Applied Systems and Technologies Journal (ASTJ)

# Design and Development of a Web-Based Capital Expenditure Management Information System

**Noper Ardi [1]*, Bagas Hilmi Arib [2]**
*correspondence: noperardi@polibatam.ac.id
[1,2]Informatics Engineering
Politeknik Negeri Batam
Batam, Indonesia

**Abstract-** Capital Expenditure (Capex) management plays an essential role in supporting corporate investment planning, asset acquisition, and financial decision-making processes. However, Capex data management activities in organizations often encounter challenges due to the use of manual processes involving Enterprise Resource Planning (ERP) systems and spreadsheet applications, which may reduce data accessibility and increase the risk of input errors. This study aims to design and develop a web-based Capital Expenditure Management Information System to improve the effectiveness and efficiency of Capex monitoring and reporting processes. The system was developed using the Laravel framework with an Agile software development approach. The proposed system provides several functionalities, including Capex master data management, cumulative balance monitoring, document management, user administration, and automated reporting features. System evaluation was conducted using the Black Box Testing method to verify the functionality of each feature based on predefined requirements. The testing results indicate that all implemented features operate successfully and meet user requirements. The developed system improves the efficiency of Capex data processing, minimizes manual errors, and facilitates users in monitoring investment activities through a centralized web-based platform. This system can support organizations in implementing a more structured and effective Capital Expenditure management process

**Keywords**: Capital Expenditure, Information System, Web-Based Application, Laravel Framework, Black Box Testing

**Article info**: *submitted October 11, 2025, revised November 2, 2025, accepted December 15, 2025*



## 1. Introduction

In the modern industrial era, the management of long-term investments is one of the key factors in maintaining business sustainability and supporting company growth [1]. One form of such investment is Capital Expenditure (Capex), which refers to the allocation of funds planned for the development and construction of major and supporting infrastructure, as well as project assets. In general, Capex includes all expenditures incurred to acquire fixed assets that provide long-term benefits beyond the current operating period [2–3].

Examples of Capital Expenditure include the purchase of new machinery, expansion of production facilities, technology improvements, and the renewal of company properties and equipment. Properly managed capital expenditures can increase production capacity, enhance business competitiveness, and support long-term organizational growth [4].

In the oleochemical industry, Capital Expenditure (Capex) plays an important role in the company's strategy for developing production capacity, technology, and infrastructure. Currently, these company relies on the SAP Enterprise Resource Planning (ERP) system for managing Capex data. However, observations indicate that the process of retrieving and processing data from SAP is still performed manually, such as exporting data into Microsoft Excel for further processing separately. This process results in an inefficient workflow and increases the risk of data recording errors.

Furthermore, accessing Capex data in SAP requires the use of several different transaction codes (t-codes), which slows down the process of retrieving information. This condition makes it difficult for staff to obtain Capex information quickly, comprehensively, and in real time. The reporting process, which is still performed manually, also increases workload and creates a higher possibility of data inconsistencies between documents.

These findings indicate the need for a more integrated, efficient, and user-friendly system for Capex management. Such a system should be able to simplify the processes of data recording, searching, and reporting without requiring direct dependence on manual processing through SAP.

This research aims to develop a web-based Capex information system that can simplify the processes of recording, retrieving, and reporting Capital Expenditure data digitally. This research is aligned with previous studies conducted by Martin Siregar and Sholihin, I Gusti Agung Istri Siva Larasathi et al., and Andrew Robot Feldy Najoan et al., which demonstrate that the development of web-based information systems can improve efficiency in data management and administrative processes [5–7].

The system is developed using the Agile Development methodology, which enables the software development process to be divided into smaller components that can be developed incrementally and iteratively according to user requirements [11]. In addition, Laravel is selected as the development framework because it supports web-based application development using the Model-View-Controller (MVC) architecture, while providing ease of implementation and an efficient development structure [12].

Through a web-based approach, the proposed system is expected to improve Capex management efficiency, accelerate operational processes, and minimize errors in data processing.

## 2. Methods

### A. *System Development Method*

The development of the Capex Information System application adopts the Agile Development approach, as illustrated in Figure 1. Agile is an advanced approach derived from the System Development Life Cycle (SDLC) methodology, designed to support application development processes within a relatively shorter period while improving the success rate compared to traditional design methodologies [7]. This method is also recognized as a software development approach that enables development teams to adapt more quickly to changing requirements [13-15]. The stages involved in the Agile methodology are described as follows:


Fig 1. Agile Methodology

*1) Planning Phase*

The planning phase is conducted to understand user requirements and determine the features to be developed. The researcher collects information through direct observation of the workflow within the Operation Control Department and conducts informal interviews with one of the involved staff members.

Based on the collected data, it was identified that employees experienced difficulties in accessing, viewing, and correlating Capex-related data. Therefore, the system requirements were defined, including several main features such as Capex master data management, cumulative balance management, document upload management, and reporting features.

*2) System Design Phase*

The system design phase involves creating an application model based on the previously collected requirements. The researcher designs Unified Modeling Language (UML) diagrams consisting of a Use Case Diagram to define user roles and activities, an Activity Diagram to describe the workflow and processes within the system, a Sequence Diagram to illustrate interactions between system objects based on time sequence, and an Entity Relationship (ER) Diagram to design the database structure.

In addition, interface mockups are created to design the user interface layout and improve the usability of the developed system.

*3) Development Phase*

During the development phase, the researcher begins implementing the system by writing program code based on the design specifications defined in the previous stage.

The implementation process includes developing the designed features, integrating the database, and developing the user interface. The development process is carried out iteratively to ensure that each component functions according to the specified requirements.

*4) Testing Phase*

The testing phase is conducted after the implementation process has been completed to ensure that the system operates properly and meets the expected requirements.

System testing is performed using the Black Box Testing method, which focuses on evaluating system functionalities based on input and output behavior without considering the internal structure of the program[16].

*5) Review Phase*

The review phase is conducted after the system has been used by users. At this stage, the system is evaluated comprehensively based on testing results and user feedback.

This evaluation aims to determine whether the developed system meets user needs and expectations. The findings and feedback obtained are used to identify system strengths, weaknesses, and aspects that require improvement.

If inconsistencies or deficiencies are identified, the system will return to the design or development phase for further refinement. This process reflects the iterative principle of the Agile methodology, allowing the system to adapt and evolve according to actual conditions in the operational environment[17].

*6) Implementation Phase*

The implementation phase involves deploying the system to a production server so that it can be accessed and utilized by users.

Once the system has been installed and operates online, users can directly access and utilize the application according to the requirements of Capex data management processes within the company.

### B. *Kanban Method*

In the development of this system, the researcher applies the Kanban method to manage the workflow visually and efficiently[18]. The Kanban approach divides the development activities into several stages as follows:

*1) Backlog Phase*

In the Backlog phase, the researcher collects and documents all system requirements, development ideas, and findings obtained from observations and interviews into a backlog list.

The tasks listed in this phase have not yet been prioritized for implementation but serve as the initial reference for the development task selection process.

2) To Do Phase

After evaluating the tasks available in the backlog, the researcher moves the prioritized tasks into the To Do column. These tasks are planned to be implemented in the near future based on system requirements and the development schedule.

3) Doing Phase

In this phase, the researcher begins working on the selected tasks. Tasks that are currently being developed are moved to the Doing column so that their progress can be monitored visually. The researcher also limits the number of tasks being processed simultaneously to maintain focus, productivity, and the quality of the development results.

4) Done Phase

After completing the development process and conducting testing on specific tasks, the researcher moves the completed tasks into the Done column. Tasks in this phase have been validated and declared complete, and they are ready to be implemented into the system.

### C. *System Overview*

The general overview of the proposed Web-Based Capital Expenditure (Capex) Information System describes an application developed to support the Operation Control (OC) department in monitoring data progress and managing Capex information. The overall system workflow is illustrated in Figure 2.

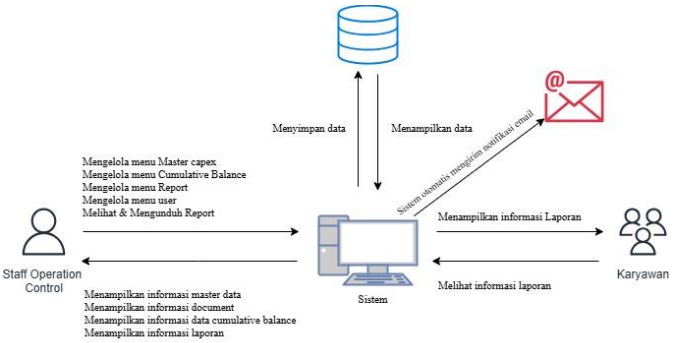


Fig 2. System Overview

The system workflow begins with the Staff OC performing the login process. After successful authentication, the staff is directed to the Master Capex page, where they can manage Capital Expenditure master data. Furthermore, on the Cumulative Balance page, Staff OC can manage cumulative balance data related to Capex projects. On the Upload Document page, Staff OC can manage supporting documents, including FAGLB and ZLIS1 files[17].

The system also provides a CIP Report page, where Staff OC can view detailed report information. The displayed data is generated by combining Master Capex data and uploaded Excel files. The report can also be exported and printed in PDF format.

In addition, the Category Report page allows Staff OC to view Capex reports based on specific categories. Meanwhile, the Summary Report page provides a detailed summary of Capex information for monitoring and analysis purposes. For Employees, the workflow begins with the login process, after which users are redirected to the CIP Report page. On this page, employees can view their respective Capex-related data and download reports in PDF format[19].

Furthermore, on the Category Report page, employees can view their data based on predefined categories. On the Summary Report page, employees can access detailed summaries of their respective Capex information.

### D. *Functional Requirements*

Functional requirements define the main features that must be provided by the system to support the Capital Expenditure (Capex) data management process. The required functionalities include master data management, cumulative balance management, document uploading, and report presentation.

All functional requirements are identified and formulated based on the results of observations and interviews conducted with system users.

The details of the functional requirements are presented in Table 1.

Table 1. Fungsional requirements

| Code | Fungsional requirements |
|---|---|
| KF-01 | Staff Operation and employee can login. |
| KF-02 | Staff Operation Control can manage Master Capex |
| KF-03 | Staff Operation Control can manage Budget |
| KF-04 | Staff Operation Control can manage Cumulative Balance |
| KF-05 | Staff Operation Control can manage Document faglb & zlis1 |
| KF-06 | Staff Operation Control can manage User |
| KF-07 | Staff Operation Control can manage Requester |
| KF-08 | Employee can view Report CIP |
| KF-9 | Employee can view Report Category |
| KF-10 | Employee can view Report Summary |

### E. *Non-Functional Requirements*

Non-functional requirements refer to aspects that are not directly related to the main functions of the system but significantly affect the usability, security, and reliability of the system during operation. In the development of this Capex Information System, non-functional requirements are categorized into three main aspects: security, ergonomics, and language.

The security aspect focuses on protecting data and managing user access to ensure that the system can only be accessed by authorized users. The ergonomics aspect is related to system usability, including a simple user interface design and clear navigation to support user interaction with the system.

Meanwhile, the language aspect focuses on the use of appropriate and understandable language that enables users to easily comprehend each feature provided by the system.

The details of the non-functional requirements are presented in Table 2.

Table 2. non-functional requirements

| Code | Category | Non-functional requirements |
|---|---|---|
| NF-01 | Security | The system is designed for internal company use only, and access is restricted to users who have registered accounts. |
| NF-02 | Ergonomy | The system interface is designed to be as simple and intuitive as possible to |

| | | |
|---|---|---|
| | | ensure that every user can easily understand and operate the system. |
| NF-03 | Language | Using english |

*F. Usecase diagram*

The system design in this research utilizes the Unified Modeling Language (UML) as a modeling language, which consists of several types of diagrams, including the Use Case Diagram [8].

A Use Case Diagram is used to describe the interaction between actors and the system in performing its main functions, as illustrated in Figure 3.

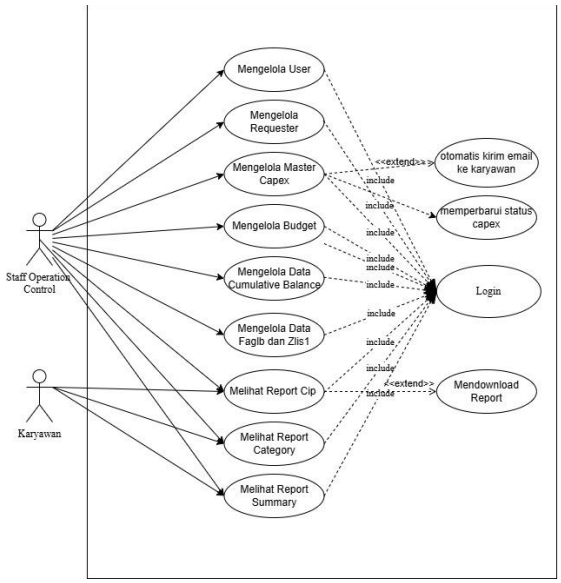


Fig 3. Usecase diagram

This diagram visualizes the roles of users within the system and the main features that can be accessed, including Capex data management, cumulative balance monitoring, document uploading, and report generation.

Through the use of a Use Case Diagram, developers can gain a comprehensive understanding of system requirements from the user perspective and use it as a reference in designing the application's functionalities.

*G. Activity Diagram*

An Activity Diagram is used to describe the flow of activities or business processes within the system, from the beginning to the completion of a process. This diagram illustrates the sequence of steps performed by users and the system in a structured manner, as well as the decisions or conditions that influence the execution of the process. By utilizing an Activity Diagram, developers can understand how processes within the system operate logically from one stage to another. The Activity Diagram of the developed Capex Information System is presented in Figure 4.

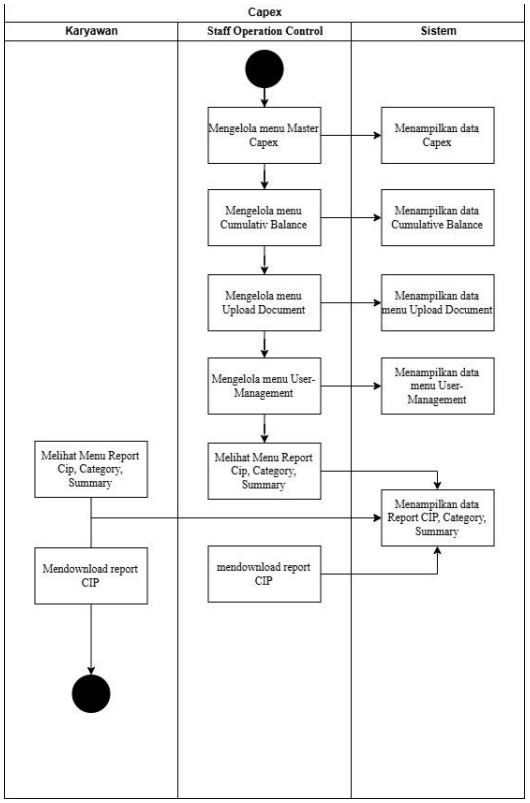


Fig 4. Activity Diagram

diagram illustrates the sequence of steps performed by users and the system in a structured manner, as well as the decisions or conditions that influence the execution

*H.* Sequence Diagram

A Sequence Diagram is a diagram that illustrates the interactions between objects within a system based on a specific time sequence. It shows how messages are exchanged between objects to execute a particular process.

This diagram is used to visualize the communication flow that occurs between users, the system, and other components in response to an action or request.

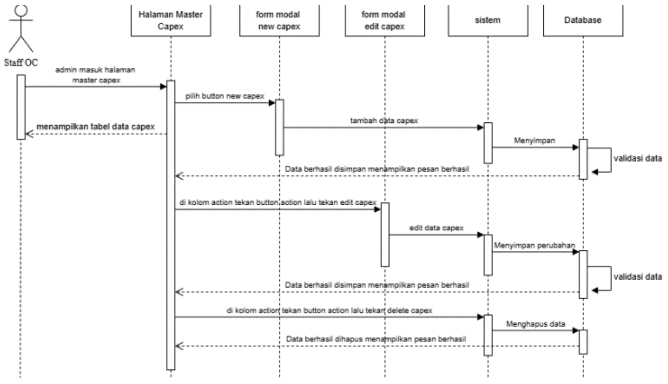


Fig 5. Sequence diagram

By utilizing a Sequence Diagram, developers can understand the execution order of processes in greater detail, thereby facilitating the system implementation process. An example of a Sequence Diagram is presented in Figure 5, which illustrates an interaction scenario within the developed system.

## 3. Result and Discussion

The implementation stage is the process of translating the system design into a functional application. The developed Capital Expenditure Information System is a web-based application built using the Laravel framework.

The system provides several features that support Capex management activities, including Capex data management, cumulative balance monitoring, document management, user management, and report generation.

### A. *Login Page*

The login page is used by users to access the system. Users must enter their registered email address and password. The authentication process ensures that only authorized users can access system features according to their access rights.

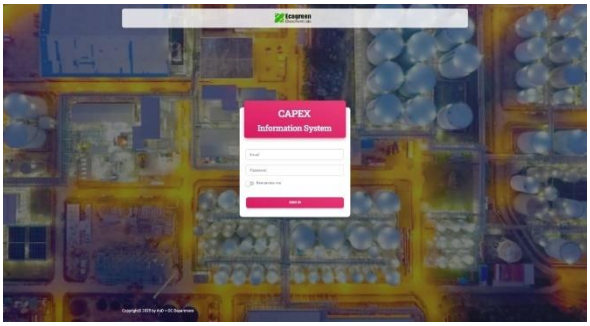


Fig 6. Login Page

### B. *Master Capex Page*

The Master Capex page functions as the main page for managing Capital Expenditure data. On this page, Staff Operation Control (OC) can add, update, delete, and filter Capex data based on specific criteria.

The available filtering features allow users to search data based on categories, years, and specific periods, making the data monitoring process easier.

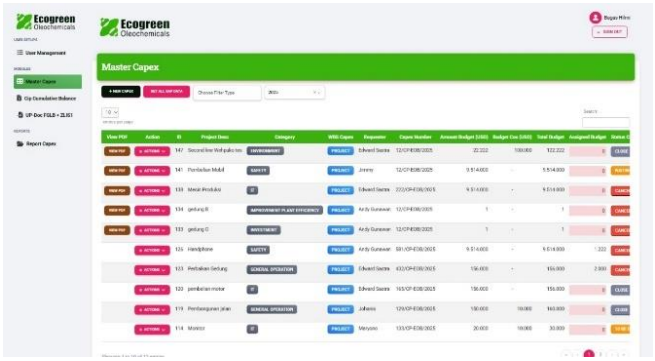


Fig 7. Master CapEx

### C. *Cumulative Balance Page*

The Cumulative Balance page is used to manage and monitor the development of Capex project balances. Through this page, Staff OC can add, modify, delete, and release cumulative balance data. The information displayed helps users track the financial status of each Capex project.

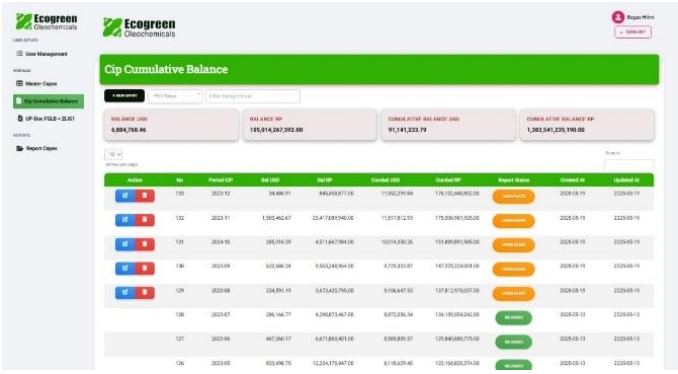


Fig 8. Cumulative Balance page

### D. *Upload Document FAGLB & ZLIS1 Page*

This page provides functionality for uploading supporting documents related to Capex activities, including FAGLB and ZLIS1 documents. Users with Staff OC access rights can upload, edit, delete, and manage documents through this feature. The uploaded documents are stored in the system database and can be accessed when needed.

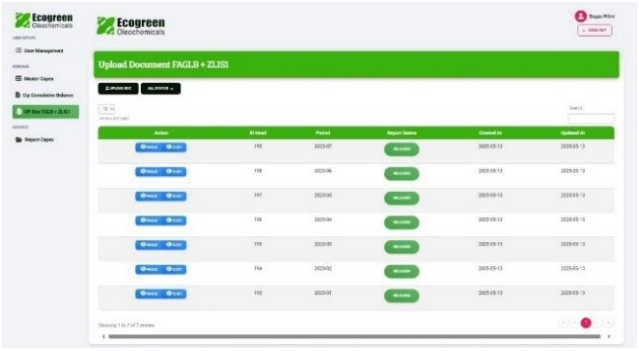


Fig 9. Upload Document

### E. *User Management Page*

The User Management page is used to manage user accounts and access permissions within the system. Administrators can add, update, and delete user data to ensure that each user has appropriate access rights according to their role.

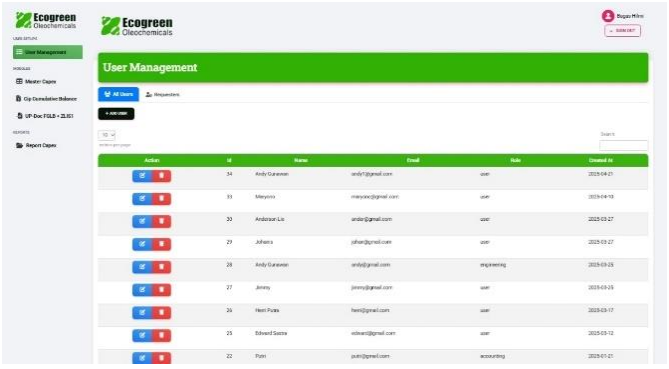


Fig 10. User management

### F. *Report CIP Page*

The Report CIP page provides information related to Capex projects. Users can apply filters based on project status and other parameters to display specific reports. The system also provides a

download feature that allows users to export reports for further analysis.

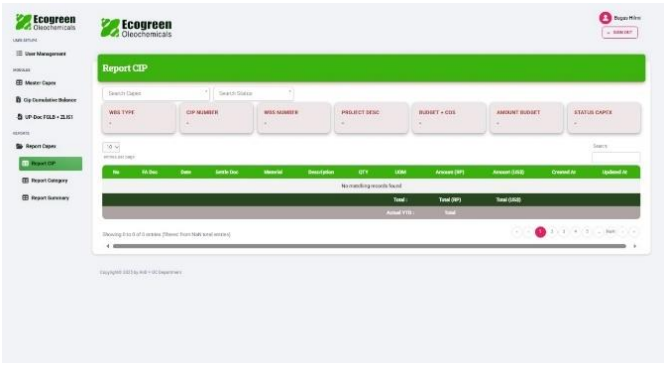


Fig 11. Report CIP Page

### G. *Report Category Page*

The Report Category page displays Capex data grouped based on specific categories and periods. This feature helps users analyze Capex distribution and monitor investment activities according to different categories.

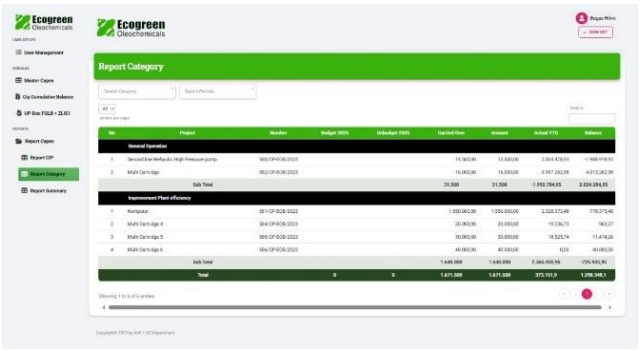


Fig 12. Report Category Page

### H. *Report Summary Page*

The Report Summary page allows Staff Operation Control and employees to view Capex data summaries. Users can apply filters based on categories, years, and time ranges to obtain specific information required for monitoring and reporting purposes.

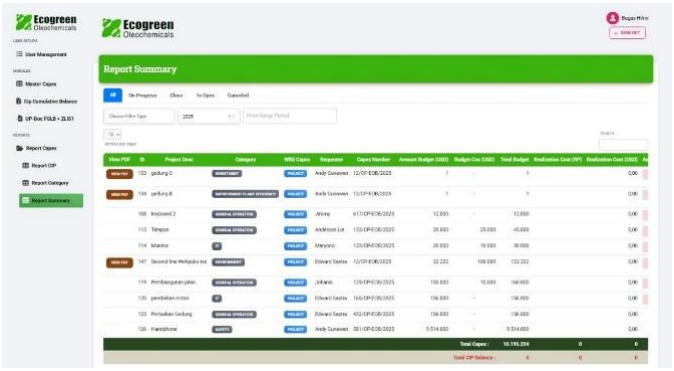


Fig 13. Report Summary Page

### I. *System Testing Results*

System testing was conducted to evaluate the functionality and performance of the developed Web-Based Capital Expenditure (Capex) Information System. The testing process was performed using the Black Box Testing method, which focuses on verifying whether each system function operates according to the specified requirements by examining the relationship between input and expected output without analyzing the internal program structure.

The testing process involved several functional scenarios covering all major features within the system, including user authentication, Capex data management, cumulative balance management, document management, user management, and reporting features. Testing was conducted based on different user access levels, namely Staff Operation Control (OC) and Employee, to ensure that the authorization mechanism and system functionalities worked properly according to each user's role.

The authentication testing results showed that the login mechanism successfully validated user credentials. When users entered incorrect login information, the system provided an appropriate error notification, preventing unauthorized access. Meanwhile, users with valid credentials were successfully redirected to their respective dashboards based on assigned access rights. Staff OC users were directed to the Master Capex page, while Employee users were directed to the CIP Report page.

Testing on the Master Capex Management feature demonstrated that Staff OC users could successfully perform data management operations, including adding new Capex records, updating existing data, deleting unnecessary records, and searching data using filtering parameters. The system successfully stored and retrieved Capex information from the database, indicating that the data management process operated according to the defined requirements.

The Cumulative Balance Management feature was also tested to ensure that users could manage financial progress information related to Capex projects. The testing results showed that Staff OC users were able to add, modify, delete, and update cumulative balance data successfully. This feature supports users in monitoring the development of project budgets and investment realization.

The document management functionality, including the upload and management of FAGLB and ZLIS1 documents, was tested to verify the capability of the system in handling supporting files. The results indicated that users could upload, modify, and manage documents properly. The system successfully stored document information and allowed users to access the required files when needed.

Testing was also performed on the User Management and Requester Management features. The results showed that authorized users could manage user accounts, including creating, updating, and removing user data. The access control mechanism worked properly by ensuring that only users with administrator privileges could manage system accounts.

Furthermore, the reporting features were tested to evaluate the ability of the system to generate Capex information. The CIP Report, Category Report, and Summary Report features successfully displayed data based on selected parameters, including category, period, and user-related information. Staff OC users could access comprehensive Capex reports, while Employee users could only view reports related to their respective data. The system also successfully generated downloadable reports in PDF format.

Based on the overall Black Box Testing results, all tested scenarios were successfully executed and produced outputs according to the expected results. Therefore, the developed Web-Based Capital Expenditure Information System meets the defined functional requirements and is capable of supporting Capex data management activities more effectively. The testing results also indicate that the implemented features are reliable, user access

control is properly applied, and the system can reduce manual processing activities in Capex monitoring and reporting workflows.

## 4. Conclusions

This research aims to develop a web-based Capital Expenditure Information System to support Capex data management and monitoring processes in a more structured and efficient manner. The developed system was built using the Laravel framework and designed to simplify user access to Capex information without requiring manual processing through the SAP ERP system or Microsoft Excel. Based on the Black Box Testing results, all developed features functioned properly according to user requirements. The system provides three main features: Master Capex, which is used to manage primary Capex data. Cumulative Balance, which is used to monitor project balance development. Document Upload, which supports uploading documents such as FAGLB and ZLIS1.

The implementation of this system improves work efficiency, reduces data entry errors, and accelerates the reporting process. With a user-friendly interface and features that meet user requirements, this system can become an effective digital solution for supporting Capital Expenditure management. Future development can focus on integrating the system with existing ERP platforms, improving data visualization, and adding more advanced analytical features.

## References


[1] S. C. Augustpaosa Nariman, "Analisis Faktor-Faktor yang Mempengaruhi Capital Expenditure", JE, vol. 26, no. 11, pp. 17–36, Sep. 2022.

[2] W. Mulya, "Capital Expenditure Dan Operational Expenditure Dalam Perancangan Instalasi Pengolahan Air di Kota Balikpapan," INFO-TEKNIK, vol. 23, no. 1, pp. 15-28, 2022.

[3] I. Islamiyah and F. Fidiana, "Pengaruh Capital Expenditure, Corporate Social Responsibility Dan Profitabilitas Terhadap Nilai Perusahaan," Jurnal Ilmu dan Riset Akuntansi (JIRA), vol. 13, no. 9, 2024.

[4] N. Nurlatifah and P. Purwatiningsih, "Pengaruh Capital Expenditure, Risiko Keuangan dan Pertumbuhan Penjualan terhadap Nilai Perusahaan", jmeb, vol. 4, no. 3, pp. 1307–1316, Sep. 2024.

[5] M. Siregar, "Sistem Informasi Manajemen Anggaran Keuangan dan Pendapatan Perusahaan Berbasis Web Pada PT. Arelosoft Inovasi Dunia," OKTAL: Jurnal Ilmu Komputer dan Sains, vol. 1, no. 11, pp. 2133–2143, 2022.

[6] Larasathi, I. G. A. I. S., Estiyanti, N. M., & Satwika, I. P. (2020). Sistem Informasi Usulan Anggaran Dana Desa Berbasis Web Di Kantor Desa Petang. Jurnal Teknologi Informasi Dan Komputer, 6(2).

[7] Ardi, N,. Silalahi, R. "Usability Analysis of Online Loan Application System using the System Usability Scale (SUS)", *astj* , vol. 1, no. 1, pp. 27–32, Jul. 2025, .

[8] S. Suhari, A. Faqih, and F. M. Basysyar, "Sistem Informasi Kepegawaian Mengunakan Metode Agile Development di CV. Angkasa Raya," Jurnal Teknologi Dan Informasi, vol. 12, no. 1, pp. 30-45, 2022.

[9] C. A. A. Binangkit, A. Voutama, and N. Heryana, "Pemanfaatan UML (Unified Modeling Language) dalam perencanaan sistem pengelolaan sewa alat musik berbasis website," JATI (Jurnal Mahasiswa Teknik Informatika), vol. 7, no. 2, pp. 1429–1436, 2023.

[10] Ardi, N., Lubis, A.I. and Arrafi, I.A.S., 2025. Analysis of docker container implementation in SIEM infrastructure. *Journal of Applied Informatics and Computing, 9*(3), pp.973-978.

[11] H. Diandara Tresya, J. A. Ramadhan, and A. Suharso, "Systematic Literature Review Penggunaan Metodologi Pengembangan Sistem Informasi Waterfall, Agile, dan Hybrid," JIEET (Journal of Information Engineering and Educational Technology), vol. 7, no. 1, pp. 36–42, 2023.

[12] Ardi, N., Razali, S.M. and Thohari, A.H., 2025, December. Web-Based Ground Control Station for Real-Time Monitoring Quadcopter UAVs. In *8th International Conference on Applied Engineering (ICAE 2025)* (pp. 132-150). Atlantis Press.

[13] Huda, A., & Ardi, N. (2021). Predictive Analytic on Human Resource Department Data Based on Uncertain Numeric Features Classification. *International Journal of Interactive Mobile Technologies (iJIM)*, *15*(08), pp. 172–181.

[14] Ardi, N., Lubis, A. I. ., & Isnayanti. (2023). Decision Tree for Predicting the Mortality in Hemodialysis Patient with Diabetes. *Jurnal Minfo Polgan*, *12*(1), 346-356.

[15] Ardi, Noper, S. Supardianto, and Ahmadi Irmansyah Lubis. "Predicting missing value data on IEC TC10 datasets for dissolved gas analysis using tertius algorithm." *Journal of Applied Informatics and Computing* 7, no. 1 (2023): 50-56.

[16] Ardi, N., Adri, M. and Azhar, N., 2022. Implementasi Arsitektur Hierarchical Model View Controller (HMVC) Dalam Portal Akademik. *Jurnal Teknik Komputer dan Informatika*, *1*(1| Agustus), pp.21-30.

[17] Ardi, N. and Isnayanti, I., 2022. Implementasi Artificial Neural Network dalam Memprediksi Jumlah Peserta Les Bahasa Inggris Menggunakan Metode Back Propagation (Studi Kasus di Lembaga Kursus Global English). *Jurnal Teknik Komputer dan Informatika*, *1*(1| Agustus), pp.45-51.

[18] N. Ardi and D. A. Azhari, "Analisis Aplikasi Lembur Berbasis Web menggunakan System Usability Scale (SUS) Pada PT. XYZ", *J. Ilm. Inform. Glob.*, vol. 16, no. 1, pp. 10–19, Apr. 2025.

[19] Ardi, N., 2025. Generative AI: A Comprehensive Review of Foundational Models and Emerging Methods. *Applied Systems and Technologies Journal (ASTJ)*, *1*(1), pp.1-6.