

PERANCANGAN SISTEM MANAJEMEN PENGETAHUAN BERBASIS WEB UNTUK PENGELOLAAN PENGETAHUAN SEKOLAH TK ABC

Ananda Dwi Fajar¹⁾, Aprizal Ferdinand Saragi²⁾

1) Sistem Informasi Universitas Machung, Villa Puncak Tidar N-1 Malang, Indonesia

2) Sistem Informasi Universitas Machung, Villa Puncak Tidar N-1 Malang, Indonesia

email : 322210003@student.machung.ac.id¹⁾, 322210026@student.machung.ac.id²⁾

Abstrak

Penelitian ini mengembangkan prototipe Sistem Manajemen Dokumen berbasis web untuk memfasilitasi distribusi pengetahuan eksplisit antar-guru di tingkat Taman Kanak-Kanak (TK). Dengan pendekatan SDLC Waterfall, sistem dibangun menggunakan React untuk antarmuka pengguna, Node.js sebagai platform backend, serta Firebase sebagai basis data NoSQL dan penyimpanan file. Fitur utama mencakup unggah dokumen disertai metadata, klasifikasi berdasarkan kategori pembelajaran, dan pencarian cepat melalui kata kunci, kategori, dan rentang tanggal. Kontrol akses berbasis peran menjamin keamanan dan privasi dokumen.

Kata Kunci :

Firebase, Node.js, React, SDLC Waterfall, Sistem Manajemen Pengetahuan

Abstract

This study presents the development of a web-based Document Management System prototype aimed at facilitating the distribution of explicit knowledge among kindergarten teachers. Employing a Waterfall SDLC approach, the system was implemented using React for the frontend, Node.js for the backend, and Firebase as the NoSQL database and file storage solution. Key functionalities include document upload with descriptive metadata, categorization by learning topics, and swift search capabilities via keywords, categories, and date filters. Role-based access control ensures document security and privacy.

Keywords :

Document Management System; Firebase; Knowledge Management; Node.js; React

1. PENDAHULUAN

Dalam dunia pendidikan, khususnya di jenjang prasekolah, pengetahuan yang tersimpan dalam bentuk dokumen memegang peran vital sebagai fondasi dan panduan bagi seluruh rangkaian proses belajar-mengajar. Setiap dokumen—mulai dari silabus, rencana pelajaran harian, modul materi ajar, hingga catatan evaluasi perkembangan siswa—mencerminkan hasil pengamatan, pemikiran, dan inovasi strategi pedagogis yang telah teruji oleh pengalaman [1]. Keberadaan dokumen-dokumen ini tidak hanya sekadar catatan administratif, melainkan juga media transfer pengetahuan antar-guru, referensi kurikulum, serta bukti audit internal terhadap praktik pembelajaran.

Namun dalam praktik sehari-hari, dokumen-dokumen pengajaran sering tersebar di berbagai media yang tidak terorganisir, seperti folder digital individual, lembar kerja kertas di ruang guru, penyimpanan eksternal seperti *flash disk*, atau bahkan aplikasi penyimpanan *cloud* pribadi. Fragmentasi ini menghasilkan keterbatasan aksesibilitas ketika guru membutuhkan materi secara cepat, meningkatkan risiko duplikasi dokumen, serta menghambat proses

kolaborasi dan refleksi instruksional [2]. Situasi tersebut menjadi semakin rumit ketika terjadi rotasi guru atau penggantian staf, karena banyak informasi berharga yang tidak terdokumentasi secara terpusat dan berpotensi hilang.

Dalam konteks lembaga Taman Kanak-Kanak (TK), kompleksitas pengelolaan dokumen bertambah berkat keberagaman format media pembelajaran. Selain teks dan angka yang mendeskripsikan rencana dan catatan progres perkembangan anak, guru juga menggunakan gambar hasil karya anak, audio perekaman interaksi verbal, dan video dokumentasi kegiatan bermain yang sarat nilai pedagogis [3]. Kemampuan untuk menangani berbagai format file ini menjadi tantangan tersendiri: foto karya anak harus dapat dipratinjau dengan cepat tanpa perlu mengunduh file berukuran besar, sedangkan catatan audio memerlukan pemutar langsung di antarmuka sistem. Ketidadaan sistem terpusat membuat guru menghabiskan waktu untuk mencari dan memindahkan file di berbagai lokasi, sehingga waktu yang seharusnya digunakan untuk perencanaan dan interaksi langsung dengan siswa menjadi tereduksi [4].

Untuk menjawab tantangan tersebut, sistem manajemen dokumen (*Document Management System/DMS*) diusulkan sebagai solusi strategis yang menyediakan repositori terpusat. DMS memungkinkan unggahan berbagai jenis dokumen ke dalam platform terpadu, dilengkapi dengan *metadata* deskriptif seperti judul materi, kategori pembelajaran, kata kunci, nama pengunggah, dan tanggal unggahan. Solusi manajemen pengetahuan mengacu pada cara-cara mengelola aspek-aspek manajemen pengetahuan, yaitu penemuan pengetahuan, penangkapan pengetahuan, berbagi pengetahuan, dan penerapan pengetahuan [5], [6]. *Metadata* tersebut memfasilitasi klasifikasi dokumen secara sistematis, memudahkan proses pencarian, serta memastikan bahwa setiap materi dapat diakses kembali dengan cepat dan akurat [7]. Lebih jauh, sistem ini mencakup mekanisme versi (*version control*) yang melacak riwayat perubahan dokumen, memberikan transparansi terhadap modifikasi, serta peluang untuk mengembalikan versi sebelumnya jika diperlukan.

Keunggulan penerapan DMS tidak terbatas pada efisiensi pencarian dan klasifikasi. Dengan kontrol akses berbasis peran (*Role-Based Access Control/RBAC*), sistem memastikan bahwa hanya guru terdaftar yang memiliki izin untuk menambah, mengubah, atau menghapus dokumen [8]. Hal ini mencegah pemanfaatan materi secara tidak sah serta menjaga kerahasiaan konten, khususnya dokumen yang memuat informasi pribadi siswa. Peran tambahan seperti administrator DMS dapat memantau alur unggah dan pemeliharaan dokumen, memberikan lapisan pengawasan administrasi yang terstruktur dan terdokumentasi [9].

Dalam kerangka operasional, implementasi DMS di lingkungan TK berpusat pada tiga pilar utama: unggah, klasifikasi, dan pencarian. Pilar unggah menyediakan antarmuka sederhana yang mendukung *drag-and-drop* atau seleksi *file* manual untuk berbagai format—PDF, JPEG/PNG, dokumen teks, serta *file* presentasi. Sistem melakukan validasi tipe *file* dan batasan ukuran sesuai infrastruktur jaringan sekolah, sehingga proses unggah berjalan lancar tanpa memerlukan pengetahuan teknis mendalam [10]. Pilar klasifikasi mengagregasi *metadata* yang diinput oleh pengguna, menghadirkan struktur folder virtual yang dapat disesuaikan dengan taksonomi pembelajaran, mulai dari tema aktivitas hingga level perkembangan anak. Pilar pencarian mengimplementasikan algoritma penelusuran berbasis *full-text* dan filtrasi lanjutan, memfasilitasi kombinasi kata kunci serta filter tanggal, kategori, dan nama pengunggah.

Implementasi DMS juga menawarkan manfaat signifikan dalam memfasilitasi kolaborasi antar-guru. Melalui visibilitas dokumen yang diunggah rekan sejawat, guru dapat mempelajari contoh rencana pembelajaran yang efektif, metode evaluasi yang teruji, serta inovasi kegiatan belajar yang atraktif. Meskipun sistem ini tidak menyertakan fitur forum diskusi terintegrasi, notifikasi real-time terhadap unggahan baru dan revisi dokumen membantu menjaga komunikasi informal antar-guru [11].

Aspek teknologi menjadi faktor penting dalam kesuksesan DMS. Pilihan teknologi berbasis web responsif memungkinkan akses melalui berbagai perangkat, termasuk desktop, laptop, tablet, atau *smartphone*, tanpa memerlukan instalasi perangkat lunak khusus. Penyimpanan *cloud* menjadi solusi untuk mengatasi keterbatasan infrastruktur lokal, menawarkan skalabilitas kapasitas sesuai pertumbuhan dokumen, serta mekanisme *backup* otomatis untuk menjamin keamanan data jangka panjang. Pada sisi server, arsitektur mikroservis atau *service-oriented* memungkinkan pengembangan fitur terpisah—seperti modul unggah, modul pencarian, dan modul autentikasi—tanpa memengaruhi keseluruhan sistem [12].

Desain antarmuka pengguna (UI/UX) berperan sentral dalam memastikan adopsi sistem oleh guru yang mungkin tidak memiliki latar belakang teknis. Prinsip desain minimalis dan ikon yang intuitif memudahkan navigasi, dengan tata letak yang konsisten di seluruh halaman unggah, daftar dokumen, dan hasil pencarian. Tombol aksi utama seperti "Unggah Dokumen" atau "Refresh Daftar" ditempatkan strategis di area tampilan utama, sementara panel filter ditempatkan di sisi kiri untuk akses cepat. Penggunaan warna netral dan kontras tinggi membantu memperjelas status unggahan—misalnya indikator progres bar hijau yang menampilkan persentase unggahan.

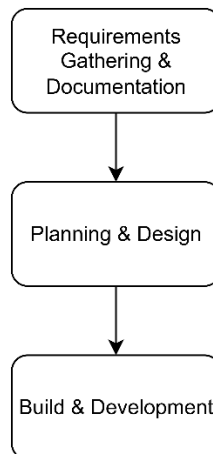
Penerapan metode pengembangan *Software Development Life Cycle* (SDLC) *Waterfall* memungkinkan dokumentasi menyeluruh di setiap fase: analisis kebutuhan, perancangan, implementasi, dan dokumentasi teknis [13]. Pendekatan linier ini memudahkan verifikasi artefak, meminimalkan revisi besar di tengah pengembangan, serta memastikan bahwa tim pengembang memiliki pemahaman yang komprehensif tentang kebutuhan sistem sebelum menulis kode. Pengujian formal dan uji terima pengguna (*User Acceptance Testing/UAT*) tidak digunakan pada tahap pengembangan [14].

Kontribusi penelitian ini terletak pada penyusunan prototipe DMS yang menitikberatkan pada kemudahan penggunaan, fleksibilitas *metadata*, dan skalabilitas penyimpanan *cloud*. Dengan antarmuka yang sederhana, sistem diharapkan mendorong budaya berbagi pengetahuan antar-guru, meningkatkan akuntabilitas dokumen, serta mempercepat proses perencanaan dan evaluasi pembelajaran. Ke depan, penelitian ini membuka peluang pengembangan modul analitik untuk memantau tren penggunaan dokumen, integrasi dengan *Learning Management System* (LMS), dan penerapan fitur keamanan tambahan seperti enkripsi *end-to-end* dan autentikasi multi-faktor [15].

2. METODE

Penelitian ini menggunakan model SDLC *Waterfall* karena metode ini menyediakan kerangka kerja yang jelas dengan aliran linier antar-tahapan. Pendekatan ini dipilih karena memastikan bahwa setiap fase diselesaikan secara menyeluruh sebelum melanjutkan ke fase berikutnya,

sehingga memudahkan tim dalam melakukan verifikasi dan validasi artefak. Dengan rancangan yang matang sejak awal, risiko revisi besar selama implementasi dapat diminimalkan, memungkinkan komunikasi yang lebih terstruktur di antara anggota tim dan memastikan stabilitas fungsi sistem.



Gambar 1. Alur Diagram Waterfall

Pada fase *Requirements Gathering & Documentation*, tim peneliti melakukan observasi mendalam terhadap proses pengelolaan dokumen yang berjalan. Observasi ini mencakup aktivitas pembuatan dokumen, penyusunan *metadata*, serta alur verifikasi dan arsip dokumen. Setiap detail proses—termasuk aktor yang terlibat, masukan yang diperlukan, dan keluaran yang dihasilkan—dicatat secara sistematis dalam format narasi proses. Hasil observasi kemudian diformulasikan ke dalam dokumen Spesifikasi Kebutuhan Sistem (SRS) yang merinci kebutuhan fungsional seperti mekanisme unggah *file*, pengelompokan berdasarkan kategori atau atribut, fungsi pencarian berbasis *metadata*, dan manajemen versi dokumen. Selain itu, SRS juga memuat kebutuhan non-fungsional berupa standar keamanan data, target waktu respons untuk operasi dasar, dan ketersediaan layanan. Dokumen SRS ini menjadi pedoman utama bagi tim pengembang dan menjadi tolok ukur keberhasilan pada fase selanjutnya.

Fase *Planning & Design* memfokuskan pada penerjemahan dokumen SRS ke dalam rancangan teknis yang terukur. Tim merancang arsitektur sistem berlapis, terdiri dari lapisan presentasi untuk antarmuka pengguna, lapisan logika bisnis yang menangani aturan alur proses, dan lapisan akses data yang mengatur penyimpanan dan pengambilan dokumen. Pada lapisan presentasi, dibuat *wireframe* interaktif yang memperlihatkan tampilan *form* unggah dokumen, daftar arsip, serta fitur pencarian dan filter yang memudahkan navigasi. Diagram alir proses CRUD dokumen dirinci untuk menunjukkan langkah pengguna, validasi data, dan mekanisme notifikasi. Skema konseptual data menggambarkan struktur entitas dokumen, relasi antarentitas, serta model penyimpanan *metadata* dalam basis data. Desain ini juga mencakup definisi mekanisme autentikasi, pengelolaan hak akses, dan enkripsi data sesuai standar keamanan yang ditetapkan.

Pada fase *Build & Development*, seluruh rancangan yang telah disetujui diimplementasikan menjadi aplikasi fungsional. Tim memulai dengan menyiapkan lingkungan pengembangan dan *repository* kode, menetapkan standar pengkodean dan sistem versi kontrol. Implementasi antarmuka dilakukan dengan membangun komponen modular sesuai *wireframe*, memperhatikan prinsip aksesibilitas dan responsivitas tampilan. Di sisi *backend*, logika bisnis dan API dirancang untuk memproses permintaan CRUD, melakukan validasi *input*, dan menangani autentikasi serta otorisasi pengguna. Struktur penyimpanan data diatur sesuai skema yang telah dirancang, dengan konfigurasi *metadata* dan *file* yang sesuai. Setiap modul kode disertai dengan dokumentasi internal yang menjelaskan parameter, alur eksekusi, dan contoh respons API.

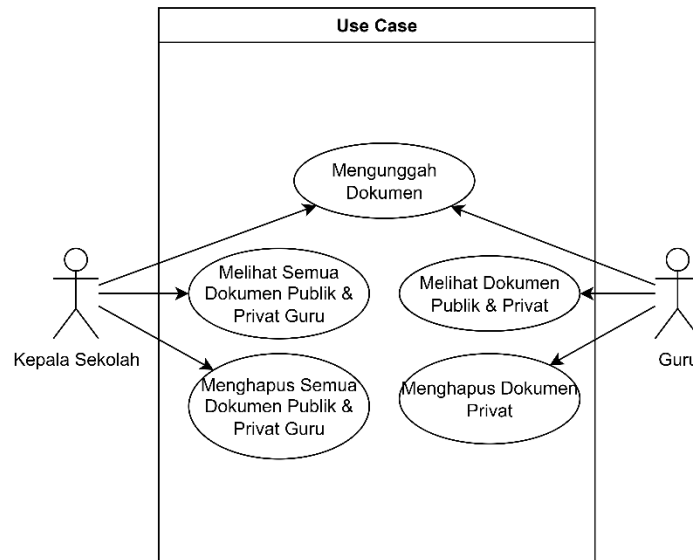
Dengan mengikuti alur *Waterfall* yang sistematis dan terdokumentasi dengan baik, penelitian ini berhasil menghasilkan artefak spesifikasi, desain, dan kode yang terstruktur. Pendekatan ini memudahkan penelusuran kembali setiap keputusan desain dan menyediakan dasar yang kuat untuk pengembangan lanjutan atau perbaikan sistem di masa mendatang.

3. HASIL DAN PEMBAHASAN

Pada tahap *Requirements Gathering & Documentation*, tim pengembang melakukan observasi menyeluruh terhadap alur kerja pengelolaan dokumen harian. Aktivitas yang dicatat meliputi penyusunan rencana pelajaran, pembuatan laporan perkembangan anak, pencatatan daftar hadir, dan pengarsipan surat izin orang tua. Hasil observasi diolah menjadi dokumen Spesifikasi Kebutuhan Sistem (SRS) yang menjabarkan kebutuhan fungsional—unggah dokumen, pengelompokan per siswa, dan pencarian *metadata*—serta kebutuhan non-fungsional seperti keamanan data dan performa layanan. Kelengkapan SRS memastikan bahwa semua fitur inti sudah terdefinisi dengan jelas sebelum perancangan dimulai.

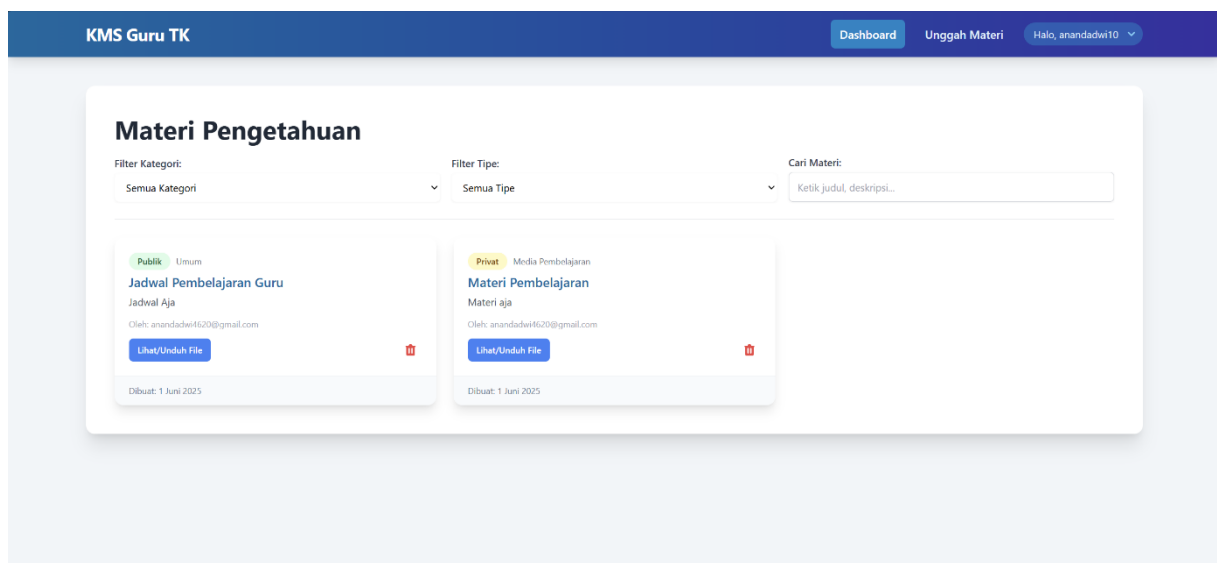
Fase *Planning & Design* mengambil SRS sebagai acuan untuk merumuskan rancangan teknis sistem secara menyeluruh. Arsitektur sistem dibagi menjadi tiga lapis: *presentation layer* yang mendefinisikan sketsa antarmuka pengguna, *business logic layer* yang mengatur alur validasi dan kontrol izin, serta data *access layer* yang menjelaskan struktur penyimpanan dokumen. Diagram alir proses CRUD dan skema konseptual data menggambarkan urutan langkah pengguna dari pemilihan file hingga penyimpanan *metadata*, sekaligus relasi antarentitas dokumen. Dokumen desain ini menjadi cetak biru bagi tim pengembang sehingga memudahkan proses implementasi.

Pada tahap *Build & Development*, rancangan diimplementasikan menggunakan *React* untuk membangun antarmuka yang responsif, *Node.js* sebagai platform *backend* untuk menangani permintaan API, dan *Firebase* sebagai *database NoSQL* untuk menyimpan *metadata* serta *file* dokumen. *React* dipilih karena kemampuannya dalam merender komponen antarmuka secara dinamis dan mendukung *update real-time* tanpa memuat ulang halaman. *Node.js* memungkinkan pembuatan server yang ringan dan efisien, menggunakan *JavaScript* di sisi server untuk mengelola logika bisnis dan *routing API*. *Firebase* menyediakan layanan *Cloud Firestore* untuk penyimpanan data terstruktur, *Firebase Authentication* untuk otentikasi pengguna, dan *Firebase Storage* untuk menyimpan *file*. Kombinasi ketiganya mempercepat siklus pengembangan prototipe.



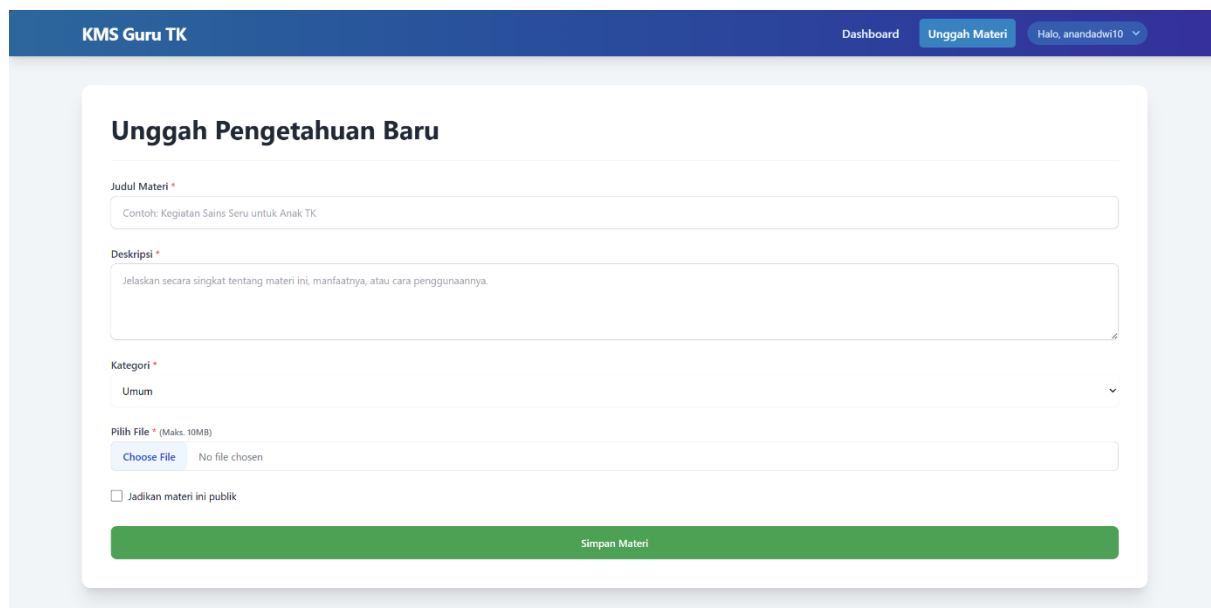
Gambar 2. Use Case

Use case diagram menjelaskan interaksi antara aktor Kepala Sekolah dan Guru dengan fungsi utama sistem. Kepala Sekolah memiliki wewenang penuh untuk mengunggah dokumen, melihat semua dokumen publik dan privat milik guru, serta menghapus dokumen apa pun. Guru hanya dapat mengunggah dokumen, melihat dokumen publik bersama dan dokumen privat miliknya, serta menghapus dokumen privat yang tidak lagi diperlukan. Pembagian hak akses ini diterapkan melalui konfigurasi *Firebase Authentication* dan *security rules* pada *Firebase Storage* serta *Cloud Firestore*.



Gambar 3. Dashboard KMS

Prototipe aplikasi web yang dideploy menampilkan dua halaman utama. Halaman “Materi Pengetahuan” menyajikan daftar dokumen dalam bentuk kartu, lengkap dengan filter kategori, filter tipe *file*, dan kolom pencarian (Gambar 2). Setiap kartu menampilkan label visibilitas—Publik atau Privat—judul dokumen, nama pengunggah, tombol “Lihat/Unduh File”, dan ikon hapus dokumen. Halaman ini memudahkan guru dan kepala sekolah dalam menelusuri, mengunduh, atau menghapus materi.



Gambar 4. Halaman Mengunggah Dokumen

Halaman “Unggah Pengetahuan Baru” menampilkan formulir interaktif berisi *input* judul materi, deskripsi singkat, pilihan kategori dari daftar yang telah ditentukan, seleksi *file* (maksimal 10 MB), dan *checkbox* untuk menandai visibilitas publik (Gambar 3). Setelah pengguna menekan tombol “Simpan Materi”, *metadata* dikirim ke Cloud Firestore, sementara *file* dokumen diunggah ke *Firestore Storage*.

Pembahasan ilmiah menunjukkan bahwa pendekatan linier *Waterfall* memastikan kesinambungan dan keterpaduan antarfase. Dokumen SRS yang matang meniadakan kebutuhan revisi fitur besar di tengah pengembangan, sedangkan desain berlapis mendukung modularitas dan kemudahan pemeliharaan. Implementasi langsung menggunakan *React*, *Node.js*, dan *Firebase* mempercepat realisasi prototipe, tetapi uji informal mengungkapkan peningkatan latensi saat mengunggah *file* berukuran besar. Temuan ini menegaskan perlunya optimasi, seperti *caching* lokal dan *batch upload*, serta pelaksanaan pengujian formal dan UAT sebelum penerapan skala penuh.

Selain aspek teknis dan antarmuka, pengujian informal terhadap guru sebagai pengguna sistem memberikan wawasan penting terkait adopsi teknologi di lingkungan pendidikan prasekolah. Mayoritas guru menyatakan bahwa antarmuka sistem cukup intuitif dan tidak memerlukan pelatihan intensif untuk mulai digunakan. Fitur pencarian yang dilengkapi filter

kategori dan tanggal dianggap sangat membantu dalam menemukan dokumen pembelajaran yang relevan, terlebih ketika menghadapi jadwal pengajaran yang padat dan dinamis. Namun, beberapa pengguna menyarankan penambahan fitur *preview* langsung di dalam aplikasi untuk meminimalkan proses unduh-unggah ulang ketika terjadi kesalahan *file*.

Selain itu, sistem menunjukkan stabilitas dan performa yang baik dalam skenario penggunaan normal. Akan tetapi, tantangan muncul ketika dilakukan unggahan serentak oleh beberapa pengguna atau saat koneksi jaringan tidak stabil. Kondisi ini berdampak pada kegagalan unggah sebagian dokumen yang kemudian tidak tercatat dalam repositori. Hal ini menunjukkan pentingnya pengembangan fitur *fail-safe* seperti *retry* otomatis atau sistem antrian unggahan. Dari sisi pengelolaan, *dashboard* admin juga disarankan untuk dilengkapi statistik aktivitas pengguna agar kepala sekolah dapat memantau seberapa aktif guru dalam berbagi dokumen dan berkolaborasi melalui sistem.

4. KESIMPULAN

Prototipe Sistem Manajemen Dokumen yang dikembangkan dalam penelitian ini telah berhasil menghadirkan platform terpusat yang sederhana namun efektif untuk berbagi pengetahuan eksplisit antar-guru TK melalui fitur unggah, klasifikasi, dan pencarian dokumen yang responsif, meminimalkan duplikasi data dan menjaga keamanan konten dengan kontrol akses berbasis peran; penerapan model SDLC *Waterfall* memastikan keteraturan alur pengembangan dan kejelasan artefak di setiap fase, sedangkan integrasi teknologi *React*, *Node.js*, dan *Firebase* menghasilkan prototipe yang ringan, skalabel, dan mudah dirawat; meski pengujian awal bersifat informal, kinerja sistem sudah memenuhi kebutuhan operasional harian, dan untuk pengembangan lanjutan, direkomendasikan pelaksanaan uji formal dan *User Acceptance Test*, optimasi proses unggah *file* besar, penambahan modul analitik untuk memantau tren penggunaan, serta integrasi dengan LMS dan penerapan keamanan ekstra seperti enkripsi *end-to-end* dan autentikasi multi-faktor untuk memperluas fungsi dan meningkatkan perlindungan data.

Penelitian ini menghasilkan sebuah prototipe sistem manajemen pengetahuan berbasis web yang dirancang khusus untuk mendukung pengelolaan dokumen dan distribusi pengetahuan eksplisit di lingkungan sekolah Taman Kanak-Kanak. Dengan menggunakan pendekatan *Waterfall*, sistem ini berhasil melalui tahapan pengembangan mulai dari perumusan kebutuhan, desain sistem, hingga implementasi awal menggunakan teknologi *React*, *Node.js*, dan *Firebase*. Fitur unggulan seperti unggah dokumen dengan *metadata*, klasifikasi kategori, pencarian cepat, serta kontrol akses berbasis peran telah mampu menyelesaikan berbagai permasalahan fragmentasi dokumen dan kesulitan aksesibilitas yang sebelumnya dihadapi para guru.

Sistem ini tidak hanya memberikan solusi teknis terhadap penyimpanan dan pencarian materi, tetapi juga memfasilitasi kolaborasi informal antar-guru melalui visibilitas dokumen bersama dan notifikasi pembaruan. Penerapan arsitektur tiga lapis (*presentation*, *logic*, dan *data access layer*) juga memungkinkan sistem dikembangkan secara modular dan dapat ditingkatkan di masa mendatang. Hasil uji coba informal menunjukkan adanya peningkatan efisiensi kerja guru, terutama dalam proses perencanaan pembelajaran dan evaluasi.

Ke depan, sistem ini dapat dikembangkan lebih lanjut dengan menambahkan fitur analitik untuk memantau tren penggunaan dokumen, integrasi ke *Learning Management System* (LMS), serta penerapan teknologi keamanan tambahan seperti enkripsi end-to-end dan autentikasi multi-faktor. Selain itu, pengujian formal seperti *usability testing* dan evaluasi performa perlu dilakukan guna memperoleh validasi yang lebih komprehensif. Dengan begitu, sistem ini memiliki potensi besar untuk diadopsi tidak hanya di lingkungan TK ABC, namun juga oleh institusi pendidikan lain dengan kebutuhan serupa.

5. REFERENSI

- [1] A. C. Vallberg-Roth, *Assessment and documentation in early childhood education*. 2014. doi: 10.4324/9781315818504.
- [2] K. Cowan dan R. Flewitt, "Moving from paper-based to digital documentation in Early Childhood Education: democratic potentials and challenges," vol. 9760, 2023, doi: 10.1080/09669760.2021.2013171.
- [3] S. Sumiyati dan S. Wijayanti, "Manajemen Pengelolaan Kelas Di Kelompok a Tk Bintang Kecil Sukoharjo Kecamatan Margorejo Kabupaten Pati," *AWLADY: Jurnal Pendidikan Anak*, vol. 6, no. 1, hlm. 124, 2020, doi: 10.24235/awlad.v6i1.5622.
- [4] T. H. Khotimah, N. S. Rokaibah, S. Zahrotunnisa, M. W. Firdaus, R. Wiqi, dan C. Rochman, "Persepsi Guru Terhadap Tantangan Dan Peluang Pendidikan," *NATURALISTIC: Jurnal Kajian Penelitian Pendidikan dan Pembelajaran*, vol. 4, no. 2a, hlm. 514–520, 2020, doi: 10.35568/naturalistic.v4i2a.690.
- [5] Riswanto dan D. I. Sensuse, "Knowledge management systems development and implementation: A systematic literature review," dalam *IOP Conference Series: Earth and Environmental Science*, IOP Publishing Ltd, Apr 2021. doi: 10.1088/1755-1315/704/1/012015.
- [6] K. Iskandar, T. Tony, C. H. Phankova, dan W. Agustino, "Perancangan Knowledge Management System Pada IT Bina Nusantara menggunakan Blog, Wiki, Forum dan Document," *ComTech: Computer, Mathematics and Engineering Applications*, vol. 5, no. 1, hlm. 110, 2014, doi: 10.21512/comtech.v5i1.2595.
- [7] H. Yudiastuti dan I. Irwansyah, "Perancangan Knowledge Management System Pada Kantor Dinas Pendidikan Pemda Ogan Ilir," *Jurnal Ilmiah Matrik*, vol. 20, no. 3, hlm. 234–246, 2019, doi: 10.33557/jurnalmatrik.v20i3.471.
- [8] N. H. Wendi Usino, Teddy Mantoro, Media A. Ayu, "Role Based Access Control for Knowledge Sharing in Remote Areas," *0*, hlm. 15–20, 2010.
- [9] A. Ikhwan, R. A. A. Raof, P. Ehkan, Y. Yacob, dan M. Syaifuddin, "Data Security Implementation using Data Encryption Standard Method for Student Values at the Faculty of Medicine, University of North Sumatra," *Journal of Physics: Conference Series*, vol. 1755, no. 1, 2021, doi: 10.1088/1742-6596/1755/1/012022.
- [10] P. A. Mutia dan S. Ismet, "Pembelajaran Komputer Anak Usia Dini Di Taman Kanak-Kanak Telkom Padang," *Generasi Emas*, vol. 2, no. 1, hlm. 1–12, 2019, doi: 10.25299/ge.2019.vol2(1).3294.
- [11] A. Takyudin, H. Faturrahman, J. Juwono, M. R. Julianto, dan Y. Yulianti, "Analisa dan Perancangan Sistem Informasi Elearning Berbasis Web untuk Memantau Aktivitas Belajar Peserta Didik di SMP Fioretti Tigaraksa," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 4, no. 2, hlm. 110, 2021, doi: 10.32493/jtsi.v4i2.10831.

- [12] A. Novanto, T. E. Siregar, dan T. Hidayat, “Impact of Web Quality on Employees’ Satisfaction in Indonesia State Employment Department,” *Jurnal Ilmiah Informatika Komputer*, vol. 24, no. 3, hlm. 161–169, 2019, doi: 10.35760/ik.2019.v24i3.2278.
- [13] H. Pangestu, H. Alianto, dan S. F. Wijaya, “Hasil Rancang Bangun Sistem Erp Dengan Sdlc Model Waterfall :,” *Jurnal Comtech*, vol. 3, no. 2, hlm. 1036–1042, 2012.
- [14] M. Ridwan, I. Fitri, dan B. Benrahman, “Rancang Bangun Marketplace Berbasis Website menggunakan Metodologi Systems Development Life Cycle (SDLC) dengan Model Waterfall,” *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, vol. 5, no. 2, hlm. 173, 2021, doi: 10.35870/jtik.v5i2.209.
- [15] S. Gunawan, M. Nurwegiono, and R. Setiawan, “Sistem Informasi Bimbingan Belajar RSDC Berbasis Website Dengan Menggunakan Metode Waterfall Untuk Meningkatkan Efisiensi Administrasi”, *RJTI*, vol. 4, no. 3, pp. 424 – 435 , Nov. 2025.