

Design and Implementation of a Web-Based Cadre Data Management Information System for HMI Tembilahan Branch

Moh Idris^{1*}, Abdul Muni², Samsudin³

^{1,2,3}Department of Information Systems, Universitas Islam Indragiri, Indragiri Hilir, Riau, Indonesia
idrisdaxter@gmail.com^{1*}, abdulmuni@gmail.com², samsudinsadek@gmail.com³

Abstract

This study designs and implements a web-based information system for managing cadre data at the Himpanan Mahasiswa Islam (HMI) Tembilahan Branch. The study responds to handwritten and fragmented administrative practices that slow data retrieval, increase duplication risk, and weaken information control. Data were collected through direct observation at the branch secretariat, an interview with the branch chairman, and a literature review. System requirements were analyzed using the PIECES framework, while development followed the Waterfall model and was represented through Unified Modeling Language diagrams. The application was implemented with Laravel and MySQL and provides role-based access for administrators and the HMI chairman, covering member, commissariat, announcement, activation, position, and profile management. Functionality testing involved 25 respondents and 25 system functions, producing 625 successful executions and no failures, equivalent to a functionality value of 1. Usability testing with ten statements produced 1,095 of 1,250 possible points, or 87.64%, categorized as very good. The results indicate that the system is functionally reliable, usable, and suitable for supporting more organized cadre administration at HMI Tembilahan Branch.

Keywords: Cadre data; HMI; Information system; Laravel; Web application

1. Introduction

Information systems are increasingly used to consolidate organizational records, shorten administrative cycles, and provide reliable information for managerial decisions. A well-designed management information system does not merely convert paper records into digital files. It reorganizes data flows, access rights, and reporting procedures so that information can be used consistently across organizational levels [3][4]. In organizations whose membership changes frequently, the need for a centralized database becomes more pronounced because duplicated, incomplete, or outdated records can directly affect coordination and accountability.

Himpunan Mahasiswa Islam (HMI) Tembilahan Branch manages cadre data originating from branch and commissariat activities. Before this study, the principal records were still handled through handwritten or separate manual documents. Registration data, membership status, organizational position, announcements, and commissariat information were therefore difficult to retrieve quickly. The process also depended heavily on the administrator who kept the documents, creating a practical risk of data loss, inconsistent updates, and delayed reporting.

A web-based application was selected because it enables centralized processing through a browser and can be accessed from desktop or mobile devices without installing a dedicated client. Laravel offers a structured web-development framework, while MySQL provides relational storage suited to connected records such as users, members, positions, news, and commissariats [2][7][9][14]. The combination also supports authentication and role-based authorization, two controls that are essential when organizational data should not be available to every user in the same manner.

Structured software development is required to prevent the application from becoming a collection of disconnected features. Software engineering literature emphasizes the importance of sequential requirements analysis, design, implementation, and verification, particularly when user needs are first translated into process and data models [1][5][6][10]. For this reason, the study used the Waterfall model, PIECES analysis, and Unified Modeling Language (UML) to connect the observed administrative problems with the final system functions.

Previous web-based membership and student-organization systems summarized in the thesis generally reported improvements in data centralization, searching, activity documentation, and report preparation. The specific administrative structure of HMI Tembilahan, however, requires a distinction between branch administration and commissariat information, member activation, organizational positions, announcements, and the limited monitoring access of the branch chairman. The contribution of this study lies in translating those domain-specific activities into a role-based application and evaluating the resulting system through functionality and usability tests.

Accordingly, this study aims to design, implement, and evaluate a web-based cadre data management information system for HMI Tembilahan Branch. The evaluation focuses on whether all specified functions operate correctly and whether prospective users consider the system useful, understandable, learnable, and satisfactory.

2. Research method

2.1. Research object and data collection

The research was conducted at the HMI Tembilahan Branch Secretariat on Budiman Street, Silaturahmi 2 Alley, Tembilahan Kota. Primary information was obtained through direct observation of the existing data-management process and a face-to-face interview with the chairman of HMI Tembilahan Branch. Observation was used to identify how cadre records were created, stored, searched, and updated, while the interview clarified user roles, expected functions, and administrative constraints. A literature study was also conducted to examine software-development concepts and previous web-based data-management applications. The collected qualitative information was interpreted narratively to identify recurring problems and convert them into system requirements.

2.2. Analysis and system development

The system was developed using the Waterfall model. The first stage identified functional requirements and analyzed the existing process with PIECES, which evaluates performance, information, economy, control, efficiency, and service [13]. The second stage translated the requirements into a design using use case, activity, sequence, and class diagrams. UML was applied to represent user interaction, workflow, message order, and data structure before coding [1][8]. The third stage implemented the application using the Laravel framework and a MySQL relational database. The fourth stage tested the completed functions and collected user assessments. Figure 1 summarizes the research workflow.

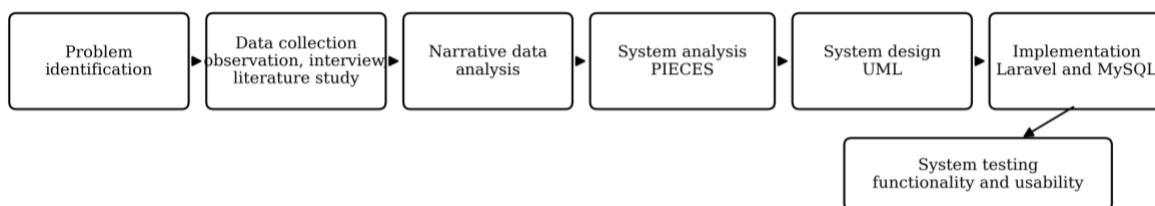


Fig. 1: Research workflow used to analyze, design, implement, and evaluate the system

2.3. System testing

Functionality was evaluated through black-box testing, which observes whether each input and operation produces the expected output without examining the internal source code. Twenty-five respondents consisting of HMI administrators and cadres tested 25 functions according to their access rights. The functionality value was calculated as $X = 1 - A/B$, where A is the number of failed executions and B is the total number of executed test scenarios. Usability was measured through ten Likert-scale statements covering usefulness, ease of use, ease of learning, and satisfaction. Each item used a score of 1 to 5, giving a maximum total of 1,250 points for 25 respondents. User-oriented evaluation is important because technical correctness alone does not guarantee that an application will be understandable or comfortable in routine work [11][12][15].

2.4. System requirements and research scope

The requirements were derived from the manual administrative problems documented during observation and interview. They were organized according to actor, information object, and expected output so that each requirement could be traced to a use case and later verified during black-box testing. The scope remained limited to HMI Tembilahan Branch, web-browser access, a relational database, and two principal user roles. Native Android and iOS applications, wider inter-branch integration, and long-term maintenance were outside the reported implementation.

Table 1: Functional requirements derived from the research object

Code	Actor	Requirement
FR-01	Administrator and chairman	Authenticate using a registered username and password and display a role-appropriate dashboard.
FR-02	Administrator	View the administrative dashboard and use the main navigation menu.
FR-03	Administrator	Add, view, update, and delete commissariat information.

Code	Actor	Requirement
FR-04	Administrator	Add, view, update, and delete news or announcements.
FR-05	Administrator	Add, view, search, update, and delete cadre or member records.
FR-06	Administrator	Activate eligible members and preserve the current membership status.
FR-07	Administrator	Manage organizational positions used for cadre placement.
FR-08	Administrator	Add a new commissariat with the required identity and contact information.
FR-09	Administrator	View and update the administrator profile and password.
FR-10	HMI chairman	View announcements published by the administrator without editing them.
FR-11	HMI chairman	View registered commissariats and their contact information.
FR-12	HMI chairman	View and update the personal profile and password.
FR-13	Both roles	Terminate the authenticated session through the logout function.

The non-functional boundaries required the application to operate through common web browsers on desktop and mobile devices, use PHP-based web technology with Laravel, store connected records in MySQL, and restrict access through authentication and differentiated permissions. The thesis also planned compatibility checks across Windows, Android, iOS, Chrome, Firefox, Safari, and Microsoft Edge, although the reported numerical results in Chapter IV concentrate on functionality and usability. Consequently, this article does not claim a quantified compatibility outcome that is absent from the source document.

3. Results and discussion

3.1. PIECES analysis

The comparison between the previous manual process and the proposed web-based process is summarized in Table 2. The analysis shows that the main weakness was not a single missing feature, but the absence of an integrated administrative mechanism. Each PIECES dimension therefore informed a corresponding system response.

Table 2: PIECES comparison between the previous and proposed systems

Aspect	Previous system	Proposed system
Performance	Recording, searching, and reporting require a long time because documents are handled manually.	Computerized processing shortens data entry, searching, updating, and reporting.
Information	Records may be incomplete, duplicated, or inconsistent because no centralized database is used.	Data are stored in a relational database and can be updated and retrieved consistently.
Economy	Paper, stationery, printing, and physical storage increase recurring administrative costs.	Digital storage reduces paper use and physical archiving requirements.
Control	Documents may be lost, damaged, or accessed by unauthorized persons.	Authentication and differentiated access rights improve control over organizational records.
Efficiency	The same data may be written repeatedly and reports require repeated manual compilation.	Data are entered once and reused for searching, monitoring, and administrative needs.
Service	Administrative responses are slow because information must be searched manually.	Users receive faster access to cadre, commissariat, and announcement information.

The proposed design addresses the performance problem by reducing repetitive work, but its effect also extends to control and service. Centralized storage allows a member record to be updated once and used across administrative views. Authentication limits access, while the chairman receives monitoring functions without being granted full editing authority. This division reduces the possibility that oversight activities unintentionally alter operational data.

3.1.1. Performance and information implications

Under the previous procedure, the duration of an administrative task depended on the location and completeness of physical records. Searching for one cadre could require checking handwritten lists or separate files, while a report could only be prepared after the data were compiled again. The proposed application changes this pattern by placing cadre, commissariat, announcement, position, and activation data in structured modules. Searching and updating can therefore be performed from the same interface rather than recreating the record for every administrative need.

The information dimension is equally important because speed without consistency would merely produce errors more quickly. A centralized relational database provides one maintained source for recurring data objects. The member page, activation page, and position page serve different administrative purposes but remain connected through the implemented data structure. This arrangement reduces the likelihood that an active status, organizational position, or commissariat affiliation is maintained in an unrelated document.

3.1.2. Economy and control implications

The economic improvement reported in the analysis concerns operational media rather than a calculated financial return. Digital records reduce dependence on paper, ink, stationery, and physical storage. The thesis does not report a monetary cost comparison, so the benefit should be interpreted as a reduction in recurring manual resources and duplicated administrative effort, not as a quantified saving.

Control is implemented through login validation and role separation. The administrator can create and modify operational records, whereas the chairman receives read-oriented access to announcements and commissariat information. Profile management remains available to both roles for their own accounts. This distribution is consistent with the source use case diagram and provides a basic access boundary, although the thesis recommends stronger security in future development through backup, encryption, and two-factor authentication.

3.1.3. Efficiency and service implications

Efficiency emerges from data reuse. Once a cadre record has been entered, it can be searched, updated, activated, and associated with an organizational position without rewriting the complete identity in several manual formats. The same principle applies to commissariat information and announcements. The system therefore shifts administrative effort from repeated transcription toward validation and maintenance of existing records. Service improvement is reflected in the ability of authorized users to retrieve organizational information directly from the application. The administrator can respond to data requests through the relevant module, while the chairman can independently inspect published announcements and commissariat data. The application does not remove the need for administrative judgment, but it reduces the waiting time created by manual searching and fragmented storage.

3.2. Functional and data design

Two actors interact with the application. The administrator manages operational records, while the HMI chairman primarily monitors organizational information and maintains a personal profile. Table 3 describes the access boundary, and Figure 2 presents the use case model.

Table 3: User roles and principal system functions

Role	Access and functions
Administrator	Login and dashboard access; manage commissariats; manage news and announcements; add, update, search, and delete members; activate membership; manage organizational positions; add commissariats; update profile; logout.
HMI chairman	Login and dashboard access; view announcements; view commissariat information; update personal profile and password; logout.

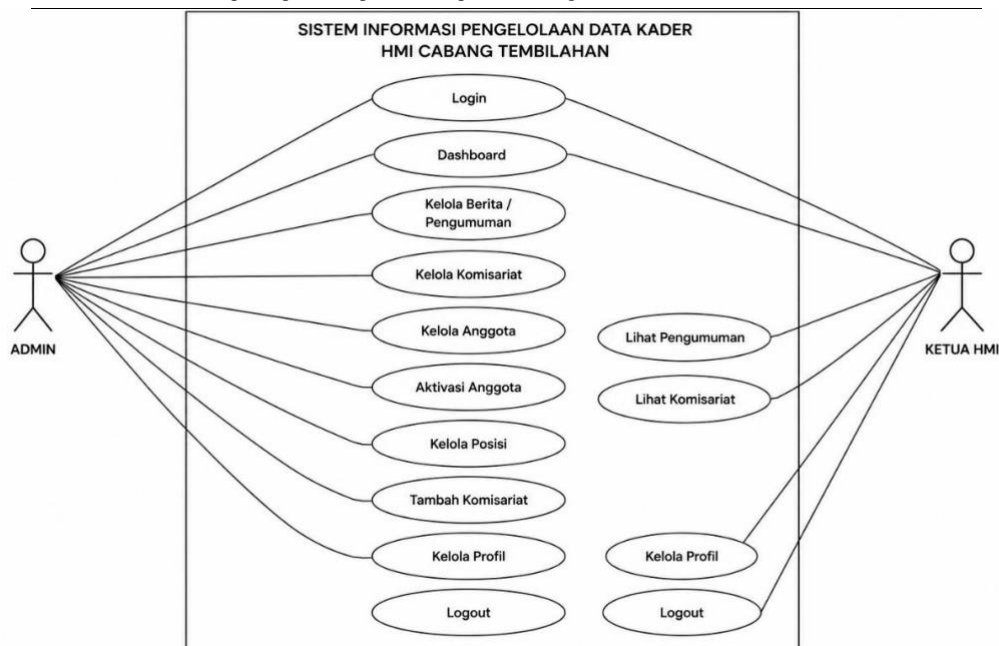


Fig. 2: Use case diagram of the HMI cadre data management information system

The use case model shows a deliberate separation between full data-management authority and monitoring access. The administrator controls record creation and modification, whereas the chairman can read announcements and commissariat information. This design supports accountability while maintaining data integrity. The login workflow in Figure 3 further illustrates that system access depends on credential validation before the appropriate dashboard is displayed.

3.2.1. Activity and interaction design

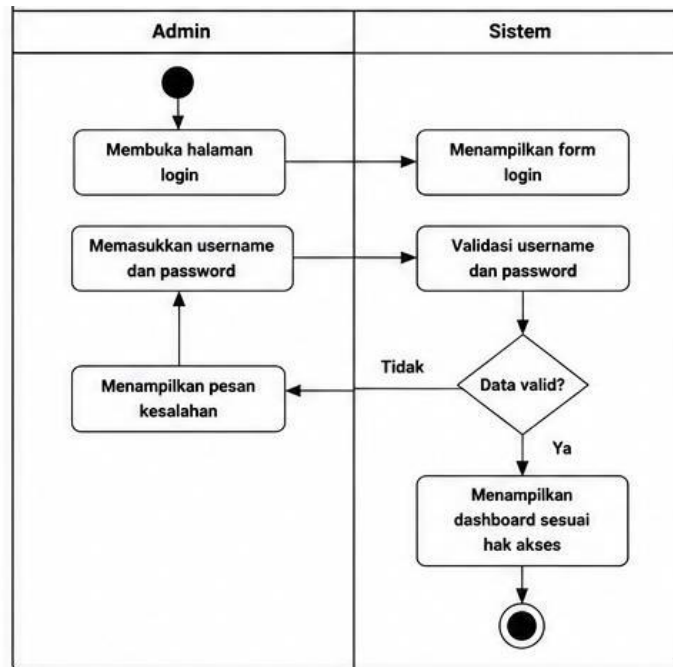


Fig. 3: Administrator login activity diagram

At the interaction level, the login sequence begins when a user submits a username and password. The login controller validates the credentials against the database and returns either the authorized dashboard or an error notification. Figure 4 shows the sequence of messages among the actor, interface, controller, and database.

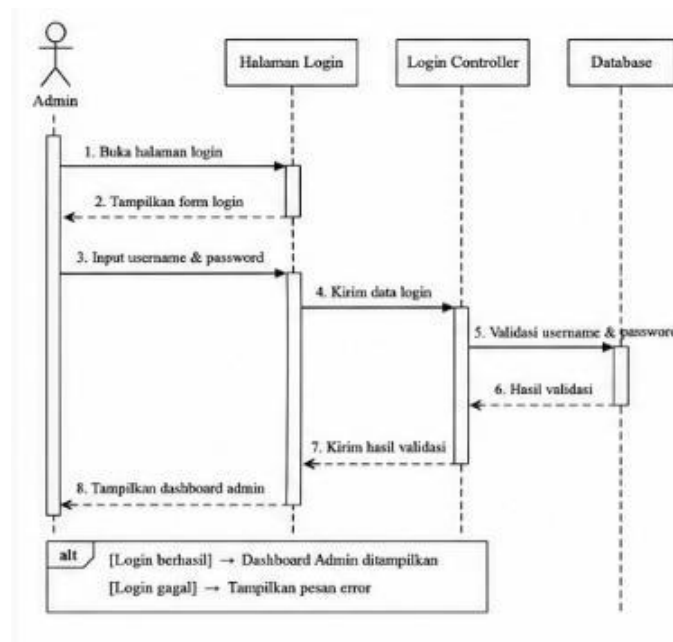


Fig. 4: Administrator login sequence diagram

Login is only one of the workflows represented in the design. The management activities in Figure 5 show a recurring control pattern. The administrator selects a module, the system displays existing records, the administrator chooses an add, update, or delete action, and the

system validates and stores the change before returning an updated list. This pattern is applied to commissariat and member data and creates a consistent interaction model across administrative modules.

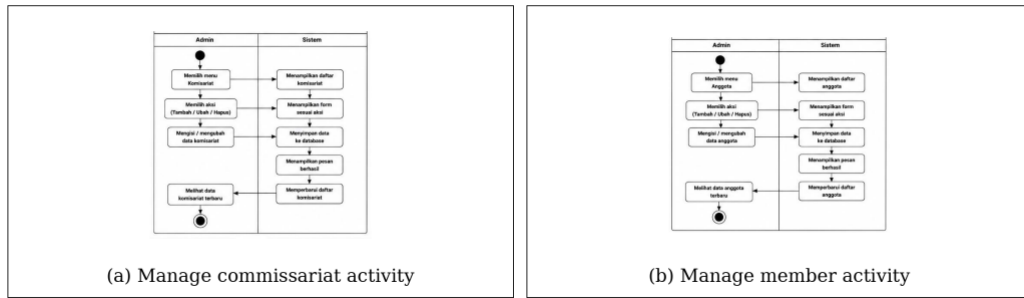


Fig. 5: Activity diagrams for commissariat and member management

The corresponding sequence diagrams in Figure 6 explain how the visible workflow is executed by the interface, controller, and database. The page requests the current data through its controller, the controller retrieves the records from the database, and the updated list is returned after an authorized operation. Separating page presentation from the controller and database operations is consistent with the framework-based implementation reported in the thesis.

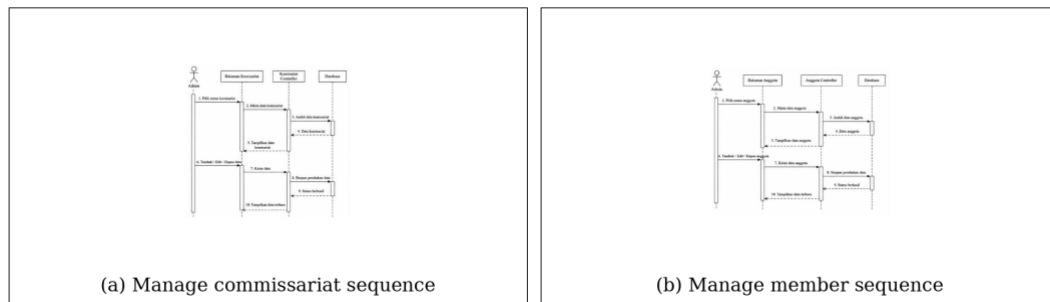


Fig. 6: Sequence diagrams for commissariat and member management

3.2.2. Data structure

The class structure contains interconnected entities for users, members, commissariats, news, activation, positions, and related profile data. Relationships among these entities form the basis for database implementation, as shown in Figure 7. The relational design prevents major modules from being stored as isolated documents and supports consistent references across organizational records.

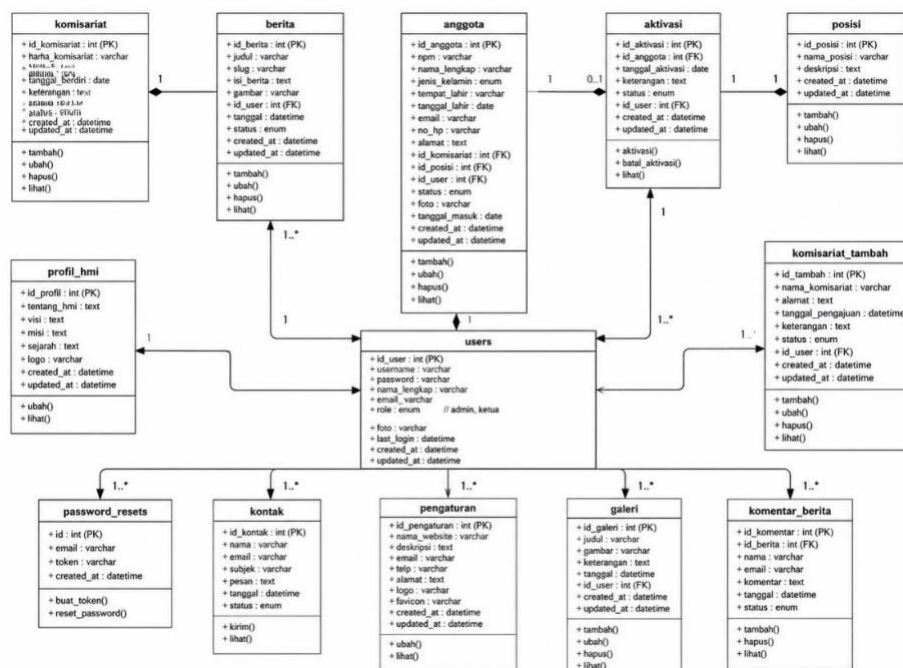


Fig. 7: Class diagram of the implemented information system

Table 4: Principal data objects represented in the design

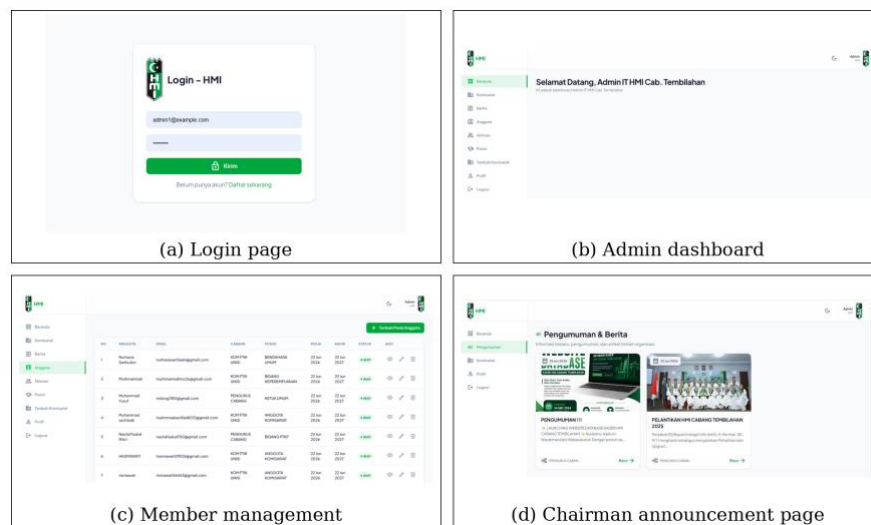
Data object	Role in the system
User account	Stores authenticated-user information and supports role-based access.
Member or cadre	Stores personal, organizational, and membership information for HMI cadres.
Commissariat	Stores the identity, address, telephone number, e-mail, and related branch-unit information.
News or announcement	Stores information published by the administrator and viewed by authorized users.
Activation	Represents the membership-activation process and current status.
Position	Stores organizational positions used in cadre or management placement.
Profile data	Stores account-related personal information that can be updated by the corresponding user.

The class diagram in the thesis contains additional technical classes and relationships beyond this summary. Table 4 therefore presents the principal business objects rather than replacing the complete model. Together, the objects support the separation of authentication, cadre records, organizational units, information publication, membership status, and organizational structure.

3.3. System implementation

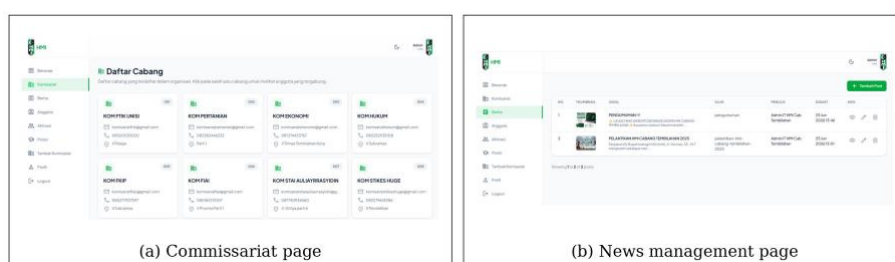
The application was implemented with Laravel and MySQL. After authentication, the administrator dashboard provides navigation to commissariat, announcement, member, activation, position, add-commissariat, profile, and logout functions. The chairman dashboard presents a more limited navigation structure containing the homepage, announcements, commissariats, profile, and logout. The interface sequence below follows the way the application is used, beginning with authentication and continuing through operational and monitoring modules.

3.3.1. Authentication and administrator dashboard

**Fig. 8:** Authentication, administrator dashboard, member management, and chairman announcement examples

The login page is the single entry point for both user roles. Valid credentials direct the user to the dashboard associated with the stored access right. The administrator dashboard acts as the operational control panel and exposes the complete management menu. Its function is not to display a complex analytical summary, but to give the administrator direct access to the modules required for daily cadre administration.

3.3.2. Commissariat and information management

**Fig. 9:** Administrator interfaces for commissariat and news management

The commissariat interface presents the branch units and their contact information in a card-based list. This supports rapid identification of a commissariat without searching through paper records. The news interface provides a tabular management view in which the administrator can publish, inspect, update, or remove announcements. These two modules address different information needs. Commissariat data are relatively stable organizational records, while news items are time-dependent communications that may require frequent revision.

3.3.3. Member and activation management

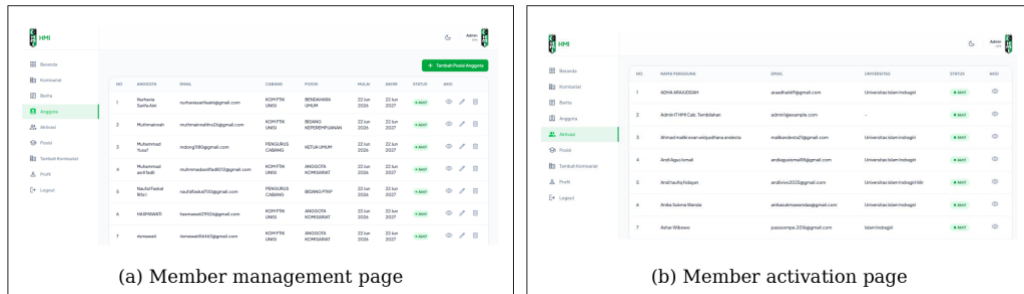


Fig. 10: Administrator interfaces for member records and membership activation

The member-management page presents cadre records in a searchable table and enables the administrator to add, edit, or delete data. The displayed columns provide the organizational and personal information needed for administration. Membership activation is handled on a separate page so that the presence of a registration record does not automatically imply an active membership status. The separation makes the status decision explicit and allows the current state to be inspected directly.

3.3.4. Organizational structure and commissariat creation

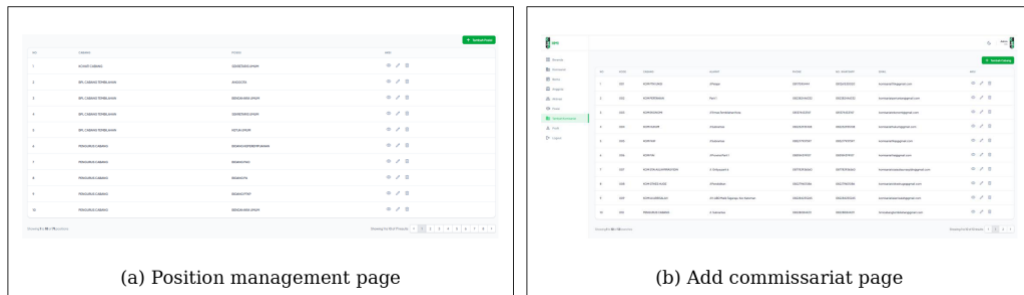


Fig. 11: Administrator interfaces for position management and commissariat creation

The position module stores organizational roles that can be used when maintaining cadre or management information. The add-commissariat page collects the identity and contact fields required for a new organizational unit. Before storage, the submitted data are validated. By separating master data such as positions and commissariats from member records, the application avoids repeated free-text entry and supports more consistent organizational information.

3.3.5. Profile and chairman access

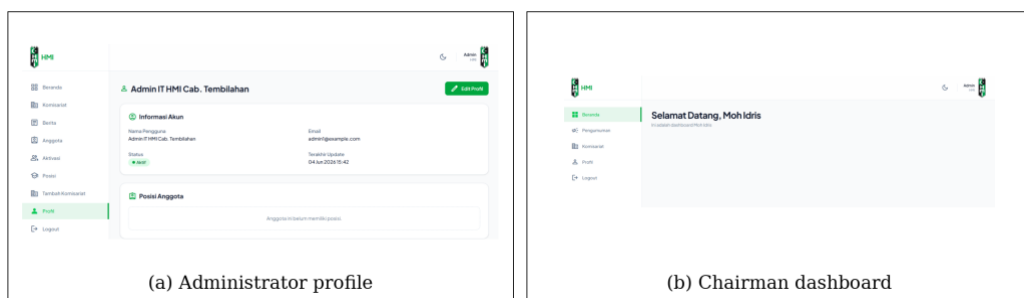


Fig. 12: Administrator profile and chairman dashboard

The profile page allows an authenticated administrator to maintain account information and change the password. The chairman dashboard demonstrates the restricted menu produced by the access-control design. It omits operational create, update, and delete functions and instead emphasizes the information that the chairman is authorized to inspect.

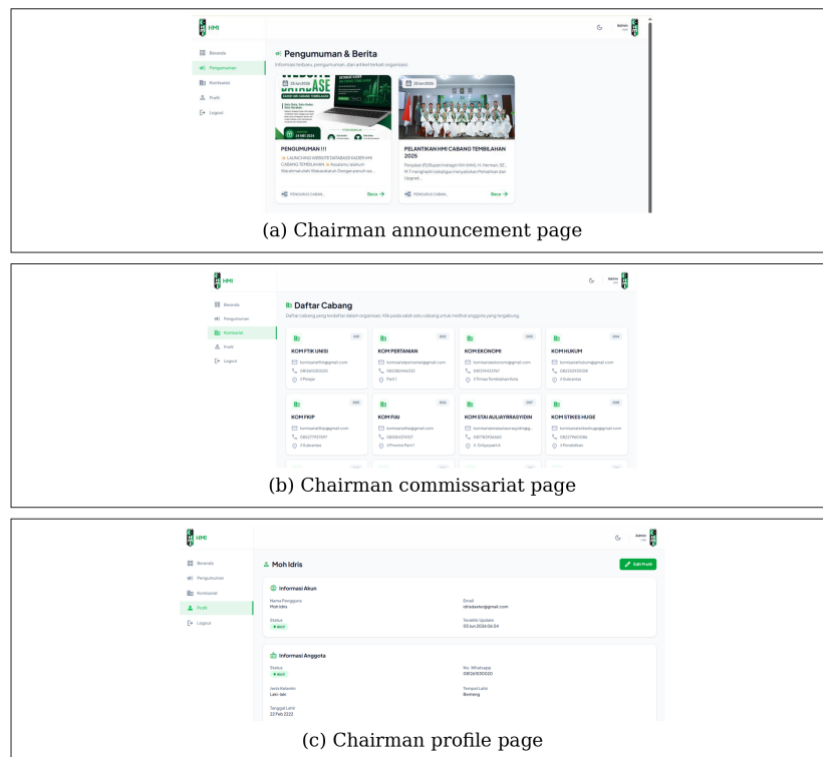


Fig. 13: Chairman interfaces for announcements, commissariats, and profile management

Through the chairman interfaces, published announcements can be read and commissariat contact information can be reviewed without granting permission to modify the underlying records. The chairman can update the personal profile and password, preserving control over the individual account. The implemented interface therefore corresponds to the use case boundary established during design.

3.4. Functionality test

The black-box test covered all 25 specified functions. Each function was executed by 25 respondents, resulting in 625 test scenarios. The grouped results are presented in Table 5. No failed output was recorded.

Table 5: Detailed functionality test results

No.	Function	Successful	Failed
1	Administrator login	25	0
2	Administrator dashboard	25	0
3	View and manage commissariats	25	0
4	Add commissariat	25	0
5	Update commissariat	25	0
6	Delete commissariat	25	0
7	View and manage news	25	0
8	Add news	25	0
9	Update news	25	0
10	Delete news	25	0
11	View and manage members	25	0
12	Add member	25	0
13	Update member	25	0
14	Delete member	25	0

No.	Function	Successful	Failed
15	Activate member	25	0
16	Manage positions	25	0
17	Manage administrator profile	25	0
18	Administrator logout	25	0
19	Chairman login	25	0
20	View announcements	25	0
21	View commissariats	25	0
22	Manage chairman profile	25	0
23	Chairman logout	25	0
24	Validate access rights	25	0
25	System-menu navigation	25	0
Total		625	0

Using $A = 0$ failed executions and $B = 625$ total executions, the functionality calculation is $X = 1 - A/B = 1 - 0/625 = 1$. This value indicates that every tested function produced the expected output. The result supports the technical feasibility of replacing the manual workflow with the implemented application. Nevertheless, the test establishes correctness under the tested scenarios and does not by itself demonstrate long-term performance under heavy concurrent use.

3.5. Usability test

The usability questionnaire produced an actual score of 1,095 from a maximum of 1,250 points. The resulting percentage was 87.64%, categorized as very good in the thesis. Table 6 provides the result for each statement, and Figure 14 visualizes the distribution.

Table 6: Usability questionnaire results

No.	Statement	Actual score	Maximum score	Percentage
1	The system makes cadre data management faster.	109	125	87.27%
2	The system is beneficial for HMI cadre data management.	114	125	90.91%
3	The system is easy to use on the first attempt.	98	125	78.18%
4	The interface is easy to understand.	111	125	89.09%
5	The system can be used without assistance.	107	125	85.45%
6	The available features are easy to understand.	109	125	87.27%
7	Users can become accustomed to the system quickly.	102	125	81.82%
8	Users are satisfied with the system.	111	125	89.09%
9	The system makes member-data management easier.	116	125	92.73%
10	The system is suitable for use at HMI Tembilahan Branch.	118	125	94.55%
Total		1095	1250	87.64%

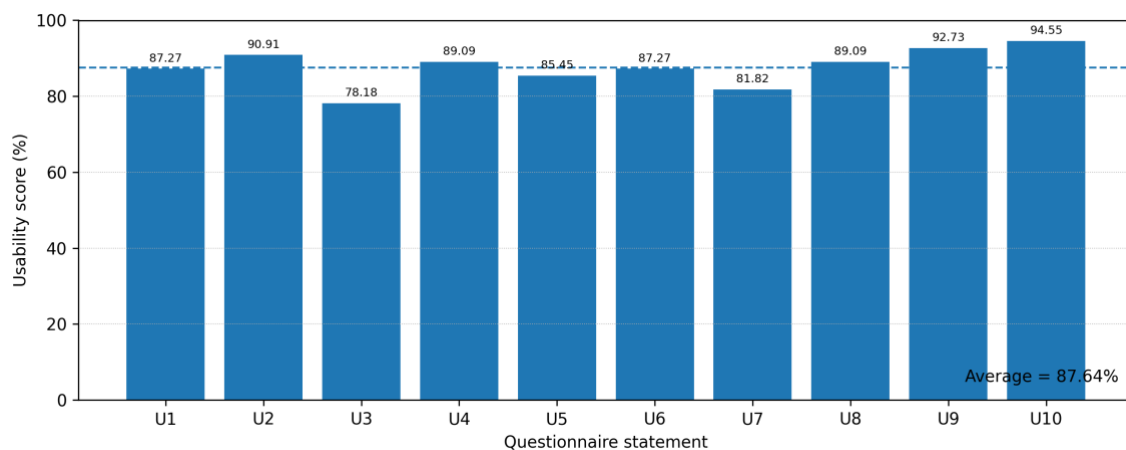


Fig. 14: Usability percentage for each questionnaire statement and the overall average

The highest score, 94.55%, was obtained for system suitability at HMI Tembilahan Branch, followed by the ease of member-data management at 92.73%. These results suggest that respondents recognized the practical relevance of the application. The lowest value, 78.18%, concerned ease of use on the first attempt. Although this value remained above 75%, it signals that initial orientation, clearer field instructions, or a short user guide would further reduce the learning barrier. The second-lowest value, 81.82%, concerned how quickly users became accustomed to the system, reinforcing the same improvement priority.

Taken together, the functionality and usability findings show that the system is not only operational but also accepted positively by its intended users. The strongest contribution is the conversion of a scattered manual process into a role-based data structure tailored to the branch and commissariat context. The evaluation, however, was conducted with 25 respondents in an initial implementation setting. Future studies should examine security resilience, backup recovery, larger user populations, and performance under simultaneous access.

4. Conclusion

The study successfully designed and implemented a web-based cadre data management information system for HMI Tembilahan Branch. The application integrates member, commissariat, announcement, activation, organizational position, and profile data within a role-based structure for administrators and the HMI chairman. PIECES analysis showed that the design directly addresses delays, fragmented information, paper dependence, weak access control, repetitive work, and slow administrative service. Testing of 25 functions by 25 respondents produced 625 successful executions and no failures, resulting in a functionality value of 1. The usability score reached 87.64%, with system suitability and easier member management receiving the strongest assessments. The application is therefore suitable as an administrative support tool for the branch, while the lower first-use score indicates that user guidance should accompany deployment. Further development should add automated backup, stronger authentication, exportable reports, mobile access, and integration with wider HMI organizational levels.

References

- [1] A. S. Rosa and M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika, 2018.
- [2] B. Raharjo, *Belajar Otodidak Framework Laravel*. Bandung: Informatika, 2019.
- [3] E. Turban, R. Sharda, and D. Delen, *Decision Support and Business Intelligence Systems*, 10th ed. Pearson, 2017.
- [4] K. C. Laudon and J. P. Laudon, *Management Information Systems: Managing the Digital Firm*, 16th ed. Pearson, 2020.
- [5] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. New York: McGraw-Hill, 2020.
- [6] I. Sommerville, *Software Engineering*, 10th ed. Pearson, 2016.
- [7] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Pearson, 2017.
- [8] H. M. Jogiyo, *Analisis dan Desain Sistem Informasi*. Yogyakarta: Andi Offset, 2017.
- [9] B. Nugroho, *Aplikasi Pemrograman Web Dinamis dengan PHP dan MySQL*. Yogyakarta: Gava Media, 2019.
- [10] R. A. Sukanto and M. Shalahuddin, *Modul Pembelajaran Rekayasa Perangkat Lunak*. Bandung: Modula, 2020.
- [11] D. A. Norman, *The Design of Everyday Things*, Revised ed. New York: Basic Books, 2013.
- [12] J. Brooke, "SUS: A Quick and Dirty Usability Scale," in *Usability Evaluation in Industry*. London: Taylor & Francis, 1996.
- [13] A. H. Hendrawan, "Penerapan Metode PIECES dalam Analisis Sistem Informasi," *Jurnal Sistem Informasi dan Teknologi Informasi*, vol. 5, no. 2, pp. 45-53, 2021.
- [14] M. F. Hidayat and A. Prasetyo, "Perancangan Sistem Informasi Berbasis Web Menggunakan Framework Laravel," *Jurnal Teknologi Informasi dan Komputer*, vol. 8, no. 1, pp. 32-41, 2022.
- [15] ISO/IEC 25010:2011, *Systems and Software Engineering - Systems and Software Quality Requirements and Evaluation (SQuaRE) - System and Software Quality Models*. Geneva: International Organization for Standardization, 2011.