

Pengaruh Implementasi *Headless Content Management System* Terhadap Performa dan Kinerja *E-Learning*

Muhammad Ramzi Adhitya¹, I Made Suartana²

^{1,3} Jurusan Teknik Informatika/Program Studi S1 Teknik Informatika, Universitas Negeri Surabaya

¹muhammadramzi.21037@mhs.unesa.ac.id

²madesuartana@unesa.ac.id

Abstrak— Perkembangan sistem e-learning saat ini masih menghadapi berbagai kendala, terutama dalam hal performa dan kecepatan akses. Permasalahan seperti waktu muat halaman yang lambat dan kinerja sistem yang kurang optimal dapat mengganggu proses pembelajaran. Penelitian ini bertujuan untuk mengimplementasikan arsitektur Headless Content Management System (CMS) pada website e-learning guna meningkatkan performa dan fleksibilitas sistem. Penelitian menggunakan Strapi sebagai Headless CMS, React.js sebagai frontend, Railway untuk hosting backend, dan Vercel untuk hosting frontend. Pengujian dilakukan dengan empat konten kursus yang memiliki ukuran data bervariasi mulai dari 50 MB hingga 300 MB. Hasil pengujian menunjukkan bahwa website mampu merespons dengan waktu render di bawah 1,5 detik dan skor performa konsisten di atas 90. Metrik seperti First Contentful Paint (FCP), Largest Contentful Paint (LCP), Total Blocking Time (TBT), dan Speed Index menunjukkan performa yang optimal dan stabil. Implementasi Headless CMS pada website e-learning memberikan hasil performa secara signifikan, ditunjukkan dengan nilai rata-rata Speed Index sebesar 1 detik, Total Blocking Time hanya 6 milidetik, serta skor performa Lighthouse mencapai 93,6.

Kata Kunci— : E-learning, Headless CMS, Strapi, React.js, Performa Website

I. PENDAHULUAN

Perkembangan teknologi informasi yang sangat pesat telah merubah segala aspek kehidupan, termasuk di dalam sektor pendidikan. E-learning atau LMS (Learning Management System) hadir sebagai solusi efektif untuk menjawab kebutuhan metode pembelajaran berbasis teknologi untuk menjawab tantangan pendidikan di era digital [1]. Namun, seiring perkembangan zaman, ekspektasi pengguna dan kebutuhan juga ikut meningkat, performa dan kecepatan akses e-learning juga menjadi salah satu faktor krusial yang mempengaruhi pengalaman pengguna. Terutama ketika menampilkan konten dengan berbagai macam bentuk format seperti video, gambar, ataupun materi. Hal ini tentunya akan menjadi faktor utama yang sangat mempengaruhi pengalaman pengguna dalam menggunakan e-learning.

Sistem e-learning yang banyak beredar tidak jarang mengalami berbagai kendala dalam hal kecepatan loading halaman yang lambat serta kinerja yang kurang optimal. Hal ini bisa disebabkan oleh beberapa faktor yaitu beban kerja server yang berat, jumlah permintaan HTTP yang tinggi, dan minimnya optimasi pada konten dan kode yang terdapat pada e-learning [2]. Sebagai contoh, setiap kali pengguna mengakses konten, server harus memproses permintaan tersebut, apabila tidak dikelola dengan baik, dapat

menyebabkan penundaan yang signifikan. Hal ini berakibat menurunnya efektivitas pembelajaran dan kepuasan pengguna.

Berdasarkan penelitian yang dilakukan oleh Muchamad Lutfi Maftuh dan Nurkhamid yang berjudul “Implementasi *Next.js* dan *Headless CMS* Untuk Peningkatan Kualitas *Website* MI AL ISLAM Pare”, dilakukan pengujian terhadap *website* dengan 4 parameter yaitu *Functional Suitability*, *Usability*, *Reliability*, dan *Performance Efficiency*. Didapatkan hasil untuk setiap parameter pengujian adalah sebagai berikut, untuk *Functional Suitability* mendapatkan skor tingkat kelayakan sebesar 98,7% dan dikategorikan “Sangat Layak”. Hasil pengujian *Usability* mendapatkan skor 75,3%, dari hasil pengujian tersebut untuk parameter *Usability* dikategorikan “Layak”. Selanjutnya pengujian parameter *Reliability* didapatkan skor total 1 atau dikonversikan ke dalam persentase menjadi 100% yang telah memenuhi parameter *Reliability* menurut standar Telcordia yaitu 95%. Yang terakhir dilakukan pengujian *Performance Efficiency* dan didapatkan skor rata-rata 94,2 untuk *mobile* dan 99,86 untuk *desktop* berdasarkan hasil yang didapat, *website* tersebut telah memenuhi standar ISO/EIC 25010 [3]. Penggunaan *Headless CMS* dapat dijadikan salah satu solusi untuk mengatasi permasalahan *e-learning* yang banyak beredar terutama dari segi performa dan kecepatan halaman, kendala dalam pengalaman belajar bisa saja terganggu akibat lambatnya akses materi ketika pengguna mengakses suatu materi pembelajaran.

Headless Content Management System (CMS) memberikan solusi yang inovatif untuk menjawab permasalahan tersebut. Headless Content Management System (CMS) bekerja dengan cara memisahkan backend dan frontend, sehingga memungkinkan untuk mengirim konten yang lebih cepat dan lebih fleksibel. Pada Headless Content Management System (CMS), Backend berperan sebagai penyedia konten melalui API, sementara frontend bisa dikembangkan menggunakan teknologi yang paling terkini untuk mendapatkan performa yang optimal [4]. Menggunakan Headless Content Management System bukan hanya mempercepat kecepatan loading halaman, tetapi juga memberikan fleksibilitas yang besar dalam pengembangan sistem e-learning.

Dalam penelitian ini dibahas pengaruh implementasi headless content management system terhadap performa dan kinerja e-learning. Penelitian ini juga bertujuan untuk mengidentifikasi dan mengatasi permasalahan yang dihadapi oleh sistem e-learning tradisional serta mengembangkan

sebuah platform e-learning berbasis Headless Content Management System yang diharapkan mampu memberikan performa dan kecepatan halaman yang lebih baik.

II. METODOLOGI PENELITIAN

Pada penelian ini terdapat alur penelitian guna mengetahui dan mengidentifikasi langkah-langkah dalam pengembangan website e-learning.



Gbr. 1 Flowchart Penelitian

Pada Gbr. 1 merupakan alur penelitian yang akan dilakukan, seperti yang terlihat pada *flowchart* tersebut, penelitian ini akan dilaksanakan pada enam tahapan yaitu identifikasi kebutuhan pengguna, tahap perancangan website, tahap pengembangan website, tahap pengujian website, analisis hasil, dan kesimpulan.

A. Identifikasi Kebutuhan Pengguna

Tahap pertama dalam penelitian ini adalah identifikasi kebutuhan pengguna. Pada tahap ini, peneliti bertujuan untuk merumuskan fitur-fitur yang akan dikembangkan dalam sistem e-learning. Penentuan fitur dilakukan melalui kajian literatur serta diskusi dengan para ahli guna mengidentifikasi komponen dan fitur utama yang diperlukan. Aspek-aspek yang dianalisis meliputi kebutuhan fungsional, kebutuhan non-fungsional, kriteria keberhasilan fitur, serta perencanaan tahapan pengembangan (roadmap). Melalui pendekatan ini, diharapkan proses pengembangan sistem e-learning dapat

lebih terarah, fokus pada tujuan utama, dan mampu memenuhi kebutuhan pengguna secara optimal.

B. Tahap Perancangan Website

Pada tahap perancangan sistem, peneliti merancang prototipe website e-learning berdasarkan pendekatan Headless Content Management System (CMS). Proses perancangan ini mencakup beberapa tahapan, yaitu konfigurasi backend menggunakan Strapi, konfigurasi API dengan pendekatan RESTful API, pengembangan antarmuka frontend menggunakan React.js, serta proses hosting dan deployment menggunakan Railway dan Vercel. Rancangan prototipe ini dijelaskan secara rinci pada poin-poin berikut:

1) Konfigurasi Backend (Strapi)

Strapi merupakan CMS headless berbasis open-source yang dikembangkan oleh Strapi Solutions SAS. Platform ini memungkinkan pengguna untuk membuat model konten dan menyajikannya melalui API secara otomatis. Salah satu fitur utama Strapi adalah penyediaan REST API endpoint yang dapat digunakan untuk menjalankan operasi CRUD dan logika bisnis lainnya dalam pengembangan sistem e-learning [5].

Dalam penelitian ini, peneliti merancang beberapa jenis *content type* seperti *Course*, *Assignment*, *User*, *Instructor*, dan *Student*. Setelah konten dibuat, Strapi akan secara otomatis menghasilkan endpoint API yang dapat dimanfaatkan pada sisi frontend [10].

2) Konfigurasi API (RESTful API)

Setelah konfigurasi konten selesai dilakukan, langkah berikutnya adalah menghubungkan API yang dihasilkan oleh Strapi dengan lapisan presentasi (frontend). Strapi menyediakan dua opsi akses API, yaitu melalui GraphQL atau RESTful API. Dalam penelitian ini, peneliti memilih RESTful API sebagai metode komunikasi antara backend dan frontend. API ini akan dikonsumsi oleh aplikasi frontend yang dikembangkan menggunakan React.js.

3) Pengembangan Frontend (React.js)

React.js merupakan pustaka JavaScript bersifat open-source yang dikembangkan oleh Facebook, dan banyak digunakan dalam pengembangan antarmuka pengguna yang dinamis dan responsif. Keunggulan utama React.js terletak pada arsitektur berbasis komponen dan pemanfaatan Virtual DOM, yang memungkinkan pembuatan UI yang efisien dan *reusable* [6].

Pada penelitian ini, React.js digunakan untuk mengembangkan antarmuka pengguna dengan melakukan pengambilan data (fetching) dari Strapi melalui REST API menggunakan *library* Axios.

4) Hosting Backend (Railway)

Railway adalah platform *cloud-based* yang memungkinkan pengembang membangun dan menjalankan aplikasi secara menyeluruh di cloud. Railway menyediakan dukungan untuk berbagai bahasa pemrograman, termasuk Node.js yang digunakan oleh Strapi. Oleh karena itu, peneliti memilih Railway sebagai platform hosting backend karena kemudahan konfigurasi serta dukungannya terhadap kebutuhan pengembangan [7].

5) Hosting Frontend (Vercel)

Frontend aplikasi di hosting menggunakan Vercel, sebuah platform penyedia layanan hosting untuk framework frontend dan situs statis. Vercel dipilih karena menawarkan solusi hosting yang cepat, efisien, dan mendukung pengembangan React.js dengan konfigurasi minimal. Selain itu, Vercel menyediakan layanan gratis yang cukup optimal untuk tahap pengembangan dan pengujian [8].

C. Tahap Perancangan Headless CMS

Dalam pengembangan sistem e-learning, pendekatan Headless Content Management System (Headless CMS) dipilih guna memisahkan manajemen konten (backend) dari antarmuka pengguna (frontend). Pendekatan ini memberikan fleksibilitas yang lebih tinggi dalam proses pengembangan, karena memungkinkan backend dan frontend untuk dikembangkan, diperbarui, serta dihosting secara terpisah dan independen.

1) Konsep Headless CMS

Headless CMS adalah sistem manajemen konten yang tidak terikat langsung dengan sistem frontend. Konten – konten yang dikelola akan disediakan dalam bentuk API (Application Programming Interface) yang nantinya dapat dikonsumsi oleh berbagai jenis frontend seperti website, aplikasi mobile, dll.

2) Pemilihan Strapi sebagai Headless CMS

Peneliti memilih Strapi sebagai penyedia Headless CMS untuk kebutuhan manajemen konten backend karena :

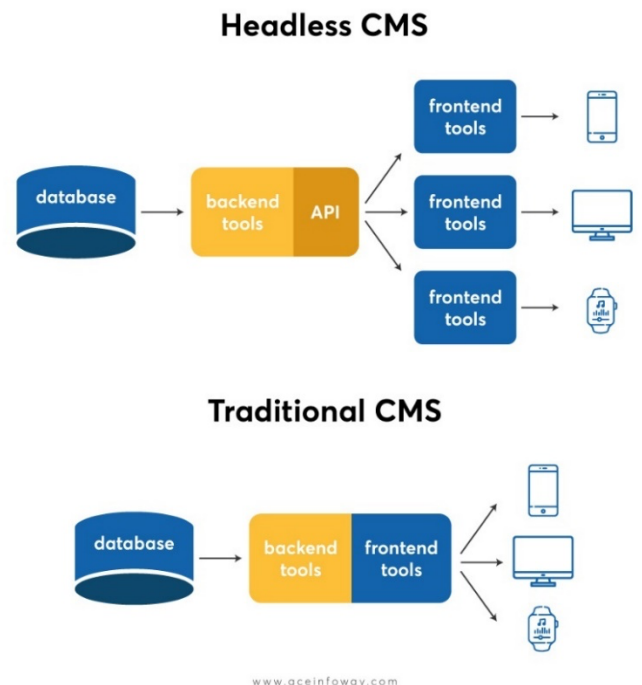
- Mendukung pembuatan konten secara modular melalui fitur Content- Type Builder.
- Mendukung penggunaan REST API
- Mendukung autentikasi, relasi antar konten, dan upload media
- Dapat dihosting secara fleksibel di berbagai platform hosting

3) Perbedaan arsitektur Headless CMS dan Tradisional CMS

Alasan pemilihan Strapi sebagai Headless CMS juga karena keunggulannya dibandingkan dengan CMS tradisional (monolitik) seperti wordpress konvensional, frontend dan backend berada pada satu platform, sehingga perubahan pada frontend sangat bergantung struktur backend yang disediakan

oleh sistem. Hal ini dapat mengurangi fleksibilitas dalam hal desain dan pengalaman pengguna.

Berikut ini adalah perbedaan dalam pengembangan dalam arsitektur Headless CMS dan arsitektur tradisional dapat dilihat pada Gbr. 2 berikut ini.



Gbr. 2 Arsitektur Headless CMS

Seperti yang terlihat pada Gbr. 2, Headless CMS memiliki arsitektur backend dan frontend yang dikembangkan secara modular (terpisah) sedangkan CMS tradisional memiliki arsitektur monolitik yang dimana frontend dan backend dikembangkan pada satu repository yang sama.

Hal ini tentunya Headless CMS lebih unggul dari segi fleksibilitas dan skalabilitas apabila backend atau frontend mengalami perubahan. Dalam hal skalabilitas, frontend bisa dikembangkan ke berbagai platform seperti mobile apps, website, bahkan IoT.

4) Arsitektur Sistem

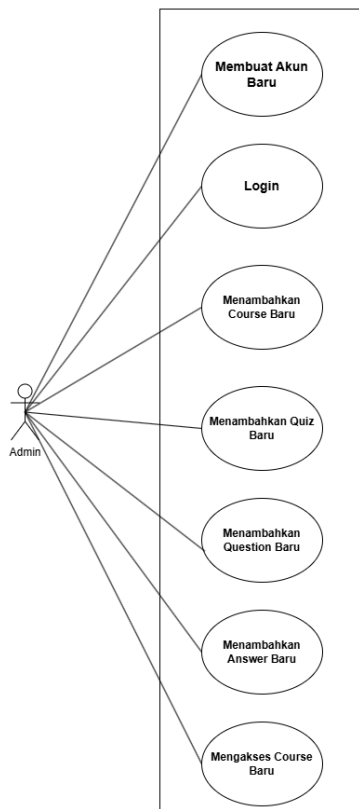
Sistem e-learning ini akan dibangun menggunakan arsitektur Headless CMS dengan struktur sebagai berikut :

- Strapi CMS : Bertindak sebagai backend yang akan menyimpan dan mengelola data course, user, video, dan quiz
- React.js : Bertindak sebagai antarmuka pengguna yang nantinya akan menampilkan data melalui konsumsi API dari backend.
- Komunikasi antara frontend dan backend dilakukan menggunakan HTTP request berbasis REST API.

5) Alur Data Sistem

Alur untuk pengolahan data dalam sistem ini akan dijelaskan sebagai berikut :

- Instructor/User login ke dalam website melalui frontend yang telah terintegrasi dengan sistem autentikasi Strapi.
- Instructor/User menginput data melalui dashboard frontend pada website e – learning.
- Frontend mengirim data input ke backend Strapi melalui API dengan autentikasi (JWT) untuk validasi.
- Strapi menyimpan data kursus ke dalam database dan menyediakan endpoint API untuk konten tersebut.
- Frontend (user atau murid) kemudian mengambil data kursus dari endpoint API untuk konten tersebut.



Gbr. 3 Use case diagram website e-learning

Pada Gbr. 3 ditampilkan *use case diagram* untuk sistem e-learning yang dirancang. Diagram ini terdiri dari satu aktor, yaitu Admin, yang berperan sebagai pengguna dengan hak akses penuh terhadap pengelolaan konten pembelajaran. Admin memiliki kemampuan untuk melakukan enam *use case* utama, yaitu:

- 1) Membuat akun baru - Proses registrasi akun pengguna baru pada sistem e-learning, yang menyimpan informasi akun di basis data.
- 2) Login – Proses autentikasi yang memverifikasi kredensial admin agar dapat mengakses fitur manajemen konten.
- 3) Menambahkan Course Baru – Admin dapat membuat dan menyimpan kursus baru, termasuk data terkait seperti judul, deskripsi, kategori, dan media pendukung.
- 4) Menambahkan Quiz Baru – Admin dapat membuat kuis yang terkait dengan kursus tertentu sebagai bagian dari evaluasi pembelajaran.
- 5) Menambahkan Question Baru – Admin dapat menambahkan pertanyaan yang akan digunakan dalam kuis, baik dalam format pilihan ganda maupun esai.
- 6) Menambahkan Answer Baru – Admin dapat menambahkan jawaban yang terkait dengan setiap pertanyaan kuis untuk menentukan kunci jawaban yang benar.
- 7) Mengakses Course Baru – Admin dapat melihat atau melakukan pengujian terhadap kursus yang baru dibuat untuk memastikan materi telah sesuai sebelum dipublikasikan.

Use case diagram ini bertujuan untuk memodelkan interaksi antara aktor dan sistem, sehingga memudahkan identifikasi kebutuhan fungsional pada tahap analisis dan perancangan. Setiap *use case* mewakili fungsi yang harus diimplementasikan dalam sistem untuk mendukung proses pengelolaan konten e-learning secara efisien.

D. Tahap Pengembangan Website

Tahap pengembangan merupakan proses implementasi dari rancangan yang telah disusun pada tahap sebelumnya. Pada tahap ini, peneliti melaksanakan eksekusi rancangan secara bertahap, dimulai dari konfigurasi backend menggunakan Strapi, konfigurasi API menggunakan pendekatan RESTful API, pengembangan antarmuka frontend dengan React.js, hingga proses hosting sistem, di mana backend dihosting menggunakan Railway dan frontend menggunakan Vercel.

Tahap ini berfokus pada realisasi system e-learning berdasarkan hasil identifikasi kebutuhan pengguna dan desain sistem yang telah dirancang sebelumnya. Tujuannya adalah untuk menghasilkan sistem e-learning yang sesuai dengan kebutuhan pengguna dan siap untuk digunakan dalam lingkungan operasional.

E. Tahap Pengujian Website

Pada tahap pengujian, sistem e-learning yang telah dikembangkan akan diuji berdasarkan fitur-fitur fungsional

dan non-fungsional yang telah dirancang sebelumnya. Pengujian ini juga mencakup evaluasi performa sistem menggunakan alat bantu seperti Lighthouse dan Google PageSpeed Insights.

Penilaian performa berdasarkan Lighthouse dibagi ke dalam tiga kategori: skor 0 – 49 (merah) menunjukkan performa buruk, skor 50 – 89 (oranye) menunjukkan performa perlu peningkatan, dan skor 90 – 100 (hijau) menunjukkan performa baik.

Selain pengujian performa, setiap halaman dalam sistem e-learning akan diuji untuk memastikan stabilitas saat menampilkan konten dengan ukuran besar dan beragam format, seperti gambar, video, dan dokumen materi pembelajaran. Tujuan dari pengujian ini adalah untuk menjamin bahwa seluruh komponen sistem bekerja dengan optimal dan responsif dalam berbagai kondisi penggunaan.

F. Tahap Analisis Hasil

Tahap analisis hasil akan membahas hasil pengujian website. Pengujian website akan menggunakan tools Google Pagespeed Insight. Peneliti akan memantau performa website pada e-learning untuk melihat kelebihan penggunaan arsitektur Headless Content Management System. Pada tahap analisis hasil ini fitur – fitur yang ada pada e-learning juga akan di pantau fungsinya untuk mengetahui apakah ada error pada sistem dan fitur di website e-learning yang telah dikembangkan.

G. Tahap Kesimpulan

Tahap kesimpulan akan menjelaskan semua hasil pengujian website berdasarkan tahap analisis. Serta untuk mengetahui apakah tujuan awal terpenuhi yaitu pengembangan aplikasi e-learning berbasis Headless Content Management System. Kesimpulan juga mencakup saran untuk mengembangkan fitur-fitur baru dan hasil penelitian ini diharapkan juga bisa menjadi referensi untuk pengembangan website menggunakan arsitektur Headless Content Management System di masa yang akan datang.

III. HASIL DAN PEMBAHASAN

Proses penelitian ini mengikuti tahapan yang telah dijabarkan pada Bab II, yaitu identifikasi kebutuhan pengguna, perancangan, pengembangan, pengujian, analisis hasil, dan kesimpulan yang akan disampaikan pada bab berikutnya.

A. Identifikasi Kebutuhan Pengguna

Pada tahap awal, peneliti mengidentifikasi dan merumuskan fitur-fitur utama yang akan dikembangkan pada platform e-learning. Penentuan fitur ini dilakukan melalui diskusi dengan para ahli di bidang terkait serta studi literatur dari penelitian-penelitian terdahulu. Berdasarkan hasil analisis tersebut, ditetapkan sejumlah fitur utama yang dikelompokkan ke dalam tiga modul utama, yaitu:

- 1) *Modul Pengguna, yang terdiri atas :*
 - a.Registrasi akun
 - b.Autentikasi masuk (Log in)
 - c. Dashboard pengguna
 - d. Profil pengguna
 - e. Pengeditan profil
- 2) *Modul Instruktur, yang terdiri atas :*
 - a.Penambahan materi pembelajaran dalam bentuk video dan teks
 - b. Pembuatan quiz
 - c. Melihat jumlah peserta didik
- 3) *Modul Siswa, yang meliputi :*
 - a.Pendaftaran pada kursus (enrollment)
 - b. Akses terhadap materi pembelajaran (teks dan video)
 - c.Pengumpulan atau penyelesaian quiz

B. Tahap perancangan Website

Pada tahap perancangan, peneliti menetapkan kebutuhan sistem baik dari sisi *backend* maupun *frontend* berdasarkan hasil identifikasi sebelumnya. Pendekatan yang digunakan adalah *Headless Content Management System* guna memungkinkan pengembangan *backend* dan *frontend* secara terpisah dan fleksibel. Untuk *backend*, digunakan Strapi CMS yang menyediakan antarmuka *content-type builder* serta secara otomatis menghasilkan REST API. Sementara itu, *frontend* dikembangkan menggunakan React.js untuk menampilkan data dari *backend* melalui HTTP request. Setelah proses pengembangan selesai, sistem akan di-*deploy* menggunakan Railway untuk *backend* dan Vercel untuk *frontend*.

C. Implementasi Backend

Proses implementasi website e-learning ini diawali dengan pembuatan content type builder di strapi sesuai dengan rancangan yang telah dibuat sehingga semua data – data dan koneksi yang diperlukan untuk disimpan di database dapat terpenuhi.

Berikut ini adalah Content Type Builder dari website e-learning yang akan dikembangkan.

1) Modul Pengguna

Berikut ini adalah struktur content type builder untuk user yang berfungsi untuk autentikasi pengguna ketika register dan login. Content type builder User yang telah dikembangkan melalui strapi dapat dilihat pada Tabel 1 berikut ini.

Tabel 1

Content Type Builder User

No	Nama	Tipe Data
1	Username	Text
2	Email	Email

No	Nama	Type Data
3	Provider	Text
4	Password	Password
5	resetPasswordToken	Text
6	confirmationToken	Text
7	Confirmed	Boolean
8	Blocked	Boolean
9	Role	Relation
10	Profile Picture	Media
11	Course	Relation

2) Modul Course

Berikut ini adalah struktur content type builder untuk modul course yang nantinya akan menyimpan seluruh data course yang telah ditambahkan instructor. Content type builder Course yang telah dikembangkan melalui strapi dapat dilihat pada Tabel 2 berikut ini.

Tabel 2

Content Type Builder Course

No	Nama	Type Data
1	Title	Text
2	Slug	UID
3	Description	Rich Text (Blocks)
4	Category	Text
5	Rating	Number
6	Thumbnail	Media
7	studentsCount	Number
8	Instructor	Relation
9	Price	Number
10	Language	Text
11	Duration	Text
12	Level	Enumeration
13	Certification	Boolean
14	Video	Media
15	Quizzes	Relation

3) Modul Instructor

Berikut ini adalah struktur content type builder untuk instructor yang nantinya akan berelasi ke user dan course. Instructor berfungsi untuk menyimpan data diri instructor serta relasinya dengan course akan berfungsi untuk identitas course yang telah dibuat. Content type builder Instructor yang telah dikembangkan melalui strapi dapat dilihat pada Tabel 3 berikut ini.

Tabel 3

Content Type Builder Instructor

No	Nama	Type Data
1	Name	Text
2	jobTitle	Text
3	Bio	Rich Text (Blocks)
4	Avatar	Media

5	Socials	Component (Reusable)
---	---------	----------------------

4) Modul Quiz

Berikut ini adalah modul quiz yang akan berfungsi untuk menyimpan data quiz baik itu soal maupun jawaban yang telah di submit oleh murid pada website e-learning. Content type builder Quiz yang telah dikembangkan melalui strapi dapat dilihat pada Tabel 4 berikut ini.

Tabel 4

Content Type Builder Quiz

No	Nama	Type Data
1	Title	Text
2	Description	Rich Text (Blocks)
3	Course	Relation
4	Questions	Relation
5	Quiz Submission	Relation

D. Implementasi Frontend

Implementasi bagian antarmuka akan dikembangkan menggunakan react.js berbasis typescript untuk mendukung tampilan yang bersih responsif.

Berikut ini adalah halaman yang berhasil diimplementasikan antara lain :

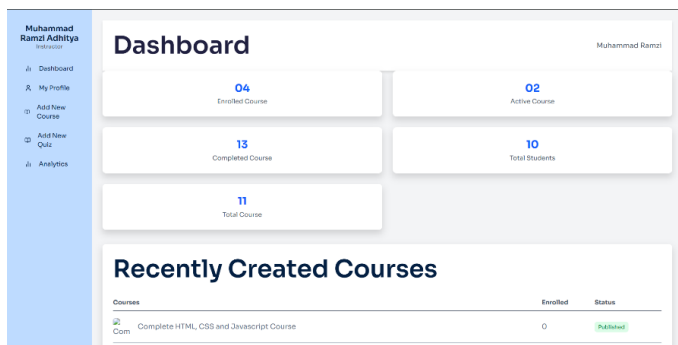
1) Halaman Register dan Login

Pada Gbr. 4 menampilkan kolom untuk pengguna mengisi username dan password.

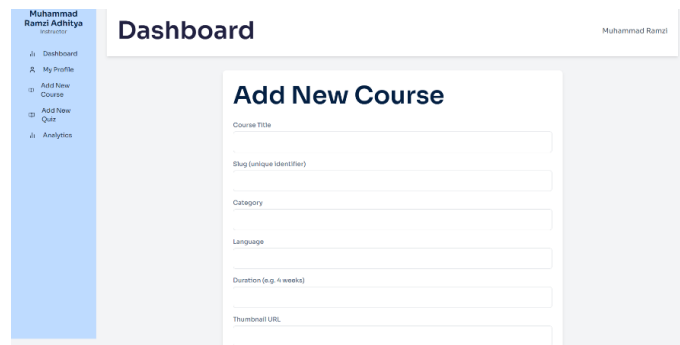
Gbr. 4 Page Register

2) Halaman Dashboard

Pada Gbr. 5 menampilkan halaman untuk melihat semua menu yang dapat diakses pengguna.



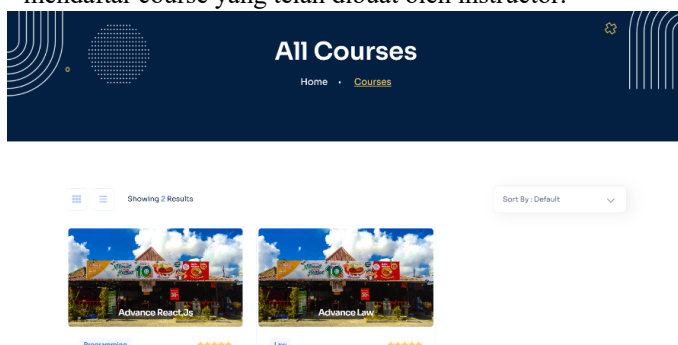
Gbr. 5 Page Dashboard



Gbr. 8 Page Tambah Course

3) Halaman Course

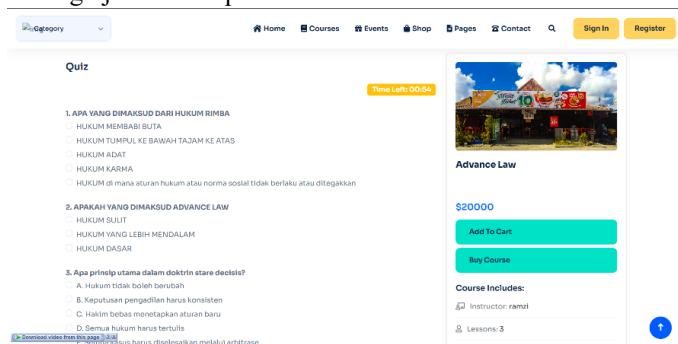
Pada Gbr. 6 menampilkan halaman untuk melihat dan mendaftar course yang telah dibuat oleh instructor.



Gbr. 6 Page Course

4) Halaman Quiz

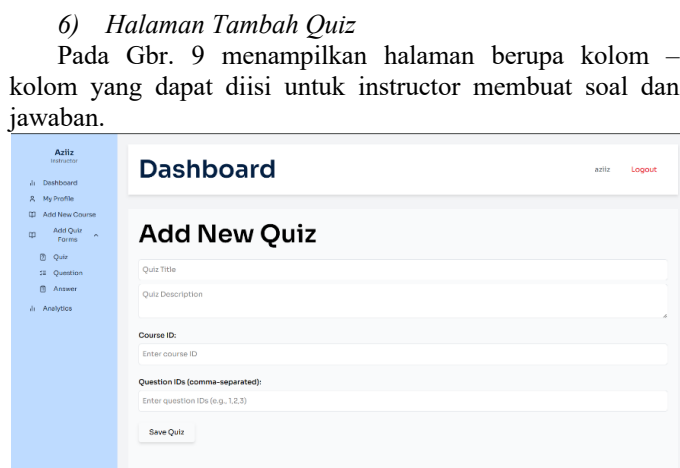
Pada Gbr. 7 menampilkan halaman quiz berupa soal, pilihan jawaban, timer, dan button submit untuk pengguna mengerjakan soal quiz.



Gbr. 7 Page Quiz

5) Halaman Tambah Course

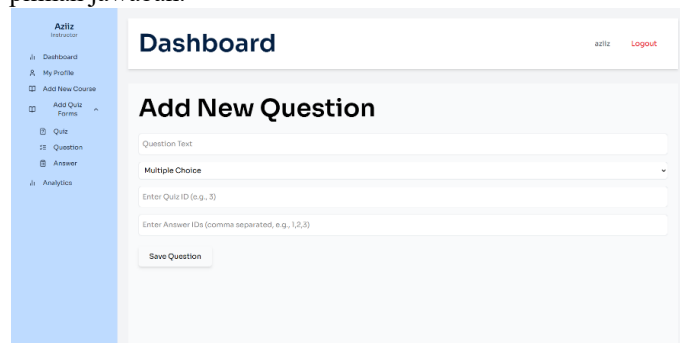
Pada Gbr. 7 menampilkan Halaman berupa kolom – kolom yang dapat diisi oleh instructor untuk menambahkan course baru.



Gbr. 9 Page Tambah Quiz

7) Halaman Tambah Question

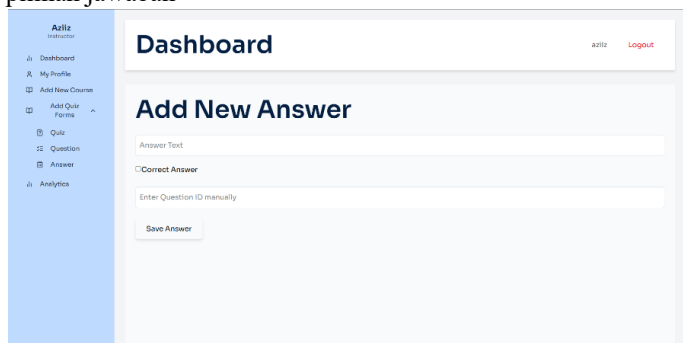
Pada Gbr. 10 menampilkan halaman berupa kolom – kolom yang dapat diisi untuk instructor membuat soal dan pilihan jawaban.



Gbr. 10 Page Tambah Question

8) Halaman Tambah Answer

Pada Gbr. 11 menampilkan halaman berupa kolom – kolom yang dapat diisi untuk instructor membuat soal dan pilihan jawaban



Gbr. 11 Page Tambah Answer

E. Pengujian

Setelah pengembangan backend dan frontend website e-learning selesai, tahap selanjutnya adalah pengujian. Pengujian dibagi menjadi dua jenis, yaitu uji fungsionalitas untuk memastikan seluruh fitur berjalan sesuai dengan spesifikasi, serta uji performa untuk mengevaluasi dampak penggunaan Headless CMS terhadap kinerja sistem secara keseluruhan.

1) Uji Fungsionalitas

Uji fungsionalitas dilakukan untuk memastikan semua fitur – fitur utama dapat berjalan dengan lancar sebagaimana mestinya. Pada pengujian website e-learning ini akan dilakukan beberapa skenario seperti mendaftarkan akun, login ke akun yang dibuat, menambahkan course dll.

Hasil dari uji fungsionalitas dapat dilihat pada Tabel 5 berikut ini.

Tabel 5
Uji Fungsionalitas

No	Fitur yang Diuji	Langkah Uji	Hasil yang Diharapkan	Status
1	Register Akun	Klik tombol register > isi form register > klik tombol submit	Pesan “Register Successfully” muncul dan pendaftaran akun berhasil	✓
2	Login Akun	Klik tombol sign in > isi form sign in	Pengguna berhasil masuk dan diarahkan ke dashboard	✓
3	Menambahkan Course Baru	Klik tombol	Pesan “Course	✓

No	Fitur yang Diuji	Langkah Uji	Hasil yang Diharapkan	Status
		add new course > isi form untuk course baru	Successfully Added” muncul dan diarahkan ke halaman dashboard	
4	Menambahkan Question Baru	Klik tombol add new question > isi form untuk question baru	Pesan “Question Successfully Added” muncul dan diarahkan ke halaman dashboard	✓
5	Menambahkan Answer Baru	Klik tombol add new answer > isi form untuk answer baru	Pesan “Answer Successfully Added” muncul dan diarahkan ke halaman dashboard	✓
6	Menambahkan Quiz Baru	Klik tombol add new quiz > isi form untuk quiz baru	Pesan “Quiz Successfully Added” muncul dan diarahkan ke halaman dashboard	✓
7	Akses Course	Klik tombol Course	Daftar Course yang telah berhasil dibuat akan tampil	✓

2) Uji Performa

Uji performa dilakukan menggunakan alat bantu Lighthouse untuk mengevaluasi kinerja website serta dampak penggunaan *Headless CMS* terhadap performa sistem. Pengujian dilakukan dengan beberapa skenario berdasarkan ukuran file *course*, di mana setiap skenario diuji sebanyak lima kali. Fokus utama pengujian ini adalah pada metrik performa seperti *First Contentful Paint* (FCP), *Total Blocking Time* (TBT), *Cumulative Layout Shift* (CLS), dan *Speed Index* (SI).

Berdasarkan hasil pengujian masing – masing course didapatkan nilai rata – rata untuk masing – masing metrik pengujian dan rata – rata performa dapat dilihat pada Tabel 6 berikut ini

Tabel 6
Rekapitulasi pengujian performa

No	Course	Rata – rata FCP	Rata – rata TBT	Rata – rata LCP	Rata – rata SI	Rata – rata Performa
----	--------	-----------------	-----------------	-----------------	----------------	----------------------

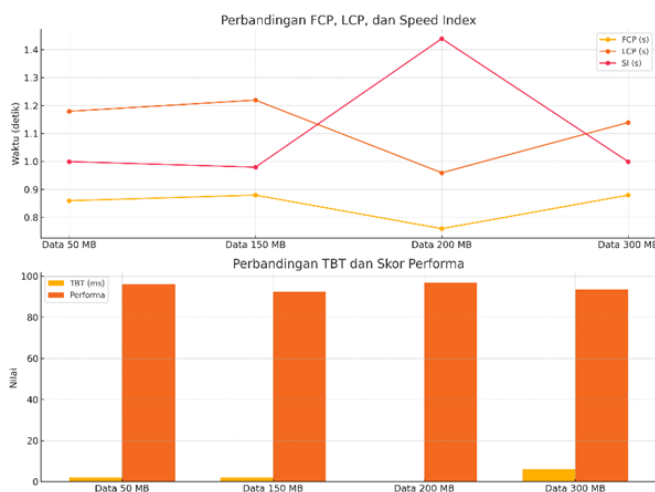
1	Data 50 MB	0,86 s	2 ms	1,18 s	1 s	96,2
2	Data 150 MB	0,88 s	2 ms	1,22 s	0,98 s	92,4
3	Data 200 MB	0,76 s	0 ms	0,96 s	1,44 s	96,8
4	Data 300 MB	0,88 s	6 ms	1,14 s	1 s	93,6

3) Analisis Hasil

Setelah pengujian selesai akan dilakukan analisis hasil. analisis hasil ini bertujuan untuk mengetahui pengaruh penggunaan Headless CMS terhadap performa website e-learning. hasil pengujian menunjukkan skor rata-rata performa yang sangat baik. Berdasarkan hasil pengujian dengan empat course yang memiliki beban data yang bervariasi mulai dari 50 MB sampai dengan yang terbesar 300 MB, terdapat hasil yang stabil dan sangat baik untuk sistem website e-learning. Masing-masing metrik seperti FCP, LCP, TBT, dan Speed Index menunjukkan hasil yang konsisten dengan waktu render dibawah 1,5 detik dengan skor performa selalu di atas 90.

Chart perbandingan pengujian performa dari masing – masing beban data dapat dilihat pada Gbr. 11 berikut ini.

Visualisasi Hasil Pengujian Course Berdasarkan Ukuran Data



Gbr. 12 Chart Pengujian Performa

IV. KESIMPULAN

Berdasarkan hasil pengujian performa terhadap halaman kursus pada sistem e-learning berbasis *Headless CMS*, diperoleh skor rata-rata performa yang tinggi, yaitu 96,2 pada dataset 50 MB, 92,4 (150 MB), 96,8 (200 MB), dan 93,6 (300 MB). Pengujian dilakukan lima kali untuk masing-masing ukuran data guna menjamin konsistensi hasil.

Metrik utama menunjukkan kinerja optimal, dengan nilai *First Contentful Paint* (FCP) berkisar antara 0,76–0,88 detik, *Largest Contentful Paint* (LCP) 0,96–1,22 detik, *Total Blocking Time* (TBT) 0–6 ms, dan *Speed Index* (SI) 0,98–1,44 detik.

Hasil ini menegaskan bahwa arsitektur *Headless CMS* mampu memberikan performa sistem yang lebih responsif dan efisien.

REFERENSI

- [1] I. Rizqi Sabila, I. Ardiansah, A. Yohari, I. R. Sabila, T. I. Pertanian, and U. Padjadjaran, "Optimalisasi E-Learning dengan Analisis Kelayakan Learning Management System TutorLMS untuk Startup Education Technology".
- [2] B. Gezani, "Challenges of Online Learning in the Comprehensive Open Distance and Elearning Context – A Case Study of the University of South Africa," *E-Journal of Humanities, Arts and Social Sciences*, pp. 1265–1274, Jul. 2024, doi: 10.38159/ehass.20245716.
- [3] E. Prakarsa Mandyarta and E. Y. Puspaningrum, "ANALISA PENGGUNAAN CONTENT MANAGEMENT SYSTEM (CMS) DALAM PEMBUATAN WEBSITE SEKOLAH UNTUK MENINGKATKAN KOMPETENSI GURU." [Online]. Available: <http://jiimat.upnjatim.ac.id/>
- [4] N. Pinnis, "MODERN WEBSITE DEVELOPMENT WITH STRAPI AND NEXT.JS," 2022.
- [5] L. Pyykölä and E. Petróczi, "Design and development of a web application for Vihdin-Nummela Kylähistoria Ry with WordPress as a Headless CMS and React," 2020.
- [6] P. Danielsson, T. Postema, and H. Munir, "Railway-based innovative platform for web-based deployment in product development at axis," *IEEE Access*, vol. 9, pp. 10805–10819, 2021, doi: 10.1109/ACCESS.2021.3050255.
- [7] R. Bangun and W. Kesehatan, "Yanuar Faturahman, Sigit Sugiyanto."
- [8] M. Lutfi Maftuh, "IMPLEMENTASI NEXT.JS DAN HEADLESS CMS UNTUK PENINGKATAN KUALITAS WEBSITE MI AL ISLAM PARE."
- [9] <https://www.aceinfo.com/blog/headless-vs-traditional-cms>
- [10] Y. Pandu Dewanata, "Analysis and Design of Headless CMS and GraphQL in Back-End Development," vol. 16, no. 2, 2023.