ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 13, Issue 3, December 2024

USABILITY AND PERFORMANCE COMPARISON: IMPLEMENTATION OF TIBERO AND ORACLE DATABASES IN THE CONTEXT OF CAMS SOFTWARE DEVELOPMENT

Komang Yuli Santika[1], Dandy Pramana Hostiadi[2], Putu Desiana Wulaning Ayu[3]

[1] *Magister Program*, Department of Magister Information Systems, Institut Teknologi dan Bisnis STIKOM Bali, Indonesia
[2,3] *Department of Magister Information Systems*, Institut Teknologi dan Bisnis STIKOM Bali, Indonesia

email: 222012009@stikom-bali.ac.id[1], dandy@stikom-bali.ac.id[2], wulaning_ayu@stikom-bali.ac.id[3]

Abstract

In software development, selecting the right database is crucial for ensuring application performance and efficiency. Oracle is a leading enterprise database known for its reliability, but there is a need for more efficient and cost-effective alternatives. Tibero, developed by TmaxSoft, has emerged as a potential alternative, offering competitive performance and a smooth transition from Oracle. This study compares the performance of Tibero 7 and Oracle 12c in the Consumer Asset Management System (CAMS) application developed by Telkomsigma. The evaluation is conducted using Performance Testing to measure processing time, CPU, and memory usage, as well as the System Usability Scale (SUS) to assess usability. The results show that Oracle excels in Transactions Per Second (TPS) and completion time, while Tibero is more efficient in resource usage. The SUS scores also indicate that Oracle has higher usability.Overall, Oracle remains the best choice for enterprise applications, but Tibero provides an attractive alternative for specific needs. This study offers guidance for software developers in selecting the appropriate database based on technical performance and user experience.

Keywords : *Database, Tibero, Oracle, DBMS, Performance Testing*

Received: 11-07-2024 | **Revised:** 02-09-2024 | **Accepted:** 22-09-2024
DOI: https://doi.org/10.23887/janapati.v13i3.82519

INTRODUCTION

In the world of software development, database systems play a crucial role in supporting the functionality and efficiency of applications. Enterprise software, often referred to as business software, is a type of software specifically designed to meet the complex needs of organizations or large companies [4]. This software is designed to address unique challenges, such as managing daily operations and efficiently handling resource management, faced by large enterprises [1].

Oracle is one of the most popular enterprise databases and has long been the primary choice in many industries to support critical business applications. The reliability and flexibility offered by Oracle have made it a common choice in the software world [7]. However, as the information technology industry evolves, there is a growing need for more reliable, efficient, and cost-effective databases. Tibero, a relational database management system (DBMS) developed by TmaxSoft from South Korea, has emerged as a potential alternative to meet these needs [17]. Tibero is designed to meet the data management requirements of large enterprises by offering key features such as high performance, reliability, and scalability, which are crucial in business environments that require large-scale data storage and management [17]. As a relational DBMS, Tibero supports a relational data model that uses tables to store data and SQL (Structured Query Language) to access and manipulate data, thus facilitating companies in organizing and managing their information [9].

Telkomsigma, a subsidiary of Telkom Indonesia, focuses on providing information and communication technology (ICT) services by offering various IT solutions for multiple industries, including telecommunications, finance, banking, e-commerce, and government. In the development of enterprise software, Telkomsigma uses various databases such as Oracle, Microsoft SQL Server, PostgreSQL, and MySQL, with the choice of the database tailored to the specific business needs of each customer. However, this creates challenges since most products developed by Telkomsigma rely heavily on databases (PL SQL), and switching

databases can cause significant changes in the software process [19]. One of the products used as a research object is CAMS (Consumer Asset Management System), an application for managing loan customer data in banks. This application often faces issues related to database licensing, ease of use, and performance [19]. Therefore, research is needed to optimize the use of alternative databases other than Oracle, such as Tibero, to address these issues [17].

This study introduces a new comparative analysis between Tibero 7 and Oracle 12c in the context of the Customer Asset Management System (CAMS), which is one of the banking application products developed by Telkomsigma. While Oracle's dominance in the enterprise database market has been widely discussed, Tibero's potential as an alternative, particularly within Telkomsigma's ecosystem, remains underexplored. This study uniquely positions itself by focusing on performance testing and System Usability Scale (SUS) evaluation to provide a comprehensive assessment of both databases. The results of this comparison are expected to serve as a reference for Telkomsigma in selecting the most suitable database for their application development needs.

The choice of Tibero as an alternative database in this study is based on its similarity in data structure to Oracle, which allows for a smoother transition with minimal changes to both the database and the application, especially in business processes that use Stored Procedures (PL SQL) [26]. The methods used in this study to analyze the technical comparison between Oracle and Tibero are Performance Testing and the System Usability Scale (SUS). Performance testing provides quantitative data on how a system operates under specific conditions, such as response time and reliability, which are critical to ensuring the smooth and efficient operation of the system [13]. Meanwhile, SUS is an effective tool for measuring usability from a user perspective when using each database [4]. These two methods provide a comprehensive overview of technical performance and system usability; performance testing focuses on technical aspects, while SUS highlights user experience.

By using this approach, this study aims to provide better insights for software developers in choosing the right database, as well as contributing to the development of software technology and the information technology industry as a whole. The methods used, namely Performance Testing and the System Usability Scale (SUS), provide quantitative and qualitative data on the technical performance and usability of the system from the user's perspective, which can serve as a guide in data-driven decision-making for future software development [2].

METHOD

Performance Data Analysis (Performance Testing)

In the performance data analysis process, according to existing data collection, take data from PROCESS_MONITOR, calculate the processing time for each module:

$$\text{RESULT} = \text{END\_PROCESS} - \text{START\_PROCESS} \quad (1)$$

From the formula (1) calculations, performance comparison results will be obtained:

1. Analyze the processing time for each module in the Tibero 7 and Oracle 12c databases.
2. Compare the average processing time between the two databases to determine the database with better performance.
3. Analyze CPU, Memory and Storage usage of each database.

Usability Data Analysis (SUS)

In the performance data analysis process according to the existing data collection, the SUS Score calculation becomes. Each questionnaire answer is calculated based on the SUS rules:
Odd questions: (Q1-1) + (Q3-1) + (Q5-1) + (Q7-1) + (Q9-1)
Even questions: (5-Q2) + (5-Q4) + (5-Q6) + (5-Q8) + (5-Q10)
Average SUS Score:
Calculating the average SUS score of all respondents :

$$U = \frac{\sum R \, x \, 2.5}{n} \quad (2)$$

Where ∑R is the total number of answers, and n is the number of respondents.

The analysis flow of this research consists of several stages of activities and their explanations are in Figure 1.

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 13, Issue 3, December 2024

Criteria for Respondents in the SUS Questionnaire

The respondents involved in filling out the SUS (System Usability Scale) questionnaire consist of professionals with a deep understanding of software development, testing, and analysis, as well as database performance. Respondents are drawn from various positions, including:

1. Programmer is Individuals responsible for developing software code and understanding the impact of database performance on applications.
2. Quality Assurance (QA) is Professionals involved in software quality testing to ensure that the software meets established standards.
3. Business Analyst (BA) is Those who analyze business requirements and ensure that the developed technological solutions align with those needs.
4. Junior Tester is Entry-level testers who perform software testing to identify bugs and other issues.
5. Programmer Analyst is Professionals who combine the roles of programmer and system analyst, understanding both coding and system analysis.
6. Senior Consultant is Experts with extensive experience who provide strategic advice related to complex business solutions.
7. Senior Business Analyst is Individuals who handle more complex projects and collaborate with senior management for strategic solutions.
8. System Analyst: Professionals responsible for analyzing and designing IT solutions that meet business needs.
9. Specialist is Individuals with specialized expertise in specific fields, such as cybersecurity or databases.
10. Tester is Software testers involved in thorough testing to ensure product quality before release.

These respondents are selected because they have relevant experience and knowledge about application performance and database usage, which is crucial for evaluating application usability using the SUS scale.

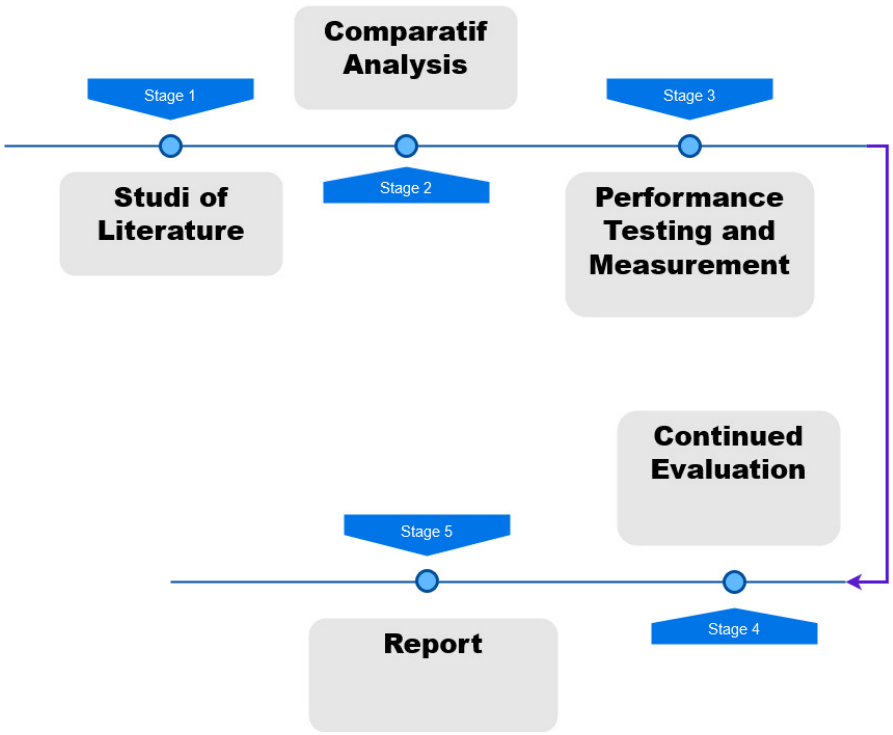


Figure 1. Research Flow

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 13, Issue 3, December 2024

Study of literature

The initial step is to conduct an in-depth literature study on Oracle Database, Tibero Database, as well as previous research relevant to software performance optimization. The results of this literature study help understand existing frameworks and knowledge regarding the performance of these two databases.

Comparative Analysis

A comprehensive analysis of Oracle and Tibero Database features and performance will be performed. This analysis includes a functional comparison between Tibero 7 and Oracle 12c, a technical comparison in terms of CPU, memory and storage resource usage which will be tested in the performance testing stage.

Performance Testing and Measurement

Testing of the CAMS application on Telkomsigma was carried out on both databases, using measurement methods relevant to performance. Benchmarking is used to measure performance by recording the processing time of modules in each database. The results of these recordings are then compared to determine whether tuning is needed in the database. After tuning is carried out, retesting is carried out to compare the processing time results before and after tuning.

Performance Measurement Process

Initial Measurement: The measurement process starts from the user who processes the modules of the application that uses the Oracle and Tibero databases. Processing time was recorded for each database. Results Analysis: The processing time recording results were compared to determine whether there were significant differences in the performance of the two databases. Database Tuning: If results indicate the need for improvement, tuning is performed on the database that requires performance improvement. Retesting: After tuning, the module is retested and runtime recorded again to see any performance improvements that occur.

Continued Evaluation

Follow-up evaluation is carried out after fixes and adjustments are implemented to ensure that the changes made have significantly improved application performance.

Report

Complete reports are prepared regarding research results, guidelines for using the database, as well as steps that have been taken to optimize application performance. This documentation will be a practical guide for software developers.

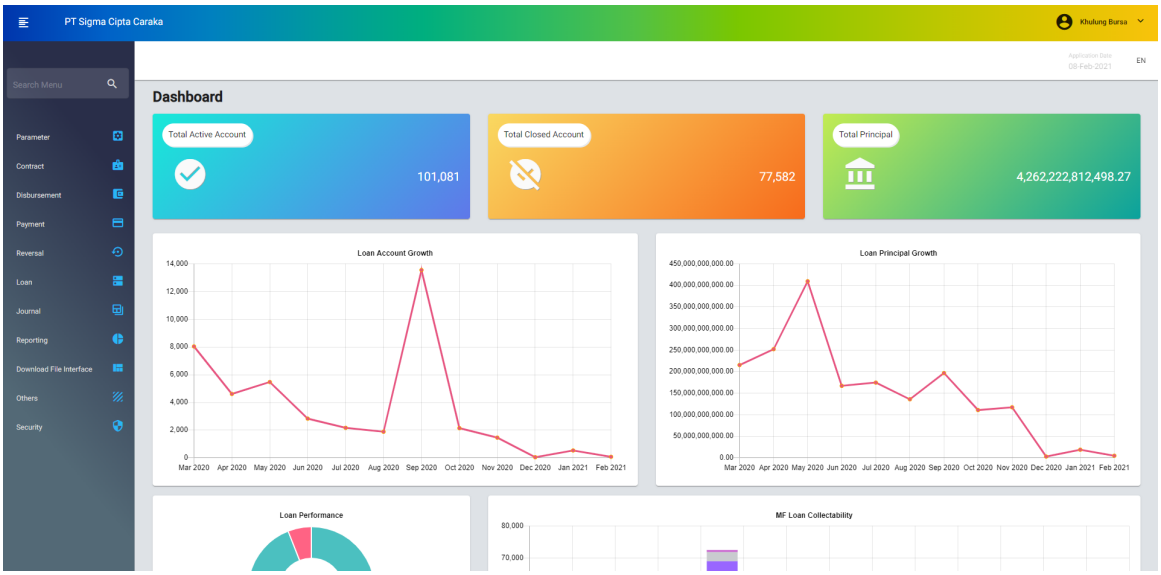


Figure 2. CAMS Application

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 13, Issue 3, December 2024

RESULT AND DISCUSSION

In this Performance Test, the Test Application used is:

1. Tibero Studio and Oracle SQL Developer were used for all performance tuning of Database Server and Application Server.
2. Tibero Studio and Oracle SQL Developer to analyze all queries sent by the Application to the database and manipulate the queries.
3. Management Console to monitor memory usage on the server.
4. System Monitor is used to monitor CPU usage on the server
5. Configure the Arium CAMS NG Tibero Application Performance Test requirements in Table 1 and the database in Table 2.
6. Configure the Arium CAMS NG Oracle Application Performance Test requirements in Table 3 and the database in Table 4.

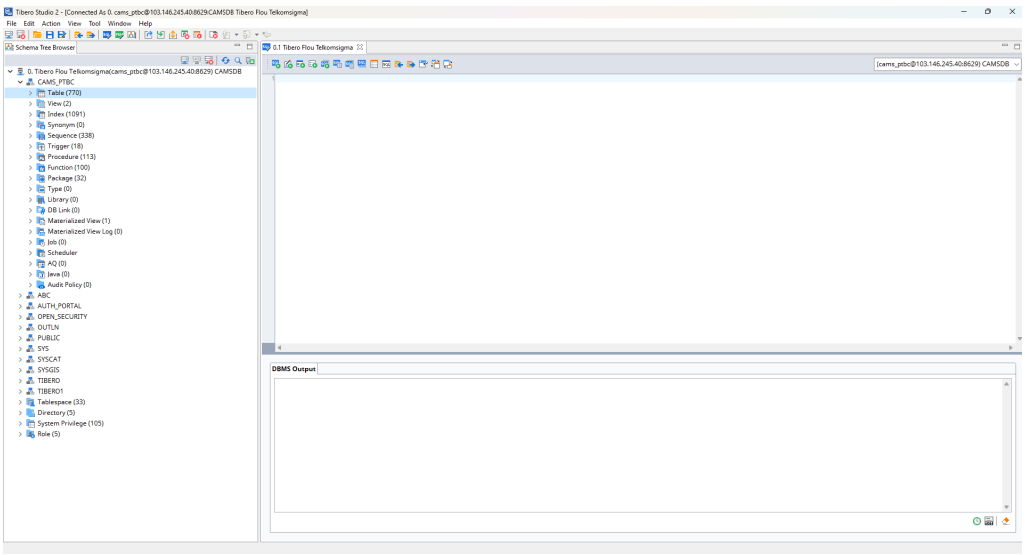


Figure 3. Tibero Studio 7

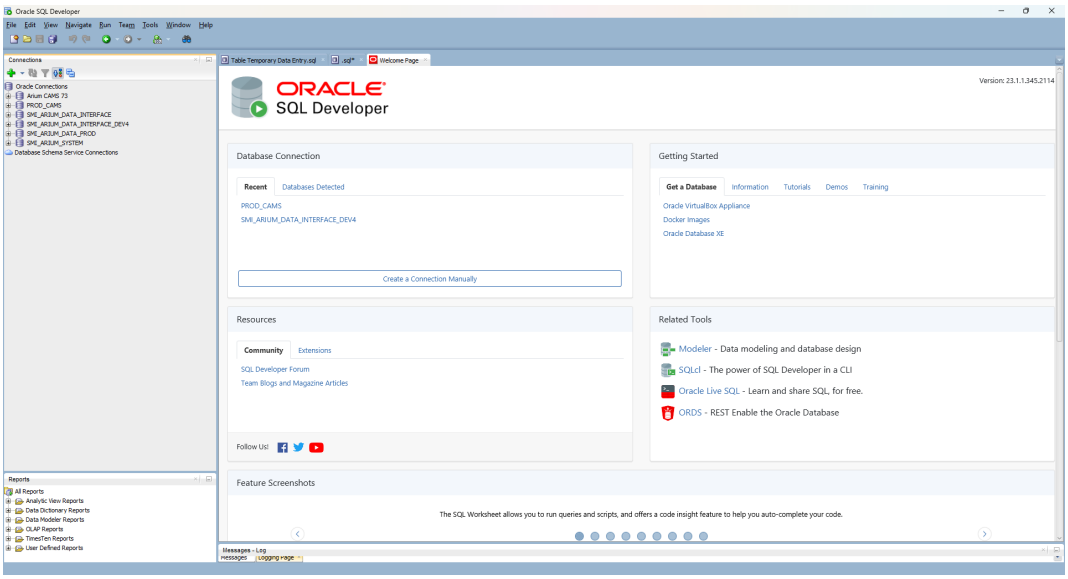
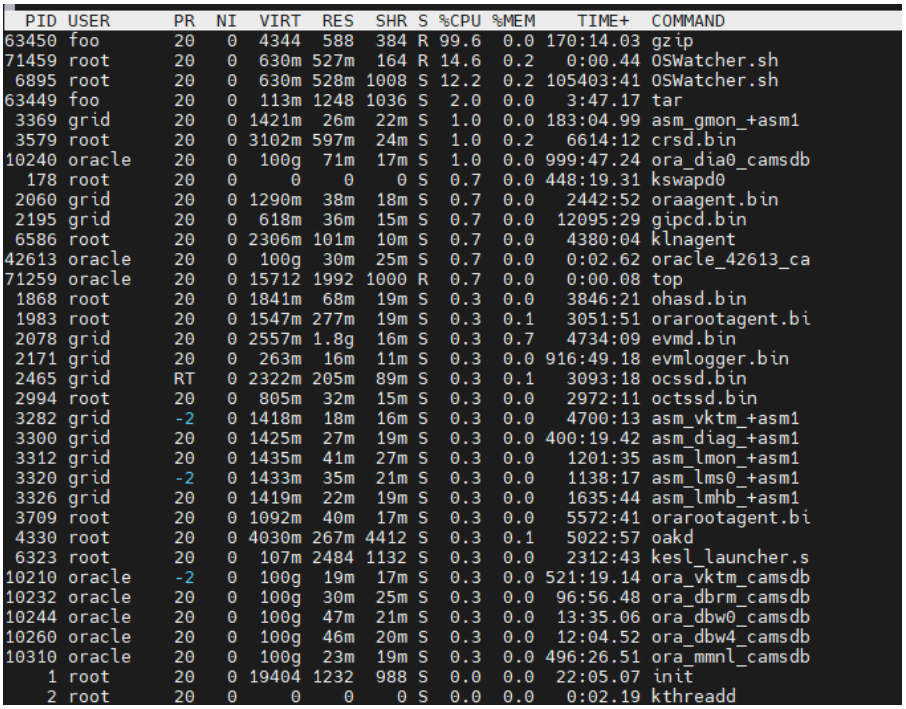


Figure 4. Oracle SQL Developer

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 13, Issue 3, December 2024



```
  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
63450 foo       20   0    4344    588    384 R  99.6   0.0 170:14.03 gzip
71459 root      20   0  630m 527m    164 R  14.6   0.2   0:00.44 OSWatcher.sh
 6895 root      20   0  630m 528m   1008 S  12.2   0.2 105403:41 OSWatcher.sh
63449 foo       20   0  113m 1248   1036 S   2.0   0.0   3:47.17 tar
 3369 grid      20   0 1421m   26m   22m S   1.0   0.0 183:04.99 asm_gmon_+asm1
 3579 root      20   0 3102m 597m   24m S   1.0   0.2   6614:12 crsd.bin
10240 oracle    20   0  100g   71m   17m S   1.0   0.0 999:47.24 ora_dia0_camsdb
  178 root      20   0     0     0     0 S   0.7   0.0 448:19.31 kswapd0
 2060 grid      20   0 1290m   38m   18m S   0.7   0.0   2442:52 oraagent.bin
 2195 grid      20   0  618m   36m   15m S   0.7   0.0  12095:29 gipcd.bin
 6586 root      20   0 2306m 101m   10m S   0.7   0.0   4380:04 klnagent
42613 oracle    20   0  100g   30m   25m S   0.7   0.0   0:02.62 oracle_42613_ca
71259 oracle    20   0 15712 1992  1000 R   0.7   0.0   0:00.08 top
 1868 root      20   0 1841m   68m   19m S   0.3   0.0   3846:21 ohasd.bin
 1983 root      20   0 1547m 277m   19m S   0.3   0.1   3051:51 orarootagent.bi
 2078 grid      20   0 2557m 1.8g   16m S   0.3   0.7   4734:09 evmd.bin
 2171 grid      20   0  263m   16m   11m S   0.3   0.0 916:49.18 evmlogger.bin
 2465 grid      RT   0 2322m 205m   89m S   0.3   0.1   3093:18 ocssd.bin
 2994 root      20   0  805m   32m   15m S   0.3   0.0   2972:11 octssd.bin
 3282 grid      -2   0 1418m   18m   16m S   0.3   0.0   4700:13 asm_vktm_+asm1
 3300 grid      20   0 1425m   27m   19m S   0.3   0.0 400:19.42 asm_diag_+asm1
 3312 grid      20   0 1435m   41m   27m S   0.3   0.0   1201:35 asm_lmon_+asm1
 3320 grid      -2   0 1433m   35m   21m S   0.3   0.0   1138:17 asm_lms0_+asm1
 3326 grid      20   0 1419m   22m   19m S   0.3   0.0   1635:44 asm_lmhb_+asm1
 3709 root      20   0 1092m   40m   17m S   0.3   0.0   5572:41 orarootagent.bi
 4330 root      20   0 4030m 267m 4412 S   0.3   0.1   5022:57 oakd
 6323 root      20   0  107m 2484  1132 S   0.3   0.0   2312:43 kesl_launcher.s
10210 oracle    -2   0  100g   19m   17m S   0.3   0.0 521:19.14 ora_vktm_camsdb
10232 oracle    20   0  100g   30m   25m S   0.3   0.0  96:56.48 ora_dbrm_camsdb
10244 oracle    20   0  100g   47m   21m S   0.3   0.0  13:35.06 ora_dbw0_camsdb
10260 oracle    20   0  100g   46m   20m S   0.3   0.0  12:04.52 ora_dbw4_camsdb
10310 oracle    20   0  100g   23m   19m S   0.3   0.0 496:26.51 ora_mmnl_camsdb
    1 root      20   0 19404 1232   988 S   0.0   0.0  22:05.07 init
    2 root      20   0     0     0     0 S   0.0   0.0   0:02.19 kthreadd
```

Figure 5. Server Monitoring

Table 1. Application Server Configuration (Tibero)

Processor	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz (8 Core / 16 Thread)
Memory	32 GB
Storage	100 GB
Operating System	CentOS 8 Stream
Aplikasi	Arium CAMS NG

Table 2. Database Server Configuration (Tibero)

Processor	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz (8 Core / 16 Thread)
Memory	124 GB
Storage	500 GB + 1,500 GB
Operating System	CentOS 8 Stream
Database	Tibero 7
Others	Apache DS

Table 3. Application Server Configuration (Oracle)

Processor	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz (8 Core / 16 Thread)
Memory	32 GB
Storage	100 GB
Operating System	CentOS 8 Stream
Aplikasi	Arium CAMS NG

Table 4. Database Server Configuration (Oracle)

Processor	Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40GHz (8 Core / 16 Thread)
Memory	124 GB
Storage	2 TB
Operating System	CentOS 8 Stream
Database	Oracle 12c
Others	Apache DS

The resulting parameter is Transactions per Second (TPS).
TPS = Number of successful transactions / (test end time – test start time)
Example: If the transaction is successful = 15000 and the test takes place from 09:00 to 09:08 then TPS = 15000/(8 * 60) = 135
Steps taken in each test:

1. Prepare tools for monitoring and analyzing applications.
2. Run the application.
3. Monitoring applications both in terms of database, application and query.
4. Conduct analysis.

The table 5 shows that Tibero generally performs better than Oracle in terms of speed and resource usage. For the Upload Disburse task, Tibero completes it in 476 seconds with 42 TPS, while Oracle takes 677 seconds with 29.54 TPS. Tibero uses 60% memory and 70% CPU, compared to Oracle's 64% memory and 55% CPU. In the Approve Disbursements task, Tibero is faster at 105 seconds and 190 TPS, whereas Oracle takes 347 seconds and 57.64 TPS. Tibero's memory usage is 85% and CPU usage is 80%, both slightly lower than Oracle's 92% and 84%. For Upload Payment, Tibero takes 148 seconds with 135 TPS, using 33.3% memory and 10.5% CPU, while Oracle takes 75 seconds with 266.67 TPS, using 48% memory and 33.8% CPU. In Approve Payment, Tibero completes it in 489 seconds with 41 TPS, using 20% memory and 7% CPU, compared to Oracle's 533 seconds, 37.57 TPS, 24% memory, and 28% CPU. For Upload Early Termination, Tibero takes 527 seconds with 19 TPS, using 5% memory and 13% CPU, while Oracle takes 538 seconds with 18.58 TPS, using 19% memory and 18% CPU. In Approve Early Termination, Tibero is slower at 633 seconds with

16 TPS, using 30% memory and 18% CPU, compared to Oracle's 529 seconds, 18.88 TPS, 32% memory, and 21% CPU. For Input Write Off, Tibero completes it in 54 seconds with 185 TPS, using 11% memory and 8% CPU, while Oracle takes 48 seconds with 208 TPS, using 13% memory and 11% CPU. In Approve Write Off, Tibero takes 761 seconds with 13 TPS, using 10% memory and 8% CPU, while Oracle takes 772 seconds with 12.92 TPS, using 13% memory and 9% CPU. For the EOD End of Day Process, Tibero takes 4,375 seconds with 268.62 TPS, using 35% memory and 12% CPU, while Oracle takes 4,140 seconds with 283.94 TPS, using 38% memory and 14% CPU.

Overall, Tibero shows superiority in some tasks such as disburse approval and several other processes, while Oracle is superior in tasks such as uploading payments. Both databases are competitive in performance, and selecting the most appropriate database will depend on specific needs and the type of tasks dominant in the operational environment in which it is used. However, overall, if averaged, the Oracle TPS is superior to Tibero with a value of 103.75 compared to 102.75 for Tibero.

Tabel 5. Performance Test Results

Task	Tibero Total Time (s)	Oracle Total Time (s)	Tibero TPS	Oracle TPS	Tibero Memory Avg (%)	Oracle Memory Avg (%)	Tibero CPU Avg (%)	Oracle CPU Avg (%)
Upload Disburse	476	677	42.0	29.54	60	64	70	55
Approve Disburse	105	347	190.0	57.64	85	92	80	84
Upload Payment	148	75	135.0	266.67	33.3	48	10.5	33.8
Approve Payment	489	533	41.0	37.57	20	24	7	28
Upload Early Termination	527	538	19.0	18.58	5	19	13	18
Approve Early Termination	633	529	16.0	18.88	30	32	18	21
Input Write Off	54	48	185.0	208.0	11	13	8	11
Approve Write Off	761	772	13.0	12.92	10	13	8	9
End Of Day EOD	4,375	4,140	268.62	283.94	35	38	12	14

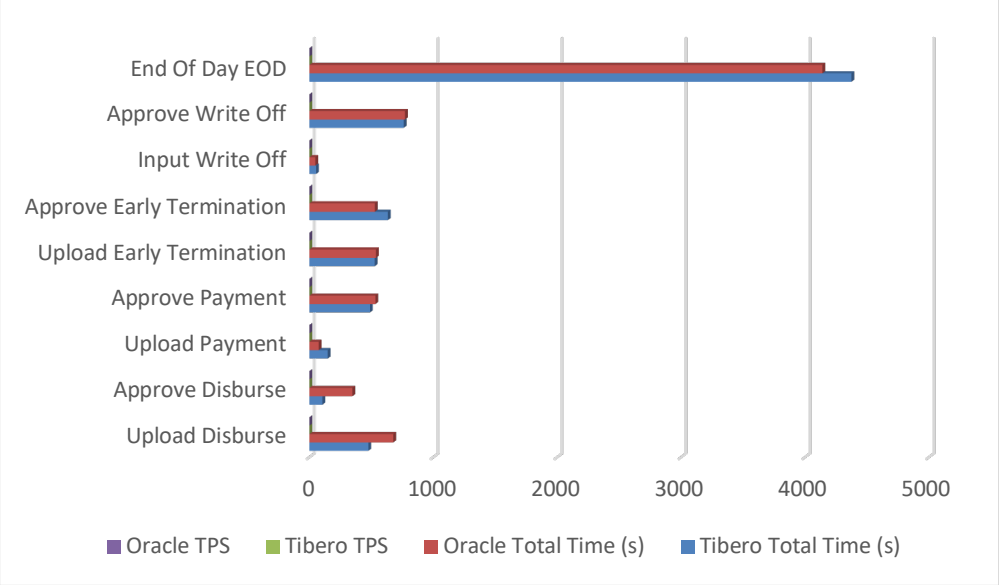


Figure 6. TPS Chart

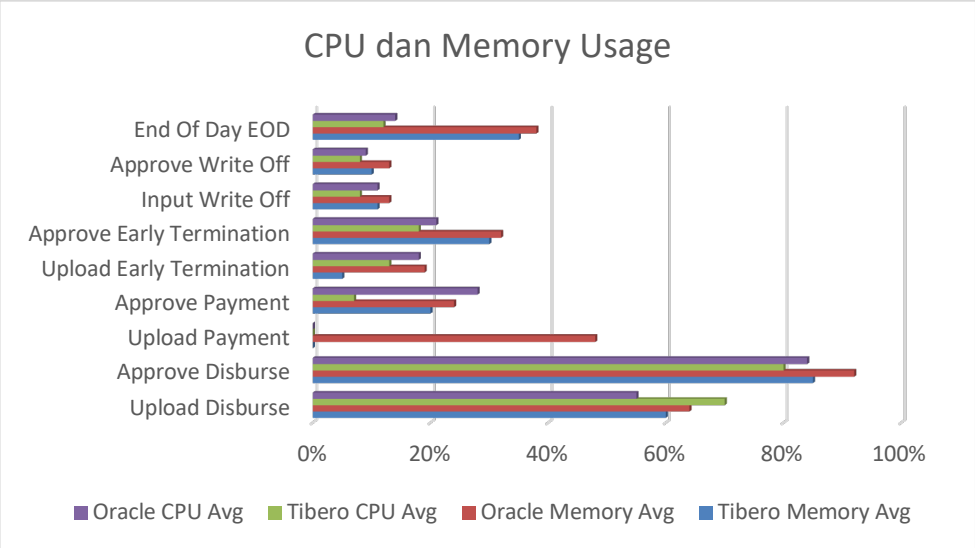


Figure 7. CPU and Memory Chart

Overall, this graph shows that there is a significant difference in CPU and memory resource usage between Oracle and Tibero systems depending on the processes performed. Resource usage needs to be considered when selecting a system to ensure optimal performance according to specific needs. From the average CPU and memory results of Oracle and Tibero, we find that the average memory usage for Oracle is 40.33% while Tibero's is 32.14%, and the average CPU usage for Oracle is 30.42% while Tibero's is 25.17%. The Oracle database maximizes CPU and memory usage compared to Tibero.

In Table 6 are the score results from respondents via Google Form. To calculate the SUS score results, a review is required using the SUS calculation formula, for this the SUS score calculation results are obtained. The results of the SUS score according to Table 7. In table 7, the average SUS score for all respondents is 75.25. The score shows the level of user satisfaction with Oracle, where the higher the score, the better the user assesses the experience of using Oracle.

Example of calculation for first responders:
(4-1) + (5-1) + (5-1) + (5-2) + (5-1) + (5-1) + (5-1) + (5-1) + (4-1) + (5-5) = 33

Score SUS = 33 * 2.5 = 82.5

Table 6. Recapitulation of Google Form Oracle Respondent Answer Points

No	1	2	3	4	5	6	7	8	9	10
1	4	1	5	2	5	1	5	1	4	5
2	4	1	4	3	5	2	3	2	4	5
3	4	4	3	5	4	4	4	3	4	4
4	5	5	5	3	5	1	5	1	5	3
5	3	2	4	1	4	2	4	2	4	2
6	2	4	3	5	3	3	3	3	4	5
7	4	5	3	4	4	3	3	3	4	4
8	5	1	5	4	4	1	5	2	5	3
9	2	3	3	4	4	3	3	3	4	4
10	5	1	5	1	4	1	5	1	5	1
11	3	2	4	3	4	2	4	2	4	3
12	4	2	3	2	3	3	3	3	4	2
13	4	3	3	2	4	3	3	3	4	3
14	4	2	5	2	5	1	5	1	5	3
15	3	2	4	3	4	2	4	2	4	3
16	4	2	3	2	3	3	3	3	4	2
17	3	1	5	2	5	1	5	1	5	2
18	5	3	4	4	4	3	4	3	4	3
19	3	2	3	2	3	2	3	2	3	3
20	4	3	4	3	4	2	3	2	4	2
21	4	1	5	1	4	1	5	1	5	1
22	4	1	4	1	5	1	5	1	5	1
23	4	1	4	1	4	1	5	1	5	1
24	4	1	5	1	5	1	4	1	5	1
25	5	1	4	1	4	1	5	1	5	1
26	3	1	4	1	5	1	5	1	4	1
27	3	1	5	1	5	1	4	1	5	1
28	5	1	4	1	5	1	5	1	5	1
29	3	1	5	1	5	1	5	1	5	1
30	2	2	3	2	3	2	3	2	3	3

Table 7. Oracle Score Calculation Results using the SUS formula

No	1	2	3	4	5	6	7	8	9	10	SCORE
1	3	4	4	3	4	4	4	4	3	0	82.5
2	3	4	3	2	4	3	2	3	3	0	67.5
3	3	1	2	0	3	1	3	2	3	1	47.5
4	4	0	4	2	4	4	4	4	4	2	80
5	2	3	3	4	3	3	3	3	3	3	75
6	1	1	2	0	2	2	2	2	3	0	37.5
7	3	0	2	1	3	2	2	2	3	1	47.5
8	4	4	4	1	3	4	4	3	4	2	82.5
9	1	2	2	1	3	2	2	2	3	1	47.5
10	4	4	4	4	3	4	4	4	4	4	97.5
11	2	3	3	2	3	3	3	3	3	2	67.5
12	3	3	2	3	2	2	2	2	3	3	62.5
13	3	2	2	3	3	2	2	2	3	2	60
14	3	3	4	3	4	4	4	4	4	2	87.5
15	2	3	3	2	3	3	3	3	3	2	67.5
16	3	3	2	3	2	2	2	2	3	3	62.5
17	2	4	4	3	4	4	4	4	4	3	90
18	4	2	3	1	3	2	3	2	3	2	62.5
19	2	3	2	3	2	3	2	3	2	2	60
20	3	2	3	2	3	3	2	3	3	3	67.5
21	3	4	4	4	3	4	4	4	4	4	95
22	3	4	3	4	4	4	4	4	4	4	95
23	3	4	3	4	3	4	4	4	4	4	92.5
24	3	4	4	4	4	4	3	4	4	4	95
25	4	4	3	4	3	4	4	4	4	4	95
26	2	4	3	4	4	4	4	4	3	4	90
27	2	4	4	4	4	4	3	4	4	4	92.5
28	4	4	3	4	4	4	4	4	4	4	97.5
29	2	4	4	4	4	4	4	4	4	4	95
30	1	3	2	3	2	3	2	3	2	2	57.5
Score Average											75.25

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 13, Issue 3, December 2024

Table 8. Recapitulation of Google Form Tibero Respondent Answer Points

No	1	2	3	4	5	6	7	8	9	10
1	5	1	5	2	5	1	5	1	4	3
2	4	3	4	3	4	2	3	2	4	2
3	4	3	5	5	5	3	5	3	5	2
4	5	3	5	3	5	1	5	1	5	3
5	5	2	4	2	4	2	4	2	4	2
6	4	3	4	4	3	3	4	3	4	2
7	4	3	3	2	4	3	3	3	4	3
8	4	2	5	2	5	1	5	1	5	3
9	3	2	4	3	4	2	4	2	4	3
10	4	2	3	2	3	3	3	3	4	2
11	5	1	5	2	5	1	5	1	5	2
12	5	3	4	4	4	3	4	3	4	3
13	4	2	3	2	3	2	3	2	3	3
14	4	3	4	3	4	2	3	2	4	2
15	3	2	3	2	4	3	4	3	3	3
16	4	3	5	2	5	2	5	2	5	3
17	5	2	4	3	4	3	4	3	4	3
18	5	1	5	2	5	1	5	1	5	2
19	4	3	3	4	4	3	3	3	3	4
20	4	2	4	2	4	2	4	2	4	3
21	3	2	3	3	3	2	4	3	3	2
22	4	3	5	2	5	2	5	2	5	3
23	3	2	4	4	4	3	3	3	4	2
24	5	1	5	2	5	1	5	1	5	2
25	4	3	3	3	3	3	3	3	3	2
26	5	2	4	2	4	2	4	2	4	3
27	5	3	4	2	4	3	4	3	4	2
28	4	2	5	2	5	2	5	2	5	3
29	4	2	3	2	3	2	3	2	3	3
30	4	3	4	3	4	2	3	3	4	2

In table 8 are the score results from respondents via Google Form. To calculate the SUS score results, a review is required using the SUS calculation formula, for this the SUS score calculation results are obtained. The results of the SUS score according to table 9.

In table 9, the average SUS score for all respondents is 71.50. This score shows the

level of user satisfaction with Tibero, with the same interpretation scale as in the previous table. The higher the score, the better the user assesses the experience of using the Tibero.

Example of calculation for first responders:
(5-1) + (5-1) + (5-1) + (5-2) + (5-1) + (5-1) + (5-1) + (5-1) + (4-1) + (5-3) = 36
Score SUS = 36 * 2.5 = 90%

Table 9. Tibero Score Calculation Results using the SUS formula

No	1	2	3	4	5	6	7	8	9	10	SCORE
1	4	4	4	3	4	4	4	4	3	2	90
2	3	2	3	2	3	3	2	3	3	3	67.5
3	3	2	4	0	4	2	4	2	4	3	70
4	4	2	4	2	4	4	4	4	4	2	85
5	4	3	3	3	3	3	3	3	3	3	77.5
6	3	2	3	1	2	2	3	2	3	3	60
7	3	2	2	3	3	2	2	2	3	2	60
8	3	3	4	3	4	4	4	4	4	2	87.5
9	2	3	3	2	3	3	3	3	3	2	67.5
10	3	3	2	3	2	2	2	2	3	3	62.5
11	4	4	4	3	4	4	4	4	4	3	95
12	4	2	3	1	3	2	3	2	3	2	62.5
13	3	3	2	3	2	3	2	3	2	2	62,5
14	3	2	3	2	3	3	2	3	3	3	67.5
15	2	3	2	3	3	2	3	2	2	2	60
16	3	2	4	3	4	3	4	3	4	2	80
17	4	3	3	2	3	2	3	2	3	2	67.5
18	4	4	4	3	4	4	4	4	4	3	95
19	3	2	2	1	3	2	2	2	2	1	50
20	3	3	3	3	3	3	3	3	3	2	72.5
21	2	3	2	2	2	3	3	2	2	3	60
22	3	2	4	3	4	3	4	3	4	2	80
23	2	3	3	1	3	2	2	2	3	3	60
24	4	4	4	3	4	4	4	4	4	3	95
25	3	2	2	2	2	2	2	2	2	3	55
26	4	3	3	3	3	3	3	3	3	2	75
27	4	2	3	3	3	2	3	2	3	3	70
28	3	3	4	3	4	3	4	3	4	2	82.5
29	3	3	2	3	2	3	2	3	2	2	62.5
30	3	2	3	2	3	3	2	2	3	3	65
Score Average											71.50

ISSN 2089-8673 (Print) | ISSN 2548-4265 (Online)
Volume 13, Issue 3, December 2024

From the results of calculating the formula in the equation, the SUS score is obtained which shows the level of user acceptance. The SUS score from the CAMS (Consumer Asset Management System) application system using the Oracle database is 75.25 while using the Tibero database is 71.50. Based on the image, the SUS score assessment from the CAMS (Consumer Asset Management System) application system using the Oracle database and using the Tibero database has good value because it is in the range 68 – 80.3.

Table 10. Mapping Score SUS

>80	A	Excellent
68-80.3	B	Good
68	C	Okay
51-68	D	Poor
<51	F	Awful

So it can be said that the CAMS (Consumer Asset Management System) application using the Oracle database has a higher value than the one using the Tibero database but is still in the same range, namely "good" value, thus Oracle is still the best enterprise database while Tibero can be used as one. Alternatives in application development with enterprise databases.

To ensure that the difference in SUS scores between Oracle and Tibero is not just due to random chance or sample variability, an independent two-sample t-test was conducted. This t-test helps determine whether the difference in average SUS scores between Oracle and Tibero is statistically significant.

Null Hypothesis (H0): There is no significant difference between the SUS scores of Oracle and Tibero.
Alternative Hypothesis (H1): There is a significant difference between the SUS scores of Oracle and Tibero.

Data from the SUS scores of Oracle and Tibero respondents were used to calculate the t-value and p-value. For example:

1. Average SUS Score for Oracle: 75.25
2. Average SUS Score for Tibero: 71.50
3. Standard Deviation (Oracle): 13.5 (example from respondent data)
4. Standard Deviation (Tibero): 12.0 (example from respondent data)
5. Number of Respondents (Oracle): 30
6. Number of Respondents (Tibero): 30

The t-test calculation is performed using the formula:

$$t = \frac{(\bar{X}1 - \bar{X}2)}{\sqrt{\frac{s1^2}{n1} + \frac{s2^2}{n2}}} \quad (3)$$

where:
- $\bar{X}1$ and $\bar{X}2$ are the means of the two samples (Oracle and Tibero).
- $s1^2$ and $s2^2$ are the variances of the two samples.
- n1 and n2 are the number of respondents for each sample.

Using the example numbers above, the t-test result is calculated as follows:

$$t = \frac{(75.25 - 71.50)}{\sqrt{\frac{13.5^2}{30} + \frac{12.5^2}{30}}} \approx 1.07 \quad (4)$$

After calculating, we compare this t-value with the t-distribution table for degrees of freedom (df = n1 + n2 - 2 = 58) at a significance level of 0.05. If the p-value < 0.05, the null hypothesis (H0) is rejected, indicating that the difference is statistically significant.

t-Test Results:
In this example, suppose the p-value obtained is 0.29 (> 0.05), which means we fail to reject the null hypothesis. This indicates that although there is a difference in SUS scores between Oracle and Tibero, this difference is not statistically significant.

With these results, we can conclude that there is no statistically significant difference between the user experiences using Oracle and Tibero based on SUS scores

CONCLUSION

From the results of the research, several key conclusions have been drawn:

1. Competitive Performance Between Tibero and Oracle: Both Tibero and Oracle show competitive performance across various tasks. Tibero performs better in some tasks, such as disbursement approval and certain processes, where it demonstrates faster completion times and lower CPU and memory usage. Conversely, Oracle excels in tasks like payment uploads, with a slightly higher average Transactions Per Second (TPS) value of 103.75 compared to 102.75 for Tibero.
2. Resource Usage Considerations: Oracle tends to maximize CPU and memory usage more than Tibero, suggesting that Oracle may be more suitable for environments requiring high performance with optimized resource utilization. On the other hand, Tibero is more efficient in specific cases where lower memory

and CPU usage are prioritized. This makes Tibero a viable alternative depending on the specific needs of the application and the operational environment.

3. Suitability Based on Specific Needs: The choice between Tibero and Oracle should be guided by the specific needs and context of the application. While Oracle generally outperforms Tibero in terms of overall performance metrics, Tibero's efficiency in resource usage makes it a strong alternative for applications where such characteristics are prioritized.
4. Usability Evaluation Using the SUS Method: The SUS (System Usability Scale) analysis of the CAMS (Consumer Asset Management System) application reveals that Oracle has a higher usability score of 75.25 compared to 71.50 for Tibero. Although both scores fall within the "Good" range on the SUS scale, Oracle's higher score indicates a better user experience.
5. Importance of Robust Statistical Analysis: While descriptive analysis provides a good overview of the comparative performance and usability of Oracle and Tibero, using more in-depth statistical methods, such as the t-test, is crucial. These methods provide a more comprehensive understanding of whether the observed differences are statistically significant, thus ensuring that conclusions drawn are not biased and are supported by robust statistical evidence.
6. Oracle as the Preferred Choice for Enterprise Databases: Overall, the findings suggest that Oracle remains the best choice for enterprise databases due to its superior performance and usability scores, making it highly suitable for environments that require high performance with optimized resource use. Although Tibero should not be overlooked as it can serve as a viable alternative, especially in scenarios where Tibero's resource efficiency can be effectively leveraged, the emphasis on overall performance makes Oracle the more favorable database choice.

These conclusions highlight that with a focus on performance, Oracle becomes the better choice for enterprise databases. The decision should be based on the specific requirements of the application, the operational environment, and the nature of the tasks being performed, ensuring that the chosen database aligns well with organizational goals and performance expectations.

ACKNOWLEDGMENT

I wish to express my heartfelt thanks to my advisor and all those who have contributed to the completion of this journal publication, which forms part of my studies in the Master's Program at the Department of Information Systems, Institut Teknologi dan Bisnis STIKOM Bali.

REFERENCES

[1] Ningsih, M., & Muzakir, A. (2021). Mengevaluasi User Interface Untuk Meningkatkan User Experience (Ux) Menggunakan Metode System Usability Scale (Sus). *Bina Darma Conference on Computer Science*, 365–374.

[2] Laksana, F. F., & Suyoto, S. (2019). Pengukuran Kualitas Ux Website Menggunakan Sus. *Computer Engineering, Science and Systems Journal, 4*(2), 138. doi: 10.24114/cess.v4i2.12928.

[3] Herdiana, G. A., Swarmardika, I. B. A., & Hartati, R. S. (2022). Pengukuran User Experience (Ux) Desain Aplikasi Trouble Ticket Menggunakan Metode Supergolden Ratio. *SINTECH (Science Information Technology Journal), 5*(1), 42–48. doi: 10.31598/sintechjournal.v5i1.1093.

[4] Nopita, M., Purnamasari, S. D., & Yudiastuti, H. (2022). Evaluasi Usability Website SMA PGRI 2 Palembang Menggunakan System Usability Scale (SUS). *Jurnal Mantik, 6*(3), 3299–3307.

[5] Praba, A. D., & Safitri, M. (2020). Studi Perbandingan Performasi Antara MySQL dan PostgreSQL.

[6] Ilić, M., Kopanja, L., Zlatković, D., Trajković, M., & Ćurguz, D. (2021). Microsoft SQL Server and Oracle: Comparative Performance Analysis.

[7] Xu, Y., Park, Y., Park, J. D., & Sun, B. (2023). Predicting Nurse Turnover for Highly Imbalanced Data Using the Synthetic Minority Over-Sampling Technique and Machine Learning Algorithms.

[8] Fuertes, A. B., Pérez, M., & Meza, J. (2023). Nmorph Framework: An Innovative Approach to Transpiler-Based Multi-Language Software Development.

[9] Costa, C., Faria, J. M., Guimarães, D., Matos, D., & Moreira, A. H. J. (2023). A Wearable Monitoring Device for COVID-19 Biometric Symptoms Detection.

[10] Al-Khiami, M. I., Jaeger, M., & Soleimani, S. M. (2023). Enhancing Concrete Structures Education: Impact of Virtual Reality on Motivation, Performance, and Usability for Undergraduate Engineering Students.

[11] Naito, H., Matsumoto, M., & Nakano, M. (2023). Performance Analysis of Comparison Operations on Homomorphically Encrypted Sensitive Data.

[12] TmaxSoft, "Tibero SQL Reference Guide," 2024. [Online]. Available:

https://technet.tmaxsoft.com/upload/download/online/tibero/pver-20170217-000001/sql-reference/index.html. [Accessed: Jul. 10, 2024].
[13] Naito, H., Matsumoto, M., & Nakano, M. (2023). Performance Analysis of Comparison Operations on Homomorphically Encrypted Sensitive Data.
[14] Tongkaw, S., & Tongkaw, A. (2019). A Comparison of Database Performance of MariaDB and MySQL with OLTP Workload. Songkhla Rajabhat University, Songkhla, Thailand
[15] Roy, A.M.; Bhaduri, J. (2021). A Deep Learning Enabled Multi-Class Plant Disease Detection Model Based on Computer Vision. AI, 2, 413–428.
[16] Roy, A.M. (2022). An efficient multi-scale CNN model with intrinsic feature integration for motor imagery EEG subject classification in brain-machine interfaces. Biomed. Signal Process. Control., 74, 103496.
[17] Singh, A., Raj, K., Kumar, T., Verma, S., Roy, A.M. (2023). Deep Learning-Based Cost-Effective and Responsive Robot for Autism Treatment. Drones, 7, 81.
[18] Alias, N., Suhari, N.N., Saipol, H.F., Dahawi, A.A., Saidi, M.M., Hamlan, H.A., Teh, C.R. (2016). Parallel computing of numerical schemes and big data analytic for solving real life applications. Jurnal Teknologi, 78.
[19] Chang, M.-L.E., Chua, H.N. (2018). SQL and NoSQL Database Comparison. In Proceedings of the Future of Information and Communication Conference, Vienna, Austria, 4–6 December 2018; pp. 294–310.
[20] Ansari, H. (2022). Performance Comparison of Two Database Management Systems MySQL vs. MongoDB; Umeå University: Umeå, Sweden.
[21] Journal of Educational Technology, 15(2), 123-130
[22] Hendra, & Andriyani, W. (2020). Studi Komparasi Menyimpan dan Menampilkan Data Histori antara Database Terstruktur MariaDB dan Database tidak Terstruktur InfluxDB. Journal of Data Management, 33(2), 150-160
[23] Sandhiyasa, I. M. S., Yanti, C. P., & Hendrawati, T. (2021). Implementation and Evaluation of Accounting Information Systems in Manufacturing Company Using System Usability Scale. Indonesian Journal of Accounting, 19(3), 45-55
[24] Permatasari, D. I., Ardani, M., Ma'ulfa, A. Y., Ilhami, N., Pratama, S. G., Astuti, S. R. D., & Naufalita, N. W. (2020). Pengujian Aplikasi Menggunakan Metode Load Testing dengan Apache Jmeter pada Sistem Informasi Pertanian. Journal of Agricultural Informatics, 14(1), 67-76
[25] Juniawan, F. P., Laurentinus, & Sylfania, D. Y. (2021). Evaluasi Usability Sistem Pelaporan Publikasi Penelitian Dosen Berbasis Android. Journal of Mobile Computing, 29(4), 89-98.
[26] D. Madhani, E. Darwiyanto, dan A. Gandhi, (2023) "Performance Testing Menggunakan Metode Load Testing dan Stress Testing pada Sistem Core Banking PT. XYZ," e-Proceeding of Engineering, vol. 10, no. 6, pp. 5431, Desember 2023.
[27] Permatasari D I, Ardani M, & Ma'ulfa A Y. (2020). Pengujian Aplikasi Menggunakan Metode Load Testing dengan Apache Jmeter pada Sistem Informasi Pertanian. JUSTIN (Jurnal Sistem dan Teknologi Informasi), 8(1), 135-139.
[28] Diastama I G N P D D, Sukarsa I M, Wirdiani N K A. (2021). Pengembangan Test Script untuk Load Testing Web dengan metode Software Testing Life Cycle. JITTER. 2(1).
[29] Chou C Y, Fang Y B, Wang S T, & Kuo F A. (2020). Smoke and Stress Tests for Travel Service Applications via LoadRunner. LNISA, 11894, 366-373.
[30] Fansha D A, Setyawan M Y H, & Fauzan M N.(2021). Load Test pada Microservice yang menerapkan CQRS dan Event Sourcing. Jurnal Buana Informatika, 12(2), 126-134.