



INTERNSHIP REPORT

DESIGN AND DEVELOPMENT OF AN AUTOMATED FIELD OPERATIONS MANAGEMENT SYSTEM (SAFED) TO IMPROVE EQUIPMENT CHECKLIST AND DIGITAL REPORTING USING PERSONAL EXTREME PROGRAMMING: A CASE STUDY AT PT. BINTANG INDAYU GEMILANG.

NORBERTINO EURAKHA NANDATOTI
10231071

Academic Supervisor
Henokh Lugo Hariyanto, S.Si.,M.Sc.

INFORMATION SYSTEM

Department of Electrical Engineering, Informatics, and Business
Faculty of Science and Information Technology
Kalimantan Institute of Technology
Balikpapan, 2026

APPROVAL PAGE

**WORK PRACTICAL REPORT
DEVELOPMENT DEPARTMENT PT. BINTANG INDAYU
JANUARY - FEBRUARY 2026 PERIOD**



PREPARED BY

Norbertino Eurakha Nandatoti

REVIEWED AND APPROVED BY

**INFORMATION SYSTEM
COORDINATOR**

A handwritten signature in black ink, appearing to read 'Rahayu'.

Sri Rahayu Natasia, S.Komp, M.Si., M.Sc
NIP. 199001082020122003

ACADEMIC SUPERVISOR

A handwritten signature in black ink, appearing to read 'Henokh Lugo Hariyanto'.

Henokh Lugo Hariyanto, S.Si., M.Sc.
NIP. 199303062022041001

APPROVAL SHEET

WORK PRACTICAL
DEVELOPMENT DEPARTMENT PT BINTANG INDAYU GEMILANG
JANUARY - FEBRUARY 2026 PERIOD

SAFEBOX.

PREPARED BY

NORBERTINO EURAKHA NANDATOTI

10231071

REVIEWED AND APPROVED BY

HEAD OF DEVELOPMENT DEPARTMENT
PT BINTANG INDAYU GEMILANG

FIELD SUPERVISOR

 **SAFEBOX.**
DANI IKHSAN
Signed: Mar 13, 2026-18:57:46 GMT +0800

 *Haidar Dzaky*
Signed: Mar 13, 2026-18:24:38 GMT +0800

MUHAMMAD PRIANDANI
NUR IKHSAN

HAIDAR DZAKY SUMPENA, S.Kom.



STATEMENT OF ORIGINALITY WORK PRACTICAL REPORT

I, the undersigned:

Name / Student ID : Norbertino Eurakha Nandatoti / 10231071

Study Program : Information System

Department : Department of Electrical Engineering, Informatics, and Business

Faculty : Faculty of Science and Information Technology

Hereby solemnly declare that the Practical Work Report entitled: “Design and Development of an Automated Field Operations Management System (SAFED) to Improve Equipment Checklist and Digital Reporting Using Personal Extreme Programming: A Case Study at PT. Bintang Indayu Gemilang.” is entirely my own original work. This report has not been copied, plagiarized, or unlawfully derived from the work of others. All sources of information and references used in the preparation of this report have been properly acknowledged in accordance with academic standards.

Should any evidence of plagiarism or academic misconduct be discovered at any time in the future, I fully understand and accept the consequences and am willing to receive sanctions in accordance with the academic regulations of Kalimantan Institute of Technology.

This statement is made truthfully, consciously, and without any coercion from any party.

Balikpapan, 13 Maret 2026



Norbertino Eurakha Nandatoti
10231071

PREFACE

All praise and gratitude be to God Almighty for His blessings and grace, which have enabled me to complete this Practical Work Report entitled :

Design and Development of an Automated Field Operations Management System (SAFED) to Improve Equipment Checklist and Digital Reporting Using Personal Extreme Programming: A Case Study at PT. Bintang Indayu Gemilang.

This Work Practice Report is submitted as one of the requirements to complete the Bachelor's Degree Program in the Information Systems Study Program, Department of Electrical Engineering, Informatics, and Business, Faculty of Science and Technology, Kalimantan Institute of Technology (ITK), Balikpapan. On this occasion, I would like to express my deepest gratitude to :

1. My beloved parents, who have always provided unconditional support in every form, enabling me to reach this stage of my academic journey.
2. Mr. Henokh Lugo Hariyanto, S.Si.,M.Sc., as my Academic Supervisor, who has consistently provided direction, guidance, and constructive suggestions during the completion of this work practice.
3. Mr. Haidar Dzaky Sumpena, S.Kom., as my Field Supervisor, who has guided me and provided me with opportunities to learn and grow throughout this work practice.
4. Mr. Muhammad Priandani Nur Ikhsan, as the Head of the Development Department, who provided me with the opportunity to conduct this work practice.
5. Mr. Najibullah Muhariri, as my mentor and advisor during this work practice, who continuously offered valuable advice, support, and insights into the professional world that I had not previously experienced.
6. Mr. Adi Mahmud Jaya Marindra, as the Dean of the Faculty of Science and Technology, ITK.
7. Mr. M. Ihsan Alfani Putera, S.Tr.Kom., M.Kom., as the Head of the Department of Electrical Engineering, Informatics, and Business, ITK.

8. Ms. Sri Rahayu Natasia, S.Komp, M.Si., M. Sc., as the Coordinator of the Information Systems Study Program, Department of Electrical Engineering, Informatics, and Business, Faculty of Science and Technology, ITK.
9. All colleagues during my work practice who continuously provided encouragement and support throughout the program.
10. All my friends who consistently supported and accompanied me during the preparation of this Work Practice Report.

We realize that this Practical Report is still far from perfect. Therefore, constructive criticism and suggestions are highly appreciated. I sincerely hope that this work practice report will provide benefits to all parties involved. Thank you for your attention

Balikpapan, 11 March 2025

A handwritten signature in black ink, consisting of a large loop followed by a horizontal stroke and a diagonal line extending upwards and to the right.

Norbertino Eurakha Nandatoti

**DESIGN AND DEVELOPMENT OF AN AUTOMATED FIELD OPERATIONS
MANAGEMENT SYSTEM (SAFED) TO IMPROVE EQUIPMENT CHECKLIST
AND DIGITAL REPORTING USING PERSONAL EXTREME
PROGRAMMING: A CASE STUDY AT PT. BINTANG INDAYU GEMILANG.**

Student Name	: Norbertino Eurakha Nandatoti
Student ID	: 10231071
Name of Internship Partner	: PT Bintang Indayu Gemilang
Supervising Lecture	: Henokh Lugo Hariyanto, S.Si., M.Sc.
Field Supervisor	: Haidar Dzaky Sumpena, S.Kom.

ABSTRACT

Field operations in a manufacturing service environment require structured recording, progress monitoring, and operational documentation processes. At PT Bintang Indayu Gemilang, field operations management in the Manufacturing Division was previously carried out manually using checklist documents and spreadsheets that were not integrated, making it difficult to monitor work progress and compile operational reports. Therefore, during this internship, a system called SAFED (System Automation for Field) was developed to help manage field operations in a more structured and integrated manner. This system was developed using the Personal Extreme Programming (XP) method, which allows the development process to be carried out iteratively through several development cycles. The SAFED system was developed as a web-based application using the Nuxt.js framework with a PostgreSQL database. The main features of the system include user authentication, user management, operations management, activity checklist management, work progress monitoring, and automatic generation of field activity reports. The system was tested using the User Acceptance Testing (UAT) method to ensure that each feature of the system works according to user requirements. The results of the development show that the SAFED system can help improve the efficiency of field operations management, facilitate the monitoring of work progress, and support the documentation and reporting of operational activities more effectively.

Keyword : SAFED, Field Operations Management, Personal Extreme Programming, Operational Information Systems, User Acceptance Testing

**RANCANG BANGUN *AUTOMATED FIELD OPERATIONS MANAGEMENT*
SYSTEM (*SAFED*) UNTUK MENINGKATKAN EFISIENSI *CHECKLIST*
PERALATAN DAN PELAPORAN DIGITAL MENGGUNAKAN METODE
PERSONAL EXTREME PROGRAMMING: STUDI KASUS PADA PT. BINTANG
INDAYU GEMILANG**

Nama Mahasiswa	: Norbertino Eurakha Nandatoti
NIM	: 10231071
Nama Mitra KP	: PT Bintang Indayu Gemilang
Dosen Pembimbing	: Henokh Lugo Hariyanto, S.Si., M.Sc.
Pembimbing Lapangan	: Haidar Dzaky Sumpena, S.Kom.

INTISARI

Kegiatan operasional lapangan pada lingkungan layanan manufaktur memerlukan proses pencatatan, pemantauan progres pekerjaan, serta dokumentasi operasional yang terstruktur. Pada PT Bintang Indayu Gemilang, proses pengelolaan operasi lapangan pada Divisi Manufaktur sebelumnya masih dilakukan secara manual menggunakan dokumen *checklist* dan *spreadsheet* yang tidak terintegrasi, sehingga menyulitkan pemantauan progres pekerjaan serta penyusunan laporan operasional. Oleh karena itu, dalam kegiatan Kerja Praktik ini dikembangkan sebuah sistem bernama SAFED (*System Automation for Field*) untuk membantu pengelolaan aktivitas operasional lapangan secara lebih terstruktur dan terintegrasi. Sistem ini dikembangkan menggunakan metode *Personal Extreme Programming (PXP)* yang memungkinkan proses pengembangan dilakukan secara iteratif melalui beberapa siklus pengembangan. Sistem SAFED dikembangkan sebagai aplikasi berbasis web menggunakan *framework Nuxt.js* dengan basis data *PostgreSQL*. Fitur utama sistem meliputi autentikasi pengguna, manajemen pengguna, manajemen operasi, pengelolaan *checklist* kegiatan, pemantauan progres pekerjaan, serta pembuatan laporan kegiatan lapangan secara otomatis. Pengujian sistem dilakukan menggunakan metode *User Acceptance Testing (UAT)* untuk memastikan setiap fitur sistem berjalan sesuai kebutuhan pengguna. Hasil pengembangan menunjukkan bahwa sistem SAFED dapat membantu meningkatkan efisiensi pengelolaan operasi lapangan, mempermudah pemantauan progres pekerjaan, serta mendukung proses dokumentasi dan pelaporan kegiatan operasional secara lebih efektif.

Keyword : SAFED, Manajemen Operasi Lapangan, Personal Extreme Programming, Sistem Informasi Operasional, *User Acceptance Testing*

TABLE OF CONTENTS

APPROVAL SHEET.....	2
APPROVAL SHEET.....	3
STATEMENT OF ORIGINALITY	
WORK PRACTICAL REPORT.....	4
PREFACE.....	5
ABSTRACT.....	7
INTISARI.....	8
TABLE OF CONTENTS.....	9
LIST OF FIGURES.....	12
LIST OF TABLE.....	15
CHAPTER 1	
INTRODUCTION.....	1
1.1 Background.....	1
1.2 General Objective.....	3
1.3 Specific Objectives.....	3
1.4 Benefits of Practical Work.....	4
CHAPTER 2	
OVERVIEW OF PT. BINTANG INDAYU GEMILANG.....	5
2.1. History.....	5
2.2. Vision Mission.....	5
2.3. Location.....	5
2.4. Company Structure.....	6
2.5. Department Description.....	6
CHAPTER 3	
LITERATURE REVIEW.....	8
3.1. Operations Management System.....	8
3.2. Field Operations and Digital Reporting.....	8
3.3. System Development Life Cycle (SDLC).....	9
3.3.1 Agile Development Method.....	9
3.3.1.1 Extreme Programming (XP).....	10
3.3.1.2 Personal Extreme Programing (PXP).....	10
3.3.2 Kanban.....	13
3.4 Unified Modeling Language (UML).....	14
3.4.1 Use Case Diagram.....	15
3.4.2 Database Schema.....	17
3.5 Data.....	19
3.6 PostgreSQL.....	19
3.7 Object Relational Mapping (ORM).....	20
3.8 Nuxt.js Framework.....	20
CHAPTER 4	
METHODOLOGY.....	22

4.1. Place and Location.....	22
4.2. Activity Stages.....	22
4.2.1 Preliminary Orientation and Environmental Analysis.....	22
4.2.2 Stakeholder Interview and Requirement Identification.....	23
4.2.3 System Requirement Analysis.....	23
4.2.4 System Development.....	24
4.2.5 Documentation and Reporting.....	24
4.3 Data Collection Methods.....	24
4.3.1 Observation.....	24
4.3.2 Interviews.....	25
4.3.3 Documentation Study.....	25
4.3.4 Literature Study.....	25
4.4 System Development Method.....	25
4.4.1 Personal Extreme Programming (XP).....	26
4.4.2 Kanban Implementation.....	27
4.5 System Design.....	27
4.5.1 Use Case Diagram.....	27
4.5.2 Database Schema.....	31
4.6 Product Backlog and Iteration Planning.....	31
4.7 Analysis and Evaluation Techniques.....	34
CHAPTER 5	
RESULTS AND DISCUSSION.....	36
5.1 System Development Results.....	36
5.1.1 System Overview.....	36
5.1.2 Iteration Development Process.....	39
5.1.2.1 Iteration 1: Core System Initialization.....	40
5.1.2.2 Iteration 2: Operation Management Features.....	41
5.1.2.3 Iteration 3: Operational Execution and Reporting Features.....	43
5.2 System Feature Implementation.....	44
5.2.1 User Authentication Interface.....	45
5.2.2 User Management Module.....	45
5.2.3 Tools Management Module.....	47
5.2.4 Operation Management Module.....	48
5.2.5 Checklist Execution Module.....	50
5.2.6 Operation Monitoring Module.....	51
5.2.7 Field Report Generation Module.....	51
5.3 System Testing.....	52
5.3.2 User Acceptance Testing Results.....	53
5.4 System Evaluation and Analysis.....	71
5.4.1 System Impact on Operational Workflow.....	71
5.4.2 Development Challenges and Solutions.....	72
CHAPTER 6	
CONCLUSION AND RECOMMENDATIONS.....	74

6.1. Conclusion.....	74
6.2 Recommendations.....	75
6.2.1 Recommendation for the Company.....	75
6.2.2 Recommendation for the Institution.....	75
6.2.3 Recommendation for Future Students.....	76
REFERENCES.....	77
ATTACHMENT.....	80
Weekly Log Sheet.....	80
Photos of Field Work Activities.....	80
Practical Work Completion Certificate.....	80

LIST OF FIGURES

Figure 2.1 Company Structure.....	6
Figure 3.1 PXP Process Phase.....	11
Figure 3.2 Kanban Workflow Visualization.....	13
Figure 3.3 Classification of UML Diagrams.....	15
Figure 4.1 Practical Work Activity Flow.....	23
Figure 4.2 SAFED System Use Case Diagram.....	30
Figure 4.3 SAFED Database Schema.....	33
Figure 5.1: System Architecture Overview.....	40
Figure 5.2: Kanban Board Implementation.....	42
Figure 5.3 Login Interface.....	47
Figure 5.4 User Management Interface.....	48
Figure 5.5 User Creation Form.....	49
Figure 5.6 Tools Management Interface.....	50
Figure 5.7 Operation List Interface.....	51
Figure 5.8 Operation Creation Form.....	52
Figure 5.9 Checklist Execution Interface.....	52
Figure 5.10 Operation Monitoring Interface.....	5
Figure 5.11 Field Report Generation Interface.....	54

LIST OF TABLE

Table 3.1 Use Case Element.....	16
Table 3.2 Database Schema Element.....	17
Table 4.1 Use Case Description.....	29
Table 4.2 Product Backlog.....	32
Table 5.1 User Roles.....	37
Table 5.2 Iteration 1 Development Tasks.....	41
Table 5.3 Iteration 2 Development Tasks.....	42
Table 5.4 Iteration 3 Development Tasks.....	43
Table 5.5 User Acceptance Testing Results.....	54

CHAPTER 1

INTRODUCTION

1.1 Background

The effectiveness of field operation in manufacturing service environments is highly influenced by the accuracy of equipment preparation, the efficiency of reporting processes, and the ability of management to monitor ongoing activities. In companies that provide installation, commissioning, maintenance, and system upgrades, operational delays and service inconsistencies are frequently caused by insufficient equipment control and fragmented documentation systems. Research in the field of equipment maintenance and management shows that improper classification and management of equipment can have a significant impact on maintenance performance and operational efficiency (Roda et al., 2014). Additionally, the implementation of digital integration in the manufacturing environment has been proven to improve operational transparency and decision making accuracy through better system connectivity and improved data visibility (Tortorella & Diego Fettermann, 2018) . These findings indicate that effective field operations management requires the support of a structured, integrated, and automated system.

Several operation challenges have been identified at PT. Bintang Indayu Gemilang, particularly in the Manufacturing Division, which is responsible for the installation, commissioning, maintenance, and upgrading of ship monitoring systems. Based on interviews with the Head of the Manufacturing Division, the existing equipment checklist is still general and not adjusted to specific types of operations. This often leads to discrepancies between the equipment being prepared and the actual needs in the field, potentially causing delays and additional logistical adjustments. In order to prepare and process the Completion Report (*Berita Acara/BA*), manual processes are still used and not integrated into a system. This manual process increases the risk of reporting delays exceeding the operational standard of more than 2 days and leads to scattered documentation across spreadsheet and messaging platforms. This results in limitations for management in monitoring task progress and evaluating operational performance.

The importance of integrating operational activities with digital information systems has been emphasized in industry 4.0 and digital operations management literature. Digital integration enables centralized data management, better coordination,

and enhanced visibility of ongoing activities, which support more effective decision-making processes (Tortorella & Diego Fettermann, 2018). Similarly, research on smart manufacturing highlights that operational transparency and monitoring are essential components for improving responsiveness and control in service-oriented manufacturing environments (Shi et al., 2020). Although previous studies and implementations have developed digital reporting or monitoring platforms, many of them focused on recording activities and storing documents without integrating adaptive equipment checklists based on specific operation types into integrated field operations management systems.

In contrast to previous implementations, this practical work introduces a different perspective through the design and development of the Automated Field Operations Management System (SAFED). This system integrates adaptive equipment checklist automation, centralized digital reporting, and structured task progress monitoring into a single platform. Furthermore, this system was developed using the Personal Extreme Programming (PXP) approach, which was emphasized with short development iterations, continuous testing and incremental refinement in a single development environment. Although agile methodologies are commonly applied in team based systems development, the application of PXP in industrial case studies focused on field operations management provides methodological contributions.

Based on observations at PT. Bintang Indayu Gemilang, there are several problems that can be identified. First, the current equipment checklist system is not flexible enough for various types of operations, which increases the risk of human error and equipment mismatches during field operations. Second, manual and non-integrated reporting processes create unclear validation mechanisms and limit the transparency in task progress monitoring, making it difficult for management to ensure that operational standards are consistent. Third, documentation practices remain inefficient and fragmented across multiple platforms, resulting in disorganized records, potentially inconsistent data, and challenges in accessing information for evaluation. These conditions collectively indicate that existing field operations management processes are lacking in automation, structured validation, and centralized control. Therefore, it is necessary to design and develop an Automated and Centralized Field Operations Management System (SAFED) to reduce human error, improve task validation

transparency, increase documentation efficiency, and facilitate structured monitoring of operational activities in the Manufacturing Division of PT. Bintang Indayu Gemilang.

1.2 General Objective

The general objective of this practical work includes several things that we want to achieve. Here are some of the general objectives we want to achieve in the implementation of this activity :

1. Applying knowledge and theory in the field of Information Systems to real work practices at PT. Bintang Indayu Gemilang in order to understand business processes, work systems, and the issues faced by the company.
2. Learning about work dynamics, organizational, structure, and professionalism environments in the manufacturing and technical service industries.
3. Fulfilling the number of Semester Credit Units (*SKS*) required as an academic requirement in the Information Systems Study Program at the Kalimantan Institute Of Technology

1.3 Specific Objectives

The specific objectives of this practical work include several things that we want to achieve. Here are some of the specific objectives we want to achieve in the implementation of this activity :

1. Conducting an analysis of user requirements for the reporting and monitoring system for the field team at PT. Bintang Indayu Gemilang.
2. Designing and developing the SAFED (System Automation for Field) information system to support the operational activities of the field team at PT. Bintang Indayu Gemilang.
3. Providing a transparent platform for monitoring field activity progress for Observers and Implementation Managers.
4. Conducting system testing to ensure that the developed features function as expected.

1.4 Benefits of Practical Work

This practical work is expected to provide benefits and positive contributions to both academics and industry through the development of the SAFED system and direct involvement in company operations, with the following benefits:

A. For College Students

- a. Improving technical skills in developing information systems, from the needs analysis stage to the implementation of automation features.
- b. Training problem-solving skills to deal with operational obstacles that occur in real industries, especially in field management.
- c. Sharpening soft skills such as professional communication and time management in a manufacturing industry work environment.

B. For Company

- a. Accelerate the digital transformation of the Manufacturing Division through document and checklist automation.
- b. Simplifies management in monitoring activity progress transparently in order to improve decision-making processes.
- c. Ensuring the security and availability of operational activity documentation data through a centralized digital storage system.

C. For Academics

- a. Building synergy and strategic cooperation between universities and the manufacturing industry.
- b. Serving as a reference or case study material for curriculum development that is relevant to current industry needs.

CHAPTER 2

OVERVIEW OF PT. BINTANG INDAYU GEMILANG.

2.1. History

PT. Bintang Indayu Gemilang is a logistics and technology company located in Balikpapan. Since 2022, PT Bintang Indayu Gemilang has been developing the Internet of Things (IoT) to expand their business lines. The company provides an Engine Monitoring System (EMS) that currently focuses on monitoring ship engines and calculating fuel consumption. With this EMS, ship owners and related stakeholders can perform semi-real-time monitoring. This monitoring helps them to track ship movements and also check whether the ship is operating in accordance with established procedures. The EMS also provides a report feature to summarize the ship's activities during its voyage. Currently, PT Bintang Indayu Gemilang is improving the quality of its services to maintain its relationship with clients.

2.2. Vision Mission

In conducting its business operations, PT Bintang Indayu Gemilang refers to its established vision and mission. The following are the vision and mission of PT Bintang Indayu Gemilang.

2.2.1 Vision

The vision of PT Bintang Indayu Gemilang is “To make the maritime world more transparent and trustworthy through data-driven operations.”

2.2.1 Mission

The mission carried out by PT Bintang Indayu Gemilang is “Supporting data-based maritime operations for 1,000 ships by 2030.”

2.3. Location

PT Bintang Indayu Gemilang is located at Jalan Ruhui Rahayu No. 115, South Balikpapan, Balikpapan.

2.4. Company Structure

PT Bintang Indayu Gemilang has a structured organizational structure to ensure smooth operations and business management. This structure reflects the division of tasks at various levels of leadership and work units in support of the company's vision and mission. The following is the organizational structure of PT Bintang Indayu Gemilang.

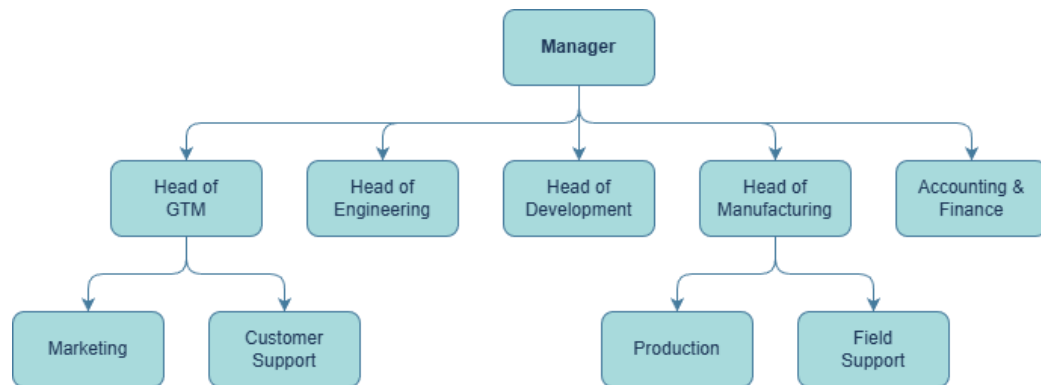


Figure 2.1 Company Structure
Source : Company Documentation, 2026

2.5. Department Description

During the practical work program, the student was assigned to the Development Division under the supervision of the Head of Development at PT. Bintang Indayu Gemilang. This division is responsible for developing and maintaining the company's internal digital systems and technology platforms that support operational activities across different divisions within the organization. The Development Division focuses on designing, building, and improving information systems that enhance operational efficiency, data management, and system integration within the company.

One of the operational units supported by the Development Division is the Manufacturing Division, which is responsible for field activities such as installation, commissioning, maintenance, and system upgrades of ship monitoring equipment installed on maritime vessels. These operational activities require proper coordination, equipment preparation, and structured documentation to ensure that field operations are carried out efficiently and according to established operational standards.

During the internship, the student was involved in analyzing operational workflows related to field activities within the Manufacturing Division. The activities included observing operational processes, identifying challenges in equipment preparation and

documentation practices, and studying how operational data was managed during field operations. These activities provided an understanding of the operational needs that later became the basis for designing a system-based solution to support field operation management.

CHAPTER 3

LITERATURE REVIEW

3.1. Operations Management System

Operational management involves planning, coordination, and control of organizational processes that transform inputs into valuable services or products. In a service-oriented manufacturing environment, operations management goes beyond production control and includes scheduling, resource allocation, performance monitoring, and quality assurance (Heizer et al., 2019). Effective operations management ensures that organizational activities are carried out efficiently, consistently, and according to established standards.

Operations in industrial service companies often take place in field environments, requiring coordination between technical personnel. According to Slack et al., (2022), operational systems need to integrate information flow with physical processes so that visibility, responsiveness, and control are ensured. Without a structured operational system, organizations may experience inefficiencies, inconsistent service performance, and poor managerial oversight.

Field operations particularly require structured task validation and accurate equipment management. Tsang (2002) emphasizes that the effectiveness of field services and maintenance is highly dependent on systematic coordination, standard procedures, and monitoring mechanisms. Therefore, the implementation of an operation management system supported by digital integration is crucial to improving operational consistency and transparency.

3.2. Field Operations and Digital Reporting

Field operations are operational activities that happen outside the main office, usually involving installing, inspecting, maintaining, or even upgrading systems at client sites. These activities need the right equipment, structured task execution, and systematic documentation to make sure that services are reliable. Improper classification and management of equipment can significantly affect operational efficiency and increase the probability of delay in service delivery (Roda et al., 2014). Meanwhile, fragmented documentation often leads to inconsistencies in data and poor managerial visibility.

Digital reporting systems tackle these challenges by facilitating centralized data storage, automated document generation, and structured validation processes. The integration of digital technology into operational environments has been shown to improve transparency, decisional accuracy, and coordination efficiency (Tortorella & Diego Fettermann, 2018). Smart manufacturing research shows that operational visibility and monitoring are super important for improving responsiveness in service-oriented industrial systems (Shi et al., 2020).

3.3. System Development Life Cycle (SDLC)

The system development life cycle (SDLC) is a structured framework used to plan, design, develop, test, and maintain information systems. SDLC provides a systematic process that guides developers through several phases to ensure that the resulting system meets the users requirements and operates effectively within the organizational environment. According to Pressman & Maxim (2019), SDLC aims to ensure that software development is carried out in a controlled and organized manner, minimizing risk while maintaining product quality throughout the development process.

SDLC traditionally consists of several sequential phases, including planning, requirements analysis, system design, implementation, testing, deployment, and maintenance. Each phase has its own specific objectives and deliverables that support the next stage of development. This structured approach allows developers to monitor progress, maintain documentation, and ensure that system requirements are clearly defined before the development process begins (Sommerville, 2016).

Traditional SDLC models such as the waterfall model assume that the system requirements remain static during the development process. Today's software development environment, especially for operational systems and evolving user needs, require the ability to change requirements during the development process. That means a more adaptive development approach is needed to deal with these changes efficiently. One such approach is the Agile development methodology, which introduces an iterative and flexible approach to the concept of SDLC.

3.3.1 Agile Development Method

In the process of dynamic software development, this study uses the Agile software development methodology as an approach in designing and implementing information

systems. Agile is a software development methodology that prioritizes flexibility, collaboration, and iterative processes enabling developers to respond quickly and efficiently to changes in user requirements (Martin, 2003).

Agile development focuses on short development cycles, known as iterations or sprints, when the system features are developed, evaluated, and refined incrementally. Through continuous development cycles, Agile allows the system to be evaluated and refined repeatedly based on user feedback, ensuring the final system meets operational requirements. Agile promotes values such as customer collaboration, adaptive planning, and continuous improvement through the development lifecycle, according to Beck et al, (2001). These characteristics make agile particularly well suited for projects where system requirements may involved during the development process, such as operational management systems that are developed in an industrial environment.

3.3.1.1 Extreme Programming (XP)

Extreme Programming (XP) is a software development method based on the Agile methodology framework introduced by Kent Beck. XP emphasizes iterative development, continuous testing, and the ability to adapt when requirements change, during the software development process (Beck, 2012). This methodology supports frequent delivery, rapid feedback cycles, and close alignment between system functionality and user needs. XP provides several technical practices aimed at improving software quality and development efficiency, including incremental development, test-driven development, and continuous refactoring. These practices enable developers to maintain code quality while accommodating evolving requirements through the development lifecycle.

Although XP was initially designed for team-based development environments, its principles and practices were later adapted for development carried out by a single developer. This adaptation resulted in the Personal Extreme Programming (PXP) approach, which maintains the core values and practices of XP while adjusting the process structure to suit in an independent development environment.

3.3.1.2 Personal Extreme Programing (PXP)

Personal Extreme Programming (PXP) is an Agile-based software development approach specifically designed for independent developers. This methodology was

introduced by Dzhurov, Krasteva, and Ilieva (2009) as an adaptation of Personal Software Process (PSP) by integrating several technical practices from XP. The main goal of PXP is to simplify the software development process while maintaining product quality and increasing individual developer productivity through iterative development and continuous evaluation.

Compared to the PSP, which requires comprehensive documentation and detailed measurement activity, PXP offers a lighter and more flexible development process. This methodology uses short development iterations ranging from one to three weeks, allowing developers to deliver iterative improvements while continuously monitoring progress and performance. The PXP development model consists of several structured phases that guide the development process. These phases include requirement, planning, iteration initialization, design, implementation, system testing, and retrospective evaluation.

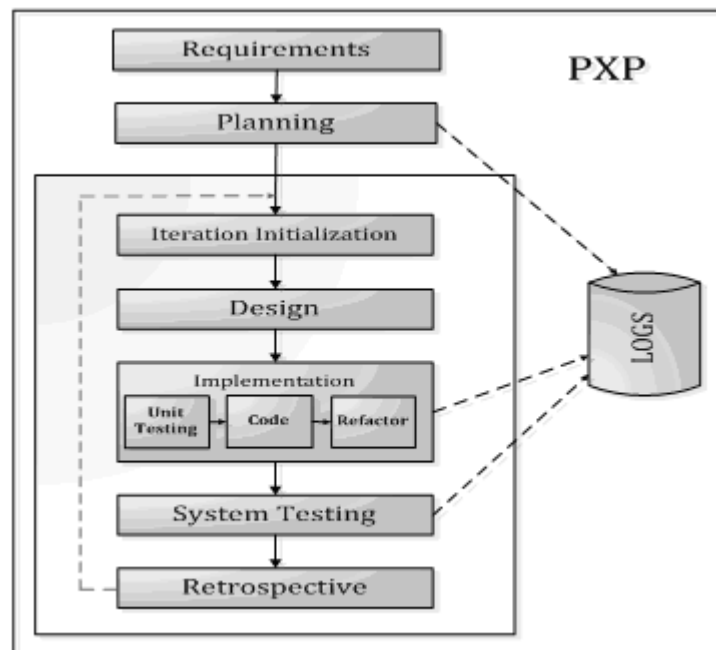


Figure 3.1 PXP Process Phase

Source : Dzhurov et al, 2009

Based on the process model illustrated in Figure 3.1, the phase of PXP can be described as follows:

1. Requirements

The requirements phase is the initial stage in the development process where

developers identify and document the functional and non-functional requirements of the system. These requirements represent the expected capabilities of the system and form the basis for further development activities.

2. Planning

During this phase, developers organize the identified requirements into a list of development tasks. These tasks are divided into smaller, more manageable components. Each task is then estimated based on its difficulty and previous development experience to improve the accuracy of the time estimate.

3. Iteration Initialization

This phase marks the beginning of each development iteration cycle. Developers can select several tasks to be completed in the current iteration cycle. Each iteration focuses on a specific group of tasks and can result in additional system features or improvements.

4. Design

The design phase involves developers modeling the components and system structure needed to implement the selected tasks. The design process follows the principle of simplicity, focusing on meeting current system requirements without attempting to predict future functionality.

5. Implementation

The implementation phase involves developing the system components defined in the design phase. This phase includes unit testing, code development, and refactoring activities to ensure the software maintains a clean structure and high code quality.

6. System Testing

System testing is performed to verify that all implemented features operate correctly and meet the specified requirements. During this phase, developers identify defects or inconsistencies in the system and make the necessary corrections.

7. Retrospective

The retrospective phase marks the end of each development iteration. In this phase, developers evaluate the development process by comparing the estimated task completion time with the actual development time. Developers also identify

potential improvements to increase the efficiency and effectiveness of future iterations.

With its structured yet flexible development approach, PXP is considered suitable for software projects carried out by a single developer, especially in dynamic development environments that require rapid adaptation to changing requirements.

3.3.2 Kanban

Kanban is a workflow management method used for visualizing tasks, controlling work progress, and improving process efficiency through continuous delivery. This concept was originally developed for Toyota's production systems and later adapted for knowledge-based work environments and software development. In software development, Kanban helps development teams manage work items through a visual system that represents the stages of task completion (Anderson, 2010).

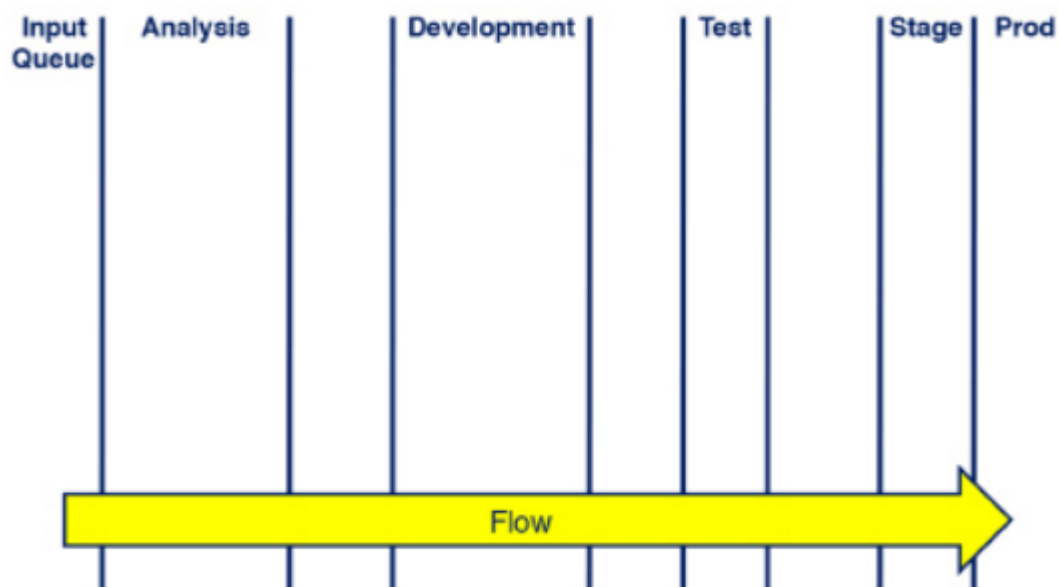


Figure 3.2 Kanban Workflow Visualization

Source : Anderson, 2010

The core principle of Kanban is to visualize the workflow, allowing developers and stakeholders to clearly observe the task progress and identify bottlenecks during the development process. Tasks are typically represented as cards and moved across columns representing work stages. This visual approach increases transparency and

facilitates monitoring of ongoing activities in the development lifecycle (Anderson, 2010).

Another important principle in Kanban is Work-In-Progress (WIP) control, which limits the number of tasks that are being worked on at the same time. By controlling WIP, development activities become more focused and organized, reducing task load and increasing productivity. In fact, Kanban encourages continuous improvement, where workflow efficiency is routinely evaluated and refined based on observations and process performance.

The Kanban board is often used as a complimentary tool to support iterative development methods in the context of Agile software development. While methodologies such as XP or PXP define development practice and iterations cycles, Kanban assists in managing and visualizing the progress of development tasks within each iteration. This combination allows developers to maintain a structured development phase while ensuring clear visibility into task progression.

3.4 Unified Modeling Language (UML)

Unified Modeling Language (UML) is a standard modeling language that is used to visualize, define, build and document the software systems. UML provides a set of graphical notations that help developers represent the structure and behavior of the system clearly and organized. By using UML diagrams, the concept of complex systems can be effectively communicated between developers, analysts, and stakeholders during the software development process (Seidl et al., 2015).

The UML plays an important role in system analysis and design because it allows developers to translate system requirements into structured visual models. These models help to describe how system components interact and how system functionality is organized. The use of UML improves system understanding, reduces ambiguity in system specification, and supports systematic software development.

According to Seidl et al, (2015), UML diagrams can generally be categorized into two main types: structural diagrams and behavioral diagrams. Structural diagrams describe structural aspects of the system, such as system components and the relationship between them, while behavioral diagrams describe dynamic interactions and the functional behavior of the system.

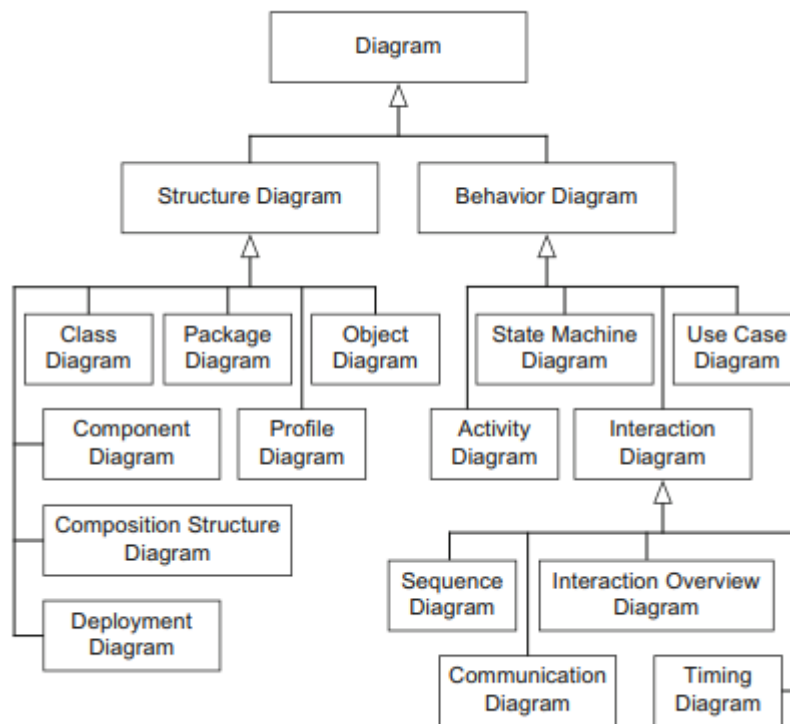


Figure 3.3 Classification of UML Diagrams

Source : Seidl et al, 2015

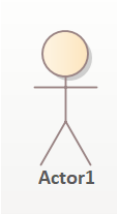
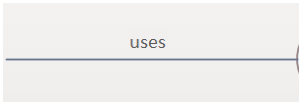
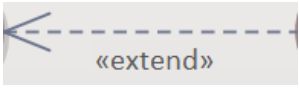
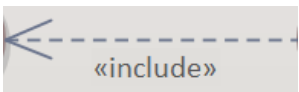
Because of its standardized notation and widespread adoption in software engineering, UML has become one of the most frequently used tools for modeling information systems during the analysis and design phases.

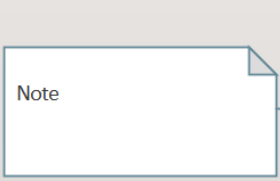
3.4.1 Use Case Diagram

The Use Case Diagram is one of the behavior diagrams in UML that is used to present the functional requirement of a system. This diagram represents the interaction between external entities, known as actors, and the system through a series of functions called the use cases. This diagram provides a high level overview of how users interact with the system and what services are provided by the system (Seidl et al., 2015).

Use Case Diagrams consist of several key elements, including actors, use cases, and system boundaries. Actors represent users or external systems that interact with the system. Use cases describe the functions or services performed by the system. System boundaries define the scope of the system and distinguish internal system processes from external interactions.

Table 3.1 Use Case Element

Symbol	Name	Description
	Use Case	A use case is represented by an ellipsis containing the name of the use case. The name of the use case is usually formed by a noun and a time word, which briefly and precisely describes the manipulated object and the activity performed.
	Actor	A use case is triggered by an actor. The representation corresponds to a stick figure. You can also represent an actor in a rectangle and specify the stereotype 'Actor' above the name of the actor.
	Uses	An actor has a relationship to the use case when the use case triggers it. This relationship is represented by a connecting line between the use case and the actor.
	Extend	If a use case is extended by a second one under a certain condition, this relationship is indicated by connecting the use cases with an arrow that is labelled with the 'extend' stereotype. The arrowhead points to the use case that is being extended.
	Include	If a use case is contained in a second use case, i.e. it is an integral part of it, both use cases are connected with an arrow labelled with the stereotype 'include'. The arrowhead points to the contained use case.
	Generalization	This relationship can be modelled between actors and between use cases and means that a use case or an actor is specialised. The arrowhead points to the actor or use case that is being specialised.

	Notes	Notes are diagram elements that are attached to other modelling elements. They contain information for understanding the model and are connected to the element by a broken connecting line.
---	-------	--

Source : <https://www.sparxsystems.eu/languages/uml/diagrams/usecasediagram/>

The main purpose of a Use Case Diagram is to help developers and stakeholders understand the functionality of the system from the user's perspective. By illustrating the relationship between actors and system functions, this diagram helps clarify system requirements and provides a visual representation of system interactions. Use Case Diagrams are often used in the early stages of system analysis and design because they simplify complex system requirements into an easy-to-understand visual representation. This makes them very useful for communication between system analysts, developers, and stakeholders during the requirements analysis phase.

3.4.2 Database Schema

The relational database schema represents the logical structure of a database that defines how data is organized and stored in a database management system. This schema determines the structure of tables, attributes, relationships, and constraints that govern how data is managed and maintained in the system. This schema serves as a guideline for database design and implementation. It describes the entities in the system, the attributes associated with those entities, and the relationships between them. Through this structured representation, developers can ensure that data is stored consistently and can be accessed efficiently when needed (Elmasri & Navathe, 2016).

Table 3.2 Database Schema Element

Element	Example	Description
Schema	Namespace, database schema	A logical space containing a set of database objects (tables, views, procedures, etc.) in a single database.
Table	Entity, relation	The main structure for storing data in rows and columns.
Column	Field, attribute	Specifies certain data types in a table, such as vesselName, reportContent, jobSection, etc

Row / Record	Tuple	One instance of data in a table
Primary Key (PK)	The main key	A column (or combination of columns) that uniquely identifies each row in the table.
Foreign Key (FK)	Guest key, referential key	A column that references the primary key in another table to form a relationship and maintain referential integrity.
Index	Indeks	Additional structure to accelerate data search and sorting in specific columns.
Constraint	Integrity rules	Rules such as NOT NULL, UNIQUE, CHECK, PRIMARY KEY, FOREIGN KEY that restrict data values.
Relationship	1 to 1, 1 to many, many to many	Relationships between tables, which are usually implemented with PK–FK, describe how data is related.
View	Virtual table	Stored queries that display data from one or more tables as if they were regular tables.
Sequence	Sequence, auto increment	Objects that generate sequential numerical values, often used for PK.
Stored Procedure	Stored procedures	A collection of SQL commands stored on the database server that can be called again.
Function	Stored function	Similar to a procedure, but returns a value and can be used in a query.
Trigger	Trigger	Code that is automatically executed when a specific event occurs on a table (INSERT/UPDATE/DELETE).

Source : <https://dbschema.com/documentation/schema.html>

In relational database systems, schemas typically consist of multiple tables connected through relationships using primary keys and foreign keys. These relationships ensure data integrity and facilitate structured data management across different parts of the system.

Properly designed database schemas are essential for ensuring data consistency, minimizing redundancy, and supporting efficient data operations. Therefore, database schema design is considered an important stage in information system development, especially during the system design phase.

3.5 Data

The definition of data refers to raw facts, observations, or measurements that can be processed into meaningful information. When it comes to information systems, data represents various types of entities, attributes, and records that are collected, stored, and processed to support decision making and operational activities (Stair & Reynolds, 2018).

Data has a crucial role in information system development because it serves as the basis for system functionality and information processing. Effective data management ensures that the information stored in the system stays accurate, consistent, and accessible when needed. Most information systems organize their data in a structured format that allows for efficient data storage and recovery through a database management system.

According to Laudon & Laudon (2020), data in information systems can be categorized based on its structure and use, including transactional data, operational data, and analytical data. The proper organization of data enables the system to support operational processes, generate reports, and provide insights for decision making.

In software development, proper data modeling is necessary to ensure that data can be processed efficiently and integrated with other system components. Therefore data management and organization are important aspects of information system design.

3.6 PostgreSQL

PostgreSQL is an open source relational database management system (RDMS) designed to support reliable data storage, efficient data retrieval, and advanced database operations. PostgreSQL is widely known for its reliability, scalability, and support for complex queries and large datasets (Stonebraker & Kemintz, 1991).

As a relational database management system, PostgreSQL organizes data into structured tables consisting of rows and columns, where relationships between tables can be defined through keys and constraints. This relational model allows data to be stored systematically while maintaining consistency and integrity across different datasets.

PostgreSQL also supports advanced database features such as transactions, indexing, stored procedures, and concurrency control, which ensure reliable data management in multi user environments. These features make PostgreSQL suitable for modern web applications and enterprise systems that require high performance and reliable data processing.

Due to its flexibility and scalability, PostgreSQL has become one of the most widely used open source databases for web based applications and data driven systems.

3.7 Object Relational Mapping (ORM)

Object-Relational Mapping (ORM) is a programming technique used to bridge the gap between object-oriented programming languages and relational database systems. ORM allows developers to interact with databases using programming language objects rather than writing complex SQL queries directly (Fowler, 2003).

In object oriented programming, applications typically represent data as objects, while relational databases store data in tables. ORM frameworks provide mapping mechanisms that connect these two representations, allowing developers to perform database operations such as data retrieval, addition, updating, and deletion using object oriented syntax. The use of ORM simplifies interaction with databases by reducing the need for repetitive SQL queries and improving code maintainability. In addition, ORM frameworks often provide features such as automatic query generation, data validation, and management of relationships between entities.

According to Fowler (2003), ORM improves development productivity by allowing developers to focus on application logic while the framework manages the underlying database interactions.

3.8 Nuxt.js Framework

Nuxt.js is a modern JavaScript framework built on Vue.js and designed to simplify web application development. This framework provides a structured architecture that helps developers build scalable, maintainable, and high-performance web applications. One of the main advantages of Nuxt.js is its ability to support various rendering strategies, including server-side rendering (SSR), static site generation (SSG), and client-side rendering (CSR). This capability allows developers to optimize application performance and improve user experience. (*Nuxt.js Documentation*, 2023).

Nuxt.js also provides several built-in features that simplify web development, including automatic routing, state management integration, modular architecture, and support for modern JavaScript development tools. These features allow developers to develop web applications more efficiently while maintaining code organization and scalability.

CHAPTER 4

METHODOLOGY

The development of the Field Automation System (SAFED) follows a structured and iterative approach designed to bridge the gap between theoretical information system principles and the operational requirements of the maritime manufacturing industry. By adopting the Personal Extreme Programming (PXP) framework, the development process remains responsive to the dynamic requirements of PT Bintang Indayu Gemilang while maintaining technical quality and system reliability.

4.1. Place and Location

The practical work was conducted at PT Bintang Indayu Gemilang, located at Jl. Ruhui Rahayu No. 115, South Balikpapan, Balikpapan City. The internship program was carried out over a period of approximately eight weeks, starting on January 5th, 2026, and ending on February 27th, 2026. During this period, the system development lifecycle was executed, beginning with the observation of operational activities, followed by requirement identification, system development, and finally documentation of the development results. This timeframe allowed it to conduct a complete cycle of system development while adapting the solution to real operational needs within the organization.

4.2. Activity Stages

The practical work activities were organized into several stages in order to ensure a systematic and structured development process. These stages include preliminary orientation and environmental analysis, stakeholder interviews and requirement identification, system requirement analysis, system development, and documentation of the development results. Through these stages, the developed SAFED system was designed to align with both operational requirements and academic software engineering practices.

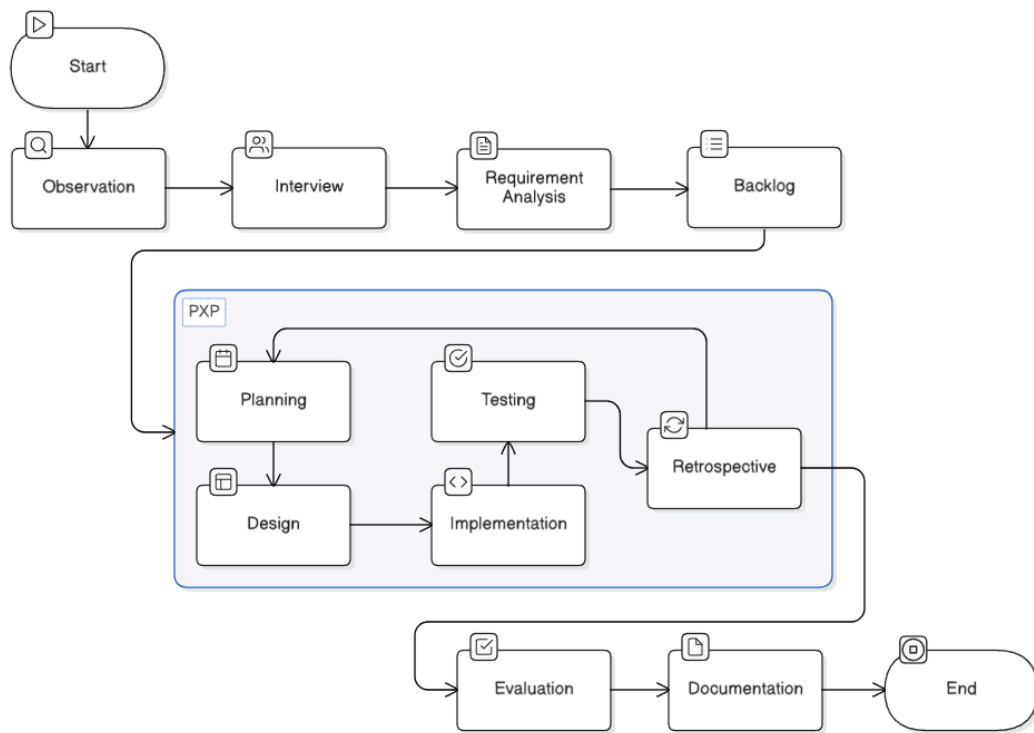


Figure 4.1 Practical Work Activity Flow

Source : Practical Work Documentation, 2026

Figure 4.1 illustrates the sequence of activities carried out during the practical work program. The process begins with observation and stakeholder interviews to identify operational problems and system requirements. The identified requirements are then organized into a product backlog, which becomes the basis for system development using the Personal Extreme Programming (PXP) methodology.

The development phase follows an iterative approach consisting of planning, design, implementation, and testing activities in each iteration, allowing continuous verification of system functionality. After multiple iterations are completed, final system evaluation and validation are conducted to ensure that the overall system meets the defined requirements. The final stage involves documenting the development process and the results obtained during the practical work.

4.2.1 Preliminary Orientation and Environmental Analysis

The initial stage of the practical work involved understanding the organizational environment and operational workflow within the Manufacturing Division. At this stage, observations were conducted to analyze how field operations were planned and

implemented, including equipment preparation procedures, operational scheduling, documentation practices, and communication between operational personnel.

Through direct observation, several inefficiencies were identified in the existing workflow. Operational activities relied heavily on manual documentation and spreadsheet-based records, which created difficulties in tracking operational progress and managing equipment preparation. The absence of a centralized system also resulted in fragmented documentation and reduced visibility of ongoing operational activities.

4.2.2 Stakeholder Interview and Requirement Identification

In order to obtain deeper insight into operational challenges, interviews were conducted with stakeholders responsible for managing field operations. The primary interview session was conducted with the Head of the Manufacturing Division, who oversees operational activities such as installation, commissioning, maintenance, and system updates performed on ships.

The interviews focused on several operational aspects, including equipment preparation procedures, documentation requirements, reporting workflows, user responsibilities, and operational constraints encountered during field activities. The results of these interviews revealed several key operational issues, including missing or damaged checklist documents, delays in compiling operational reports, difficulties in tracking operational progress, and fragmented documentation stored across spreadsheets and messaging platforms.

These findings provided valuable insight into the operational problems faced by the organization and served as the foundation for defining the requirements of the proposed system.

4.2.3 System Requirement Analysis

After completing the data collection phase, the identified operational problems were translated into system requirements. The analysis focused on designing a system capable of supporting the entire lifecycle of field operations within the Manufacturing Division.

The proposed system was expected to support several key operational activities, including operation scheduling, equipment checklist management, monitoring of operational tasks, documentation of field activities, and automated generation of

operational reports. These requirements were formulated in the form of user stories, which describe system functionality from the perspective of the system users.

The collection of user stories was then organized into a product backlog, which served as the primary reference for planning the development iterations of the SAFED system.

4.2.4 System Development and Iterative Testing

Once the system requirements had been defined, the system development phase was carried out using the Personal Extreme Programming (PXP) methodology. The development process followed an iterative approach, allowing system features to be implemented incrementally through multiple development iterations.

Each iteration involved several development activities, including task planning, system design, feature implementation, and system testing. The testing process was performed in every iteration to ensure that each developed feature functioned correctly and met the specified requirements.

This iterative testing approach allowed early detection and correction of errors before proceeding to subsequent iterations. As a result, the system could be continuously refined and improved, ensuring alignment with operational needs identified during the analysis phase.

4.2.5 Documentation and Reporting

The final stage of the practical work involved documenting the development process and the results of the implemented system. This stage included preparing the internship report, which describes the methodology, system design, development process, and evaluation of the proposed solution. The documentation serves as both an academic requirement and a record of the system development carried out during the internship program.

4.3 Data Collection Methods

Several data collection methods were used during the practical work to obtain information related to operational workflows and system requirements within the Manufacturing Division.

4.3.1 Observation

Observations were conducted to understand how operational activities were carried out within the division. Through direct observation, the author analyzed processes such as equipment preparation, operational scheduling, and field activity documentation. This method provided insight into the existing workflow and helped identify inefficiencies related to equipment checklist management and operational documentation.

4.3.2 Interviews

Interviews were conducted with stakeholders involved in operational activities, particularly the Head of the Manufacturing Division. The purpose of these interviews was to obtain detailed information about operational procedures, documentation requirements, and challenges encountered during field operations.

The interviews also provided valuable insights into the difficulties faced by operational teams, including delayed reporting processes, incomplete documentation, and the lack of a centralized monitoring system for tracking operational progress.

4.3.3 Documentation Study

A documentation study was conducted by reviewing existing operational records such as equipment checklists, operational reports, and other related documents. This analysis helped identify how operational data was previously recorded and managed within the organization.

4.3.4 Literature Study

To support the theoretical foundation of the system development process, a literature study was conducted by reviewing academic sources such as books, journal articles, and technical documentation. These sources provided insights into concepts related to Agile development methodologies, Personal Extreme Programming, system modeling techniques, and workflow management systems.

4.4 System Development Method

The development of the SAFED system was conducted using the Personal Extreme Programming (PXP) methodology, which is an Agile-based software development approach designed for independent developers. This methodology was selected because the system development was carried out by a single developer while still requiring

structured development practices such as iterative development, continuous testing, and periodic evaluation.

Through the PXP approach, the system was developed incrementally across several iterations. Each iteration focused on implementing a set of prioritized features derived from the product backlog.

4.4.1 Personal Extreme Programming (PXP)

The development process of the SAFED system follows the phases defined in the Personal Extreme Programming methodology proposed by Dzhurov et al. (2009). These phases guide the development process from requirement identification to system evaluation through iterative cycles. The overall structure of the PXP development process is illustrated in Figure 3.1 in Chapter 3.

The development phases consist of the following stages:

1. Requirement Identification

This phase involves identifying system requirements based on the problems observed during operational activities and information obtained from stakeholder interviews.

2. Planning

During the planning phase, identified requirements were structured into development tasks and organized into a product backlog. Each task was assigned an estimated development effort using story points.

3. Iteration Initialization

At the beginning of each iteration, development tasks were selected from the product backlog according to their priority and estimated effort. These tasks were then scheduled to be completed within the current iteration cycle.

4. Design

The design phase involves modeling the system structure and defining database schemas and system interactions. System modeling techniques such as Use Case Diagrams and database schema design were used to define the functional structure of the system.

5. Implementation

In this phase, system functionalities were implemented through coding activities.

The system was developed using the Nuxt.js framework for the application interface and PostgreSQL as the database management system.

6. System Testing

After implementation, system testing was conducted to verify that each feature operates according to the defined requirements. Testing activities ensured that the developed functionalities supported operational workflows effectively.

7. Retrospective Evaluation

At the end of each iteration, an evaluation was conducted to analyze the development process and identify potential improvements. The evaluation results were used to refine system features and update development tasks for the next iteration cycle.

4.4.2 Kanban Implementation

To support task organization and development workflow management, the Kanban method was used during the system development process.

The Kanban board was implemented using the ClickUp project management platform, which provides a digital workspace for organizing and tracking development tasks. The workflow was structured into several stages, consisting of To Do, In Progress, System, Testing, Review, and Done.

Each development task was represented as a card that moved across these columns as the development progressed. This visual workflow allowed the author to monitor task progress, manage iteration workloads, and maintain a structured development process.

4.5 System Design

Before entering the implementation phase, the system was designed using system modeling techniques in order to define system functionality and data structure. The design stage included the creation of Use Case Diagrams to represent system functionality and Database Schema models to define the structure of the system's data storage.

4.5.1 Use Case Diagram

The Use Case Diagram was developed to illustrate the interaction between system users and the SAFED system. The diagram identifies the actors involved in the system and the main functionalities available to each actor.

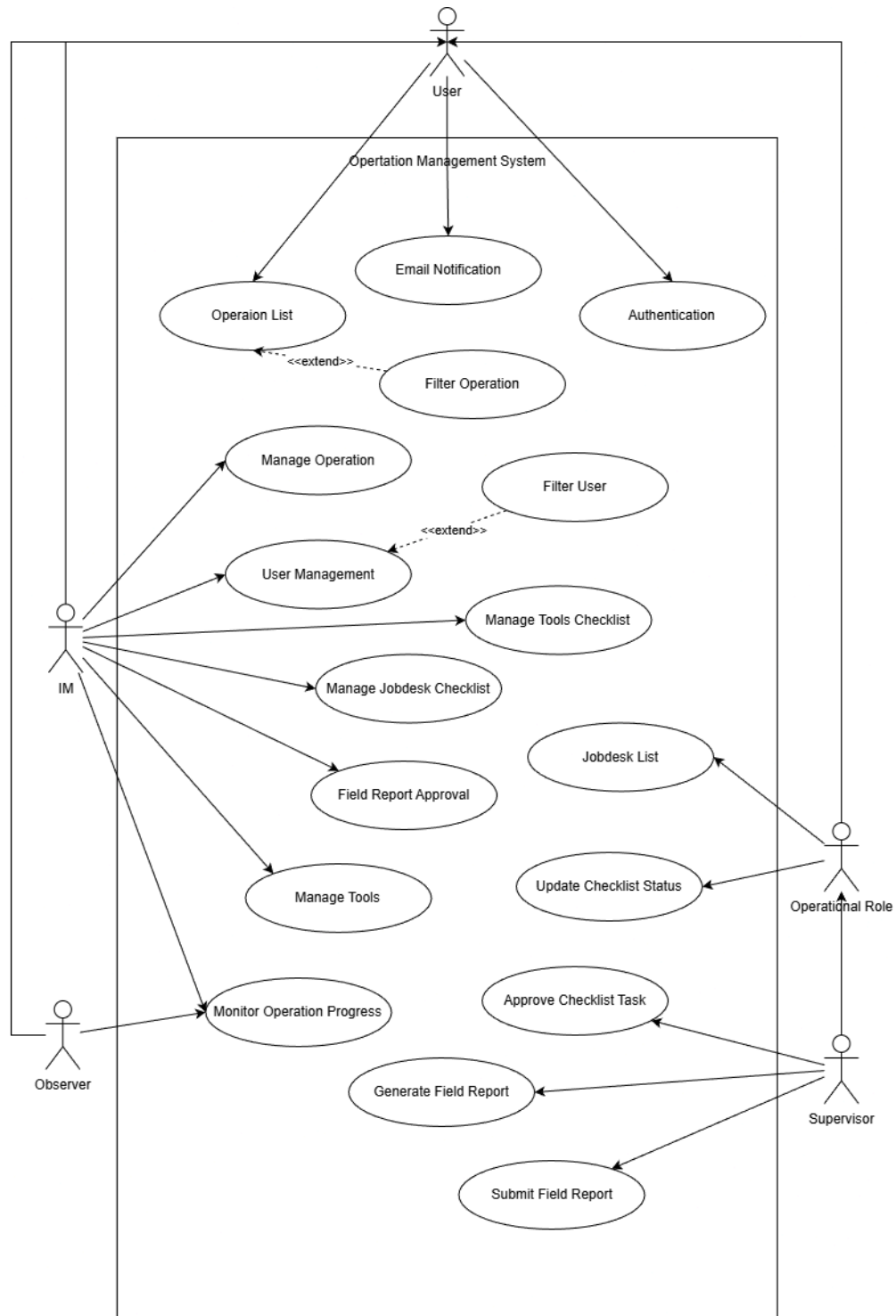


Figure 4.2 SAFED System Use Case Diagram

Source : Practical Work Documentation, 2026

The SAFED system involves four primary actors: Implementation Manager, Supervisor, Staff, and Observer. Each actor interacts with the system according to their respective responsibilities in the operational workflow.

The description of each use case implemented in the system is presented in the following table.

Table 4.1 Use Case Description

No	Use Case	Description	Actor
UC-01	User Management	IM can view a list of user data with all attributes belonging to each user, and can also create, delete, and change user data details and user status.	IM
UC-02	Filter User	On the User Management page, IM can filter user data according to the role and status of the user when selecting the Sort By option, and can search based on the name and email address of a specific user when entering the search form.	IM
UC-03	Authentication	Users can log in to the system with authentication based on their respective data, and can log out of the system.	IM, OBSERVER, STAFF
UC-04	Operation List	Users can view a list of all operations both those that are running and those that have been completed.	IM, OBSERVER, STAFF
UC-05	Filter Operation	On the Operation List page, users can filter operation data by date or type of operation, and can search for specific operations by entering the search form.	IM, OBSERVER, STAFF
UC-06	Manage Operation	IM can create new operational data by entering details of the activities to be carried out. The system also allows users to change the data if there are changes to the plan. Operational data that is still in the planning stage can be saved as a draft to be adjusted later. In addition, IM can delete data if the	IM

		operational plan is no longer relevant, and IM can also assign staff to be in charge of the operation.	
UC-07	Manage Tools Checklist	On the page section set checklist tools, IM can add tools as needed for specific operations, and can also determine how many tools need to be brought on each field operation.	IM
UC-08	Manage Jobdesk Checklist	On the IM operation checklist page, you can create sections to add job descriptions that correspond to those sections. Within those sections, you can also determine which jobs require photo documentation and which jobs do not.	IM
UC-09	Jobdesk List	After accessing the desired operation, SUPERVISORS and STAFF can view all task data to be performed in the field operation.	SUPERVISOR, STAFF
UC-10	Update Checklist Status	On the SUPERVISOR and STAFF task checklist page, they can update the relevant task status, as well as add notes and documentation for tasks that require documentation.	SUPERVISOR, STAFF
UC-11	Approve Checklist Task	The SUPERVISOR can check all task checklists that have been completed by staff and give approval to verify that all field work has been completed and meets standards.	SUPERVISOR
UC-12	Generate Field Report	SUPERVISORS can create draft reports based on the progress data that has been entered. The system will integrate checklist data, documentation photos, and notes into a standard report format.	SUPERVISOR
UC-13	Submit Field Report	The SUPERVISOR can send the completed incident report to IM for review and validation.	SUPERVISOR
UC-14	Monitor Operation Progress	IM and OBSERVER can monitor the progress of the operation through the progress bar for each operation.	IM, OBSERVER,
UC-15	<i>Manage Tools</i>	On the Manage Tools page, the IM can view	IM

		a list of registered tools, create new tool data in the tool data list, change and delete tool data to keep the stored tools relevant, and search for specific tools by entering them in the search form.	
--	--	---	--

Source : Practical Work Report Documentation, 2026

4.5.2 Database Schema

The SAFED system uses a relational database structure implemented using PostgreSQL. The database schema defines the entities, attributes, and relationships required to store operational data such as users, operations, tools, checklists, and field reports.

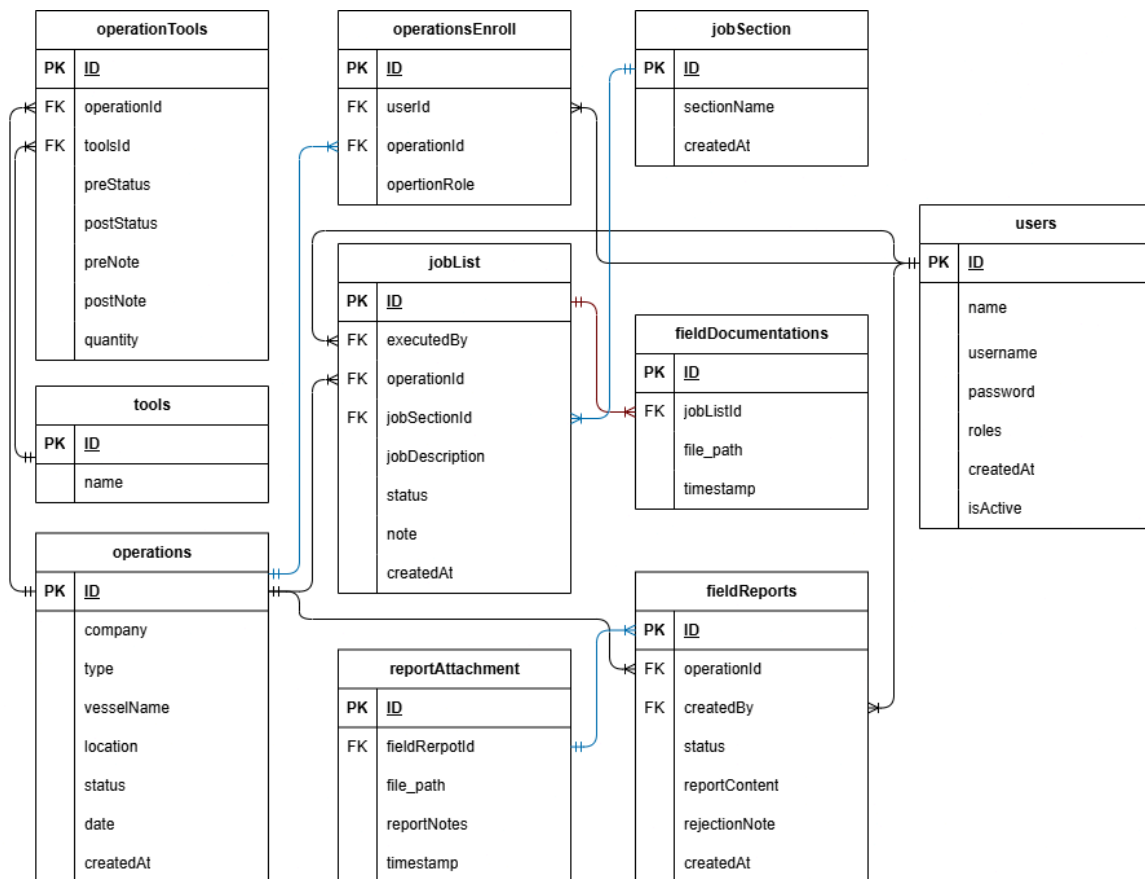


Figure 4.3 SAFED Database Schema

Source : Practical Work Documentation, 2026

The database design ensures data consistency and supports efficient storage and retrieval of operational information within the system.

4.6 Product Backlog and Iteration Planning

The development of the SAFED system was organized using a product backlog, which contains a list of system features derived from the identified system requirements. Each backlog item was written in the form of a user story, describing the functionality required by specific system users.

Each backlog item was assigned a story point value to estimate the development effort required for its implementation. The backlog items were prioritized based on their importance to the operational workflow of the Manufacturing Division.

The backlog items were then implemented gradually through three development iterations during the system implementation phase.

Table 4.2 Product Backlog

ID	Actor	User Story	Task	SP	Priority
PB-01	Implementation Manager	As an Implementation Manager, I want to create a new user accounts, So that i can grant access to new team members	Creating a users account	3	High
PB-02	Implementation Manager	As an Implementation Manager, I want to edit the detail of user information, So that their information remains accurate	Edit users account	2	Medium
PB-03	Implementation Manager	As an Implementation Manager, I want to delete/deactive users account, So that i can revoke access for departed team members	Delete/deactive users account	2	Medium
PB-05	Implementation Manager	As an Implementation Manager, I want to view a list of all users , So that i can easily manage them	View/read list of all users	1	High
PB-06	Users	As a User, I want to log into the system, so that I can access features according to my role	Authentication	5	High
PB-07	User	As a User, I want to see all the lists of the operations, so that I can find out which operations will be carried out and which have been completed.	View operation list	2	High

PB-08	Implementation Manager	As an Implementation Manager, I want to make the operation schedule, So that field activities can be executed	Creating operation schedule	5	High
PB-09	Implementation Manager	As an Implementation Manager, I want to assign staff to an operation, so that each users has their own job	Assign operation roles	3	High
PB-10	Users	As a User, I want to receive an automated email notification whenever a new operation is created, so that I can get information about the operation that will be performed	Automated email notifications for new operations	3	Medium
PB-11	Implementation Manager	As an Implementation Manager, I want to edit the detail of operation schedule, so that the information about the operation schedule remains accurate	Edit operation schedule	2	Medium
PB-12	Implementation Manager	As an Implementation Manager, I want to delete the operation schedule, so that I can delete irrelevant operation schedule data	Delete operation schedule	2	Medium
PB-13	Implementation Manager	As an Implementation Manager, I want to configure tools checklists for each operation, so that the types of tools used in the operation are appropriate	Manage checklist tools	3	High
PB-14	Implementation Manager	As an Implementation Manager, I want to customize the job list for every operation, so that only relevant tasks are executed	Manage checklist job desk	3	High
PB-15	Implementation Manager	As an Implementation Manager, I want to create field report templates, So that staff have standard format to make report	Create field report template	3	High
PB-16	Operational Role	As a member of Operational Role, I want to see the entire list of job desks so that I can perform my tasks in line with the ongoing operations.	View job desk list	2	High
PB-17	Operational Role	As a member of Operational Role, I want to update the checklist status, so that the progress is recorded	Update checklist status	5	High
PB-18	Supervisor	As a Supervisor, I want to finish the operation, so that i can verify all the task are completed	Finish Operation	3	Medium

PB-19	Supervisor	As a Supervisor, I want to generate a field report draft, So that I can start reporting field operation	Generate field report draft	5	High
PB-20	Supervisor	As a Supervisor, I want to create a completed field report, so that it can be reviewed and validated.	Create Field report	2	High
PB-21	Implementation Manager	As Implementation Manager, I would like to see the created field report, so that I can download and review the report.	Download field report approval	2	Medium
PB-22	Observer	As an Observer, i want to view operation progress, so that I can monitor overall operation progress	View operation progress	3	High
PB-23	Implementation Manager	As an Implementation Manager, I want to monitor job progress, so that I can ensure work completion	Monitor job progress	3	High
PB-24	Implementation Manager	As an Implementation Manager, I want to register new tools to the database, so that I can manage all the tools we have.	Create tools	2	High
PB-25	Implementation Manager	As an Implementation Manager, I want to edit/change the name of the stored tools, so that the tools remain accurate.	Edit tools	2	Medium
PB-26	Implementation Manager	As an Implementation Manager, I want to remove specific tools from the list, so that the list remains accurate and free of irrelevant tools.	Delete tools	1	Medium
PB-27	Implementation Manager	As an Implementation Manager, I want to view a list of all the tools , So that i can easily manage them	View tools list	1	High
PB-28	Supervisor	As a Supervisor, I want to edit the generated field report, so that i can maintain the document keep align	Edit Field Report	2	Medium

Source : Practical Work Report Documentation, 2026

During the development process, the product backlog evolved based on feedback obtained through daily stand-up discussions and weekly coordination meetings with stakeholders. Several additional features and improvements were identified during early development iterations and were incorporated into subsequent development cycles.

4.7 Analysis and Evaluation Techniques

The analysis technique used in this practical work is primarily qualitative analysis, focusing on identifying operational problems and evaluating how the developed system addresses these issues.

Data collected through observations, interviews, and documentation studies were analyzed to identify operational inefficiencies related to equipment preparation, documentation management, and monitoring of operational activities. The evaluation of the developed system was conducted through functional testing, where each system feature was tested to verify that it operates according to the defined requirements.

This testing process ensured that the developed SAFED system is capable of supporting operational activities and improving workflow management for field operations within the Manufacturing Division.

CHAPTER 5

RESULTS AND DISCUSSION

This chapter presents the results obtained from the development of the system during the practical work program at PT Bintang Indayu Gemilang. The results include the implementation of system features, system design outputs, and the analysis of how the developed system supports operational activities within the Manufacturing Division. Furthermore, this chapter discusses the challenges encountered during development, the solutions implemented, and the overall impact of the developed system.

5.1 System Development Results

The development of the field operations management system was carried out using the Personal Extreme Programming (PXP) approach. The development process followed an iterative workflow where system features were gradually implemented through several development cycles. This approach allowed the system to evolve progressively while adapting to requirement adjustments identified during development.

The system was implemented as a web-based application designed to support operational activities within the Manufacturing Division. The application provides functionality for managing operational schedules, equipment preparation, field task execution, documentation, and automated report generation.

Technically, the system was developed using Nuxt.js as the web application framework, Drizzle ORM as the data access layer, and PostgreSQL as the relational database management system. This technology stack enables efficient data management and structured system architecture suitable for handling operational data within the organization.

The development process was organized into three iterations, where each iteration focused on implementing specific system features derived from the product backlog.

5.1.1 System Overview

The developed system functions as a centralized digital platform designed to support field operation management activities. The system facilitates several operational processes, including operation scheduling, equipment preparation, activity documentation, and operational report generation.

Through this system, operational data can be recorded, stored, and monitored in a structured and centralized manner. This approach helps ensure that operational activities are properly documented and easily accessible for monitoring and evaluation purposes. The system also enables different users to interact with the application according to their respective responsibilities within the operational workflow.

To support these activities, the system defines several user roles that represent different responsibilities in the operational process. Each role has specific permissions and access levels within the system. The defined user roles are presented in Table 5.1.

Table 5.1 User Roles

Role	Description
Implementation Manager (IM)	responsible for managing operational schedules, configuring system settings, and monitoring operation progress.
Observer	responsible for monitoring operational progress without modifying system data.
Staff	responsible for executing operational tasks and updating checklist progress during field activities.
Supervisor	responsible for supervising field activities and verifying operational reports.

Source : Practical Work Report Documentation, 2026

The operational workflow of the system begins when the Implementation Manager creates an operational schedule and assigns staff members to a specific operation. After the schedule is created, the assigned Staff perform the field activities according to the predefined job desk checklist while documenting operational progress within the system.

During the execution of the operation, the Supervisor monitors the activities and ensures that the operational procedures are properly followed. After the operation is completed, the Supervisor generates and submits the operational report through the system for documentation

and review. Meanwhile, the Observer can monitor operational progress without having the ability to modify the system data.

To support the implementation of these operational processes, the system architecture is designed using a layered structure that separates user interaction, application logic, and data management components. The overall system architecture is illustrated in Figure 5.1

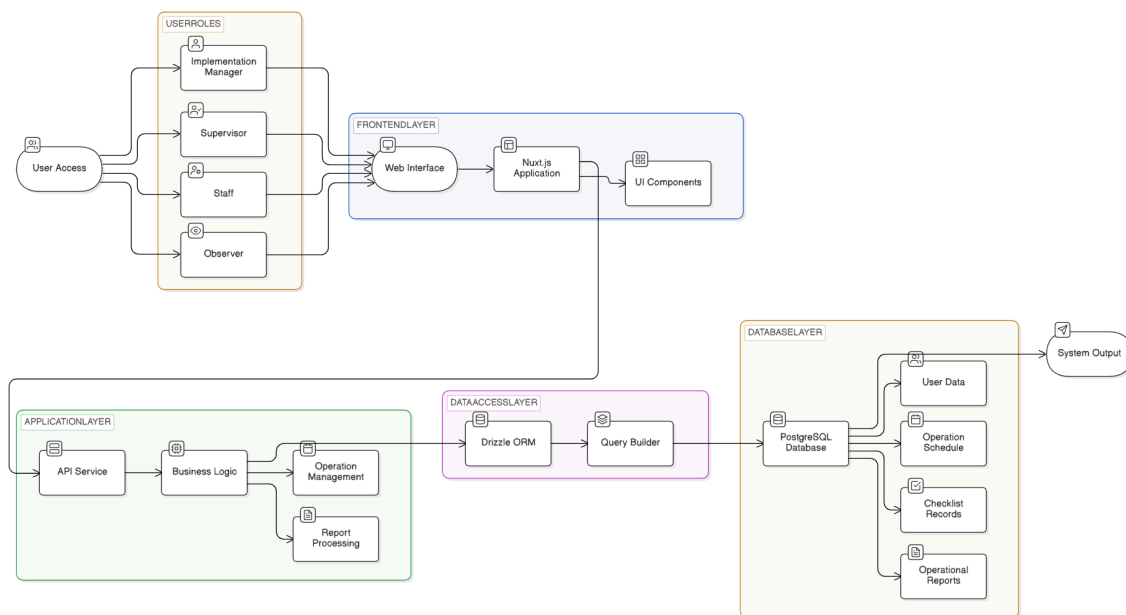


Figure 5.1: System Architecture Overview

Source : Practical Work Documentation, 2026

The system architecture presented in Figure 5.1 illustrates the overall structure of the field operation management system. The architecture adopts a layered design consisting of five main layers: User Roles Layer, Frontend Layer, Application Layer, Data Access Layer, and Database Layer.

- The User Roles Layer represents the actors who interact with the system, namely the Implementation Manager, Supervisor, Staff, and Observer. Each user accesses the system through a web-based interface according to their responsibilities and permissions within the operational workflow.
- The Frontend Layer serves as the user interface of the system and is implemented using the Nuxt.js framework. This layer provides the web interface and UI components that allow users to interact with the system through a browser. All user actions such as managing schedules, updating checklists, and generating reports are initiated from this layer.

- The Application Layer handles the core processing of the system. This layer consists of several components including the API Service, Business Logic, Operation Management, and Report Processing modules. The API service manages communication between the frontend and backend, while the business logic processes operational data and manages system functionality related to operations and reporting.
- The Data Access Layer functions as an intermediary between the application logic and the database. In this system, Drizzle ORM is used to manage database interactions through a structured query builder. This approach simplifies database operations while maintaining consistency and reducing direct query complexity within the application.
- The Database Layer uses PostgreSQL as the primary data storage system. This layer stores various operational data including user data, operation schedules, checklist records, and operational reports. The stored data is retrieved and processed by the system before being delivered back to the users through the frontend interface.

Overall, this layered architecture ensures a clear separation between the user interface, application processing, and data storage components, which improves the maintainability, scalability, and overall performance of the system.

5.1.2 Iteration Development Process

The system development process was conducted through three development iterations based on the Personal Extreme Programming (PXP) methodology. Each iteration consisted of several development activities, including planning, feature implementation, system testing, and retrospective evaluation. This iterative approach allowed the system to be developed incrementally while enabling improvements and adjustments to be made throughout each development cycle.

To facilitate the management and monitoring of development activities, a Kanban board was implemented using the ClickUp project management platform. The Kanban board served as a visual tool to organize and track the progress of development tasks throughout the project.

Development tasks were categorized into several workflow stages, namely To Do, In Progress, System Testing, Review, and Done. Each stage represented the status of a task within the development process, allowing tasks to move progressively from initial planning to final completion.

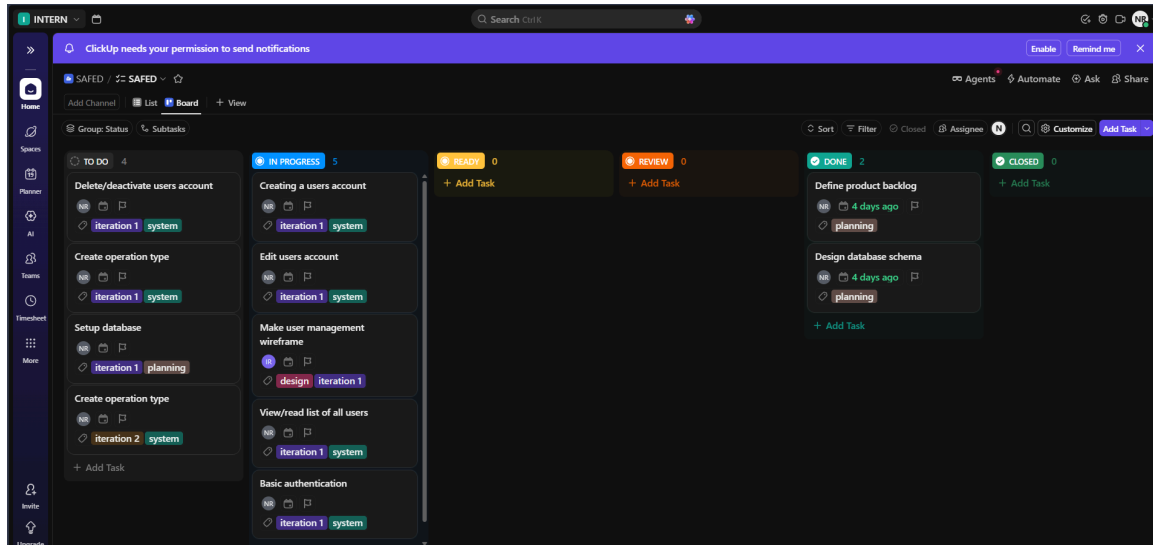


Figure 5.2: Kanban Board Implementation

Source : Practical Work Documentation, 2026

Figure 5.2 illustrates the implementation of the Kanban board used during the system development process. The board provided a visual overview of task distribution and progress, enabling better task management and improving the efficiency of the development workflow.

Through this workflow structure, development tasks could be monitored more effectively while ensuring that each task passed through a structured development process before being completed.

5.1.2.1 Iteration 1: Core System Initialization

The first iteration focused on establishing the foundational components of the system. The primary objective of this iteration was to develop the basic system architecture and implement the core functionalities required for user access management.

The features implemented during this iteration included user account creation, user account editing, user activation and deactivation, user data viewing, and the implementation of the user authentication mechanism. These functionalities were essential to ensure that system access could be properly managed according to the roles assigned to each user within the operational workflow.

The development tasks planned for Iteration 1 are presented in Table 5.2

Table 5.2 Iteration 1 Development Tasks

No	Task	Start Date	Finish Date	SP
1	Create user account	20/1/2026	23/1/2026	3
2	Edit user account	20/1/2026	23/1/2026	2
3	Active and Deactive user account	20/1/2026	24/1/2026	2
4	View users	20/1/2026	23/1/2026	1
5	User Authentication	25/1/2026	29/1/2026	5
Velocity		9 Days		13

Source : *Practical Work Report Documentation*, 2026

Based on the iteration plan, the development process for Iteration 1 was completed within 9 days, with a total of 13 story points successfully implemented. All planned tasks were completed according to the development schedule.

The successful completion of this iteration established the initial system structure required for further feature development in the subsequent iterations, particularly for managing operational data and supporting role-based system access.

5.1.2.2 Iteration 2: Operation Management Features

The second iteration focused on implementing system functionalities related to operational data management and operation scheduling. This iteration aimed to develop system features that support the preparation of operational activities, including equipment management, operation scheduling, and checklist configuration.

The features implemented during this iteration included tools management, operation schedule viewing, operation scheduling creation, role assignment for operational activities, and checklist configuration for tools and job desks. These functionalities were designed to support the preparation phase of field operations and ensure that the required equipment and tasks are properly organized before operational activities are carried out.

The development tasks planned for Iteration 2 are presented in Table 5.3

Table 5.3 Iteration 2 Development Tasks

No	Task	Start Date	Finish Date	SP
1	Create tools	2/2/2026	4/2/2026	2
2	Edit tools	2/2/2026	4/2/2026	1
3	Delete tools	2/2/2026	4/2/2026	1
4	View tools list	2/2/2026	3/2/2026	1
5	View operation list	4/2/2026	8/2/2026	2
6	Creating operation schedule	7/2/2026	-	5
7	Assign operation roles	7/2/2026	-	3
8	Edit operation schedule	7/2/2026	-	2
9	Delete operation schedule	7/2/2026	-	2
10	Manage checklist tools	10/2/2026	-	3
11	Manage checklist jobdesk	10/2/2026	-	3
12	Create field report template	12/2/2026	-	3
13	View job desk list	11/2/2026	-	2
Velocity		14 Days		25
Finish		7 Story Point		
Delayed		18 Story Point		

Source : Practical Work Report Documentation, 2026

Iteration Duration : 14 Days

Total Story Points : 25

Completed : 7 Story Points

Carried Over : 18 Story Points

During this iteration, several tasks were successfully implemented, including tools management and operational data visualization features. However, several features related to operation scheduling and checklist configuration experienced development delays due to external constraints encountered during the development process.

As a result, some tasks were only partially completed and were subsequently carried over to Iteration 3 for further refinement and completion. This adjustment reflects the iterative nature of the PXP development process, where unfinished tasks can be re-prioritized and continued in subsequent development cycles.

5.1.2.3 Iteration 3: Operational Execution and Reporting Features

The third iteration focused on completing the remaining tasks from the previous iteration while implementing additional features required for operational execution and reporting. This iteration aimed to finalize the core functionalities of the SAFED system and ensure that the system could fully support the operational workflow of the Manufacturing Division.

The features implemented in this iteration included completing operation scheduling functionalities, checklist management, operational progress tracking, and field report generation. Additional functionalities such as operation completion, report submission, report approval, and email notifications were also implemented to enhance operational transparency and communication between users.

The development tasks planned for Iteration 3 are presented in **Table 5.4**.

Table 5.4 Iteration 3 Development Tasks

No	Task	Start Date	Finish Date	SP
1	Creating operation schedule	17/2/2026	20/2/2026	5
2	Assign operation roles	17/2/2026	20/2/2026	3
3	Edit operation schedule	17/2/2026	20/2/2026	2
4	Delete operation schedule	17/2/2026	20/2/2026	2
5	Manage checklist tools	17/2/2026	20/2/2026	3
6	Manage checklist jobdesk	17/2/2026	20/2/2026	3
7	Create field report template	17/2/2026	21/2/2026	3
8	View job desk list	17/2/2026	20/2/2026	2
9	Update checklist status	17/2/2026	20/2/2026	5

10	Finish Operation	22/2/2026	25/2/2026	3
11	Generate field report draft	22/2/2026	25/2/2026	5
12	Field report submission	22/2/2026	25/2/2026	2
13	Final field report approval	22/2/2026	25/2/2026	2
14	View operation progress	22/2/2026	25/2/2026	3
15	Monitor job progress	22/2/2026	25/2/2026	3
16	Email Notification	24/2/2026	27/2/2026	3
Velocity		10 Days		44

Source : *Practical Work Report Documentation, 2026*

Iteration Duration : 10 Days

Story Points Completed in Iteration 3 : 26

Carried Over from Iteration 2 : 18

Total Story Points Delivered : 44

This iteration also included the continuation of several tasks that were partially implemented during Iteration 2. Through this process, the remaining system components were successfully refined and integrated into the overall system workflow.

At the end of this iteration, the SAFED system successfully supported the complete operational lifecycle, including operation scheduling, equipment checklist management, task progress monitoring, and automated operational report generation.

5.2 System Feature Implementation

This section presents the implementation of the main features of the SAFED system. Each feature was developed based on the operational requirements identified during the analysis phase and implemented throughout the iterative development process. The implemented features support various operational activities, including user management, operation scheduling, checklist management, operational monitoring, and report generation.

The following subsections describe the main system features along with their interface implementation.

5.2.1 User Authentication Interface

The user authentication feature allows registered users to securely access the SAFED system according to their assigned roles. This feature ensures that only authorized users can access the system and perform specific operational tasks.

Users are required to enter their registered credentials, including their username and password, through the login interface. The system then verifies the credentials and grants access to the application dashboard based on the user's role within the operational workflow.

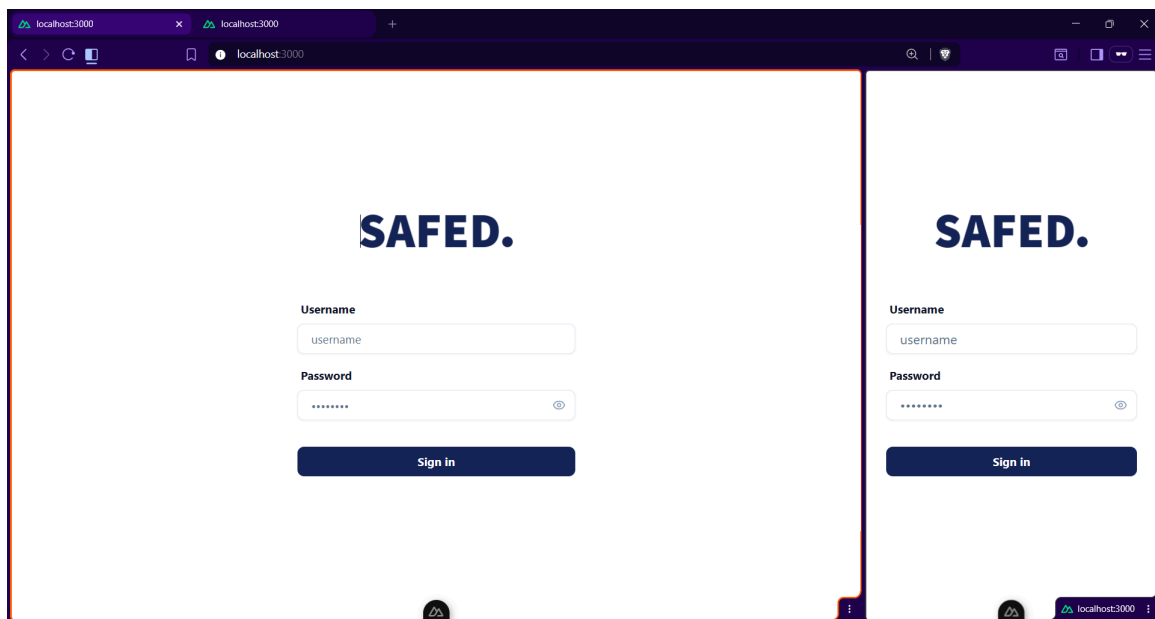


Figure 5.3 Login Interface

Source : Practical Work Documentation, 2026

Users are required to enter their registered credentials, including their username and password, through the login interface. The system then verifies the credentials and grants access to the application dashboard based on the user's role within the operational workflow.

5.2.2 User Management Module

The user management module enables the Implementation Manager to manage system user accounts within the SAFED system. This module provides administrative functionalities for maintaining user data and controlling system access permissions.

Through this module, the Implementation Manager can view the list of registered users, create new user accounts, update user information, and activate or deactivate user access when necessary.

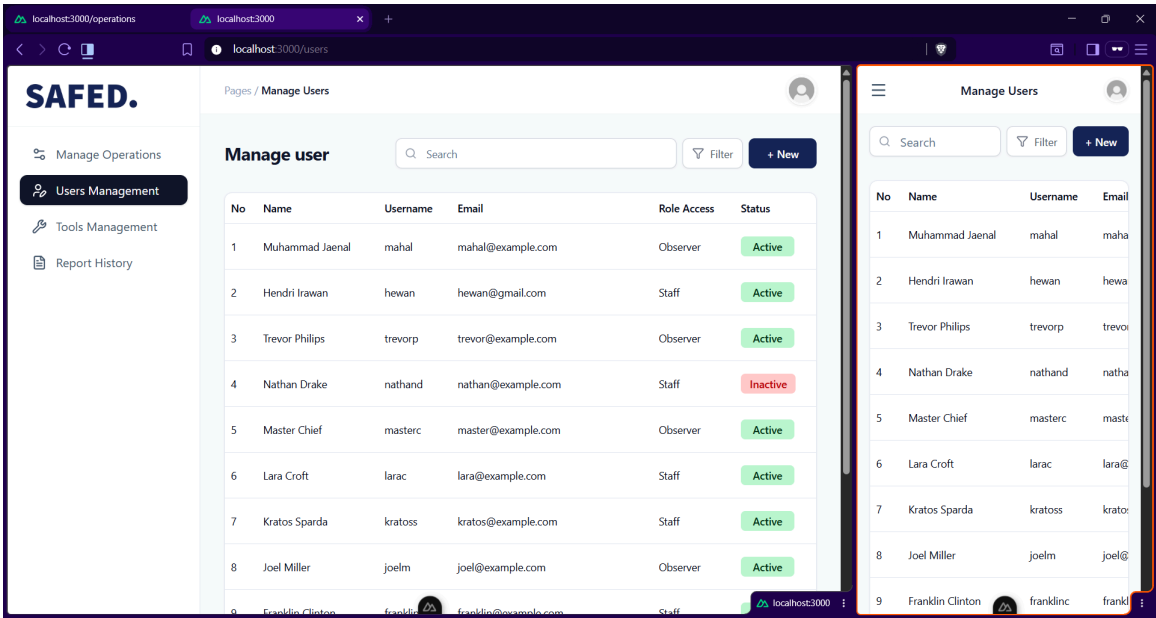


Figure 5.4 User Management Interface

Source : Practical Work Documentation, 2026

The user management interface displays a table containing a list of users along with relevant information such as name, email, role, and account status. This interface allows the Implementation Manager to monitor all registered users and perform account management operations through the available action controls.

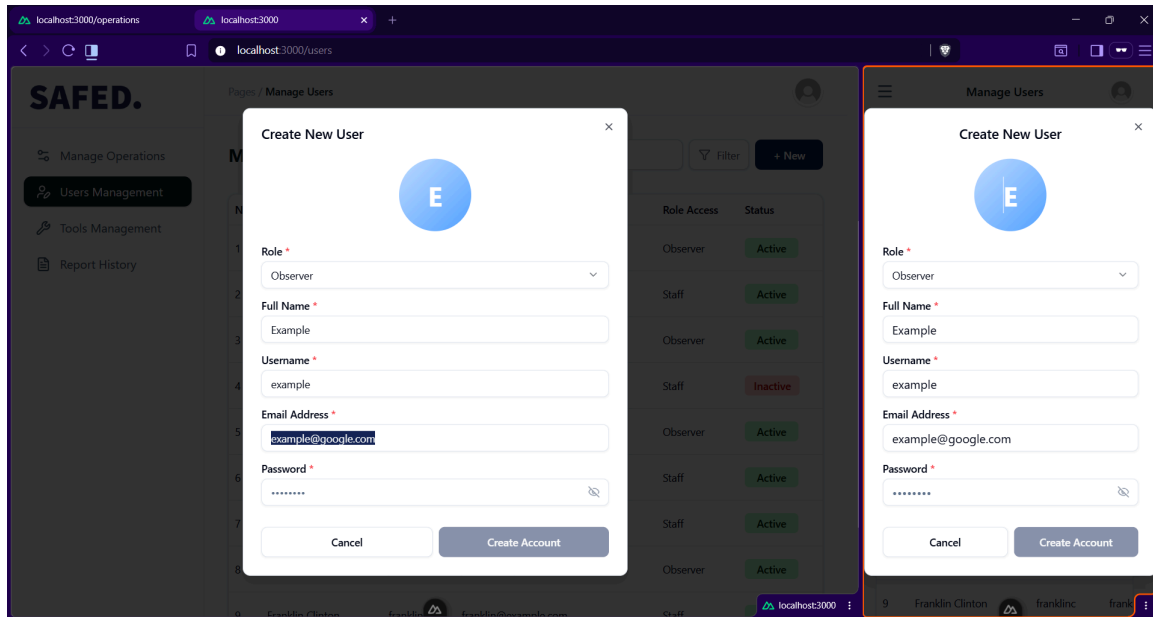


Figure 5.5 User Creation Form

Source : Practical Work Documentation, 2026

The user creation form allows administrators to register new users by entering relevant information such as the user's name, email address, role, and login credentials. Once the data is submitted, the system stores the information in the database and generates a new user account that can be used to access the system.

5.2.3 Tools Management Module

The tools management module enables the Implementation Manager to maintain a centralized list of operational equipment used during field activities. This feature helps ensure that all required tools are properly recorded and managed within the system.

Through this module, the Implementation Manager can create new tool entries, update existing tool information, and remove tools that are no longer relevant to operational activities.

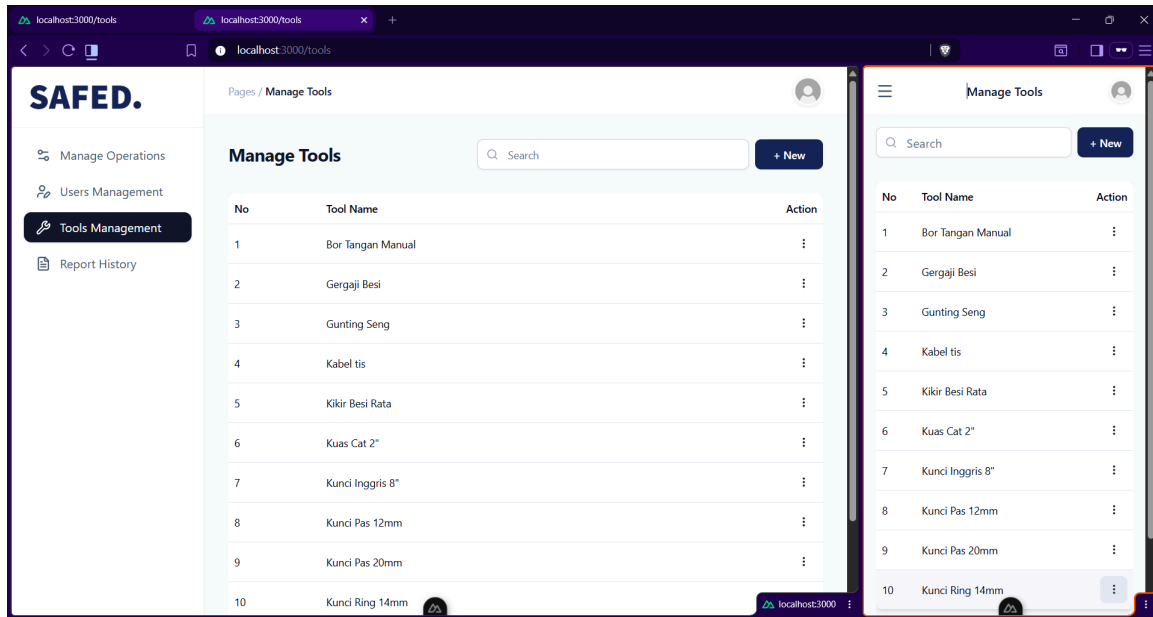


Figure 5.6 Tools Management Interface

Source : Practical Work Documentation, 2026

The tools management interface displays a list of tools stored in the system database. Each entry represents a piece of equipment that may be required during field operations. The interface provides action buttons that allow administrators to add new tools, modify tool information, or remove tools from the system.

5.2.4 Operation Management Module

The operation management module allows the Implementation Manager to organize and manage operational schedules within the SAFED system. This module plays an important role in coordinating field activities by defining operational plans and assigning staff members to specific operations.

Through this module, the Implementation Manager can create operation schedules, update operation information, assign operational roles, and manage operation status.

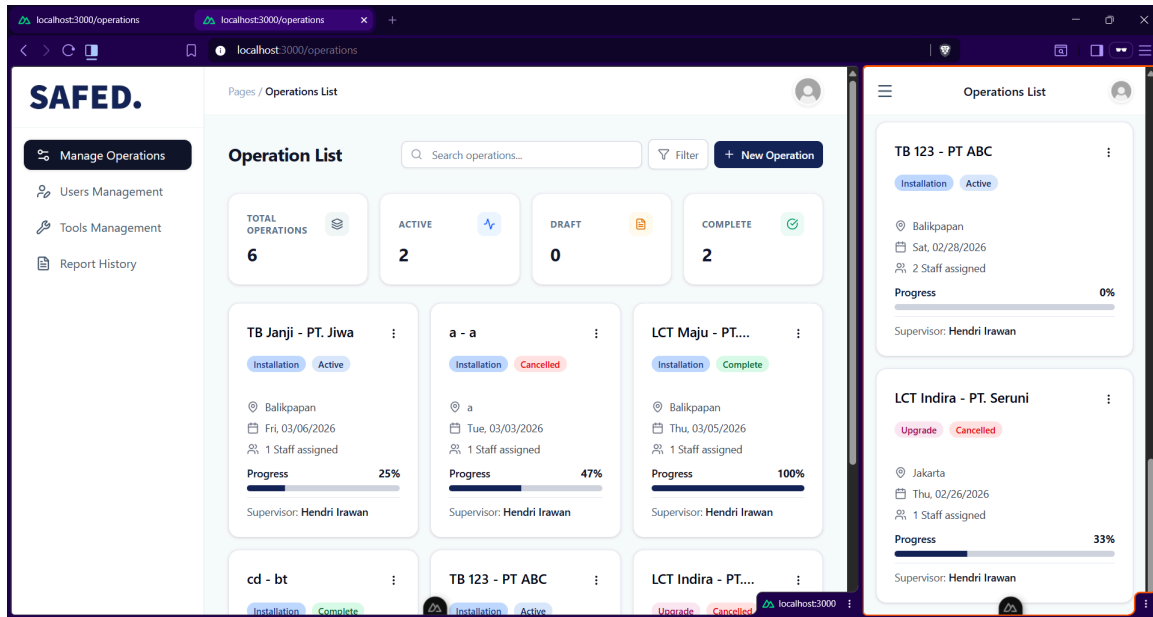


Figure 5.7 Operation List Interface

Source : Practical Work Documentation, 2026

The operation management interface displays a list of scheduled operations along with relevant information such as operation name, scheduled date, assigned personnel, and operation status. From this interface, the Implementation Manager can manage operational data and monitor planned activities.

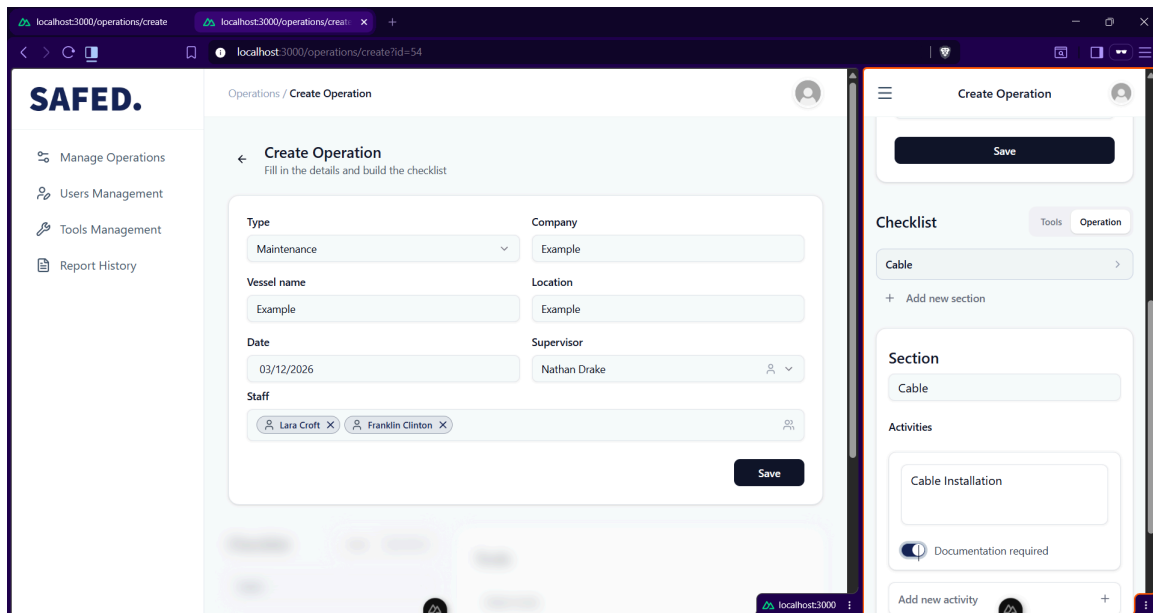


Figure 5.8 Operation Creation Form

Source : Practical Work Documentation, 2026

The operation creation form allows administrators to define the details of a new operation. This includes entering the operation name, scheduling the operation date, assigning responsible staff members, and specifying operational information required for field activities. This functionality ensures that operational planning is organized and properly documented within the system.

5.2.5 Checklist Execution Module

The checklist execution module supports the execution of operational tasks during field activities. This feature allows operational staff and supervisors to perform and record task completion directly through the system.

The checklist is generated based on predefined job desk configurations and equipment preparation requirements defined during the operation planning stage.

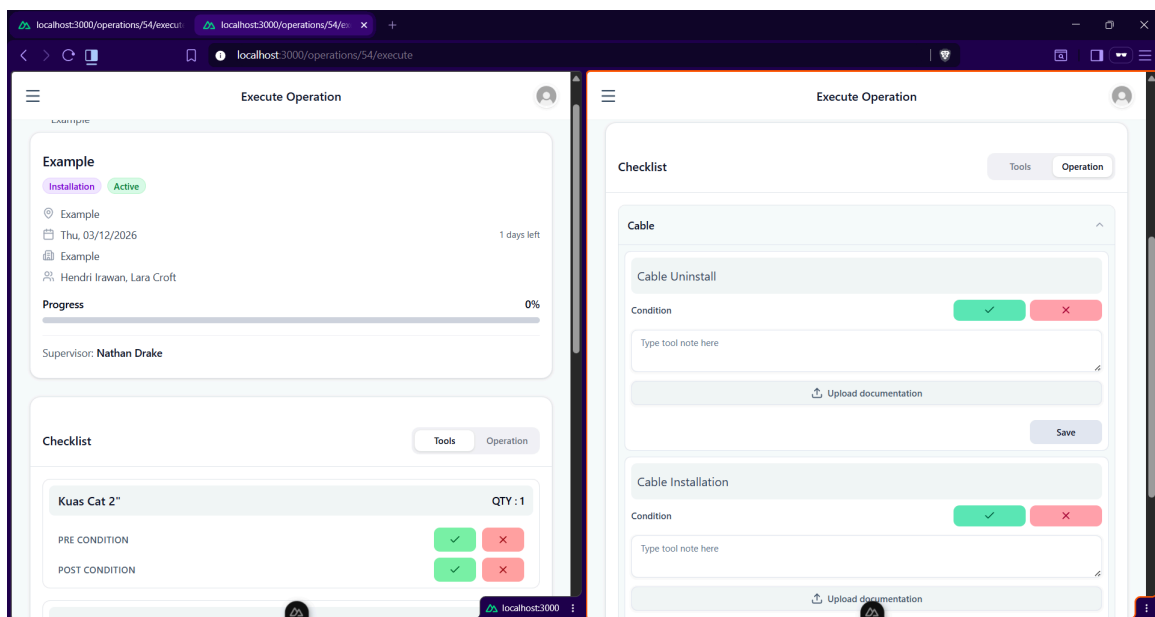


Figure 5.9 Checklist Execution Interface

Source : Practical Work Documentation, 2026

The checklist execution interface displays a list of operational tasks assigned to staff members. Users can update the completion status of each task and provide additional information such as notes or documentation when necessary. This functionality

enables the system to track operational progress and maintain accurate documentation of field activities.

5.2.6 Operation Monitoring Module

The operation monitoring module enables users such as Implementation Managers and Observers to monitor the progress of ongoing operations. This feature provides visibility into operational activities and allows stakeholders to evaluate task completion status.

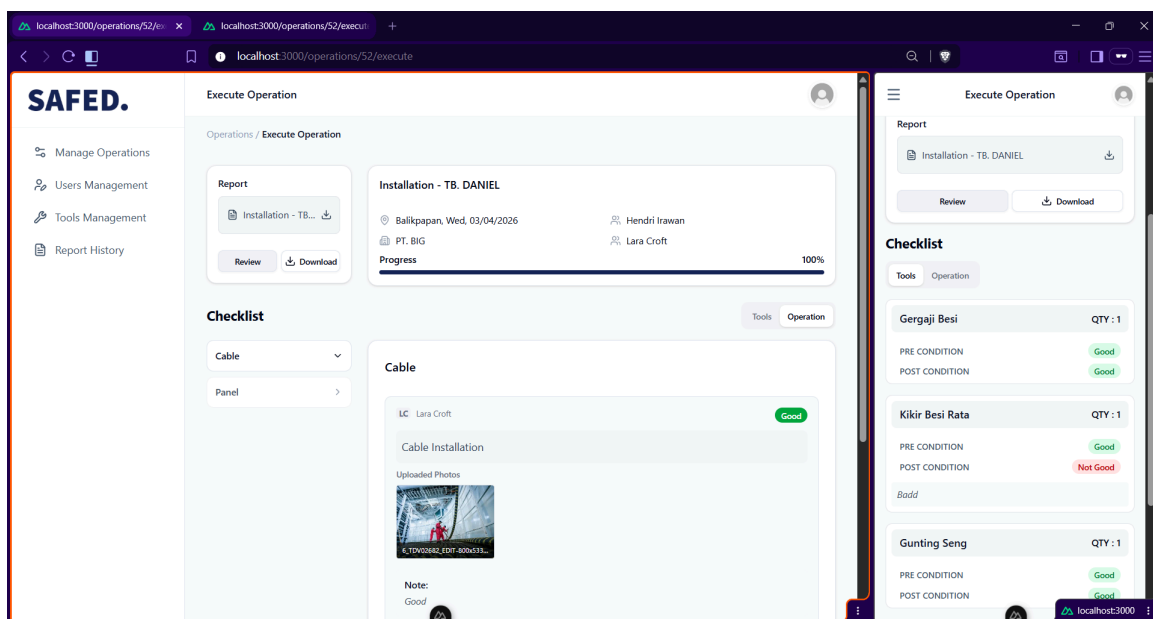


Figure 5.10 Operation Monitoring Interface
Source : Practical Work Documentation, 2026

The monitoring interface presents operational progress information through visual indicators that display the completion status of tasks within each operation. This allows users to quickly assess the progress of field activities and identify tasks that still require completion.

5.2.7 Field Report Generation Module

The field report generation module enables supervisors to generate operational reports based on the data recorded during field operations. This feature integrates operational data such as checklist completion records, notes, and documentation into a structured reporting format.

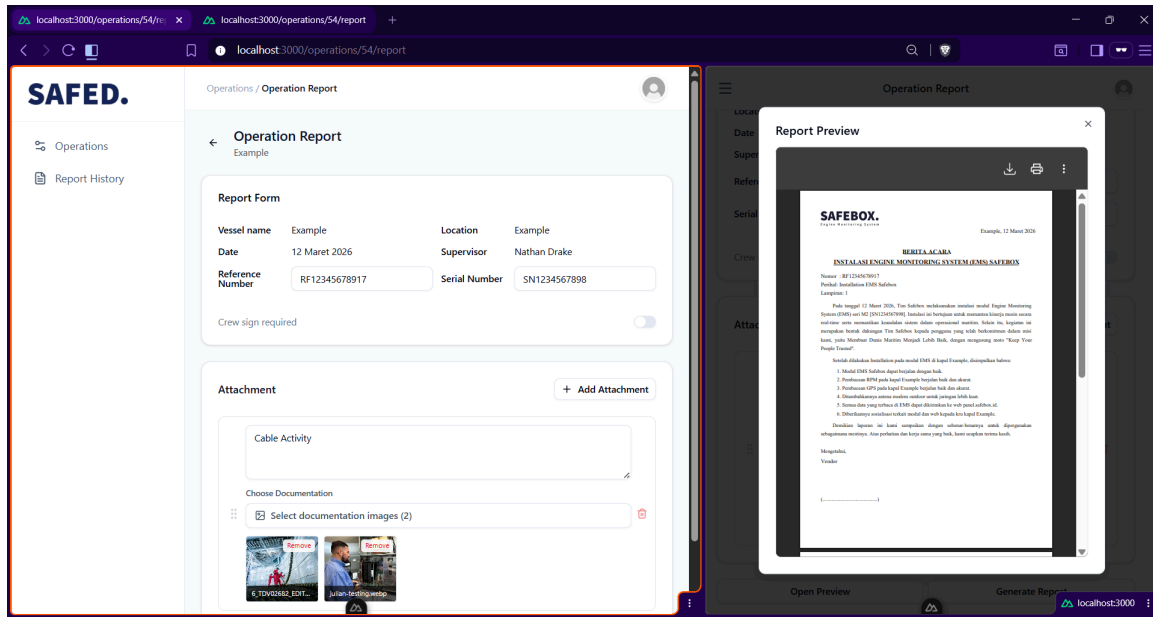


Figure 5.11 Field Report Generation Interface

Source : Practical Work Documentation, 2026

Through this interface, supervisors can review operational data collected during field activities and generate a draft field report. The system compiles the relevant information into a standardized report format that can be submitted for validation and documentation purposes.

5.3 System Testing

System testing was conducted to verify that the SAFED system operates according to the defined functional requirements. The testing process aimed to ensure that each implemented feature functions correctly and supports the operational workflow of the Manufacturing Division.

The testing approach used in this study was User Acceptance Testing (UAT). This method evaluates whether the developed system meets the expectations and operational requirements of the intended users. Through UAT, the system functionality was validated based on real user interactions with the system interface.

The testing activities were carried out by executing predefined test scenarios that simulate actual operational tasks performed by system users.

5.3.2 User Acceptance Testing Results

User Acceptance Testing was conducted to evaluate the functionality of the SAFED system based on the operational requirements identified during the development process. The testing covered several core system features including user management, authentication, tools management, operation scheduling, checklist execution, task monitoring, and report generation.

Each feature was tested through multiple test scenarios representing typical user interactions with the system. These scenarios included operations such as creating and managing user accounts, configuring operational schedules, executing operational checklists, updating task status, and generating operational reports.

The user acceptance testing results are presented in **Table 5.5**

Table 5.5 User Acceptance Testing Results

NO	Feature	Action Step	System Navigation	Expected Result	Test Result	Test Date	Test Name	Test Signature
US-01	User Management	Open the user management page without any filters	/users	The system can display a list of users with a maximum of 10 user data entries.	PASS	22/1/2026	Rakha	
		Select the +new option -> Fill in all form fields according to the requirements, then confirm "Create Account."	Dialog User Form	New user data is entered into the user database and displayed in the user table, and a message appears stating that the user has been successfully created.	PASS	22/1/2026	Rakha	
		Selecting the +new option -> Not filling in the form field according to the requirements	Dialog User Form	The "Create Account" button is disabled and cannot be clicked to create user data.	PASS	22/1/2026	Rakha	
		Select the +new option -> Do not fill in the form field according to the requirements, then click "Cancel."	Dialog User Form	No new user data has been registered in the database.	PASS	22/1/2026	Rakha	

		Select the Action -> Edit option on one of the users	Dialog User Form	User data is updated in the table and database according to the data entered in the edit form.	<i>PASS</i>	22/1/2026	Rakha	
		Press the Action -> Status option on one of the users.	Click Toggle Switch	The user status in the database changes, the color and label on the table change (Active/Inactive), and a success message appears.	<i>PASS</i>	22/1/2026	Rakha	
		Select the Action -> Delete option for one of the users, then confirm "Yes."	Dialog Confirm	User data is deleted from the database, the table automatically refreshes, and the total amount of data decreases.	<i>PASS</i>	22/1/2026	Rakha	
		Select the Action -> Delete option for one of the users, then confirm "No."	Dialog Confirm	No user data has been deleted from the database or table list.	<i>PASS</i>	22/1/2026	Rakha	
UC-02	Filter User	Press the page numbers "2, 3, 4..." at the bottom of the table.	/users?page=2, 3,4....	The system takes the next 10 data from the database. The sequence number (No) starts from 11, 21, 31....	<i>PASS</i>	22/1/2026	Rakha	
		Enter the user's name or email address in the Search field.	/users?search=jhon....	The automatically updated table only displays relevant users. The URL changes to include ?search=....	<i>PASS</i>	22/1/2026	Rakha	

		(example: john)						
		Select the Filter option, check “STAFF,” then click Apply.	/users?roles=STAFF	The table only displays users with the STAFF role. The number of filters badge appears on the Filter button.	PASS	22/1/2026	Rakha	
		Select the Filter option, check “OBSERVER,” then click Apply.	/users?roles=OBSERVER	The table only displays users with the OBSERVER role. The filter count badge appears on the Filter button.	PASS	22/1/2026	Rakha	
		Select the Filter option, check “IM,” then click Apply.	/users?roles=IM	The table only displays users with the IM role. The badge count filter appears on the Filter button.	PASS	22/1/2026	Rakha	
		Select the Filter option, check “STAFF” AND “OBSERVER,” then click Apply.	/users?roles=STAFF,OBSERVER	The table only displays users with the roles STAFF and OBSERVER. The number of filters appears on the Filter button.	PASS	22/1/2026	Rakha	
		Select the Filter option, check “STAFF” AND “IM,” then click Apply.	/users?roles=STAFF,IM	The table only displays users with the roles STAFF and IM. The badge number filter appears on the Filter button.	PASS	22/1/2026	Rakha	

		Select the Filter option, check “OBSERVER” AND “IM”, then click Apply.	/users?roles=OBSERVER,IM	The table only displays users with the OBSERVER AND IM roles. The number of filters badge appears on the Filter button.	PASS	22/1/2026	Rakha	
		Select the Filter option, then check “STAFF” and the “Active” status, then click Apply.	/users?roles=STAFF&status=true	The system displays users who are Active AND have the STAFF role at the same time. The number filter badge appears on the Filter button.	PASS	22/1/2026	Rakha	
		Select the Filter option, then check “STAFF” and the “Inactive” status, then click Apply.	/users?roles=STAFF&status=false	The system displays users who are Inactive AND have the STAFF role at the same time. The filter count badge appears on the Filter button.	PASS	22/1/2026	Rakha	
		Select the Filter option, then check “OBSERVER” and the “Active” status, then click Apply.	/users?roles=OBSERVER&status=true	The system displays users who are Active AND have the OBSERVER role at the same time. The filter count badge appears on the Filter button.	PASS	22/1/2026	Rakha	
		Select the Filter option, then check “OBSERVER” and the “Inactive” status, then click Apply.	/users?roles=OBSERVER&status=false	The system displays users who are Inactive AND have the OBSERVER role at the same time. The filter count badge appears on the Filter	PASS	22/1/2026	Rakha	

				button.				
		Select the Filter option, then check “IM” and the “Active” status, then click Apply.	/users?roles=IM&status=true	The system displays users who are Active AND have the IM role at the same time. The filter count badge appears on the Filter button.	<i>PASS</i>	22/1/2026	Rakha	
		Select the Filter option, then check “IM” and the “Inactive” status, then click Apply.	/users?roles=IM&status=false	The system displays users who are Inactive AND have the IM role at the same time. The filter count badge appears on the Filter button.	<i>PASS</i>	22/1/2026	Rakha	
		Select the Filter option, then click the “Clear” button or uncheck all boxes.	/users	The filter is removed, the URL is clean again, and the table displays all users again.	<i>PASS</i>	22/1/2026	Rakha	
US-03	Authentication	Enter the appropriate username and password -> Click “Sign In”	/	The user successfully logs in, a login success message appears, and the user is directed to a specific page according to their role, and receives a session token.	<i>PASS</i>	28/1/2026	Rakha	

		Just enter your username and leave the password field blank -> Click “Sign In”	/	An error message appears in the password field, as well as an error message at the top of the screen.	<i>PASS</i>	28/1/2026	Rakha	
		Just enter your password and leave the username field blank -> Click “Sign In”	/	An error message appears in the username field, as well as an error message at the top of the screen.	<i>PASS</i>	28/1/2026	Rakha	
		Do not fill in any data in the login form fields -> Click “Sign In”	/	An error message appears in the username and password fields, as well as at the top of the screen.	<i>PASS</i>	28/1/2026	Rakha	
UC-15	Manage Tools	Open the management tools page without any filters	/tools	The system can display a list of tools with a maximum of 10 data records.	<i>PASS</i>	3/2/2026	Rakha	
		Select the +new option -> Fill in the tools form field according to the requirements, then confirm “Create.”	Dialog Tools Form	The new data tools are entered into the database tools and displayed in the tools table, and a message appears stating that the tools have been successfully created.	<i>PASS</i>	3/2/2026	Rakha	

		Selecting the +new option -> Not filling in the tools form field according to the requirements	Dialog User Form	A validation message appears on the form and a notification that the tools failed to be added and require tool names according to the provisions, there is no new tool data in the tool database.	PASS	3/2/2026	Rakha	
		Select the Action -> Edit option on one of the tools items	Dialog Tools Form	Updated data tools in tables and databases according to the data entered in the edit form.	PASS	3/2/2026	Rakha	
		Select the Action -> Delete option on one of the tools, then confirm "Yes."	Dialog Confirm	Data tools were deleted from the database, tables automatically refreshed, and the total amount of data decreased.	PASS	3/2/2026	Rakha	
		Select the Action -> Delete option on one of the tools, then confirm "No."	Dialog Confirm	No data tools have been deleted from the database or tables.	PASS	3/2/2026	Rakha	
		Enter the name of the tool in the Search field. (example: pliers)	/users?search=tag....	The automatically updated table only displays relevant tools. The URL changes to include ?search=....	PASS	3/2/2026	Rakha	

UC-06	Manage Operation	Open the management operation page without any filters as IM	/operations	The system can display all lists of operations that will run, are running, or have been completed.	PASS	19/2/2026	Rakha	
		Open the Operations page -> click +New Operation	/operations/create	The system displays the form for creating new operation data. The input fields (Type, Company, Vessel Name, Location, Date, Supervisor, and Staff) appear correctly, while the 'Set Checklist Tools and Job Desc' section is inaccessible and blurred.	PASS	19/2/2026	Rakha	
		Open the Operations page -> click +New Operation -> fill out the entire form accordingly -> click "Finish Preparation" -> click "Yes"	/operations/create	The new data operation is entered into the database operations and displayed in the operations list table, and a message appears stating that the operation has been successfully created.	PASS	19/2/2026	Rakha	
		Open the Operations page -> click +New Operation -> Do not fill in the form fields according to the requirements.	/operations/create	The "Save" button is disabled and cannot be clicked to create data operations.	PASS	19/2/2026	Rakha	

		Open the Create page -> Assign Supervisor -> Assign more than 1 staff member	/operations/create	The system can retrieve user data and then assign it to the operation.	<i>PASS</i>	19/2/2026	Rakha	
		Open the Operations page -> click the 3 dots (kebab menu) -> Try selecting the “Edit” option on operations with the status “Complete, Active, Canceled”	Dropdown	The menu for the “Edit” option does not appear in operations with that status.	<i>PASS</i>	19/2/2026	Rakha	
		Open the Operations page -> click the 3 dots (kebab menu) -> Select the “Edit” option for operations with “Draft” status	/operation/create?id=	Updated data operations in tables and databases according to the data entered in the edit form.	<i>PASS</i>	19/2/2026	Rakha	
		Open the Operations page -> click the 3 dots (kebab menu) -> Select the “Delete” option for operations with ‘Draft’ status, then confirm “Yes.”	Dialog Confirm	The deleted data operation from the database, the table automatically refreshes, and the total amount of data decreases.	<i>PASS</i>	19/2/2026	Rakha	

		Open the Operations page -> click the 3 dots (kebab menu) -> Select the “Delete” option for operations with ‘Draft’ status, then confirm “No.”	Dialog Confirm	No operation data has been deleted from the database or the operation list.	<i>PASS</i>	19/2/2026	Rakha	
		Open the Operations page -> click the 3 dots (kebab menu) -> Try selecting the “Delete” option for operations with the status “Complete, Active, Canceled”	Dropdown	The menu for the “Delete” option does not appear for operations with that status.	<i>PASS</i>	19/2/2026	Rakha	
UC-07	Manage Tools Checklist	Open the “New Operation” page -> Go to the Checklist section -> Select and add several tools and set the quantity -> Click “Save”	/operations/create -> Checklist -> Tools	Data tools and quantities are stored in the database and will appear on the Execute Operation page.	<i>PASS</i>	19/2/2026	Rakha	
		Open one of the operations -> Fill in the Pre condition (Good/Not Good) -> Add a note if	/operations/{id}/execute -> Tools	Pre-tools condition data is stored in a single batch, with status and notes remaining after page refresh.	<i>PASS</i>	19/2/2026	Rakha	

		necessary -> Click Save						
		Open one of the operations -> Fill in the Post condition (Good/Not Good) -> Add a note if necessary -> Click Save	/operations/{id}/execute -> Tools	Post tool status data is stored in a single batch; status and notes remain after page refresh.	PASS	19/2/2026	Rakha	
		Open the Execute page -> Leave the Pre/Post conditions blank -> Click Save	/operations/{id}/execute -> Tools	The system rejects the storage and displays a validation message that the required fields must be filled in.	PASS	19/2/2026	Rakha	
		Fill in all the tool conditions correctly -> Click Save	/operations/{id}/execute -> Tools	The progress of the operation increases according to the number of tools that have been completed.	PASS	19/2/2026	Rakha	
UC-08	Manage Jobdesk Checklist	Open the “New Operation” page -> Checklist -> Operation -> Add Section, Module, and Activity -> Click Save	/operations/create -> Checklist -> Operation	The Section/Module/Activity structure is saved and available on the Execute page.	PASS	19/2/2026	Rakha	

		Open the “New Operation” page -> Checklist -> Operation -> Turn on the “Documentation Required” toggle	/operations/create -> Checklist -> Operation	In certain activities, users can add documentation.	PASS	19/2/2026	Rakha	
UC-04	Operation List	Open the operation list page as IM/OBSERVER.	/operations	System displays Operation List header with Search + Filter; shows operation stats; shows a grid of operation cards with Type/Status badges, Location, Date, Staff assigned count, Progress bar/%, and Supervisor name; cards are clickable.	PASS	19/2/2026	Rakha	
		Click an operation card.	/operations -> /operations/{id}/execute	System navigates to the selected operation Execute page.	PASS	19/2/2026	Rakha	
		Login as IM, open the kebab menu (3 dots) on an operation card.	/operations	The system shows actions: “View Details” always; “Edit” only when status = Draft; “Delete” only when status = Draft; “Cancel Operation” when status is not Complete and not Cancelled.	PASS	19/2/2026	Rakha	

		Log in as STAFF and open the operation list page.	/operations	The system lists only operations where the STAFF user is enrolled/assigned; operation cards still show progress and details; users can open assigned operations.	<i>PASS</i>	19/2/2026	Rakha	
UC-05	Filter Operation	Type a keyword into Search (company/location/vessel name).	/operations?search=...	System updates URL with search=... and refreshes the grid to only matching operations.	<i>PASS</i>	19/2/2026	Rakha	
		Open Filter -> select a Date -> click Apply Filters.	/operations?date=YYYY-MM-DD	System filters operations to the selected date; filter badge count increases by 1; Clear all removes the date filter and restores list.	<i>PASS</i>	19/2/2026	Rakha	
		Open Filter -> select one or more Types -> Apply Filters.	/operations?type=TYPE1,TYPE2	System filters operations by selected type(s); selected types appear as badges in the filter UI; URL uses comma-separated type=....	<i>PASS</i>	19/2/2026	Rakha	
		Open Filter -> select one or more Statuses -> Apply Filters.	/operations?status=STATUS1,STATUS2	System filters operations by selected status(es); URL uses comma-separated status=....	<i>PASS</i>	19/2/2026	Rakha	

		Use multiple filters together (Date + Type + Status) then Clear all.	/operations?... -> /operations	System applies combined filtering; badge count reflects how many filter categories are active (date/types/status); Clear all resets filters and URL back to /operations.	<i>PASS</i>	19/2/2026	Rakha	
UC-10	Update Checklist Status	Open the Operations page -> Select the Tools tab	/operation/[id]/execute	The system displays a list of tasks on the Tools checklist.	<i>PASS</i>	19/2/2026	Rakha	
		Open the Operations page -> Select the Operations tab	/operation/[id]/execute	The system displays a list of tasks on the Operations checklist.	<i>PASS</i>	19/2/2026	Rakha	
		Open the Operations page -> Select the assigned operation -> Select the "Good" option	/operation/[id]/execute	The status checklist data is saved as "Good."	<i>PASS</i>	19/2/2026	Rakha	
		Open the Operations page -> Select the assigned operation -> Select the "Not Good" option	/operation/[id]/execute	The status checklist data is saved as "Not Good."	<i>PASS</i>	19/2/2026	Rakha	

		Open the Operations page -> Fill in the note accordingly	/operation/[id]/execute	Operational data notes are stored in the database.	PASS	19/2/2026	Rakha	
UC-11	Approve Checklist Task	Open the Operations page -> Fill in all the checklists correctly -> Click save	Save Checklist Form Dialog	The system saves the checklist status with a notification of success, the progress bar increases, and IM/OBSERVER can see tasks with updated statuses.	PASS	20/2/2026	Rakha	
		Open the Operations page -> There are several checklists that have not been filled in -> click save	Save Checklist Form Dialog	The system displays a notification that there are tasks that have not been filled in.	PASS	20/2/2026	Rakha	
		Open the Operations page -> click "Finish Operation"	Finish Checklist Form Dialog	The operation checklist data is stored in the database along with the progress status becoming 100% and	PASS	20/2/2026	Rakha	
UC-12	Generate Field Report	On the SUPERVISOR report page, fill in required fields (Reference Number, Serial Number, at least 1 Attachment	/operation/[id]/report	System validates inputs; shows "Preparing preview..." then opens a preview dialog with an embedded PDF (iframe) rendered from the entered data; no data is permanently	PASS	24/2/2026	Rakha	

		note) and optional Attachment items -> click Preview (Open Preview).		saved yet.				
		On the SUPERVISOR report page, do not fill in required fields -> click Preview (Open Preview).	/operation/[id]/report	System blocks preview; shows validation toast message for the first failing rule (e.g., “Reference number is required” / “Serial number is required” / “At least one attachment note is required”); preview dialog is not shown.	PASS	24/2/2026	Rakha	
UC-13	Submit Field Report	On the SUPERVISOR report page, fill in required fields (Reference Number, Serial Number, at least 1 Attachment note) and optional Attachment items -> click Generate Report.	/operation/[id]/report	System validates inputs, generates the field report PDF, shows success toast “Field report generated”, downloads the PDF, then page refresh shows “Generated Report” section with generated timestamp and in-page PDF viewer; Review/Download buttons work.	PASS	24/2/2026	Rakha	
		On the SUPERVISOR report page, do not fill in required fields	/operation/[id]/report	System blocks generation; shows validation toast message for the first failing rule; no report PDF is	PASS	24/2/2026	Rakha	

		-> click Generate Report.		generated/downloaded; “Generated Report” section does not appear.				
UC-14	Monitor Operation Progress	Open the Operations page	/operations	The progress bar/percentage on each card corresponds to the number of checklists that have been completed.	PASS	24/2/2026	Rakha	
		Complete one checklist activity/tool -> Refresh the list page	/operations	The progress value increases according to the latest changes	PASS	24/2/2026	Rakha	
		Log in as an Observer -> Open the Operation Details page	/operations/{id}/execute	Observers can view the entire checklist and progress but cannot edit or save changes.	PASS	24/2/2026	Rakha	
		Log in as an Observer -> Try changing the activity or tools status	/operations/{id}/execute	The system rejects changes and fields in disabled/read-only state	PASS	24/2/2026	Rakha	

Source : Practical Work Report Documentation, 2026

5.4 System Evaluation and Analysis

This section evaluates the implementation of the SAFED system and analyzes how the developed system supports the operational activities within the Manufacturing Division of PT Bintang Indayu Gemilang. The evaluation focuses on how the system addresses the operational challenges identified during the initial analysis phase and how the implemented solution improves the efficiency of field operation management.

The analysis is conducted based on the results of the system development process, system testing outcomes, and observations made during system implementation.

5.4.1 System Impact on Operational Workflow

Prior to the development of the SAFED system, several operational activities within the Manufacturing Division were conducted using manual documentation and spreadsheet-based records. Operational checklists were often recorded using printed documents or shared spreadsheets, which created several operational challenges.

One of the main issues encountered was the difficulty in monitoring the progress of field operations. Since operational data was recorded manually, supervisors and managers were required to collect updates through direct communication with field staff. This process made it difficult to obtain a real-time overview of the operational status.

Another issue involved the management of equipment preparation and checklist documentation. Field teams were required to manually track the tools required for each operation, which increased the risk of incomplete equipment preparation. In addition, documentation of operational tasks was often scattered across different platforms, including messaging applications and spreadsheet files.

The SAFED system addresses these challenges by providing a centralized digital platform that integrates several operational processes into a single system environment. Through this system, operational schedules, equipment checklists, and field task documentation can be managed in a structured and centralized manner.

The implementation of the checklist execution module allows staff members to update task completion status directly within the system during field operations. This functionality enables supervisors and managers to monitor operational progress more efficiently without relying solely on manual updates or communication through external platforms.

The system also improves equipment preparation management through the tools checklist feature, which allows operational teams to define the required tools before the operation begins. This feature helps ensure that the necessary equipment is properly prepared and documented prior to field execution.

In addition, the field report generation module simplifies the reporting process by automatically compiling operational data into a structured report format. Previously, field reports had to be manually compiled from various sources of documentation, which often caused delays in report preparation. Through the SAFED system, operational reports can now be generated directly from the recorded checklist and documentation data.

Overall, the implementation of the SAFED system provides a more structured workflow for managing field operations. By centralizing operational data and providing real-time monitoring capabilities, the system helps improve operational transparency and documentation accuracy within the Manufacturing Division.

5.4.2 Development Challenges and Solutions

During the development process of the SAFED system, several challenges were encountered that affected the implementation timeline and development workflow. These challenges primarily involved requirement adjustments, operational constraints during the internship period, and coordination with stakeholders responsible for field operations.

One of the main challenges occurred during the second development iteration, where several planned features could not be fully completed within the scheduled iteration timeframe. The delay was primarily caused by external factors, including the need to

clarify operational procedures with stakeholders and adjustments to system requirements based on feedback obtained during development meetings.

In addition, the dynamic nature of operational requirements required several modifications to the initial system design. During daily stand-up discussions and weekly coordination meetings with stakeholders, several additional requirements were identified, including improvements to the field reporting workflow and the addition of tools management functionality.

To address these challenges, the development process adopted the iterative nature of the Personal Extreme Programming (PXP) methodology. Tasks that could not be fully completed during the second iteration were re-prioritized and carried over to the third development iteration. This approach allowed unfinished features to be refined and completed without disrupting the overall system development process.

Another challenge encountered during development involved ensuring that the system interface remained intuitive for different types of users, including supervisors, staff members, and observers. Since the system supports multiple operational roles, additional adjustments were made to ensure that the interface displays only relevant information and functionality for each user role.

Through continuous evaluation and iterative refinement, the SAFED system was gradually improved throughout the development process. The flexibility provided by the PXP methodology enabled the development process to adapt to requirement changes while maintaining a structured development workflow.

At the conclusion of the development process, the SAFED system successfully implemented the core functionalities required to support operational scheduling, equipment checklist management, field task monitoring, and operational report generation within the Manufacturing Division.

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1. Conclusion

Based on the entire process of problem identification, requirement analysis, system development, and evaluation conducted during the practical work program at PT Bintang Indayu Gemilang, several conclusions can be drawn.

The main problem identified within the Manufacturing Division was the difficulty in recording and monitoring field operation activities in a structured and integrated manner. Prior to the development of the system, operational checklists and documentation were primarily managed using manual records and spreadsheet-based files, which made it difficult to track operational progress and increased the potential for documentation inconsistencies.

To address these challenges, a system called the Field Automation System (SAFED) was developed to support the management of field operations within the Manufacturing Division. The system was developed using the Personal Extreme Programming (PXP) methodology, which allowed the development process to be carried out iteratively through several development cycles. This approach enabled the system to be gradually improved and adjusted based on feedback obtained during the development process.

The development process resulted in the implementation of several core system features, including user authentication, user management, tools management, operation scheduling, checklist management for operational activities, progress monitoring, and automated field report generation. These features were designed to support the operational workflow of field teams by providing a structured digital platform for recording operational activities and monitoring work progress.

System testing was conducted using the User Acceptance Testing (UAT) method to evaluate whether the developed system met the defined functional requirements. The

testing results, as summarized in Table 5.3, show that all evaluated system modules successfully passed the testing scenarios. The tested features include user management, authentication, tools management, operation scheduling, checklist execution, operational monitoring, and field report generation. These results indicate that the SAFED system is capable of supporting operational activities such as equipment preparation, task monitoring, and operational reporting in a structured and efficient manner.

Overall, the SAFED system provides a centralized platform that improves the management of field operations by integrating operational scheduling, equipment preparation, task monitoring, and operational reporting into a single system environment. The implementation of this system is expected to enhance operational efficiency, improve documentation accuracy, and facilitate better monitoring of field operation activities within the Manufacturing Division.

6.2 Recommendations

Based on the results of the practical work and the development of the SAFED system, several recommendations can be proposed to support future improvements and implementation of the system.

First, the company may consider further developing the SAFED system so that it can be fully implemented within the operational workflow of the Manufacturing Division. Additional features such as mobile device support, real-time notification systems, and enhanced operational analytics may be added to improve system usability and facilitate monitoring during field operations.

Second, educational institutions are encouraged to strengthen collaboration with industry partners in order to provide students with broader opportunities to gain practical experience in real-world professional environments. Institutions may also consider developing courses related to modern software development methodologies and system implementation practices to better prepare students for industry-based projects.

Third, future students who participate in practical work programs are encouraged to prepare their technical skills and knowledge before beginning the internship period. In addition, maintaining effective communication with supervisors and team members is important to ensure a clear understanding of project requirements and operational workflows. Students are also advised to document their daily activities and development progress carefully, as this documentation will assist in preparing the final practical work report.

REFERENCES

- Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press.
- Beck, K. (2012). *Extreme Programming Explained Second Edition Embrace Change* (Don O'Hagan, Ed.). John Wait.
- Beck, K., Beedle, M., Bennekum, A. v., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development* [We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value: Individuals and interactions over processes and tools Working software over comprehensive documentation Customer collaborati].
- Elmasri, R., & Navathe, S. (2016). *Fundamentals of Database Systems* (7th ed.). Pearson.
- Fowler, M. (2003). *Patterns of Enterprise Application Architecture*. Addison-Wesley.
- Heizer, J., Render, B., & Munson, C. (2019). *Operations Management: Sustainability and Supply Chain Management* (13th ed.). Pearson.
- Laudon, K. C., & Laudon, J. P. (2020). *Management Information Systems: Managing the Digital Firm*. Pearson.
- Lee, J., Bagheri, B., & Kao, H.-A. (2015, January). Manufacturing Letters. *A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems*, 3, 18-23. <https://doi.org/10.1016/j.mfglet.2014.12.001>
- Martin, R. C. (2003). *Agile Software Development: Principles, Patterns, and Practices*. Pearson Education.

- Nuxt.js Documentation*. (2023). Nuxt: The Full-Stack Vue Framework. Retrieved March, 2026, from <https://nuxt.com/>
- Pressman, R. S., & Maxim, B. R. (2019). *Software Engineering: A Practitioner's Approach* (9th ed.). McGraw-Hill Education.
- Roda, I., Macchi, M., Fumagalli, L., & Viveros, P. (2014, July 25). A review of multi-criteria classification of spare parts : From literature analysis to industrial evidences. *Journal of Manufacturing Technology Management*, 25, 528-549.
<https://doi.org/10.1108/JMTM-04-2013-0038>
- Seidl, M., Scholz, M., Huemer, C., & Kappel, G. (2015). *UML @ Classroom: An Introduction to Object-Oriented Modeling*. Springer International Publishing.
- Shi, Z., Xie, Y., Xue, W., Chen, Y., Fu, L., & Xu, X. (2020, June 21). Systems Research and Behavioral Science. *Smart factory in Industry 4.0.*, 37(4), 607-617.
<https://doi.org/10.1002/sres.2704>
- Slack, N., Brandon-Jones, A., & Burgess, N. (2022). *Operations Management* (10th ed.). Pearson.
- Sommerville, I. (2016). *Software Engineering* (10th ed.). Pearson.
- Stair, R., & Reynolds, G. (2018). *Principles of Information Systems* (13th ed.). Cengage Learning.
- Stonebraker, M., & Kemintz, G. (1991). The POSTGRES Next Generation Database Management System. *Communications of the ACM*, 34(10), 78-92.
- Tortorella, G. L., & Diego Fettermann. (2018, March 2). Implementation of Industry 4.0 and lean production in Brazilian manufacturing companies. *International Journal of Production Research*, 56(8), 2975-2987. 10.1080/00207543.2017.1391420

- Dzhurov, Y., Krasteva, I. and Ilieva, S., 2009. Personal Extreme Programming—An Agile Process for Autonomous Developers. ResearchGate, pp.252-259.
- Tsang, A.H.C. (2002), “Strategic Dimensions of Maintenance Management”, Journal of Quality in Maintenance Engineering, Vol. 8, No. 1, pp. 7-39.

ATTACHMENT

Weekly Log Sheet,



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**
Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801
Email: humas@itk.ac.id

Form. KP-002

LOG SHEET MINGGUAN KERJA PRAKTIK

Nama Mahasiswa : Norbertino Eurakha Nandatoti 10231071
Program Studi : Sistem Informasi
Jurusan : Jurusan Teknik Elektro Informatika dan Bisnis
Fakultas : Fakultas Sains dan Teknologi Informasi
Dosen Pembimbing : Henokh Lugo Hariyanto, S.Si., M.Sc.
Mitra KP : PT. Bintang Indayu Gemilang
Periode KP : Bulan Januari - Februari, Tahun 2026
Minggu ke- : 1

Hari/Tanggal	Uraian Aktivitas	Jam Datang	Jam Pulang	Paraf Pembimbing Lapangan
05/01/2026	Melakukan riset dan penyusunan user story serta diskusi intensif bersama Mentor dan Incha. Aktivitas mencakup setup ClickUp, perbaikan draf use case, dan persiapan asistensi dengan Dosen.	08:30 WITA	17:00 WITA	
06/01/2026	Melakukan riset UML melalui buku dan video untuk menyusun use case beserta class diagram proyek "SAFED". Aktivitas mencakup penetapan framework, penyusunan project requirements, serta diskusi intensif bersama Mentor dan Incha. Proses diakhiri dengan perbaikan dokumen berdasarkan revisi	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	untuk persiapan asistensi bersama Dosen pembimbing.			
07/01/2026	Mengikuti Weekly Retrospective serta mulai menyusun catatan tugas harian. Aktivitas fokus pada eksplorasi Nuxt Framework, perancangan skema DB, dan perbaikan use case. Proses diperkuat melalui diskusi bersama Mentor dan Incha untuk mematangkan project requirements dan struktur implementasi kode.	08:30 WITA	17:00 WITA	
08/01/2026	Mengikuti Morning Brief dan menyusun catatan tugas harian. Aktivitas dilanjutkan dengan penyusunan Project Requirements melalui diskusi intensif bersama Mentor.	08:30 WITA	17:00 WITA	
09/01/2026	Mengikuti Morning English Session dan Weekly Report Progress untuk melaporkan perkembangan proyek. Aktivitas dilanjutkan dengan penyusunan catatan tugas harian, perumusan Project	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127

Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	Requirements, serta analisis breakdown tugas lintas fungsi pada divisi Marketing. Seluruh rangkaian ditutup dengan mengasah kemampuan komunikasi melalui presentasi mingguan.			
--	---	--	--	--

Catatan Dosen Pembimbing:

Balikpapan,
Dosen Pembimbing

Henokh Lugo Hariyanto,
S.Si.,M.Sc.
NIP 199303062022041001



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

Form. KP-002

LOG SHEET MINGGUAN KERJA PRAKTIK

Nama Mahasiswa : Norbertino Eurakha Nandatoti 10231071
Program Studi : Sistem Informasi
Jurusan : Jurusan Teknik Elektro Informatika dan Bisnis
Fakultas : Fakultas Sains dan Teknologi Informasi
Dosen Pembimbing : Henokh Lugo Hariyanto, S.Si., M.Sc.
Mitra KP : PT. Bintang Indayu Gemilang
Periode KP : Bulan Januari - Februari, Tahun 2026
Minggu ke- : 2

Hari/Tanggal	Uraian Aktivitas	Jam Datang	Jam Pulang	Paraf Pembimbing Lapangan
12/01/2026	Mengikuti Daily Core Values dan melakukan perakitan fasilitas kerja baru. Aktivitas utama difokuskan pada pendefinisian peran sistem melalui studi kasus aplikasi riil, perancangan skema database, serta riset literatur basis data. Selain itu, dilakukan pendalaman metodologi melalui buku Extreme Programming (XP) untuk memahami fase implementasinya secara mendalam.	08:30 WITA	17:00 WITA	
13/01/2026	Mempresentasikan progres harian. Aktivitas teknis difokuskan pada eksplorasi Nuxt Framework, mulai dari struktur folder,	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	sistem routing (statis/dinamis), hingga penggunaan reference. Proses ini diperkuat dengan pembuatan playground project, refaktorisasi product backlog, serta pengembangan diri melalui kelas Udemmy dan latihan teknik presentasi.			
14/01/2026	Mengikuti Weekly Retrospective dan koordinasi bersama tim Marketing. Aktivitas teknis difokuskan pada penyelesaian skema database serta pendalaman implementasi SEO dan Metadata pada framework Nuxt.	08:30 WITA	17:00 WITA	
15/01/2026	Mengikuti koordinasi bersama tim Marketing dan mendalami teknis pengembangan Nuxt. Aktivitas difokuskan pada penguasaan layouting, link, dan navigasi, serta implementasi middleware baik secara global maupun lokal untuk memperkuat struktur komponen aplikasi.	08:30 WITA	17:00 WITA	
16/01/2026	Libur			



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

Catatan Dosen Pembimbing:

Balikpapan,
Dosen Pembimbing

Henokh Lugo Hariyanto,
S.Si.,M.Sc.
NIP 199303062022041001



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801
Email: humas@itk.ac.id

Form. KP-002

LOG SHEET MINGGUAN KERJA PRAKTIK

Nama Mahasiswa : Norbertino Eurakha Nandatoti 10231071
Program Studi : Sistem Informasi
Jurusan : Jurusan Teknik Elektro Informatika dan Bisnis
Fakultas : Fakultas Sains dan Teknologi Informasi
Dosen Pembimbing : Henokh Lugo Hariyanto, S.Si., M.Sc.
Mitra KP : PT. Bintang Indayu Gemilang
Periode KP : Bulan Januari - Februari, Tahun 2026
Minggu ke- : 3

Hari/Tanggal	Uraian Aktivitas	Jam Datang	Jam Pulang	Paraf Pembimbing Lapangan
19/01/2026	Melakukan persiapan dan menghadiri Weekly Report Progress untuk melaporkan perkembangan proyek. Aktivitas teknis difokuskan pada penyusunan Iteration Plan pertama, melakukan penyesuaian product backlog dan db schema, serta merancang class diagram. Selain itu, dilakukan pemindahan backlog ke ClickUp untuk iterasi pertama dengan penentuan story point menggunakan teknik anchoring dan deret Fibonacci berdasarkan hasil diskusi bersama Incha.	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127

Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

20/01/2026	Melakukan persiapan dokumentasi dan menghadiri sesi asistensi bersama Bpk. Henokh. Aktivitas teknis difokuskan pada inisiasi proyek "SAFED" melalui pembuatan repositori GitHub, instalasi Nuxt, dan penyusunan struktur folder. Selain itu, dilakukan pendalaman materi serta perancangan class diagram serta riset mengenai implementasi operasi CRUD pada framework Nuxt.	08:30 WITA	17:00 WITA	
21/01/2026	Melakukan commit pertama pada repositori dan menetapkan struktur folder proyek. Aktivitas teknis difokuskan pada pembuatan tabel dasar serta pemuatan data pengguna menggunakan static data. Selain itu, dilakukan diskusi bersama Incha mengenai color palette design system serta pendalaman materi Type Assertion pada TypeScript untuk memperkuat	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	validasi tipe data.			
22/01/2026	Melakukan pengembangan UI pada User List, mencakup pembaruan header, penambahan fitur pencarian, serta filter peran menggunakan dropdown. Aktivitas teknis difokuskan pada pembuatan UI dialog untuk tambah/edit pengguna, tombol aksi, serta penyusunan user composible. Selain itu, dilakukan pendalaman mengenai pembuatan API di Nuxt dan pemecahan masalah terkait identifikasi tipe data pada TypeScript.	08:30 WITA	17:00 WITA	
23/01/2026	Mengikuti Morning English Day dengan topik manfaat aktivitas fisik bagi kesehatan mental. Aktivitas teknis difokuskan pada refaktorisasi API routes ke dalam metode spesifik, penyiapan fase Iterasi 2, serta konfigurasi Drizzle ORM untuk koneksi dan basis data pengguna. Selain itu,	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801
Email: humas@itk.ac.id

	dilakukan pendalaman mengenai standar GitHub commit message serta pemecahan masalah terkait kendala koneksi database.			
--	--	--	--	--

Catatan Dosen Pembimbing:

Balikpapan,
Dosen Pembimbing

Henokh Lugo Hariyanto,
S.Si.,M.Sc.
NIP 199303062022041001



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

Form. KP-002

LOG SHEET MINGGUAN KERJA PRAKTIK

Nama Mahasiswa : Norbertino Eurakha Nandatoti 10231071
Program Studi : Sistem Informasi
Jurusan : Jurusan Teknik Elektro Informatika dan Bisnis
Fakultas : Fakultas Sains dan Teknologi Informasi
Dosen Pembimbing : Henokh Lugo Hariyanto, S.Si., M.Sc.
Mitra KP : PT. Bintang Indayu Gemilang
Periode KP : Bulan Januari - Februari, Tahun 2026
Minggu ke- : 4

Hari/Tanggal	Uraian Aktivitas	Jam Datang	Jam Pulang	Paraf Pembimbing Lapangan
26/01/2026	Berpartisipasi dalam M-Series Sprint Review dan melakukan pengujian sistem secara komprehensif menggunakan Playwright (E2E) dan Vitest (Unit Test). Aktivitas teknis difokuskan pada pengembangan UI/UX, mencakup implementasi layout desktop, sidebar, halaman login, serta komponen DeleteConfirmDialog yang independen. Selain itu, dilakukan pendalaman metode testing serta pemecahan masalah terkait konfigurasi testing tools dan optimasi layout	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	agar responsif di perangkat seluler.			
27/01/2026	Melakukan optimasi pada User List dengan implementasi pengurutan berdasarkan ID terkecil, sistem paginasi, serta filter berdasarkan peran dan status. Aktivitas teknis mencakup kustomisasi identitas visual melalui perubahan warna primer sesuai desain Incha (menggantikan default Shadcn), integrasi notasi Sonner setelah penghapusan data, serta pendalaman materi mengenai reusable UI components.	08:30 WITA	17:00 WITA	
28/01/2026	Melakukan pengembangan fitur pada halaman Users, mencakup implementasi sistem paginasi berbasis URL query (limit 10 item), fitur bulk selection menggunakan checkbox, serta filter tingkat lanjut berdasarkan peran dan status. Aktivitas teknis difokuskan pada refaktorisasi API users.get untuk	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	mendukung parameter kueri, integrasi Sonner toast, pembaruan aset visual, serta optimasi UI melalui komponen Separator dan dropdown actions. Selain itu, dilakukan pendalaman materi mengenai manajemen state halaman melalui URL serta pemecahan masalah terkait logika filtrasi kompleks yang menggabungkan pencarian dan filter status.			
29/01/2026	Implementasi sistem autentikasi dasar dengan fitur sesi menggunakan modul nuxt-utils serta pengamanan kata sandi melalui metode hashing bcrypt. Aktivitas teknis difokuskan pada penerapan Global Middleware untuk proteksi rute, serta penyusunan dan pengujian tabel User Acceptance Testing (UAT) untuk validasi fitur. Selain itu, dilakukan pendalaman materi mengenai atribut formulir UAT serta	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	pemecahan masalah terkait strategi pengujian yang tepat dan alur autentikasi pada middleware.			
30/01/2026	Melakukan persiapan Weekly Meeting dan memperbarui User Story untuk fitur-fitur yang telah selesai diimplementasikan agar statusnya sesuai di backlog. Aktivitas utama difokuskan pada sinkronisasi progres teknis dengan dokumentasi proyek, sembari memecahkan kebingungan terkait klasifikasi fitur yang harus masuk ke dalam Use Case dibandingkan dengan yang masuk ke dalam Product Backlog.	08:30 WITA	17:00 WITA	

Catatan Dosen Pembimbing:

Balikpapan,
Dosen Pembimbing



Henokh Lugo Hariyanto,
S.Si., M.Sc.
NIP 199303062022041001



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127

Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

Form. KP-002

LOG SHEET MINGGUAN KERJA PRAKTIK

Nama Mahasiswa : Norbertino Eurakha Nandatoti 10231071
Program Studi : Sistem Informasi
Jurusan : Jurusan Teknik Elektro Informatika dan Bisnis
Fakultas : Fakultas Sains dan Teknologi Informasi
Dosen Pembimbing : Henokh Lugo Hariyanto, S.Si.,M.Sc.
Mitra KP : PT. Bintang Indayu Gemilang
Periode KP : Bulan Januari - Februari, Tahun 2026
Minggu ke- : 5

Hari/Tanggal	Uraian Aktivitas	Jam Datang	Jam Pulang	Paraf Pembimbing Lapangan
02/02/2026	Mengikuti Morning Core Values dan menyusun Planning Iterasi 2 sebagai kelanjutan proyek. Aktivitas utama difokuskan pada finalisasi Use Case dan pembaruan Product Backlog agar lebih akurat. Selain itu, dilakukan koordinasi dengan tim GTM untuk mendalami SB7 Framework (StoryBrand) menggunakan studi kasus film, serta menerima umpan balik krusial mengenai mekanisme iterasi untuk laporan KP.			
03/02/2026	Melakukan wawancara dengan Stakeholder untuk mendapatkan umpan balik atas	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	progres Iterasi Pertama, diikuti dengan diskusi bersama Incha untuk merumuskan langkah perbaikan Aktivitas teknis difokuskan pada implementasi fitur Management Tools (CRUD), mulai dari UI, integrasi API (GET, POST, PUT, DELETE), hingga validasi formulir. Meskipun sempat terhambat pemadaman listrik akibat hujan deras, progres telah dipindahkan ke tahap Review. Selain itu, dilakukan sesi bersama tim GTM untuk mendalami implementasi praktis SB7 Framework di lingkungan perusahaan.			
04/02/2026	Melakukan perencanaan dan perancangan UI untuk fitur Operation List. Aktivitas teknis difokuskan pada desain antarmuka serta optimasi fitur Management Tools dengan mengimplementasikan pengurutan (sorting) berdasarkan abjad	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	secara ascending (A - C).			
05/02/2026	Melakukan perbaikan pada UI Operation List setelah sesi diskusi desain bersama Incha dan pengumpulan umpan balik dari Stakeholder. Aktivitas teknis difokuskan pada konfigurasi basis data melalui pembuatan tabel Operation dan Operation Enroll, serta implementasi fitur pencarian menggunakan mock data untuk pengujian antarmuka. Selain itu, dilakukan pertemuan pendalaman mengenai Storybrand - Brandscript untuk memperkuat narasi produk.	08:30 WITA	17:00 WITA	
06/02/2026	Melakukan penyempurnaan detail kecil pada fitur User Management serta memulai penyusunan Laporan Magang hingga Bab 2. Aktivitas utama difokuskan pada riset jurnal dan artikel ilmiah sebagai dasar teori,	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127

Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	sembari memecahkan kebingungan terkait struktur penulisan literatur dan kaidah penulisan laporan formal.			
--	--	--	--	--

Catatan Dosen Pembimbing:

Balikpapan,
Dosen Pembimbing

Henokh Lugo Hariyanto,
S.Si., M.Sc.
NIP 199303062022041001



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

Form. KP-002

LOG SHEET MINGGUAN KERJA PRAKTIK

Nama Mahasiswa : Norbertino Eurakha Nandatoti 10231071
Program Studi : Sistem Informasi
Jurusan : Jurusan Teknik Elektro Informatika dan Bisnis
Fakultas : Fakultas Sains dan Teknologi Informasi
Dosen Pembimbing : Henokh Lugo Hariyanto, S.Si., M.Sc.
Mitra KP : PT. Bintang Indayu Gemilang
Periode KP : Bulan Januari - Februari, Tahun 2026
Minggu ke- : 6

Hari/Tanggal	Uraian Aktivitas	Jam Datang	Jam Pulang	Paraf Pembimbing Lapangan
09/02/2026	Melakukan peninjauan mendalam terhadap tugas-tugas yang tertunda dari iterasi sebelumnya untuk mengidentifikasi fitur yang belum selesai dan minor bugs. Aktivitas utama difokuskan pada penyusunan Perencanaan Iterasi 3, dengan mengintegrasikan seluruh pending tasks dan fitur baru ke dalam roadmap pengembangan yang lebih komprehensif.	08:30 WITA	17:00 WITA	
10/02/2026	Melakukan refaktorisasi pada fitur Operation untuk meningkatkan kebersihan kode serta mengoptimalkan struktur	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	components dan composables agar lebih modular. Aktivitas teknis difokuskan pada pengembangan fitur Field Reporting melalui pembaruan skema basis data, termasuk penambahan tabel untuk lampiran laporan (report attachment) dan dokumentasi teknis.			
11/02/2026	Melanjutkan optimasi basis data untuk fitur Field Reporting dengan fokus pada integritas data dan efisiensi relasi antar tabel. Aktivitas teknis difokuskan pada penyambungan tabel pelaporan ke tabel operasional serta analisis mendalam terhadap keterkaitan antara dokumentasi joblist	08:30 WITA	17:00 WITA	
12/02/2026	Melakukan inisiasi Iterasi 3 dengan mengadopsi metodologi Test-Driven Development (TDD) untuk meningkatkan kualitas kode. Aktivitas utama difokuskan pada riset mendalam mengenai siklus TDD	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

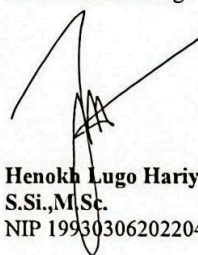
Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	(Red-Green-Refactor) serta penyusunan dokumentasi User Acceptance Testing (UAT) sebagai tolok ukur keberhasilan fitur dari perspektif pengguna.			
13/02/2026	Gemini berkata Melakukan pendalaman perencanaan untuk fitur Field Reporting dengan menyusun templat laporan yang terstruktur. Aktivitas teknis difokuskan pada persiapan library pendukung serta peninjauan ulang (review) terhadap Perencanaan Iterasi 3 untuk memastikan seluruh tugas yang tertunda (delayed tasks) telah terakomodasi dengan baik.	08:30 WITA	17:00 WITA	

Catatan Dosen Pembimbing:

Balikpapan,
Dosen Pembimbing


Henokh Lugo Hariyanto,
S.Si., M.Sc.
NIP 199303062022041001



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801
Email: humas@itk.ac.id

Form. KP-002

LOG SHEET MINGGUAN KERJA PRAKTIK

Nama Mahasiswa : Norbertino Eurakha Nandatoti 10231071
Program Studi : Sistem Informasi
Jurusan : Jurusan Teknik Elektro Informatika dan Bisnis
Fakultas : Fakultas Sains dan Teknologi Informasi
Dosen Pembimbing : Henokh Lugo Hariyanto, S.Si., M.Sc.
Mitra KP : PT. Bintang Indayu Gemilang
Periode KP : Bulan Januari - Februari, Tahun 2026
Minggu ke- : 7

Hari/Tanggal	Uraian Aktivitas	Jam Datang	Jam Pulang	Paraf Pembimbing Lapangan
16/02/2026	Libur			
17/02/2026	Melakukan finalisasi Perencanaan Iterasi 3 dengan fokus utama pada stabilitas sistem dan keamanan akses. Aktivitas teknis difokuskan pada penyelesaian masalah build aplikasi, pembersihan kode lama (dead code), serta pengorganisasian ulang halaman Operation agar lebih mudah dikelola. Selain itu, dilakukan penguatan arsitektur melalui pemisahan kode client-side dan server-side yang lebih ketat, serta pembaruan sistem	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	Role-Based Access Control.			
18/02/2026	elakukan sesi bimbingan bersama Dosen Pembimbing (Lecture) untuk meninjau progres dan arah pengembangan proyek. Aktivitas teknis difokuskan pada refaktorisasi fitur Operation, penyempurnaan diagram skema basis data agar lebih representatif, serta riset mengenai metode pembuatan dokumen PDF dari struktur HTML.	08:30 WITA	17:00 WITA	
19/02/2026	Melakukan refaktorisasi pada fitur Operation untuk meningkatkan efisiensi kode, keterbacaan, dan skalabilitas sistem.	08:30 WITA	17:00 WITA	
20/02/2026	Melakukan penguatan pada fitur operasional dengan implementasi Checklist Photo Upload, yang mencakup fungsi pratinjau (preview) dan penghapusan foto secara real-time. Aktivitas teknis difokuskan pada integrasi daftar alat langsung dari basis data, penguatan validasi	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127

Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	pada aksi save/update/delete, serta penanganan edge-cases untuk meningkatkan stabilitas aplikasi. Selain itu, dilakukan optimasi pada struktur komponen agar lebih reusable, sembari memecahkan masalah terkait berkas migrasi/riwayat dan bug kueri pada kasus data spesifik.			
--	---	--	--	--

Catatan Dosen Pembimbing:

Balikpapan,
Dosen Pembimbing

Henokh Lugo Hariyanto,
S.Si.,M.Sc.
NIP 199303062022041001



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127

Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

Form. KP-002

LOG SHEET MINGGUAN KERJA PRAKTIK

Nama Mahasiswa : Norbertino Eurakha Nandatoti 10231071
Program Studi : Sistem Informasi
Jurusan : Jurusan Teknik Elektro Informatika dan Bisnis
Fakultas : Fakultas Sains dan Teknologi Informasi
Dosen Pembimbing : Henokh Lugo Hariyanto, S.Si.,M.Sc.
Mitra KP : PT. Bintang Indayu Gemilang
Periode KP : Bulan Januari - Februari, Tahun 2026
Minggu ke- : 8

Hari/Tanggal	Uraian Aktivitas	Jam Datang	Jam Pulang	Paraf Pembimbing Lapangan
23/02/2026	Menyelesaikan dokumentasi UAT dan Unit Testing untuk menjamin validitas alur kerja riil. Fokus utama pada penyelesaian fitur End-to-End Reporting (Formulir, Lampiran, Validasi, hingga ekspor PDF) serta perbaikan antarmuka Tool Picker. Meski alur fungsional sudah mantap, masih terdapat kendala teknis pada sinkronisasi migrasi database dan ketidaksesuaian penamaan kolom data.	08:30 WITA	17:00 WITA	
24/02/2026	Melakukan optimasi pada sistem Filtering untuk meningkatkan pengalaman	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	pengguna dalam mengelola data. Aktivitas teknis difokuskan pada pembaruan UI filter (tanggal, tipe, status) serta sinkronisasi state filter dengan URL agar navigasi lebih presisi.			
25/02/2026	Melakukan UI Polish dengan fokus pada responsivitas mobile dan fungsionalitas komponen, termasuk perbaikan dialog, sidebar berbasis Role, serta optimalisasi gambar pada PDF. Berhasil meningkatkan loading state per tombol, namun masih terkendala masalah layout overlap dan respons sentuhan (tap issues) pada perangkat tertentu.	08:30 WITA	17:00 WITA	
26/02/2026	Melakukan pemolesan akhir pada antarmuka fitur pelaporan dan manajemen pengguna untuk meningkatkan efisiensi kerja. Fokus utama adalah menghadirkan In-page Report Preview sehingga pengguna tidak	08:30 WITA	17:00 WITA	



**KEMENTERIAN PENDIDIKAN TINGGI,
SAINS DAN TEKNOLOGI
INSTITUT TEKNOLOGI KALIMANTAN**

Kampus ITK Karang Joang, Balikpapan 76127
Telp. 0542-8530800 Fax. 0542-8530801

Email: humas@itk.ac.id

	perlu berpindah halaman, serta menjaga agar bagian checklist tetap terbuka otomatis setelah proses simpan guna mempercepat pengisian data berulang.			
27/02/2026	Melakukan finalisasi pada aspek Responsivitas dan Aksesibilitas aplikasi untuk memastikan tampilan optimal di berbagai perangkat, mulai dari ponsel hingga tablet (medium-screen). Aktivitas utama juga difokuskan pada persiapan akhir masa magang, termasuk penyusunan materi demonstrasi SAFED dan slide presentasi final di bawah bimbingan mentor.	08:30 WITA	17:00 WITA	

Catatan Dosen Pembimbing:

Balikpapan,
Dosen Pembimbing



Henokh Lugo Hariyanto,
S.Si.,M.Sc.
NIP 199303062022041001

Photos of Field Work Activities,







Practical Work Completion Certificate,