Prosiding Seminar Implementasi Teknologi Informasi dan Komunikasi
Vol. 4, No. 1, Januari 2025
ISSN: 2809-9834, DOI: 10.31284/p.semtik.2025-1.7202

Rancang Bangun Sistem Manajemen Dokumen Perusahaan Menggunakan Object Storage dan Caching, Queue System

Aryana Diaz Cakasana, Ruli Utami*, Sulistyowati

Program Studi Sistem Informasi, Institut Teknologi Adhi Tama Surabaya

*Penulis korespondensi. E-mail: ruli.utami@itats.ac.id

ABSTRACT

PT M is a B2B company specializing in the sale of mining pipes and storage, utilizing digital software such as ERP, CRM, and SCM, alongside an accessible marketplace. Despite the integration of these systems, document management remains complex and often inaccessible when needed. Therefore, a system is required that can manage and share documents independently or connect with other systems, ensuring optimal performance given the large number of documents and users. An agile approach was chosen for the development of this system due to its flexibility and adaptability to changing requirements and audit outcomes, making it the most suitable method to address the evolving needs of the company.

Keywords

Queue System;
File Management;
Enterprise System;
Role Based Authentication;
Object Storage;
Caching

ABSTRAK

PT M adalah perusahaan B2B yang bergerak dalam penjualan pipa pertambangan dan penyimpanan, serta telah mengadopsi perangkat lunak digital seperti ERP, CRM, dan SCM, dan memiliki marketplace yang dapat diakses. Meski sistem-sistem tersebut telah diintegrasikan, manajemen dokumen tetap kompleks dan sering kali sulit diakses saat dibutuhkan. Oleh karena itu, diperlukan sistem yang dapat mengelola dan membagikan dokumen secara mandiri maupun terhubung dengan sistem lain, dengan kinerja yang optimal mengingat banyaknya dokumen dan pengguna. Metode pengembangan sistem dengan pendekatan agile dipilih karena fleksibilitas dan adaptabilitasnya terhadap perubahan kebutuhan dan hasil audit, sehingga dianggap paling sesuai untuk mengatasi kebutuhan yang terus berkembang di perusahaan ini.

PENDAHULUAN

Hampir semua sistem yang terdapat di PT M, sebuah perusahaan B2B yang bergerak di bidang penjualan pipa pertambangan dan penyimpanan, telah mengadopsi perangkat lunak digital. Sistem tersebut mencakup ERP, CRM, dan SCM, serta memiliki marketplace yang dapat diakses. Setiap tahun, sistem ini diaudit dan direncanakan untuk pengembangan dan peningkatan di tahun berikutnya. Namun, dengan banyaknya sistem dan data yang terintegrasi, manajemen dokumen menjadi lebih kompleks. Dokumen yang disimpan, dibagikan, dan beredar di dalam sistem sering kali membuat manajemen dokumen menjadi sulit. Bahkan, terkadang dokumen hilang dan sulit diakses saat dibutuhkan. Dalam beberapa tahun terakhir, berbagai survei dan studi menunjukkan adanya permasalahan serius terkait kinerja aplikasi, terutama di konteks penggunaan bisnis dan organisasi. Salah satu masalah utama yang diidentifikasi adalah aplikasi yang memiliki performa lambat (lemot), yang secara signifikan mengurangi kepuasan pengguna dan menghambat produktivitas [1]. Selain itu, aplikasi yang tidak praktis atau tidak ramah pengguna memperburuk tantangan ini, karena aplikasi-aplikasi tersebut menciptakan kompleksitas yang tidak perlu pada tugas-tugas yang seharusnya sederhana. Antarmuka yang tidak intuitif sering kali menyebabkan frustrasi pengguna, yang pada gilirannya dapat meningkatkan kemungkinan aplikasi ditinggalkan [2]. Permintaan terhadap aplikasi yang dapat terintegrasi secara mulus dengan alur kerja yang ada juga semakin meningkat, namun banyak aplikasi yang gagal memenuhi ekspektasi ini, sehingga mengarah pada ketidakefisienan operasional [3]. Masalah-masalah ini menunjukkan pentingnya perbaikan berkelanjutan dalam desain aplikasi, sehingga aplikasi tidak hanya berfungsi dengan baik, tetapi juga memenuhi standar praktikalitas dan pengalaman pengguna yang dibutuhkan.

Berdasarkan permasalahan di atas, diperlukan suatu sistem yang mampu membagikan dan mengelola dokumen, serta dapat terhubung ke sistem lain. Sistem ini harus mampu beroperasi secara mandiri maupun diintegrasikan dengan sistem yang ada. Mengingat jumlah dokumen dan pengguna yang banyak, performa juga menjadi poin penting dalam sistem ini.Dari data dan

permasalahan yang telah dikumpulkan, metode pengembangan sistem yang menggunakan pendekatan agile dipilih karena sistem yang sudah ada terpecah menjadi beberapa modul kecil yang saling berhubungan. Selain itu, kebutuhan dan persyaratan sistem sering berubah seiring dengan hasil audit, sehingga pendekatan agile yang fleksibel dan adaptif dianggap paling sesuai untuk mengatasi perubahan tersebut.

PT M telah mengadopsi perangkat lunak digital untuk hampir semua sistemnya, termasuk ERP, CRM, dan SCM, serta memiliki marketplace yang dapat diakses [4] .[15] implementasi sistem ERP, CRM, dan SCM dalam sebuah perusahaan dapat meningkatkan efisiensi operasional dan integrasi data, namun juga menghadirkan tantangan dalam manajemen dokumen. Dokumen yang disimpan dan dibagikan dalam sistem sering kali sulit diakses saat dibutuhkan [17]. Menurut [5], kompleksitas manajemen dokumen dapat diatasi dengan sistem yang mampu mengelola dan membagikan dokumen, serta terhubung dengan sistem lain.Pendekatan agile dalam pengembangan sistem telah terbukti efektif dalam menghadapi perubahan kebutuhan dan persyaratan sistem yang sering berubah [6], [7]. Agile menawarkan fleksibilitas dan adaptabilitas yang diperlukan untuk mengatasi perubahan hasil audit dan kebutuhan pengguna [8]. Dengan demikian, metode pengembangan sistem berbasis agile dianggap paling sesuai untuk menghadapi tantangan yang ada di PT M.

TINJAUAN PUSTAKA

Agile dalam Sistem Manajemen Dokumen

Agile merupakan metodologi pengembangan perangkat lunak yang menekankan fleksibilitas, kolaborasi, dan respons cepat terhadap perubahan kebutuhan. Pada sistem manajemen dokumen, pendekatan Agile dapat digunakan untuk meningkatkan efisiensi dalam pengembangan aplikasi manajemen dokumen dan pengelolaan perubahan yang cepat.



Gambar 1 Metode Agile (Sumber: Alexandes, 2022)

Penerapan metodologi Agile pada sistem e-dokumen di kantor pemerintahan menunjukkan bagaimana Agile dapat meningkatkan pengelolaan dokumen dengan lebih adaptif terhadap kebutuhan pengguna [9]. Praktik Agile, khususnya Scrum, memungkinkan tim untuk secara iteratif merancang dan menguji aplikasi manajemen dokumen, sambil terus mendapatkan umpan balik pengguna untuk meningkatkan kualitas sistem secara berkelanjutan [10]. Agile juga mendukung pengelolaan dokumentasi secara efisien dengan mengurangi fokus pada dokumentasi formal dan lebih mengutamakan interaksi langsung antar tim [11], meskipun tetap menjaga dokumentasi yang relevan dan diperlukan dalam konteks tertentu. Dengan integrasi antara Agile dan teknik manajemen dokumen, perusahaan dapat mengelola dokumen secara lebih dinamis dan responsif terhadap kebutuhan perubahan yang cepat.

PHP

PHP adalah bahasa pemrograman yang sangat populer dan telah digunakan secara luas dalam pengembangan aplikasi web, termasuk dalam lingkungan enterprise. PHP menawarkan fleksibilitas, skalabilitas, dan kemudahan integrasi dengan berbagai basis data, menjadikannya ideal untuk pengembangan aplikasi enterprise [4]. Kombinasi PHP dengan metode pengembangan agile meningkatkan efisiensi pengembangan melalui siklus pengembangan yang cepat dan adaptif, memungkinkan PHP untuk mendukung pendekatan yang responsif terhadap perubahan kebutuhan dan persyaratan sistem [6].

PostgreSQL

Ekstensibilitas PostgreSQL memungkinkan integrasi yang mulus dengan sistem yang ada. Stonebraker dan Rowe (1986) menunjukkan bahwa PostgreSQL dapat diubah secara dinamis oleh pengguna melalui dynamic loading, memungkinkan penambahan fungsi-fungsi dan tipe-tipe data baru tanpa harus mengubah kode sumber. Hal ini menjadikan PostgreSQL sangat adaptif terhadap kebutuhan aplikasi yang terus berkembang dan menjadikannya pilihan yang unggul untuk pengembangan sistem enterprise yang membutuhkan fleksibilitas dan kemampuan integrasi tinggi.

Redis

Redis adalah sistem manajemen basis data yang cepat dan sering digunakan untuk caching dalam aplikasi enterprise. Menurut [17], Redis menawarkan performa tinggi dengan latensi rendah, menjadikannya ideal untuk meningkatkan kecepatan akses data dan respons aplikasi. Dengan menggunakan Redis sebagai cache, perusahaan dapat mengurangi beban pada basis data utama, meningkatkan throughput sistem, dan memastikan pengalaman pengguna yang lebih lancar. Fitur seperti persistensi data dan replikasi juga menambah keandalan Redis dalam lingkungan produksi, menjadikannya pilihan yang sangat cocok untuk kebutuhan caching di sistem enterprise

Caching dalam Sistem Manajemen Dokumen dengan Object Storage dan Queue System

Caching adalah teknik yang digunakan untuk meningkatkan kinerja sistem dengan menyimpan data yang sering diakses dalam penyimpanan sementara yang lebih cepat. Pada sistem manajemen dokumen yang menggunakan object storage, penerapan caching sangat penting untuk mengurangi latensi akses data dan meningkatkan efisiensi. Salah satu metode caching yang efektif adalah menggunakan sistem cache terdistribusi seperti Redis, yang dapat digunakan untuk menyimpan data yang sering diminta dan mempercepat pengambilan informasi dari object storage [12]. Selain itu, sistem antrean (queue system) juga berperan penting dalam memanage permintaan data secara efisien, dengan memastikan bahwa data yang diperlukan dikelola dalam urutan yang tepat tanpa menyebabkan tumpang tindih atau keterlambatan akses [13]. Penerapan caching yang terkoordinasi di antara beberapa node penyimpanan juga dapat meningkatkan hit ratio dan mengurangi beban I/O, seperti yang ditunjukkan dalam implementasi sistem penyimpanan terdistribusi dengan C2P middleware [14]. Caching dan queue system yang digabungkan dapat meningkatkan responsivitas sistem manajemen dokumen secara keseluruhan, terutama pada skala besar dengan banyak pengguna.

Object Storage

Sistem file yang biasa digunakan untuk menyimpan data di server rata-rata menggunakan sistem UNIX yang mengimplementasikan POSIX-IO. Konsistensi kuat yang diperlukan oleh semantik POSIX-IO menyebabkan statefulness dengan metadata yang luas. Hal ini memiliki efek samping seperti penurunan kinerja ketika banyak permintaan IO datang dari beberapa node [18]. Untuk mengatasi hal tersebut maka object storage dipilih karena dapat menangani naik turun nya permintaan IO didalam aplikasi enterprise yang tidak tentu

METODE

Perencanaan Alur Pengembangan Sistem

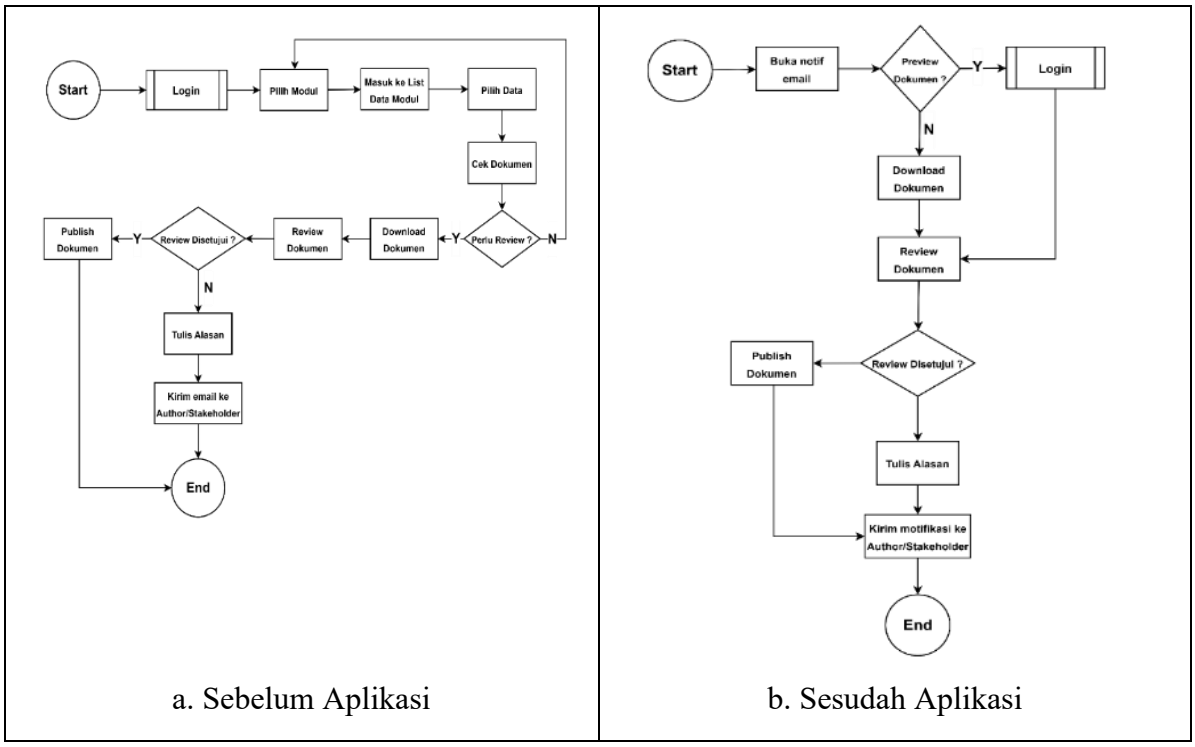
Pengembangan sistem yang efektif memerlukan perencanaan alur yang terorganisir dengan baik. Berikut adalah langkah-langkah yang dapat diikuti dalam perencanaan alur pengembangan sistem:

1. **Perencanaan dan Analisis Kebutuhan** Pada tahap ini, identifikasi kebutuhan pengguna dan rencanakan tujuan sistem. Hal ini dapat dilakukan melalui wawancara dan survei untuk mengumpulkan informasi tentang masalah dan kebutuhan yang ada. Semua kebutuhan yang diidentifikasi harus didokumentasikan dengan baik untuk referensi selanjutnya.
2. **Identifikasi dan Analisis Risiko** Langkah berikutnya adalah mengidentifikasi risiko potensial yang terkait dengan pengembangan sistem, seperti perubahan kebutuhan, masalah teknis, dan batasan anggaran. Setelah itu, lakukan analisis risiko untuk menentukan dampak dan kemungkinan terjadinya risiko, serta merencanakan mitigasi yang tepat.
3. **Desain Prototipe** Berdasarkan kebutuhan yang diidentifikasi, desain prototipe awal sistem. Buat antarmuka pengguna dan alur kerja dasar untuk mendapatkan umpan balik dari pengguna.
4. **Implementasi Prototipe** Kembangkan dan implementasikan prototipe yang telah dirancang. Pengguna kemudian dapat berinteraksi dengan sistem untuk memberikan umpan balik yang konstruktif.
5. **Evaluasi dan Umpan Balik** Kumpulkan umpan balik dari pengguna mengenai prototipe. Identifikasi kekuatan dan kelemahan sistem serta area yang perlu diperbaiki untuk iterasi berikutnya.
6. **Pengulangan Siklus** Berdasarkan umpan balik yang diperoleh, kembali ke tahap perencanaan untuk memperbarui kebutuhan dan desain. Identifikasi risiko baru yang mungkin muncul dan rencanakan mitigasi yang sesuai. Proses ini diulang dari tahap desain hingga evaluasi sampai sistem memenuhi semua kebutuhan pengguna.
7. **Implementasi Final** Setelah beberapa iterasi dan umpan balik, lakukan implementasi akhir dari sistem. Lakukan pengujian menyeluruh untuk memastikan sistem berfungsi dengan baik sesuai dengan spesifikasi.
8. **Pemeliharaan dan Dukungan** Setelah implementasi, berikan dukungan teknis dan pemeliharaan sistem secara berkelanjutan. Terus kumpulkan umpan balik untuk perbaikan lebih lanjut dan penyesuaian sistem di masa mendatang.

Perencanaan alur pengembangan yang terstruktur ini akan membantu memastikan bahwa sistem yang dikembangkan memenuhi kebutuhan pengguna dan dapat berfungsi dengan optimal dalam jangka panjang.

Business Process Diagram

Tabel 3. Bisnis Proses Diagram

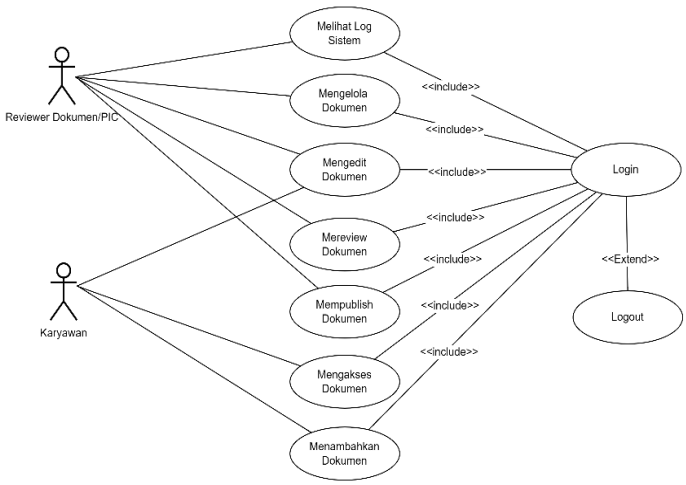


a. Sebelum Aplikasi

b. Sesudah Aplikasi

a. Sebelum Aplikasi: reviewer harus membuka setiap menu/modul (contoh: Modul Ledger, Modul Invoicing) yang terdapat proses pembagian dokumen. Permasalahan yang sering terjadi adalah banyaknya modul memakan waktu yang lama bagi reviewer untuk mencari dokumen mana yang harus di review dan di bagikan **b. Sesudah Aplikasi:** setelah adanya aplikasi maka reviewer dapat melihat dokumen mana saja yang harus direview tanpa perlu mengecek modul satu per satu. Bilamana ada dokumen baru yang perlu di review atau ada perubahan dokumen maka sistem akan mengirim notifikasi ke pihak terkait

Usecase Diagram

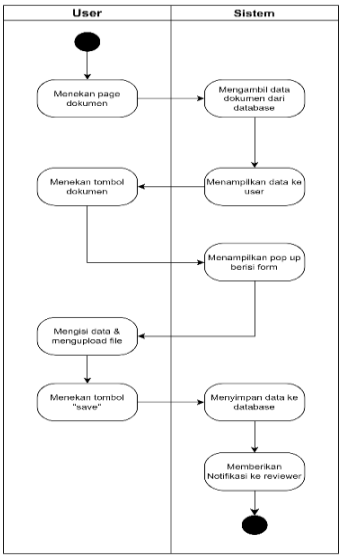


Berisikan setiap fitur yang bisa diakses oleh reviewer dan karyawan. Disini karyawan hanya bisa mengupload dokumen dan mengakses. Karyawan dapat membagikan dokumen tersebut ke tim lain atau keluar organisasi apabila telah di review dan disetujui oleh reviewer/PIC.

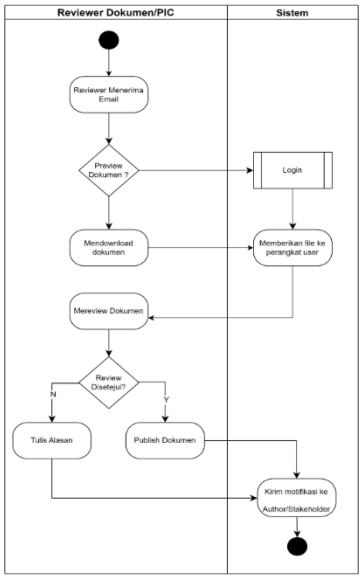
Gambar 2. Usecase Diagram

Activity Diagram

Tabel 3. Activity Diagram



a. Input Dokumen

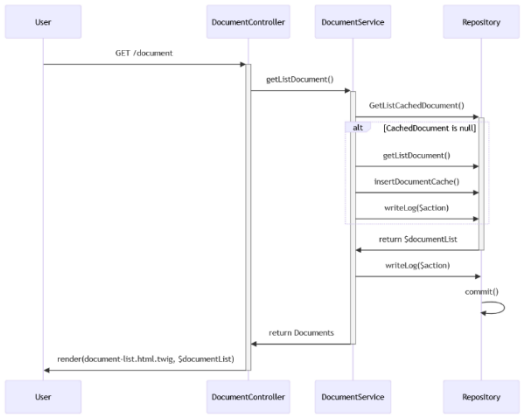


b. Review Dokumen

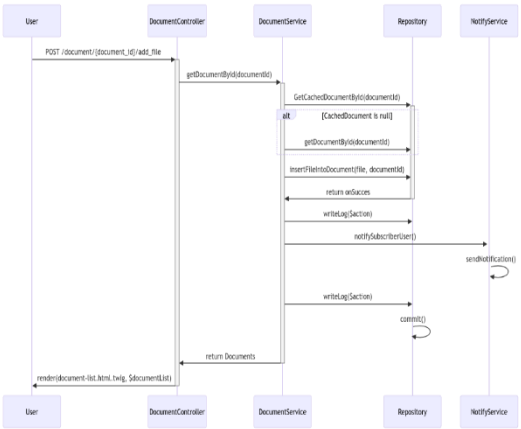
a.Input Dokumen: Setelah login, user memilih dokumen yang tersedia dan sistem menampilkan data dari server. User menambahkan file baru dengan mengisi form dalam popup. Sistem menyimpan file dalam object storage, memperbarui cache, dan mengirim notifikasi ke user terkait.
b. Review Dokumen: Reviewer menerima notifikasi via email, membuka link, dan mulai melakukan review. Jika dokumen disetujui, maka akan dipublikasikan; jika tidak, reviewer memberikan alasan dan notifikasi dikirim ke user terkait.

Sequence Diagram

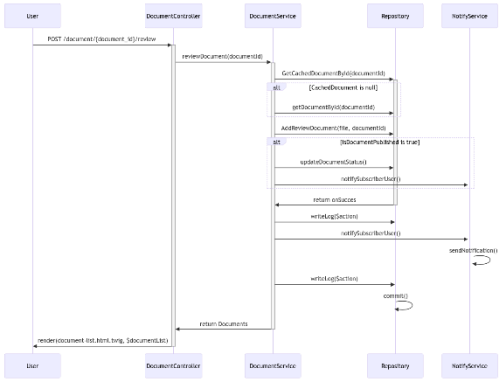
Tabel 4. Sequence Diagram



a. List Document



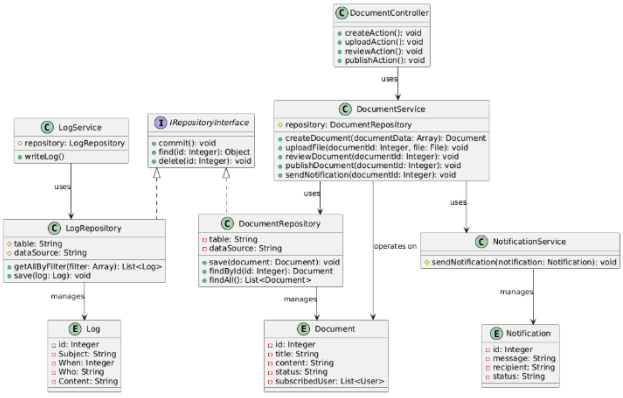
b. Add File Document



c. Review Document

a. List Document: Proses dimulai ketika user membuka url halaman /document. Setelah itu controller akan melakukan pemanggilan function ke service. Disini service bertugas untuk melakukan "Cache Invalidation" yaitu proses pengecakan data ke system cache. Bilamana cache data tersedia maka data cache tersebut akan dikirimkan ke user dengan view bila tidak ada maka system akan melakukan query data ke database**. b. Add File Dokumen**: Proses dimulai ketika user selesai mengisi form dan melakukan submit. Selanjutnya Document Service akan mencari dokumen mana yang perlu di update dan melakukan proses update cache serta mengirimkan notifikasi ke pengguna terkait. **C. Review Dokumen**: proses dimulai ketika user melakukan submit hasil review dokumen .Setelah itu sistem akan melakukan update status pada dokumen dan mengirimkan notifikasi ke pengguna terkait dokumen yang telah di review. Dokumen yang reviewnya telah disetujui dan di publish nantinya akan muncul dengan dokumen lain di proses **List Document**

Class Diagram



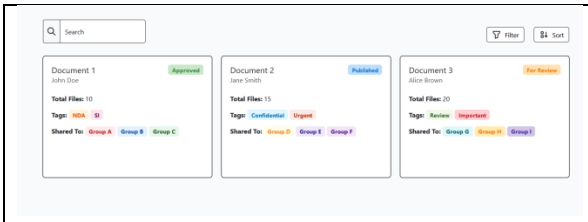
Gambar 6. Class Diagram

Class diagram menunjukkan keterkaitan antar class dan entity data yang saling terhubung menggunakan pattern service repository. Pattern ini dipilih karena bertujuan untuk kemudahan dalam melakukan maintenance (testing, patch, update feature) dan pengaksesan data cache karena sumber data yang akan dijadikan entity telah di handle oleh repository dan setiap bisnis logic yang terjadi dan mengubah state entity akan di handle oleh service

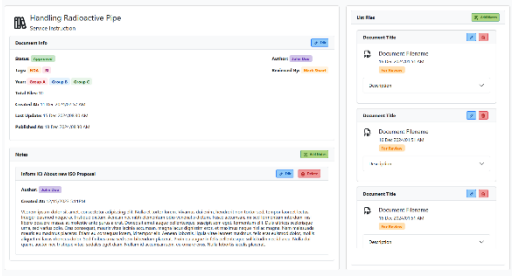
HASIL DAN PEMBAHASAN

Tampilan Program

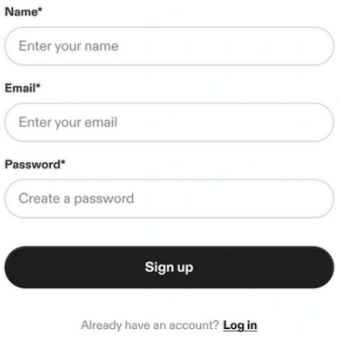
Tabel 5. Tampilan Program



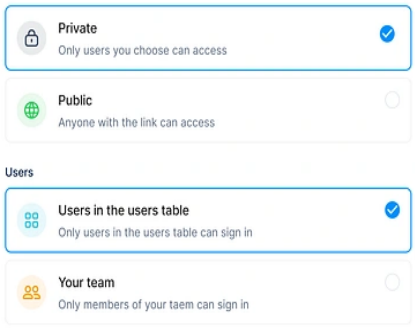
a.Tampilan List Dokumen



b.Tampilan Document & File



c.Tampilan User Login/Sign Up



d.Tampilan Pemilihan Publish Dokumen

a. Tampilan List Dokumen: menampilkan list dokumen beserta beberapa data dan fitur yang bisa diakses oleh user yang sedang login saat ini. **b. Tampilan Document & File:** menampilkan detail dokumen dan file yang terdapat didalam suatu document. Di halaman ini juga user bisa mengedit dan melakukan perubahan terhadap dokumen yang sudah ada. **c. Tampilan User Login/Sign Up:** tampilan yang digunakan untuk meng autentikasi user sebelum masuk kedalam aplikasi. **d. Tampilan Pemilihan Publish Dokumen:** digunakan untuk mengubah target dokumen yang akan di publish. Disini pilihan publish dapat dipilih lebih dari 1

Testing

Tabel 6. Tabel Blackbox Testing

Skenario	Deskripsi	Capaian	Status
Login User	User memasukkan nama pengguna dan kata sandi untuk mengakses sistem.	User berhasil masuk ke halaman list Document	Lulus
Input Dokumen Baru	User memasukan data baru yang akan digunakan sebagai dokumen dan melakukan assignment ke reviewer untuk me review dokumen	User berhasil membuat dokumen dan mengirimkan notifikasi ke reviewer	Lulus
Menambahkan File ke Dokumen	User memasukan file baru kedalam dokumen yang telah tersedia	File berhasil masuk kedalam dokumen dan tersimpan kedalam object storage	Lulus
Review & Publish Dokumen	Reviewer mendownload file yang tersedia dalam dokumen. Dan melakukan publish dokumen setelah review	Dokumen yang di publish dapat diakses oleh target user. User mendapatkan notifikasi	Lulus

Tabel ini berisi hasil pengujian sistem menggunakan metode black-box testing, yang berfokus pada pengujian fungsionalitas sistem berdasarkan masukan (input) dan keluaran (output) yang

diharapkan. Setiap skenario pengujian dirancang untuk memastikan bahwa fitur utama dalam sistem berfungsi sesuai kebutuhan pengguna. Diatas adalah deskripsi rinci untuk setiap scenario yang diuji.

Tabel 7. Survei user terhadap aplikasi sistem manajemen dokumen

No	Kategori Pertanyaan	Pertanyaan	Skala Penilaian (1-5)	Hasil Survei (Rata-rata)
1	Kemudahan Penggunaan	Seberapa mudah aplikasi digunakan untuk mengelola dokumen?	1: Sangat Tidak Puas, 5: Sangat Puas	4,8
2	Kecepatan Akses	Seberapa cepat dokumen dapat diakses melalui aplikasi?	1: Sangat Lambat, 5: Sangat Cepat	4,5
3	Keamanan Dokumen	Apakah Anda merasa dokumen yang dikelola melalui aplikasi aman?	1: Sangat Tidak Aman, 5: Sangat Aman	4,7
4	Integrasi dengan Sistem Lain	Seberapa baik aplikasi terintegrasi dengan sistem lain seperti ERP/CRM?	1: Tidak Terintegrasi, 5: Sangat Terintegrasi	4,6
5	Dukungan Multi-User	Apakah aplikasi mendukung penggunaan oleh banyak pengguna secara bersamaan?	1: Tidak Mendukung, 5: Mendukung dengan Baik	4,5

Tabel 8. Survei reviewer atau PIC terkait sistem manajemen dokumen

No	Kategori Pertanyaan	Pertanyaan	Skala Penilaian (1-5)	Hasil Survei (Rata-rata)
1	Kesesuaian Fungsi	Apakah fitur aplikasi sudah sesuai dengan kebutuhan organisasi?	1: Sangat Tidak Sesuai, 5: Sangat Sesuai	4,5
2	Keamanan Data	Apakah aplikasi memberikan keamanan yang memadai terhadap dokumen?	1: Sangat Tidak Aman, 5: Sangat Aman	4,3
3	Kemudahan Penggunaan	Seberapa mudah aplikasi digunakan oleh pengguna?	1: Sangat Sulit Digunakan, 5: Sangat Mudah	4,7
4	Kinerja Aplikasi	Bagaimana penilaian Anda terhadap kinerja aplikasi dalam kondisi beban tinggi?	1: Sangat Buruk, 5: Sangat Baik	4,6
5	Integrasi Sistem	Apakah aplikasi dapat diintegrasikan dengan sistem lain yang digunakan organisasi?	1: Sangat Tidak Terintegrasi, 5: Sangat Terintegrasi	4,4

Tabel ini berisi hasil survey yang ditujukan kepada user dan reviewer setelah menggunakan aplikasi. Tabel ini juga menunjukan dampak yang diberikan setelah aplikasi diimplementasi dan digunakan dalam kebutuhan sehari hari

KESIMPULAN

PT M berhasil meningkatkan efisiensi dan aksesibilitas manajemen dokumen. Sistem baru ini memungkinkan pengelolaan dan pembagian dokumen secara mandiri maupun terhubung dengan sistem lain, serta mampu menangani jumlah dokumen dan pengguna yang banyak dengan kinerja optimal. Dengan pendekatan pengembangan sistem yang agile, PT M mampu beradaptasi terhadap perubahan kebutuhan dan hasil audit secara efektif, sehingga mencapai peningkatan signifikan dalam manajemen dokumen dan proses bisnis keseluruhan.

DAFTAR PUSTAKA

[1] U. Rahayu dan U. Chadhiq, "THE EFFECT SAP APPLICATION SYSTEM QUALITY AND THE QUALITY OF INFORMATION ON INDIVIDUAL PERFORMANCE DIMEDIATE USER SATISFACTION (At PT. PLN (Persero) CENTRAL JAVA AND D.I.YOGYAKARTA DISTRIBUTION MAIN UNITS)," vol. 5, 2021, doi: 10.29040/IJEBAR.V5I2.2271.

[2] M. Hort, M. Kechagia, F. Sarro, dan M. Harman, "A Survey of Performance Optimization for Mobile Applications," *IEEE Trans. Softw. Eng.*, vol. 48, no. 8, hal. 2879–2904, 2022, doi: 10.1109/TSE.2021.3071193.

[3] S. Widodo dan N. Sulistyowati, "The Effect of Implementation of Report Online Application on User Satisfaction and Operational Performance," vol. 5, hal. 328–338, 2020, doi: 10.38124/ijisrt20aug251.

[4] D. Satria, D. W. Rahma, dan Q. E. Muftikhali, "Analyzing Engineering Approaches in Agile Software Development A Systematic Study," *2024 3rd Int. Conf. Creat. Commun. Innov. Technol.*, hal. 1–6, 2024, doi: 10.1109/iccit62134.2024.10701200.

[5] M. Younas, D. Jawawi, I. Ghani, T. Fries, dan R. Kazmi, "Agile development in the cloud computing environment: A systematic review," *Inf. Softw. Technol.*, vol. 103, hal. 142–158, 2018, doi: 10.1016/J.INFSOF.2018.06.014.

[6] E. C. Daraojimba, C. N. Nwasike, A. O. Adegbite, C. A. Ezeigweneme, dan J. O. Gidiagba, "COMPREHENSIVE REVIEW OF AGILE METHODOLOGIES IN PROJECT MANAGEMENT," *Comput. Sci. & IT Res. J.*, 2024, doi: 10.51594/csitrj.v5i1.717.

[7] R. Hoda, N. Salleh, J. Grundy, dan H. M. Tee, "Systematic literature reviews in agile software development: A tertiary study," *Inf. Softw. Technol.*, vol. 85, hal. 60–70, 2017, doi: 10.1016/J.INFSOF.2017.01.007.

[8] P. Amajuoyi, L. B. Benjamin, dan K. B. Adeusi, "Optimizing agile project management methodologies in high-tech software development," *GSC Adv. Res. Rev.*, 2024, doi: 10.30574/gscarr.2024.19.2.0182.

[9] D. Alexandes, R. Aditio, dan Y. Jumaryadi, "Implementasi Metode Agile dalam Pengembangan Sistem E-document," *J. Inf. Syst. Res.*, vol. 4, no. 1 SE-Articles, Nov 2022, doi: 10.47065/josh.v4i1.2349.

[10] A. Azanha, A. R. T. T. Argoud, J. B. de Camargo Junior, dan P. D. Antoniolli, "Agile project management with Scrum," *Int. J. Manag. Proj. Bus.*, vol. 10, no. 1, hal. 121–142, Jan 2017, doi: 10.1108/IJMPB-06-2016-0054.

[11] R. Wohlrab, P. Pelliccione, E. Knauss, dan M. Larsson, "Boundary objects and their use in agile systems engineering," *J. Softw. Evol. Process*, vol. 31, no. 5, hal. e2166, Mei 2019, doi: https://doi.org/10.1002/smr.2166.

[12] A. A. Shvidkiy, A. A. Savelieva, dan A. A. Zarubin, "Caching Methods Analysis for Improving Distributed Storage Systems Performance," in *2021 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO*, 2021, hal. 1–5. doi: 10.1109/SYNCHROINFO51390.2021.9488328.

[13] S. J. Nadgowda, R. C. Sreenivas, S. Gupta, N. Gupta, dan A. Verma, "C2P: Co-operative Caching in Distributed Storage Systems BT - Service-Oriented Computing," X. Franch, A.

K. Ghose, G. A. Lewis, dan S. Bhiri, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, hal. 214–229.

[14] S. Seshadri *dkk.*, "Software defined just-in-time caching in an enterprise storage system," *IBM J. Res. Dev.*, vol. 58, no. 2/3, hal. 7:1-7:13, 2014, doi: 10.1147/JRD.2014.2303595.

[15] Kumar, V., & Reinartz, W. "Customer Relationship Management: Concept, Strategy, and Tools. Springer", 2018

[16] Schroeder, R. Managing Digital Documents: A Framework for Effective Practice. Springer, 2021

[17] Mehta, N., & Reddy, P. K. Cache Management in Enterprise Systems: An Evaluation of Redis. International Journal of Information Management, 38(1), 110-120, 2018

[18] Lackschewitz, N. M., Krey, S., Nolte, H., Christgau, S., Oeste, S., & Kunkel, J. (2022). Performance Evaluation of Object Storages. NHR2022.