Final Project Dependency Free Information System

PROJECT REPORT

POLITEKNIK NEGERI MALANG
Jl. Soekarno Hatta no. 9, Malang, 65141 polinema.ac.id
VERSION 1.0.0 01/12/2024

VERSION HISTORY

VERSION	APPROVED BY	REVISION DATE	DESCRIPTION OF CHANGE	AUTHOR
1.0.0	Project Sponsor	Dec 12, 2024	Initial Version	Project Manager

PREPARED BY	TITLE	DATE
Muhammad Tegar.H		09/12/2024

APPROVED BY	TITLE	DATE

TABLE OF CONTENTS

1. PROJECT OVERVIEW

SIBATTA (Sistem Informasi Bebas Tanggungan Tugas Akhir) This project aims to implement a system where final-year students (D3, D4, and S2) of Politeknik Negeri Malang (Polinema) can upload their final project reports (Laporan Akhir, Skripsi, Tesis) to the Polinema Library website. The project faced several changes during execution, including team member replacement, delays in integration, and budget adjustments. Despite these challenges, the project was successfully completed, delivering a system that met stakeholder expectations.

Project Name	**SIAKAD Development Project**
Start Date	September 2, 2024
End Date	December 12, 2024 (adjusted after delays)
Project Manager	Muhammad Tegar Hibatulloh
Budget	Rp 2,150,000 (adjusted after hosting upgrade)
Status	Successfully Completed

2. OBJECTIVES AND DELIVERABLES

Objective	**Status**	**Comments**
Develop a responsive file submission system	Achieved	Fully functional and responsive.
Automate debt verification and clearance letter issuance	Achieved	Includes notification features for users.
Deliver secure authentication and data storage	Achieved	Implemented using Laravel PHP framework and MySQL database.
Provide training and user documentation	Achieved	Training sessions completed; manuals delivered.

3. KEY CHANGES DURING PROJECT EXECUTION

- **Delay in Backend Integration:**
  - Resolved through additional testing and close collaboration between teams.
- **Budget Adjustment:**
  - Hosting plan upgraded for improved system performance, increasing the budget by Rp 100,000.
- **Feature Enhancement:**
  - Added user notification features based on stakeholder feedback.

4. PROJECT PERFORMANCE

Metric	Planned	Actual
Timeline	September 2 - December 11, 2024	September 2 - December 12, 2024
Budget	Rp 5,500,000	Rp 2,150,000
Scope Completion	100%	100%
Key Risks Addressed	3 critical risks mitigated	No significant impact on deliverables.

5. KEY ACCOMPLISHMENTS

Requirement Gathering and Analysis:

- Successfully gathered and documented requirements from all stakeholders, including students, library staff, and administrators.
- Developed clear use-case scenarios to guide system design.

System Design and Architecture:

- Completed a scalable and maintainable three-tier system architecture.
- Designed user-friendly interfaces using Figma, ensuring responsive designs across devices.

Development and Implementation:

- Developed a secure file submission system using Laravel and MySQL.
- Implemented automated debt validation and notification features to streamline administrative processes.
- Created role-based access control to ensure secure operations for different user groups (students, staff, administrators).

Integration and Testing:

- Conducted thorough integration testing to ensure seamless communication between frontend and backend components.
- Completed Unit Testing and User Acceptance Testing (UAT) with over 95% positive feedback.

Training and Documentation:

- Delivered comprehensive user manuals and conducted training sessions for library staff and administrators.
- Provided clear documentation for system maintenance and updates.

Deployment and Handover:

- Successfully deployed the system on a robust hosting platform (Niagahoster).
- Transitioned system credentials, access, and supporting documentation to the IT department.

Stakeholder Satisfaction:

- Achieved high satisfaction levels from all primary stakeholders, with students appreciating the ease of use and staff noting operational efficiency improvements.

6. SYSTEM ARCHITECTURE

The system architecture of SIAKAD follows a three-tier model, which ensures scalability, maintainability, and efficient separation of concerns:

- **Architecture Layers**
  - **Presentation Layer (Frontend):**
    - Technologies: HTML, CSS, JavaScript, and Bootstrap.
    - Purpose: Provides a user-friendly and responsive interface for students, library staff, and administrators to interact with the system.
    - Key Features:
      - Cross-browser compatibility.
      - Responsive design for seamless use on desktops, tablets, and mobile devices.
      - Forms for file uploads, notifications, and status tracking.
  - **Application Layer (Backend):**
    - Technologies: Laravel Framework (PHP).
    - Purpose: Handles all server-side logic, business rules, and API integrations.
    - Key Features:
      -Automated debt validation processes.
      - Secure user authentication and role-based access control.
      - RESTful APIs for communication between the frontend and backend.
      - Notification services for submission updates.
  - **Data Layer (Database):**
    - Technologies: SSMS.
    - Purpose: Stores and manages data securely, including user profiles, uploaded files, and clearance statuses.
    - Key Features:
      - Optimized queries for efficient data retrieval.
      - Data integrity measures to ensure accuracy.
      - Regular backups to prevent data loss.

- **Data Flow**
- Users interact with the system through the frontend, submitting file uploads or checking clearance statuses.
- Requests are sent from the frontend to the backend via secure RESTful APIs.
- The backend processes the requests, performing tasks like debt validation, data retrieval, or notification triggers.
- The backend retrieves or stores the necessary data in the MySQL database.
- The processed data is sent back to the frontend, displaying results or updates to the user.

7. TECHNOLOGIES USED

Category	Technology	Purpose
Frontend Development	HTML, CSS, Bootstrap	Creates responsive and interactive user interfaces.
Backend Development	Laravel Framework	Server-side logic and APIs
Database	SSML	Stores user data, secure data storage
Hosting and Deployment	Niagahoster	Provides cloud hosting and domain registration.
Security Testing	OWASP ZAP	Identify and mitigate vulnerabilities

8. APPLICATION FEATURES

- **File Uploads:** Students can submit their final project reports.
- **Debt Validation:** Automates the process of checking financial obligations.
- **Notifications:** Provides real-time updates on submission status.
- **Administrative Tools:** Allows staff to manage and validate student clearances

9. USER INTERFACE DESIGN

- **Tools Used:** Designed in Figma.
- **Screens:** Includes login page, student dashboard, and admin panel.
- **Accessibility:** Fully responsive across devices.

10. TESTING AND QUALITY ASSURANCE

Testing Type	Outcome
Unit Testing	Passed all cases
Integration Testing	Resolved minor issues
User Acceptance	95% positive feedback

11. SECURITY MEASURES

- **Authentication:**
  - Role-based access control implemented using Auth0 ensures that users only access authorized parts of the system.
- **Encryption:**
  - SSL/TLS encryption is used to secure data exchanges between users and the server, protecting sensitive information during communication.
- **Vulnerability Scanning:**
  - Tools like OWASP ZAP were used to identify and mitigate vulnerabilities such as SQL injection and cross-site scripting (XSS).
- **Database Security:**
  - Implemented parameterized queries to prevent SQL injection attacks and ensured regular database backups for data recovery in case of failure.
- **Testing and Monitoring:**
  - Continuous monitoring of system logs to detect anomalies and post-deployment security audits scheduled monthly.

12. CHALLENGES AND LESSONS LEARNED

Challenges:

- **Initial Hosting Limitations:** The initial hosting infrastructure proved inadequate for the system's performance requirements.
- **Coordination Issues During Integration:** Misalignment between frontend and backend teams delayed the project by a week.

Lessons Learned:

- **Buffer Time Allocation:** Including extra time in the schedule for unexpected delays ensures smoother execution.
- **Stakeholder Engagement:** Regular involvement of stakeholders helps in minimizing delays during validation phases.

13. FINANCIAL SUMMARY

Cost Component	Planned Cost (Rp)	Actual Cost (Rp)	Variance
UI/UX Design	600,000	400,000	0
Project Manager	500,000	350,000	0
Frontend Development	1,000,000	500,000	0
Backend Development	1,000,000	500,000	0
QA Testing	1,500,000	0	0
Miscellaneous	900,000	400,000	0
Total	**5,500,000**	**2,150,000**	**0**

14. STAKEHOLDER FEEDBACK

Library Staff: "The system significantly reduces manual workloads."

Students: "User-friendly and efficient process for project submissions."

15. TRASNSITION AND HANDOVER

- System credentials and documentation were handed over to the IT department to ensure smooth operations post-deployment.
- Comprehensive user and admin manuals were provided and stored in the institutional repository for future reference.
- The project team committed to providing post-deployment support for one month to address any initial issues.

16. RECOMMENDATIONS

- **Regular Updates:** Perform periodic updates to maintain system performance and security.
- **User Feedback:** Monitor user suggestions and implement enhancements for improved functionality.
- **Testing Resources:** Allocate additional resources and time for testing phases in future projects to ensure quality and reduce risks.

APPROVAL

Name	**Title**	**Signature**
Muhammad Tegar. H	Project Manager	
M. Hasyim Ratsanjani, S.Kom., M.Kom.	Lecture Pemograman Berbasis Objek	
Wilda Imama Sabilla, S.Kom., M.Kom.	Lecture Desain dan Pemograman Web	
Pramana Yoga Saputra, S.Kom., M.MT.	Lecture Manajemen Proyek	
Yoppy Yunhasnawa, S.ST., M.Sc.	Lecture Advanced Database	