
ANALISIS PERBANDINGAN PEMANFAATAN ERD UNTUK PROSES PEMBUATAN PROGRAM

Muhammad Farid Al Jabbar¹, Fitri Adelina Harahap², Julius Immanuel Sijabat³

Universitas Pamulang, Tangerang Selatan, Banten, Indonesia.

muhfarid039@gmail.com¹, fitriadelina91@gmail.com²,

immanueljulius627@gmail.com³

Abstrak

Entity Relationship Diagram (ERD) merupakan alat yang sangat penting dalam perancangan basis data yang mendukung pengembangan perangkat lunak. ERD memberikan visualisasi yang jelas mengenai struktur data dan hubungan antar entitas, sehingga mempermudah tim pengembang dalam memahami dan mengelola data secara terorganisir. Penelitian ini menganalisis perbandingan penggunaan ERD dalam proses pembuatan program dengan pendekatan tanpa ERD. Hasil penelitian menunjukkan bahwa ERD tidak hanya meningkatkan akurasi dan konsistensi desain basis data, tetapi juga mempercepat proses pengembangan dengan mengurangi risiko kesalahan desain yang dapat berdampak pada efisiensi dan kualitas sistem secara keseluruhan. Sebaliknya, pendekatan tanpa ERD menyebabkan kesulitan dalam pemahaman struktur data, tingginya risiko kesalahan, serta kendala dalam pemeliharaan dan kolaborasi tim. Dengan menggunakan studi kasus pada sistem informasi, penelitian ini menegaskan bahwa ERD merupakan komponen penting yang mendukung keberhasilan proyek pengembangan perangkat lunak.

Kata Kunci: Entity Relationship Diagram (ERD), Perancangan Basis Data, Visualisasi Data, Pengembangan Perangkat Lunak, Efisiensi Sistem.

Abstract

The Entity Relationship Diagram (ERD) is a crucial tool in database design that supports software development. ERD provides a clear visualization of data structures and relationships between entities, making it easier for development teams to understand and manage data in an organized manner. This study analyzes the comparison between the use of ERD in the program creation process and an approach without ERD. The findings indicate that ERD not only improves the accuracy and consistency of database design but also accelerates the development process by reducing the risk of design errors that could impact the overall efficiency and quality of the system. Conversely, the approach without ERD results in difficulties in understanding data structures, higher risks of errors, and challenges in system maintenance and team collaboration. Using a case study on information systems, this research emphasizes that ERD is an essential component in ensuring the success of software development projects.

Keywords: Entity Relationship Diagram (ERD), Database Design, Data Visualization, Software Development, System Efficiency

PENDAHULUAN

Di zaman sekarang, teknologi informasi sudah menjadi bagian yang tidak terpisahkan dari kehidupan sehari-hari. Hampir semua aspek kehidupan, termasuk pendidikan, sangat dipengaruhi oleh kemajuan teknologi ini. Salah satu bidang yang

sangat penting adalah pengembangan perangkat lunak, di mana aplikasi dan teknologi informasi dibuat untuk mempermudah berbagai proses. Dalam konteks ini, *Entity Relationship Diagram* (ERD) muncul sebagai alat yang sangat berguna dalam merancang sistem yang efisien.

ERD (*Entity Relationship Diagram*) atau diagram hubungan entitas adalah sebuah diagram yang digunakan untuk perancangan suatu database dan menunjukkan relasi atau hubungan antar objek atau entitas beserta atribut-atributnya secara detail. Dengan menggunakan ERD, sistem database yang sedang dibentuk dapat digambarkan dengan lebih terstruktur dan terlihat rapi. Selain digunakan dalam perancangan database, ERD sendiri sering digunakan untuk debugging database jika terjadi masalah pada database. Untuk melakukan debug pada database bukanlah hal yang mudah, terlebih lagi jika database yang mengalami masalah memiliki banyak table dan memerlukan penulisan SQL yang kompleks. Dengan menggambarkan skema database menggunakan ERD, kamu menjadi lebih mudah untuk menemukan permasalahan yang terjadi dalam database dan menyelesaikan masalah dengan mudah (Setiawan, Dicoding Space, 2021).

Desain sistem yang buruk sering kali mengarah pada berbagai masalah operasional, salah satunya adalah keterlambatan layanan. Hal ini bisa terjadi apabila sistem yang dirancang tidak memiliki pemahaman yang jelas tentang struktur data yang mendasarinya. Misalnya, dalam pengelolaan data mahasiswa, jika sistem tidak dibangun dengan efisiensi yang baik, proses layanan akademik bisa terhambat. Untuk mengatasi hal ini, penggunaan *Entity Relationship Diagram* (ERD) sangatlah penting. ERD membantu memastikan bahwa sistem yang dibangun memiliki struktur data yang terorganisir dengan baik, mempermudah pengelolaan dan akses data, serta memastikan bahwa sistem tersebut dapat memenuhi kebutuhan pengguna dengan lebih efektif. Dalam konteks ini, ERD tidak hanya menjadi alat desain, tetapi juga sebagai fondasi untuk membangun sistem yang dapat diandalkan dan berfungsi optimal.

Penelitian ini bertujuan untuk melihat bagaimana ERD dapat dimanfaatkan dalam proses pembuatan program, terutama dalam konteks pengembangan teknologi informasi. Kami akan mengumpulkan data melalui observasi dengan pengembang dan pengguna sistem, serta mengamati bagaimana proses pengelolaan data berlangsung. Dengan pendekatan ini, diharapkan dapat diketahui seberapa besar pengaruh ERD terhadap kualitas dan efisiensi sistem yang dibangun.

Lebih jauh lagi, penelitian ini juga akan membahas tantangan yang mungkin dihadapi dalam penerapan ERD. Terkadang, ada kesulitan dalam mengimplementasikan desain yang dihasilkan dari ERD ke dalam program nyata. Dengan memahami tantangan ini, kami berharap dapat memberikan rekomendasi yang bermanfaat bagi pengembang lainnya untuk mengatasi masalah yang ada.

Dengan semua informasi di atas, penelitian ini ingin menunjukkan bahwa pemanfaatan ERD bukan hanya sekadar langkah tambahan, tetapi merupakan bagian yang sangat penting dalam proses pengembangan perangkat lunak. Dengan menerapkan ERD dengan baik, pengembang dapat menciptakan sistem yang lebih baik, lebih cepat, dan lebih sesuai dengan kebutuhan pengguna. Semoga penelitian ini bisa memberikan wawasan baru tentang bagaimana ERD dapat membantu dalam menciptakan sistem informasi yang berkualitas.

METODE PENELITIAN

Dalam penelitian ini, pendekatan Metode Perbandingan diterapkan untuk menganalisis perbandingan penggunaan Entity Relationship Diagram (ERD) dalam proses pembuatan program dengan perancangan sistem yang tidak menggunakan diagram sama sekali. Tujuan utama dari pendekatan ini adalah untuk mengevaluasi secara mendalam bagaimana pemanfaatan ERD dapat mempengaruhi efisiensi dan efektivitas dalam perancangan program, serta membandingkannya dengan pendekatan lain yang tidak menggunakan diagram sama sekali. Pendekatan ini bertujuan untuk menggali kelebihan dan kekurangan dari kedua metode tersebut, serta mengidentifikasi dampaknya terhadap pemahaman struktur data dan implementasi sistem yang lebih terorganisir dan mudah dikelola.

Penelitian ini dilakukan dengan cara membandingkan pengalaman pribadi peneliti dalam menerapkan ERD dengan pendekatan lainnya dalam berbagai tahap pengembangan perangkat lunak, mulai dari analisis kebutuhan hingga tahap perancangan sistem. Penggunaan ERD sebagai alat untuk memodelkan hubungan antar entitas dalam sistem diharapkan dapat mempercepat pemahaman mengenai struktur data dan hubungan antar komponen yang ada dalam sistem. Namun, penting untuk mengetahui sejauh mana penggunaan ERD benar-benar memberikan kontribusi terhadap kualitas dan kecepatan proses perancangan sistem.

Dalam proses perbandingan ini, peneliti mencatat berbagai faktor yang mempengaruhi proses pembuatan program, seperti waktu yang dibutuhkan untuk menyelesaikan tahap perancangan, jumlah revisi yang diperlukan, dan tingkat keberhasilan dalam mengidentifikasi kebutuhan sistem dengan menggunakan ERD dibandingkan dengan metode perancangan lainnya. Peneliti juga memperhatikan bagaimana masing-masing pendekatan mempengaruhi tingkat kesalahan yang muncul selama pengembangan dan bagaimana diagram atau model yang digunakan membantu atau menghambat pemahaman pengembang terhadap struktur dan logika aplikasi yang sedang dikembangkan.

Metode perbandingan ini memungkinkan peneliti untuk memberikan analisis yang lebih objektif dan mendalam mengenai manfaat dan tantangan dalam menggunakan ERD dalam proses pembuatan program. Hasil dari perbandingan ini diharapkan dapat memberikan wawasan baru mengenai efektivitas pemanfaatan ERD dalam konteks pengembangan perangkat lunak, serta memberikan panduan bagi praktisi dalam memilih metode yang tepat untuk digunakan dalam proyek pengembangan mereka.

Dengan menggunakan pendekatan ini, diharapkan dapat diperoleh pemahaman yang lebih komprehensif mengenai bagaimana ERD berfungsi dalam konteks praktis, serta bagaimana penerapannya dapat berkontribusi terhadap keberhasilan proyek perangkat lunak secara keseluruhan. Penelitian ini juga diharapkan dapat memberikan rekomendasi yang bermanfaat bagi pengembang perangkat lunak yang mempertimbangkan penggunaan ERD dalam perancangan sistem mereka.

HASIL PENELITIAN

Pembuatan ERD Menggunakan draw.io



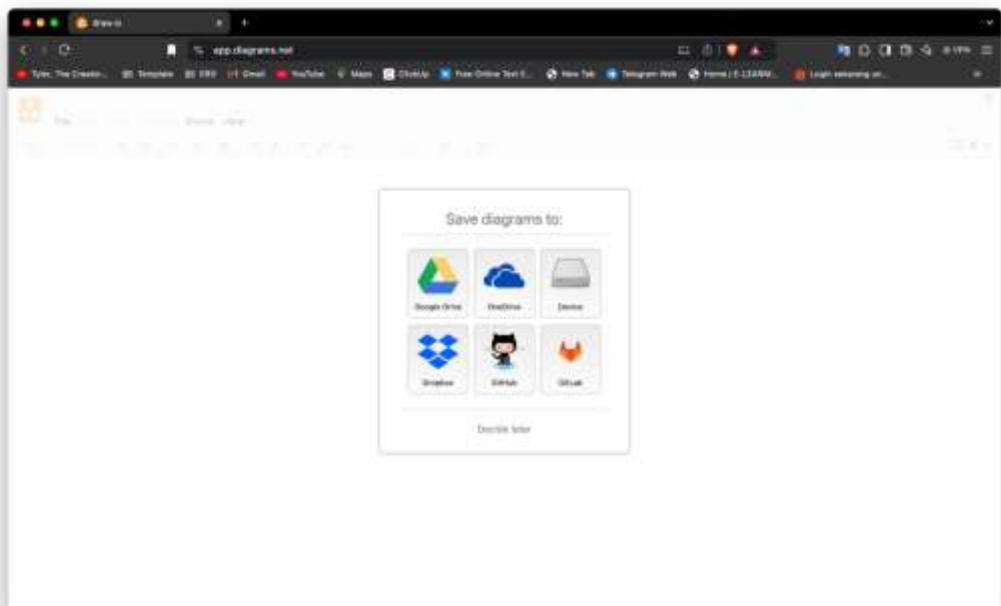
Untuk membuat ERD menggunakan platform draw.io dapat melakukan step by step seperti Langkah berikut

- a. Buka situs draw.io atau <https://app.diagrams.net/> di browser



Gambar 1. Tampilan awal saat membuka situs <https://app.diagrams.net/>

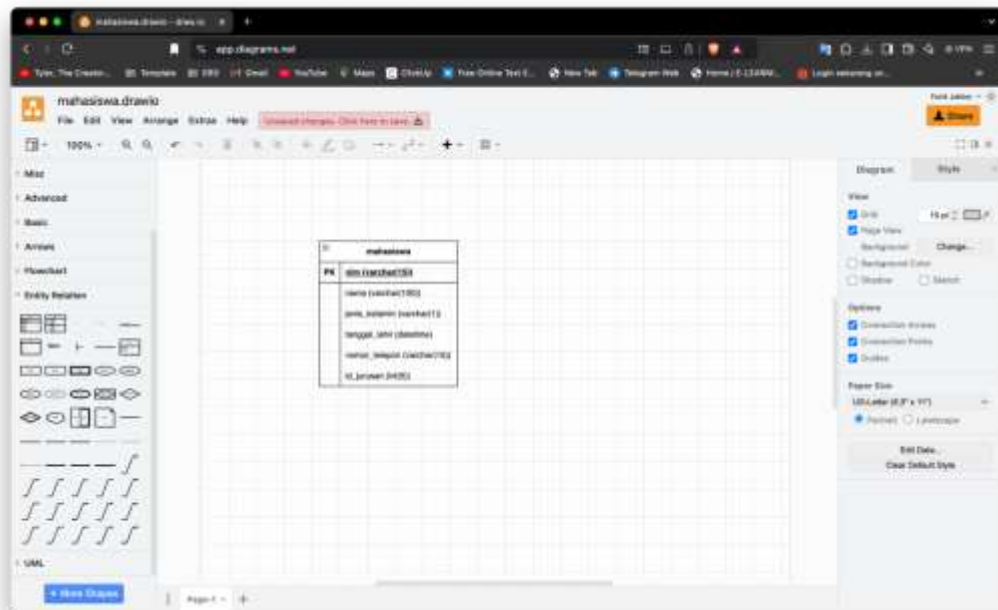
- b. Pilih penyimpanan lokasi ERD, bisa di simpan melalui Google Drive, OneDrive, perangkat sendiri, Dropbox, Github, dan Gitlab.



Gambar 2. Tampilan saat pemilihan lokasi penyimpanan

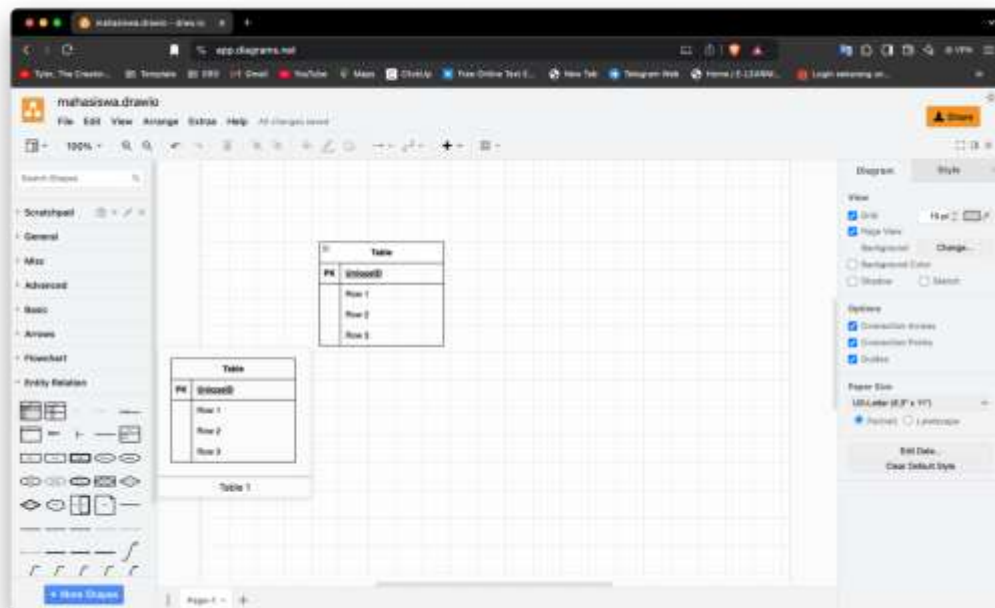
- c. Setelah sudah memilih lokasi penyimpanan ERD, selanjutnya pilih Create New Diagram, dan selanjutnya Blank Diagram. Jangan lupa memberikan nama

filename ERD tersebut, penamaan file misalnya mahasiswa.drawio, jika sudah klik tombol Create untuk membuat workspace baru.



Gambar 3. Penamaan file & pilih template diagram

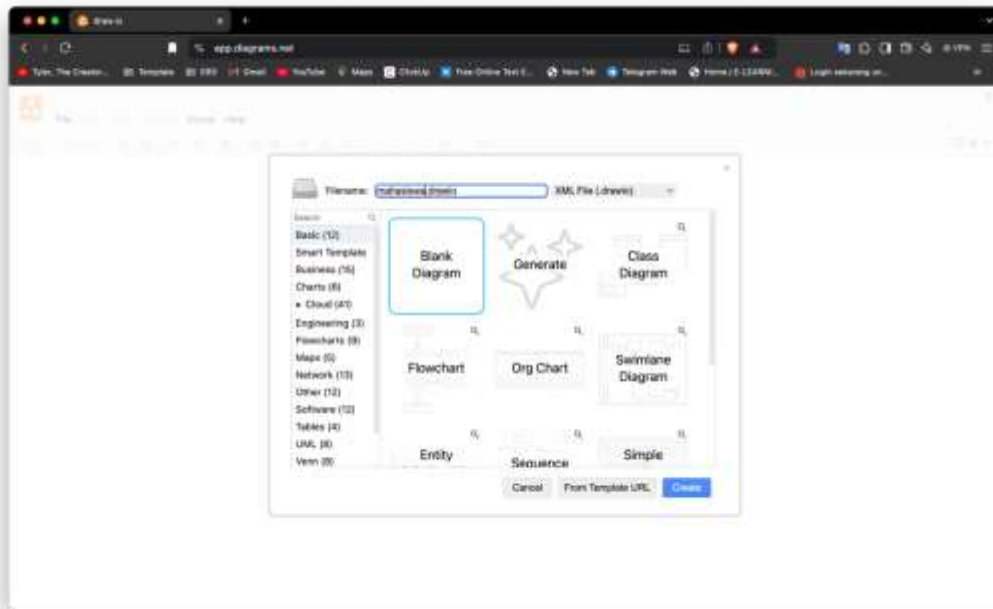
- d. Setelah workspace terbuka, di sisi kiri layar terdapat berbagai simbol atau elemen yang bisa digunakan. Pilih kategori Entity Relation, kemudian seret elemen Table ke area kerja. Elemen ini dirancang khusus untuk membuat ERD yang merepresentasikan table dalam basis data.



Gambar 4. Lokasi elemen table

- e. Setelah itu ganti nama entitas / table dengan basis data yang ada, contoh di ganti dengan nama mahasiswa, dengan klik dua kali text table dan di ganti mahasiswa,

jika sudah tekan esc pada keyboard. Catatan, Primary Key (PK) adalah pengidentifikasi unik dalam table. Ini berbeda dari Foreign Key (FK), yang mewakili primary key dari table lain. Inilah sebabnya mengapa ini disebut foreign key (Macaulay, 2023). Selanjutnya adalah membuat atribut / kolom sesuai konteks dengan table, contoh di table mahasiswa mempunyai kolom nim, nama, jenis kelamin, tanggal lahir, dan lain lain. Opsional setiap kolom bisa di buat tipe data berdasarkan kolom, contoh nim menggunakan tipe data string / varchar dengan ukuran 15. Jika ingin menambahkan atribut / kolom, klik kolom yang



sudah ada, jika sudah tekan shortcut di keyboard Ctrl + D, maka otomatis akan terbuat kolom baru dan membuat nama kolom sesuai kebutuhan.

Gambar 5. Pembuatan entitas / table di draw.io

- f. Untuk menghemat waktu dan tenaga dalam pembuatan table di draw.io, pembuatan table bisa dilakukan dengan cara menduplikasi table yang sudah ada. Caranya cukup mudah, pertama pilih table yang ingin diduplikasi, lalu tekan shortcut Ctrl + D di keyboard. Setelah itu, table baru akan muncul, dan cukup tarik table tersebut ke area kosong di workspace. Dengan menggunakan cara ini, proses pembuatan table bisa lebih cepat, terutama jika table baru hanya membutuhkan sedikit perubahan, seperti penggantian nama entitas / table & penambahan atribut / kolom, seperti contoh pembuatan table jurusan dengan kolom nama. Untuk menghubungkan satu table ke table lainnya, hal pertama yang perlu dilakukan adalah menentukan relasi antar entitas / table. Relasi ini umumnya digambarkan dengan foreign key di table yang satu, yang merujuk pada primary key di table lainnya. Sebagai contoh, jika memiliki dua table seperti mahasiswa dan jurusan, relasi antara keduanya dapat digambarkan dengan menambahkan kolom id_jurusan di table mahasiswa. Kolom ini kemudian mengarah ke id_jurusan di table jurusan, yang merupakan primary key dari table tersebut. Proses ini mempermudah untuk menghubungkan data dari table yang berbeda dan memastikan bahwa data tetap konsisten dan terstruktur dengan baik. Di dalam ERD, relasi ini digambarkan dengan menarik garis antara dua entitas dan

menambahkan kardinalitas yang sesuai, seperti one-to-many (satu ke banyak) atau many-to-one (banyak ke satu), yang menggambarkan seberapa banyak entitas / table yang saling berhubungan. Penting untuk memperhatikan jenis relasi yang tepat, karena hal ini akan mempengaruhi bagaimana data dikaitkan di antara table dan bagaimana sistem basis data akan mengelola integritas data, seperti mencegah duplikasi atau memastikan konsistensi data antar table.

- g. Setelah selesai membuat diagram ERD di draw.io, langkah selanjutnya adalah menyimpan file tersebut untuk memastikan ERD yang telah dibuat tidak hilang dan dapat diakses kapan saja. Untuk menyimpan ERD, klik menu File yang ada di bagian kiri atas, kemudian pilih opsi Save As. Anda dapat memilih lokasi penyimpanan sesuai dengan keinginan, baik itu di penyimpanan sendiri di komputer atau di cloud seperti Google Drive atau OneDrive. Pastikan file disimpan dengan format yang sesuai, seperti .drawio atau .xml, agar bisa dibuka kembali dengan mudah di masa mendatang. Selain itu, penting untuk memberi nama file yang jelas dan deskriptif agar memudahkan pencarian di kemudian hari. Setelah itu, ERD yang telah dibuat dapat diakses, diperbarui, dan dibagikan sesuai kebutuhan.

Keuntungan menggunakan ERD dalam proses pembuatan program

Entity Relationship Diagram (ERD) merupakan alat yang sangat penting dalam proses pembuatan program, terutama dalam tahap perancangan basis data. Salah satu keuntungan utama menggunakan ERD adalah kemampuan untuk memberikan visualisasi yang jelas mengenai struktur data serta hubungan antar entitas yang terlibat dalam sistem. Dengan adanya visualisasi ini, seluruh anggota tim pengembang dapat dengan mudah memahami bagaimana data dalam sistem akan diorganisir dan berinteraksi. Hal ini tidak hanya membantu dalam menyelaraskan pemahaman di antara anggota tim, tetapi juga meningkatkan komunikasi dalam diskusi teknis, terutama saat membahas perancangan atau pengambilan keputusan terkait data. Sebagai contoh, ketika terdapat perubahan dalam desain sistem, ERD dapat digunakan sebagai panduan untuk melihat dampak perubahan tersebut terhadap hubungan antar entitas, sehingga meminimalkan risiko miskomunikasi di dalam tim.

Selain itu, ERD juga berkontribusi pada efisiensi pengembangan program. Dengan menggambarkan struktur entitas dan hubungannya secara visual, pengembang dapat lebih cepat mengidentifikasi potensi masalah atau kesalahan dalam desain sistem sebelum tahap implementasi dimulai. Sebagai alat perancangan, ERD memungkinkan pengembang untuk melihat celah-celah yang mungkin tidak terlihat jika desain hanya dilakukan secara verbal atau tekstual. Dengan demikian, perbaikan dapat dilakukan lebih awal, sehingga menghemat waktu dan biaya yang mungkin timbul akibat perbaikan di tahap implementasi atau pengujian. Dalam konteks pengembangan perangkat lunak yang sering kali melibatkan banyak pihak dan iterasi yang kompleks, penggunaan ERD sebagai alat bantu perancangan tidak hanya mempermudah proses pengembangan tetapi juga memastikan kualitas desain sistem yang lebih baik secara keseluruhan.

Perbandingan dengan pendekatan tanpa ERD

Untuk membuat perbandingan yang mendalam dan terstruktur tentang pendekatan dengan ERD vs tanpa ERD, beberapa aspek utama dapat dianalisis. Berikut merupakan perbandingan yang perlu dilakukan berdasarkan poin yang diberikan:

a. Pemahaman Struktur Data dan Relasi

Dengan ERD proses pengembangan perangkat lunak, terutama yang melibatkan pengelolaan basis data, penggunaan ERD sangat membantu pengembang dalam memahami struktur data secara menyeluruh. ERD memberikan representasi visual yang jelas tentang bagaimana table- table atau entitas dalam sistem saling berhubungan. Visualisasi ini mempermudah tim untuk memahami alur data dan hubungan antar entitas, baik itu hubungan one-to-one, one-to-many, atau many-to-many. Sebagai hasilnya, anggota tim pengembang yang memiliki tingkat pengalaman atau pemahaman yang berbeda tentang basis data dapat memiliki panduan yang sama dalam memahami struktur data. Dengan begitu, kesalahpahaman atau miskomunikasi terkait desain sistem dapat diminimalisir, sehingga proses pengembangan dapat berjalan lebih lancar.

Dengan tanpa ERD sebaliknya, jika pengembangan dilakukan tanpa menggunakan ERD, pemahaman tentang struktur data sering kali hanya bergantung pada dokumentasi teks atau bahkan ingatan dari pengembang individu. Pendekatan ini menimbulkan risiko besar, terutama pada tim yang bekerja secara kolaboratif, karena setiap anggota mungkin memiliki interpretasi yang berbeda tentang struktur data dan hubungan antar entitas. Akibatnya, kebingungan dapat terjadi ketika mencoba memahami bagaimana data diorganisir atau bagaimana entitas tertentu berinteraksi. Pada proyek yang melibatkan banyak table dengan relasi yang kompleks, ketidakjelasan ini dapat menjadi penyebab utama munculnya kesalahan yang berdampak besar pada kualitas sistem.

b. Potensi Kesalahan Desain

Dengan ERD penggunaan dalam tahap perancangan sistem basis data memungkinkan tim untuk mengidentifikasi potensi kesalahan sejak awal. ERD membantu memastikan bahwa hubungan antar entitas / table telah dirancang dengan benar dan tidak ada kekeliruan seperti redundansi data, atribut yang tidak relevan, atau hubungan yang tidak diperlukan. Selain itu, ERD juga memungkinkan pengembang untuk menguji logika desain sebelum melangkah ke tahap implementasi. Misalnya, apabila ditemukan hubungan many-to-many yang tidak semestinya, ERD memberikan panduan untuk memperbaikinya dengan menambahkan table pivot atau intermediate yang sesuai. Dengan cara ini, sistem dapat dirancang dengan lebih efisien, dan masalah-masalah besar dapat dicegah sebelum implementasi dilakukan.

Dengan tanpa ERD potensi kesalahan desain menjadi jauh lebih tinggi karena tim pengembang harus mengandalkan interpretasi individu yang sering kali tidak terstruktur. Dalam kasus seperti ini, kesalahan mungkin baru ditemukan saat sistem sudah memasuki tahap implementasi atau bahkan setelah sistem berjalan. Contohnya, pengembang mungkin melewati hubungan penting antara table tertentu atau membuat desain yang redundan sehingga data menjadi tidak efisien. Kesalahan-kesalahan ini tidak hanya mempengaruhi kualitas desain, tetapi juga

dapat menyebabkan biaya perbaikan yang jauh lebih tinggi, karena perubahan pada tahap implementasi memerlukan usaha yang lebih besar dibandingkan jika diperbaiki saat tahap perancangan.

c. Efisiensi Pengembangan

Dengan ERD alat yang sangat berguna untuk meningkatkan efisiensi selama proses pengembangan sistem. Dengan adanya ERD yang memetakan struktur data dan hubungan antar entitas / table, pengembang memiliki panduan yang jelas untuk merancang dan mengimplementasikan basis data tanpa perlu membuat asumsi atau menebak-nebak. Ini sangat penting, terutama pada proyek besar dengan banyak table dan hubungan yang kompleks. Pengembang dapat langsung merujuk pada ERD untuk memahami bagaimana data harus dikelola dan bagaimana logika bisnis diterapkan, sehingga waktu yang dibutuhkan untuk desain dan implementasi menjadi lebih singkat. ERD juga membantu tim dalam memastikan bahwa semua aspek sistem telah direncanakan dengan baik sebelum implementasi dimulai.

Dengan tanpa ERD sebaliknya, pendekatan tanpa ERD sering kali memperlambat proses pengembangan karena pengembang harus menghabiskan lebih banyak waktu untuk memahami struktur data secara manual. Tidak adanya panduan visual menyebabkan banyak waktu terbuang untuk memeriksa hubungan antar table, mencocokkan atribut, atau bahkan menyelesaikan konflik desain yang muncul di tengah proses pengembangan. Hal ini tidak hanya memperpanjang durasi proyek, tetapi juga meningkatkan risiko munculnya desain yang tidak konsisten, terutama jika ada perubahan kebutuhan dari pengguna selama proses pengembangan berlangsung.

d. Pemeliharaan dan Skalabilitas Sistem

Dengan ERD salah satu keuntungan besar menggunakan ERD adalah kemudahan dalam memelihara dan mengembangkan sistem di masa depan. Sistem yang dirancang menggunakan ERD memiliki dokumentasi yang jelas tentang struktur data, termasuk hubungan antar entitas / table, atribut / kolom, dan dependensi data. Dokumentasi ini menjadi sangat penting, terutama jika sistem perlu diperbarui, ditingkatkan, atau diperbaiki. ERD memberikan gambaran yang mudah dipahami bagi pengembang baru yang bergabung dalam tim, sehingga mereka dapat dengan cepat memahami desain sistem tanpa harus mempelajari kode secara mendalam. Selain itu, ERD mempermudah proses perencanaan untuk menambah fitur baru, karena pengembang dapat dengan mudah melihat bagaimana penambahan entitas / table baru atau modifikasi pada hubungan tertentu akan memengaruhi keseluruhan sistem.

Dengan tanpa adanya ERD, sistem cenderung lebih sulit untuk dipelihara dan dikembangkan. Dokumentasi yang kurang jelas atau bahkan tidak ada sama sekali membuat pengembang baru harus menghabiskan waktu lebih lama untuk mempelajari desain sistem. Ini dapat memperlambat proses perbaikan atau pengembangan fitur baru, terutama jika struktur data yang kompleks hanya didokumentasikan secara parsial atau tersebar dalam berbagai dokumen. Kurangnya dokumentasi juga dapat menyebabkan perubahan yang tidak

terkontrol, sehingga memperbesar risiko terjadinya kerusakan pada bagian sistem lain saat pemeliharaan dilakukan.

e. Kolaborasi Tim

Dengan ERD dalam proyek yang melibatkan tim pengembang, ERD menjadi alat komunikasi yang sangat efektif untuk memastikan semua anggota tim memiliki pemahaman yang sama tentang struktur data. Karena ERD menggunakan representasi visual, pengembang dari berbagai latar belakang, seperti backend, frontend, atau bahkan analis bisnis, dapat dengan mudah memahami konsep desain yang sama. Hal ini meningkatkan efisiensi kolaborasi tim, karena diskusi terkait perancangan basis data dapat dilakukan dengan lebih terarah. ERD juga membantu dalam menjembatani komunikasi antara tim teknis dan tim non-teknis, karena diagram ini dapat menjelaskan konsep yang kompleks dengan cara yang mudah dimengerti.

Dengan tanpa ERD, kolaborasi antar tim menjadi lebih sulit karena setiap anggota tim mungkin memiliki pemahaman yang berbeda tentang desain sistem. Tidak adanya acuan visual dapat menyebabkan miskomunikasi atau bahkan konflik terkait bagaimana struktur data seharusnya dirancang. Dalam situasi seperti ini, diskusi tim sering kali menjadi tidak produktif, karena setiap anggota harus menghabiskan waktu untuk menjelaskan desain mereka secara verbal atau menggunakan dokumentasi yang mungkin sulit dipahami.

KESIMPULAN

Berdasarkan hasil analisis yang telah dilakukan, penelitian ini menegaskan bahwa Entity Relationship Diagram (ERD) merupakan alat yang sangat penting dalam proses pembuatan program, khususnya dalam perancangan basis data (Sugiarti, 2024). Penggunaan ERD memberikan banyak keuntungan yang signifikan, termasuk visualisasi yang jelas terhadap struktur data dan relasi antar entitas, yang tidak hanya mempermudah pemahaman tim pengembang tetapi juga meningkatkan efisiensi pengembangan secara keseluruhan. Dengan adanya ERD, tim dapat lebih cepat mengidentifikasi potensi kesalahan desain, meminimalisir risiko miskomunikasi, serta memastikan sistem yang dibangun terorganisir dengan baik dan dapat diandalkan.

Sebaliknya, pendekatan tanpa ERD menimbulkan berbagai kelemahan, seperti kurangnya pemahaman yang terstruktur terhadap hubungan data, tingginya risiko kesalahan desain, dan meningkatnya kompleksitas dalam pemeliharaan sistem di masa mendatang. Tanpa panduan visual yang jelas, pengembang sering kali menghadapi tantangan besar dalam menyelaraskan pemahaman tim, memperlambat proses pengembangan, dan meningkatkan biaya perbaikan akibat kesalahan yang terlambat terdeteksi.

Penelitian ini juga menunjukkan bahwa ERD tidak hanya membantu dalam tahap awal perancangan, tetapi juga memberikan manfaat jangka panjang, seperti kemudahan dalam pemeliharaan, skalabilitas, dan kolaborasi tim. Dengan dokumentasi yang baik dan representasi visual yang sederhana namun efektif, ERD membantu memastikan sistem tetap relevan dan mudah dikelola, bahkan ketika ada perubahan atau pengembangan lanjutan di masa depan.

Secara keseluruhan, pemanfaatan ERD bukan hanya langkah tambahan, melainkan kebutuhan esensial dalam pengembangan perangkat lunak yang lebih terstruktur, efisien, dan berkualitas. Oleh karena itu, ERD sebaiknya digunakan sebagai bagian integral dari proses pembuatan program untuk mengurangi risiko kegagalan proyek dan memastikan kesuksesan pengembangan sistem informasi yang optimal.

DAFTAR PUSTAKA

- Afrizal Zein, 2024, Implementation of Service Oriented Architecture in Mobile Applications to Improve System Flexibility, Interoperability, and Scalability, *Journal of Information System, Technology and Engineering* 2 (1), 171-174.
- Afrizal Zein, C Rozali, 2024, Information Technology Investment Assessment Using Val IT Framework, *International Journal of Engineering, Science and Information Technology* 4, 57-62
- Davenport, T. H., & Harris, J. G. (2017). *Keunggulan melalui Analitik: Ilmu Baru untuk Sukses*. Harvard Business Review Press.
- Firmansyah, R., & Persada, G. N. (2024). Perancangan Sistem Pemesanan Produk pada Toko Merchandise. Id Berbasis Website Menggunakan Metode Prototype. *Jurnal Informatika Utama*, 2(1), 1-13.
- Gartner, Inc. (2019). *Laporan Teknologi Big Data dan Inovasi 2019*. Gartner Research.
- Macaulay, M. (2023, November 13). Create an ERD Diagram with me: Using draw.io. Retrieved from medium: <https://medium.com/@macaulaymercy1/how-to-create-an-erd-diagram-using-draw-io-a5267c85d788>
- Prasetyo, D. E., Nurfaizal, H., & Effendi, A. (2024). Analisis Perbandingan Metode Analytical Hierarchy Process (AHP) dan Technique for Order Preference By Similarity To Ideal Solution (TOPSIS) dalam Pemilihan Jurusan Siswa Baru:(Studi Kasus: SMKS Binong Permai An-Nurmaniyah). *Jurnal Informatika Utama*, 2(1), 43-49.
- Pugu, M. R., Riyanto, S., & Haryadi, R. N. (2024). *Metodologi Penelitian; Konsep, Strategi, dan Aplikasi*. PT. Sonpedia Publishing Indonesia.
- Rouse, M. (2018). *Pengertian dan Penerapan Big Data*. TechTarget Database Trends and Applications.
- Setiawan, R. (2021, August 25). Dicoding Space. Retrieved from Memahami ERD, Model Data, dan Komponennya: <https://www.dicoding.com/blog/memahami-erd/>
- Sugiarti, U. (2024, October 17). Apa itu ERD? Arti, Komponen, dan Contohnya. Retrieved from Lawencon: <https://www.lawencon.com/entity-relationship-diagram/#:~:text=dan%20tanggal%20kedaluwarsa.-,Kesimpulan,tim%20teknis%20maupun%20non%2Dteknis.>
- Turban, E., Sharda, R., & Delen, D. (2014). *Sistem Pendukung Keputusan dan Bisnis Intelijen*. Edisi ke-10. Pearson Education.
- Zein, A.;& Suharyanto, E. (2022). Analisis Data Minat Calon Mahasiswa Universitas Pamulang Dengan Menggunakan Algoritma Naive Bayes Classifier. *Sainstech* Vol. 32 No. 3, 70-76.