

Development of a Complaint Application for the Education Agency Using the Agile Development Method

Sinta Tumbo^{1*}, Medi Hermanto Tinambunan²

^{1,2}Program Studi Teknik Informatika, Fakultas Teknik, Universitas Negeri Manado
sintatumbo458@gmail.com^{1*}, meditinambunan@unima.ac.id²

Abstract

The rapid advancement of information technology has driven changes in performance and problem solving in society and government agencies. In the education sector, the Education Office, as the main service provider, requires an effective complaint mechanism for students, teachers, employees, and the community. The current complaint method still often relies on face-to-face submission, which causes problems such as difficulty in tracking the status of complaints, poor documentation, and limitations in evaluating the quality of complaint handling. This study aims to develop a web-based complaint application to digitize and harmonize the complaint process at the Minahasa Regency Education Office. The Agile Development method was used due to its iterative, flexible, and collaborative nature, allowing for the gradual development of features based on direct feedback from stakeholders. Data collection techniques included observation, interviews, documentation studies, and literature studies. The system was designed using UML diagrams, including Use Case, Activity, Sequence, and Class diagrams. Development was carried out in sprints, focusing on core features: user registration with NIK verification, complaint submission and tracking, and an admin dashboard for complaint management. Functional testing using the Black-Box method confirmed that all key features operate correctly as required. The resulting application successfully transformed the manual complaint process into a more structured, transparent, and efficient digital system, thereby contributing to improved public service quality in the field of education.

Keywords: *Complaint System, Agile Development, Public Services*

1. Introduction

Rapid advances in science, technology, and information have led to changes in problem-solving, both in society and in government agencies. One technological development that can be felt today is the internet [1]. Advances in computer technology, both in terms of hardware and software, have driven innovation in various fields [2]. The development of information technology has opened up opportunities to improve the quality of public services through digitalization. The government must be responsible for providing good public services in order to support good governance [3].

In the education sector, the Education Agency, as the spearhead of service provision, needs an effective complaint mechanism for students, teachers, employees, and the community. Currently, complaints are generally still made face-to-face, so the public must come to the agency to file a complaint [4]. Human activities are also inseparable from communication between humans and industrial technology [5]. This method causes several problems, including difficulty in tracking the status of complaints, lack of historical documentation, and limitations in evaluating the quality of complaint handling. As a result, the potential for improving the quality of public services is hampered by the absence of structured data that can be further analyzed [6].

Digital-based complaint applications have emerged as a solution to overcome these problems. Complaint handling is a form of community participation in regional government [7]. This web-based service complaint application is an application that is indispensable in the process of evaluating complaints about services [8]. This type of application not only facilitates easier and faster reporting but also creates a clear track record, supports monitoring, and improves agency accountability. In its development, a method that is adaptive to changing user needs is required. The Agile Development method was chosen because of its iterative, flexible, and collaborative characteristics, which allow for the gradual development of features (sprints) based on direct feedback from stakeholders. Agile Software Development itself is currently trending due to several advantages it has to offer in facilitating information system development [9].

2. Research Methods

This study discusses the Agile Development Method, a software development approach that focuses on flexibility, collaboration, and adaptability to change. According to Sommerville, the Agile method is an incremental development method that focuses on rapid development, gradual software release, reducing process overhead, and producing high-quality code, and involves customers directly in

the development process [10]. Agile Software Development comes from the word Agility/Agile, which means fast, light, free to move, effective (fast and adaptive) in responding to change [11]. Agile Development is a software development methodology based on the same principles as short-term system development that requires developers to adapt quickly to changes of any kind [12]. Agile emphasizes a gradual (iterative) and repetitive (incremental) development process, where each stage of development produces a product that can be tested and evaluated by users. The Agile method is based on philosophies such as continuous improvement, consistent user involvement, and periodic software delivery [13]. System development using the agile method is effectively applied to almost all running systems given its flexibility, which can be applied to any stage of development without disrupting the running system [14]. This development research was carried out at the Minahasa Regency Education Office.

2.1. Data Collection Techniques

Data collection techniques are techniques used in collecting data for research [15]. Data collection was carried out as follows:

1. Observation

Observation was conducted by directly observing the current process of submitting, recording, and handling complaints at the education office. This technique was used to understand the manual workflow, obstacles that arise, and the system requirements needed to support the complaint process. The results of the observation formed the basis for designing relevant features for the application.

2. Interviews

Interviews were conducted with relevant parties, such as education office employees, teachers, students, and members of the community who had filed complaints. The interviews aimed to gather more in-depth information about user needs, obstacles encountered in the previous complaint system, and expectations for the application-based complaint system. Interviews can also be used in the agile phase as part of feedback on each iteration.

3. Documentation Study

A documentation study was conducted by collecting and studying documents related to the complaint process, such as complaint forms, previous complaint reports, standard operating procedures (SOPs), and other administrative documents used by the education office. This technique helped to understand the formal rules, data structure, and procedures that must be followed in application development.

4. Literature Study

A literature study was conducted by reviewing journals, books, scientific articles, and previous research related to complaint systems, agile development methods, and information system development. This technique was used as a theoretical basis and comparison with other studies, so that system development could follow scientific standards and best practices.

2.2 Research Stages

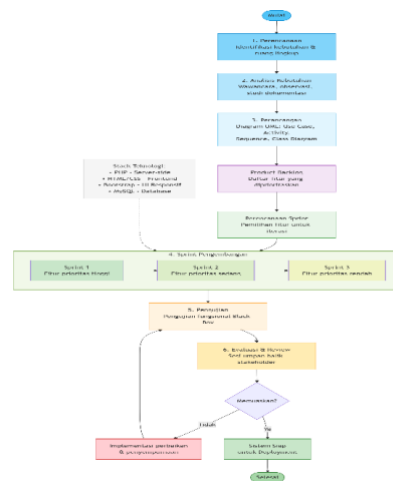


Fig. 1: Development Method Flowchart

This development method involves the following stages:

1. Planning – identifying initial requirements, determining problems, system scope, and establishing the objectives and main features of a system.
2. Requirement analysis – collecting and formulating the system through interviews, observations, and documentation studies.
3. Design – designing the system model using UML diagrams, namely Use Case Diagrams for actor interactions, Activity Diagrams for process flows, Sequence Diagrams for message sequences between objects, and Class Diagrams for data structures.
4. Development – building the application iteratively. Each sprint focuses on developing priority features. The technologies used include PHP as the server-side programming language, HTML and CSS for structure and display, and the Bootstrap framework for interface responsiveness. The database is managed using MySQL.
5. Testing – performing Black-Box Testing functional testing on each feature that has been developed to ensure the application runs according to the requirements specifications.
6. Review & iteration – Evaluating the results of each sprint with stakeholders (agencies) and making improvements based on the feedback received before continuing to the next sprint.

Ngmmf

3. Result and Discussion

All paragraphs must be justified alignment. With justified alignment, both sides of the paragraph are straight.

3.1. Design

Design is the process of determining the information needed for a new system [16]. System design is a process carried out in building a system or improving a previously created system so that it becomes better and can be more effective and efficient in its operation [17]. Based on the analysis, two main actors were identified: Users (the public, students, teachers) and Admins (government employees). The main functional requirements include: NIK registration/verification, login, complaint creation, complaint status tracking for users, as well as a dashboard, management, and complaint handling for admins. The results of the system design are outlined in:

1. Use Case Diagram: Maps all interactions between users/admins and the system.
2. Activity & Sequence Diagram: Details the logical flow for main scenarios such as registration, login, creation, and complaint handling.
3. Class Diagram: Designs the main class structure such as Community, Admin, Complaints, and Categories.

Here is an explanation of the design results:

3.1.1 Use Case Diagram

Use case diagrams are important for visualizing, defining, and documenting the behavioral requirements of a system [18]. Use case diagrams serve to display the relationships that occur between actors and cases [19]. Use Case Diagrams are also diagrams used to model the behavior of a designed system by describing the interactions between one or more actors using the system and to identify the needs and objectives of the system to be developed, as shown in Figure 1.

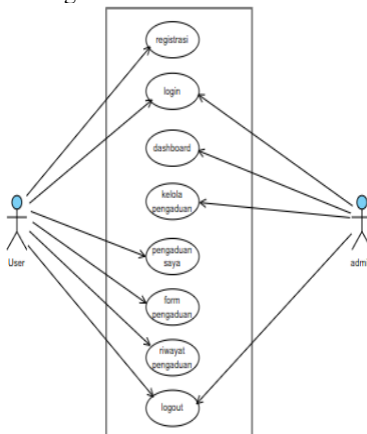


Fig. 2: Use Case Diagram

The use case descriptions are shown in the following table:

Table 1: User Use Case Descriptions

Use Case	Description
Registration	Users must register before proceeding to the login page.
Login	Users enter their username and password to log in to the system page.
My Complaints	Users can view the “My Complaints” page.
Complaint Form	Users fill out the complaint form.
Complaint History	Users can view the history of complaints that have been submitted.
Logout	Users can exit the page when they are finished.

Table 2: Admin Use Case Descriptions

Use Case	Description
Login	The administrator enters their username and password.
Dashboard	The administrator can view the dashboard page containing user complaints.
Manage Complaints	The administrator manages or responds to user complaints.
Logout	The administrator can log out of the system when finished.

3.1.2. Activity Diagram

According to Sukamanto and Shaluddin, activity diagrams illustrate the workflow or activities of a system, business process, or menu in software. Activity diagrams illustrate the activities of a system, not what actors do.

1. Activity Diagram Registrasi

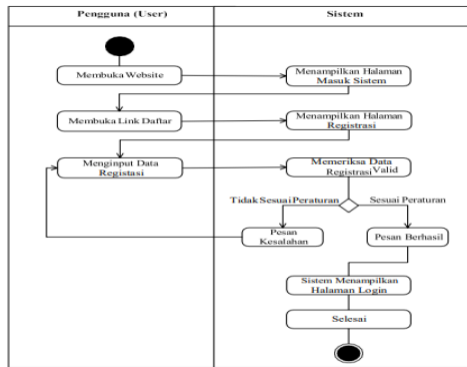


Fig. 3: Activity Diagram Registrasi

2. Activity Diagram for Complaint Filing

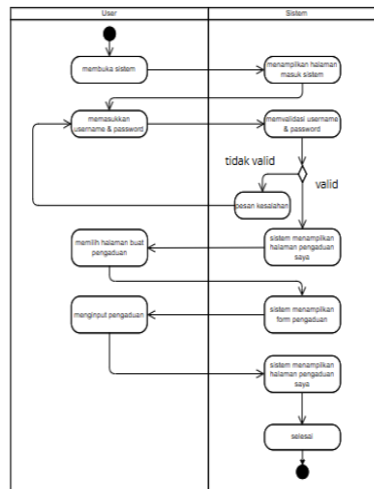


Fig. 4: Activity Diagram for Complaint Filing

3. Activity Diagram Respond to Complaints

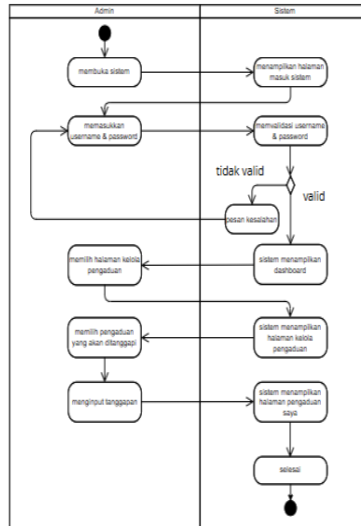


Fig. 5: Activity Diagram Respond to Complaints

3.1.3. Sequence Diagram

A Sequence Diagram is a type of UML (Unified Modeling Language) diagram that displays interactions between objects or components in a system based on chronological order.

1. Sequence Diagram Registrasi

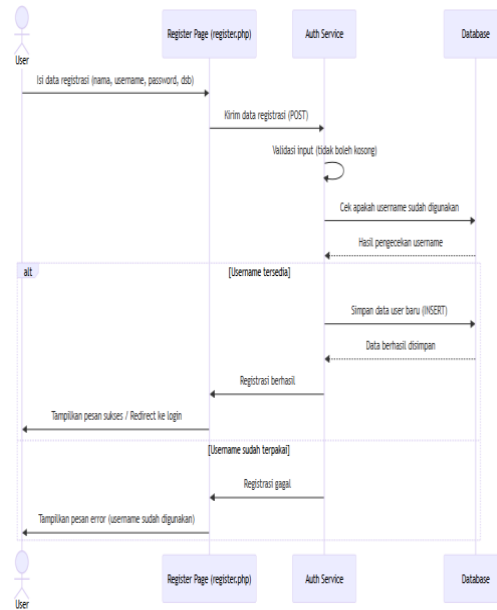


Fig. 6: Sequence Diagram Registrasi

2. Sequence Diagram Create a Complaint to the User

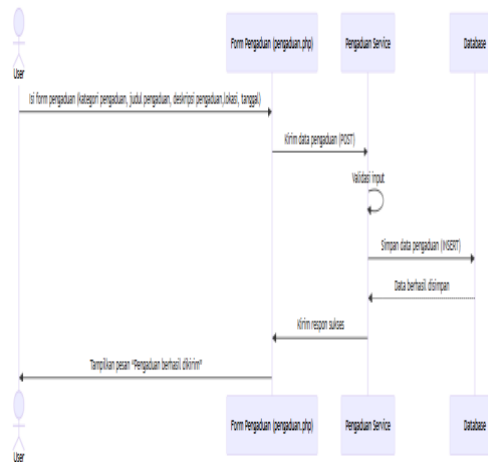


Fig. 7: Sequence Diagram Create a Complaint to the User

3.1.4. Class Diagram

A Class Diagram is a type of structural diagram in UML that shows the static structure of a system by depicting the classes in the system, their attributes, operations/methods, and the relationships between classes.

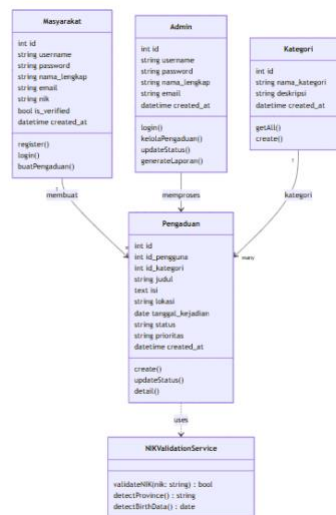


Fig. 8: Class Diagram

3.2. Sprint Development

All Sprint is the heart or core of scrum itself, which means that a time limit of one month or less, during which the work is considered complete, usable, and potentially releasable, can be used. Sprints usually have a fixed duration throughout the system development process until the system is complete [20]. The development process is carried out in several sprints:

1. Sprint 1: Basic Feature Development
 - a. Implementation of login and registration system
 - b. Creation of user dashboard page
 - c. Integration of initial UI design
 Result: Authentication feature works well, users can register and log in to the system.
2. Sprint 2: Complaint Feature
 - a. Complaint form based on category
 - b. Data storage to database
 Result: Users can create complete complaint reports.
3. Sprint 3: Admin System
 - a. Complaint management display
 - b. Status change feature (pending, processing, completed)
 - c. Notifications to users
 Result: Admins can monitor complaints and update progress in real time.
4. Sprint 4: Reporting & Optimization
 - a. Complaint report dashboard
 - b. Graph showing the number of complaints per category
 - c. Optimization of display and performance
 Result: Leaders can monitor the number of complaints and trends based on specific periods.

3.3. Testing

Testing was conducted using the Black Box Testing method. The results showed that all key features functioned as required. Examples of test results:

Table 3: Testing

Features	Results	Description
Registration & Login	Successful	Validation works
Complaint Input	Successful	Data is saved
Status Update	Successful	Changes are visible to users
Notifications	Successful	Users receive notifications

3.4. System Implementation

This section describes the results of system development visually through several interface displays. The following images serve as concrete evidence that the system has been successfully built and can be used by users.

3.4.1. Registration Page

The first display is the registration page, where each user must create a new account to log into the system. There are two steps on this page, namely NIK verification and filling in account data.

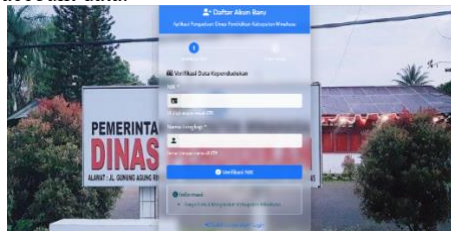


Fig. 9: NIK Verification Page



Fig. 10: Account Information Page

3.4.2. Login Page

The second screen shown is the login page, where every user and administrator must authenticate before entering the system. This page is the first step before users can access the complaint feature.



Fig. 11: Login Page

3.4.3.Admin Dashboard Page

The fourth view, where this page displays the admin dashboard, which contains the complaint title, complainant, date, status, and action. This dashboard makes it easy for admins to monitor the number of complaints received in real time.

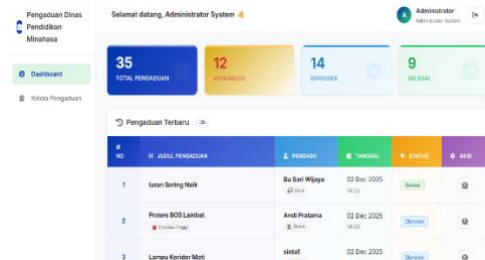


Fig. 12: Admin Dashboard Page

3.4.4.User Page

For general users such as students or the public, the system provides a simpler user dashboard. Users can view the status of their personal complaints or create new reports.

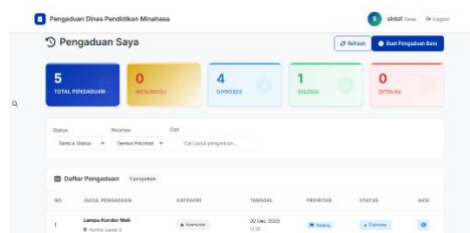


Fig. 13: User Page

3.4.5.Complaint Form Page

This section shows the results of implementing the core feature, namely complaint creation. The complaint form contains categories, complaint titles, full descriptions, locations, dates of occurrence, and priorities.

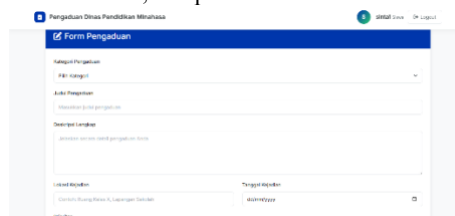


Fig. 14: Complaint Form Page

3.4.6.List of Submitted Complaints

All reports submitted by users are displayed in a table. Admins can monitor the report title, category, priority, and status.

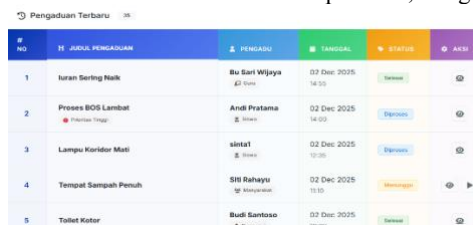


Fig. 15: List of Submitted Complaints

3.4.7.Manage Complaint

On this page, the administrator will select one of the reports to display the details page to change the complaint status.

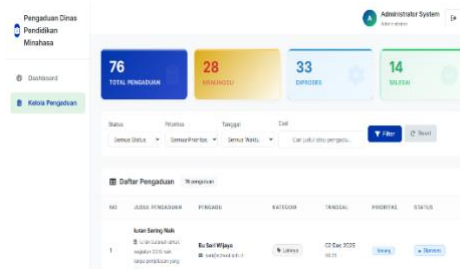


Fig. 16: Manage Complaint

3.4.8. Complaint Details Page and Change Status

When the admin selects a report, the system displays a details page to view the complete contents of the complaint and update its status.



Fig. 17: Complaint Details Page and Change Status

4. Conclusion

This research was successful in implementing the Agile Development method, which proved to be effective in developing a complaint application that meets the dynamic needs of users. The resulting application was able to transform the manual complaint process into a more structured, transparent, and efficient digital system, thereby contributing to improving the quality of public services in the field of education. The iterative and incremental Agile approach proved to be effective in adjusting feature development to the dynamic needs of users and stakeholders. Functional (black-box) testing confirmed that all key features, such as registration with NIK verification, complaint creation and tracking, and an admin dashboard for report management, operate well as needed. With comprehensive system design documentation using UML, this research not only produced a functional product but also provided a clear foundation for future system development and maintenance.

References

- [1] Y. Sansena, "Implementasi Sistem Layanan Pengaduan Masyarakat Kecamatan Medan Amplas Berbasis Website," *J. Ilm. Teknol. Inf. Asia*, vol. 15, no. 2, pp. 91–102, 2021, doi: 10.32815/jitika.v15i2.611.
- [2] A. Syear, Y. Samuel, D. A. Pratama, and R. A. Althaaf, "Perancangan Sistem Informasi Absensi Karyawan Berbasis Web Pada Restoran Sejahtera Menggunakan Metode Agile Development," vol. 2, no. 9, pp. 1739–1746, 2025.
- [3] A. Alkodri, B. Isnanto, and S. Sujono, "Aplikasi Pengaduan Masyarakat Untuk Pelaporan Kejadian Dan Bencana Di Basarnas Bangka Belitung," *CSRID (Computer Sci. Res. Its Dev. Journal)*, vol. 11, no. 2, p. 96, 2021, doi: 10.22303/csrid.11.2.2019.96-104.
- [4] T. Sumarno and A. Mubarak, "Aplikasi Pengaduan Masyarakat Terhadap Infrastruktur Bebas Android Pada Desa Sindangsari," *Competitive*, vol. 16, no. 2, pp. 80–86, 2021, doi: 10.36618/competitive.v16i2.1487.
- [5] S. Sudaryono, N. P. Lestari, and K. Gunawan, "Perancangan Virtual Assistant Entrepreneurship Menggunakan Metode Scrum," *J. Innov. Futur. Technol.*, vol. 2, no. 2, pp. 66–77, 2020, doi: 10.47080/ifttech.v2i2.1021.
- [6] S. Khodijah and W. S. Aulia, "Analisis Sistem Informasi Pengaduan Masyarakat Berbasis Web Menggunakan Metode Agile," *Journal.Jci.Co.Id/Jisbt*, vol. 1, no. 1, pp. 8–14, 2025, [Online]. Available: <https://journal.jci.co.id/jisbt/id/article/view/4>
- [7] W. A. Yohanitas and T. H. Prayitno, "Pengelolaan Pengaduan Masyarakat Kota Bekasi," *J. Borneo Adm.*, vol. 10, no. 3, pp. 328–352, 2014.
- [8] B. Sakti, R. A. Saragih, and R. Hamdani, "Aplikasi Pengaduan Pelayanan Berbasis Web Pada Kantor Imigrasi Kelas II TPI Pematangsiantar," *J. Comput. Sci. Informatics Eng.*, vol. 01, no. 2, pp. 54–65, 2022, doi: 10.55537/cosie.v1i2.43.
- [9] K. Anwar, L. D. Kurniawan, M. I. Rahman, and N. Ani, "Aplikasi Marketplace Penyewaan Lapangan Olahraga Dari Berbagai Cabang Dengan Metode Agile Development," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 2, pp. 264–274, 2020, doi: 10.32736/sisfokom.v9i2.905.
- [10] I. M. Widiarta, Y. Mulyanto, and A. Sutrianto, "Rancang Bangun Sistem Informasi Inventory Menggunakan Metode Agile Software Development (Studi Kasus Toko Nada)," *Digit. Transform. Technol.*, vol. 3, no. Maret, p. 20, 2023.
- [11] I. Hariman and C. Meilisa, "SISTEM INFORMASI MANAJEMEN TRAINING MENGGUNAKAN METODE AGILE SOFTWARE DEVELOPMENT THE PAPANDAYAN HOTEL BANDUNG," *Ensains J.*, vol. 3 No 1, pp. 60–67, 2020.
- [12] M. Y. Suntana, T. Terttiaavini, P. Oktariani, F. Sanawi, M. D. Akbar, and A. W. Pratama, "Perancangan Aplikasi Solusi Publik Untuk Pengaduan Masyarakat Mengenai Fasilitas Umum," *J. Inf. Syst. Manag. Digit. Bus.*, vol. 1, no. 3, pp. 332–334, 2024, doi: 10.59407/jismdb.v1i3.738.
- [13] Kornelia Kidi Olarne and Arief Rahman Hakim, "Pengembangan Aplikasi Pengaduan Masyarakat DP3A Kabupaten Nabire Berbasis Web menggunakan Metode Agile," *J. Teknol. dan Inform.*, vol. 3, no. 1, pp. 93–106, 2025, doi: 10.70539/jti.v3i1.54.
- [14] L. Langgeng, A. Fahrudin, and A. Maksam, "Perancangan Aplikasi Cafe Untuk Efisiensi Order Menggunakan Metode Agile," *Semin. Nas. Teknol. Inf.*, pp. 113–120, 2017.
- [15] S. Suhari, A. Faqih, and F. M. Basysyar, "Human Resources Information System Using Agile Development Method at CV. Angkasa Raya," *J. Teknol. dan Inf.*, vol. 12, no. 1, pp. 30–45, 2022, doi: 10.34010/jati.v12i1.
- [16] L. Trisnawati and D. Setiawan, "Sistem Monitoring Kegiatan Kemahasiswaan Menggunakan Metode Agile Development," *JOISIE J. Inf. Syst. Informatics Eng.*, vol. 6, no. 1, pp. 49–57, 2022.

-
- [17] A. C. Hutauruk and A. F. Pakpahan, "Perancangan Sistem Informasi Organisasi Kemahasiswaan Berbasis Web pada Universitas Advent Indonesia Menggunakan Metode Agile Development (Studi Kasus: Universitas Advent Indonesia)," *CogITo Smart J.*, vol. 7, no. 2, pp. 315–328, 2021, doi: 10.31154/cogito.v7i2.328.315-328.
- [18] H. Handayani, A. M. Ayulya, K. U. Faizah, D. Wulan, and M. F. Rozan, "Perancangan Sistem Informasi Inventory Barang Berbasis Web Menggunakan Metode Agile Software Development," *J. Test. dan Implementasi Sist. Inf.*, vol. 1, no. 1, pp. 29–40, 2023, doi: 10.55583/jtisi.v1i1.324.
- [19] H. Hutrianto and A. Putra, "Implementation of Scrum Model in the Development of Waste Reporting Applications As a Smart Cleaning World," *JIPi (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 5, no. 1, p. 9, 2020.
- [20] M. A. Damanhuri, I. B. Praptono, and M. Dellarosawati, "Perancangan Aplikasi Pencatatan Keuangan Berbasis Website Dengan Menggunakan Metode Agile Development Studi Kasus Pada Warung Nasi Sunda Mang Ujang," *e-Proceeding Eng.*, vol. 7, no. 2, pp. 5638–5645, 2020, [Online]. Available: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/12682>