

Mekanisme MVC-Pemrograman Web pada CMS yang Memiliki Basis Artificial Intelligence

Akhmat Qavidhufahmi¹, M. Ihtifanul Montaghib¹, Migel Ray Sirait¹, Yudi Setiawan², Aan Erlanshari²,
Yusran Panca Putra², Andang Wijanarko²

¹Program Studi Informatika, Universitas Bengkulu, Indonesia

²Program Studi Sistem Informasi, Universitas Bengkulu, Indonesia

Corresponding author: Akhmat Qavidhufahmi (email: ahmad61539@gmail.com)

Abstract—Pemanfaatan Model-View-Controller (MVC) dalam pengembangan Content Management System (CMS) telah menjadi standar dalam praktik pemrograman web modern (Kurniawan & Sutopo, 2021). Integrasi kecerdasan buatan (AI) pada CMS memungkinkan pengelolaan konten menjadi lebih efisien, adaptif, dan otomatis (Rahardjo & Purnomo, 2023). Artikel ini membahas penerapan arsitektur MVC dalam pengembangan CMS berbasis AI, dengan menekankan pada mekanisme kerja, keunggulan, dan tantangan integrasinya. Dengan menggabungkan pendekatan arsitektural yang terstruktur dan teknologi kecerdasan buatan yang terus berkembang, CMS masa kini mampu memberikan pengalaman pengguna yang lebih personal dan relevan (Russell & Norvig, 2021). Studi ini bertujuan memberikan pemahaman menyeluruh terhadap pendekatan desain perangkat lunak MVC dalam konteks pemrograman web dan bagaimana AI meningkatkan kinerja sistem manajemen konten secara signifikan.

Kata kunci: Pemrograman Web, Kecerdasan Buatan, Arsitektur Perangkat Lunak

I. PENDAHULUAN

Perkembangan teknologi informasi yang pesat mendorong munculnya berbagai pendekatan baru dalam pengembangan aplikasi web (Bass et al., 2021). Salah satu pendekatan yang banyak digunakan adalah arsitektur Model-View-Controller (MVC) yang memisahkan logika aplikasi, antarmuka pengguna, dan pengelolaan data secara terstruktur (Ghosh, 2022). MVC memungkinkan proses pengembangan perangkat lunak menjadi lebih modular, memudahkan pengujian serta pemeliharaan kode (Widodo, 2022). Di sisi lain, kebutuhan untuk mengelola konten secara dinamis dan cerdas dalam skala besar melahirkan CMS yang ditenagai oleh teknologi kecerdasan buatan (AI) (Nguyen & Nguyen, 2020). Seiring meningkatnya ekspektasi pengguna terhadap konten yang relevan dan tepat waktu, CMS konvensional semakin tidak mencukupi (Rahardjo & Purnomo, 2023). Hal ini mendorong pengembang untuk mengintegrasikan teknologi AI seperti machine learning, natural language processing (NLP), dan analisis prediktif ke dalam sistem manajemen konten (Goodfellow et al., 2016). CMS berbasis AI mampu memahami preferensi pengguna, menganalisis perilaku pengunjung, serta menyajikan konten secara otomatis dan adaptif (Mitchell, 2021). Kombinasi antara arsitektur MVC dan kecerdasan buatan memberikan potensi besar dalam meningkatkan efektivitas dan efisiensi pengelolaan konten dalam CMS (Bass et al., 2021). MVC menawarkan struktur pengembangan yang modular dan mudah diuji, sedangkan AI memberikan kemampuan adaptif seperti rekomendasi konten, analisis perilaku pengguna, dan automasi editorial (Nguyen & Nguyen, 2020). Artikel ini akan membahas secara mendalam

tentang mekanisme MVC dalam pemrograman web untuk CMS berbasis AI, serta mengevaluasi kelebihan, kekurangan, dan tantangan yang dihadapi dalam implementasinya.

II. TINJAUAN PUSTAKA

A. Model-View-Controller (MVC)

User Model-View-Controller (MVC) adalah pola arsitektur perangkat lunak yang digunakan secara luas dalam pengembangan aplikasi modern. Konsep ini pertama kali diperkenalkan oleh Trygve Reenskaug pada tahun 1979 dan berkembang menjadi fondasi dalam berbagai framework modern seperti Laravel, Django, Ruby on Rails, dan Spring. Dalam konteks ini, Model menangani logika bisnis dan data, View bertanggung jawab menyajikan antarmuka pengguna, dan Controller sebagai penghubung antara input pengguna dan pemrosesan logika (Bass et al., 2021; Ghosh, 2022).

Keunggulan MVC terletak pada pemisahan yang jelas antara komponen-komponen utama sistem, yang menghasilkan kode yang lebih terstruktur, mudah dipelihara, dan mendukung pengembangan skala besar (Widodo, 2022). Struktur MVC memungkinkan tim pengembang bekerja secara paralel: satu tim mengembangkan logika bisnis (model), tim lain fokus pada antarmuka pengguna (view), dan tim ketiga menangani logika aplikasi (controller).

B. Content Management System (CMS)

CMS adalah platform perangkat lunak yang memungkinkan pengguna untuk membuat, mengelola, dan mempublikasikan konten digital dengan cara yang efisien tanpa perlu memahami teknis pemrograman. CMS modern seperti WordPress, Joomla, dan Drupal menyediakan antarmuka visual yang memudahkan pengguna dalam memodifikasi struktur dan isi situs web (Hevner & Chatterjee, 2020).

Dengan berkembangnya kebutuhan bisnis dan teknologi, CMS mulai mengadopsi pendekatan arsitektur MVC untuk meningkatkan skalabilitas dan modularitas sistem. WordPress misalnya, meskipun awalnya berbasis pendekatan procedural, kini telah mendukung MVC dalam beberapa framework tematiknya. CMS headless seperti Strapi mengimplementasikan MVC secara eksplisit dan memungkinkan integrasi dengan frontend modern berbasis React atau Vue (Nguyen & Nguyen, 2020).

C. Kecerdasan Buatan dalam CMS AI

CMS Integrasi kecerdasan buatan dalam CMS membuka peluang baru dalam manajemen dan penyajian konten. Teknologi AI seperti Natural Language Processing (NLP),

Machine Learning (ML), dan Deep Learning memungkinkan sistem mengenali pola perilaku pengguna, menganalisis sentimen, dan secara otomatis menyarankan atau membuat konten baru (Goodfellow et al., 2016; Mitchell, 2021).

AI dalam CMS digunakan untuk fitur seperti rekomendasi konten personal, optimasi waktu publikasi, klasifikasi otomatis konten, bahkan pembuatan artikel melalui AI generatif. Dengan pemrosesan bahasa alami, CMS dapat mengenali topik dan preferensi pengguna sehingga menyajikan konten yang lebih relevan dan kontekstual (Russell & Norvig, 2021). Namun, penerapan AI juga menimbulkan tantangan etis seperti privasi data dan potensi bias algoritmik (Hevner & Chatterjee, 2020).

D. Pemrograman Web

CMS Pemrograman web adalah proses pengembangan aplikasi berbasis web yang dapat diakses melalui browser internet. Bahasa pemrograman yang umum digunakan mencakup HTML, CSS, dan JavaScript untuk sisi klien (frontend), serta PHP, Python, Ruby, dan Node.js untuk sisi server (backend). Dengan meningkatnya kompleksitas aplikasi web, pendekatan berbasis framework seperti Laravel (PHP), Django (Python), dan Express (Node.js) menjadi pilihan utama karena mendukung struktur MVC (Ghosh, 2022).

Pemrograman web modern juga didukung oleh teknologi REST API, GraphQL, dan WebSocket yang memungkinkan komunikasi data secara real-time. Dalam konteks CMS berbasis AI, kemampuan untuk memproses data besar secara efisien dan melakukan analisis perilaku pengguna secara dinamis menjadi sangat penting (Nguyen & Nguyen, 2020).

E. Integrasi MVC, CMS, dan AI dalam Konteks Pengembangan Web Modern

Dalam pengembangan sistem informasi modern, sinergi antara arsitektur Model-View-Controller (MVC), Content Management System (CMS), dan teknologi kecerdasan buatan (AI) menjadi fondasi penting untuk menciptakan aplikasi web yang efisien, modular, dan adaptif. Arsitektur MVC, sebagaimana dijelaskan oleh Ghosh (2022) dan Widodo (2022), menawarkan pemisahan yang tegas antara lapisan logika bisnis, antarmuka pengguna, dan kontrol alur data, sehingga mendukung pengembangan skala besar yang mudah dipelihara. Penerapan MVC pada CMS memungkinkan pengembang untuk membangun sistem yang tidak hanya terstruktur namun juga dapat diintegrasikan dengan berbagai komponen eksternal seperti API dan layanan AI.

Seiring meningkatnya kebutuhan akan personalisasi dan automasi, integrasi teknologi AI ke dalam CMS menjadi semakin krusial. Teknologi seperti Natural Language Processing (NLP), machine learning, dan deep learning digunakan untuk mendukung fitur-fitur seperti klasifikasi konten otomatis, penjadwalan publikasi cerdas, dan rekomendasi konten yang disesuaikan berdasarkan preferensi pengguna (Goodfellow et al., 2016; Mitchell, 2021; Russell & Norvig, 2021). Nguyen & Nguyen (2020) menekankan bahwa AI memungkinkan CMS tidak hanya sebagai alat manajemen konten, tetapi juga sebagai sistem yang mampu belajar dari interaksi pengguna untuk meningkatkan relevansi dan kualitas informasi yang disajikan.

Selain itu, adopsi arsitektur modular dalam framework seperti Strapi menunjukkan bahwa pendekatan headless CMS dengan dukungan AI semakin diminati (Nguyen & Nguyen, 2020). Modularitas memungkinkan integrasi plugin seperti analisis sentimen dan tag otomatis, yang bekerja berdampingan dengan controller dan model dalam struktur MVC. Pendekatan ini secara langsung meningkatkan skalabilitas dan fleksibilitas sistem, serta membuka jalan bagi pengembangan CMS yang tidak hanya informatif tetapi juga proaktif dalam memahami dan memenuhi kebutuhan pengguna (Rahardjo & Purnomo, 2023).

Namun demikian, tantangan tetap ada. Widodo (2022) mencatat bahwa kompleksitas integrasi antara modul AI dan sistem MVC dapat menjadi hambatan dalam implementasi di dunia nyata. Selain itu, isu privasi data dan bias algoritmik sebagaimana diingatkan oleh Russell & Norvig (2021), perlu mendapatkan perhatian serius dalam perancangan sistem CMS berbasis AI. Oleh karena itu, pengembangan CMS modern tidak hanya harus mempertimbangkan aspek teknis dan arsitektural, tetapi juga dimensi etika dan regulatif yang mengiringinya (Hevner & Chatterjee, 2020).

F. Integrasi Arsitektur MVC, CMS, dan Kecerdasan Buatan: Kajian Konseptual dan Praktis

Dalam dekade terakhir, perkembangan teknologi perangkat lunak menunjukkan kecenderungan kuat ke arah sistem yang modular, adaptif, dan cerdas. Salah satu fondasi arsitektur dalam pengembangan web modern adalah pola Model-View-Controller (MVC), yang pertama kali diperkenalkan oleh Trygve Reenskaug dan kini menjadi standar dalam berbagai framework seperti Laravel, Django, dan Spring (Ghosh, 2022; Widodo, 2022). Struktur MVC memungkinkan pengembangan yang lebih terorganisir melalui pemisahan tanggung jawab: *Model* menangani logika bisnis dan pengelolaan data, *View* menyajikan data kepada pengguna, sedangkan *Controller* bertindak sebagai perantara antara keduanya (Bass et al., 2021).

Penerapan arsitektur MVC dalam pengembangan Content Management System (CMS) menghadirkan peluang untuk menciptakan sistem yang tidak hanya efisien dan terstruktur, tetapi juga mudah dikembangkan dan dipelihara dalam jangka panjang. CMS seperti WordPress, meskipun awalnya tidak dirancang dengan pola MVC murni, mulai mengadopsi prinsip-prinsip ini pada pengembangan tema dan plugin-nya (Hevner & Chatterjee, 2020). Di sisi lain, CMS headless seperti Strapi mengadopsi MVC secara eksplisit dengan pemisahan komponen backend dan frontend, sehingga mendukung integrasi yang lebih fleksibel dengan aplikasi berbasis React atau Vue (Nguyen & Nguyen, 2020).

Di tengah tuntutan terhadap sistem yang semakin responsif terhadap perilaku pengguna, integrasi teknologi kecerdasan buatan (AI) menjadi krusial dalam CMS modern. Teknologi AI seperti *machine learning*, *natural language processing* (NLP), dan *deep learning* berperan dalam menyajikan konten secara otomatis berdasarkan analisis data pengguna dan konteks waktu (Goodfellow et al., 2016; Mitchell, 2021). Dalam konteks ini, AI tidak hanya meningkatkan efisiensi manajemen konten, tetapi juga memperkaya pengalaman pengguna melalui personalisasi konten, klasifikasi otomatis, hingga penulisan artikel berbasis AI generatif (Russell & Norvig, 2021).

Rahardjo & Purnomo (2023) menegaskan bahwa penerapan AI dalam CMS membuka ruang bagi transformasi digital yang menyeluruh, terutama dalam hal otomatisasi editorial dan penyusunan strategi konten berbasis data. Misalnya, dengan menggunakan algoritma machine learning, CMS dapat mempelajari pola kunjungan pengguna, menentukan waktu publikasi terbaik, serta menyarankan topik yang paling relevan. Dalam implementasinya, modul AI ini sering kali dihubungkan ke lapisan *model* atau *controller* dalam arsitektur MVC, memungkinkan sistem untuk secara real-time menyesuaikan tampilan (*view*) sesuai hasil analisis (Nguyen & Nguyen, 2020).

Namun, integrasi antara AI dan MVC bukan tanpa tantangan. Kompleksitas teknis dalam membangun interaksi antara modul AI dan komponen MVC membutuhkan keahlian lintas disiplin dalam software engineering dan data science (Widodo, 2022). Di samping itu, aspek etika dan regulatif seperti perlindungan data pribadi dan keberpihakan algoritma menjadi perhatian yang harus diantisipasi sejak tahap perancangan sistem (Hevner & Chatterjee, 2020; Russell & Norvig, 2021). Kebutuhan sumber daya komputasi yang tinggi juga menjadi hambatan tersendiri, mengingat pemrosesan model AI dalam CMS memerlukan dukungan infrastruktur seperti GPU dan memori besar (Goodfellow et al., 2016).

Studi implementasi pada sistem seperti Strapi menunjukkan potensi besar dari CMS modular yang dilengkapi AI. Sistem ini tidak hanya mendukung pembuatan konten secara fleksibel melalui REST atau GraphQL API, tetapi juga memungkinkan integrasi plugin AI seperti klasifikasi konten, penilaian sentimen, dan rekomendasi judul berita (Nguyen & Nguyen, 2020). Pendekatan semacam ini mendemonstrasikan bagaimana CMS berbasis MVC-AI mampu menggabungkan keunggulan arsitektural dan kecerdasan komputasional dalam satu platform.

Secara keseluruhan, perpaduan antara MVC, CMS, dan AI dalam pengembangan web mencerminkan arah evolusi sistem informasi yang lebih cerdas, modular, dan dinamis. Sementara tantangan tetap ada, seperti integrasi teknologi dan kepatuhan terhadap regulasi, prospek pemanfaatan sistem ini dalam berbagai sektor—seperti pendidikan, e-commerce, dan pemerintahan—sangat menjanjikan (Bass et al., 2021; Rahardjo & Purnomo, 2023). Dengan pendekatan arsitektural yang matang dan penerapan AI yang etis dan bertanggung jawab, CMS modern dapat menjadi landasan utama transformasi digital di masa depan.

III. HASIL DAN PEMBAHASAN

A. Implementasi MVC dalam CMS

Pada Framework modern seperti Laravel, Django, dan Ruby on Rails secara eksplisit menerapkan MVC dalam pengembangan CMS (Ghosh, 2022). Misalnya, Laravel (PHP) menggunakan routing yang diarahkan ke controller, memanipulasi model, dan menghasilkan view. Django (Python), di sisi lain, mengusung konsep MTV (Model-Template-View) yang secara fungsional serupa dengan MVC. Dalam kedua framework ini, pengembang dapat memisahkan fungsi logika bisnis dari presentasi antarmuka (Widodo, 2022).

B. Integrasi AI ke dalam Arsitektur

Perkembangan MVC Integrasi AI ke dalam sistem MVC dilakukan dengan menambahkan layer tambahan atau modul AI yang berinteraksi dengan controller dan model (Nguyen & Nguyen, 2020). Contohnya, modul rekomendasi konten dapat mengambil data pengguna dari model, memprosesnya dengan algoritma machine learning, lalu mengembalikan hasil ke controller yang menampilkannya melalui view (Rahardjo & Purnomo, 2023).

C. Studi Kasus : CMS AI Modular

Pada Salah satu contoh CMS yang mulai mengadopsi pendekatan modular berbasis AI adalah "Strapi". Strapi adalah CMS headless open-source yang mendukung penggunaan plugin eksternal dan integrasi dengan layanan AI (Nguyen & Nguyen, 2020). Modul AI seperti sentiment analysis, content tagging otomatis, dan rekomendasi headline telah dapat diintegrasikan dengan sistem manajemen konten Strapi.

D. Keuntungan dan Tantangan

Keuntungan	Tantangan
Peningkatan personalisasi konten berdasarkan perilaku dan preferensi pengguna (Rahardjo & Purnomo, 2023)	Kompleksitas integrasi antara modul AI dan sistem MVC yang sudah ada (Widodo, 2022)
Efisiensi pengelolaan data dan otomatisasi tugas editorial (Mitchell, 2021)	Masalah privasi data pengguna dan kepatuhan terhadap regulasi seperti GDPR (Russell & Norvig, 2021)
Skalabilitas sistem yang lebih baik melalui arsitektur modular (Bass et al., 2021)	Kebutuhan sumber daya komputasi yang tinggi, terutama untuk model AI yang kompleks (Goodfellow et al., 2016)

E. Tabel Perbandingan: CMS Berbasis AI vs Non-AI dalam Konteks MVC

Aspek	CMS Berbasis AI (dengan MVC)	CMS Non-AI (dengan MVC)
Arsitektur	MVC dengan integrasi modul AI	MVC konvensional
Rekomendasi Konten	Otomatis berdasarkan perilaku pengguna (AI-driven)	Manual, ditentukan oleh admin
Otomasi	Tinggi, meliputi tagging, jadwal publikasi, klasifikasi konten	Rendah, sebagian besar berbasis input manual

Skalabilitas	Tinggi, mendukung modularitas dan pemrosesan data besar	Menengah, tergantung desain awal
Responsivitas Sistem	Adaptif terhadap perilaku pengguna secara real-time	Tidak adaptif, tetap meskipun perilaku pengguna berubah
Integrasi Eksternal	Mendukung API untuk AI (NLP, ML, vision)	Umumnya API terbatas pada CRUD standar
Analisis Performa Konten	Disediakan melalui dashboard berbasis AI	Umumnya perlu plugin tambahan atau analisis manual
Kebutuhan Infrastruktur	Lebih tinggi (memerlukan GPU, RAM besar untuk model AI)	Lebih ringan, cukup server umum
Kustomisasi User	Otomatis, profil pengguna disesuaikan melalui AI	Statis, sesuai pengaturan admin
Contoh Implementasi	Strapi + AI Plugin, Django CMS + NLP Python	WordPress + MVC Theme, Joomla (tanpa AI plugin)

F. Analisis Implementasi dan Integrasi MVC-AI dalam CMS

Implementasi arsitektur Model-View-Controller (MVC) dalam sistem Content Management System (CMS) memberikan fondasi teknis yang kuat dalam membangun aplikasi web yang modular, scalable, dan mudah dikembangkan. Laravel sebagai framework berbasis PHP dan Django berbasis Python, merupakan contoh nyata penerapan pola MVC yang telah terbukti meningkatkan efisiensi pengembangan serta pemeliharaan sistem (Ghosh, 2022; Widodo, 2022). Struktur MVC pada Laravel misalnya, memanfaatkan *controller* untuk mengatur alur logika aplikasi yang menghubungkan model data dan tampilan pengguna. Pemisahan ini memudahkan integrasi dengan layanan pihak ketiga seperti API atau modul AI eksternal, sekaligus mempercepat debugging dan pengujian sistem (Bass et al., 2021).

Di sisi lain, penerapan kecerdasan buatan dalam CMS menghadirkan pendekatan baru terhadap pengelolaan konten, terutama dalam hal personalisasi dan otomasi. CMS berbasis AI memungkinkan sistem mengenali perilaku dan preferensi pengguna untuk menyarankan atau bahkan membuat konten yang sesuai secara otomatis (Russell & Norvig, 2021). Misalnya, dengan menggunakan pendekatan *machine learning* yang dijelaskan oleh Mitchell (2021), sistem dapat melakukan analisis terhadap pola interaksi pengguna untuk

menghasilkan rekomendasi konten yang tepat waktu dan relevan.

Strapi sebagai CMS headless modular menunjukkan bagaimana integrasi AI dapat dilakukan dalam kerangka arsitektur MVC. Dalam studi yang dilakukan Nguyen & Nguyen (2020), disebutkan bahwa Strapi memungkinkan pengembangan CMS yang fleksibel dengan penggunaan plugin berbasis AI seperti *sentiment analysis*, *auto-tagging*, dan *headline generation*. Dalam konteks MVC, modul AI ini biasanya diintegrasikan pada lapisan *controller* atau secara langsung pada *model* untuk mengakses dan memproses data pengguna secara real-time sebelum ditampilkan di *view*. Hal ini mencerminkan mekanisme integrasi vertikal antara AI dan struktur MVC yang menjaga keterpisahan fungsi sekaligus mendukung interoperabilitas sistem.

Namun, pembahasan hasil tidak akan lengkap tanpa memperhatikan tantangan dan keterbatasan dari pendekatan ini. Salah satu tantangan utama adalah meningkatnya kompleksitas teknis akibat penambahan lapisan AI di atas arsitektur MVC yang sudah ada (Widodo, 2022). Modul AI, seperti model NLP atau rekomendasi konten, memerlukan pemrosesan data besar dan algoritma yang kompleks, yang pada gilirannya membutuhkan sumber daya komputasi tinggi (Goodfellow et al., 2016). Selain itu, persoalan privasi pengguna dan etika algoritma juga menjadi perhatian penting yang harus diantisipasi oleh pengembang, seperti yang ditekankan oleh Russell & Norvig (2021) serta Hevner & Chatterjee (2020).

Analisis perbandingan antara CMS berbasis AI dan CMS konvensional menunjukkan bahwa sistem berbasis AI dengan dukungan MVC menawarkan responsivitas yang lebih tinggi, skalabilitas sistem yang lebih baik, dan kemampuan adaptasi terhadap perubahan perilaku pengguna secara real-time (Rahardjo & Purnomo, 2023). Sebaliknya, CMS non-AI cenderung bersifat statis dan memerlukan intervensi manual dalam pengelolaan konten, sehingga kurang efisien ketika dihadapkan pada volume data dan permintaan pengguna yang besar dan dinamis.

Secara keseluruhan, pembahasan ini menunjukkan bahwa integrasi MVC dan AI dalam CMS tidak hanya memperbaiki cara sistem dikembangkan dan dipelihara, tetapi juga meningkatkan performa serta kualitas interaksi pengguna dengan konten. Meski menghadirkan tantangan dalam hal integrasi, sumber daya, dan regulasi, pendekatan ini mencerminkan arah masa depan dari sistem informasi berbasis web yang semakin cerdas, adaptif, dan manusiawi.

G. Analisis Implementasi dan Integrasi MVC-AI dalam CMS: Hasil dan Pembahasan Lanjutan

Implementasi arsitektur Model-View-Controller (MVC) dalam sistem Content Management System (CMS) memberikan fondasi teknis yang kuat dalam membangun aplikasi web yang modular, scalable, dan mudah dikembangkan. Laravel sebagai framework berbasis PHP dan Django berbasis Python, merupakan contoh nyata penerapan pola MVC yang telah terbukti meningkatkan efisiensi pengembangan serta pemeliharaan sistem (Ghosh, 2022; Widodo, 2022). Struktur MVC pada Laravel misalnya, memanfaatkan *controller* untuk mengatur alur logika aplikasi yang menghubungkan model data dan tampilan pengguna. Pemisahan ini memudahkan integrasi dengan layanan pihak

ketiga seperti API atau modul AI eksternal, sekaligus mempercepat debugging dan pengujian sistem (Bass et al., 2021).

Di sisi lain, penerapan kecerdasan buatan dalam CMS menghadirkan pendekatan baru terhadap pengelolaan konten, terutama dalam hal personalisasi dan otomatisasi. CMS berbasis AI memungkinkan sistem mengenali perilaku dan preferensi pengguna untuk menyarankan atau bahkan membuat konten yang sesuai secara otomatis (Russell & Norvig, 2021). Misalnya, dengan menggunakan pendekatan *machine learning* yang dijelaskan oleh Mitchell (2021), sistem dapat melakukan analisis terhadap pola interaksi pengguna untuk menghasilkan rekomendasi konten yang tepat waktu dan relevan.

Strapi sebagai CMS headless modular menunjukkan bagaimana integrasi AI dapat dilakukan dalam kerangka arsitektur MVC. Dalam studi yang dilakukan Nguyen & Nguyen (2020), disebutkan bahwa Strapi memungkinkan pengembangan CMS yang fleksibel dengan penggunaan plugin berbasis AI seperti *sentiment analysis*, *auto-tagging*, dan *headline generation*. Dalam konteks MVC, modul AI ini biasanya diintegrasikan pada lapisan *controller* atau secara langsung pada *model* untuk mengakses dan memproses data pengguna secara real-time sebelum ditampilkan di *view*. Hal ini mencerminkan mekanisme integrasi vertikal antara AI dan struktur MVC yang menjaga keterpisahan fungsi sekaligus mendukung interoperabilitas sistem.

Namun, pembahasan hasil tidak akan lengkap tanpa memperhatikan tantangan dan keterbatasan dari pendekatan ini. Salah satu tantangan utama adalah meningkatnya kompleksitas teknis akibat penambahan lapisan AI di atas arsitektur MVC yang sudah ada (Widodo, 2022). Modul AI, seperti model NLP atau rekomendasi konten, memerlukan pemrosesan data besar dan algoritma yang kompleks, yang pada gilirannya membutuhkan sumber daya komputasi tinggi (Goodfellow et al., 2016). Selain itu, persoalan privasi pengguna dan etika algoritma juga menjadi perhatian penting yang harus diantisipasi oleh pengembang, seperti yang ditekankan oleh Russell & Norvig (2021) serta Hevner & Chatterjee (2020).

Analisis perbandingan antara CMS berbasis AI dan CMS konvensional menunjukkan bahwa sistem berbasis AI dengan dukungan MVC menawarkan responsivitas yang lebih tinggi, skalabilitas sistem yang lebih baik, dan kemampuan adaptasi terhadap perubahan perilaku pengguna secara real-time (Rahardjo & Purnomo, 2023). Sebaliknya, CMS non-AI cenderung bersifat statis dan memerlukan intervensi manual dalam pengelolaan konten, sehingga kurang efisien ketika dihadapkan pada volume data dan permintaan pengguna yang besar dan dinamis.

Secara keseluruhan, pembahasan ini menunjukkan bahwa integrasi MVC dan AI dalam CMS tidak hanya memperbaiki cara sistem dikembangkan dan dipelihara, tetapi juga meningkatkan performa serta kualitas interaksi pengguna dengan konten. Meski menghadirkan tantangan dalam hal integrasi, sumber daya, dan regulasi, pendekatan ini mencerminkan arah masa depan dari sistem informasi berbasis web yang semakin cerdas, adaptif, dan manusiawi.

KESIMPULAN

Perkembangan Penerapan arsitektur MVC dalam pengembangan CMS berbasis AI memberikan manfaat besar dalam hal struktur sistem dan kecerdasan pengelolaan konten (Bass et al., 2021). AI memperkaya fungsi CMS melalui personalisasi dan otomatisasi, sedangkan MVC memastikan sistem tetap modular dan mudah dipelihara (Widodo, 2022). Dengan memanfaatkan AI, CMS mampu menyajikan konten yang lebih relevan dan responsif terhadap kebutuhan pengguna (Nguyen & Nguyen, 2020).

Studi perbandingan CMS menunjukkan bahwa fleksibilitas dan kekuatan AI sangat tergantung pada arsitektur dan bahasa pemrograman yang digunakan. Di masa depan, tren teknologi seperti AI generatif dan CMS headless akan memainkan peran penting dalam evolusi sistem manajemen konten. Namun, pemanfaatan AI dalam CMS harus mempertimbangkan aspek sosial dan etis agar tidak menimbulkan dampak negatif terhadap pengguna dan masyarakat secara umum (Russell & Norvig, 2021).

Kombinasi MVC dan AI membuka peluang besar dalam pengembangan sistem manajemen konten yang tidak hanya efisien tetapi juga mampu belajar dan berkembang seiring waktu. Dengan inovasi dan penerapan etis, CMS berbasis AI berpotensi menjadi fondasi utama dalam transformasi digital di berbagai sektor industri, pemerintahan, dan pendidikan.

Penerapan arsitektur Model-View-Controller (MVC) dalam pengembangan Content Management System (CMS) telah terbukti memberikan struktur perangkat lunak yang modular, terorganisir, dan mudah dipelihara. Dengan pemisahan antara logika aplikasi, tampilan, dan pengelolaan data, MVC memungkinkan proses pengembangan perangkat lunak menjadi lebih efisien dan kolaboratif (Bass et al., 2021; Ghosh, 2022; Widodo, 2022). Ketika arsitektur ini diintegrasikan dengan teknologi kecerdasan buatan (AI), CMS tidak hanya berfungsi sebagai alat manajemen konten, tetapi juga berkembang menjadi sistem yang adaptif, cerdas, dan berorientasi pengguna.

Teknologi AI seperti machine learning, natural language processing (NLP), dan deep learning memungkinkan CMS untuk menganalisis perilaku pengguna, menyajikan konten yang relevan, serta mengotomatisasi proses editorial (Goodfellow et al., 2016; Mitchell, 2021; Russell & Norvig, 2021). Studi implementasi seperti pada CMS Strapi menunjukkan bahwa pendekatan modular berbasis MVC dan AI dapat mendukung integrasi layanan cerdas seperti rekomendasi konten, analisis sentimen, dan tagging otomatis secara efisien (Nguyen & Nguyen, 2020). Dengan demikian, integrasi ini mampu meningkatkan kualitas interaksi pengguna dan efektivitas pengelolaan konten secara signifikan.

Meskipun menawarkan berbagai keunggulan, penerapan AI dalam arsitektur MVC menghadirkan tantangan, terutama dalam hal kompleksitas integrasi, kebutuhan sumber daya komputasi, serta isu-isu etis terkait privasi data dan bias algoritmik (Hevner & Chatterjee, 2020; Widodo, 2022). Oleh karena itu, pengembangan CMS masa depan perlu memperhatikan prinsip keterbukaan, keamanan, dan keberlanjutan sistem dalam penerapan teknologi AI.

Secara keseluruhan, kombinasi MVC dan AI dalam CMS merepresentasikan lompatan besar dalam evolusi sistem informasi web. Tidak hanya meningkatkan fleksibilitas teknis

dan efisiensi pengembangan, tetapi juga menciptakan pengalaman pengguna yang lebih personal, responsif, dan kontekstual. Dengan pendekatan arsitektural yang matang dan penerapan AI yang bertanggung jawab, CMS berbasis AI dan MVC memiliki potensi besar sebagai fondasi transformasi digital di berbagai sektor—mulai dari pendidikan, bisnis, media, hingga pemerintahan.

DAFTAR PUSTAKA

- [1] Bass, L., Clements, P., & Kazman, R. (2021). *Software Architecture in Practice*. Addison-Wesley.
- [2] Ghosh, S. (2022). *Beginning Laravel: Build and Deploy Web Applications with Laravel*. Apress.
- [3] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- [4] Hevner, A. R., & Chatterjee, S. (2020). *Design Research in Information Systems: Theory and Practice*. Springer.
- [5] Kurniawan, A., & Sutopo, W. (2021). "Implementasi Model MVC pada Sistem Informasi Berbasis Web". *Jurnal Teknologi Informasi dan Ilmu Komputer*, 8(2), 167-175.
- [6] Mitchell, T. M. (2021). *Machine Learning*. McGraw-Hill.
- [7] Nguyen, T. H., & Nguyen, V. (2020). "AI-Driven CMS in Modern Web Development". *International Journal of Computer Applications*, 175(20), 10-15.
- [8] Rahardjo, B., & Purnomo, H. (2023). "Penerapan AI dalam CMS untuk Personalisasi Konten Web". *Jurnal Teknologi dan Sistem Komputer*, 11(1), 44-50.
- [9] Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach*. Pearson.
- [10] Widodo, S. (2022). "Arsitektur MVC dalam Pengembangan Web Modern". *Seminar Nasional Teknologi Informasi dan Multimedia*, 9(1), 30-36.