



jsret.knpub.com

E-ISSN: 2962-6110
P-ISSN: 2964-7843

JSRET

Journal of Scientific Research, Education, and Technology

Interdisciplinary science, open access, and peer-reviewed journal that disseminates research findings from lecturers, researchers, teachers, and scientists in a variety of scientific and technological fields. This is an open-access article distributed under the terms of the Creative Commons Attribution-ShareAlike 4.0 International License (<https://creativecommons.org/licenses/by-sa/4.0/>)

Optimization of Elementary School Learning Processes through a Multi-Platform E-Learning System

Wahyu Akbar, Rr. Hajar Puji Sejati

Universitas Teknologi Yogyakarta, Yogyakarta, Indonesia

ABSTRACT

The transformation of digital learning requires an integrated platform that can effectively support academic processes at the elementary school level. This study aims to develop an integrated multi-platform e-learning system for SD Negeri Tengah 08 Jakarta to optimize the efficiency of academic management and the flexibility of learning access. The system was built using a RESTful API architecture with Golang as the backend, PostgreSQL as the database, React.js for the web interfaces for teachers and administrators, and Flutter for the mobile application for students. Based on the system development method and Black Box Testing, the results show that the integration successfully enables real-time data synchronization without technical issues. All functional features, including NISN-based authentication, automated scheduling, material distribution, and interactive quiz evaluation, were validated successfully. This implementation significantly establishes a responsive digital learning ecosystem, streamlines teachers' administrative workload, and enhances student engagement in flexible self-directed learning, while strengthening the consistency of a structured, adaptive, and sustainable online learning process for all school stakeholders. This study contributes to the field by providing a practical framework for scalable and interoperable e-learning systems that can be adopted in other elementary schools to improve academic management and student learning experiences.

Keywords: *E-Learning, Multi-Platform, Golang, React.js, Flutter*

Corresponding author

Name: *Wahyu Akbar*

Email: *wahyuakbr136@gmail.com*

INTRODUCTION

The rapid development of information and communication technology has brought fundamental changes to the field of education. One significant change is the shift from conventional learning methods to digital learning or e-learning (Auliya Rahma et al., 2024). E-learning systems provide students with opportunities to learn flexibly, interactively, and accessibly anytime and anywhere (Wulansari & Suhartini, 2024). On the other hand, these systems also facilitate educators in managing learning materials, conducting assessments, and monitoring student progress in real time (Palandi et al., 2025). However, most existing e-learning systems still face challenges in cross-platform integration, system performance

efficiency, and scalability, especially in the context of elementary education, which requires simple, fast, and stable interfaces. This situation also occurs at SD Negeri Tengah 08 Jakarta, where digital learning is still conducted using multiple separate platforms, often resulting in data inconsistency, limited access, and a high administrative workload for teachers.

Previous studies have attempted to improve the effectiveness and efficiency of e-learning systems. For example, (Jinan et al., 2025) developed a web-based e-learning system using the Laravel Framework, emphasizing ease of access and structured content management. Meanwhile, (Sopandi et al., 2024) designed a mobile-based e-learning application using Flutter to enhance flexibility and learning accessibility across various devices. Additionally, (Calista Chandra et al., 2024) investigated the integration of web and mobile platforms using Node.js-based RESTful API to support real-time data synchronization across devices in digital learning systems.

From these studies, it can be concluded that although web and mobile technologies have been widely applied separately, a research gap still exists regarding how to fully integrate both into a single, efficient, scalable, and user-friendly e-learning ecosystem for educators and students. Most previous studies focused on developing single applications, either web or mobile, without considering cross-platform interoperability in real time. An integrative approach that combines web and mobile technologies within a service-oriented architecture remains underexplored, even though such integration is crucial for ensuring data consistency, communication efficiency, and a uniform user experience across devices. This gap forms the primary basis for this study, particularly because SD Negeri Tengah 08 Jakarta requires a system capable of unifying all academic processes into a single integrated platform.

To address these issues, this study proposes the development of an integrated e-learning system using Golang as the backend service, React.js for the web user interface, Flutter for the mobile application, and PostgreSQL as the relational database management system. Golang was chosen for its advantages in supporting high-performance concurrent processes with low resource consumption (Harjoseputro et al., 2020). React and Flutter were selected for their ability to produce modern, responsive, and consistent interfaces across platforms (Bismoputro et al., 2017; Piter & Lee, 2025). PostgreSQL ensures transaction reliability, data security, and storage scalability.

The primary objective of this study is to design and implement a fully integrated web and mobile e-learning system to support elementary education. This integration is expected to improve accessibility, accelerate the learning process, and facilitate interactions between teachers and students within a unified digital ecosystem. Moreover, this study aims to contribute to the development of a hybrid e-learning architecture that combines modern technologies to achieve system efficiency and an optimal user experience.

Thus, this study not only provides technical contributions to system development but also offers practical benefits for elementary schools. The resulting system is expected to assist SD Negeri Tengah 08 Jakarta in expanding digital learning access, improving administrative efficiency, and strengthening interactions among teachers, students, and

parents through an adaptive and user-friendly learning ecosystem. More broadly, the results of this study can serve as a reference for other developers and researchers in building e-learning systems that emphasize high performance, data security, and cross-platform interoperability in the digital era.

METHOD

1. Requirement Analysis

The requirement analysis stage was conducted to identify user characteristics, system needs, and the scope of the integrated e-learning application development. The analysis was carried out through observations of the learning process at SD Negeri Tengah 08 Jakarta, as well as interviews with teachers and school staff, to collect data on challenges in online learning, required features, and user expectations.

The primary system users consist of three roles: administrators, teachers, and students, each with different access rights and functions. Administrators are responsible for user management, academic data administration, and system configuration. Teachers are responsible for uploading learning materials, assigning tasks, and evaluating student learning outcomes. Students act as recipients of learning materials, task executors, and participants in online learning activities. Specifically, the target students for this e-learning system are those in grades 4 to 6, who require an easy-to-use interface, flexible access to learning materials, and the ability to complete quizzes and assignments independently in accordance with the upper elementary school curriculum.

In addition to functional requirements, non-functional requirements were also identified, including data security, cross-platform interoperability, access speed, and usability on various devices, both desktop and mobile (Syahputra & Aryanto, 2025). The results of this analysis serve as the foundation for designing a multi-platform e-learning system with real-time synchronization and a distributed service-oriented architecture to ensure efficiency, scalability, and a consistent user experience.

2. Design Planning

The design planning stage focuses on translating the results of the requirement analysis into the system's technical design. This process includes designing the system architecture, process flows using flowcharts, database structure through an Entity Relationship Diagram (ERD), and user interface design. Flowcharts are used to illustrate the logical sequence of system operations, such as user authentication, material uploads, and student assignment submissions. With flowcharts, the relationships between processes can be clearly visualized, minimizing errors during the implementation stage (Samsul Hadi et al., 2023).

The system is designed using a multi-tier architecture approach that separates the frontend, backend, and database components. The backend is developed with Golang as a RESTful API service provider, while the web frontend uses React.js, and the mobile application is built with Flutter to ensure a consistent interface across multiple

devices. The system's data structure is implemented using PostgreSQL and visualized through an ERD, which includes the entities User, Material, Assignment, Score, and Learning Activity. Furthermore, a multi-level authentication mechanism based on JSON Web Token (JWT) is applied to ensure secure access according to user roles, and the user interface design follows user-centered design principles to facilitate ease of use for elementary school students (Sudrajat et al., 2025).

3. Implementation

The implementation stage is the process of realizing the design into an integrated web and mobile e-learning system. The backend is implemented using Golang with the Gin Gonic framework, which manages API routes and JWT authentication middleware. Data communication between the frontend and backend is conducted through RESTful API endpoints, encrypted via HTTPS to ensure secure data transmission.

The web frontend is developed using React.js with supporting libraries such as Axios for API connections and Redux for global state management of the application (Rahmadhani et al., 2024; Wisnu Shena Arrafi & Putra, 2025). The mobile application is built using Flutter, which leverages the widget tree for layout management and API connections to synchronize learning materials, assignments, and student progress.

All system components are integrated with PostgreSQL as the primary database server, running on separate containers using Docker to support horizontal scalability. Testing was conducted across various devices, including desktops, tablets, and smartphones, to ensure consistency in functionality and user interface appearance.

4. Testing

The testing stage was conducted to ensure that the system functions in accordance with the requirements and specifications (Amin Rahim Hidayat et al., 2025). The testing method employed was Black Box Testing, which focuses on evaluating the system's functionality based on inputs and outputs without examining the source code (Wulandari et al., 2022).

Each system module was tested separately to ensure proper functionality. Testing covered features such as login and authentication, user management, material uploads, student assignment submissions, and the assessment process. The results indicated that all primary functions operated according to the expected scenarios, with no logic errors observed during use.

Black Box Testing was chosen because it is effective for verifying the system's functional compliance with user requirements without the need for internal code inspection (Adisty Yudianto Putri et al., 2025). This approach provides a comprehensive overview of the system's completeness and functional reliability before further evaluation in the results testing stage.

FINDING AND DISCUSSION

RESEARCH RESULT

1. System Architecture

Figure 1 illustrates the multi-platform e-learning system architecture developed in this study. The architecture involves three main actors: teachers, administrators, and students, each interacting with the system through different platforms according to their roles. Teachers and administrators access the system via the web interface, while students use the mobile application.

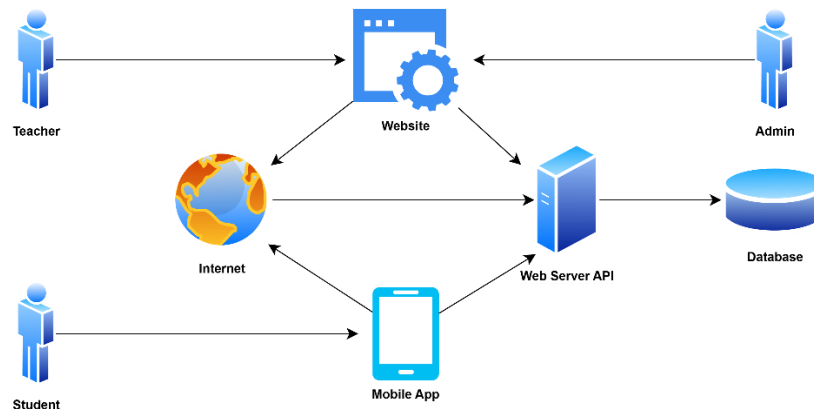


Figure 1: System Architecture

All system components are connected through a Web Server API, which serves as the main link between the frontend (web and mobile) and the database. The API server is responsible for processing user requests, managing authentication, and performing centralized data retrieval and storage. With this architectural design, the e-learning system can operate in an integrated, efficient, and real-time manner, ensuring data consistency and a uniform user experience across all platforms.

2. Requirement Analysis

The requirement analysis stage was conducted to understand user characteristics, ongoing learning processes, and challenges encountered in implementing online learning at the elementary school level. This analysis was carried out through direct observation of teaching and learning activities at SD Negeri Tengah 08 Jakarta, as well as interviews with teachers, students, and school staff. The objective of this stage was to identify user needs and determine the key features that must be available in the e-learning system to effectively and efficiently support the learning process.

The analysis revealed several challenges in online learning, such as limited access to certain devices, difficulties in cross-platform data synchronization, and complexity in managing learning materials and assessments. Based on these findings, functional requirements were formulated, including user management (administrators, teachers, students), learning material management, assignment submission and

assessment, as well as interactive communication between teachers and students. Non-functional requirements include data security through multi-level authentication, cross-platform access speed, system interoperability between web and mobile, and usability to ensure that the system can be comfortably operated by users with varying levels of digital literacy. The results of this analysis served as the foundation for the design stage of a real-time integrated multi-platform e-learning system.

3. Design

In the system design stage, the process flow and database structure were designed to illustrate the overall operation of the e-learning system. This design includes two main components: flowcharts and the Entity Relationship Diagram (ERD). Flowcharts are used to depict the sequence of processes performed by each system actor, while the ERD represents the relationships between entities in the database.

a. System Flowchart

Flowcharts are used to depict the overall workflow of the system so that each activity performed by the actors can be clearly and systematically understood. The system flow is divided into three main parts according to user roles: teachers, administrators, and students. Each flowchart shows the stages of interaction, decisions, and processes performed by each actor, from authentication to data management within the system. This modeling allows developers and users to gain a comprehensive view of system operations.

Figure 2 illustrates the teacher's flowchart, showing the workflow from authentication using the employee ID (NIP) to the management of learning activities. After successfully logging in, the system displays the teaching schedule and available class lists. The teacher then selects a class and determines the type of activity to manage, such as learning materials, assignments, or quizzes. For learning materials, the teacher can upload content and store it in the database. In assignment management, the teacher can create new assignments, review student submissions, provide grades and feedback, which are then stored as updates in the system. For quizzes, the teacher can create a new quiz by setting the title, duration, and question bank, or modify the status of an existing quiz. All teacher activities are stored within the system, ensuring that classroom management is structured and integrated.

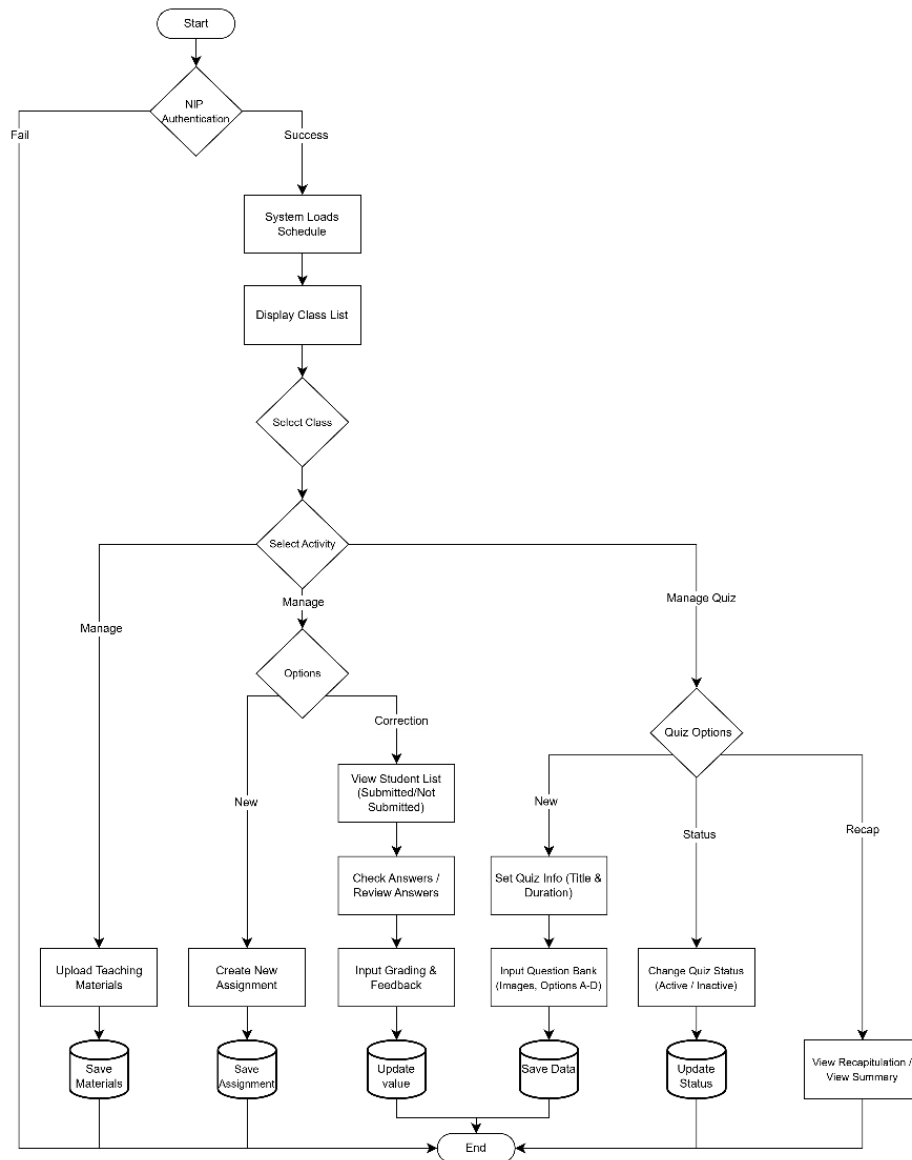


Figure 2: Teacher Flowchart

Figure 3 illustrates the administrator’s flowchart, showing the workflow starting from the admin authentication process and followed by the display of the main dashboard. The administrator then selects the management module to be accessed, which includes user data management, academic data, teacher assignment, scheduling, and announcements.

In the user data module, the administrator can add, modify, or delete data for teachers and students. In the academic data module, the administrator manages class and subject information to be used in the learning process. The assignment module allows the administrator to map teachers to specific classes, while the scheduling module is used to set the days and times of learning sessions. The

administrator can also create announcements that are either global or targeted to specific users. All updated data are stored in the database to ensure that the system remains consistent and up to date.

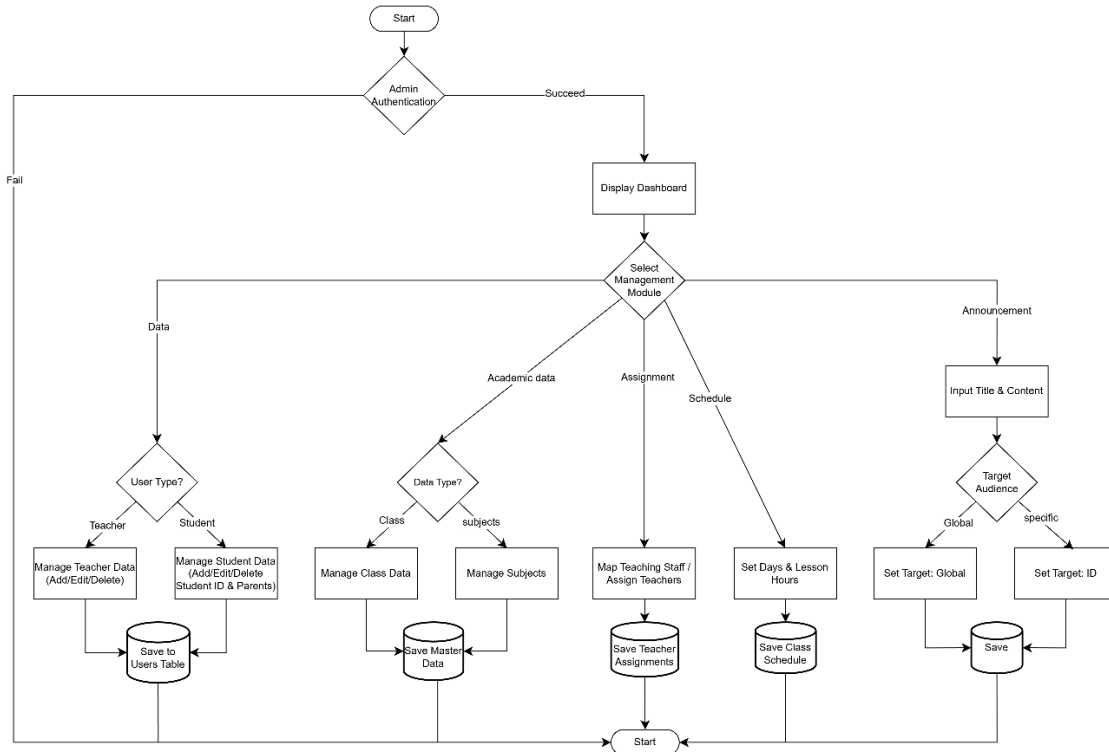


Figure 3: Administrator Flowchart

Figure 4 below illustrates the student flowchart, explaining the workflow of system usage by students, starting from authentication using their NISN. After successful login, the system validates the class and loads data to be displayed on the dashboard, including announcements and a list of assignments approaching their deadlines. Students then select the available menu options, such as learning materials, schedule, assignments, or quizzes. In the learning materials menu, students can choose a subject to view and read the learning content. In the assignments menu, students can view assignment details, upload their work if it has not been submitted, or check grades and feedback from the teacher if it has been evaluated. In the quizzes menu, students select an active quiz, complete the questions within the allocated time, and submit their answers for processing by the system. Quiz scores are displayed in real time once saved by the system. This workflow ensures that all student activities are properly recorded and supports a structured learning process.

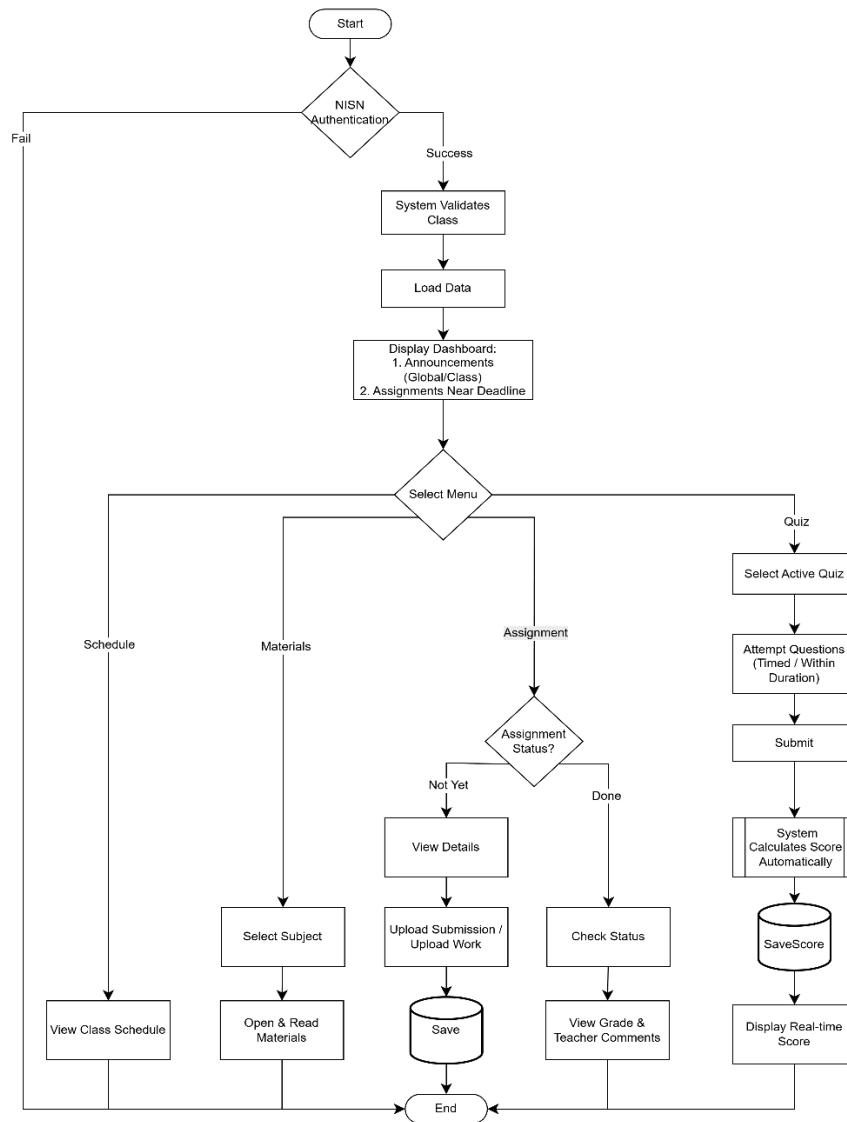


Figure 2: Student Flowchart

b. Entity Relationship Diagram (ERD)

Figure 3 illustrates the Entity Relationship Diagram (ERD), which consists of several main entities: Roles, Users, Profiles, Classes, Subjects, Teacher_Assignments, Student_Classes, Announcements, Tasks, Materials, Class_Schedules, Quizzes, Quiz_Questions, Submissions, and Student_Scores. Each entity contains attributes that describe the characteristics of data within the e-learning system, ranging from user management, classes, and subjects to learning activities. The relationships between entities represent structured connections, such as Users linked to Roles to define access rights, and a one-to-one relationship between Users and Profiles to store personal information. Teachers can teach multiple subjects and classes through

the Teacher_Assignments entity, while students are associated with classes via Student_Classes.

Learning activities such as announcements, materials, assignments, class schedules, and quizzes are directly related to the classes taught by teachers. The Tasks, Materials, Class_Schedules, and Quizzes entities are part of the learning workflow that students can access according to their respective classes. Assessment of learning outcomes is managed through the Submissions and Student_Scores entities, which serve as a link between students and their assignments or quizzes. With this design, the ERD supports structured and integrated data management and ensures that digital learning processes run more effectively and accurately.

Entity Relationship Diagram - Sistem E-Learning SD

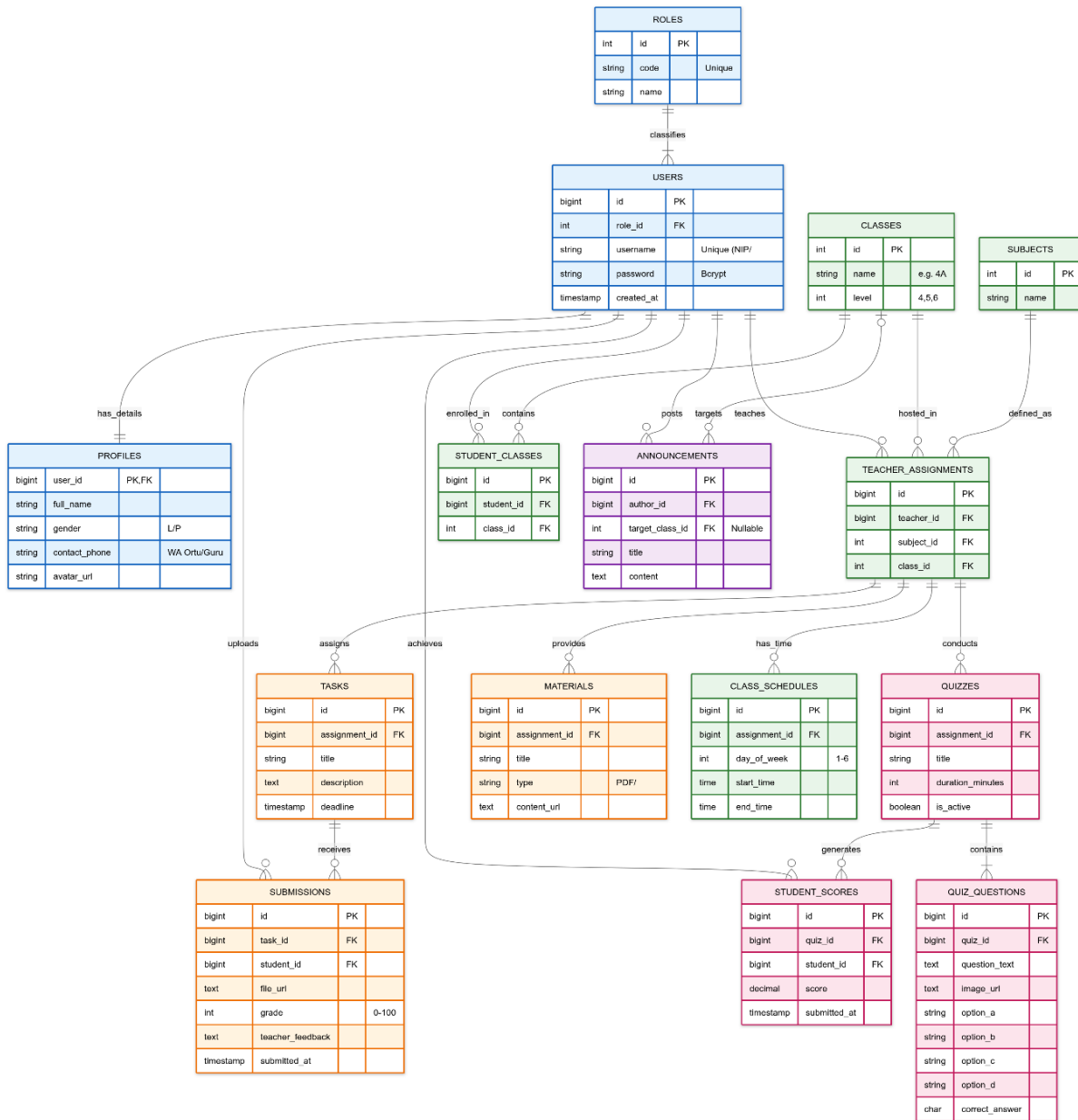


Figure 3: Entity Relationship Diagram

4. Implementation

The implementation stage is the realization of the previously defined interface and system logic. The system was built using a separated frontend-backend architecture, where the backend is developed with Golang, the web interface for administrators and teachers uses React.js, and the mobile interface for students is built with Flutter. The following are the implemented interfaces for each type of user:

a. Teacher

The Teacher Login Interface (Figure 4) serves as the primary authentication gateway for educators before accessing the system. The page features a minimalist design with the clear identity of SD Negeri Tengah 08 Jakarta to provide institutional context. Teachers are required to enter their Employee Identification Number (NIP) as the username and a valid password to ensure data security and validity. Additionally, demo account information is provided at the bottom of the login panel to facilitate functional testing of the system by users or administrators.

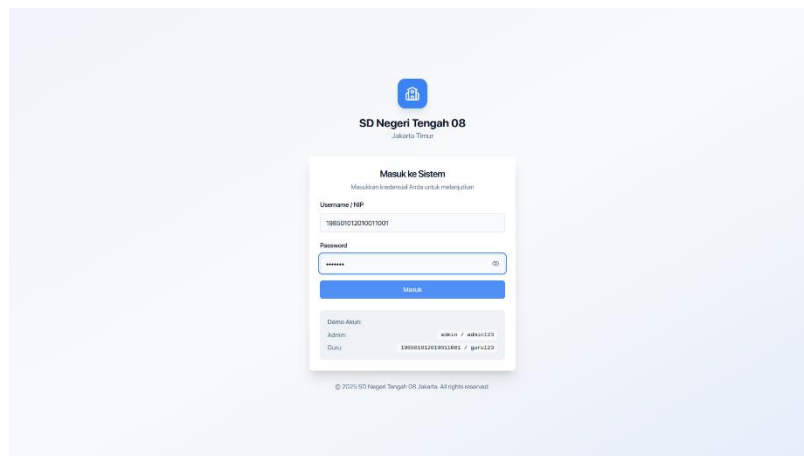


Figure 4: Admin Login Page

The teacher dashboard interface is designed to centralize all learning activities within a single main view. As shown in Figure 5, the Dashboard page displays a list of classes assigned to the teacher according to the mapping set by the administrator. Each class card contains information on the day and time of the lesson and provides quick access to three main modules: Materials, Assignments, and Quizzes. This design is structured to facilitate navigation and help teachers manage multiple classes more efficiently.

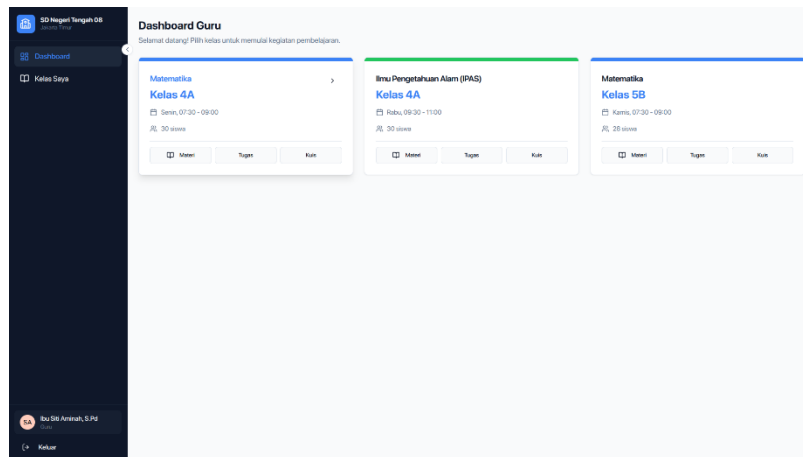


Figure 5: Teacher Dashboard Interface

The Materials module (Figure 6) was developed to facilitate the digital distribution of learning resources. In this interface, teachers can upload and manage various types of learning materials, such as text documents (PDF), videos, and external links. Support for multiple formats is intended to enrich learning resources and accommodate the diverse needs and learning styles of students.

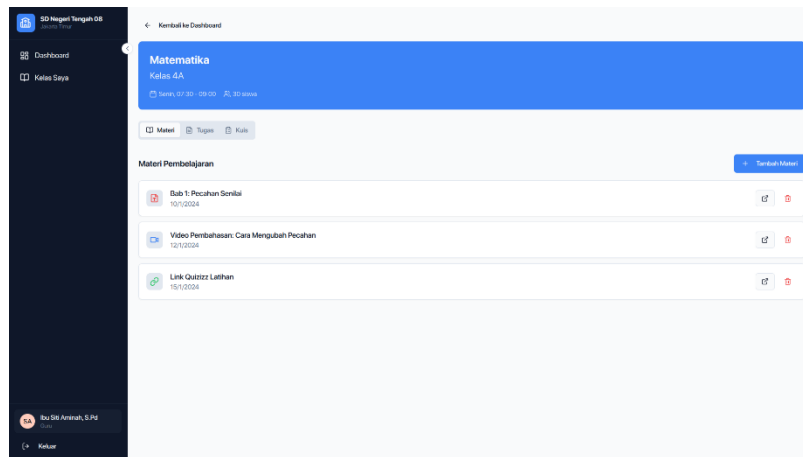


Figure 6: Learning Materials Management Interface

The Assignments Management feature (Figure 7) enables teachers to manage assignments in a structured manner. Teachers can create new assignments by specifying instructions and submission deadlines. The system automatically monitors students' submission statuses, categorizing them as submitted, not submitted, or late. This module also provides features for grading and giving individual feedback, ensuring that the evaluation process remains effective, even in remote learning conditions.

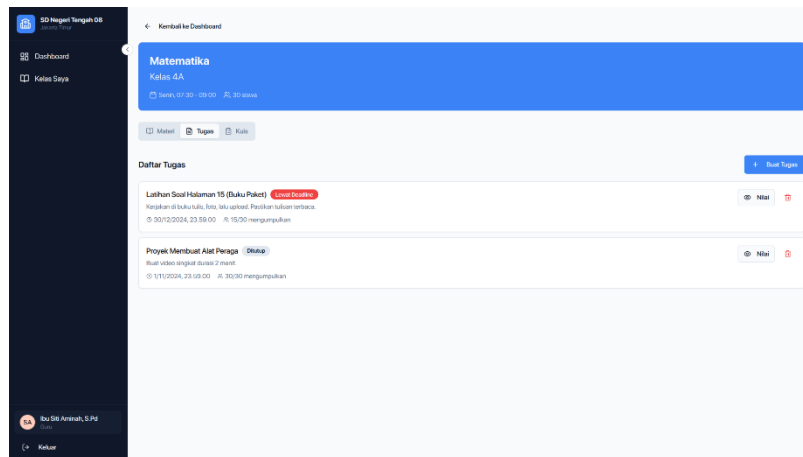


Figure 7: Student Assignment Management Interface

The Quiz Management module (Figure 8) is provided to support multiple-choice based evaluation. Teachers can create questions through a question bank, set the duration for the quiz, and activate or deactivate quizzes by adjusting their status. The system also features automatic grading, which calculates and records student scores immediately after the quiz is completed, reducing the administrative burden on teachers in the grading process.

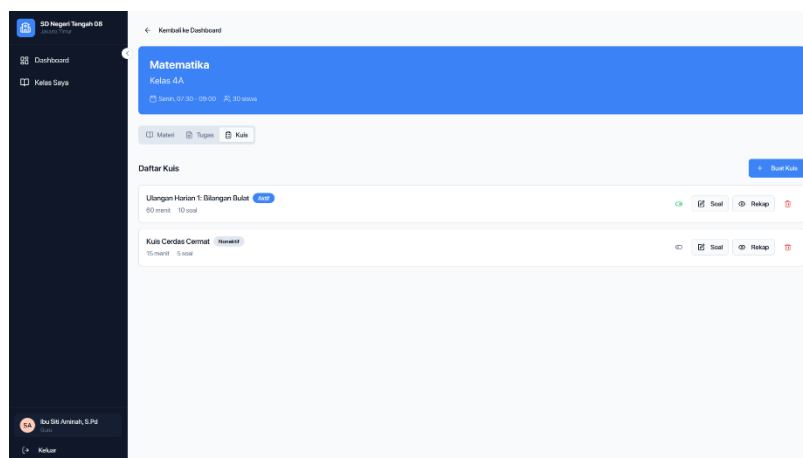


Figure 8: Quiz Management Interface

b. Admin

The administrator dashboard page (Figure 9) is the initial interface accessed by the administrator after completing the authentication process. This page is designed to present real-time system statistics. The main components of the dashboard include information cards displaying the total number of teachers, students, classes, and subjects, as well as panels for recent announcements and today's schedule. This data visualization aims to provide the administrator with an executive overview of the system's operational status.

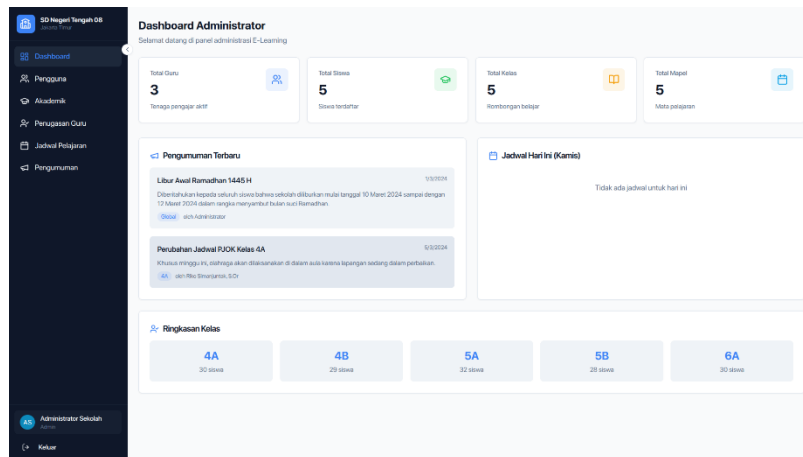


Figure 9: Administrator Dashboard Interface

The User Management module (Figure 10) serves as the central administration hub for master data of teachers and students. In this interface, the administrator can perform CRUD operations (Create, Read, Update, Delete) on user data. The data table displays key attributes such as Employee/Student ID (NIP/NISN), Full Name, Gender, and Contact Information. The implementation of this feature ensures the validity of user data that will interact within the e-learning ecosystem.

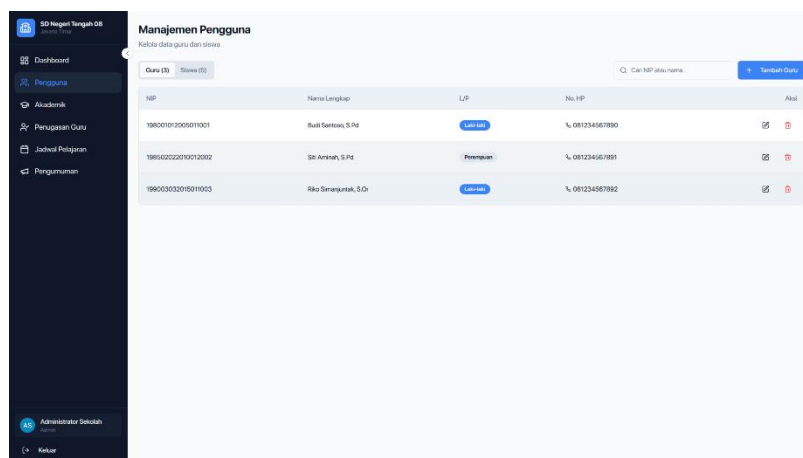


Figure 10: User Management Interface

In Figure 11, the administrator can configure class data, including determining grade levels and student capacity per class. This interface is designed with a minimalist layout and structured class lists to facilitate the administrator in mapping students to appropriate classes before the academic year begins.

Data Akademik
Kelola data kelas dan mata pelajaran

Kelas (ID) Mata Pelajaran (ID) [Tambah Kelas](#)

Nama Kelas	Tingkat	Jumlah Siswa	Aksi
4A	Kelas 4	30 siswa	Edit Hapus
4B	Kelas 4	29 siswa	Edit Hapus
5A	Kelas 5	32 siswa	Edit Hapus
5B	Kelas 5	28 siswa	Edit Hapus
6A	Kelas 6	30 siswa	Edit Hapus

Figure 11: Class Data Management Interface

The Teacher Assignment feature (Figure 12) is designed to facilitate the scheduling of teachers, subjects, and target classes. This feature is crucial to ensure that each subject in every class has an assigned teacher, which subsequently affects the teacher’s access rights to upload materials and evaluate assignments.

Perugasan Guru
Kelola perugasan guru ke mata pelajaran dan kelas

[Tambah Perugasan](#)

Nama Guru	Mata Pelajaran	Kelas	Aksi
Siti Aminah, S.Pd	Matematika	4A	Edit Hapus
Siti Aminah, S.Pd	IPS	5B	Edit Hapus
Rika Smerjantika, S.Or	BK/UK	6A	Edit Hapus

Figure 12: Teacher Assignment Management Interface

Class schedule management is implemented through the Class Schedule feature (Figure 13). On this page, the administrator sets the allocation of teaching and learning activities based on parameters such as day, time, subject, and assigned teacher. The interface allows the administrator to view the existing schedule and make adjustments in case of changes, minimizing schedule conflicts in school operations.

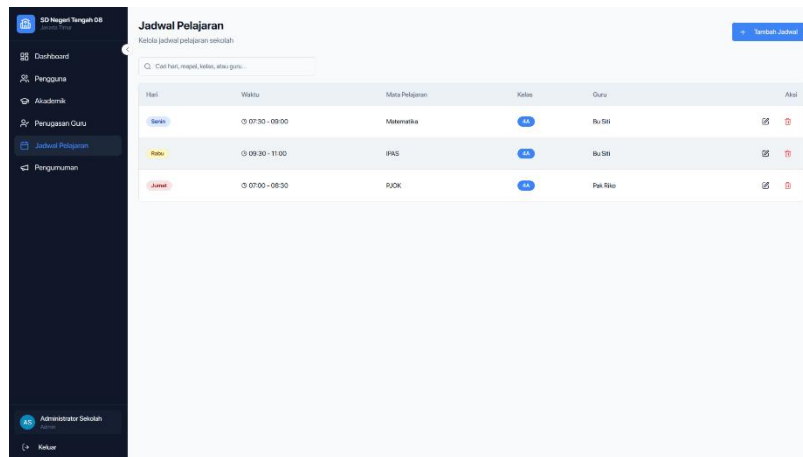


Figure 13: Class Schedule Management Interface

The system is equipped with an Announcements module (Figure 14) to allow administrators to create, edit, and delete school notifications. A key feature of this module is the flexibility in targeting the audience, where announcements can be set as "Global" for all users or specific to certain classes. This mechanism ensures that both academic and non-academic information is delivered accurately to the relevant students and teachers.

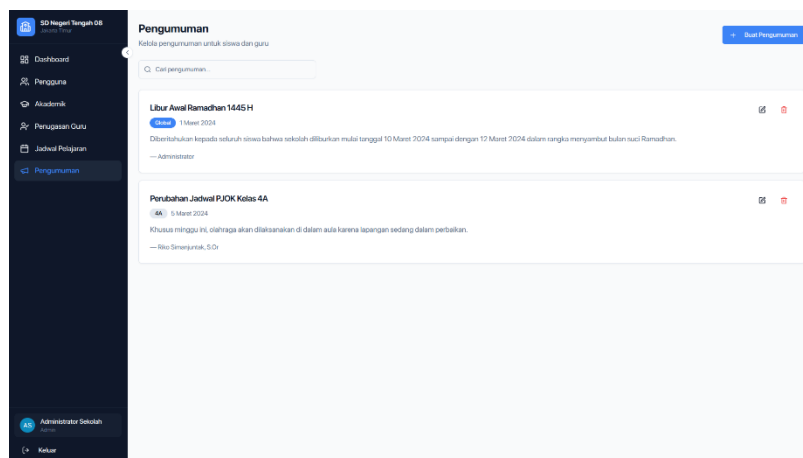


Figure 14: Announcements Management Interface

c. Student

The implementation of the student interface begins with the integration of the Login and Home pages (Figure 15). During authentication, the system uses a combination of NISN and password to validate user access through a minimally designed login interface, reducing the likelihood of input errors for elementary school students. Once credentials are verified, users are directed to the Home page, which is designed with a user-friendly approach and presents personalized information, such as a greeting to the student, an "Urgent Assignments" card as a reminder of upcoming

deadlines, and a school announcements board. This design aims to help students prioritize their learning activities from the moment they open the application.

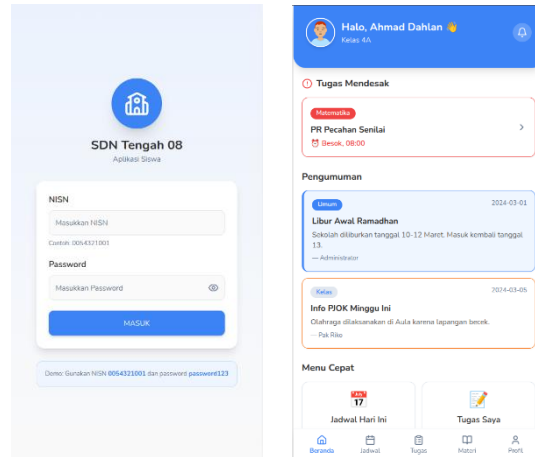


Figure 15: Student Login and Home Interface

The Schedule feature (Figure 16) functions as a digital agenda that displays the sequence of subjects by day. Information on this page includes class time, subject name, and the assigned teacher. With this feature, students can organize their daily learning activities more efficiently without needing to refer to printed schedules or manual notes.

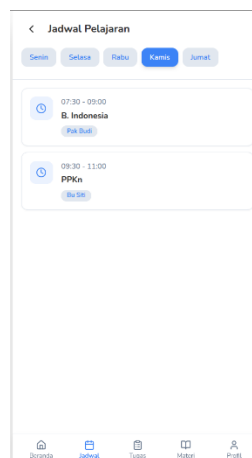


Figure 16 Class Schedule Interface

The Materials menu (Figure 17) allows students to access all learning resources uploaded by teachers. The materials are organized by subjects, such as Mathematics, Science, and Indonesian Language, facilitating easy navigation and quick access to the required content. The available resources include digital documents and instructional videos, enabling students to study and review the materials flexibly at any time.

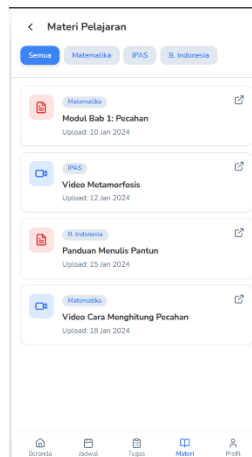


Figure 17: Learning Materials List Interface

The My Assignments feature (Figure 18) displays a list of all assignments that students need to complete, along with their completion status. Task statuses are shown using visual indicators such as Not Started, Awaiting Grade, and Graded. Once an assignment has been graded, students can immediately view their scores through the corresponding assignment card. This feature is designed to help students monitor their assignment progress in a more organized manner.

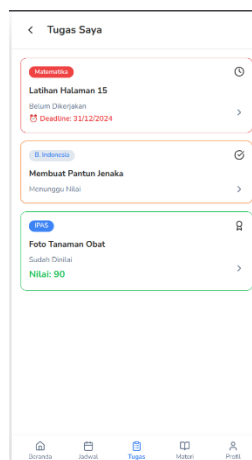


Figure 18: Student Assignments List Interface

Learning evaluation is conducted through the Interactive Quiz feature. On the quiz-taking page (Figure 19), students answer multiple-choice questions accompanied by a timer indicating the time limit. Once all questions are completed, the system automatically processes the answers and displays the Quiz Results page. On this page, students receive instant feedback, including their final score, the number of correct and incorrect answers, and visual appreciation elements to encourage motivation and engagement in learning.

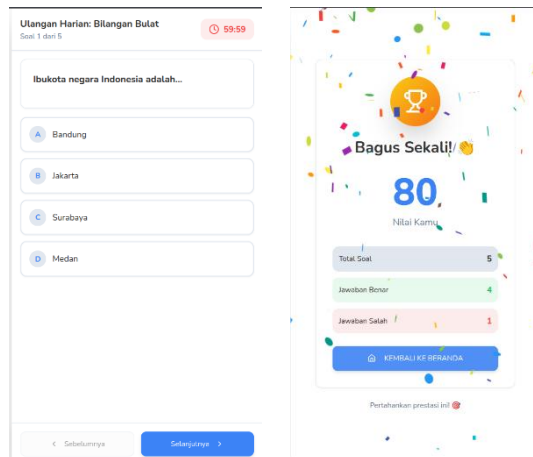


Figure 19: Quiz Taking and Results Interface

5. Testing

The system was tested using the Black Box Testing method, focusing on verifying whether each function on the Administrator, Teacher, and Student interfaces produces outputs in accordance with the specified requirements.

Table 1: Black Box Testing Scenarios

No	Question	Expected Result	Results
1.	User Data Management (Add/Edit/Delete Teachers & Students)	New user data is saved in the database and the user list is updated in real-time.	as expected
2.	Class Schedule and Teacher Assignment Settings	The schedule is saved and automatically appears on the accounts of the corresponding teachers and students.	as expected
3.	Creating School Announcements	The announcement is successfully published and displayed on the Home page of both Teacher and Student applications.	as expected
4.	Uploading Learning Materials (PDF/Video)	Learning material files are successfully uploaded and can be accessed/downloaded by students in the corresponding class.	as expected
5.	Creating New Assignments and Setting Deadlines	Assignments appear in the students assignment list with status "Not Started" and the correct deadline.	as expected
6.	Inputting Student Assignment Grades	Grades entered by teachers are saved and the student assignment status changes to "Graded".	as expected

7.	Creating Quizzes and Activating Quiz Status	Quizzes are successfully created with multiple-choice questions and can be accessed by students once the status is activated.	as expected
8.	Student Login on Mobile Application	The app loads student data, announcements, and the assignment list according to the student's class.	as expected
9.	Checking Class Schedule	The app displays the daily subject schedule accurately.	as expected
10.	Accessing Learning Materials	Students can open and view learning videos or read PDF documents.	as expected
11.	Taking Quizzes (Timer & Scoring)	Students can select answers, the timer counts down, and the final score appears immediately after completion (auto-grading).	as expected

DISCUSSION

The results of this study indicate that the integration of a React.js-based web platform and a Flutter-based mobile application, both connected through a Golang backend, has successfully improved the efficiency of the learning process at SD Negeri Tengah 08 Jakarta. Real-time data synchronization across platforms enables teachers, students, and administrators to interact within a unified and consistent digital ecosystem. These findings align with the work of (Purnama et al., 2023), who emphasize the importance of interoperability in elementary-level e-learning, despite the fact that most previous studies still focused on single-platform implementations. This result is further supported by the study of (Stevanus Ocktoberrikho & Noviyanti P, 2025), which demonstrates that Android-based mobile learning can enhance accessibility and student independence in basic education environments.

Furthermore, Black Box Testing conducted on the system shows that all core features including NISN or NIP authentication, learning material management, assignment distribution, and automated grading operate in accordance with the expected functional parameters. This finding is consistent with the study by (Nuryanto et al., 2021), which states that functional validity is a key indicator of successful e-learning implementation in elementary schools. Nevertheless, this research also identifies limitations related to students' dependence on device availability and internet access at home, as highlighted in the study by (Muhammad Syahrul & Febri Yuliani, 2023), who emphasize that the effectiveness of online learning is significantly influenced by device readiness, network stability, and parental support for young learners. Moving forward, the development of learning analytics features, a parent dashboard, and automated essay assessment

capabilities may further enhance the system's functionality and support the sustainability of digital learning ecosystems in elementary education.

CONCLUSION

This study successfully developed and implemented an integrated multi-platform e-learning system for SD Negeri Tengah 08 Jakarta, combining a React.js-based web interface for teachers and administrators with a Flutter-based mobile application for students. The system consolidates essential functions such as NISN-based authentication, assignment management, learning material distribution, automated quizzes, and school information channels into a single cohesive digital ecosystem, enhancing both administrative efficiency and flexible learning access.

Black Box Testing confirmed that all critical system functions operate as expected, ensuring real-time synchronization, accurate data processing, and reliable performance across platforms. The user-friendly interface effectively supports elementary students' engagement while reducing teachers' administrative workload. These results demonstrate the potential of cross-platform integration and user-centered design in creating effective and scalable digital learning environments for elementary education.

This research provides a practical solution for SD Negeri Tengah 08 Jakarta and a reference framework for other schools and developers seeking interoperable e-learning systems. Future enhancements, including learning analytics, parent dashboards, and automated evaluation tools, can further strengthen the system's functionality, sustainability, and impact on adaptive, inclusive, and high-quality digital learning practices.

REFERENCES

- Adisty Yudianto Putri, D., Cahyo Wibowo, N., & Karunia Farista Ananto, P. (2025). Implementasi Pengujian Blackbox Menggunakan Automation Testing Tools Pada Website Monitoring Balita. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(3), 4278–4285. <https://doi.org/10.36040/jati.v9i3.13627>
- Amin Rahim Hidayat, M., Nura Izzati, N., Rizky Putra Prirhatama, A., Patria, Y., & Kurmilasari, S. (2025). Pengujian Perangkat Lunak Pada Website Ka'cake: Implementasi Unit Testing, Integration Testing, System Testing, Dan Validation Testing Untuk Menjamin Kualitas Dan Keandalan Sistem. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(4), 6805–6811. <https://doi.org/10.36040/jati.v9i4.13969>
- Auliya Rahma, M., Nabilla, I., Khasanah, K., Sintanil Asri, N., Alvi Nadia, I., Khumaedy, A., & Abdurrahman Wahid Pekalongan, U. K. (2024). *Transformasi Dinamika Metode Konvensional ke Digital pada Pembelajaran di MA Pembangunan Jakarta* (Vol. 03, Issue 03).
- Bismoputro, I., Al Huda, F., & Hendra Brata, A. (2017). *Pengembangan Single Page Application Berbasis Reactjs Untuk Usaha Percetakan Online (Studi Kasus: Global Grafika)* (Vol. 1, Issue 1). <http://j-ptiik.ub.ac.id>
- Calista Chandra, Wijaya, F., Gunawan, J. A., Lee, J. R., & Maulana, A. (2024). Perancangan dan Implementasi RESTful API untuk Aplikasi Mobile Pembelajaran Flora dan

- Fauna pada Google Cloud Platform. *SATESI: Jurnal Sains Teknologi Dan Sistem Informasi*, 4(1), 58–69. <https://doi.org/10.54259/satesi.v4i1.2850>
- Harjoseputro, Y., Albertus Ari Kristanto, & Joseph Eric Samodra. (2020). Golang and NSG Implementation in REST API Based Third-Party Sandbox System. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 4(4), 745–750. <https://doi.org/10.29207/resti.v4i4.2218>
- Jinan, A., Siregar, M. P., Suryani, D. F., Rolanda, V., & Muis, A. (2025). Desain dan Rancang Bangun Sistem E-Learning Menggunakan Framework Laravel Berbasis WEB. *ROUTERS: Jurnal Sistem Dan Teknologi Informasi*, 85–93. <https://doi.org/10.25181/rt.v3i2.4182>
- Muhammad Syahrul, & Febri Yuliani. (2023). Efektivitas Pembelajaran Daring Berbasis Digital Classroom Bagi Siswa Sekolah Dasar (SD) Islam Al-Azhar 54 Pekanbaru. *Jurnal Pendidikan Indonesia*, 1(3), 72–76. <https://doi.org/10.62007/joupi.v1i3.80>
- Nuryanto, S., Muzanil, Y. A., & Masya, F. (2021). SISTEM INFORMASI E-LEARNING BERBASIS ANDROID UNTUK TINGKAT SEKOLAH DASAR (STUDI KASUS : SDI AL-HADIRIYAH). *Jurnal Sistem Informasi, Teknologi Informatika Dan Komputer*, 11(3), 44–52. <https://jurnal.umj.ac.id/index.php/just-it/index>
- Palandi, E. H., Sriyuliawati, F., Aziz, A., Malang, P. N., Al, U. I., Kuningan, I., & Semarang, U. I. (2025). Peran Teknologi dalam Pengembangan Sistem E-Learning yang Interaktif dan Efektif bagi Pendidikan. *Journal Scientific of Mandalika*, 6(7), 2809–0543.
- Piter, D., & Lee, F. S. (2025). Rancang Bangun Aplikasi Pelacak Keuangan Berbasis Ocr Untuk Manajemen Pengeluaran Departemen Keuangan. *JIKA (Jurnal Informatika)*, 9(2), 238. <https://doi.org/10.31000/jika.v9i2.13637>
- Purnama, H. I., Wilujeng, I., & Abdul Jabar, C. S. (2023). Web-based E-learning in Elementary School: A Systematic Literature Review. *JOIV : International Journal on Informatics Visualization*, 7(3), 749. <https://doi.org/10.30630/joiv.7.3.1203>
- Rahmadhani, S., Wildana, D. W., Arumdanie, H. W., & Hakim, L. (2024). Penerapan React JS dan Axios untuk Pengembangan Front-end Aplikasi iCare. *Software Development Digital Business Intelligence and Computer Engineering*, 2(02), 40–46. <https://doi.org/10.57203/session.v2i02.2024.40-46>
- Samsul Hadi, Liling Tryasmono, Deni Setiawan, Gunawan Gunawan, Kunti Nastiti, Noval Noval, & Yusri Yusri. (2023). Kegiatan Pendampingan Pembuatan Teh Herbal Bawang Dayak (E.bulbosa), Sungkai (P.canescens) Limpasu (B.lancolata) di GKN, Tanah Bumbu dengan Pendekatan CPOTB. *Jurnal Informasi Pengabdian Masyarakat*, 1(4), 94–99. <https://doi.org/10.47861/jipm-nalanda.v1i4.550>
- Sopandi, A., Hannan, A. R., & Khotimah, H. (2024). Perancangan Aplikasi Mobile Menggunakan Framework Flutter Pada Sistem Informasi Akademik. *JIKA (Jurnal Informatika)*, 8(3), 304. <https://doi.org/10.31000/jika.v8i3.11402>
- Stevanus Ocktoberrikho, & Noviyanti P. (2025). Pengembangan Platform E-Learning Berbasis Android untuk Siswa Sekolah Dasar. *Instink: Inovasi Pendidikan, Teknologi Informasi Dan Komputer*, 4(1), 1–11. <https://doi.org/10.30599/8dkacm46>

- Sudrajat, V. V., Adityo, R. D., & Arizal, A. (2025). Simulation of the Single Sign-On Method for Service Provider Applications: A Case Study of Bhayangkara University Surabaya. *JITE (Journal of Informatics and Telecommunication Engineering)*, 1(9), 36–47. <https://doi.org/10.31289/jite.v9i1.13750>
- Syahputra, R. A., & Aryanto, J. (2025). Pengembangan Sistem Rekam Medis Digital Multiplatform Berbasis Tauri Framework. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 5, 1547–1558. <https://doi.org/10.57152/malcom.v5i4.2332>
- Wisnu Shena Arrafi, & Putra, R. E. (2025). Implementasi dan Manajemen State Pada Website Next.js: Perbandingan Context API dan Redux Pada Website Mangaice. *Journal of Informatics and Computer Science (JINACS)*, 6(04), 1131–1144. <https://doi.org/10.26740/jinacs.v6n04.p1131-1144>
- Wulandari, A. S., Saepudin, A., Kinanti, M. P., Sudesi, Z., Saifudin, A., & Yulianti, Y. (2022). Pengujian Aplikasi Sistem Informasi Akademik Berbasis Web Menggunakan Metode Black Box Testing Equivalence Partitioning. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 5(2), 102. <https://doi.org/10.32493/jtsi.v5i2.17561>
- Wulansari, S., & Suhartini, R. (2024). Pengembangan Media Pembelajaran Berbasis E-Learning di SMK Negeri 1 Bojonegoro. *Jurnal Pendidikan Madrasah*, 9(2), 251–256. <https://doi.org/10.14421/jpm.2024.251-256>