windows virtual machines.
2. Run tests and collect test results.
3. Compare results.

A. Prepare an environment for running tests in Linux and Windows virtual machines

The tests were conducted on two platforms - Linux Ubuntu 22.04.3 LTS and Windows Server 2019. Both operating systems had 4 DBMSs of the latest version installed at the time of testing. The following DBMSs were used for testing – PostgreSQL 16.2, MySQL 8.3, MongoDB 7.0, SQLite 3450200.

B. Run tests and collect test results

The testing was performed using the Junit 5.10 library and the results were recorded in a CSV file upon completion of the test. The test results are available at the link:

https://docs.google.com/spreadsheets/d/1nAsdjWZGHtQ08SsQMYyWvY_k4RSluuOOAt5eW5ShExY

IV. Experimental Results Analysis

First of all, it is worth noting that SQLite does not have a separate application compared to other DBMSs and the database is managed in the application itself without sending commands to a separate DBMS application compared to other DBMSs under consideration. Because of this, it is impossible to determine exactly how much SQLite uses CPU and RAM, the only thing that can be measured is the speed of query execution.

A. Analyzing the results on the Linux platform

Let's start by looking at the results on Linux. The RAM usage among these four DBMSs differs, but not much. On average, PostgreSQL consumes the least amount of memory, followed by MySQL and after them MongoDB. The graph with the results can be seen in Figure 1.

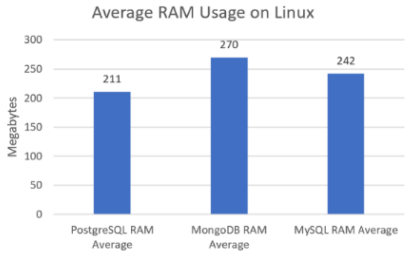


Figure 1: Average RAM usage on Linux

If we look at the results of CPU usage, we can see strong differences between these DBMSs. MySQL consumed the least, PostgreSQL and MongoDB consume significantly more CPU. The graph with the results can be seen in Figure 2.

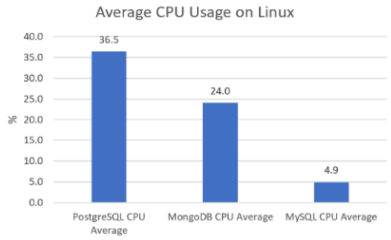


Figure 2: Average CPU usage on Linux

An important performance indicator is the speed of query execution, and here we can see that MongoDB is the leader in speed. If we take into account only relational databases, then PostgreSQL is the leader with a strong margin, followed by MySQL, which is 2 times slower. SQLite turned out to be very slow. The graph with the results can be seen in Figure 3.

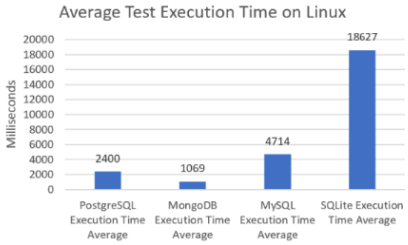


Figure 3: Average test execution time on Linux

Рисунок Г.2 – Тези магістерського дослідження з конференції “2024 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)”. Друга сторінка

B. *Analyzing the results on the Windows platform*

Figure 4 shows the average memory usage on the Windows platform. From this graph, you can see that MySQL consumes the least amount of memory, followed by PostgreSQL with slightly higher results. MongoDB consumes more than 2 times more memory than MySQL and PostgreSQL.

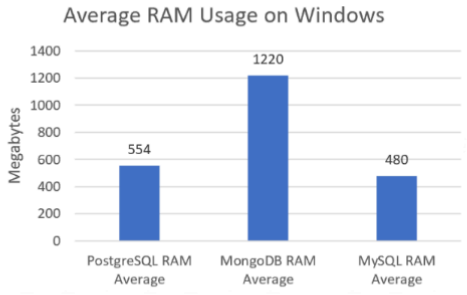


Figure 4: Average RAM usage on Windows

Figure 5 shows a graph comparing CPU usage during testing on the Windows platform. In terms of CPU usage, PostgreSQL uses the least CPU. MySQL uses more than 2 times more CPU than PostgreSQL. MongoDB uses the most CPU among all the contenders.

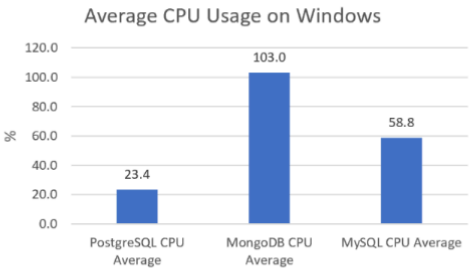


Figure 5: Average CPU usage on Windows

Figure 6 shows a graph of the average execution time of operations on the Windows platform.

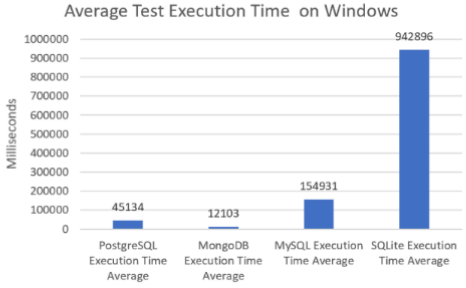


Figure 6: Average test execution time on Windows

In terms of query execution speed, MongoDB is the leader by a wide margin, using more memory and CPU, MongoDB executes queries much faster than other contenders. Among SQL databases, PostgreSQL was the fastest by a wide margin, followed by MySQL. SQLite turned out to be very slow and significantly inferior to other contenders.

V. Conclusion

Summarizing the performance of these four DBMSs on Ubuntu Linux, we can see that SQLite is a very slow DBMS and can hardly be considered a good performance DBMS. MongoDB was faster than the other candidates, but it consumes more CPU than MySQL. The choice of a DBMS depends on the requirements of the company. If we take the optimal choice in terms of speed, then for NoSQL DBMS it is MongoDB, for SQL - PostgreSQL. If CPU utilization is critical, MySQL can be a good choice.

To summarize, it seems that PostgreSQL is the best choice for the Windows platform in terms of resource utilization and transaction execution time. If it's necessary to execute transactions as quickly as possible and resource utilization is less important, the choice is MongoDB.

References

[1] Greg L. Turnquist, Dave Syer, Mark Heckler and Josh Long, "Learning Spring Boot 3.0: Simplify the development of production-grade applications using Java and Spring", *Packt Publishing*, Electronic ISBN: 9781803249896.

[2] M. Mythily, A. Samson Arun Raj, Iwin Thanakumar Joseph, "An Analysis of the Significance of Spring Boot in The Market", *2022 International Conference on Inventive Computation Technologies (ICICT)*, Nepal, 2022, DOI: 10.1109/ICICT54344.2022.9850910.

[3] M. Mythily, Anand Deva Durai C, V RaviTeja Kanakala, Iwin Thanakumar Joseph S, Radhika Nambiar, "An Extensive Review of Spring Boot Testing Based on Business Requirements of the Software", *2023 4th International Conference on Smart Electronics and Communication (ICOSEC)*, Trichy, India, 2023, DOI: 10.1109/ICOSEC58147.2023.10276283.

[4] K. Smelyakov, A. Chupryna, D. Sandrkin and M. Kolisnyk, "Search by Image Engine for Big Data Warehouse," 2020 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), Vilnius, Lithuania, 2020, pp. 1-4, DOI: 10.1109/eStream50540.2020.9108782.

[5] K. Smelyakov, A. Chupryna, O. Bohomolov and I. Ruban, "The Neural Network Technologies Effectiveness for Face Detection," 2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP), 2020, pp. 201-205, DOI: 10.1109/DSMP47368.2020.9204049.

[6] Geet Kiran Kaur, Sanjay Singla, Vishal Khawas, "Database Management System: A Study of Increasing Impact of NoSQL Databases", *2023 International Conference on Advanced Computing & Communication Technologies (ICACCTech)*, Banur, India, 2023, DOI: 10.1109/ICACCTech61146.2023.00067.

[7] Yishan Li, Sathiamoorthy Manoharan, "A performance comparison of SQL and NoSQL databases", *2013 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, Victoria, BC, Canada, 2023, DOI: 10.1109/PACRIM.2013.6625441.

[8] Seyyed Hamid Aboutorabi, Mehdi Rezapour, Milad Moradi, Nasser Ghadiri "Performance evaluation of SQL and MongoDB databases for big e-commerce data", *International Symposium on Computer Science and Software Engineering (CSSE)*, Tabriz, Iran, 2015, DOI: 10.1109/CSICSSE.2015.7369245.

[9] Petr Filip, Lukáš Čegan, "Comparison of MySQL and MongoDB with focus on performance", *2020 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*, Jakarta, Indonesia, DOI: 10.1109/ICIMCIS51567.2020.9354307.

[10] Zhang Guirong, Mu Yuxin, "Study on Auto Enterprise Inventory Management", *2011 International Conference on Information*

Рисунок Г.3 – Тези магістерського дослідження з конференції "2024 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)". Третя сторінка

Management, Innovation Management and Industrial Engineering, Shenzhen, China, 2011, DOI: 10.1109/ICIII.2011.191.

[11] Ashley Marie N. Margate, Ma. Cathyrine F. Ravina, Jeric James G. Pido, Michael N. Young, "Seiton: A Mobile Inventory Management System Application for Micro, Small and Medium-sized Enterprise", *2020 IEEE 7th International Conference on Engineering Technologies and Applied Sciences (ICETAS),* Kuala Lumpur, Malaysia, 2020, DOI: 10.1109/ICETAS51660.2020.9484183.

[12] Xiao-Rong Lei, Da-Xi Wang, "Inventory Management System of Auto Parts Enterprises Based on the Agile Supply Chain", *2012 Second International Conference on Instrumentation, Measurement, Computer, Communication and Control*, Harbin, China, 2012, DOI: 10.1109/IMCCC.2012.207.

[13] Trio Adiono, Hans Ega, Hans Kasan, Carrel Suksmandhira Harimurti, "Fast Warehouse Management System (WMS) using RFID based goods locator system", *2017 IEEE 6th Global Conference on Consumer Electronics (GCCE),* Nagoya, Japan, 2017, DOI: 10.1109/GCCE.2017.8229410.

Рисунок Г.4 – Тези магістерського дослідження з конференції “2024 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)”. Четверта сторінка

Appendix 1

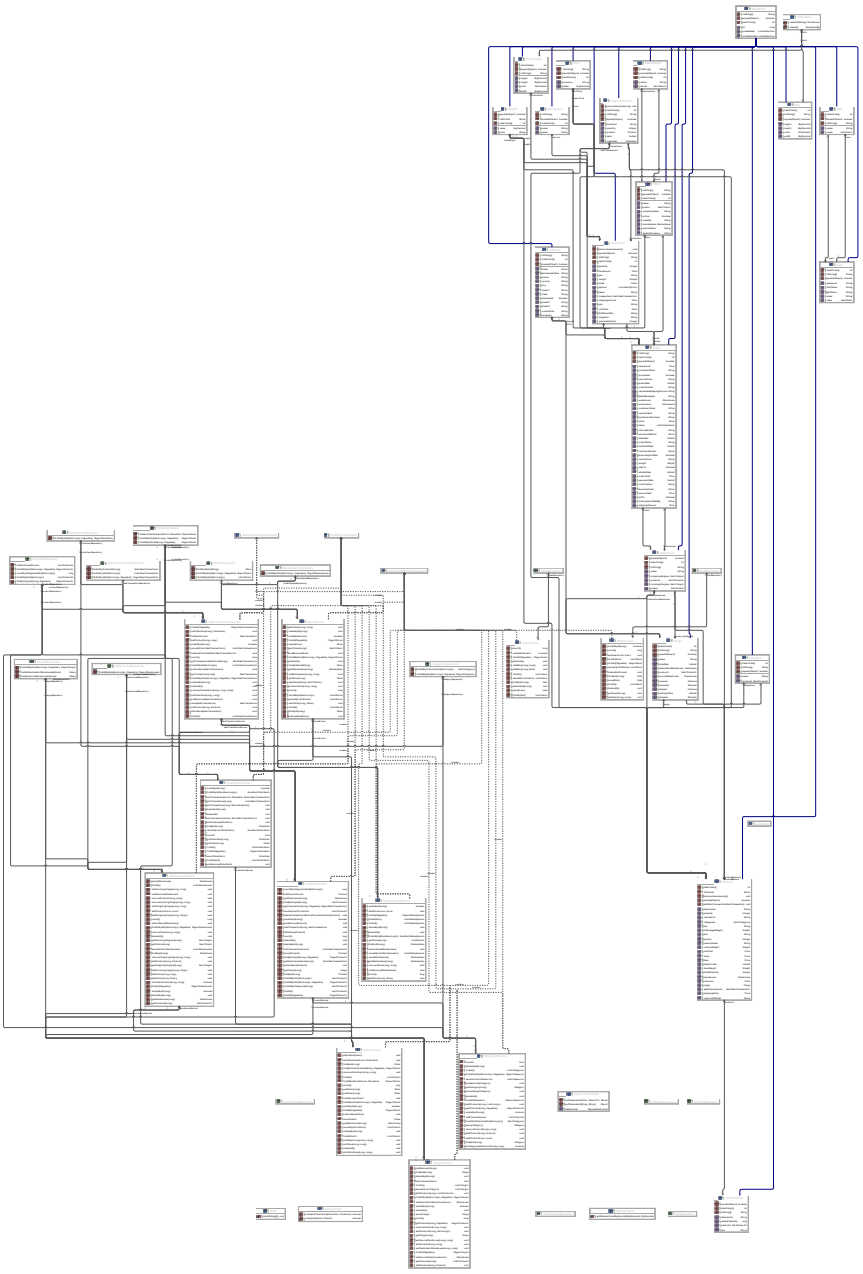


Рисунок Г.5 – Тези магістерського дослідження з конференції “2024 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)”. П’ята сторінка

Appendix 2

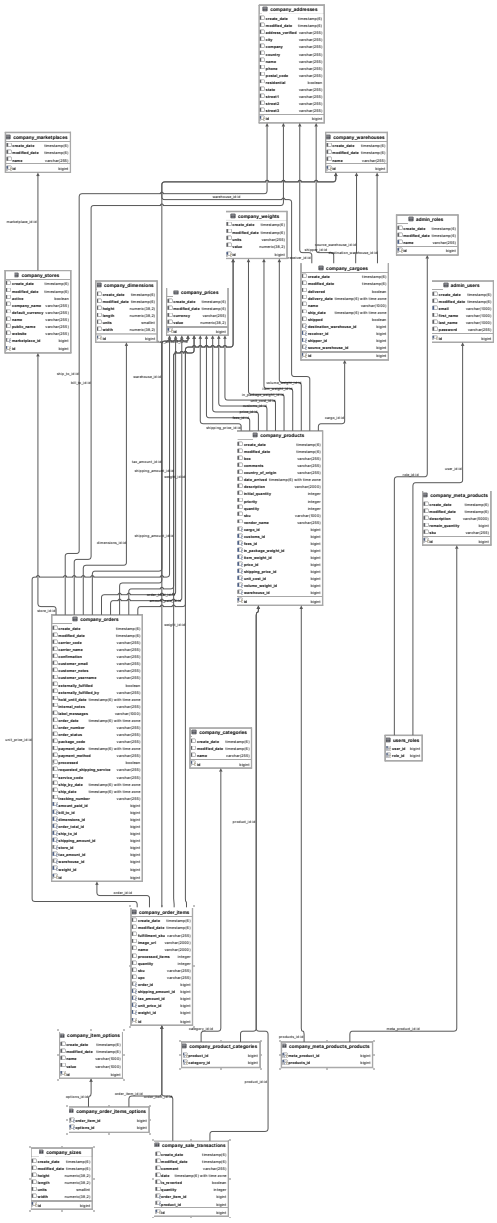


Рисунок Г.6 – Тези магістерського дослідження з конференції “2024 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)”. Шоста сторінка

ДОДАТОК Д

Експертний висновок результатів перевірки кваліфікаційної роботи на відповідність оформлення вимогам ДСТУ 3008: 2015

Експертний висновок результатів перевірки кваліфікаційної роботи

студент (посада) | програмної інженерії (кафедра) | ІПЗм-22-3 (група)

Гужов В. О.
(прізвище, ін'я, по батькові)

Зауваження

Пункт ДСТУ 3008-2015	Зміст пункту	Сторінка кваліфікаційної роботи
1	2	3
	7.1 Загальні положення	
	7.3 Нумерація сторінок звіту	
	7.4 Нумерація розділів, підрозділів, пунктів, підпунктів	
	7.5 Рисунки	
	7.6 Таблиці	
	7.7 Переліки	
7.7.2	Якщо подають переліки одного рівня підпорядкованості, на які у звіті немає посилань, то перед кожним із переліків ставлять знак «тире». Якщо у звіті є посилання на переліки, підпорядкованість позначають малими літерами української абетки, далі — арабськими цифрами, далі — через знаки «тире». Після цифри або літери певної позиції переліку ставлять круглу дужку.	10, далі за текстом
	7.8 Примітки	
	7.9 Виноски	
	7.10 Формули та рівняння	
	7.11 Посилання	
	7.13 Список авторів	
	7.14 Скорочення та умовні познаки	
	7.15 Додатки	
Методичні вказівки до виконання кваліфікаційної роботи магістра… **ЗАТВЕРДЖЕНО** кафедрою ПІ протокол № 5 від 13.11.2023р. 3.2 Оформлення пояснювальної записки згідно з ДСТУ 3008:2015 Звіти у сфері науки і техніки. Структура та правила оформлювання. **Шаблон** затверджений засіданням кафедри №3 від 16.10.2023.	Рисунок повинен розміщуватися одразу після його згадування у тексті, або на наступній сторінці. **Під рисунком повинен бути підпис із словом Рисунок, порядковим номером цього рисунку, через тире з великої літери – назва рисунку та в круглих дужках вказується джерело з якого взятий цей рисунок, або то, що його виконано самостійно.**	19, далі за текстом
Методичні вказівки до виконання кваліфікаційної роботи магістра… **ЗАТВЕРДЖЕНО** кафедрою ПІ протокол № 5 від 13.11.2023р. 3.2 Оформлення пояснювальної записки згідно з ДСТУ 3008:2015 Звіти у сфері науки і техніки. Структура та правила оформлювання. **Шаблон** затверджений засіданням кафедри №3 від 16.10.2023.	Назву таблиці друкують з великої літери і розміщують над таблицею з абзацного відступу та в круглих дужках вказується джерело з якого взята ця рисунокаблиця. ПРИКЛАД: шаблон, стор.15	33

Експерт _____ (підпис) Вадим НЕЧВОЛОД (прізвище, ініціали)

05.06.2024

Рисунок Д.1 – Експертний висновок результатів перевірки кваліфікаційної робот