

Penerapan Metode Agile dalam Pengembangan Website Online Course Berbasis Node.js dan Sequelize

Selly Berlian Rima Maulia¹, Ramadhan Renaldy²

¹Program Studi Informatika, Universitas Persatuan Guru Republik Indonesia Semarang, Kota Semarang

²Program Studi Informatika, Universitas Persatuan Guru Republik Indonesia Semarang, Kota Semarang

*Email: sellyberliannrm@gmail.com

Abstract.

An online course website serves as a digital solution to support flexible and easily accessible distance learning. In the development of this project, the Agile methodology was applied to ensure a gradual and adaptive system development process that meets user needs. The main focus of the development lies in three key features: a landing page that provides an engaging initial impression for visitors, a student dashboard that displays learning progress and structured course materials, and a checkout feature that facilitates an efficient and secure course purchasing process. The technologies used include Node.js as a lightweight and fast backend server, and Sequelize as an ORM (Object-Relational Mapping) for structured and maintainable database management. The results of the development show that the implementation of Agile methodology is highly effective in boosting team productivity and delivering features that align with user needs. This project also offers practical experience in applying modern software development concepts, both technically and collaboratively. With the successful implementation of the core features, this website is expected to serve as a valuable learning platform for users and holds potential for further development.

Keywords: Agile Methods; Online Course; Node.js; Sequelize

Abstrak

Website *online course* merupakan salah satu solusi digital dalam mendukung proses pembelajaran jarak jauh yang fleksibel dan mudah diakses. Dalam pengembangan proyek ini, metode Agile diterapkan guna memastikan proses pembangunan sistem berjalan secara bertahap dan adaptif terhadap kebutuhan pengguna. Fokus utama pengembangan terletak pada tiga fitur utama, yaitu *landing page* sebagai media informasi awal yang menarik bagi pengunjung, dashboard siswa untuk menampilkan progres dan materi pembelajaran secara terstruktur, serta fitur *checkout* yang memfasilitasi proses pembelian *course* secara efisien dan aman. Teknologi yang digunakan meliputi Node.js sebagai *backend* server yang ringan dan cepat, serta Sequelize sebagai ORM (Object-Relational Mapping) untuk pengelolaan *database* yang terstruktur dan mudah dipelihara. Hasil dari pengembangan menunjukkan bahwa penerapan metode Agile sangat efektif dalam mendorong produktivitas tim dan menghasilkan fitur yang sesuai kebutuhan. Proyek ini juga memberikan pengalaman praktis dalam menerapkan konsep-konsep pengembangan perangkat lunak modern, baik dari segi teknis maupun kolaboratif. Dengan tercapainya fitur-fitur inti yang dirancang, website ini diharapkan mampu menjadi platform pembelajaran yang bermanfaat bagi pengguna serta memiliki potensi untuk dikembangkan lebih lanjut.

Kata Kunci : Metode Agile; *Online course*; Node.js; Sequelize

1. Pendahuluan

Teknologi yang berkembang pesat telah membawa perubahan signifikan dalam dunia pendidikan, salah satunya dengan hadirnya pembelajaran daring atau *online course*. Sebuah model pembelajaran yang memberikan fleksibilitas, aksesibilitas, dan kemudahan belajar tanpa batasan ruang dan waktu, sehingga menjadi solusi atas keterbatasan pendidikan konvensional seperti biaya, geografis, dan waktu [1]. Seiring meningkatnya kebutuhan

masyarakat akan pembelajaran daring, pengembangan platform *online course* yang handal dan mudah digunakan menjadi semakin tinggi [2]. Industri *software house* dituntut untuk menghasilkan produk digital yang inovatif dan sesuai kebutuhan pasar. Dalam hal ini tentunya pengembangan aplikasi *online course* menjadi isu yang sangat relevan, karena semakin banyak institusi pendidikan dan individu yang mulai beralih ke teknologi digital. Penggunaan *Application Programming Interface* (API) menjadi elemen kunci dalam membangun sistem yang efisien dan terintegrasi [3]. API membuat pertukaran data antara *frontend* dan *backend*, seperti informasi kelas, progres belajar, dan transaksi secara *real time* [4].

Pemilihan teknologi yang tepat tentunya akan sangat menentukan keberhasilan aplikasi yang dikembangkan. Oleh karena itu, Node.js dipilih sebagai *backend* karena performanya yang efisien dan skalabilitas tinggi, sehingga cocok untuk aplikasi dengan permintaan tinggi [5]. Sedangkan untuk pengelolaan basis data, digunakan *Object-Relational Mapping* (ORM) Sequelize yang menyederhanakan interaksi dengan database, menjaga konsistensi data, dan mempercepat pengembangan tanpa perlu menulis query *Structured Query Language* (SQL) secara manual [6]. Kedua kombinasi teknologi ini mendukung pengembangan aplikasi yang terstruktur dan mudah dikembangkan. Pengembangan proyek ini berfokus pada tiga fitur utam, yaitu *landing page*, *dashboard siswa*, dan *checkout course*. Halaman *landing page* berfungsi sebagai pintu masuk yang menampilkan informasi menarik tentang platform dan daftar kelas yang tersedia. Halaman *dashboard siswa* menyajikan pusat kontrol bagi pengguna untuk melihat progres belajar dan status kelas secara *real time*. Sedangkan fitur *checkout* mengelola transaksi pembelian *course* secara aman dan efisien, memastikan proses pembayaran yang nyaman serta pencatatan data transaksi yang akurat.

2. Metode

2.1 Metode Pengembangan Sistem

Website *online course* dikembangkan dengan menggunakan metode agile. Metode ini merupakan salah satu pendekatan pengembangan perangkat lunak yang menekankan pengerjaan secara cepat dan fleksibel guna menghadapi perubahan kebutuhan yang terjadi dalam waktu singkat [7]. Metode agile lebih mengutamakan kolaborasi tim yang intensif dan pengembangan secara iteratif atau berulang, yang memungkinkan tim untuk merespons perubahan dengan cepat dan mengurangi waktu pengerjaan proyek sekaligus meningkatkan kepuasan klien [8]. Dalam implementasinya, metode agile menggunakan kerangka kerja seperti scrum yang mengorganisasi pekerjaan dalam siklus pendek yang disebut *sprint*, yang memungkinkan evaluasi dan penyesuaian berkelanjutan berdasarkan umpan balik pengguna [9]. Metode ini menjadi pilihan utama dalam pengembangan perangkat lunak modern karena kemampuannya dalam meningkatkan responsivitas, kolaborasi, dan kualitas hasil pengembangan [10].



Gambar 1. Implementasi metode agile

Implementasi metode Agile pada pengembangan sistem *online course* ditunjukkan pada gambar 1, dengan tahapannya dimulai dari *plan*, *design*, *develop*, dan *testing*. Pada tahap *plan* dirancang fitur utama yaitu *landingpage*, *dashboard siswa*, dan *checkout course*. Kemudian dipecah lagi menjadi *task-task* kecil agar lebih mudah dikelola. Tahap *design* berfokus pada perancangan antarmuka dan struktur sistem, termasuk pembuatan diagram UML serta desain database menggunakan Sequelize. Tahap *develop* dibagi dalam beberapa *sprint*. *Sprint* pertama mengerjakan *landingpage* dengan *backend* Node.js. *Sprint* kedua mengembangkan *dashboard siswa* yang menampilkan data kursus dan progres belajar. *Sprint* ketiga fokus pada *checkout course*, mencakup alur pemesanan, validasi input, dan penyimpanan transaksi. Sequelize digunakan untuk mengatur database serta relasi antar tabel seperti *user*, *course*, dan *invoice*. Setelah pengembangan selesai, dilanjutkan dengan

pengujian fungsional tiap fitur. *Landingpage* diuji untuk memastikan konten tampil tanpa error. *Dashboard* diuji agar data kursus dan progres muncul sesuai akun pengguna. Fitur *checkout* diuji pada proses pembelian, validasi data, dan penyimpanan transaksi ke database.

2.2 Analisis Kebutuhan

Sebelum melakukan perancangan sistem, terlebih dahulu dilakukan analisis kebutuhan untuk memastikan bahwa sistem yang dirancang dapat memenuhi kebutuhan pengguna. Adapun identifikasi dari kebutuhan tersebut yaitu terdiri dari kebutuhan fungsional dan kebutuhan non-fungsional. Tabel 1 menjelaskan kebutuhan fungsional dan kebutuhan non-fungsional yang dibutuhkan dalam pengembangan sistem.

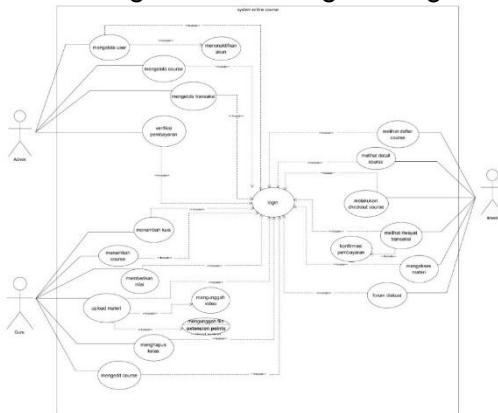
Tabel 1. Analisis kebutuhan sistem

Jenis Kebutuhan	Deskripsi
Fungsional	Menyediakan tampilan awal dengan informasi platform, daftar kursus, dan akses login/register.
	Menyediakan akses kursus yang dibeli, materi pembelajaran, dan progres belajar siswa.
	Memfasilitasi pemilihan kursus, proses pembayaran, dan pencatatan transaksi ke dalam sistem.
Non Fungsional	Menjamin perlindungan data pengguna dan transaksi dari akses tidak sah.
	Sistem merespons cepat dan tetap stabil meskipun digunakan banyak pengguna secara bersamaan.
	Menjamin data penting seperti transaksi dan progres belajar tersimpan dan ditampilkan dengan benar.
	Node.js, Express.js, Sequelize, PostgreSQL, VS Code, GitLab, dan browser untuk pengembangan dan pengujian.

2.3 Perancangan Sistem

a. Use case diagram

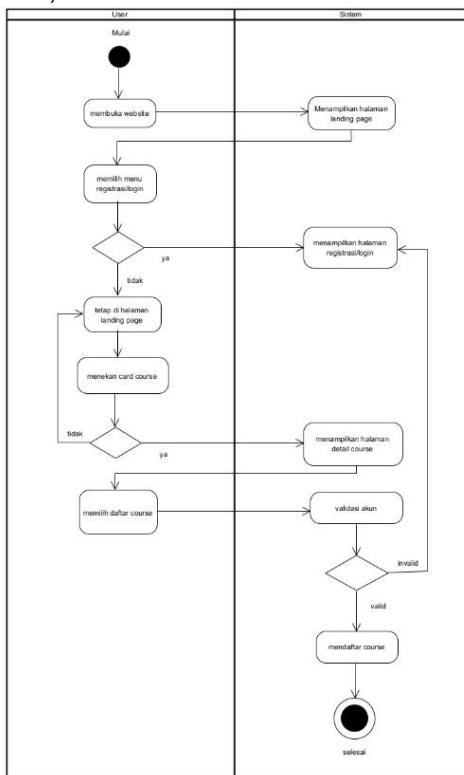
Use case diagram digunakan untuk menggambarkan interaksi antara sistem dan aktor secara visual [11]. Use case diagram berfungsi sebagai komunikasi antara pengembang dan klien untuk memberikan kejelasan tentang fungsi dan kebutuhan sistem yang sedang dikembangkan. Di dalamnya terdapat beberapa simbol yang digunakan untuk menggambarkan interaksi antara aktor dengan sistem. Aktivitas yang dapat dilakukan user digambarkan pada use case diagram pada gambar 2 di mana admin dapat mengelola course, user, transaksi, dan juga melakukan verifikasi pembayaran. Aktor guru dapat menambah kuis, menambah kelas, memberikan nilai, mengunggah materi, menghapus kelas, dan mengedit kelas. Sedangkan aktor siswa dapat melihat daftar kelas, melihat detail kelas, melakukan checkout kelas, melihat riwayat transaksi, mengakses materi, dan menggunakan forum diskusi untuk menanyakan pertanyaan. Namun untuk dapat melakukan aktivitas pada sistem, user perlu login terlebih dahulu sesuai dengan role masing-masing.



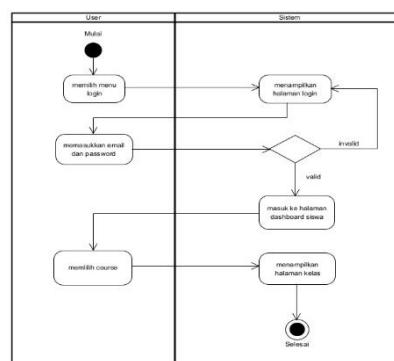
Gambar 2. Use Case Diagram

b. Activity diagram

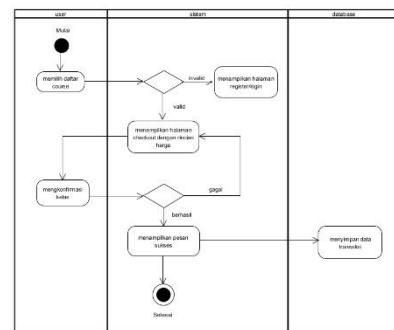
Activity diagram merupakan jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan aliran kerja atau aktivitas suatu sistem [12]. Activity diagram akan membantu memvisualisasikan urutan aktivitas dari satu aktivitas ke aktivitas lain dalam sistem, termasuk aliran paralel dan percabangan. Dengan adanya visualisasi alur aktivitas, diagram ini dapat menjadi alat komunikasi yang efektif untuk menjelaskan bagaimana sebuah sistem bekerja kepada tim pengembang dan pengguna [13]. Activity diagram pada sistem online course ini terlihat pada gambar 3 sampai dengan 5 yang meliputi halaman *landingpage*, *dashboard*, dan *checkout course*



Gambar 2. Landingpage



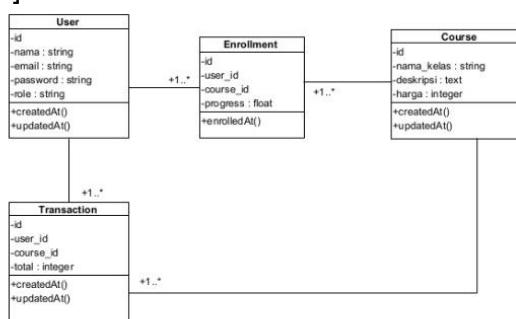
Gambar 1. Dashboard siswa



Gambar 3. Checkout

c. Class diagram

Class Diagram merupakan salah satu jenis diagram dalam UML yang menggambarkan struktur suatu sistem dengan menampilkan kelas-kelas yang ada [14]. Didalam *class diagram* terdapat kelas, atribut, metode, dan hubungan antar kelas secara statis. Diagram ini digolongkan sebagai diagram struktur karena menunjukkan komponen-komponen yang harus ada dalam sistem yang dimodelkan. Fungsi utama *class diagram* adalah untuk memvisualisasikan struktur sistem secara jelas sehingga memudahkan pemahaman dan pengembangan aplikasi [15].

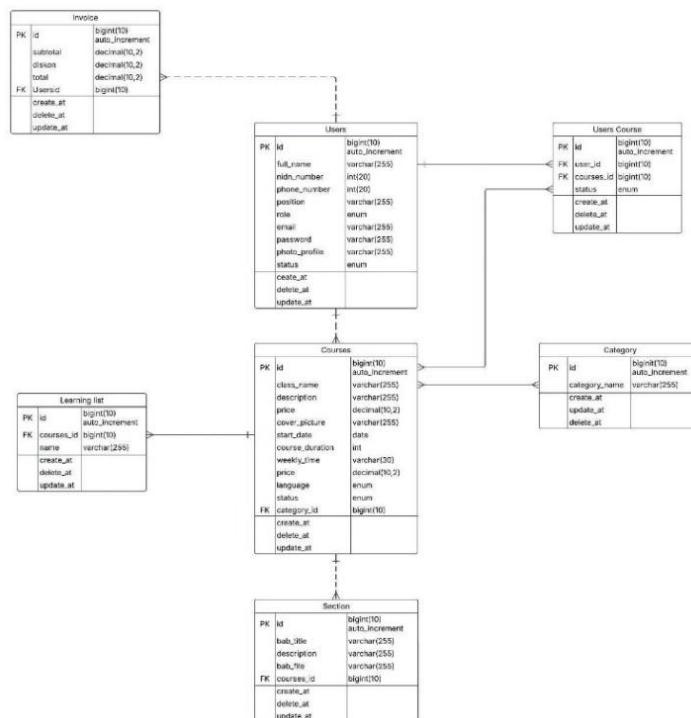


Gambar 4. Class diagram

Gambar 6 menggambarkan struktur sistem pada *website online course* yang melibatkan empat class utama yaitu *user*, *course*, *enrollment*, dan *transaction*. Class *user* menyimpan informasi pengguna seperti nama, email, password, dan peran (admin, siswa, atau guru). Setiap pengguna dapat melakukan transaksi pembelian *course* yang dicatat dalam class *transaction*, termasuk informasi total pembayaran dan relasinya terhadap *user* dan *course* yang dibeli. Setelah pembelian, siswa akan tercatat dalam class *enrollment* sebagai peserta kursus dengan progres belajar yang terpantau. Sementara class *course* menyimpan data terkait kelas yang tersedia seperti nama, deskripsi, dan harga. Relasi antar entitas menunjukkan bahwa satu *user* dapat memiliki banyak transaksi dan *enrollments*, serta satu *course* dapat diikuti oleh banyak *user*.

d. ERD (*Entity Relationship Diagram*)

ERD merupakan teknik pemodelan data yang digunakan untuk menggambarkan kebutuhan data suatu organisasi atau sistem secara konseptual [16]. ERD ini membantu sistem analis dan pengembang memahami struktur data dan hubungan antar data sebelum implementasi basis data fisik.



Gambar 5. Entity Relationship Diagram *online course*

Gambar 7 merupakan ERD yang menggambarkan struktur basis data untuk sistem *website online course*. Terdapat beberapa entitas utama, yaitu *users*, *courses*, *users course*, *invoice*, *category*, *learning list*, dan *section*. Setiap entitasnya memiliki atribut yang merepresentasikan data penting, yang mana relasi antar entitas menggambarkan keterkaitan, seperti pengguna yang bisa membeli *course*, mengikuti *course*, dan mendapatkan *invoice* transaksi.

3. Hasil dan Pembahasan

a. Halaman *Landing Page*

Halaman *landing page* menjadi halaman yang muncul pertama kali saat pengguna mengakses *website*. Halaman ini akan menampilkan informasi dari platform, keunggulan, serta daftar kelas apa saja yang tersedia. Selain itu juga di pojok kanan atas terdapat tombol “Daftar” dan “Masuk” yang akan mengarahkan pengguna untuk melakukan autentikasi agar dapat mengakses fitur-fitur lanjutan yang ada dalam *website* seperti yang ditunjukkan gambar 8.



Gambar 8. Halaman landingpage

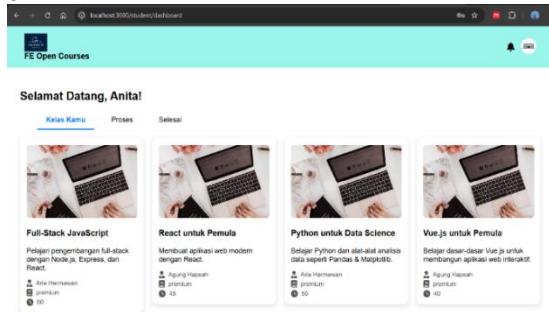


Gambar 9. Halaman detail kelas

Pada *section* semua kelas, tiap-tiap kelasnya ditampilkan dalam bentuk kartu yang mencantumkan informasi, seperti judul kursus, deskripsi singkat, total jam, dan harga kursus. Harga yang tertera ini menandakan bahwa platform menyediakan kursus berbayar, dengan tombol "Daftar" pengguna akan diarahkan ke proses pendaftaran atau pembayaran. Selanjutnya ketika pengguna menekan *card* pada kelas, sistem akan langsung mengarahkan pengguna ke halaman detail dari kelas tersebut, seperti yang terlihat pada gambar 9.

b. Halaman Dashboard Siswa

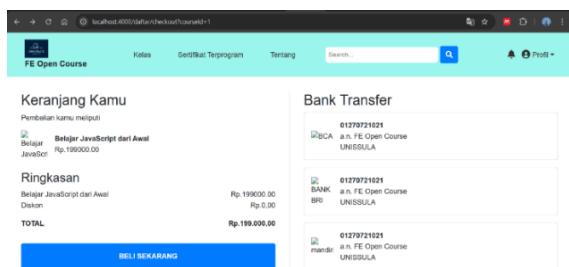
Halaman dashboard siswa berfungsi sebagai pusat kontrol bagi pengguna dalam hal ini adalah siswa, untuk mengakses materi pembelajaran dan memantau kemajuan belajar. Gambar 10 merupakan tampilan dashboard siswa ketika telah berhasil *login*. Siswa akan disambut dengan sapaan "Selamat datang, (sesuai nama pengguna)!", dan siswa juga dapat melihat daftar kelas apa saja yang diikuti. Pada bagian atas terdapat tiga tab navigasi seperti "Kelas Kamu", "Proses", dan "Selesai", yang akan membantu siswa mengelompokkan kursus berdasarkan status pembelajaran.



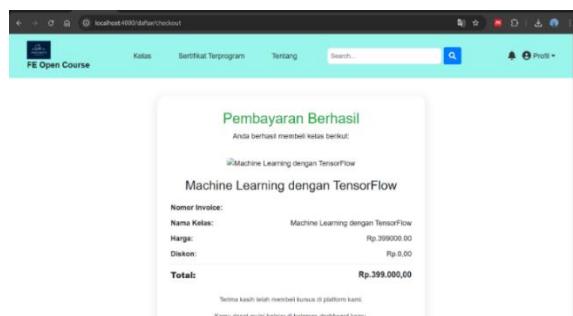
Gambar 10. Halaman dashboard siswa

c. Halaman Checkout Online Course

Ketika pengguna memilih menu "Daftar" pada *card course* di halaman *landing page*, selanjutnya pengguna akan diarahkan menuju halaman checkout yang berisi detail informasi terkait kelas yang akan diambil. Halaman *checkout course* ini dirancang untuk memberikan pengalaman transaksi yang jelas dan nyaman bagi pengguna. Setelah pengguna melakukan konfirmasi, maka akan muncul halaman berhasil melakukan pembelian seperti yang telihat pada gambar 11. Pembelian dapat dilakukan dengan menekan tombol "Beli sekarang". Selanjutnya sistem akan menyimpan data transaksi ke dalam database dan akan muncul halaman berhasil melakukan pembayaran seperti pada gambar 12. Halaman akan menampilkan detail informasi pembayaran seperti nama kelas, nomor *invoice*, harga, diskon (jika ada), dan total pembayaran.



Gambar 11. Halaman checkout course



Gambar 12. Berhasil melakukan pembelian course

d. Tahapan Testing pada Metode Agile

Tahapan testing dilakukan pengujian terhadap masing-masing fiturnya untuk memastikan seluruh fungsi telah berjalan sesuai dengan desain sistem yang telah dirancang sebelumnya. Fitur landing page diuji untuk memastikan tampilan informasi, daftar kelas, dan tombol login serta register dapat berfungsi dengan baik. Pada dashboard siswa dilakukan verifikasi terhadap tampilan data kursus yang telah dibeli, akses materi, dan progres belajar pengguna. Sedangkan pada fitur *checkout course*, pengujian meliputi alur pembelian kursus, validasi data input, dan pencatatan transaksi ke dalam database. Seluruh pengujian ini bertujuan untuk memastikan bahwa sistem berjalan stabil, akurat, dan sesuai dengan kebutuhan pengguna. Hasil dari implementasi metode agile menunjukkan bahwa pengembangan sistem dapat dilakukan secara terstruktur, fleksibel, dan adaptif terhadap perubahan, sehingga mampu menghasilkan fitur yang berfungsi optimal dan sesuai dengan ekspektasi pengguna.

4. Kesimpulan

Pengembangan website online course dengan menggunakan metode agile telah memberikan hasil yang optimal dalam menciptakan sistem yang responsif dan sesuai kebutuhan pengguna. Metode ini memungkinkan proses pengembangan dilakukan secara bertahap melalui iterasi (*sprint*), sehingga tim dapat melakukan evaluasi dan perbaikan secara berkelanjutan. Fitur *landing page* dikembangkan untuk memberikan informasi kursus secara menarik dan mudah diakses, dashboard siswa dirancang agar pengguna dapat dengan mudah memantau progres belajar mereka, serta fitur *checkout* disusun untuk mempermudah transaksi secara cepat dan aman. Penggunaan teknologi Node.js memberikan performa *backend* yang ringan dan *scalable*, sementara Sequelize membantu mengelola *database* secara efisien dengan pendekatan ORM yang mudah diintegrasikan. Secara keseluruhan, proyek ini tidak hanya menghasilkan sistem yang fungsional dan efektif, tetapi juga memberikan pembelajaran mendalam tentang praktik kerja tim, komunikasi, dan pengembangan perangkat lunak modern. Proses ini memperkuat kemampuan teknis dan soft skill pengembang, terutama dalam menyusun struktur aplikasi, merancang database, dan berkolaborasi menggunakan pendekatan agile. Dengan hasil yang dicapai, proyek ini dapat menjadi landasan awal untuk pengembangan lebih lanjut pada sistem pembelajaran daring yang lebih kompleks dan lengkap di masa mendatang.

5. Referensi

- [1] Y. Firmansyah, L. Rahmatiani, and Y. Nugraha, "Peran Buana Online Course Dalam Pembelajaran Daring," *Buana Ilmu*, vol. 6, no. 2, pp. 122–129, 2022, doi: 10.36805/bi.v6i2.2345.
- [2] S. Mutofin, S. Setiawani, and S. Hussen, "Pengembangan Media Pembelajaran Interaktif Online Menggunakan Easyclass Berbantuan Geogebra Materi Program Linier," *KadikmA*, vol. 11, no. 1, p. 27, 2020, doi: 10.19184/kdma.v11i1.17674.
- [3] A. R. Sinha, "Optimizing API Project Efficiency : Agile Configurations , Advanced Design Patterns , and Testing Strategies," pp. 1–8, 2022, doi: 10.55041/IJSREM16054.
- [4] M. Mudassir and M. Mushtaq, "The role of cloud computing in modern software development," *World J. Adv. Eng. Technol. Sci.*, vol. 13, no. 01, pp. 1045–1047, 2024.

- [5] A. P. Kejor and W. N. Purwanti, "Efektivitas Penggunaan Node Js Dalam Pembuatan Rest Api Untuk Aplikasi Katastrofa," *Pros. Semin. Nas. Sains dan Teknol. Seri III*, vol. 2, no. 1, pp. 995–1008, 2025.
- [6] I. P. A. E. Pratama and I. M. S. Raharja, "Node.js Performance Benchmarking and Analysis at Virtualbox, Docker, and Podman Environment Using Node-Bench Method," *Int. J. Informatics Vis.*, vol. 7, no. 4, pp. 2240–2246, 2023, doi: 10.30630/jiov.7.4.1762.
- [7] M. Naufal Faruq and Maryam, "Implementasi Metode Agile Pada Pengembangan Aplikasi Manajemen Pengelolaan Layanan Wifi," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 6, pp. 3472–3478, 2023, doi: 10.36040/jati.v7i6.7868.
- [8] N. Hikmah, A. Suradika, and R. A. Ahmad Gunadi, "Metode Agile Untuk Meningkatkan Kreativitas Guru Melalui Berbagi Pengetahuan (Knowledge Sharing)" *J. Instr.*, vol. 3, no. 1, p. 30, 2021, doi: 10.24853/instruksional.3.1.30-39.
- [9] A. Ariesta, Y. N. Dewi, F. A. Sariashih, and F. W. Fibriany, "Penerapan Metode Agile Dalam Pengembangan Application Programming Interface System Pada PT XYZ," *J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf.*, vol. 7, no. 1, p. 38, 2021, doi: 10.24014/coreit.v7i1.12635.
- [10] C. Ramadhan, M. A. Senubekti, and D. Amalia, "Penerapan Metodologi Agile dalam Pengembangan Perangkat Lunak," *J. Tek. Inform. dan Terap.*, vol. 3, no. 2, pp. 10–15, 2025.
- [11] Ihramsyah, V. Yasin, and Johan, "Perancangan Aplikasi Sistem Informasi Penjualan Makanan Cepat Saji Berbasis Web Studi Kasus Kedai Cheese.Box," *J. Widya*, vol. 4, no. 1, pp. 117–139, 2023, [Online].
- [12] N. Musthofa and M. A. Adiguna, "Perancangan Aplikasi E-Commerce Spare-Part Komputer Berbasis Web Menggunakan CodeIgniter Pada Dhamar Putra Computer Kota Tangerang," *OKTAL J. Ilmu Komput. dan Sains*, vol. 1, no. 03, pp. 199–207, 2022.
- [13] I. A. Ari and A. Wahid, "Perancangan Sistem Inventory Stock Packaging Material Berbasis Web Pada Pt.Amcor Specality Cartons Indonesia" *J. Cakrawala Ilm.*, vol. 2, 2023, [Online].
- [14] M. F. Pulungan and H. Purwanto, "Perancangan Sistem Informasi Penyewaan Alat Kesehatan PT XYZ Berbasis Web," *JSI (Jurnal Sist. Informasi) Univ. Suryadarma* /, pp. 79–89, 2023, doi: 10.35968/jsi.v10i2.1078.
- [15] S. Ramdany, S. A. Kaidar, B. Aguchino, C. A. A. Putri, and R. Anggie, "Penerapan UML Class Diagram dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web," *J. Ind. Eng. Syst.*, vol. 5, no. 1, 2024, doi: 10.31599/2e9afp31.
- [16] I. S. Akbar and T. Haryanti, "Pengembangan Entity Relationship Diagram Database Toko Online Ira Surabaya," *J. Ilm. Comput. Insight*, vol. 3, no. 2, pp. 28–35, 2021, doi: 10.30651/comp_insight.v3i2.12002.