

LAPORAN PKL

Implementasi Teknologi Front End dalam Proyek Pengembangan Website



Vincent Dwi Hartanto

312110017

**UNIVERSITAS
MA CHUNG**

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI DAN DESAIN

UNIVERSITAS MA CHUNG

MALANG

2024

**LEMBAR PENGESAHAN
PRAKTIK KERJA LAPANGAN**

**Implementasi Teknologi Front End dalam Proyek
Pengembangan Website**

Oleh:

Vincent Dwi Hartanto

NIM. 312110017

dari:

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI DAN DESAIN
UNIVERSITAS MA CHUNG

Dosen Pembimbing,



Paulus Lucky Tirma Irawan, S.Kom., MT.

NIP. 20100005

Dekan Fakultas Sains dan Teknologi



Dr. Eng. Romy Budhi Widodo

NIP. 20070035

KATA PENGANTAR

Puji syukur dipanjatkan kehadiran Tuhan Yang Maha Esa, yang atas restu-Nya sehingga tugas dan laporan Praktik Kerja Lapangan (PKL) ini dapat diselesaikan dengan baik. Laporan ini berisikan hasil dari PKL yang telah dilaksanakan di Muse Akademi selama kurang lebih enam bulan. Pada kesempatan kali ini penulis mengucapkan terima kasih kepada seluruh pihak-pihak yang telah membantu selama proses pelaksanaan PKL maupun dalam penyusunan laporannya, di antaranya:

1. Bapak Dr. Eng. Muhammad Ashar, S.T., M.T, selaku Pemilik dari Muse Akademi yang telah menyediakan tempat untuk kegiatan magang,
2. Bapak Dr. Eng. Romy Budhi Widodo selaku Dekan dari Fakultas Sains dan Teknologi Universitas,
3. Bapak Paulus Lucky Tirma Irawan, S.Kom., MT. selaku Dosen pembimbing magang,
4. Kedua orang tua terkasih, yang telah memberikan dukungan dan semangat selama menjalani kegiatan magang,
5. Serta teman-teman yang telah memberikan dukungan selama proses penyelesaian kegiatan magang ini.

Laporan ini disusun berdasarkan hasil kegiatan magang yang dilaksanakan di Muse Akademi selama kurang lebih enam bulan, mulai dari tanggal 01-02-2024 hingga tanggal 01-08-2024. Kegiatan magang ini merupakan salah satu mata kuliah yang wajib ditempuh mahasiswa Teknik Informatika Universitas Ma Chung sebagai prasyarat kelulusan.

Malang, 1 Agustus 2024

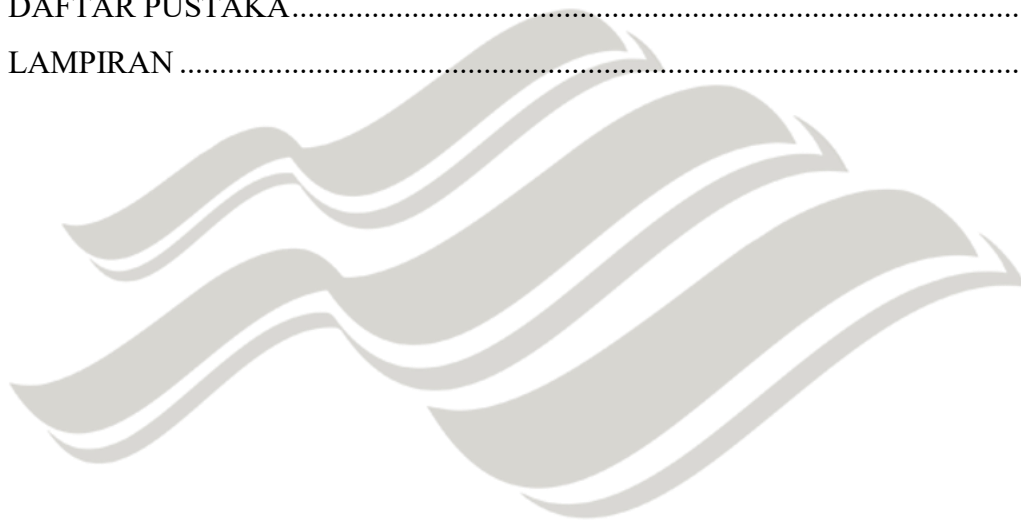


Vincent Dwi Hartanto

DAFTAR ISI

KATA PENGANTAR.....	1
DAFTAR ISI	2
DAFTAR GAMBAR.....	4
BAB 1 PENDAHULUAN.....	6
1.1 Latar Belakang	6
1.2 Identifikasi Masalah	7
1.3 Batasan Masalah.....	7
1.4 Rumusan Masalah	7
1.5 Tujuan	7
1.6 Manfaat	7
BAB 2 GAMBARAN UMUM PERUSAHAAN	9
2.1 Deskripsi Singkat Perusahaan	9
2.2 Visi dan Misi	9
2.3 Struktur Organisasi.....	10
2.4 Produk dan Layanan.....	10
2.5 Divisi Tempat Pelaksanaan Magang.....	11
BAB 3 TINJAUAN PUSTAKA.....	13
3.1 HTML	13
3.2 CSS.....	13
3.2 Javascript.....	14
3.3 Bootstrap	15
3.4 Tailwind	16
3.5 Django	16
3.6 Three.js.....	17
3.7 Swiper.js.....	18
3.8 Owl Carousel.....	18
3.9 Github.....	19
3.10 Anaconda Navigator.....	19
3.11 Visual Studio Code	20
BAB 4 HASIL DAN PEMBAHASAN	21

4.1 Proses Pengerjaan.....	21
4.2 Analisa Kebutuhan	21
4.3 Desain.....	26
4.4 Pengembangan.....	45
4.5 Implementasi dan Dokumentasi	69
BAB 5 PENUTUP.....	71
5.1 Kesimpulan.....	71
5.2 Saran.....	72
DAFTAR PUSTAKA.....	73
LAMPIRAN	75



UNIVERSITAS
MA CHUNG

DAFTAR GAMBAR

Gambar 4. 1 Persiapan menggunakan <i>Anaconda Navigator</i>	21
Gambar 4. 2 <i>Github</i> Muse Akademi	22
Gambar 4. 3 <i>Github</i> Brotani	23
Gambar 4. 4 <i>Github</i> Museverse.....	25
Gambar 4. 5 <i>Github</i> FLD	25
Gambar 4. 6 Desain laman dashboard classroom.....	26
Gambar 4. 7 Desain laman student evaluation	27
Gambar 4. 8 Desain laman classroom muse.....	27
Gambar 4. 9 Penggantian logo agensi Muse Akademi.....	28
Gambar 4. 10 Penambahan fungsi pencarian	28
Gambar 4. 11 Penggantian logo layanan Muse Akademi	28
Gambar 4. 12 Penggantian <i>cover landing page</i> Muse Akademi.....	29
Gambar 4. 13 Area kursus <i>landing page</i> Muse Akademi	29
Gambar 4. 14 Agensi <i>landing page</i> Muse Akademi	30
Gambar 4. 15 Paket kursus <i>landing page</i> Muse Akademi	30
Gambar 4. 16 Fitur <i>landing page</i> Muse Akademi.....	31
Gambar 4. 17 Sesi berita <i>landing page</i> Muse Akademi.....	31
Gambar 4. 18 Sesi review pengguna <i>landing page</i> Muse Akademi	32
Gambar 4. 19 <i>Call to action landing page</i> Muse Akademi.....	32
Gambar 4. 20 <i>Footer landing page</i> Muse Akademi.....	33
Gambar 4. 21 Halaman berita <i>landing page</i> Muse Akademi	33
Gambar 4. 22 Halaman <i>workshop landing page</i> Muse Akademi.....	34
Gambar 4. 23 <i>Pdf viewer</i>	35
Gambar 4. 24 Sesi 1 <i>website</i> Brotani	36
Gambar 4. 25 Sesi 2 <i>website</i> Brotani	36
Gambar 4. 26 Sesi 3 <i>website</i> Brotani	37
Gambar 4. 27 Sesi 4 <i>website</i> Brotani	38
Gambar 4. 28 Sesi 5 <i>website</i> Brotani	39
Gambar 4. 29 <i>Cover Fun Learning Digital</i>	40
Gambar 4. 30 Desain responsive Fun Learning Digital	40
Gambar 4. 31 Sesi 1 <i>index.html</i> Museverse	41

Gambar 4. 32 Sesi 2 index.html Museverse	42
Gambar 4. 33 Halaman product.html Museverse	43
Gambar 4. 34 Halaman about.html Museverse	44
Gambar 4. 35 Halaman about.html Museverse	45
Gambar 4. 36 Owl Carousel 1 Muse Akademi	46
Gambar 4. 37 Owl Carousel 2 Muse Akademi	47
Gambar 4. 38 Owl Carousel 3 Muse Akademi	48
Gambar 4. 39 Owl Carousel 4 Muse Akademi	49
Gambar 4. 40 Penggunaan pdf.js Muse Akademi	50
Gambar 4. 41 Pdf.js fitur <i>fullscreen</i>	50
Gambar 4. 42 Tabel <i>collapsible</i> saat dilipat	51
Gambar 4. 43 Tabel <i>collapsible</i> saat dibuka	52
Gambar 4. 44 Fungsi menampilkan deskripsi saat ditekan	53
Gambar 4. 45 Animasi hover tulisan	54
Gambar 4. 46 Animasi hover tombol	55
Gambar 4. 47 Animasi sebelum <i>scroll</i> (atas) dan sesudah <i>scroll</i> (bawah)	56
Gambar 4. 48 Animasi halaman dimuat	57
Gambar 4. 49 Transisi warna sebelum <i>scroll</i> (atas) dan sesudah <i>scroll</i> (bawah) .	58
Gambar 4. 50 Desain responsif <i>website</i> Brotani	59
Gambar 4. 51 <i>Collapsible navbar website</i> Brotani	60
Gambar 4. 52 Desain responsif <i>website</i> Fun Learning Digital	61
Gambar 4. 53 <i>Collapsible navbar website</i> Fun Learning Digital	62
Gambar 4. 54 Tampilan objek 3 dimensi Museverse	63
Gambar 4. 55 Interaksi geser objek 3 dimensi Museverse	65
Gambar 4. 56 Interaksi rotasi objek 3 dimensi Museverse	65
Gambar 4. 57 Interaksi perbesar/perkecil objek 3 dimensi Museverse	66
Gambar 4. 58 <i>Plugin Swiper.js</i> Museverse	67
Gambar 4. 59 Sistem pergantian tab Museverse	68
Gambar 4. 60 Dokumentasi <i>hosting</i> Muse Akademi	69
Gambar 4. 61 Dokumentasi <i>hosting</i> Museverse	70

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Dalam era digital yang terus berkembang, teknologi *front-end* memainkan peran penting dalam pengembangan proyek *website*. Teknologi ini melibatkan penggunaan HTML, CSS, dan *Javascript* untuk menciptakan antarmuka pengguna yang interaktif dan responsif. Seiring dengan meningkatnya ekspektasi pengguna terhadap kecepatan akses, kemudahan navigasi, dan estetika visual, penguasaan teknologi *front-end* menjadi kunci bagi para pengembang untuk menghasilkan *website* yang tidak hanya fungsional tetapi juga menarik dan intuitif. Penggunaan teknologi *front-end* yang efektif dapat meningkatkan kepuasan pengguna, memperpanjang waktu kunjungan, dan akhirnya mendukung tujuan bisnis melalui konversi yang lebih tinggi.

Perkembangan teknologi informasi yang pesat menuntut mahasiswa untuk selalu up-to-date dengan inovasi dan tren terbaru. Oleh karena itu PKL menjadi sarana yang efektif bagi mahasiswa untuk berinteraksi langsung dengan industri teknologi informasi, memahami dinamika pekerjaan, dan mengetahui kebutuhan pasar tenaga kerja. Selain itu, melalui PKL, mahasiswa dapat membangun jaringan profesional yang berguna untuk karir masa depan.

Dalam laporan ini, penulis akan memaparkan pengalaman selama menjalani PKL di Muse Akademi, yang berlangsung selama enam bulan. Laporan ini mencakup deskripsi tentang perusahaan, tugas dan tanggung jawab yang diemban, serta analisis mengenai kontribusi PKL terhadap pengembangan kompetensi teknis dan soft skills penulis. Selain itu, penulis juga akan menguraikan beberapa tantangan yang dihadapi dan solusi yang diterapkan selama PKL berlangsung.

Melalui penyusunan laporan ini, penulis berharap dapat memberikan gambaran yang jelas mengenai kegiatan PKL yang telah dilaksanakan, serta kontribusinya terhadap peningkatan kualitas pendidikan di bidang Teknik Informatika. Semoga laporan ini bermanfaat bagi pihak-pihak yang berkepentingan

dan menjadi referensi bagi mahasiswa lain yang akan melaksanakan PKL di masa mendatang.

1.2 Identifikasi Masalah

Selama menjalankan PKL di Muse Akademi, penulis diberikan tugas untuk mengembangkan dan mengoptimalkan berbagai *website*. Salah satu masalah yang diidentifikasi adalah bagaimana meningkatkan efisiensi dan kualitas *website* yang dikembangkan untuk berbagai klien, agar lebih responsif dan *user-friendly*. Selain itu, penulis juga ditugaskan untuk merancang dan membangun *website* baru, khususnya di bagian *front-end*, bukan hanya mengembangkan yang sudah ada. Untuk mengatasi masalah tersebut, Muse Akademi bekerja sama dengan berbagai tim dan klien untuk merancang dan mengimplementasikan solusi *front-end* yang inovatif dan efektif.

1.3 Batasan Masalah

Penulis fokus pada pengembangan dan pengoptimalan *website* untuk berbagai klien dengan kebutuhan yang berbeda. Batasan ini mencakup pembuatan antarmuka pengguna yang responsif dan *user-friendly* sesuai dengan spesifikasi klien.

1.4 Rumusan Masalah

Bagaimana cara memastikan bahwa *website* yang dikembangkan dapat memenuhi kebutuhan spesifik klien dan tetap responsif serta *user-friendly*

1.5 Tujuan

Untuk mengembangkan *website* yang responsif dan *user-friendly* sesuai dengan kebutuhan klien.

1.6 Manfaat

Manfaat yang diperoleh dalam pelaksanaan PKL ini adalah sebagai berikut:

1. Bagi Muse Akademi, manfaat yang didapatkan adalah memiliki berbagai macam *website* yang responsif dan *user-friendly*.

2. Bagi penulis, manfaat yang didapatkan adalah kesempatan terjun langsung ke dunia kerja untuk memperoleh ilmu serta mengasah kemampuan dalam mengimplementasikan teknologi *front-end* dalam pengembangan *website*.
3. Untuk Universitas khususnya prodi Teknik Informatika memberi manfaat yang signifikan dengan adanya kerjasama dengan Muse Akademi. Kegiatan magang ini juga bertujuan untuk mempersiapkan mahasiswa agar memiliki pengalaman kerja yang kompeten dan relevan dengan dunia industri setelah lulus.



UNIVERSITAS
MA CHUNG

BAB 2 GAMBARAN UMUM PERUSAHAAN

2.1 Deskripsi Singkat Perusahaan

Muse akademi merupakan platform kursus berbasis *live streaming* menggunakan kurikulum adaptif dengan konsep DMSO (duplikasi, modifikasi, substitusi dan omisi). Setiap orang yang mengikuti kursus premium di Muse Akademi akan mendapatkan sertifikasi keahlian yang sesuai dengan standar kurikulum. Konsep pembelajaran di Muse Akademi adalah mengikuti tatap muka secara *online (live streaming)* yang diatur dalam beberapa sesi.

Pada setiap sesi pembelajaran terdapat fitur forum diskusi dan modul pembelajaran. Setelah seluruh sesi pembelajaran selesai, siswa dapat melangsungkan pengerjaan tugas yang tersedia pada kursus tersebut waktu pengerjaan tugas habis. Kemudian, mentor akan memberikan nilai pada tugas yang telah dikerjakan setelah di submit ataupun setelah waktu pengerjaan tugas habis. Setelah siswa mendapatkan nilai dan waktu kursus telah berakhir, siswa akan mendapatkan sertifikat keahlian yang disertai dengan nilai yang diperoleh pada saat kursus.

2.2 Visi dan Misi

Muse Akademi memiliki visi sebagai berikut :

1. Pengembangan smart teknologi berbasis aplikasi inovatif dan imersif dalam mendukung society 5.0 berjaya.

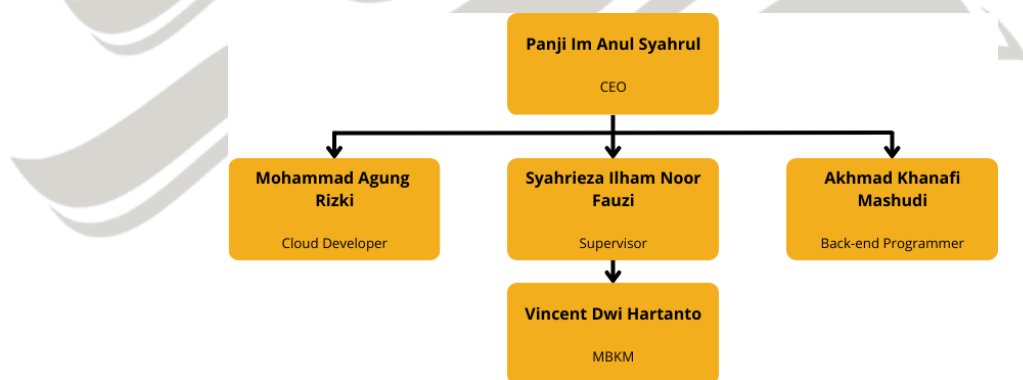
Muse Akademi juga memiliki misi sebagai berikut :

1. Menciptakan karya inovasi untuk bangsa melalui teknologi ubiquitous computing.
2. Meningkatkan life skill masyarakat Indonesia pada rekayasa teknologi cerdas (metaverse, game, AR/VR) dan mobile/web apps.
3. Menunjang pengembangan smart autonomous innovation di lingkungan kecerdasan buatan (AI) di bidang edukasi, pertanian, pariwisata, kesehatan dan umkm.

4. Membangun riset inovasi berdaya saing tinggi dalam era milenial.

2.3 Struktur Organisasi

Muse Akademi memiliki dua tim utama, yaitu tim yang bergerak di bidang objek 3D dan tim yang bergerak di bidang *Smart Coding*. Penulis menjadi bagian dari tim Smart Coding yang bertugas sebagai pengembang *front-end* website. Dalam perannya, penulis bertanggung jawab untuk membangun antarmuka pengguna yang intuitif dan responsif, serta memastikan setiap proyek yang dikerjakan memenuhi standar kualitas yang tinggi dan sesuai dengan kebutuhan pengguna. Penulis juga berkolaborasi dengan anggota tim lainnya untuk mencapai tujuan bersama dalam pengembangan website di Muse Akademi. Berikut detail struktur organisasi Muse Akademi:



Gambar 2. 1 Struktur Organisasi Muse Akademi

2.4 Produk dan Layanan

Muse Akademi menyediakan *platform* LMS (*Learning Management System*) berbasis *smart web*. Platform produk ini menggunakan teknologi AR/VR dengan *design* UI secara *custom* pada lingkungan pembelajaran *classroom* dan VR *room* yang berinteraksi dengan *ubiquitous smart sensor*. Berikut beberapa produk dan layanan yang Muse Akademi berikan:

1. *Gamification*: Interaksi dalam permainan melibatkan keterlibatan dan komunikasi dinamis antara pemain dan lingkungan permainan
2. *Augmented Reality* (AR): Pengalaman digital yang menggabungkan elemen virtual dengan dunia nyata melalui teknologi visualisasi

3. *Virtual Reality* (VR): Pengalaman simulasi digital yang menghadirkan lingkungan dan interaksi yang tampak nyata melalui penggunaan teknologi visual dan sensor
4. *Mixed Reality* (MR): Penggabungan antara dunia nyata dan virtual yang memungkinkan pengguna untuk berinteraksi dengan objek digital di lingkungan fisik mereka
5. *Virtual Design Interface*: Menggabungkan elemen-elemen estetika dengan fungsi interaktif untuk menciptakan pengalaman visual yang menarik dan berdaya

Muse Akademi juga mengadakan berbagai kursus dan pelatihan yang dirancang untuk memenuhi kebutuhan pembelajaran di berbagai bidang. Dalam upaya untuk memberikan pengalaman edukasi yang komprehensif dan berkualitas, Muse Akademi bekerja sama dengan beberapa mitra edukasi terkemuka. Kolaborasi ini memastikan bahwa kurikulum yang ditawarkan selalu terkini dan relevan dengan perkembangan industri terbaru. Berikut beberapa bidang keahlian yang ditawarkan oleh Muse Akademi:

1. *Education & Social Humaniora*
2. *Financial & Marketing*
3. *IT & Technology*
4. *Agriculture & Smart Training*
5. *Medical & Healthcare*
6. *Metaverse Multi-Industry*
7. *Media & Art Creative Industry*
8. *Business*

2.5 Divisi Tempat Pelaksanaan Magang

Sebagai pengembang *front-end* di divisi *Smart Coding* Muse Akademi, penulis memiliki tanggung jawab yang luas dalam pengembangan dan peningkatan tampilan serta fungsionalitas situs *web*. Selain menggunakan teknologi dasar seperti HTML, CSS, dan *Javascript*, penulis juga memanfaatkan berbagai plugin berbasis *Javascript* untuk memperkaya interaktivitas dan dinamika situs. Penulis tidak

hanya bertanggung jawab untuk menciptakan elemen UI yang menarik dan fungsional, tetapi juga memastikan bahwa setiap situs *web* yang dikembangkan memiliki performa yang optimal, baik dari segi kecepatan akses maupun pengalaman pengguna. Dalam prosesnya, penulis bekerja erat dengan tim lain untuk memastikan bahwa setiap proyek yang dikerjakan sesuai dengan standar kualitas tinggi, serta mampu memenuhi kebutuhan dan ekspektasi pengguna secara efektif. Penulis juga terlibat dalam pengujian dan *debugging*, serta selalu mengikuti perkembangan teknologi terbaru untuk memastikan situs *web* yang dikembangkan tetap relevan dan *up-to-date*.



UNIVERSITAS
MA CHUNG

BAB 3 TINJAUAN PUSTAKA

3.1 HTML

HTML (*Hyper Text Markup Language*) merupakan bahasa markup yang digunakan untuk membangun struktur dasar sebuah halaman web (Summit, 2023). HTML memberikan struktur dasar untuk halaman web dengan menggunakan elemen-elemen yang dikenal sebagai tag. HTML bekerja sama dengan CSS untuk menangani presentasi visual dan *Javascript* untuk menangani interaktivitas, membuat halaman web menjadi lebih dinamis dan menarik. Berkas HTML memiliki ekstensi .html dan dapat diakses di seluruh *web browser* modern. Berkas tersebut akan ditampilkan sesuai dengan struktur dan gaya yang ditentukan di dalam berkas HTML tersebut. Berkas HTML juga dapat diakses melalui dua cara yaitu melalui komputer lokal dan melalui *URL*. Dalam mengakses melalui komputer lokal, berkas HTML cukup dibuka seperti biasanya. Dan jika melalui *URL*, berkas HTML dapat diunggah ke *server web* terlebih dahulu sehingga menghasilkan *domain* yang dapat diakses.

Untuk mengetikkan skrip HTML dapat menggunakan text editor seperti Notepad sebagai bentuk paling sederhana atau text editor khusus yang dapat mengenali setiap skrip HTML dan menampilkannya dengan warna yang berbeda sehingga mudah di baca, seperti *Notepad++*, *Sublime Text* dan masih banyak lagi aplikasi lain yang sejenisnya (Permatasari & Suhendi, 2020). Skrip HTML memberikan struktur dasar untuk halaman web dengan membagi dokumen menjadi dua bagian utama: *<head>*, yang berisi metadata dan informasi penting seperti judul dan pengaturan karakter, serta *<body>*, yang berisi konten yang terlihat oleh pengguna seperti teks, gambar, dan tautan.

3.2 CSS

CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen web yang berfungsi mengatur elemen HTML dengan berbagai properti yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan (Permatasari & Suhendi, 2020). Dengan CSS, pengembang dapat mengontrol berbagai aspek visual seperti warna,

font, margin, dan tata letak elemen di halaman. CSS juga mendukung desain responsif, memungkinkan halaman web menyesuaikan tampilan secara otomatis sesuai dengan ukuran layar perangkat, seperti smartphone, tablet, dan desktop. Fungsi penting CSS meliputi pemisahan konten dari presentasi, yang mempermudah pemeliharaan dan pembaruan desain, serta kemampuan untuk menerapkan gaya konsisten di seluruh situs web dengan menggunakan kelas dan ID.

CSS dapat diterapkan ke halaman web melalui tiga metode utama: internal, *inline*, dan eksternal. Internal CSS diletakkan dalam elemen `<style>` di bagian `<head>` dari dokumen HTML, memungkinkan gaya diterapkan hanya pada halaman tersebut. *Inline* CSS diterapkan langsung pada elemen HTML menggunakan atribut *style*, ideal untuk gaya spesifik yang hanya berlaku untuk satu elemen. Eksternal CSS, yang diletakkan dalam berkas terpisah dengan ekstensi `.css` dan dihubungkan melalui elemen `<link>`, memungkinkan penerapan gaya secara konsisten di seluruh situs *web*, memudahkan pemeliharaan dan pembaruan. Masing-masing metode memiliki fungsinya tersendiri dan sering digunakan bersama untuk efisiensi dan keteraturan dalam desain *web*.

3.2 Javascript

Javascript adalah bahasa pemrograman berbentuk kumpulan *script* yang berjalan pada suatu dokumen HTML (Mariko, 2019). Fungsi utama dari *Javascript* meliputi manipulasi elemen HTML, pengolahan data, dan penanganan event seperti klik dan input pengguna. *Javascript* bekerja di sisi klien, yang berarti kode *Javascript* dieksekusi langsung di web browser pengguna, memungkinkan perubahan pada tampilan dan perilaku halaman tanpa perlu memuat ulang halaman. Adapun karakteristik dari bahasa pemrograman *Javascript* adalah: (1) Bahasa pemrograman berjenis *high-level programming*; (2) Bersifat *client-side*; (3) Berorientasi pada objek, dan (4) Bersifat *loosely typed* (Mariko, 2019).

Berkas *Javascript* biasanya memiliki ekstensi `.js` dan berisi kode yang bisa digunakan untuk berbagai keperluan seperti validasi formulir, pembuatan efek

visual, atau interaksi dengan server. *Javascript* diakses dan diintegrasikan dalam halaman web melalui beberapa metode. Kode *Javascript* dapat disertakan langsung dalam dokumen HTML menggunakan elemen `<script>` di dalam bagian `<head>` atau `<body>`. Alternatif lainnya adalah dengan menghubungkan berkas *Javascript* eksternal melalui elemen `<script>` dengan atribut `src` yang merujuk ke berkas `.js` terpisah. Saat berkas `.js` eksternal dipanggil dari HTML, kode di dalamnya akan dieksekusi oleh *browser*, memungkinkan pengembang untuk menciptakan aplikasi *web* yang tidak hanya statis, tetapi juga dinamis dan responsif. Dengan memisahkan logika aplikasi dari markup HTML, pengembang dapat membangun situs web yang lebih modular, memungkinkan pemanfaatan kembali kode, dan meningkatkan performa serta pengalaman pengguna secara keseluruhan.

3.3 Bootstrap

Bootstrap adalah *front-end framework* yang bagus dan luar biasa yang mengedepankan tampilan untuk *mobile device* (*Handphone, smartphon* dan lain-lain.) guna mempercepat dan mempermudah pengembangan *website* (Suprayogi & Rahmanesa, 2019). Artinya, tampilan web yang dibuat dengan *framework bootstrap* akan menyesuaikan ukuran layar dari browser yang akan gunakan baik di desktop, tablet ataupun *mobile device* (Riasinir & Widyasari, 2019). *Bootstrap* merupakan *framework* berbasis HTML, CSS, dan *Javascript* yang menyediakan kumpulan alat dan komponen siap pakai untuk desain responsif dan pengembangan antarmuka pengguna yang konsisten.

Framework ini menyediakan berbagai elemen desain siap pakai seperti tombol, formulir, dan sistem *grid* untuk pembuatan layout yang menyesuaikan dengan berbagai ukuran layar. Kelebihan menggunakan *Bootstrap* termasuk kemudahan dalam penerapan desain responsif, konsistensi gaya, dan penghematan waktu pengembangan dengan mengurangi kebutuhan menulis kode dari awal. Untuk mengakses *Bootstrap*, dapat menambahkan file CSS dan *Javascript Bootstrap* ke proyek web melalui tautan CDN atau dengan mengunduh berkas dari situs resmi *Bootstrap* dan mengintegrasikannya ke dalam berkas HTML.

3.4 Tailwind

Tailwind merupakan sebuah *framework* CSS yang digunakan untuk menyediakan keperluan dasar membangun komponen tampilan website, seperti pengaturan margin, ukuran objek, posisi, warna, dan lain-lain (Arhandi, et al., 2022). *Tailwind* memungkinkan pengembang untuk membangun antarmuka pengguna dengan menggabungkan berbagai kelas *utility* secara langsung di markup HTML. Ini mempermudah kustomisasi desain dan memastikan konsistensi visual dengan mengurangi kebutuhan untuk menulis CSS tambahan.

Tailwind dapat diakses dengan menambahkan berkas CSS *Tailwind* ke proyek *web* melalui tautan CDN atau dengan menginstalnya menggunakan *npm/yarn* jika menggunakan *build tool* seperti *Webpack* atau *Vite*. Untuk menggunakan *Tailwind*, perlu mengkonfigurasi berkas konfigurasi *tailwind.config.js* jika menggunakan *build tool*, atau mengimpor *Tailwind* CSS secara langsung ke dalam proyek. *Tailwind* menawarkan kontrol desain yang lebih granular dan fleksibilitas tinggi karena berbasis *utility-first*, memungkinkan penyesuaian yang lebih mendalam dibandingkan dengan komponen siap pakai di *Bootstrap*. Dengan *Tailwind*, bisa membangun desain yang sangat khusus tanpa batasan komponen bawaan. Selain itu, *Tailwind* mendukung desain responsif dan kustomisasi lebih baik dengan konfigurasi yang fleksibel melalui berkas konfigurasi, sedangkan *Bootstrap* mungkin memerlukan *overriding* yang lebih banyak pada komponen default untuk mencapai desain yang diinginkan. *Tailwind* juga sering menghasilkan berkas CSS yang lebih kecil karena hanya memuat kelas yang digunakan di proyek, sedangkan *Bootstrap* mungkin menyertakan gaya yang tidak diperlukan jika tidak dikustomisasi.

3.5 Django

Django adalah *framework web python* tingkat tinggi yang memungkinkan pengembangan situs web secara cepat, aman dan terpelihara (Sabita, et al., 2022). Salah satu keunggulan *django* adalah, *framework* ini telah merepresentasikan ORM (*Object Relational Mapper*) sehingga tidak perlu menyesuaikan *query* jika terjadi perubahan *database* yang digunakan (Rahimah, et al., 2019). *Framework* ini juga

menyediakan antarmuka *admin* otomatis yang mempermudah pengelolaan data tanpa memerlukan pengembangan antarmuka tambahan. Dengan arsitektur modular dan dukungan untuk berbagai *database*, Django dapat digunakan untuk aplikasi *web* dari yang sederhana hingga kompleks. Selain itu, Django mencakup banyak fitur bawaan seperti *routing* tautan dan autentikasi pengguna, mengurangi kebutuhan menulis kode tambahan.

Django memanfaatkan berkas konfigurasi untuk mengatur berbagai aspek aplikasi, seperti pengaturan *database*, *URL routing*, dan pengaturan keamanan. Berkas konfigurasi ini memungkinkan pengembang untuk menyesuaikan dan mengelola berbagai komponen aplikasi dengan cara yang terstruktur. Selain itu, *Django* menyediakan perintah `python manage.py runserver` untuk menjalankan server pengembangan lokal. Dengan perintah ini, aplikasi dapat diakses dan diuji langsung melalui browser, memungkinkan pengembang untuk melihat hasil perubahan secara real-time selama proses pengembangan.

3.6 Three.js

Three.js adalah *library Javascript* untuk membuat game dan aplikasi 3D. *Three.js* menggunakan *WebGL* yang mana merupakan *API Javascript* untuk merender grafik 2D dan 3D interaktif yang *native browser* tanpa memerlukan plugin browser tambahan (Ramdhan & Rosyidah, n.d.). Library ini memudahkan pengembang dalam membuat grafik 3D yang kompleks dengan menyediakan abstraksi yang lebih tinggi dan lebih mudah digunakan dibandingkan dengan *WebGL* murni. *Three.js* digunakan untuk membuat berbagai jenis objek 3D, mengatur pencahayaan, tekstur, dan kamera, serta membuat animasi dan interaksi dalam lingkungan 3D. *Library* ini sering digunakan dalam pengembangan game, visualisasi data, simulasi, dan berbagai aplikasi interaktif lainnya.

Untuk menggunakan *Three.js*, library ini dapat diakses dengan mengunduh file `three.min.js` dari situs resminya atau dengan mengimpor dari CDN (Content Delivery Network). Setelah itu, library dapat dimasukkan ke dalam proyek dengan menambahkan script pada file HTML. *Three.js* kompatibel dengan semua browser

modern yang mendukung *WebGL*, menjadikannya sangat fleksibel untuk berbagai aplikasi web.

3.7 Swiper.js

Swiper.js adalah *library Javascript* yang digunakan untuk membuat *slider* atau carousel yang responsif dan interaktif pada aplikasi web. Library ini dirancang untuk berfungsi dengan baik pada perangkat *mobile* dan *desktop*, menyediakan berbagai fitur canggih untuk navigasi dan presentasi konten. *Swiper.js* digunakan untuk membuat komponen slider yang dapat menampilkan gambar, konten teks, atau elemen HTML lainnya secara berurutan dengan efek transisi yang menarik. Library ini mendukung navigasi geser (*swipe*) di perangkat sentuh, tombol panah, dan *pagination*, sehingga meningkatkan pengalaman pengguna dalam menjelajahi konten.

Swiper.js dapat diakses dengan mengunduh file Swiper dari situs resminya atau mengimpor dari CDN. Setelah itu, *library* dapat dimasukkan ke dalam proyek dengan menambahkan *script* dan *stylesheet* pada file HTML. Performa tinggi merupakan keunggulan lain dari *Swiper.js*, karena dioptimalkan untuk kinerja cepat dan efisien, bahkan dengan jumlah *slide* yang besar dan efek transisi yang kompleks.

3.8 Owl Carousel

Owl Carousel adalah *plugin jQuery* yang digunakan untuk membuat *slider* atau carousel yang responsif dan mudah disesuaikan pada aplikasi web. Plugin ini menyediakan berbagai fitur untuk navigasi dan presentasi konten, serta mendukung tampilan yang baik di berbagai perangkat. *Owl Carousel* digunakan untuk membuat komponen *slider* yang menampilkan gambar, teks, atau elemen HTML lainnya dalam format yang bergulir secara horizontal atau vertikal. Plugin ini mendukung navigasi *swipe* pada perangkat sentuh, tombol panah, dan *pagination*, sehingga meningkatkan pengalaman pengguna.

Untuk menggunakan Owl Carousel, plugin ini dapat diunduh dari situs resminya atau diimpor dari CDN. Setelah itu, library dapat dimasukkan ke dalam

proyek dengan menambahkan script dan stylesheet pada file HTML. Owl Carousel membutuhkan *jQuery* sebagai dependensi utama. Plugin ini kompatibel dengan semua browser modern, menjadikannya sangat fleksibel untuk berbagai aplikasi web.

3.9 Github

GitHub adalah *software* hosting untuk proyek *open source* yang menggunakan sistem kontrol revisi *Git* (Hidayatulloh, 2015). Platform ini memungkinkan pengembang untuk menyimpan, mengelola, dan berbagi kode sumber proyek dengan menggunakan sistem kontrol versi *Git*. *GitHub* menawarkan berbagai fungsi utama, termasuk penyimpanan kode sumber, kolaborasi melalui fitur *pull request* dan *review* kode, pelacakan masalah, serta dukungan untuk *continuous integration/continuous deployment* (CI/CD) melalui *GitHub Actions*. Selain itu, *GitHub* menyediakan fasilitas untuk dokumentasi proyek melalui berkas *README* dan wiki, serta mencatat setiap perubahan pada kode untuk memantau histori dan versi.

Kelebihan *GitHub* meliputi kemudahan kolaborasi, integrasi dengan berbagai alat dan layanan pengembangan, dukungan untuk komunitas *open source*, serta kontrol akses yang granular. Pengguna dapat mengakses *GitHub* melalui *browser web* di github.com, menggunakan aplikasi *desktop GitHub Desktop*, atau melalui *command line* dengan perintah *Git* standar seperti *git clone*, *git pull*, dan *git push*. *GitHub* merupakan alat yang penting dalam pengembangan perangkat lunak modern, memfasilitasi pengelolaan kode sumber dan kolaborasi yang efektif.

3.10 Anaconda Navigator

Anaconda Navigator adalah antarmuka grafis yang disediakan oleh *Anaconda* untuk mempermudah pengelolaan lingkungan pengembangan *Python* dan *R*. Dengan *Anaconda Navigator*, pengguna dapat dengan mudah menginstal, mengelola, dan meluncurkan berbagai aplikasi dan paket yang digunakan dalam analisis data, pembelajaran mesin, dan pengembangan ilmiah. *Navigator* menawarkan berbagai fitur seperti pengelolaan lingkungan virtual, pemasangan

paket, dan peluncuran aplikasi seperti *Jupyter Notebook* dan *Spyder* tanpa memerlukan perintah baris. Ini membuat *Anaconda Navigator* menjadi alat yang sangat berguna bagi pengguna yang menginginkan solusi yang lebih mudah diakses dan dikelola dalam ekosistem *Anaconda*.

Salah satu kelebihan utama *Anaconda Navigator* adalah kemampuannya untuk menyederhanakan proses pengelolaan paket dan lingkungan. Dengan antarmuka yang intuitif, pengguna dapat menghindari kompleksitas pengelolaan dependensi dan konflik versi yang sering terjadi dalam pengembangan perangkat lunak. *Navigator* juga memfasilitasi integrasi dengan berbagai alat analisis dan visualisasi data, memungkinkan pengembang dan ilmuwan data untuk fokus pada pekerjaan mereka tanpa terganggu oleh konfigurasi teknis yang rumit. Sebagai tambahan, *Navigator* mendukung integrasi dengan repositori paket *Anaconda* dan *conda-forge*, memberikan akses ke ribuan paket tambahan yang dapat diinstal dan digunakan dengan mudah.

3.11 Visual Studio Code

Visual Studio Code adalah sebuah teks *editor* ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows* (Ningsih, et al., 2022). Dengan antarmuka yang intuitif, *VS Code* menawarkan pemeriksaan sintaksis, *autocompletion*, *debugging*, dan integrasi dengan sistem kontrol versi seperti *Git*. *Editor* ini juga mendukung ekstensi, yang memungkinkan pengguna untuk menambah fungsionalitas sesuai dengan kebutuhan spesifik proyek, seperti dukungan bahasa tambahan, alat pengembangan, dan tema kustom.

Salah satu kelebihan utama dari *Visual Studio Code* adalah kemampuannya untuk disesuaikan dan diperluas melalui ekstensi. Pengguna dapat mengakses ribuan ekstensi yang tersedia di *marketplace* untuk menambahkan fitur baru, seperti integrasi dengan alat lain, dukungan *framework* spesifik, atau perbaikan produktivitas. *VS Code* juga mendukung fitur produktivitas seperti *snippet* kode, *multi-caret editing*, dan terminal bawaan, yang memudahkan pengelolaan dan eksekusi tugas langsung dari editor.

BAB 4 HASIL DAN PEMBAHASAN

4.1 Proses Pengerjaan

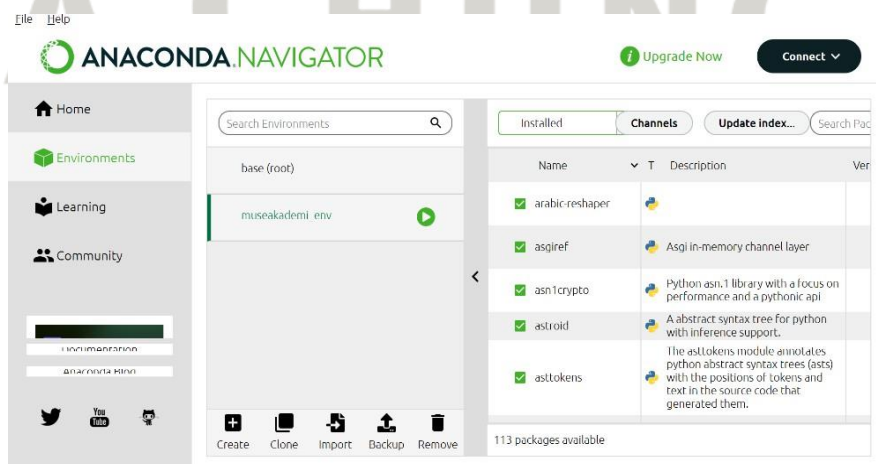
Proyek magang yang dilaksanakan di kantor Muse Akademi dimulai pada tanggal 1 Februari 2024 dan berakhir pada tanggal 1 Agustus 2024. Kegiatan magang dilakukan di kantor Muse Akademi yang terletak di Jl. Selat Karimata No.E6/1, Lesanpuro, Kec. Kedungkandang, Kota Malang, Jawa Timur. Kegiatan magang dilakukan selama 5 hari dalam seminggu yaitu hari Senin - Jumat dan berlangsung selama 8 jam sehari yaitu pukul 08.00 - 16.00.

4.2 Analisa Kebutuhan

Penulis mengembangkan kurang lebih empat website yang mencakup Muse Akademi, Brotani, Museverse, dan Fun Learning Digital. Setiap proyek memiliki alur pengerjaan yang berbeda untuk memenuhi kebutuhan spesifiknya. Berikut detail analisa kebutuhan dari setiap *website*:

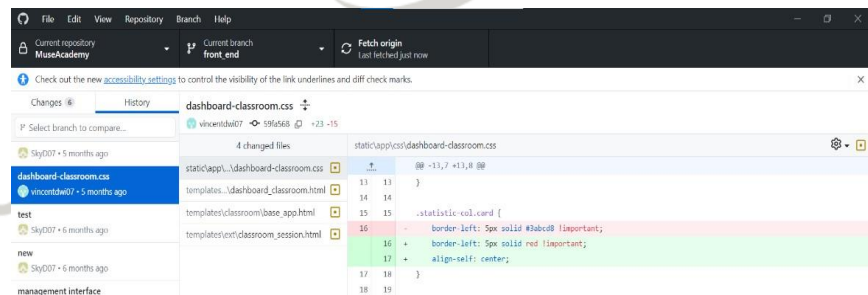
4.2.1 Muse Akademi:

- a. Memulai Lingkungan Pengembangan: Proses dimulai dengan membuka *Anaconda Navigator* untuk mengelola lingkungan pengembangan. Penulis memilih environment khusus yang telah disediakan untuk proyek Muse Akademi. Selanjutnya, penulis meluncurkan *Visual Studio Code* dari *Anaconda Navigator* sebagai *editor* kode utama untuk pengembangan.



Gambar 4. 1 Persiapan menggunakan *Anaconda Navigator*

- b. Pengembangan dan Implementasi: Setelah *environment* siap, penulis mulai mengerjakan kebutuhan yang telah ditetapkan oleh *supervisor*. Ini mencakup pengembangan fungsionalitas yang diinginkan, serta penggantian desain dan perubahan tata letak sesuai dengan permintaan *supervisor*.
- c. Desain dan Penyesuaian: Penulis melakukan penyesuaian desain untuk memastikan bahwa antarmuka pengguna sesuai dengan panduan desain dan kebutuhan spesifik Muse Akademi. Ini termasuk perubahan tata letak, pengaturan warna, dan elemen visual lainnya.
- d. Pengujian dan Validasi: Setelah implementasi dan penyesuaian selesai, penulis melakukan pengujian untuk memastikan semua fitur berfungsi dengan baik dan sesuai dengan harapan klien.
- e. Finalisasi: Setelah pengujian selesai, penulis melakukan finalisasi pada kode yang telah dikembangkan. Ini melibatkan pemeriksaan akhir dan persiapan kode untuk disimpan, yang kemudian di-push ke *GitHub* untuk versi kontrol dan kolaborasi lebih lanjut *website* siap untuk digunakan oleh pengguna akhir.



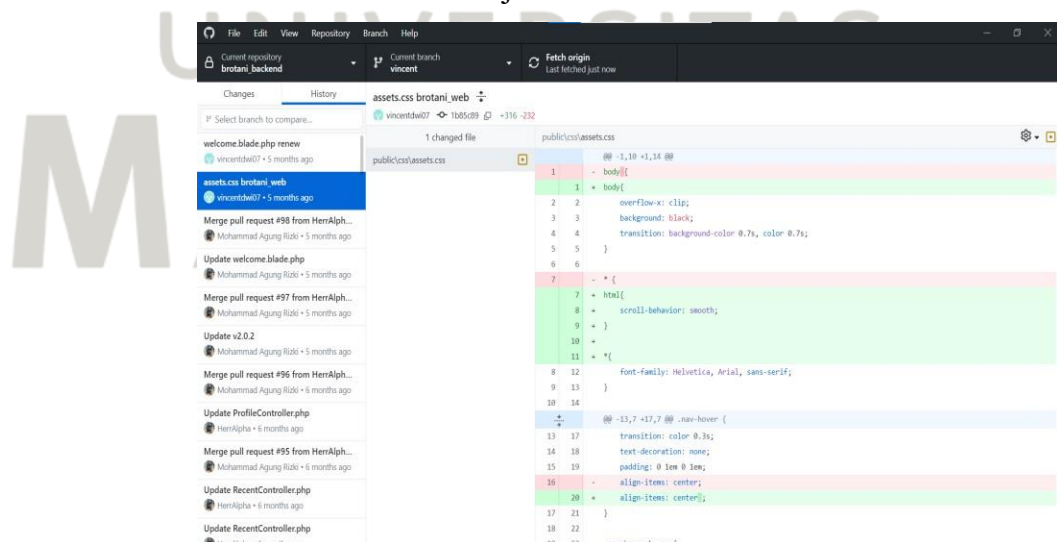
Gambar 4. 2 *Github* Muse Akademi

4.2.2 Brotani

- a. Pengembangan Awal: Penulis memulai proyek dengan membuat *website* dari awal, tanpa menggunakan *template* atau dasar yang ada. Proses ini melibatkan perancangan dan implementasi semua aspek *website*, dimulai dari struktur dasar hingga fitur fungsional.
- b. Desain dan Implementasi: Dalam tahap ini, penulis menggunakan *Visual Studio Code* sebagai *editor* teks utama dan memanfaatkan *HTML*, *CSS*, dan *Javascript* untuk pengembangan *website*. Semua

elemen desain dan fungsionalitas diimplementasikan dari awal, mencakup tata letak, gaya, dan interaktivitas sesuai dengan kebutuhan proyek.

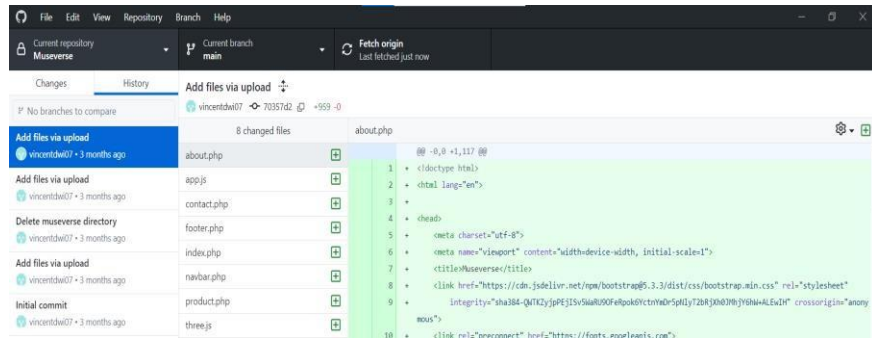
- c. Perbandingan dan Penyesuaian Desain: *Website* Brotani sebelumnya telah dibuat, namun *supervisor* ingin melakukan perubahan desain total. Penulis membandingkan desain *website* yang sudah ada sebagai referensi untuk memahami praktik terbaik dan tren terkini. Berdasarkan perbandingan tersebut, penulis melakukan perubahan desain total pada *website* Brotani, menyesuaikan tampilan dan tata letak agar sesuai dengan tujuan dan kebutuhan proyek.
- d. Verifikasi dengan *Supervisor*: Setelah desain dan implementasi selesai, penulis meminta umpan balik dari *supervisor* untuk memastikan bahwa desain dan fungsionalitas *website* sudah memenuhi standar dan harapan. Penulis melakukan penyesuaian akhir sesuai dengan masukan yang diberikan.
- e. Finalisasi: Setelah mendapatkan persetujuan dari *supervisor*, penulis melakukan finalisasi kode, memastikan semua elemen berfungsi dengan baik, dan mempersiapkan kode untuk disimpan. Kode yang telah diselesaikan kemudian di-push ke GitHub untuk versi kontrol dan kolaborasi lebih lanjut.



Gambar 4. 3 Github Brotani

4.2.3 Museverse

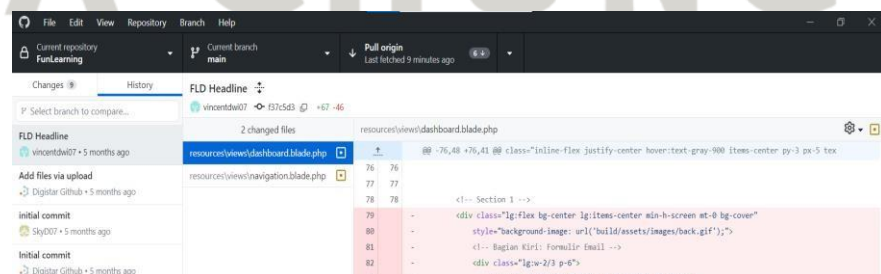
- a. Pengembangan Awal: Penulis memulai proyek dengan menyempurnakan *website* berdasarkan desain yang telah dibuat sebelumnya di *WordPress*. Desain di *WordPress* masih berbentuk abstrak dan belum sepenuhnya selesai, sehingga tugas penulis adalah menerjemahkan dan menyempurnakan desain tersebut menggunakan HTML, CSS, dan *Javascript*.
- b. Desain dan Implementasi: Penulis menggunakan *Visual Studio Code* sebagai editor teks utama untuk pengembangan. Proses ini melibatkan penerjemahan desain dari *WordPress* menjadi kode yang fungsional dan estetis, serta menyesuaikan tata letak dan gaya sesuai dengan spesifikasi yang telah ditetapkan. Beberapa plugin seperti *Three.js*, *Swiper.js*, dan *Owl Carousel* digunakan untuk menambahkan elemen interaktif dan fungsional pada *website*.
- c. Perbandingan dan Penyesuaian: Dalam tahap ini, penulis melakukan penyesuaian desain dan fungsionalitas berdasarkan umpan balik dan kebutuhan proyek. Penulis memastikan bahwa semua elemen desain yang awalnya abstrak dapat diterjemahkan dengan akurat ke dalam kode, serta mengoptimalkan penggunaan plugin untuk meningkatkan pengalaman pengguna.
- d. Verifikasi dengan *Supervisor*: Setelah implementasi selesai, penulis meminta umpan balik dari *supervisor* untuk memastikan bahwa *website* sudah sesuai dengan desain yang diinginkan dan memenuhi standar kualitas yang diharapkan. Penulis melakukan penyesuaian akhir berdasarkan masukan yang diberikan.
- e. Finalisasi: Setelah mendapatkan persetujuan dari *supervisor*, penulis melakukan finalisasi kode, memastikan semua elemen berfungsi dengan baik, dan mempersiapkan kode untuk disimpan. Kode yang telah diselesaikan kemudian di-*push* ke *GitHub* untuk versi kontrol dan kolaborasi lebih lanjut.



Gambar 4. 4 Github Museverse

4.2.4 Fun Learning Digital (FLD)

- Pengembangan Awal: Penulis memulai proyek dengan melakukan pengembangan awal menggunakan *Visual Studio Code* sebagai teks editor. Proses ini melibatkan penerapan desain awal dan implementasi fungsionalitas dasar menggunakan HTML, CSS, dan *Javascript*. *Tailwind CSS* digunakan sebagai *framework* untuk memudahkan penerapan gaya dan layout responsif.
- Penyesuaian Desain: Penulis melakukan perbaikan kecil pada desain *website* yang telah dibuat, dengan fokus pada peningkatan tampilan dan fungsionalitas sesuai kebutuhan proyek. Penyesuaian ini termasuk perubahan elemen visual untuk memastikan keselarasan dengan standar desain yang diinginkan.
- Penyerahan Proyek: Setelah perbaikan desain selesai, penulis melakukan finalisasi kode dan mempersiapkan proyek untuk diserahkan. Kode yang telah diselesaikan di-push ke *Git* untuk versi kontrol dan kolaborasi lebih lanjut, sebelum proyek diserahkan kepada tim lain untuk tahap pengembangan lebih lanjut.



Gambar 4. 5 Github FLD

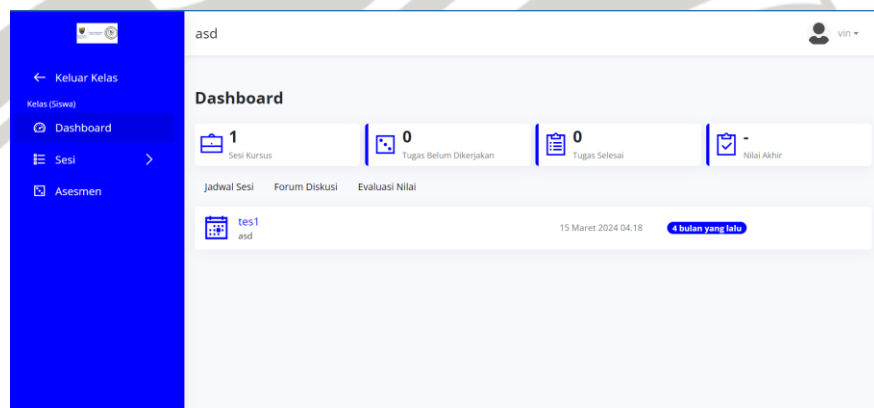
4.3 Desain

Berikut adalah desain-desain website yang telah penulis kembangkan, yang mencakup Muse Akademi, Brotani, Fun Learning Digital (FLD), dan Museverse. Setiap proyek memiliki pendekatan desain dan implementasi yang unik, mulai dari pengembangan awal hingga penyesuaian akhir. Penulis berfokus pada pembuatan dan pengoptimalan antarmuka pengguna untuk memastikan pengalaman yang menarik dan fungsional di setiap platform.

4.3.1 Muse Akademi

a. Dashboard_classroom.html

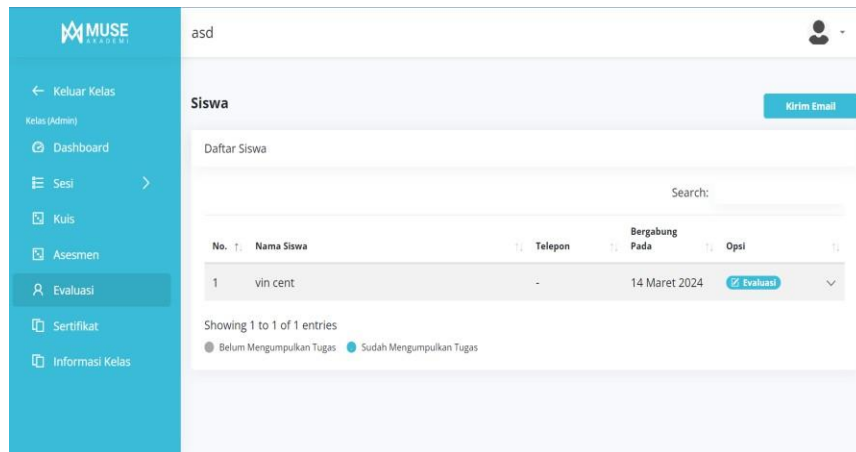
Perubahan tema warna halaman dilakukan untuk mengikuti warna dari kursus. Sebelumnya, warna ditentukan secara manual dan tidak terintegrasi dengan database.



Gambar 4. 6 Desain laman dashboard classroom

b. student_evaluation.html

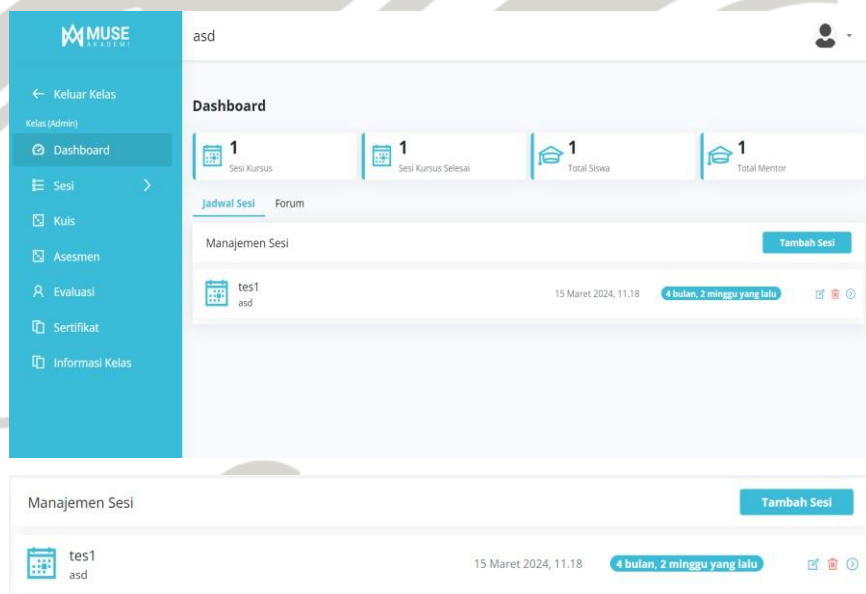
Perubahan desain tombol evaluasi di dalam tabel, penyamaan warna tema dengan tema Muse Akademi (biru muda), dan pemberian animasi rotasi untuk tanda panah.



Gambar 4. 7 Desain laman student evaluation

c. classroom_muse.html

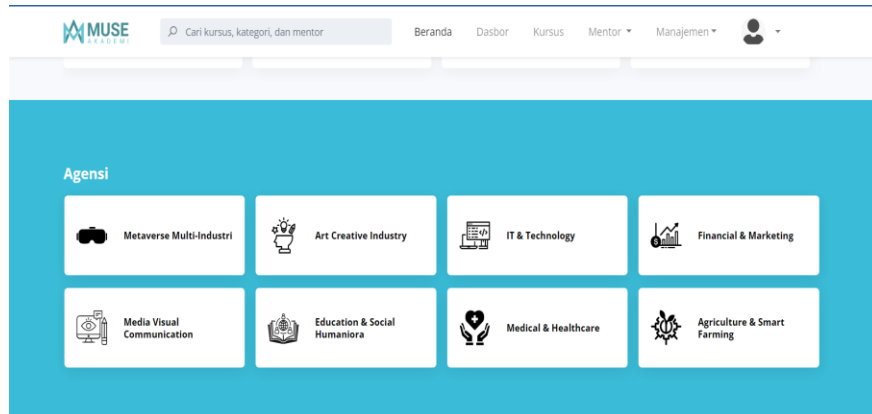
Penyamaan warna tema dengan Muse Akademi (biru muda) dan pembuatan tombol *edit*, *delete*, dan *view*.



Gambar 4. 8 Desain laman classroom muse

d. index.html

Penggantian bentuk dan format logo untuk setiap agensi di Muse Akademi dari .png ke .svg



Gambar 4. 9 Penggantian logo agensi Muse Akademi

e. index.html

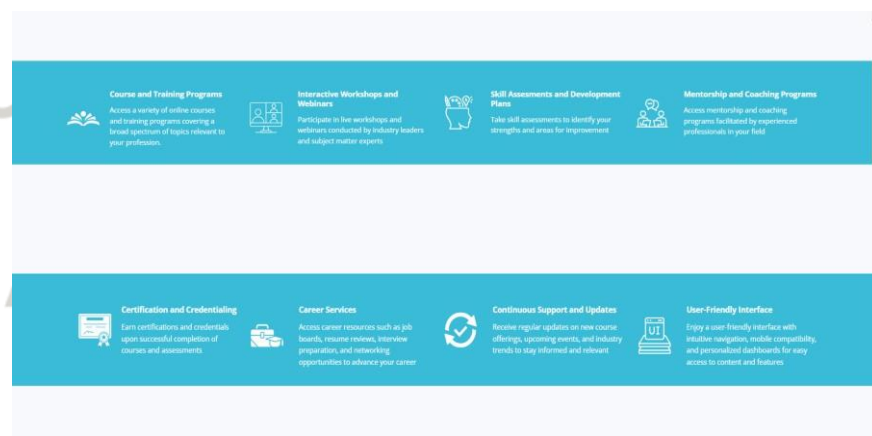
Penambahan fungsi pencarian, ketika logo ditekan, maka otomatis akan mengarahkan ke kotak input



Gambar 4. 10 Penambahan fungsi pencarian

f. index.html

Penggantian bentuk dan format logo untuk layanan Muse Akademi dari .png ke .svg dan penggantian sistem menjadi geser



Gambar 4. 11 Penggantian logo layanan Muse Akademi

g. home.html

Desain dan isi tulisan pada cover landing page Muse Akademi telah mengalami perubahan. Sebelumnya, desain

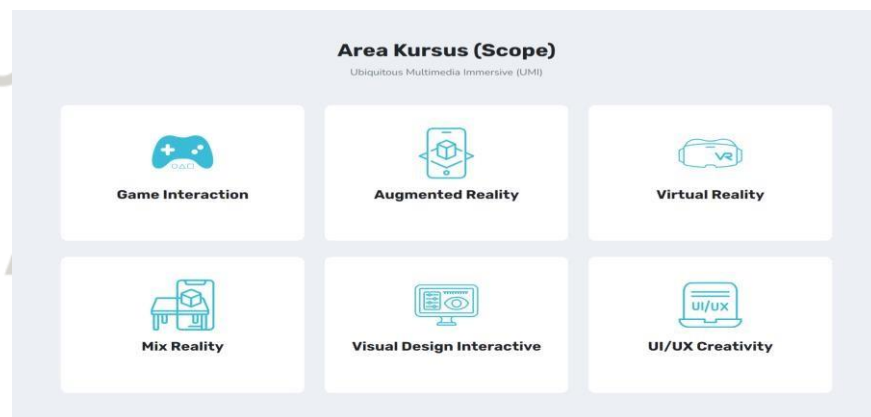
menggunakan *font* berjenis *cursive* yang memberikan kesan informal dan artistik. Namun, kini desain tersebut telah dirubah menjadi lebih formal dengan pemilihan jenis huruf yang lebih sederhana dan profesional.



Gambar 4. 12 Penggantian *cover landing page* Muse Akademi

h. home.html

Penambahan area kursus di landing page Muse Akademi bertujuan untuk memberikan informasi yang jelas dan mudah diakses oleh pengguna mengenai jenis-jenis kursus yang tersedia di Muse Akademi.



Gambar 4. 13 Area kursus *landing page* Muse Akademi

i. home.html

Penambahan sesi agensi di *landing page* Muse Akademi berfungsi untuk memberikan informasi kepada pengguna mengenai mitra-mitra yang bekerja sama dengan Muse Akademi.



Gambar 4. 14 Agensi *landing page* Muse Akademi

j. home.html

Penambahan paket kursus di *landing page* Muse Akademi bertujuan untuk memberikan informasi kepada pengguna mengenai kursus-kursus yang tersedia di Muse Akademi. Sesi ini dirancang agar pengguna dapat melihat pilihan kursus yang ditawarkan, termasuk detail paket kursus yang dapat mereka ikuti.

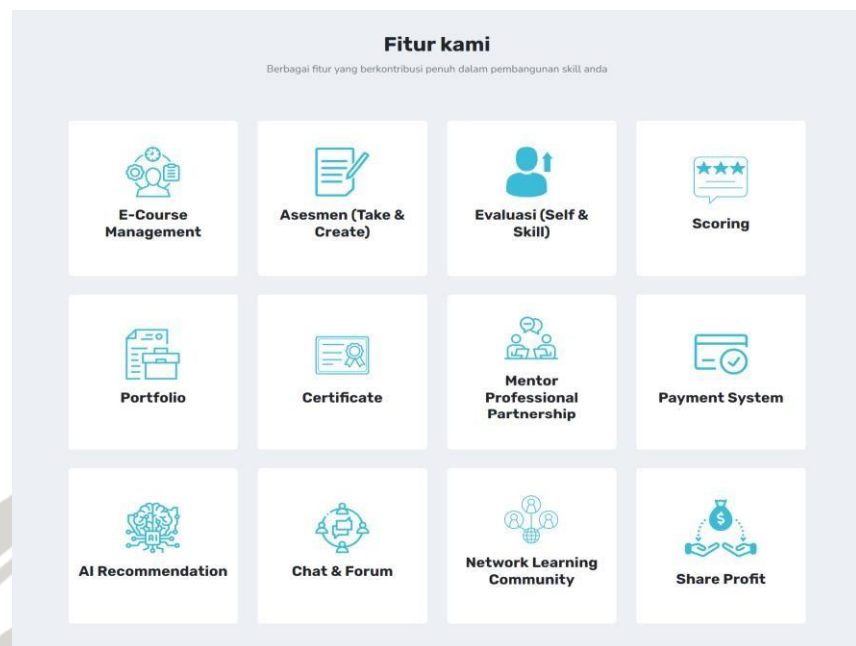


Gambar 4. 15 Paket kursus *landing page* Muse Akademi

k. home.html

Penambahan sesi fitur di *landing page* Muse Akademi bertujuan untuk memberikan informasi kepada pengguna mengenai

fitur-fitur yang akan mereka dapatkan ketika mengikuti kursus di Muse Akademi.



Gambar 4. 16 Fitur *landing page* Muse Akademi

l. home.html

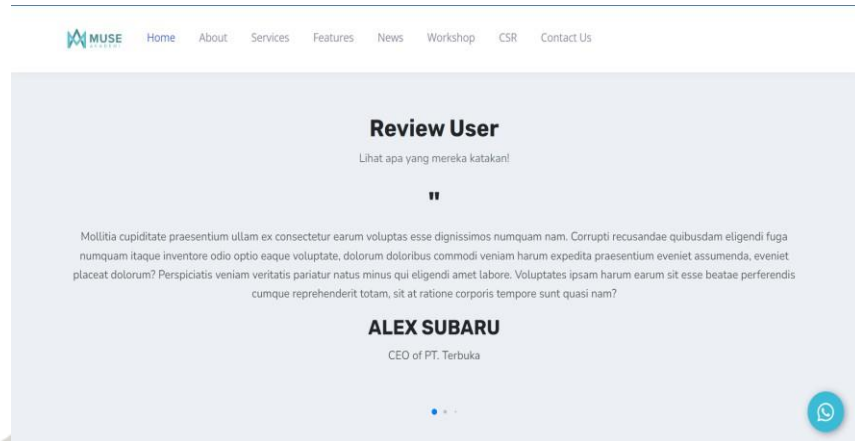
Penambahan sesi berita di *landing page* Muse Akademi bertujuan untuk memberikan informasi singkat mengenai berita-berita terbaru yang ada di Muse Akademi.



Gambar 4. 17 Sesi berita *landing page* Muse Akademi

m. home.html

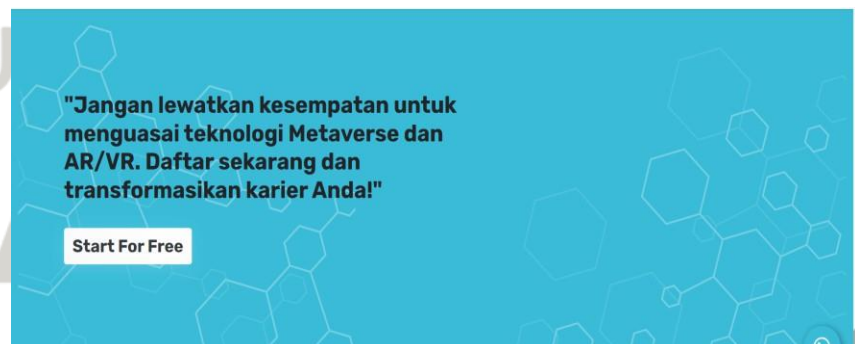
Penambahan sesi review pengguna di *landing page* Muse Akademi bertujuan untuk memberikan umpan balik dari pengguna yang telah mengikuti kursus di Muse Akademi.



Gambar 4. 18 Sesi review pengguna *landing page* Muse Akademi

n. home.html

Perubahan desain *call to action* di *landing page* Muse Akademi dilakukan untuk meningkatkan visibilitas dan daya tariknya. Sebelumnya, desain *call to action* kurang mencolok, sehingga penulis melakukan perbaikan dengan membuatnya lebih menonjol.



Gambar 4. 19 *Call to action* *landing page* Muse Akademi

o. home.html

Perubahan desain *footer* di *landing page* Muse Akademi bertujuan untuk menyederhanakan tampilan dan meningkatkan fungsionalitasnya. Sebelumnya, *footer* memuat berbagai informasi

yang tidak diperlukan, sehingga penulis merancang ulang footer agar lebih ringkas.



Gambar 4. 20 Footer landing page Muse Akademi

p. news.html

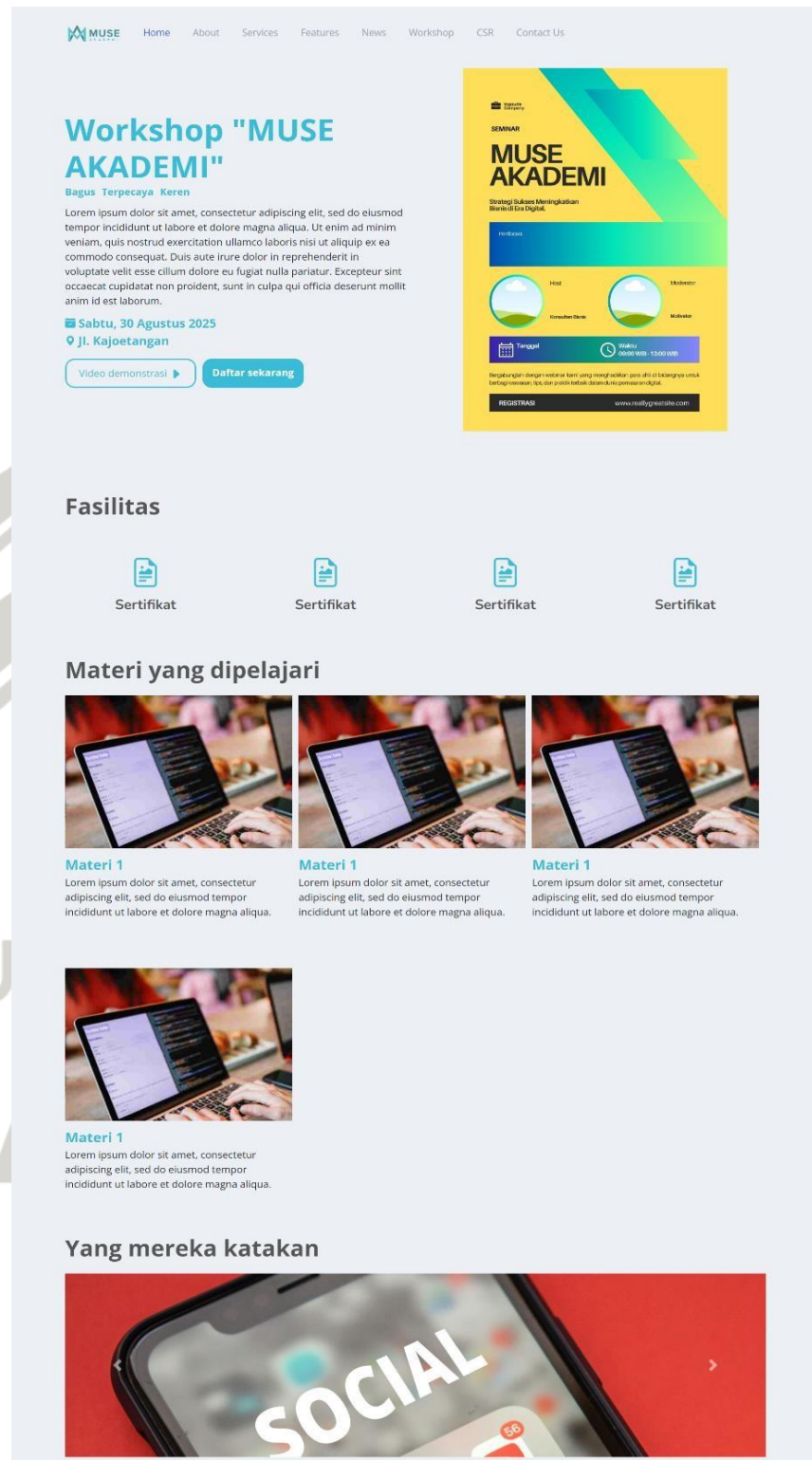
Pembuatan halaman berita baru di *landing page* Muse Akademi bertujuan untuk memberikan pengguna akses yang lebih mendalam terhadap berita dan informasi terkini dari Muse Akademi.



Gambar 4. 21 Halaman berita landing page Muse Akademi

q. workshop.html

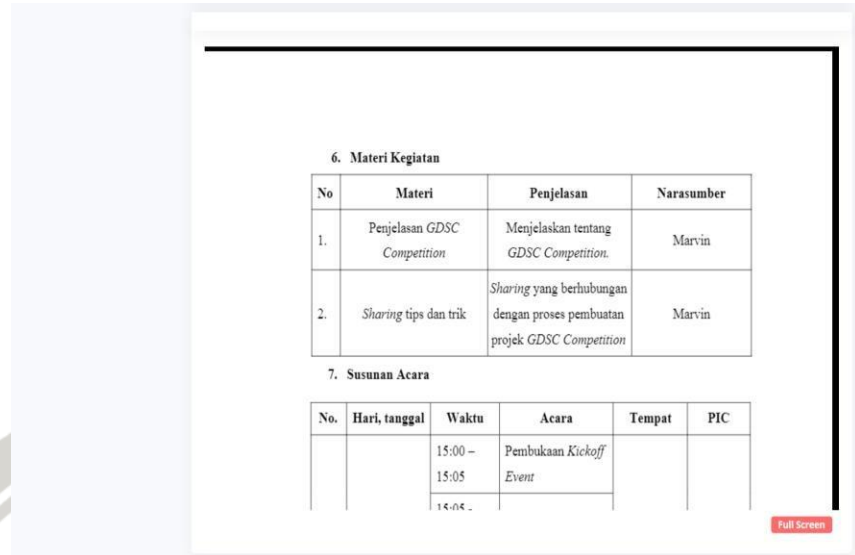
Pembuatan halaman untuk *workshop* Muse Akademi.



Gambar 4. 22 Halaman *workshop landing page* Muse Akademi

r. classroom_session.html

Pembuatan sistem *pdf viewer* untuk membaca file dalam format .pdf dan .pptx.



Gambar 4. 23 Pdf viewer

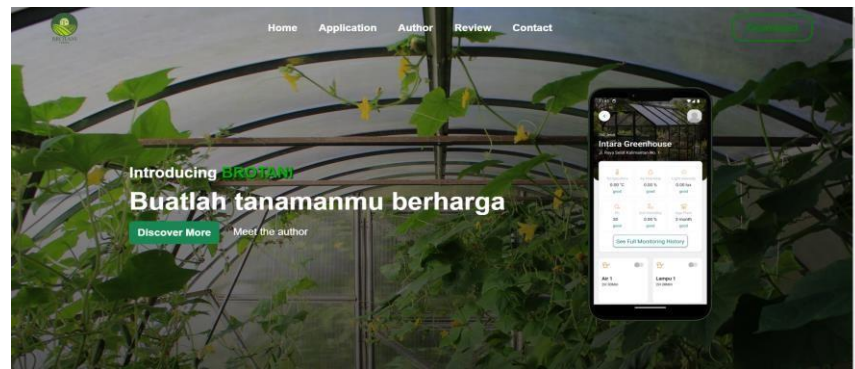
4.3.2 Brotani

Pembuatan *landing page* untuk *website* Brotani yang hanya terdiri dari satu halaman saja. Berikut ini adalah gambar-gambar potongan dari *website* Brotani. Gambar-gambar ini diambil untuk memberikan gambaran detail tentang elemen-elemen yang ada di *landing page*. Karena halaman *landing page* terdiri dari satu halaman penuh, penjelasan ini ditampilkan dalam beberapa bagian agar dapat dilihat dengan lebih jelas dan terperinci.

a. Sesi 1

Di sesi 1 ini, halaman cover awal *landing page* Brotani menampilkan *tagline* dan foto aplikasi Brotani. *Tagline* disajikan untuk menyampaikan pesan utama dan nilai dari aplikasi tersebut secara ringkas. Foto aplikasi memberikan gambaran visual mengenai antarmuka dan fitur-fitur utama yang ditawarkan oleh Brotani. Halaman ini dirancang untuk menyambut pengguna

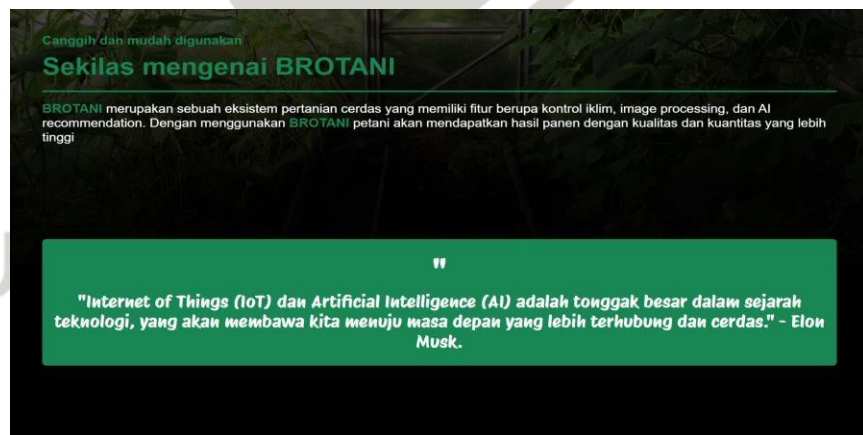
dengan informasi awal yang relevan dan menarik, mempromosikan aplikasi secara efektif.



Gambar 4. 24 Sesi 1 *website* Brotani

b. Sesi 2

Di sesi 2 ini, halaman melanjutkan dari sesi sebelumnya dengan penjelasan sekilas tentang aplikasi Brotani. Halaman ini juga menyajikan kutipan mengenai Internet of Things (IoT) yang disampaikan oleh Elon Musk.



Gambar 4. 25 Sesi 2 *website* Brotani

c. Sesi 3

Di sesi 3, halaman melanjutkan dari sesi sebelumnya dengan penjelasan mendetail mengenai aplikasi Brotani. Sesi ini bertujuan untuk memberikan informasi lebih lanjut tentang fitur dan manfaat yang ditawarkan oleh Brotani. Selain itu, sesi ini juga mencakup ajakan kepada pengguna untuk mulai menggunakan Brotani,

menekankan keuntungan dan nilai tambah yang bisa diperoleh dari aplikasi tersebut.

Apa saja yang ada di dalam BROTANI?

Preview at Glance

BROTANI memberikan kemudahan ke para petani dengan memanfaatkan teknologi IoT dan Artificial Intelligence.

Apa itu BROTANI

Selamat datang di dunia pertanian masa depan yang penuh inovasi! BROTANI adalah sahabat terbaik petani modern, membawa kontrol iklim super canggih ke dalam greenhouse Anda. Dengan sentuhan teknologi mutakhir, kami menjaga suhu, kelembaban, cahaya, dan ventilasi agar selalu sempurna sesuai keinginan tanaman Anda. Bahkan, Brotani siap memberikan peringatan instan jika ada gangguan dalam sistem iklim.

Selamatkan energi, hemat waktu, dan hasilkan hasil panen luar biasa dengan BROTANI. Kami mudah diintegrasikan dengan sistem greenhouse yang telah ada. Dengan Brotani, Anda tidak hanya berinvestasi dalam pertanian yang lebih cerdas, tetapi juga dalam masa depan pertanian yang lebih berkelanjutan. Sambut pertanian yang efisien dan berkelanjutan dengan BROTANI - Partner Terpercaya Petani Modern!

Rubah greenhouse mu menjadi BROTANI

Bergabunglah dengan BROTANI sekarang untuk mengubah cara Anda mengelola greenhouse menjadi pengalaman yang lebih inovatif dan efisien. Proses pendaftaran sangat mudah; cukup hubungi admin kami untuk mendapatkan informasi awal, lalu kami akan mengatur kunjungan tim kami ke lokasi greenhouse Anda. Tim kami akan melakukan survei dan mengevaluasi kelayakan integrasi BROTANI.

Jika greenhouse Anda memenuhi persyaratan, kami akan membantu Anda mengisi formulir pendaftaran, dan admin kami akan memverifikasinya. Selamat, Anda sekarang menjadi anggota BROTANI, siap untuk mendapatkan manfaat dari kontrol iklim, computer vision, dan bantuan AI kami untuk meningkatkan produktivitas dan kualitas tanaman Anda serta mendukung pertanian yang lebih berkelanjutan.



Gambar 4. 26 Sesi 3 *website* Brotani

d. Sesi 4

Di sesi 4, halaman melanjutkan dari sesi sebelumnya dengan memberikan penjelasan mendetail mengenai manfaat dan fungsi aplikasi Brotani. Sesi ini dirancang untuk memberikan gambaran yang komprehensif tentang berbagai keuntungan yang dapat diperoleh pengguna dari penggunaan aplikasi Brotani. Penjelasan ini mencakup fungsi-fungsi utama yang ditawarkan oleh Brotani, menjelaskan fitur-fitur inti yang mendukung berbagai kegiatan dan proses yang diharapkan dapat memenuhi kebutuhan pengguna dengan efektif. Selain itu, sesi ini juga menguraikan secara rinci apa saja yang dapat diperoleh dari Brotani, seperti akses ke alat atau sumber daya khusus serta fitur-fitur eksklusif yang memperkaya pengalaman pengguna. Dengan menyajikan informasi yang mendetail ini, sesi 4 bertujuan untuk memperjelas nilai tambah yang ditawarkan oleh aplikasi dan memastikan bahwa pengguna dapat memahami dan memanfaatkan sepenuhnya berbagai

keuntungan yang disediakan oleh Brotani, sehingga dapat membuat keputusan yang lebih terinformasi mengenai penggunaannya.

Apa saja yang didapatkan dari BROTANI



Dengan BROTANI, Anda akan mendapatkan satu set alat yang lengkap untuk mengoptimalkan pengelolaan pertanian Anda. Ini meliputi satu set sensor yang canggih, yang memungkinkan pemantauan dan pengaturan suhu, kelembaban, cahaya, dan kondisi lingkungan di dalam greenhouse Anda dengan tingkat presisi yang tinggi. Sensor ini memastikan bahwa kondisi yang optimal selalu terjaga untuk pertumbuhan tanaman Anda.

Selain sensor, Anda juga akan mendapatkan satu set aktuator yang otomatis, memungkinkan Anda untuk mengontrol perangkat dalam greenhouse Anda tanpa harus melakukannya secara manual. Hal ini tidak hanya menghemat waktu, tetapi juga memastikan bahwa tanaman Anda selalu berada dalam kondisi terbaik.

Bukan hanya itu, BROTANI juga menyediakan satu set sistem pengaliran cerdas yang akan memastikan bahwa tanaman Anda selalu mendapatkan pasokan air yang sesuai dengan kebutuhan mereka, menghindari pemborosan air dan mendukung pertumbuhan yang sehat.

Apa yang dilakukan BROTANI

"BROTANI: Menyelamatkan Bumi, Satu Tanaman Serta-waktu."

Ketika kita berbicara tentang mengurangi dampak iklim, pertanian berkelanjutan adalah kunci jawabannya, dan BROTANI telah memilih untuk menjadi bagian dari solusi ini. Kami memahami bahwa dunia memerlukan perubahan mendalam dalam cara kita memandang pertanian. Dengan berkomitmen pada praktik pertanian yang berkelanjutan dan mengadopsi teknologi terbaru, kami membawa perubahan yang nyata. Kami merawat tanah dengan cinta, mengurangi limbah, dan mengoptimalkan penggunaan sumber daya alam. Setiap biji yang kami tanam adalah langkah menuju pemulihan lingkungan dan pengurangan emisi karbon. Bergabunglah dengan BROTANI, dan bersama-sama kita akan memelihara planet ini, satu tanaman serta-waktu."



Apa yang selanjutnya ada di BROTANI

Dalam BROTANI, Anda akan menemukan berbagai fitur canggih yang akan mengubah cara Anda mengelola pertanian. Kami hadir dengan pendeteksi awal hama yang dapat mengenali serangga, jamur, dan bakteri, membantu Anda mencegah kerusakan pada tanaman secara dini. Fitur Fotosintesis memberikan pemantauan tanaman yang lebih canggih, memungkinkan Anda untuk mengikuti perkembangan tanaman secara detail.

Selain itu, kami menyediakan fitur logistik yang membantu petani dalam mendapatkan pupuk berkualitas dan menjual hasil panen mereka dengan lebih efisien. Dengan fitur ini, Anda dapat dengan mudah mengelola aspek-aspek logistik yang terkait dengan pertanian Anda.

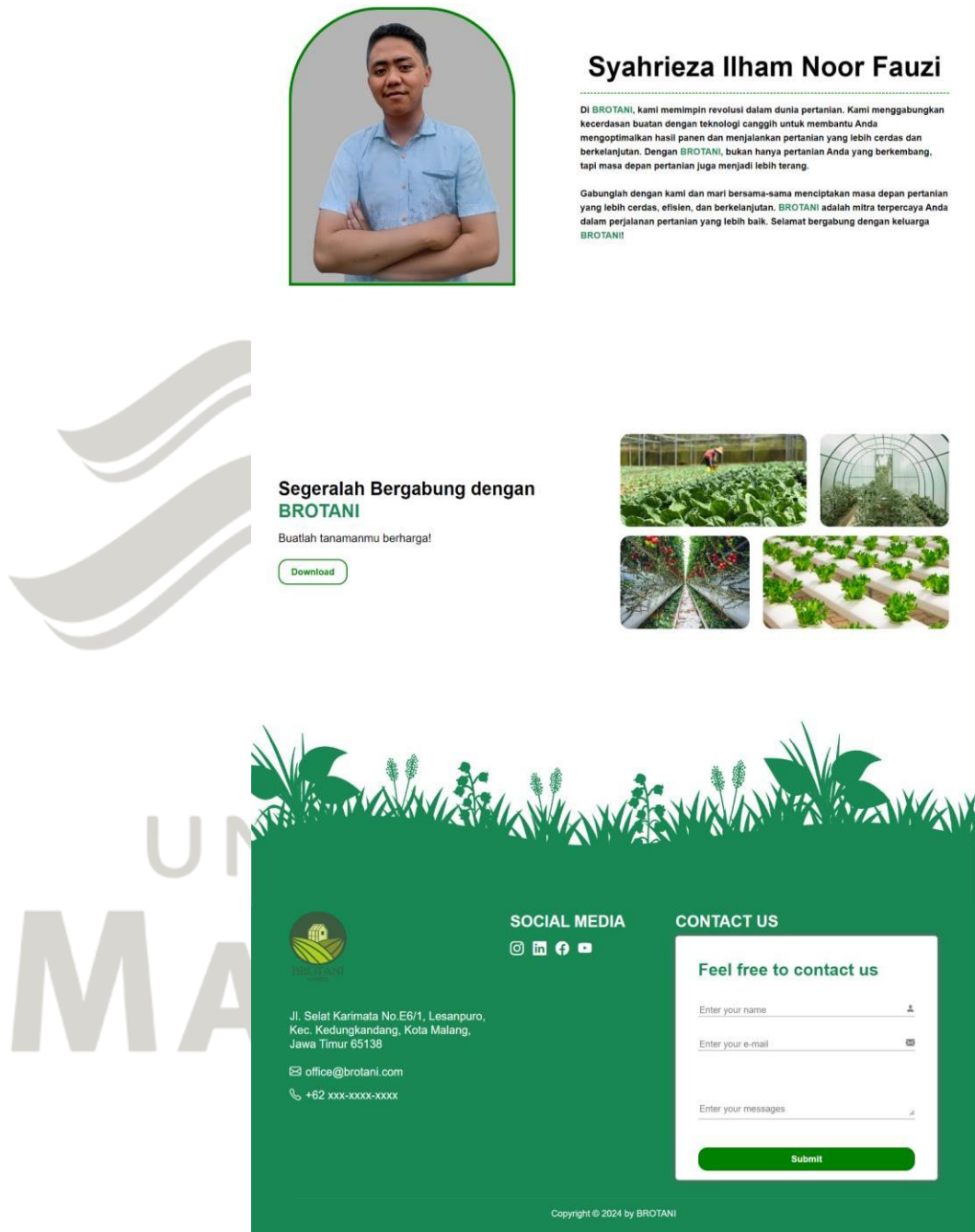
"The future is an unwritten book. We, as creators of technology, hold the pen in our own hands." - Steve Jobs

Gambar 4. 27 Sesi 4 website Brotani

e. Sesi 5

Di sesi 5, halaman melanjutkan dari sesi sebelumnya dengan memperkenalkan tokoh kunci di balik aplikasi Brotani, yaitu CEO Brotani. Halaman ini menyajikan perkenalan yang menonjolkan peran CEO serta kutipan inspiratif yang diucapkannya, dengan tujuan untuk mengajak para pengguna untuk mulai menggunakan aplikasi Brotani. Kutipan tersebut dirancang untuk menarik perhatian dan mendorong pengguna untuk mengambil langkah selanjutnya dalam menggunakan aplikasi. Selain itu, sesi ini mencakup *call to action* yang jelas, mengarahkan pengguna untuk mendaftar atau mengeksplorasi lebih lanjut fitur-fitur Brotani. Halaman ditutup dengan *footer*

yang menyediakan informasi tambahan yang relevan, memastikan pengguna memiliki akses mudah ke detail kontak dan informasi penting lainnya mengenai Brotani.

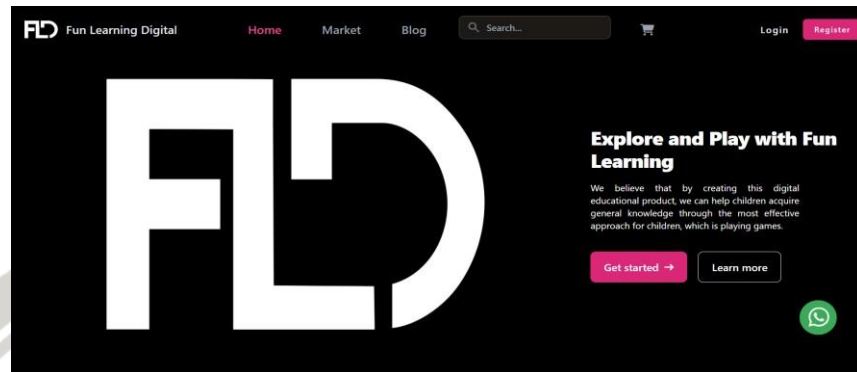


Gambar 4. 28 Sesi 5 website Brotani

4.3.3 Fun Learning Digital (FLD)

a. welcome.blade.php

Penghapusan background animasi dalam bentuk video (.mp4) yang memakan banyak memori sehingga memperlambat kinerja website, serta sedikit perbaikan untuk desain responsif pada bagian cover.



Gambar 4. 29 Cover Fun Learning Digital

Pembuatan desain yang responsif untuk sesi cover Fun Learning Digital bertujuan untuk memastikan tampilan yang optimal pada berbagai perangkat dan ukuran layar. Proses ini melibatkan penggunaan teknik desain responsif yang memungkinkan elemen-elemen halaman untuk menyesuaikan diri secara otomatis, menjaga konsistensi dan keterbacaan tampilan baik di desktop, tablet, maupun perangkat seluler.

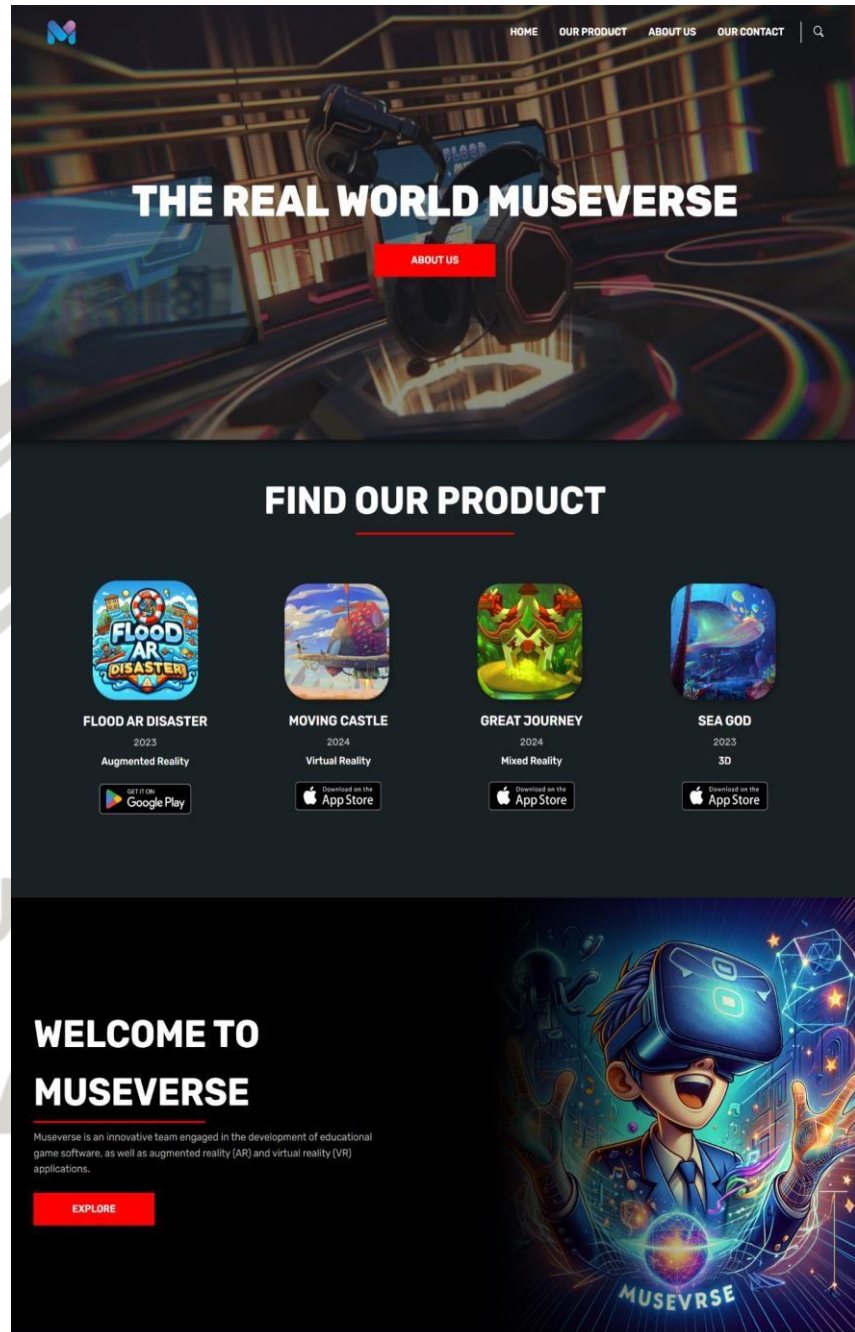


Gambar 4. 30 Desain responsive Fun Learning Digital

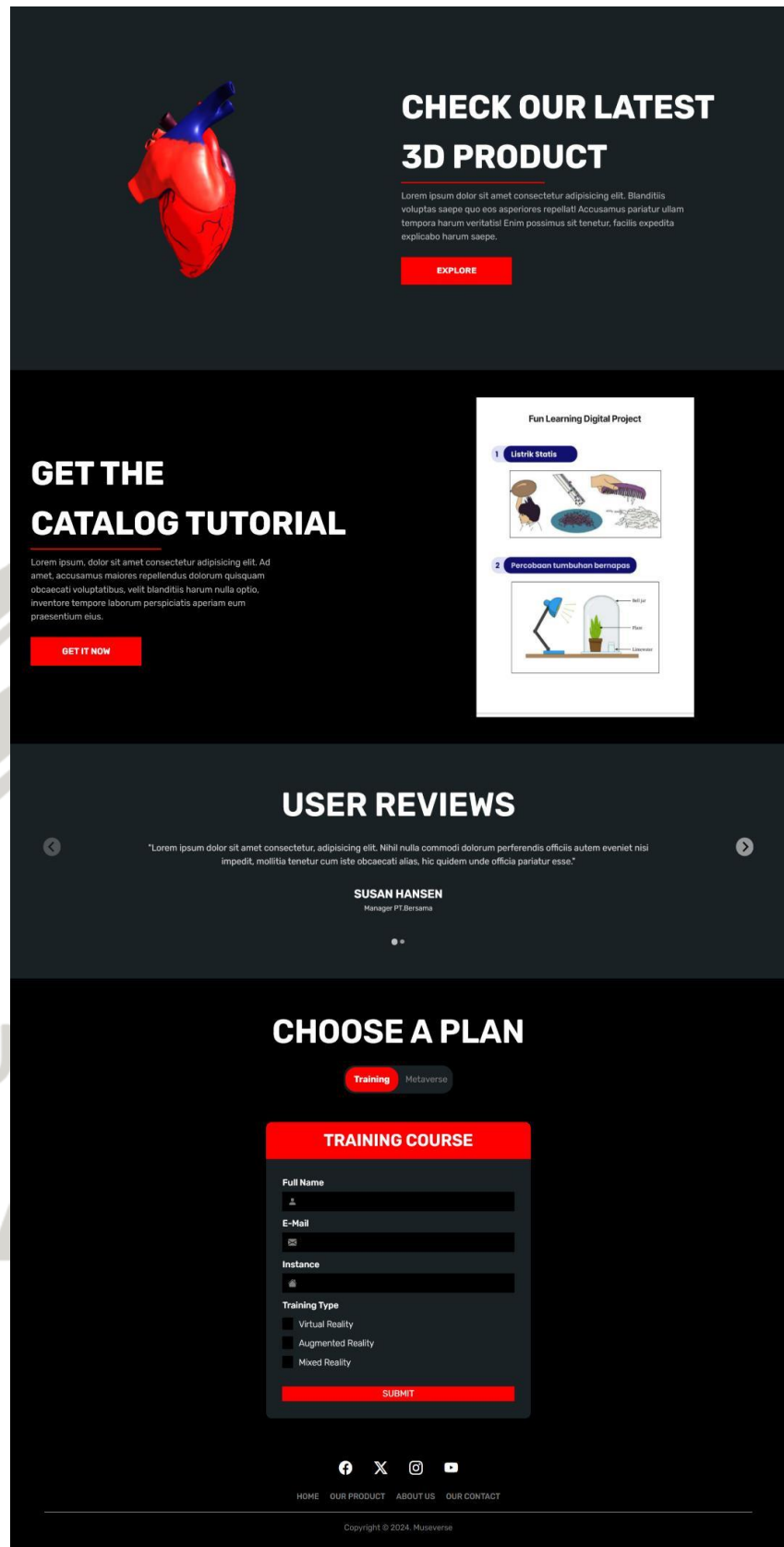
4.3.4 Museverse

a. index.html

Pembuatan desain untuk halaman awal *website* Museverse yang akan dibagi menjadi beberapa sesi



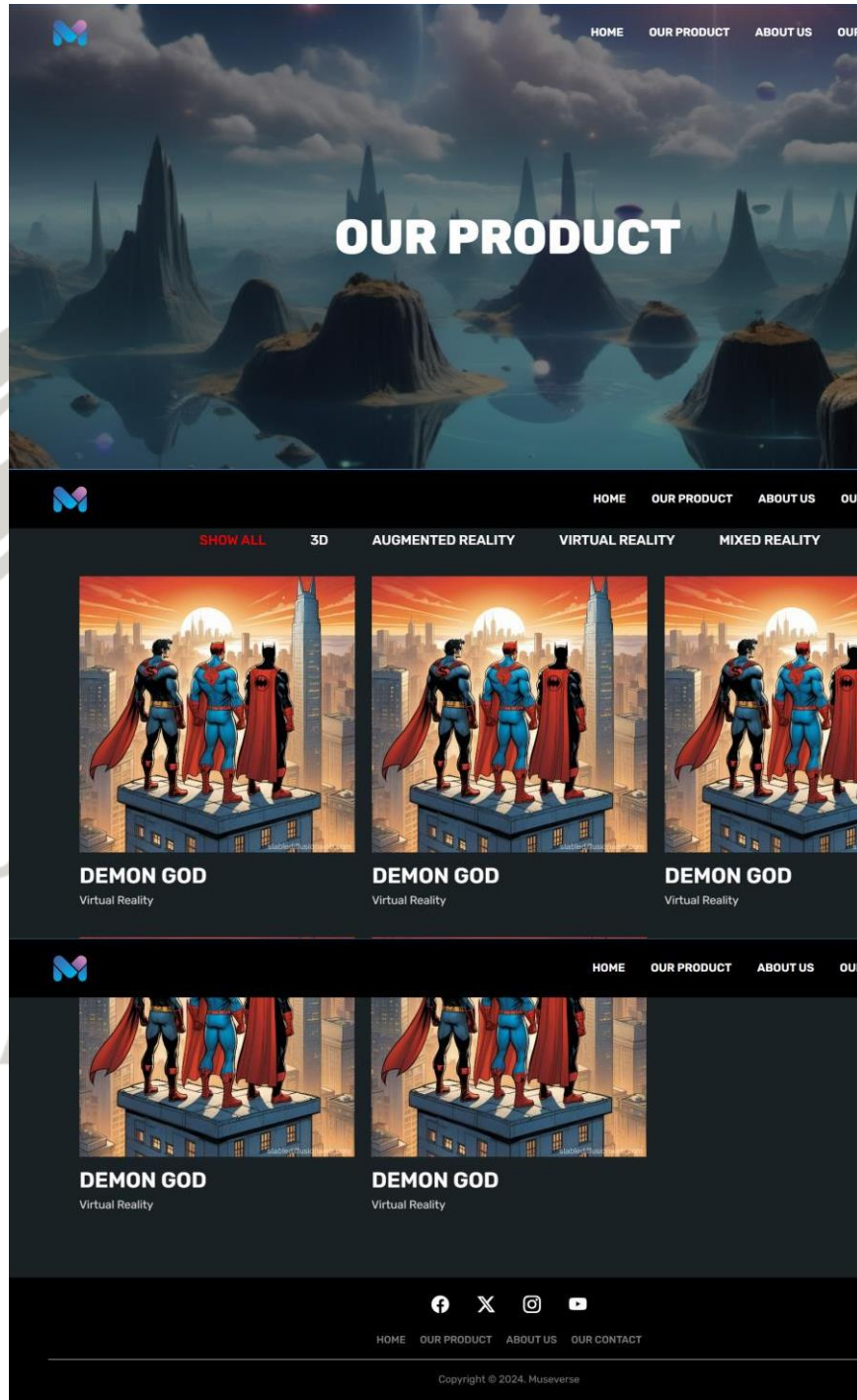
Gambar 4. 31 Sesi 1 index.html Museverse



Gambar 4. 32 Sesi 2 index.html Museverse

b. product.html

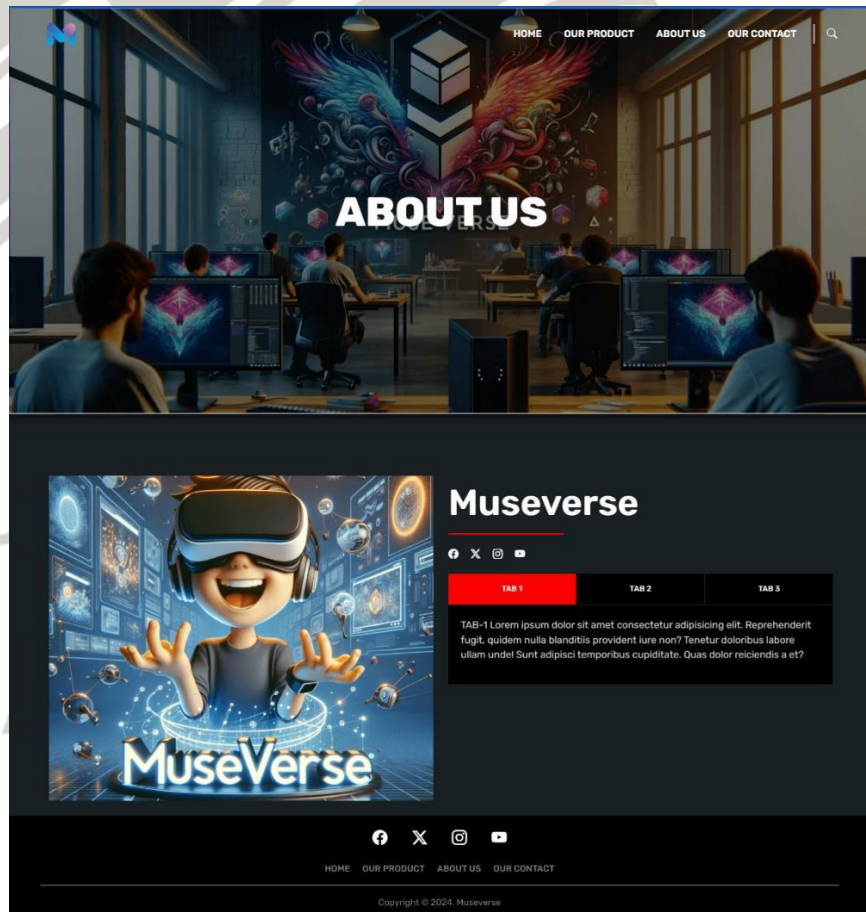
Halaman ini dirancang untuk menampilkan seluruh produk yang telah dibuat oleh Museverse. Produk-produk tersebut dibagi ke dalam berbagai kategori, termasuk objek 3D, *Virtual Reality*, *Augmented Reality*, dan *Mixed Reality*.



Gambar 4. 33 Halaman product.html Museverse

c. about.html

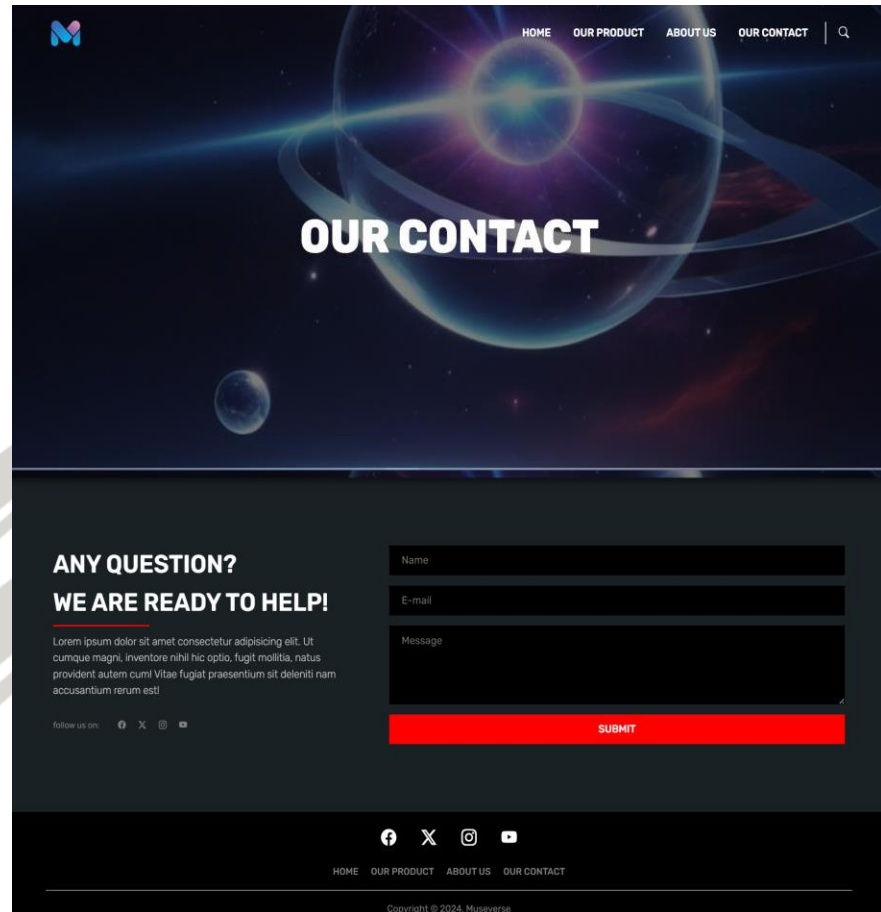
Halaman ini dirancang untuk memberikan informasi yang komprehensif mengenai Museverse. Di halaman ini, pengguna dapat mempelajari berbagai aspek penting tentang Museverse, mulai dari sejarah perkembangan perusahaan, visi yang membimbing arah dan tujuan jangka panjangnya, hingga misi yang menjadi dasar pelaksanaan kegiatan sehari-hari. Selain itu, halaman ini juga mencakup penjelasan mengenai nilai-nilai inti yang dipegang oleh Museverse, yang mencerminkan prinsip-prinsip dan etika yang mendasari setiap keputusan dan tindakan yang diambil oleh perusahaan.



Gambar 4. 34 Halaman about.html Museverse

d. contact.html

Halaman ini diperuntukkan untuk memberikan informasi kontak yang dapat dihubungi di Museverse.



Gambar 4. 35 Halaman about.html Museverse

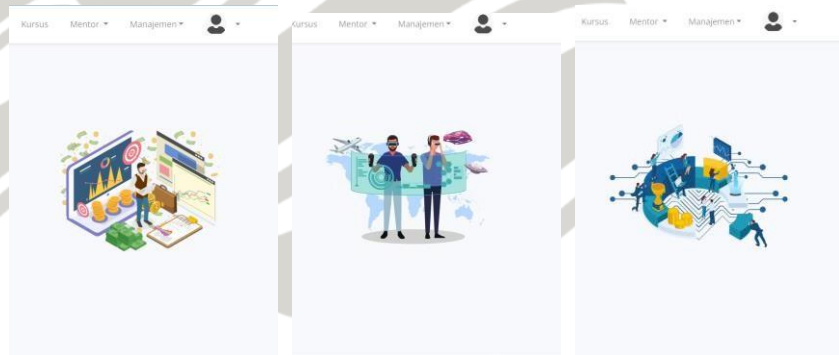
4.4 Pengembangan

Penulis telah mengimplementasikan teknologi berbasis *front-end* dalam mengembangkan berbagai fungsi yang memungkinkan pengguna untuk berinteraksi. Semua teknologi ini telah diimplementasikan dalam empat *website* yang penulis kembangkan selama PKL. Untuk mengembangkan fungsi-fungsi ini, penulis menggunakan bahasa pemrograman *Javascript* dan beberapa *plugin* berbasis *Javascript*. Berikut ini adalah beberapa detail fungsi yang telah penulis kembangkan dan implementasikan ke dalam empat website tersebut:

4.4.1 Muse Akademi

a. *Plugin Owl Carousel*

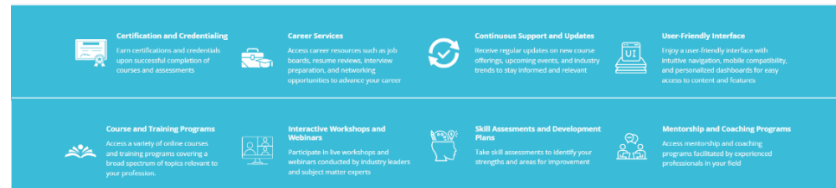
Dalam pengembangan website Muse Akademi, penulis paling banyak menggunakan fungsi dari *Owl Carousel*. Fungsi ini bertujuan untuk menampilkan gambar atau tulisan yang bisa digeser, sehingga tampilan *website* menjadi lebih modern dan dinamis. Dengan implementasi *Owl Carousel*, pengguna dapat dengan mudah melihat berbagai konten secara interaktif, meningkatkan pengalaman pengguna secara keseluruhan. Berikut adalah berbagai detail penggunaan Owl Carousel dalam website Muse Akademi dan kodenya:



Gambar 4. 36 *Owl Carousel 1* Muse Akademi

Gambar 4. 39 menunjukkan penggunaan *Owl Carousel* pada laman *index* Muse Akademi. Implementasi *Owl Carousel* di sini bertujuan untuk menambahkan gambar hiasan yang dapat digeser, memberikan elemen visual yang dinamis dan interaktif. Berikut kode dari implementasi *plugin* tersebut:

```
1 <div class="description-active owl-carousel">
2   <div class="row">
3     
5   </div>
6   <div class="row">
7     
9   </div>
10  <div class="row">
11    
13  </div>
14</div>
```



Gambar 4. 37 Owl Carousel 2 Muse Akademi

Gambar 4. 40 merupakan implementasi *Owl Carousel* di laman *index* Muse Akademi yang bertujuan untuk menampilkan fitur-fitur yang didapat oleh pengguna dari mengikuti kursus di Muse Akademi, memberikan elemen visual yang dinamis dan interaktif. Dengan fitur ini, pengguna dapat menggulir melalui berbagai informasi mengenai keuntungan dan fitur yang tersedia, membuat tampilan halaman lebih menarik dan informatif. Berikut kode dari *plugin* tersebut:

```

1 <div class="description-active owl-carousel">
2   <div class="row justify-content-center g-0">
3     <div class="col-sm-12 col-lg-3">
4       <div class="mt-3 service-item mx-3">
5         <div class="row align-items-
6           center">
7           <div class="p-0 col-xl-3 col-lg-3
8             col-md-2 col-sm-2 col-2 d-flex
9             align-items-baseline">
10              
14            </div>
15            <div class="p1-0 col-xl-9 col-lg-
16              9 col-md-10 col-sm-10 col-10 d-
17              flex align-items-start flex-
18              column">
19              <h6 class="text-white"><b>Course and
20                Training Programs</b></h6>
21              <p class="text-white mb-0">
22                Access a variety of online courses
23                and training programs covering a
24                broad spectrum of topics relevant to
25                your profession.
26              </p>
27            </div>
28          </div>
29        </div>
30      </div>
31    </div>
32  </div>

```



Gambar 4. 38 Owl Carousel 3 Muse Akademi

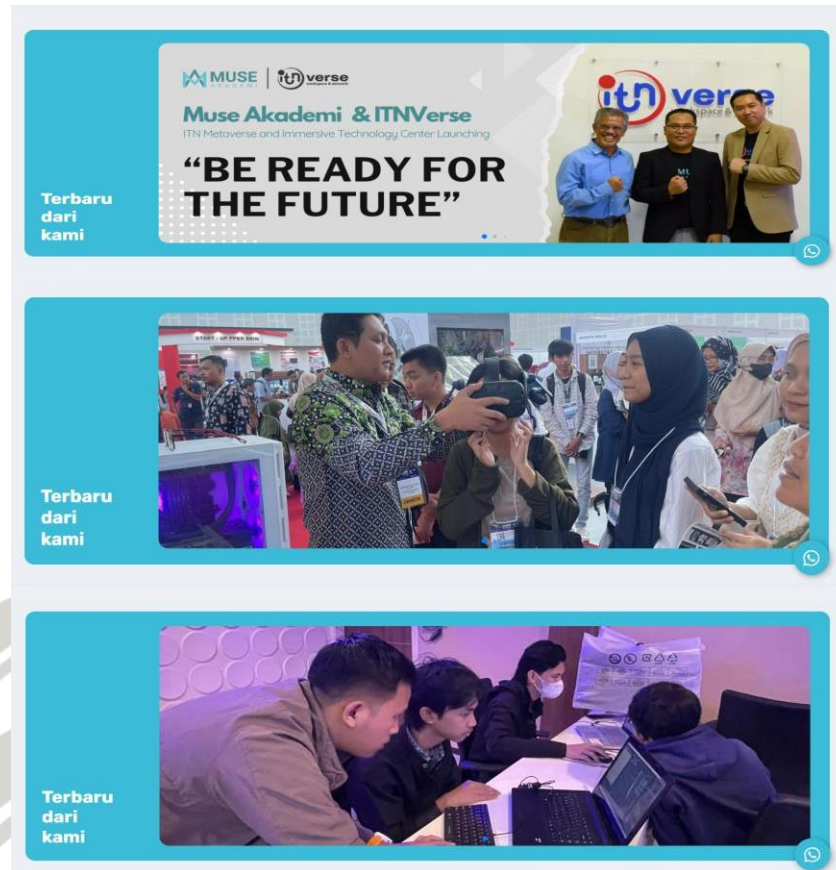
Gambar 4.41 merupakan implementasi *Owl Carousel* di *landing page* Muse Akademi yang bertujuan untuk menampilkan daftar mitra yang bekerja sama dengan Muse Akademi, memberikan elemen visual yang dinamis dan interaktif. Dengan fitur ini, pengguna dapat menggulir melalui berbagai informasi mengenai mitra yang terlibat, membuat tampilan halaman lebih menarik dan memudahkan pengenalan mitra kepada pengguna.

Berikut kode dari *plugin* tersebut:

```

1 <div class="owl-carousel owl-theme" >
2   <div class="agency-card" style="background-
3     image: url({%static 'img/agency1.svg'%})">
4     
7     <h3 class="text-gray">ITNVerse</h3>
8     <p class="text-white">Mitra Kampus
9       Institute Teknologi Nasional Malang dengan
10      fokus Metaverse dan Immersive Technology
11      Training Centre</p>
12   </div>
13   <div class="agency-card" style="background-
14     image: url({%static 'img/agency2.svg'%})">
15     
18     <h3 class="text-gray">UnsamVerse</h3>
19     <p class="text-white">Mitra Kampus
20       Universitas Samudera Aceh dengan fokus
21       Metaverse Learning Innovation Training
22       Centre</p>
23   </div>

```



Gambar 4. 39 Owl Carousel 4 Muse Akademi

Gambar 4. 42 menunjukkan implementasi *Owl Carousel* yang bertujuan untuk menampilkan berita sekilas yang ada di Muse Akademi, memberikan elemen visual yang dinamis dan interaktif. Dengan fitur ini, pengguna dapat menggulir melalui berbagai informasi terbaru mengenai berita dan update dari Muse Akademi, membuat tampilan halaman lebih menarik dan memudahkan akses ke informasi terkini.

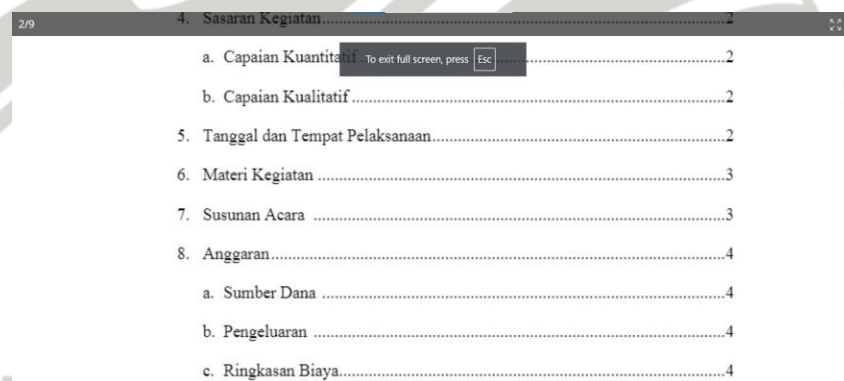
b. *Plugin Pdf.js*

Plugin berbasis *Javascript* ini digunakan untuk menampilkan berkas dalam format .pdf maupun .pptx. *Plugin* ini menyediakan fitur untuk merender seluruh halaman berkas dengan cara menggulir (*scroll*) atau menggunakan tombol *next* dan *prev*. Penulis memilih untuk menggunakan metode render dengan *scroll* ke bawah agar pengguna dapat melihat seluruh halaman berkas

dengan lebih mudah. Selain itu, plugin ini juga menyediakan fungsi tampilan layar penuh (*full screen*) untuk pengalaman membaca yang lebih nyaman. Berikut detail penggunaan Pdf.js:



Gambar 4. 40 Penggunaan pdf.js Muse Akademi



Gambar 4. 41 Pdf.js fitur *fullscreen*

Berikut adalah kode untuk menggunakan *plugin* Pdf.js yang memungkinkan seluruh berkas halaman pdf dirender dengan sempurna dan ditampilkan dalam mode layar penuh:

```
1  const url = "{ session.modul|pdf_modul }";
2  let pdfDoc = null;
3  const pdfPagesContainer =
    document.getElementById('pdf-pages-container');
4  const pageNumberDisplay =
    document.getElementById('page-number');
5
6  const visibleAreas = {};
7
8  const fullscreenButtonNormal =
    document.getElementById('normal-fullscreen-btn');
9  fullscreenButtonNormal.addEventListener('click',
    () => toggleFullscreen('normal'));
```

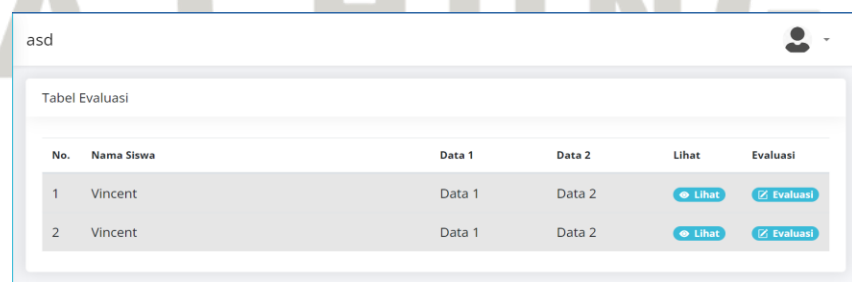
```

10
11 const fullscreenButton =
    document.getElementById('fullscreen-button');
12 fullscreenButton.addEventListener('click', () =>
    toggleFullscreen('fullscreen'));
13
14 const observer = new IntersectionObserver(entries
    => {
15     entries.forEach(entry => {
16 const pageNumber =
    entry.target.getAttribute('data-page-number');
17     if (entry.isIntersecting) {
18         const rect =
    entry.target.getBoundingClientRect();
19         const visibleHeight =
    Math.min(rect.bottom, window.innerHeight) -
    Math.max(rect.top, 0);
20         visibleAreas[pageNumber] =
    visibleHeight;
21
22         updatePageNumberDisplay();
23     } else {
24         delete visibleAreas[pageNumber];
25     }
26     });
27 }, {

```

c. Fungsi untuk tabel yang dapat dilipat (*collapsible*)

Fungsi ini dirancang untuk membuat tabel yang dapat dilipat (*collapsible*), sehingga tabel di dalam tabel utama dapat tampil lebih rapi dan terorganisir. Dengan fitur ini, pengguna dapat melihat konten tambahan dengan menekan tombol "Lihat" yang ada di tabel utama, sehingga tampilan menjadi lebih bersih dan efisien tanpa mengorbankan aksesibilitas informasi. Berikut detail gambar untuk penggunaan fitur ini:



No.	Nama Siswa	Data 1	Data 2	Lihat	Evaluasi
1	Vincent	Data 1	Data 2	Lihat	Evaluasi
2	Vincent	Data 1	Data 2	Lihat	Evaluasi

Gambar 4. 42 Tabel *collapsible* saat dilipat

No.	Nama Siswa	Data 1	Data 2	Lihat	Evaluasi
1	Vincent	Data 1	Data 2	Lihat	Evaluasi

No.	Nama Tugas	Waktu	Nilai
1	Vincent	19 Menit	90
1	Vincent	19 Menit	90
1	Vincent	19 Menit	90

No.	Nama Siswa	Data 1	Data 2	Lihat	Evaluasi
2	Vincent	Data 1	Data 2	Lihat	Evaluasi

Gambar 4. 43 Tabel *collapsible* saat dibuka

Berikut adalah kode untuk membuat tabel *collapsible* di Muse Akademi, memungkinkan tabel utama untuk menampilkan tabel lain di dalamnya, menggunakan plugin *Bootstrap*:

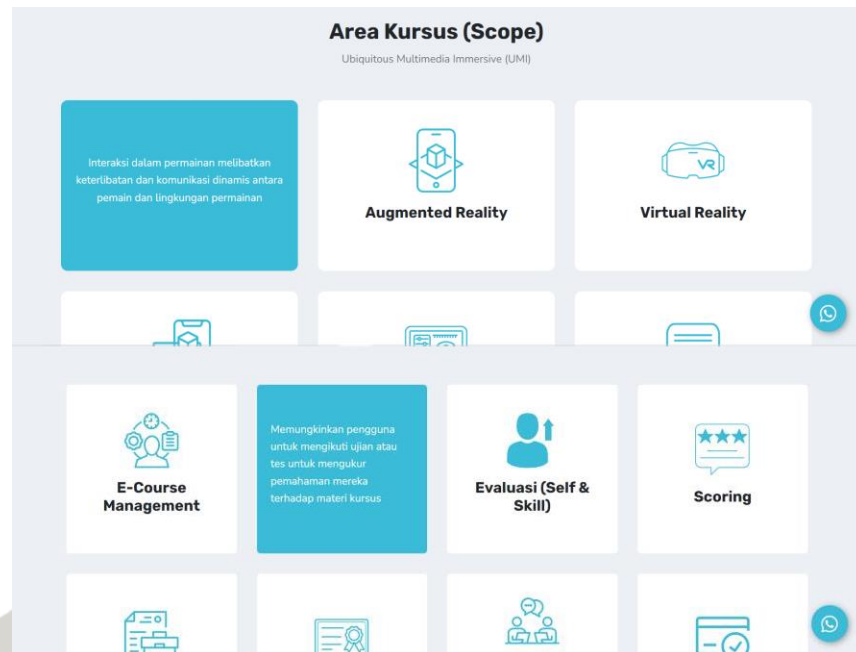
```

1 <tr class="collapse m-0" style="background-color:
  white !important;" id="collapse">
2   <td colspan="6" class="p-0">
3     <div class="table-responsive m-0">
4       <table class="table m-0 table-collapse">
5         <thead>
6           <tr class="small">
7             <th width="5%">No.</th>
8             <th width="44%">Nama Tugas</th>
9             <th width="26%">Waktu</th>
10            <th width="25%">Nilai</th>
11          </tr>
12        </thead>

```

d. Fungsi untuk menampilkan deskripsi teks saat ditekan

Fungsi ini dirancang untuk menampilkan deskripsi teks yang tersembunyi hanya ketika ditekan. Dengan fitur ini, tampilan halaman menjadi lebih rapi dan terorganisir, karena deskripsi tambahan hanya muncul saat diperlukan, menjaga agar antarmuka tetap bersih dan fokus pada informasi utama. Berikut detail penggunaan fitur ini:



Gambar 4. 44 Fungsi menampilkan deskripsi saat ditekan

Berikut adalah kode yang menggunakan CSS dan JavaScript untuk menampilkan deskripsi penjelasan hanya ketika ditekan dan menutup deskripsi ketika kursor keluar dari area kotak:

```

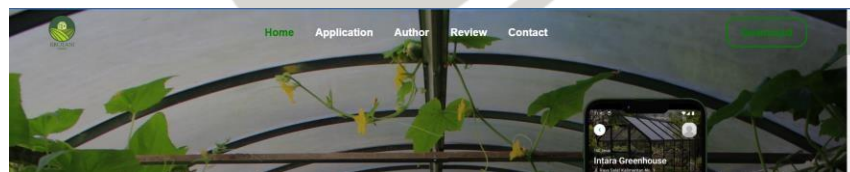
1 document.addEventListener('DOMContentLoaded',
  function() {
2     const cardFeatures =
      document.querySelectorAll('.card-feature');
3
4     cardFeatures.forEach(cardFeature => {
5         const cardDesc =
          cardFeature.querySelector('.card-feature-
            desc');
6
7         cardFeature.addEventListener('click',
          function() {
8             cardDesc.style.transform =
              'translateY(0)';
9         });
10
11        cardFeature.addEventListener('mouseleave',
          function() {
12            cardDesc.style.transform =
              'translateY(250px)';
13        });
14    });
15 });

```

4.4.2 Brotani

a. Animasi

Di dalam *website* Brotani, penulis memanfaatkan animasi berbasis CSS secara ekstensif untuk meningkatkan pengalaman pengguna. Sistem animasi *timeline* digunakan untuk efek *scroll*, di mana elemen-elemen halaman dianimasikan secara dinamis saat pengguna menggulir. Selain itu, animasi dengan sistem *translate* diterapkan untuk memberikan efek visual yang menarik saat halaman *discroll*. Penulis juga menerapkan animasi *hover* untuk mempercantik desain *website* dan memberikan interaksi visual yang lebih menarik kepada pengguna. Selain itu, terdapat juga animasi yang muncul saat halaman dimuat, menambah dinamika visual ketika pengguna pertama kali mengakses situs. Semua elemen animasi ini dirancang untuk menciptakan pengalaman pengguna yang lebih interaktif dan estetis. Berikut beberapa detail animasi yang penulis kembangkan:



Gambar 4. 45 Animasi hover tulisan

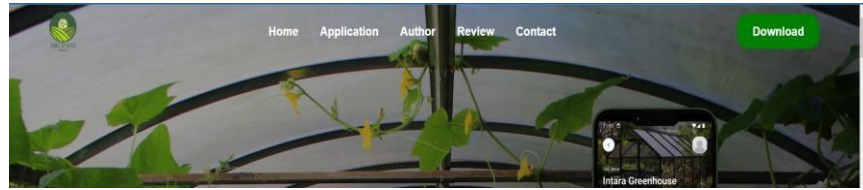
Pada gambar 4.48, ditampilkan animasi yang menggunakan CSS untuk mengubah warna tulisan saat pengguna mengarahkan kursor ke tulisan tersebut. Dengan menggunakan efek *hover* CSS, animasi ini memberikan interaksi visual yang menarik dan responsif. Berikut adalah kode yang menggunakan CSS untuk mengubah warna tulisan ketika kursor diarahkan ke elemen tersebut:

```
1 .nav-hover {  
2     color: white;  
3     transition: color 0.3s;  
4     text-decoration: none;  
5     padding: 0 1em 0 1em;  
6     align-items: center ;  
7 }  
8  
9 .nav-hover:hover {
```

```

10     color: green;
11 }

```



Gambar 4. 46 Animasi hover tombol

Pada gambar 4. 49, ditampilkan animasi untuk efek tombol ketika kursor diarahkan, fungsi ini dibuat untuk mengganti warna latar belakang tombol menjadi warna hijau. Berikut kode CSS untuk membuat animasi tersebut:

```

1  .button-download::before {
2      content: '';
3      width: 0;
4      height: 300%;
5      position: absolute;
6      top: 50%;
7      left: 50%;
8      transform: translate(-50%, -50%)
9          rotate(45deg);
9      background: green;
10     transition: .5s ease;
11     display: block;
12     z-index: -1;
13 }
14
15 .button-download:hover::before {
16     width: 105%;
17 }
18
19 .button-download:hover {
20     color: white;
21 }

```



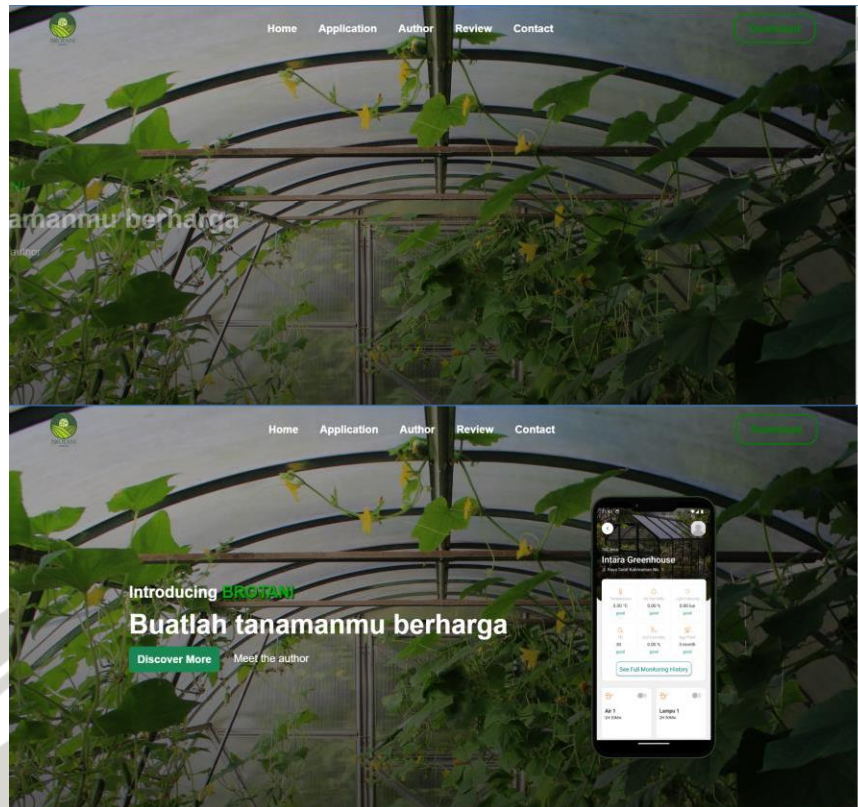
Gambar 4. 47 Animasi sebelum *scroll* (atas) dan sesudah *scroll* (bawah)

Pada gambar 4.50, ditampilkan animasi yang menggunakan sistem CSS untuk menampilkan teks saat pengguna menggulir halaman ke bawah. Animasi ini memanfaatkan properti *opacity* dan *transform* untuk memberikan efek transisi yang halus. Berikut kode CSS untuk membuatnya:

```

1  .fade-up{
2      animation: fade-up 0.3s ease forwards;
3      animation-range: 10% cover 10% entry;
4      animation-timeline: view();
5  }
6
7  @keyframes fade-up{
8      from{
9          transform: translateY(100%);
10         opacity: 0;
11     }to{
12         transform: translateY(0);
13         opacity: 1;
14     }
15 }
16

```



Gambar 4. 48 Animasi halaman dimuat

Pada gambar 4. 51, ditampilkan animasi yang mengatur tampilan elemen di dalam area *cover* sebelum dan setelah halaman dimuat. Animasi ini menggunakan properti CSS *translate* dan *opacity* untuk membuat elemen muncul dengan efek transisi yang halus. Sebelum halaman dimuat sepenuhnya, elemen tersebut tidak akan terlihat, dan setelah halaman selesai dimuat, elemen akan muncul dengan efek perpindahan dan perubahan transparansi. Berikut kode untuk membuat animasinya:

```

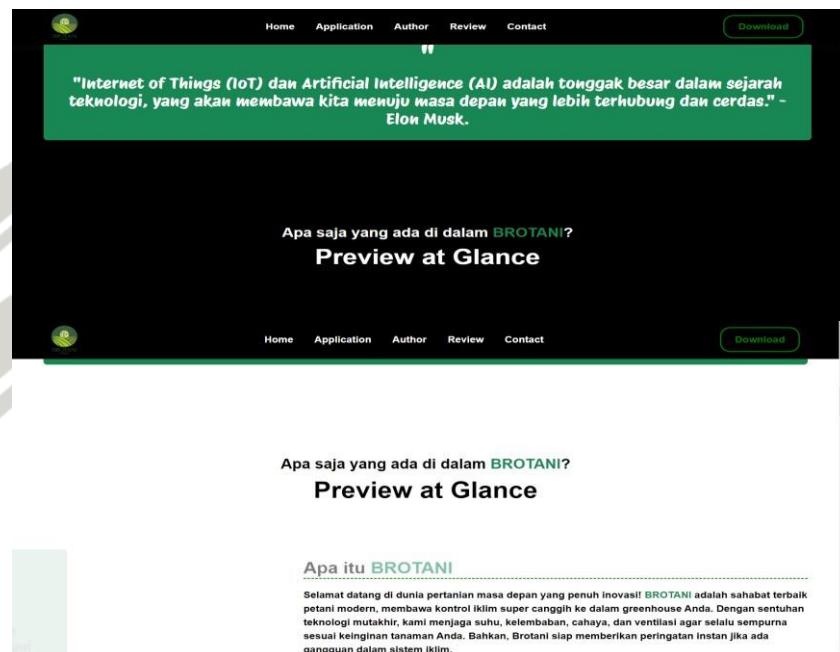
1  .fade-in-left-page{
2      opacity: 0;
3      animation: fade-in-left 0.8s ease-in-out
4      forwards;
5  }

```

- b. Fungsi untuk pergantian warna background secara dinamis saat *discroll*

Fungsi ini dikembangkan menggunakan *Javascript*, dengan menambahkan batasan *scroll* pada halaman. Ketika halaman *discroll* secara vertikal melewati tinggi layar perangkat, latar

belakang halaman yang awalnya berwarna hitam akan berubah menjadi warna putih. Perubahan warna ini dirancang untuk meningkatkan estetika dan keindahan visual website Brotani, memberikan efek transisi yang halus dan menarik saat pengguna menjelajahi konten. Fungsi ini tidak hanya menambah daya tarik visual tetapi juga menciptakan pengalaman pengguna yang lebih menyenangkan dan dinamis. Berikut detail dari implementasi fungsi ini dan kodenya:



Gambar 4. 49 Transisi warna sebelum *scroll* (atas) dan sesudah *scroll* (bawah)

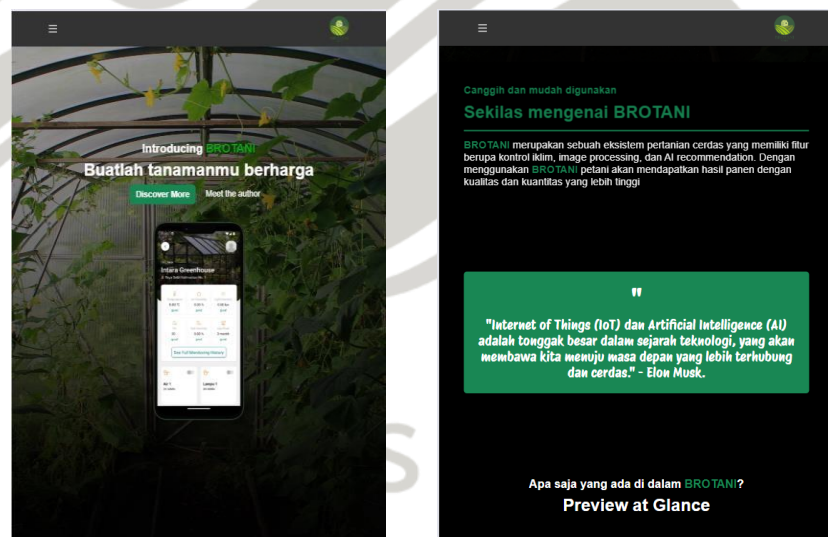
```

1  window.addEventListener("scroll", function () {
2      var scrollTopPosition = window.pageY;
3      var changingColorDiv =
4      document.querySelector(".changing-color");
5      var offset = changingColorDiv.offsetTop -
6      window.innerHeight + 350;
7
8      if (scrollTopPosition > offset) {
9          document.body.style.backgroundColor =
10         "white";
11         document.body.style.color = "black";
12     } else {
13         document.body.style.backgroundColor =
14         "black";
15         document.body.style.color = "white";
16     }
17 }

```

c. Pembuatan sistem desain yang responsif

Penulis mengembangkan sistem desain responsif untuk memastikan bahwa website dapat diakses dengan optimal di berbagai perangkat. Sistem desain ini mengintegrasikan elemen dari *framework Bootstrap* untuk memanfaatkan komponen dan *grid* yang sudah ada, serta menerapkan kustomisasi khusus untuk menyesuaikan kebutuhan dan gaya visual yang unik. Kombinasi ini memungkinkan desain untuk beradaptasi dengan mulus pada berbagai ukuran layar, dari desktop hingga perangkat mobile, memastikan pengalaman pengguna yang konsisten dan memuaskan di seluruh platform. Berikut sedikit tampilan *website* saat diakses di perangkat seluler:

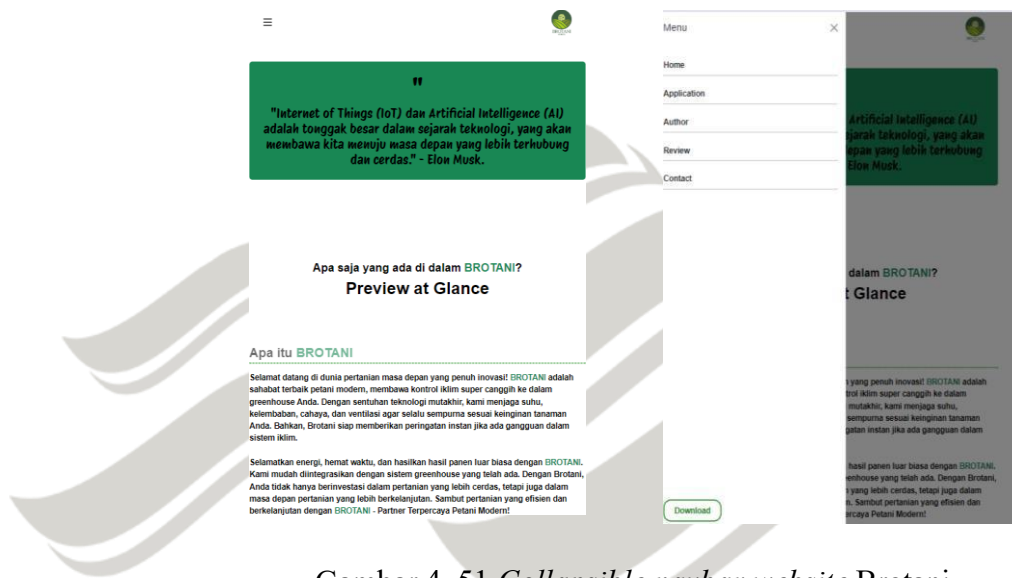


Gambar 4. 50 Desain responsif *website* Brotani

d. Sistem navbar yang *collapsible* saat di perangkat seluler

Penulis mengembangkan sistem *navbar* yang dapat dilipat (*collapsible*) untuk meningkatkan pengalaman pengguna saat mengakses website di perangkat seluler. Fitur *collapsible* ini memanfaatkan *framework Bootstrap* untuk menyediakan fungsionalitas dasar, sementara desain dan isi navbar dikustomisasi secara khusus untuk memenuhi kebutuhan estetika dan fungsional website. *Navbar* ini dapat dibuka dengan menekan tombol

hamburger menu di pojok kiri atas, memungkinkan pengguna untuk mengakses menu navigasi dengan mudah dan efisien. Dengan pendekatan ini, *navbar* dapat menyesuaikan tampilannya dengan perangkat yang digunakan, memastikan navigasi yang intuitif dan aksesibilitas yang optimal tanpa mengorbankan gaya visual. Berikut detail dari fungsi *navbar* tersebut:



Gambar 4. 51 *Collapsible navbar website Brotani*

Berikut adalah kode untuk membuat *navbar* yang *collapsible* menggunakan *plugin Bootstrap*. Kode ini memanfaatkan komponen *navbar* dari *Bootstrap*, yang memungkinkan tampilan *navbar* yang responsif dan dapat di-*collapse* pada perangkat dengan layar kecil.

```

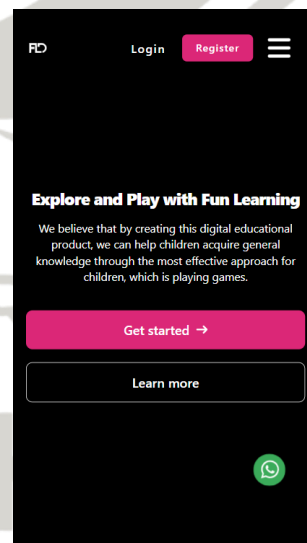
1 <div class="collapse navbar-collapse"
  id="navbarTogglerDemo03">
2   <ul class="navbar-nav mx-auto mb-2 mb-lg-0">
3     <li class="nav-item">
4       <a class="nav-hover fw-semibold"
        href="#">Home</a>
5     </li>
6     <li class="nav-item">
7       <a class="nav-hover fw-semibold"
        href="#">Application</a>
8     </li>
9     <li class="nav-item">
10      <a class="nav-hover fw-semibold"
        href="#">Author</a>
11    </li>

```

4.4.3 Fun Learning Digital (FLD)

a. Desain yang responsif

Di dalam website Fun Learning Digital, penulis menerapkan sistem desain responsif menggunakan Tailwind CSS, dengan fokus pada penyesuaian minimal untuk meningkatkan tampilan di berbagai perangkat. Penulis memperbaiki beberapa elemen yang kurang optimal, seperti elemen yang tidak menyusut dengan baik di layar kecil atau tampilan yang tidak teratur pada resolusi tinggi. Penyesuaian ini mencakup penataan ulang elemen, penggunaan media queries yang tepat, serta penyesuaian margin dan padding untuk memastikan layout yang konsisten. Meskipun perubahan tidak signifikan, perbaikan ini memastikan desain tetap responsif dan fungsional di semua ukuran layar, memberikan pengalaman pengguna yang konsisten dan memuaskan baik di perangkat mobile maupun desktop.

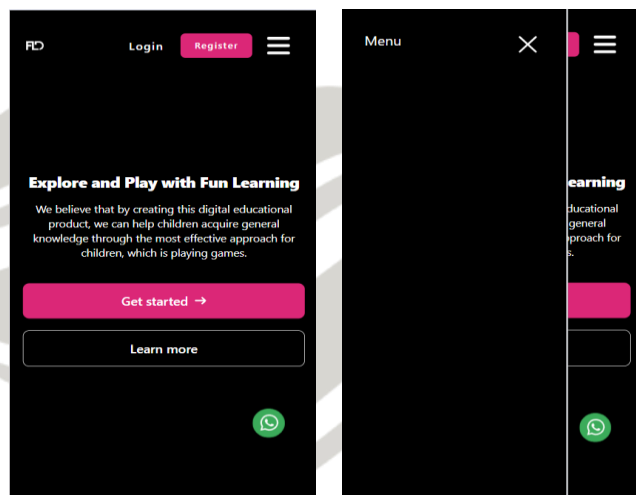


Gambar 4. 52 Desain responsif *website* Fun Learning Digital

b. Sistem navbar yang *collapsible* saat di perangkat seluler

Penulis mengembangkan sistem navbar yang dapat dilipat (*collapsible*) untuk meningkatkan pengalaman pengguna saat mengakses website di perangkat seluler. Fitur *collapsible* ini memanfaatkan *Tailwind* CSS untuk menyediakan fungsionalitas dasar. Namun, desain dari isi navbar belum selesai karena proyek

dipindahkan ke tim lain. Navbar ini dapat dibuka dengan menekan tombol hamburger menu di pojok kiri atas, memungkinkan pengguna untuk mengakses menu navigasi dengan mudah dan efisien. Dengan pendekatan ini, navbar dapat menyesuaikan tampilannya dengan perangkat yang digunakan, memastikan navigasi yang intuitif dan aksesibilitas yang optimal tanpa mengorbankan gaya visual. Berikut detail dari fungsi navbar tersebut dan kodenya:



Gambar 4. 53 *Collapsible navbar website Fun Learning Digital*

Berikut adalah kode untuk membuat navbar yang collapsible menggunakan Tailwind CSS. Kode ini memanfaatkan utilitas Tailwind untuk membuat navbar yang responsif dan dapat di-collapse pada perangkat dengan layar kecil.

```

1 <div id="offcanvas" class="top-0 left-0 w-3/4 h-
  screen absolute bg-black border-r-2 border-gray
  hidden z-50 transition-transform duration-300
  ease-in">
2   <div class="p-7 text-lg w-100 absolute top-0
    left-0 transform -translate-x-full relative"
    id="offcanvasContent">
3     <!-- Content goes here -->
4     <p class="text-white">Menu</p>
5     <button type="button"
      onclick="toggleOffcanvas()" class="absolute
      right-0 top-0 p-7">
6       <svg xmlns="http://www.w3.org/2000/svg"
        fill="white" viewBox="0 0 24 24" stroke-
        width="1.5" stroke="white" class="w-10 h-
        10">
7       <path stroke-linecap="round" stroke-

```

```

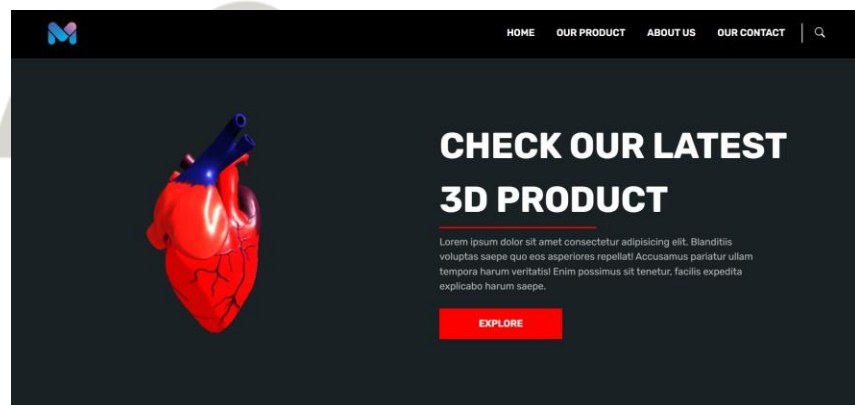
        linejoin="round" d="M6 18 18 6M6 6 12 12"
      /></svg>
8      </button>
9      </div>
10 </div>

```

4.4.4 Museverse

a. Three.js

Penulis menggunakan *plugin Three.js* untuk merender objek tiga dimensi yang akan diimplementasikan ke dalam *website* Museverse. *Plugin* berbasis *Javascript* ini memerlukan kustomisasi mendalam dalam berbagai aspek, termasuk ukuran objek, tata letak (pengaturan koordinat), pencahayaan, dan elemen visual lainnya. Penulis juga menambahkan animasi agar objek tersebut berputar secara terus-menerus, menciptakan efek dinamis yang menarik. Pengguna dapat berinteraksi dengan objek melalui beberapa metode: menggeser objek dengan menahan klik kanan pada *mouse*, seperti pada gambar 4. 71, merotasi objek dengan menahan klik kiri pada *mouse*, seperti pada gambar 4. 72, dan memperbesar atau memperkecil objek menggunakan *scroll up* dan *down* pada *mouse*, seperti pada gambar 4. 73. Fitur-fitur ini dirancang untuk meningkatkan keterlibatan pengguna dan memberikan pengalaman interaktif yang menyenangkan. Berikut detail fungsi interaksi objek 3 dimensi dari *plugin Three.js*:

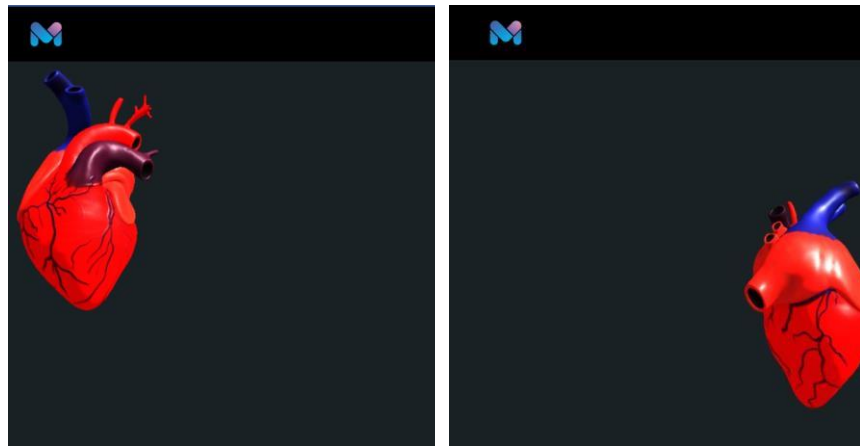


Gambar 4. 54 Tampilan objek 3 dimensi Museverse

Berikut adalah kode lengkap berbasis *Javascript* untuk merender objek 3 dimensi menggunakan *plugin Three.js*. Kode ini

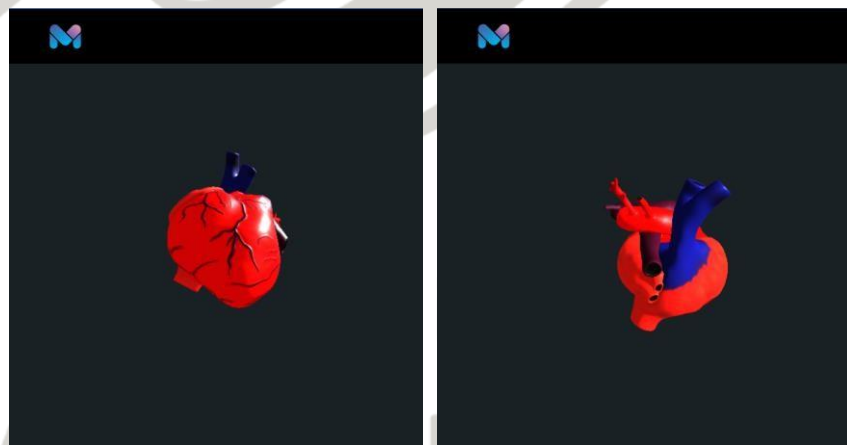
mencakup pengaturan *scene*, kamera, renderer, pencahayaan, serta pemuatan objek 3D dari file GLTF menggunakan *GLTFLoader*. Dalam kode ini, *scene* dan kamera diinisialisasi untuk mengatur sudut pandang dan jarak pandang pengguna terhadap objek 3D. Pencahayaan juga diatur dengan menambahkan *directional light* dengan intensitas yang tinggi untuk memberikan efek pencahayaan yang lebih realistis pada objek.

```
1  const scene = new THREE.Scene();
2  const camera = new THREE.PerspectiveCamera(50,
    container.clientWidth / container.clientHeight, 0.1,
    1000);
3  let object;
4  let controls;
5
6  const topLight = new
    THREE.DirectionalLight(0xffffff, 3); // Increase
    intensity
7  topLight.position.set(100, 100, 50);
8  topLight.castShadow = true;
9  scene.add(topLight);
10
11 const loader = new GLTFLoader();
12
13 loader.load(
14     './3d/heart.glb',
15     function (gltf) {
16         object = gltf.scene;
17         object.position.y = -7;
18         scene.add(object);
19
20         const boundingBox = new
21             THREE.Box3().setFromObject(object);
22         const center = boundingBox.getCenter(new
23             THREE.Vector3());
24         const size = boundingBox.getSize(new
25             THREE.Vector3());
26         const maxDim = Math.max(size.x, size.y,
27             size.z);
28
29         const distance = maxDim / Math.tan(camera.fov
30             * Math.PI / 360);
31         camera.position.copy(center);
32         camera.position.z = center.z + distance;
33
34         camera.lookAt(center);
```



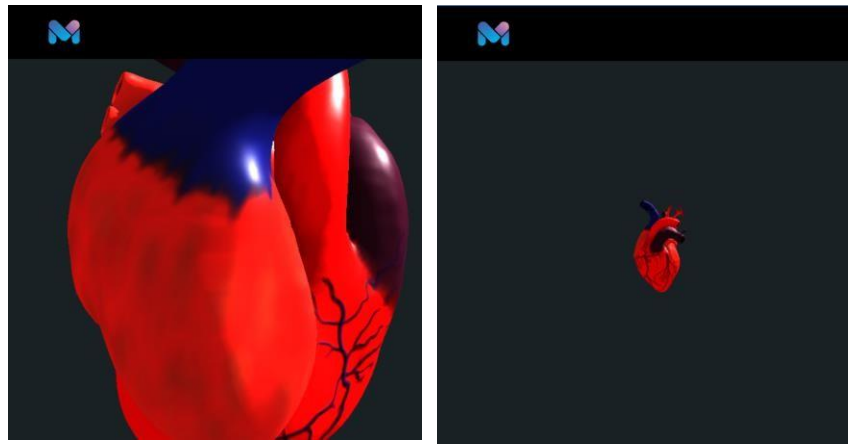
Gambar 4. 55 Interaksi geser objek 3 dimensi Museverse

Pada Gambar 4. 58, objek 3 dimensi dapat diinteraksikan menggunakan klik kanan pada mouse untuk menggeser tampilan. Interaksi ini memungkinkan pengguna untuk mengubah sudut pandang terhadap objek tersebut tanpa mengubah posisi objek itu sendiri.



Gambar 4. 56 Interaksi rotasi objek 3 dimensi Museverse

Pada Gambar 4. 59, objek 3 dimensi dapat diinteraksikan menggunakan klik kiri pada mouse untuk merotasi tampilan. Fitur rotasi ini memungkinkan pengguna untuk memutar objek sehingga dapat melihatnya dari berbagai sudut pandang, tanpa mengubah posisi objek dalam scene.



Gambar 4. 57 Interaksi perbesar/perkecil objek 3 dimensi Museverse

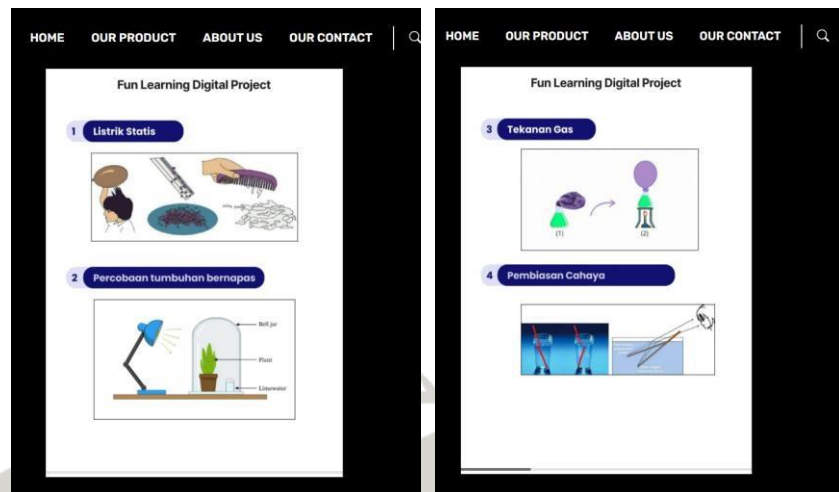
Kode ini menggunakan *OrbitControls* untuk memungkinkan pengguna memutar, memperbesar, dan memperkecil tampilan objek dengan mudah. *OrbitControls* mengubah posisi kamera sesuai dengan input pengguna, sehingga kamera bergerak di sekitar objek, memberikan efek interaksi yang dinamis dan responsif. Berikut adalah kode untuk membuat objek 3 dimensi dapat diinteraksi menggunakan Three.js:

```
1  if (controls) {
2      controls.dispose();
3  }
4
5  controls = new OrbitControls(camera,
6      renderer.domElement);
7  controls.target.copy(center);
8
9  controls.update();
```

b. Swiper.js

Penulis menggunakan *plugin Swiper.js* untuk menyediakan sistem geser gambar di *website Museverse*. Plugin berbasis *Javascript* ini memungkinkan pengguna untuk menggeser gambar dengan mulus, memberikan tampilan yang lebih rapi dan menarik pada *website*. Fitur geser gambar ini dirancang untuk meningkatkan pengalaman visual dan interaktivitas, memastikan bahwa konten gambar dapat diakses dengan mudah dan dengan

gaya yang profesional. Berikut detail dari berbagai fungsi *Swiper.js*:



Gambar 4. 58 *Plugin Swiper.js* Museverse

Berikut adalah kode untuk membuat *carousel* menggunakan *plugin Swiper.js*, yang merupakan salah satu *plugin Javascript* yang populer untuk membuat *slider* dan *carousel* yang responsif dan interaktif.

```

1 <div class="swiper-wrapper">
2   <!-- Slides -->
3   <div class="swiper-slide"
4     style="background-image:
5       url('./document/fld-1.jpg') "></div>
6   <div class="swiper-slide" style="background-
7     image: url('./document/fld-2.jpg') "></div>
8   <div class="swiper-slide" style="background-
9     image: url('./document/fld-3.jpg') "></div>
10  <div class="swiper-slide" style="background-
11    image: url('./document/fld-4.jpg') "></div>
12  <div class="swiper-slide" style="background-
13    image: url('./document/fld-5.jpg') "></div>
14  <div class="swiper-slide" style="background-
15    image: url('./document/fld-6.jpg') "></div>
16 </div>
17 <div class="swiper-scrollbar"></div>

```

c. Fungsi pergantian tab

Penulis menggunakan sistem pergantian tab untuk menghindari pembuatan halaman baru, memungkinkan pengguna untuk mengakses berbagai tab konten dalam satu halaman. Sistem ini memanfaatkan *Javascript* sebagai mekanisme toggle untuk mengganti antara tab-tab yang berbeda. Dengan cara ini, pengguna

dapat dengan mudah berpindah antar tab tanpa perlu memuat ulang halaman, meningkatkan efisiensi dan pengalaman interaktif situs web. Pendekatan ini dirancang untuk menjaga tampilan yang bersih dan terorganisir, sambil menyediakan akses cepat dan langsung ke berbagai informasi. Berikut detail fungsi dari pergantian tab:

Gambar 4. 59 Sistem pergantian tab Museverse

Sistem ini memungkinkan pengguna untuk beralih antara berbagai konten yang disembunyikan di bawah tab berbeda. Berikut adalah kode untuk membuat sistem pergantian tab menggunakan *Javascript*:

```

1  const trainingForm =
    document.getElementById("training-form");
2  const productForm =
    document.getElementById("product-form");
3  const checkbox =
    document.getElementById('status');
4      function toggleFormVisibility() {
5          if (checkbox.checked) {
6              trainingForm.style.display = 'none';
7              productForm.style.display = 'block';
8          } else {
9              trainingForm.style.display = 'block';
10             productForm.style.display = 'none';
11         }
12     }
13     toggleFormVisibility();
14     checkbox.addEventListener('change',
        toggleFormVisibility);

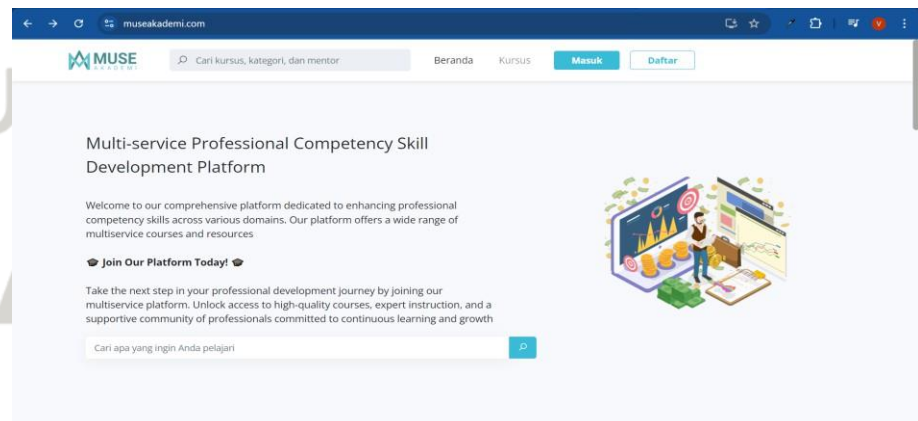
```

4.5 Implementasi dan Dokumentasi

Dari keempat *website* yang penulis kembangkan, dua diantaranya telah diunggah ke server, yaitu *website* Muse Akademi dan Museverse. *Website* Muse Akademi dapat diakses di www.museakademi.com, sementara *website* Museverse telah diunggah ke server yang sama dengan Muse Akademi yaitu www.museakademi.com/Museverse. Proses *hosting* untuk kedua situs ini dilakukan oleh tim *backend* dengan menggunakan layanan *hosting* dari Domainsia. Hal ini memastikan bahwa kedua *website* dapat diakses secara online dan berfungsi sebagaimana mestinya.

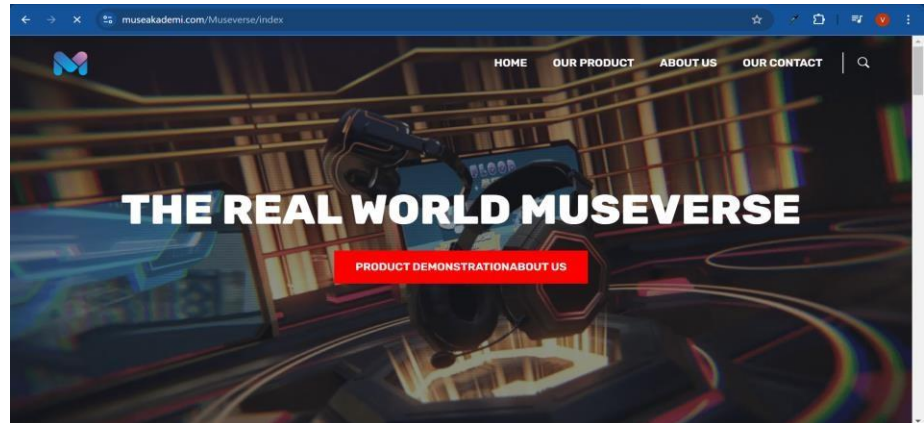
Sementara itu, *website* Brotani telah dihentikan karena belum siap untuk diunggah ke server dan memerlukan perbaikan lebih lanjut sebelum dapat dipublikasikan. Untuk *website* Fun Learning Digital, penulis belum menerima informasi dari tim *backend* mengenai status *hostingnya*. Status *hosting website* ini masih belum jelas, dan penulis menunggu konfirmasi lebih lanjut dari tim terkait. Berikut adalah gambar dokumentasi *hosting* untuk *website* yang telah diunggah ke server:

4.5.1 Muse Akademi



Gambar 4. 60 Dokumentasi *hosting* Muse Akademi

4.5.2 Museverse



Gambar 4. 61 Dokumentasi *hosting* Museverse

UNIVERSITAS
MA CHUNG

BAB 5 PENUTUP

5.1 Kesimpulan

Dari hasil laporan yang telah dibuat, dapat disimpulkan bahwa:

- Penulis telah menyelesaikan kegiatan PKL selama 6 bulan di Muse Akademi, yang terletak di Jl. Selat Karimata No.E6/1, Lesanpuro, Kec. Kedungkandang, Kota Malang, Jawa Timur, dari 1 Februari hingga 1 Agustus.
- Selama masa kegiatan PKL ini, penulis berhasil mengimplementasikan berbagai teknologi *front-end* dalam proyek pengembangan *website* di Muse Akademi. Berikut detail fitur yang penulis implementasikan selama masa kegiatan PKL di Muse Akademi:
 - a. Sudah berhasil diimplementasikan desain yang responsif untuk semua *website* yang dikembangkan di Muse Akademi. Ini termasuk pengoptimalan tampilan *website* agar tetap baik di berbagai perangkat, mulai dari *desktop* hingga *mobile*.
 - b. Sudah berhasil diimplementasikan *plugin Owl Carousel* dan *Swiper.js* pada beberapa *website*, seperti Muse Akademi dan Museverse. Kedua *plugin* ini memberikan pengalaman visual yang dinamis dengan tampilan slide yang menarik dan interaktif.
 - c. Sudah berhasil dimanfaatkan teknologi *Three.js* untuk menciptakan interaksi visual objek 3D pada *website* Museverse. Hal ini meningkatkan daya tarik konten dengan animasi dan efek visual yang lebih menarik bagi pengguna.
 - d. Sudah diterapkan *framework Bootstrap* secara efektif untuk mengembangkan tata letak yang responsif dan konsisten di *website* Muse Akademi, Brotani, dan Museverse. *Framework* ini membantu meningkatkan efisiensi pengembangan serta kualitas keseluruhan tampilan *website*.
 - e. Sudah berhasil diimplementasikan *framework Tailwind* untuk desain *website* Fun Learning Digital (FLD). Dengan pendekatan

yang lebih modular dalam pengaturan gaya dan tata letak, *Tailwind* memungkinkan penulis untuk menghasilkan desain yang konsisten dan efisien.

- Penulis telah menerapkan teknologi seperti HTML, CSS, *Javascript*, dan berbagai macam *plugin* berbasis *Javascript*.
- Penulis terlibat aktif dalam pengembangan situs web Muse Akademi, Brotani, Museverse, dan Fun Learning Digital (FLD), dengan fokus utama pada peningkatan antarmuka pengguna dan pengalaman pengguna.

5.2 Saran

Penulis memberikan beberapa saran untuk perusahaan Muse Akademi yaitu:

- Penting untuk menyusun *Standard Operating Procedure* (SOP) yang jelas dalam pengembangan website. Menyusun SOP yang memisahkan tanggung jawab antara implementasi teknologi *front-end* dan desain UI/UX oleh tim khusus dapat meningkatkan efisiensi dan kualitas proyek.
- Perjelas instruksi terkait isi *copywriting* dalam proses pengembangan. Memberikan panduan rinci tentang konten yang diperlukan dapat memastikan bahwa *copywriting* sesuai dengan desain *front-end* dan memenuhi kebutuhan konten dengan tepat. Ini akan membantu dalam meningkatkan koordinasi dan memastikan bahwa sistem *front-end* menyampaikan pesan yang diinginkan.

DAFTAR PUSTAKA

- Arhandi, P. P., Arief, S. N. & Firdausi, A. T., 2022. Pengembangan Website Pendukung Mastery Based Learning untuk Pembelajaran Mahasiswa. *Jurnal Informatika Polinema*, 9(1), 51-58.
- Hidayatulloh, S., 2015. Optimalisasi Github untuk Software Project Management dengan Memanfaatkan Notifikasi SMS. *Jurnal Informatika*, 2(1), 198-204.
- Mariko, S., 2019. Aplikasi Website Berbasis HTML dan Javascript untuk Menyelekaikan Fungsi Integral pada Mata Kuliah Kalkulus. *Jurnal Inovasi Teknologi Pendidikan*, 6(1), 80-91.
- Ningsih, K. S., Aruan, N. J. & Siahaan, A. T. A. A., 2022. Aplikasi Buku Tamu Menggunakan Fitur Kamera dan Ajax Berbasis Website pada Kantor Dispora Kota Medan. *SITek: Jurnal Sains, Informatika, dan Teknologi*, 1(3), 94-99.
- Permatasari, A. & Suhendi, 2020. Rancang Bangun Sistem Informasi Pengelolaan Talent Film berbasis Aplikasi Web. *Jurnal Informatika Terpadu*, 6(1), 29-37.
- Rahimah, A. N., Rusdianto, D. S. & Ananta, M. T., 2019. Pengembangan Sistem Pengelolaan Ruang Baca Berbasis Web dengan Menggunakan Django Framework (Studi Kasus: Ruang Baca Fakultas Ilmu Komputer Universitas Brawijaya). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(5), 4439-4446.
- Ramdhan, M. S. & Rosyidah, U. A., n.d. Analisis Penggabungan (Stitching) Foto dan Menampilkan Virtual Tour Panorama 360 Derajat Menggunakan Plugin Three.js. p. 2017.
- Riasinir, T. J. & Widyasari, 2019. Pemanfaatan Framework Bootstrap Dalam Merancang Website Responsif Untuk Toko D2 Adventure. *Jurnal ENTER*, Volume 2, 346-355.

Sabita, H., Herwanto, R., Syafitri, Y. & Prasetyo, B. D., 2022. Pengembangan Aplikasi Akreditasi Program Studi Berbasis Framework Django. Jurnal Informatika, 22(1), 33-37.

Summit, S., 2023. Memahami Dasar-Dasar HTML dan CSS: Fondasi Pengembangan Web. Teknologiterkini.org, 3(9), 1-19.

Suprayogi, B. & Rahmanesa, A., 2019. Penerapan Framework Bootstrap dalam Sistem Informasi Pendidikan SMA Negeri 1 Pacet Cianjur Jawa Barat. Jurnal Teknologi Informasi dan Komunikasi, 6(2), 119-127.

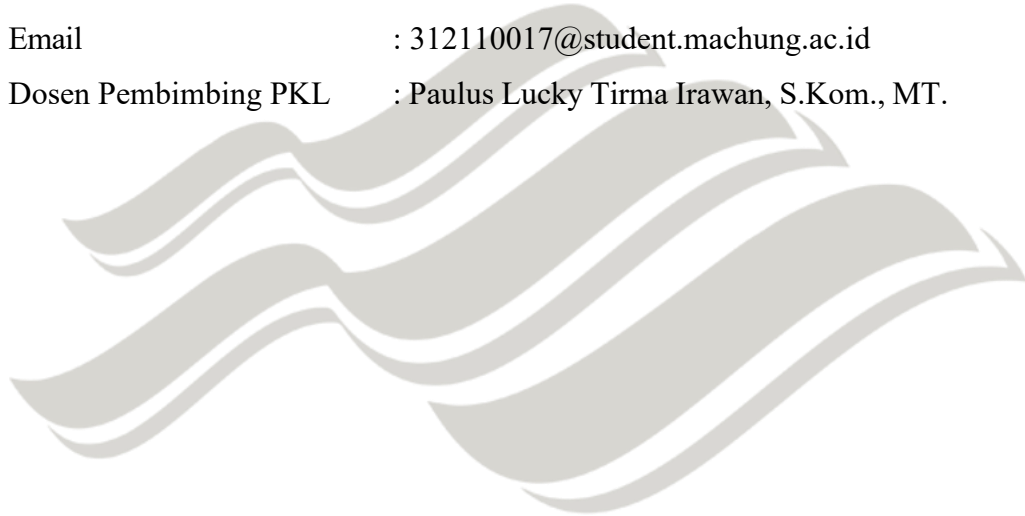


UNIVERSITAS
MA CHUNG

LAMPIRAN

Biodata Mahasiswa

Nama : Vincent Dwi Hartanto
NIM : 312110017
Program Studi : Teknik Informatika
Tempat & tanggal lahir : Surabaya, 07 November 2003
No HP : 082264725149
Email : 312110017@student.machung.ac.id
Dosen Pembimbing PKL : Paulus Lucky Tirma Irawan, S.Kom., MT.



UNIVERSITAS
MA CHUNG

FORM PKL_FTD03

LEMBAR PENILAIAN EKSTERNAL PRAKTIK KERJA LAPANGAN (PKL)

Nama Mahasiswa	:	Vincent Dwi Hartanto
NIM	:	312110017
Program Studi	:	Teknik Informatika
Pembimbing Lapangan	:	Syahrieza Ilham Noor Fauzi
Tanggal Pelaksanaan	:	1 Februari 2024
Tempat PKL	:	Muse Akademi, Jl. Selat Karimata No.E6/1, Lesanpuro, Kec. Kedungkandang, Kota Malang, Jawa Timur
Topik PKL	:	Implementasi Teknologi Front End dalam Proyek Pengembangan Website

Komponen Penilaian :

No	Komponen Penilaian	Nilai Angka (0-100)
1	Disiplin dalam kehadiran (presensi)	95
2	Kesungguhan dalam melakukan praktik kerja	95
3	Disiplin dalam pekerjaan	96
4	Kemampuan memecahkan masalah	100
5	Tanggung jawab	96
6	Kemauan untuk mengetahui hal-hal yang ada di tempat praktik kerja	96
7	Pengetahuan tentang ilmu yang dilaksanakan	94
8	Keterampilan	97
9	Kemampuan menyampaikan pendapat	90
10	Kemampuan dalam bekerja sama	98
Rerata (Jumlah Nilai : 10 komponen)		95.7

Rentang Nilai:

MUTU	RENTANG	BOBOT NILAI
A	95,00 s/d 100	4,00
AB	90,00 s/d 94,99	3,70
BA	82,00 s/d 89,99	3,30
B	73,00 s/d 81,99	3,00
BC	65,00 s/d 72,99	2,70
CB	60,00 s/d 64,99	2,30
C	56,00 s/d 59,99	2,00
CD	50,00 s/d 55,99	1,50
D	40,00 s/d 49,99	1,00
E	00,00 s/d 39,99	0,00

Senin, 8 Agustus 2024

Pembimbing Lapangan*



Syahrieza Ilham Noor Fauzi
(Co-CEO)



"EXCELLENCE THROUGH COMPETENCY"