

LAPORAN PRAKTIK KERJA LAPANGAN
PEMBUATAN BACKEND PADA WEBSITE NAZMA
OFFICE MENGGUNAKAN NODEJS DENGAN
EXPRESS

Diajukan untuk memenuhi persyaratan kelulusan
Matakuliah IF4004 Praktik Kerja Lapangan

Oleh:

Geizka Rozilia Ruicosta / 119140114



PROGRAM STUDI TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI SUMATERA
2022


Lembar Pengesahan Program Studi Teknik Informatika
Pembuatan Backend Website Menggunakan NodeJS Dengan Express Di
NaZMa Office
Di CV.NaZMa Office

Oleh:

Geizka Rozilia Ruicosta / 119140114

disetujui dan disahkan sebagai
Laporan Praktik Kerja Lapangan

Lampung Selatan, 22 Desember 2022
Pembimbing Praktik Kerja Lapangan Program Studi Teknik Informatika ITERA



Andre Febrianto, S.Kom., M.Eng
NIP: 198602142019031008

Lembar Pengesahan

**Pembuatan Backend Website Menggunakan NodeJS Dengan Express Di
NaZMa Office**

Di CV.NaZMa Office

oleh :

Geizka Rozilia Ruicosta / 119140114

disetujui dan disahkan sebagai
Laporan Praktik Kerja Lapangan

Daerah Istimewa Yogyakarta, 10 Oktober 2022

Direktur Operasional


NaZMa

Nabila Zafira Mahalia. CV

Wiji Nurastuti, SE, MT

ABSTRAK

CV. Nabila Zafira Mahalia adalah sebuah perusahaan swasta yang bergerak dibidang IT dan Management Consulting dengan layanan profesional yang diberikan untuk para pelanggan. Sebagai perusahaan IT dan Management Consulting, CV. Nabila Zafira Mahalia memerlukan sebuah platform untuk menyampaikan konten dan informasi. CV. Nabila Zafira Mahalia memerlukan website yang dapat menyampaikan konten berupa berita, foto, dan acara yang ingin diselenggarakan, dan informasi mengenai layanan yang disediakan, proyek yang dikerjakan, dan kru yang ingin ditampilkan. Hal ini diperlukan agar perusahaan CV. Nabila Zafira Mahalia dapat menyampaikan seluruh informasi secara terpusat melalui website perusahaannya.

Pada pelaksanaan kuliah praktik ini, penulis berperan sebagai backend developer untuk mengembangkan backend dari website CV. Nabila Zafira Mahalia. Pengembangan backend website ini dikembangkan menggunakan NodeJS dengan Express dan metode pengembangan Waterfall. Backend website dibuat dengan membangun API untuk konten dan informasi yang diperlukan agar dapat dimanipulasi serta ditampilkan pada bagian frontend website apabila pengguna merupakan administrator dari website tersebut.

API dibangun dengan bahasa pemrograman JavaScript menggunakan array yang berisi objek dimana isi objek merupakan tipe data berbasis string/teks. Hal ini dilakukan karena website tidak dibuat menggunakan database, sehingga backend website hanya dapat menyimpan masukan kedalam API dalam bentuk string. API dapat diakses dengan cara mengakses rute endpoint dengan permintaan spesifik (GET/POST) kemudian backend website yang dibangun dengan Express akan memproses permintaan rute kepada pengguna, setelah itu pengguna dapat melakukan kegiatan CRUD terkait konten dan informasi yang ingin ditampilkan pada website.

Kata kunci: backend, website, API, GET, POST, rute, endpoint, admin, CRUD, NodeJS, Express

KATA PENGANTAR

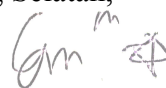
Pertama-tama saya panjatkan puja dan puji syukur atas rahmat dan ridho Allah SWT, untuk saya dapat melaksanakan dan menyelesaikan kerja praktik serta laporan pelaksanaan kerja praktik di CV. Nabila Zafira Mahalia (NaZMa Office). Penyusunan laporan ini diunjukkan untuk memenuhi persyaratan kelulusan dari mata kuliah IF4004 Praktik Kerja Lapangan.

Pelaksanaan dan penyusunan laporan kerja praktik ini dilaksanakan dengan lancar dengan bantuan dari pihak-pihak bersangkutan. Pada kesempatan ini, penulis mengucapkan terimakasih kepada pihak-pihak yang telah membantu, diantaranya:

1. Bapak Andre Febrianto, S.Kom., M.Eng selaku Dosen Pembimbing Kerja Praktik.
2. Ibu Wiji Nurastuti, SE, MT selaku Direktur Operasional CV.NaZMa Office.
3. Saudari Risti Yulianti S.Sn sebagai pembimbing dalam pelaksanaan kerja praktik.
4. Saudara Sukmadhia Aqila Risnuasa S.Kom yang telah meluangkan waktu memeriksa pengerjaan code untuk frontend dan backend.
5. Saudara Mayang Hermenda Anggraini 119140197 sebagai teman satu tim dalam pengerjaan kerja praktik serta saudara Aryo Kusuma 119140181 dan saudari Chantika Aurelia 119140117 selaku teman satu kantor.

Penulis menyadari bahwa masih banyak kekurangan baik dalam susunan, bahasa, maupun penulisan. Maka dari itu segala kritik dan saran dari pembaca kepada penulis bersifat terbuka sehingga penulis dapat memperbaiki kesalahan atau kekurangan yang ada. Semoga laporan praktik kerja lapangan ini bermanfaat dan membantu siapapun yang membacanya.

Lampung Selatan,



Geizka Rozilia Ruicosta

NIM. 119140114

DAFTAR ISI

ABSTRAK	3
KATA PENGANTAR	4
DAFTAR GAMBAR	8
DAFTAR TABEL	11
BAB I PENDAHULUAN	12
1.1 Latar belakang	12
1.2 Rumusan Masalah	13
1.3 Tujuan	13
1.4 Manfaat	13
1.4.1 Manfaat Bagi Pribadi	13
1.4.2 Manfaat Bagi Instansi	14
1.5 Ruang Lingkup	14
1.6 Metodologi	14
1.7 Sistematika Penulisan	14
BAB II GAMBARAN UMUM INSTANSI	16
2.1 Profil Organisasi	16
2.2 Visi dan Misi Organisasi	16
2.3 Struktur Organisasi	17
2.4 Deskripsi Pekerjaan	17
2.5 Jadwal Kerja	18
BAB III TINJAUAN PUSTAKA	19
3.1 Website	19
3.2 JavaScript Object Notation	19
3.3 NodeJS	19
3.4 ExpressJS	19
3.5 Embedded JavaScript Templates	20
3.6 Back end	20
3.7 Metode Waterfall	20
3.7 Black Box Testing	22
BAB IV ANALISIS DAN PERANCANGAN APLIKASI	23
4.1 Analisis Permasalahan	23
4.2 Alur Penyelesaian	24
4.3 Gambaran Umum Sistem Aplikasi	25
4.4 Spesifikasi Perangkat Lunak	25

4.5 Metodologi Pengembangan Perangkat Lunak	27
BAB V IMPLEMENTASI DAN PENGUJIAN	33
5.1 Pengembangan Backend Website	33
5.1.1 Rute Login	33
5.1.2 Rute Not Authorized	34
5.1.3 Rute Blog Create	35
5.1.4 Rute Gallery Create	36
5.1.5 Rute Event Create	37
5.1.6 Rute Blog Update	39
5.1.7 Rute Gallery Update	41
5.1.8 Rute Event Update	43
5.1.9 Rute Blog Delete	45
5.1.10 Rute Gallery Delete	46
5.1.11 Rute Event Delete	47
5.1.12 Rute Project Edit	48
5.1.13 Rute Testimony Edit	50
5.1.14 Rute Add Certified Trainer & Creative Crew	52
5.1.15 Rute Remove Certified Trainer & Creative Crew	53
5.2 Pengujian	55
5.3 Pembahasan	59
BAB VI KESIMPULAN DAN SARAN	60
6.1 Kesimpulan	60
6.2 Saran	60
Lampiran A. TOR (Term of Reference)	64
Lampiran B. Log Activity	74
Lampiran C. Dokumen Teknik	75
Lampiran D. Dokumentasi Kegiatan	76
Lampiran E. Pertanyaan	77

DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi CV. NaZMa Office	16
Gambar 3.1 Tahapan metode waterfall	20
Gambar 4.1 Flowchart Penyelesaian	23
Gambar 5.1 Rute GET Admin	32
Gambar 5.2 Halaman Login Admin	32
Gambar 5.3 Rute POST Login	33
Gambar 5.4 Halaman Beranda Administrator	33
Gambar 5.5 Rute GET Not Authorized	34
Gambar 5.6 Halaman Not Authorized	34
Gambar 5.7 Menu Create Blog	34
Gambar 5.8 Rute GET Blog Create	34
Gambar 5.9 Halaman Blog Create	35
Gambar 5.10 Rute POST Blog Create	35
Gambar 5.11 Menu Create Gallery	36
Gambar 5.12 Rute GET Gallery Create	36
Gambar 5.13 Halaman Gallery Create	36
Gambar 5.14 Rute POST Gallery Create	36
Gambar 5.15 Menu Create Event	37
Gambar 5.16 Rute GET Event Create	37
Gambar 5.17 Halaman Event Create	37
Gambar 5.18 Rute POST Event Create	37
Gambar 5.19 Menu Update Blog	38
Gambar 5.20 Rute GET Blog Update	38
Gambar 5.21 Halaman Blog Update	38

Gambar 5.22 Rute POST Blog Update	39
Gambar 5.23 Rute GET Blog Update 2	39
Gambar 5.24 Halaman Blog Update 2	39
Gambar 5.25 Rute POST Blog Update 2	40
Gambar 5.26 Menu Update Gallery	40
Gambar 5.27 Rute GET Gallery Update	40
Gambar 5.28 Halaman Gallery Update	40
Gambar 5.29 Rute POST Gallery Update	41
Gambar 5.30 Rute GET Gallery Update 2	41
Gambar 5.31 Halaman Gallery Update 2	41
Gambar 5.32 Rute POST Gallery Update 2	42
Gambar 5.33 Menu Update Event	42
Gambar 5.34 Rute GET Event Update	42
Gambar 5.35 Halaman Event Update	42
Gambar 5.36 Rute POST Event Update	43
Gambar 5.37 Rute GET Event Update 2	43
Gambar 5.38 Halaman Event Update 2	43
Gambar 5.39 Rute POST Gallery Update 2	44
Gambar 5.40 Menu Delete Blog	44
Gambar 5.41 Rute GET Blog Delete	44
Gambar 5.42 Halaman Blog Delete	45
Gambar 5.43 Rute POST Blog Delete	45
Gambar 5.44 Menu Delete Gallery	45
Gambar 5.45 Rute GET Gallery Delete	46
Gambar 5.46 Halaman Gallery Delete	46
Gambar 5.47 Rute POST Gallery Delete	46

Gambar 5.48 Menu Delete Event	46
Gambar 5.49 Rute GET Event Delete	47
Gambar 5.50 Halaman Event Delete	47
Gambar 5.51 Rute POST Event Delete	47
Gambar 5.52 Menu Project Edit	48
Gambar 5.53 Rute GET Project Edit	48
Gambar 5.54 Halaman Event Delete	48
Gambar 5.55 Rute POST Project Edit	49
Gambar 5.56 Menu Testimony Edit	49
Gambar 5.57 Rute GET Testimony Edit	50
Gambar 5.58 Halaman Testimony Edit	50
Gambar 5.59 Rute POST Testimony Edit	50
Gambar 5.60 Menu Add Certified Trainer & Creative Crew	51
Gambar 5.61 Rute GET Add Certified Trainer & Creative Crew	51
Gambar 5.62 Halaman Add Certified Trainer & Creative Crew	51
Gambar 5.63 Rute POST Add Certified Trainer & Creative Crew	52
Gambar 5.64 Menu Remove Certified Trainer & Creative Crew	52
Gambar 5.65 Rute GET Remove Certified Trainer & Creative Crew	52
Gambar 5.66 Halaman Remove Certified Trainer & Creative Crew	53
Gambar 5.67 Rute POST Remove Certified Trainer & Creative Crew	53
Gambar 5.68 Alur pada bagian Trainer	53

DAFTAR TABEL

Tabel 4.1 Kebutuhan Fungsional	24
Tabel 4.2 Kebutuhan Non Fungsional	25
Tabel 4.3 Pertanyaan dan Jawaban Kebutuhan Website	27
Tabel 4.4 Tabel EndPoint API	28
Tabel 4.5 Tabel Pengujian Black Box	29
Tabel 4.6 Kuesioner System Usability Scale	31
Tabel 5.1 Hasil Pengujian Black Box	54
Tabel 5.2 Hasil Pengujian System Usability Scale	56
Tabel 5.3 Rentang Nilai System Usability Scale	57

BAB I

PENDAHULUAN

1.1 Latar belakang

Pada dunia bisnis modern, internet sudah menjadi pintu keluar masuk untuk berbagai macam informasi dan konten. Bahkan akan terasa aneh dan asing apabila sebuah bisnis tidak memanfaatkan berbagai macam platform yang umumnya akan digunakan oleh orang banyak untuk menarik perhatian dari calon pelanggan. Website adalah salah satu platform yang dapat diakses oleh semua orang. Website ini bila dimanfaatkan dengan baik dapat menjadi media bisnis untuk memperkenalkan jasa kepada calon pelanggan.

Sebagai calon pelanggan, kita biasanya akan merasa aneh apabila sebuah bisnis tidak memiliki website resmi bukan? Calon pelanggan akan merasa apakah perusahaan yang ingin dilibatkan merupakan perusahaan yang kredibel atau tidak apabila sebuah perusahaan tidak memiliki website, khususnya untuk perusahaan IT.

CV. Nabila Zafira Mahalia atau dapat disebut juga dengan NaZMa Office adalah perusahaan swasta yang bergerak dibidang IT dan *Management Consulting* yang dipersiapkan secara profesional baik dari segi kualitas kelembagaan, Sumber Daya Manusia, maupun produk atau layanan yang disediakan, dengan mengedepankan kepuasan pelanggan.

CV. NaZMa Office sebagai perusahaan yang bergerak di bidang IT masih belum memiliki website perusahaan untuk mendukung pemasaran serta penyampaian informasi terkait. Selama ini, NaZMa Office mengandalkan media sosial umum untuk menunjukkan konten-konten informatifnya. Karena itu, penulis melakukan kerja praktik proyek pembuatan website resmi perusahaan. Pembuatan website dilakukan dengan arahan dari pihak perusahaan NaZMa Office. Diharapkan setelah website ini selesai lalu dilakukan hosting dan publikasi dapat menyelesaikan masalah perusahaan dalam penyampaian informasi.

Proyek yang dikerjakan oleh penulis merupakan backend dari website NaZMa Office. Tujuan pembuatan backend website ini adalah untuk

mengembangkan fitur create, read, update, dan delete pada admin untuk dapat menyampaikan informasi kepada pengunjung website melalui halaman blog, event, atau gallery.

1.2 Rumusan Masalah

Berdasarkan latar belakang terdapat beberapa poin rumusan masalah sebagai berikut:

1. Bagaimana cara membuat API dengan fitur create, read, update, dan delete pada backend website menggunakan NodeJS dengan Express?
2. Bagaimana cara perusahaan menyampaikan konten dan informasi kepada pengunjung website?

1.3 Tujuan

Berdasarkan rumusan masalah terdapat beberapa tujuan yang dapat dibentuk yaitu:

1. Membuat API untuk backend website NaZMa Office yang memiliki fitur create, read, update, dan delete.
2. Membuat backend pada website NaZMa Office untuk menyampaikan konten dan informasi.

1.4 Manfaat

1.4.1 Manfaat Bagi Pribadi

1. Mendapat pengalaman sebagai karyawan magang IT di perusahaan IT / Management Consulting.
2. Dapat mengimplementasikan ilmu yang diperoleh dari perkuliahan, khususnya pada mata kuliah Dasar Rekayasa Perangkat Lunak
3. Dapat mengimplementasikan ilmu yang diperoleh dari pembelajaran mandiri.

1.4.2 Manfaat Bagi Instansi

1. Perusahaan mendapat website resmi dengan fungsi-fungsi sesuai permintaan perusahaan.
2. Mendapat platform baru untuk digunakan sebagai tempat penyampaian informasi dan konten.

1.5 Ruang Lingkup

Ruang lingkup penulis dalam pelaksanaan kerja praktik sebagai berikut:

1. Kerja praktik dilaksanakan dengan rentang waktu 40 hari dari tanggal 18 Juli 2022 s/d 9 September 2022.
2. Pembuatan website dilakukan secara tim yang beranggotakan 2 orang.
3. Pengerjaan tugas harian dari kantor dikerjakan secara individu.
4. Pembuatan website pada backend menggunakan NodeJS dan ExpressJS.

1.6 Metodologi

Metodologi pencapaian pekerjaan selama aktivitas kuliah praktik dicapai dengan memanfaatkan Waterfall Software Development Life Cycle atau SDLC Waterfall. Penulis mengumpulkan informasi tentang apa saja yang dibutuhkan dan diinginkan atasan terkait proyek website yang akan dikerjakan melalui meeting, kemudian mengimplementasikan permintaan tersebut.

Setelah itu, penulis akan mempresentasikan hasil implementasi permintaan atasan untuk memverifikasi apakah permintaan yang diimplementasikan sudah sesuai sebagaimana yang diinginkan atasan. Apabila ada implementasi yang salah, maka akan dibenarkan.

1.7 Sistematika Penulisan

Adapun sistematika dalam melakukan penulisan laporan kerja praktik ini sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini membahas mengenai latar belakang dalam pembuatan proyek tugas kerja praktik, rumusan masalah, tujuan, manfaat, ruang lingkup, metodologi dan sistematika penulisan dari laporan kerja praktik.

BAB II GAMBARAN UMUM INSTANSI

Pada bab ini membahas mengenai profil organisasi, visi dan misi organisasi, struktur organisasi, deskripsi pekerjaan, dan jadwal kerja dari perusahaan/organisasi yang dijadikan tempat melakukan kerja praktik.

BAB III TINJAUAN PUSTAKA

Pada bab ini menjelaskan mengenai landasan teori atau pustaka yang digunakan dalam membuat laporan ataupun dalam pembuatan sistem yang dibuat.

BAB IV ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas mengenai perancangan dari perangkat lunak yang dikembangkan serta pengujian terhadap perangkat lunak tersebut menggunakan metode pengujian.

BAB V IMPLEMENTASI DAN PENGUJIAN

Bab ini membahas mengenai implementasi tentang perangkat lunak yang sudah dikembangkan serta pengujian terhadap perangkat lunak yang sudah dikembangkan menggunakan metode pengujian.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran dari hasil proyek kerja praktik yang telah dikembangkan.

BAB II

GAMBARAN UMUM INSTANSI

2.1 Profil Organisasi

CV. Nabila Zafira Mahalia (NaZMa Office) merupakan perusahaan swasta yang bergerak di bidang IT dan manajemen konsultan yang dipersiapkan secara profesional baik dari segi kualitas kelembagaan, sumber daya manusia, maupun produk atau layanan yang disediakan, dengan mengedepankan kepuasan pelanggan. NaZMa Office beralamat Jl. Selokan Mataram Pogung Dalangan SIA XVI No. 16 RT 11/RW 50 Sinduadi, Mlati, Sleman, Daerah Istimewa Yogyakarta. Serta memiliki kantor cabang yang beralamat Kp. Serangan Ng. II No. 191 RT 010/RW 002 Notoprajan, Ngampilan, Daerah Istimewa Yogyakarta.

2.2 Visi dan Misi Organisasi

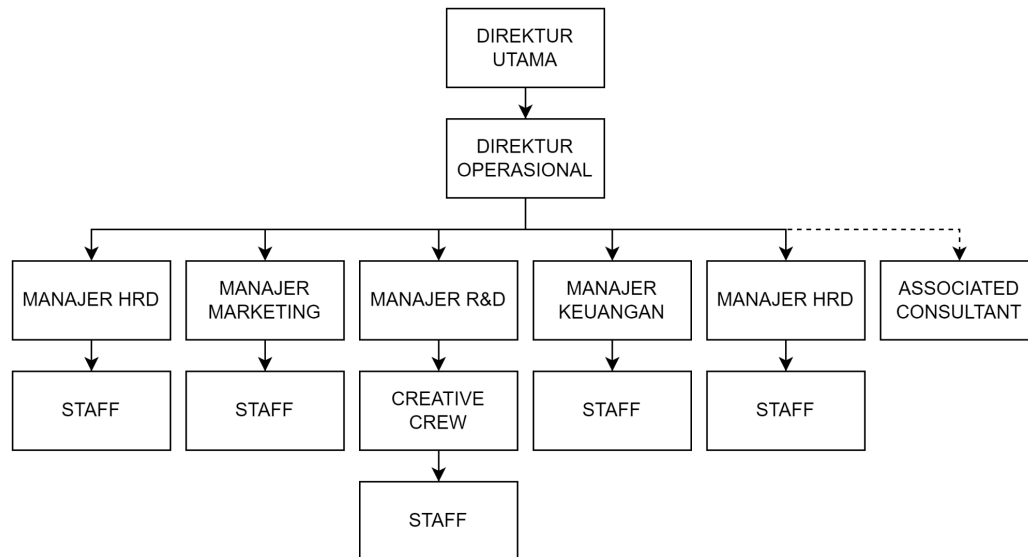
Visi

Menjadi perusahaan IT dan Management Consulting yang inspiratif, berkualitas dan barokah pada tahun 2024.

Misi

1. Memberikan layanan IT dan Management Consulting serta pengembangan lain yang berkualitas
2. memberikan value (nilai) tambah pada setiap produk dan jasa yang diberikan oleh NaZMa.
3. Membuat sistematis bisnis yang efektif, efisien, nyaman dan terintegrasi baik secara offline dan online
4. Selalu memberikan layanan pada customer dengan berprinsip “We’ll finish it with excellence” dengan target kepuasan pelanggan
5. Bekerjasama dan bersinergi dengan berbagai institusi, komunitas, perusahaan, kampus dan masyarakat untuk meningkatkan kinerja dan layanan perusahaan.

2.3 Struktur Organisasi



Gambar 2.1 Struktur Organisasi CV. NaZMa Office

Gambar diatas merupakan struktur organisasi CV. NaZMa Office, penulis pada saat pelaksanaan Kuliah Praktik berada pada bagian Staff dibawah Creative Crew. Divisi Creative Crew dikepalai atau diawasi oleh Manajer Research & Development, hal ini dikarenakan segala penelitian dan pengembangan atau pembuatan akan dilaksanakan oleh Creative Crew beserta Staff[1].

2.4 Deskripsi Pekerjaan

Pekerjaan penulis selama melaksanakan kegiatan kuliah praktik di NaZMa Office adalah backend developer yang melengkapi website dengan frontend yang telah dibuat oleh rekan satu tim penulis yaitu Mayang Hermenda Anggraini. penulis mengimplementasikan proses server dan logika, kemudian juga menerapkan konsep CRUD untuk semua konten melalui sisi administrator pada website.

Penulis juga mengubah file ekstensi HTML yang telah dibuat rekan satu tim penulis menjadi file ekstensi EJS yang dapat digunakan secara berulang dengan logika khusus pada beberapa bagian website.

2.5 Jadwal Kerja

Jadwal kerja penulis pada pelaksanaan kerja praktik di CV. NaZMa Office dilaksanakan pada:

Hari Kerja : Senin-Jumat

Jam Kerja : 08.00 - 16.00 WIB

Penulis melaksanakan kerja praktik ini selama 40 hari kerja, terhitung dari hari Senin, 18 Juli 2022 sampai dengan Jumat, 9 September 2022. Detail dari kegiatan dan jadwal kerja praktik terlampir pada lampiran Presensi dan *Logsheet* Praktek Kerja Lapangan.

BAB III

TINJAUAN PUSTAKA

3.1 Website

Website adalah keseluruhan halaman-halaman web yang terdapat dari sebuah domain yang mengandung informasi [2]. *Website* dapat memberikan efisiensi untuk organisasi dalam mengurangi ambiguitas dan ketidakpastian dengan meningkatkan hubungan yang berlangsung dengan rekan, pemasok, dan pelanggan[9].

3.2 JavaScript Object Notation

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari bahasa pemrograman JavaScript[3]. JSON terdiri dari dua struktur yaitu:

1. Pasangan nama dengan nilai.
2. Daftar nilai terurutkan.

3.3 NodeJS

NodeJS adalah run timer JavaScript yang digunakan pada browser untuk memproses HTTP. NodeJS memungkinkan penulis untuk membuat backend dari sebuah website yang dapat menerima *request* GET dan POST menggunakan bahasa pemrograman JavaScript[4].

3.4 ExpressJS

ExpressJS adalah framework web yang dibuat berdasarkan module HTTP dan Connect pada NodeJS[5]. Penggunaan ExpressJS pada website dapat mempercepat proses penguraian body dari *request* HTTP, penguraian cookies, dan pengarahan rute berdasarkan jalur URL dan *request* dari metode HTTP.

3.5 *Embedded JavaScript Templates*

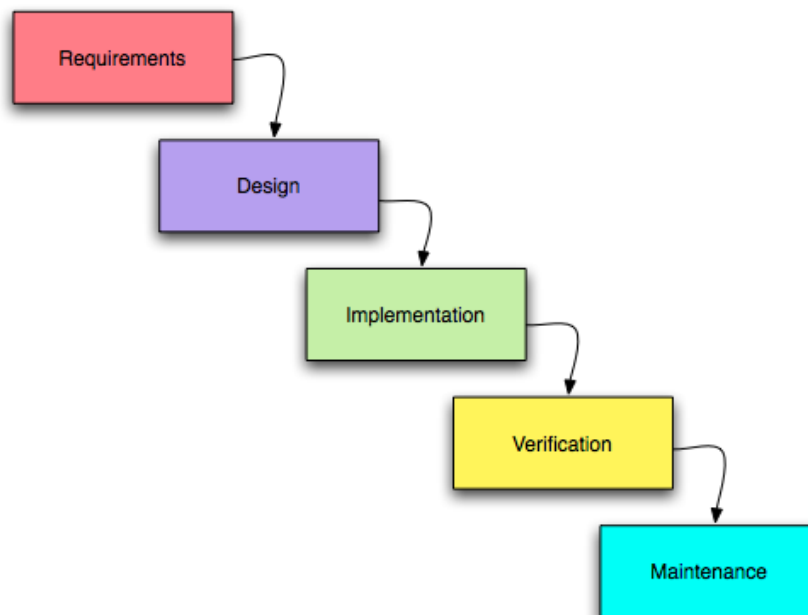
EJS adalah bahasa templating sederhana yang memungkinkan untuk membuat (generate) markup HTML dengan JavaScript biasa. Tidak ada panduan untuk mengatur sesuatu. Memiliki fitur kompilasi dan *rendering* cepat, tag template sederhana, dan lainnya.

3.6 *Back end*

Back end atau sering disebut server side pada dasarnya adalah tempat dimana proses suatu aplikasi atau sistem berjalan di *back end* ini data di proses, ditambahkan, diubah atau dihapus. *Back end* mengatur segala sesuatu yang biasanya tidak dilihat atau berinteraksi langsung kepada *user*, seperti database dan server [6].

3.7 *Metode Waterfall*

Waterfall adalah salah satu model SDLC yang seringkali digunakan dalam pengembangan sistem informasi atau perangkat lunak. Metode ini menggunakan pendekatan sistematis dan berurutan. Tahapan yang ada dalam model ini dimulai dari tahap perencanaan hingga tahap pengelolaan (*maintenance*) dan dilakukan secara bertahap. Metode waterfall atau yang biasa dikenal dengan metode waterfall sering disebut dengan siklus hidup klasik (*classical life cycle*), nama model sebenarnya adalah “model sekuensial linier”, dimana bagian ini menjelaskan pendekatan sistematis dan sekuensial untuk pengembangan perangkat lunak diproduksi [7]. Pada Gambar 3.1 berikut merupakan tahapan-tahapan dalam metode *waterfall*.



Gambar 3.1 Tahapan metode waterfall

Berikut adalah penjelasan tahapan-tahapan yang terdapat pada metode *waterfall*.

1. Analisis Kebutuhan

Pada fase ini harus memenuhi persyaratan perangkat lunak, termasuk penggunaan perangkat lunak yang diharapkan dan batasan perangkat lunak. Informasi mengenai analisis kebutuhan biasanya didapatkan melalui wawancara, diskusi atau survey.

2. Desain Sistem

Pada tahapan desain sistem harus diselesaikan terlebih dahulu sebelum melanjutkan ke proses selanjutnya yaitu pembuatan coding atau pemrograman. Bertujuan agar membangun desain berdasarkan hasil analisis yang diperoleh sesuai dengan kebutuhan perangkat keras dan sistem secara keseluruhan.

3. Implementasi

Pada tahapan ini adalah proses coding atau pemrograman. Pengembangan perangkat lunak dibagi menjadi modul kecil, yang kemudian akan digabungkan pada langkah berikutnya.

4. Pengujian

Tahapan ini dibuat untuk menggabungkan dan melakukan pengujian yang bertujuan untuk melihat apakah perangkat lunak yang dibuat sesuai dengan desain yang telah ditentukan sebelumnya.

5. Pemeliharaan

Pada tahapan ini adalah proses dilakukannya pemeliharaan, termasuk bug program, kesalahan, dan kemungkinan malfungsi dari perangkat lunak aplikasi yang diinstall.

3.7 ***Black Box Testing***

Black box testing adalah teknik untuk mengetahui apakah kita membangun perangkat lunak yang benar (*validation*). *Black box testing* disebut juga sebagai pengujian fungsional, teknik pengujian fungsional yang merancang kasus uji berdasarkan informasi dari spesifikasi. Dengan menggunakan *black box testing*, penguji perangkat lunak tidak boleh (atau tidak) memiliki akses ke kode sumber internal. Pengujian dengan menggunakan teknik ini juga tidak berkaitan dengan mekanisme internal suatu sistem, hanya berfokus pada output yang dihasilkan sebagai respons terhadap input dan kondisi eksekusi yang dipilih [8].

BAB IV

ANALISIS DAN PERANCANGAN APLIKASI

4.1 Analisis Permasalahan

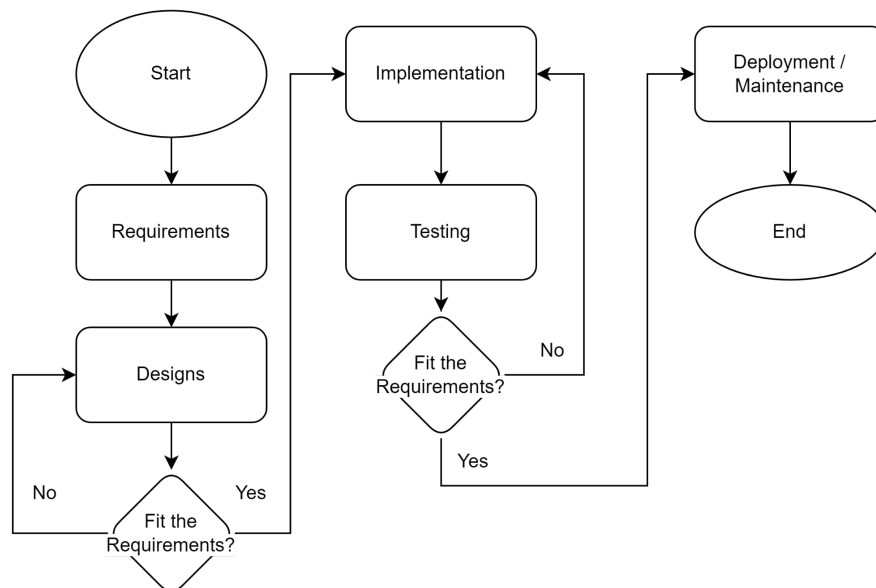
CV. Nabila Zafira Mahalia adalah perusahaan swasta yang menyediakan layanan profesional dibidang IT dan *Management Consulting* berdasarkan kebutuhan pelanggan. CV. Nabila Zafira Mahalia mengharapkan sebuah website resmi untuk dapat menyampaikan informasi dan konten serta layanan yang disediakan oleh perusahaan agar para pengunjung website bisa mendaftar layanan yang diinginkannya. Karena saat ini perusahaan masih mengandalkan *WordPress* dan sosial media seperti *Instagram*, *TikTok*, dan *LinkedIn* untuk menyampaikan konten sehingga calon client perusahaan harus mengunjungi banyak situs dan aplikasi yang berbeda untuk mendapat informasi dari sumber yang sama yaitu perusahaan NaZMa Office ini. Maka solusi dari permasalahan tersebut adalah perusahaan harus memiliki sebuah media utama sebagai tempat penyampaian informasi, pada kasus ini tim penulis membuat sebuah website resmi dengan penulis membuat backend untuk website resmi NaZMa Office.

Dalam proses penyampaian informasi dan konten pada website, sudah pasti diperlukan website yang dapat melakukan proses GET dan POST pada server yang berarti memerlukan backend dan API. Oleh karena itu, penulis menggunakan NodeJS dengan Express sebagai framework untuk memproses *request* HTTP berdasarkan jalur URL dan metode permintaannya.

Dalam penyampaian konten yang bersinergi dengan tampilan frontend, maka akan terdapat tampilan yang berulang pada frontend yang masing-masing harus dapat menampilkan informasi yang berbeda sesuai dengan apa yang ada dalam API. Oleh sebab itu, penulis mengkonversi file ekstensi HTML yang telah dibuat Saudari Mayang Hermenda Anggraini menjadi file ekstensi EJS (Embedded JavaScript) yang merupakan alternatif dari HTML dengan tambahan JavaScript yang dapat mengatur alur dari sebuah halaman website. Dengan ini, konten yang ada pada tampilan pada website akan berubah secara dinamis menggunakan konsep template yang merupakan fitur utama dari EJS.

4.2 Alur Penyelesaian

Dalam pembuatan Backend Pada Website NaZMa Office Menggunakan NodeJS dengan Express, terdapat proses yang dilakukan secara bertahap dan setiap tahapannya saling berkaitan. Alur penyelesaian project kerja praktik ini dapat dilihat pada gambar 4.1 Alur Penyelesaian.



Gambar 4.1 Flowchart Penyelesaian

Pada Gambar 4.1 dapat diperhatikan bahwa pengerjaan proyek dilakukan secara bertahap. Tahapan dimulai dari mendengarkan *user stories* dan menyusun permintaan yang dibutuhkan pada backend website NaZMa Office. Selanjutnya, penulis membuat rancangan backend yang dibutuhkan sesuai dengan permintaan yang akan divalidasi dengan client apakah sesuai dengan permintaan atau tidak. Apabila rancangan yang dibentuk sudah sesuai dengan permintaan client, maka rancangan backend akan diimplementasi pada website. Setelah pengujian pada proyek telah berhasil dicapai sesuai permintaan, maka tahapan akan berakhir pada *deployment* dan *maintenance* untuk memastikan bahwa proyek telah berhasil diselesaikan dan berfungsi dengan baik.

4.3 Gambaran Umum Sistem Aplikasi

Backend Website NaZMa Office merupakan aplikasi berbasis web yang dibangun untuk mempermudah perusahaan dalam mengelola konten dan informasi terkait berita (*blog*), foto (*gallery*), dan acara (*event*) yang ingin diumumkan pada website. Selanjutnya juga backend website akan digunakan untuk mengubah testimoni, proyek berjalan, dan kru kreatif pada halaman beranda. Backend website ini dibangun menggunakan NodeJS dengan beberapa ketergantungan (*dependencies*) utama seperti ExpressJS sebagai framework dari modul HTTP pada NodeJS untuk mengurus alur permintaan URL pada website, baik metode GET maupun POST.

Pada setiap fungsi CRUD pada backend, pengguna administrator dapat memasukkan informasi berupa seperti foto, link, dan tulisan untuk disampaikan melalui website.

4.4 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak pada pembuatan backend Pada Website NaZMa Office Menggunakan NodeJS Dengan Express dibagi menjadi dua yaitu kebutuhan fungsional dan non fungsional.

4.4.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah layanan yang disediakan oleh sistem dan respon dari sistem ketika diberi inputan. Berikut merupakan kebutuhan fungsional dari backend website yang akan dibangun dapat dilihat pada tabel 4.1 Dibawah ini.

Tabel 4.1 Kebutuhan Fungsional

ID	Deskripsi
SCT01	Website dapat menambah, mengubah dan menghapus konten pada halaman Blog
SCT02	Website dapat menambah, mengubah dan menghapus konten pada halaman Galeri
SCT03	Website dapat menambah, mengubah dan menghapus konten pada halaman Event

ID	Deskripsi
SCT04	Website dapat mengubah nama proyek berjalan pada halaman Home bagian Ongoing Project
SCT05	Website dapat menambah dan menghapus konten pada halaman Home bagian Certified Trainer & Creative Crew
SCT06	Website dapat mengubah testimoni pada halaman Home bagian Testimoni
SCT07	Website dapat melakukan login Administrator dengan email dan password dan dapat melakukan sign out

4.4.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional berisi tentang keterbatasan yang ada pada sistem. Berikut ini kebutuhan non fungsional dari website yang dibangun dapat dilihat pada tabel 4.2 dibawah ini.

Tabel 4.2 Kebutuhan Non Fungsional

SKPL-ID	Kebutuhan Non Fungsional	Penjelasan
SKPL-NF-01	<i>Availability</i>	Dapat diakses 24 jam sehari, dalam 7 hari
SKPL-NF-02	<i>Usability</i>	Website mudah dipelajari karena tampilan yang <i>user-friendly</i>
SKPL-NF-03	<i>Compatibility</i>	Tampilan website <i>responsive</i> jika dibuka di browser dengan berbagai ukuran perangkat
SKPL-NF-04	<i>Portability</i>	Website dapat digunakan selama terhubung ke komputer server dan internet

SKPL-ID	Kebutuhan Non Fungsional	Penjelasan
SKPL-NF-05	<i>Integratibility & Reusability</i>	Dapat menggabungkan semua kebutuhan dan dapat digunakan kembali di masa mendatang jika ada pengembangan untuk pembuatan sistem baru.

4.5 Metodologi Pengembangan Perangkat Lunak

Waterfall adalah salah satu model SDLC yang seringkali digunakan dalam pengembangan sistem informasi atau perangkat lunak. Metode ini menggunakan pendekatan sistematis dan berurutan. Tahapan yang ada dalam model ini dimulai dari tahap perencanaan hingga tahap pengelolaan (*maintenance*) dan dilakukan secara bertahap. Metode waterfall atau yang biasa dikenal dengan metode waterfall sering disebut dengan siklus hidup klasik (*classical life cycle*), nama model sebenarnya adalah “model sekuensial linier”, dimana bagian ini menjelaskan pendekatan sistematis dan sekuensial untuk pengembangan perangkat lunak diproduksi [7].

Berikut adalah tahapan-tahapan dilakukan untuk yang terdapat pada metode *waterfall*.

1. Analisis Kebutuhan

Informasi mengenai kebutuhan pada backend website NaZMa Office didapatkan melalui wawancara dan diskusi dengan staff kreatif perusahaan dapat dilihat pada Tabel 4.3 dibawah ini.

Tabel 4.3 Pertanyaan dan Jawaban Kebutuhan Website

No	Pertanyaan	Jawaban
1	Apa fungsi yang diharapkan ada pada website/backend website?	Website diharapkan dapat menampilkan postingan berita, foto, dan acara yang akan diadakan.
2	Pada berita, apa isi konten yang ingin ditampilkan?	Pada postingan berita, isi konten terdiri dari judul, foto, dan isi konten itu sendiri.
3	Pada bagian foto, apa isi konten yang ingin ditampilkan?	Pada postingan foto, isi konten hanyalah foto-foto seperti galeri dan memiliki tulisan karikatur (<i>caption</i>) pendek pada masing-masing foto
4	Pada bagian acara, apa isi konten yang ingin ditampilkan?	Pada acara, isi konten harus ada judul acara, tanggal, jam, dan lokasi acara yang akan diselenggarakan, poster acara, dan tempat pendaftaran acara.
5	Bagaimana postingan tersebut ingin ditampilkan pada website?	Postingan berita, foto, dan acara ingin ditampilkan dari yang paling baru ke yang paling lama berdasarkan tanggal pembuatan postingan.
6	Apa ada fungsi lain yang diharapkan dapat dilakukan oleh website/backend website?	Website diharapkan dapat menampilkan kru dan proyek yang dikerjakan saat ini. Kemudian dapat ditambah, diubah, atau dihapus melalui backend.

2. Desain Sistem

Pada tahap ini, sistem dibuat dengan teknologi NodeJS, yaitu bahasa pemrograman JavaScript khusus sisi server dengan framework Express untuk menggunakan modul HTTP pada NodeJS. Dengan ini, backend dapat dibuat berupa API untuk masing-masing konten dan informasi yang ingin disampaikan.

Penulis juga harus membuat tampilan antarmuka agar user administrator dapat melakukan manipulasi data terhadap API yang akan dibuat, baik penambahan data, perubahan data, maupun penghapusan data.

Pada Tabel 4.4 dapat dilihat daftar API yang perlu dibuat beserta fungsi yang dilakukannya.

Tabel 4.4 Tabel EndPoint API

Metode	EndPoint	Deskripsi
Get	/admin	Masuk sebagai administrator
Get	/:app	Mengakses blog / gallery / event untuk ditunjukkan
Post	/:app-create	Membuat konten pada blog / gallery / event
Post	/:app-update	Mengubah isi konten pada blog / gallery / event
Post	/:app-delete	Menghapus konten dari blog / gallery / event
Post	/logout	Keluar dari administrator

3. Implementasi

Pada tahapan ini adalah proses pemrograman backend website berdasarkan desain sistem yang telah dibuat. Program akan dikembangkan dari desain sistem menjadi lebih banyak endpoint untuk mengakses API dan melakukan fungsi CRUD. Langkah implementasi ini akan dijelaskan lebih lanjut pada Bab V Implementasi dan Pengujian.

4. Pengujian

Tahapan ini dibuat untuk menggabungkan dan melakukan pengujian yang bertujuan untuk melihat apakah perangkat lunak yang dibuat sesuai dengan

desain yang telah ditentukan sebelumnya. Pada tahap pengujian, dilakukan dengan 2 metode, yaitu Black Box dan System Usability Scale.

a. Black Box

Pengujian Black Box adalah suatu pengujian perangkat lunak yang dilakukan dengan mengamati hasil masukan dan keluaran perangkat lunak tanpa mengetahui struktur kode perangkat lunak tersebut. Pada tabel 4.5 berikut ini dapat diperhatikan pengujian yang dilakukan menggunakan metode Black Box.

Tabel 4.5 Tabel Pengujian Black Box

No	Skenario Pengujian	Kasus Pengujian	Hasil yang diharapkan
1	Masuk sebagai Administrator	Memasukkan username dan password, kemudian menekan tombol <i>sign in</i>	Sistem akan mengarahkan user ke halaman administrator apabila username dan password benar, dan akan tetap berada di halaman login apabila username atau password salah
2	Membuat berita baru pada blog	Memasukkan judul, foto, dan isi berita, kemudian menekan tombol submit	Sistem akan menambahkan judul, foto, dan isi berita menjadi berita baru, dan berita akan terlihat pada halaman blog
3	Membuat foto baru pada gallery	Memasukkan foto dan karikatur foto, kemudian menekan tombol submit	Sistem akan menambahkan foto dan karikatur terbaru pada halaman gallery di paling kiri atas.
4	Membuat acara pada event	Memasukkan judul acara, alamat, waktu, bulan, tanggal, foto poster, link pendaftaran, dan deskripsi acara.	Acara akan muncul pada halaman event pada urutan paling atas dengan rincian acara sesuai dengan data yang sudah dimasukkan
5	Mengubah berita pada blog	Memilih berita mana yang ingin diubah, kemudian dapat mengubah judul, foto, dan isi berita.	Isi berita pada blog akan berubah tanpa mengubah urutan berita.

No	Skenario Pengujian	Kasus Pengujian	Hasil yang diharapkan
6	Mengubah foto pada gallery	Memilih foto mana yang ingin diubah berdasarkan karikturnya, kemudian dapat mengubah foto dan/atau karikatur dari foto tersebut.	Foto yang diubah akan tetap berada pada urutannya
7	Mengubah acara pada event	Memilih acara dari API event berdasarkan judul acara, kemudian dapat mengubah seluruh isi acara dan menyimpan perubahannya.	Acara yang diubah akan tetap berada pada urutannya.
8	Menghapus berita pada blog	Memilih berita berdasarkan judul dan menghapusnya.	Berita akan hilang dari halaman blog
9	Menghapus foto dari gallery	Memilih foto berdasarkan karikatur untuk menghapusnya.	Foto akan hilang dari halaman gallery
10	Menghapus acara dari event	Memilih acara berdasarkan judulnya dan menghapusnya	Acara akan hilang dari halaman event
11	Mengubah tampilan project	Mengubah proyek yang ingin ditampilkan berupa teks	Dapat mengubah dan menampilkan 6 proyek untuk ditunjukkan pada halaman beranda
12	Mengubah testimoni client	Mengubah testimoni client yang ingin ditampilkan berupa testimoni, foto client, dan nama client	Dapat mengubah dan menampilkan 3 testimoni client yang ingin ditampilkan pada halaman beranda
13	Menambah kru yang ingin ditampilkan	Menambahkan kru yang ingin ditampilkan berupa foto, nama, dan keahliannya	Dapat menambahkan kru yang ingin ditampilkan pada halaman beranda tanpa batas maksimal
14	Menghapus kru yang ingin ditampilkan	Menghapus kru yang ingin ditampilkan berdasarkan nama kru	Dapat menghapus kru dari halaman beranda

b. System Usability Scale

System Usability Scale (SUS) adalah kuesioner yang digunakan untuk mengukur tingkat kemudahan dalam penggunaan sistem. Pada kegiatan

kali ini, penulis menggunakan SUS untuk mengukur tingkat kemudahan user administrator dalam mengoperasikan backend website melalui interface yang disediakan untuk melakukan fungsi CRUD pada API backend yang telah dibuat. Pada Tabel 4.6 dapat dilihat daftar pertanyaan untuk kuesioner System Usability Scale.

Tabel 4.6 Kuesioner System Usability Scale

No	Pertanyaan
1	Saya berpikir akan menggunakan backend website ini lagi
2	Saya merasa backend website ini rumit untuk digunakan
3	Saya merasa backend website ini mudah digunakan
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan backend website ini
5	Saya merasa fitur-fitur backend website ini berjalan dengan semestinya
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada backend website ini)
7	Saya merasa orang lain akan memahami cara menggunakan backend website ini dengan cepat
8	Saya merasa backend website ini membingungkan
9	Saya merasa tidak ada hambatan dalam menggunakan backend website ini
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan backend website ini

Kuesioner SUS ini akan diajukan kepada sejumlah pekerja / staff pada NaZMa Office yang bertanggung jawab untuk melakukan pembuatan dan publikasi konten. Kemudian hasil dari kuesioner SUS ini akan berupa rata-rata skor dari semua responden dengan rentang nilai dari 0 s/d 100.

BAB V

IMPLEMENTASI DAN PENGUJIAN

Implementasi kuliah praktik penulis dilaksanakan menggunakan *Software Development Life Cycle* (SDLC) Waterfall. Implementasi dilakukan dengan estimasi selama 40 kerja. Pengujian pada implementasi proyek penulis dapat dilihat sebagai berikut.

5.1 Pengembangan Backend Website

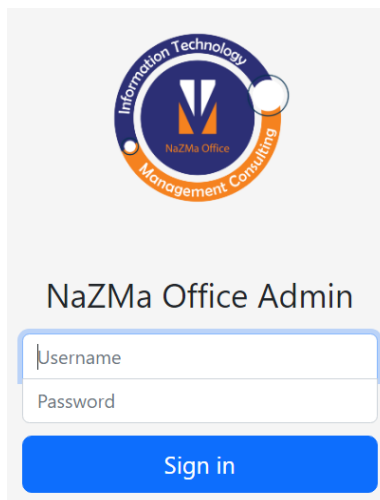
5.1.1 Rute Login

Pengguna administrator dapat melakukan login kedalam aplikasi web dengan melakukan permintaan GET ke server melalui rute URL “/admin” seperti pada Gambar 5.1.

```
router.get('/admin', function (req, res, next) {  
    res.render("admin", { invalidAdmin: adminStatement });  
    next();  
});
```

Gambar 5.1 Rute GET Admin

Setelah melakukan permintaan GET, pengguna akan masuk ke halaman login untuk mengisi username dan passwordnya seperti pada Gambar 5.2.



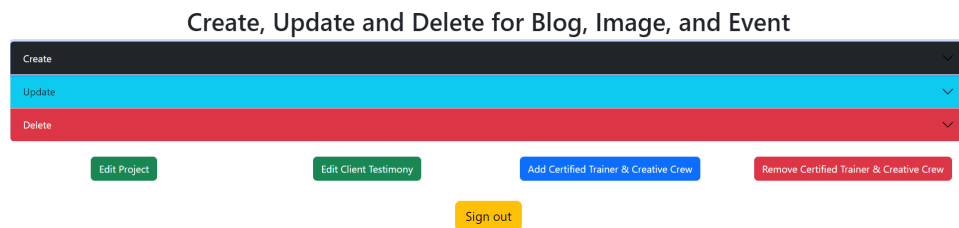
Gambar 5.2 Halaman Login Admin

Selanjutnya, pengguna harus mengisi username dan password yang benar untuk melakukan permintaan POST ke rute “/login/password” seperti pada Gambar 5.3 berikut.

```
router.post('/login/password', passport.authenticate('local', {
  successRedirect: '/',
  failureRedirect: '/login'
}));
```

Gambar 5.3 Rute POST Login

Apabila username dan password benar, pengguna akan diarahkan ke rute “/” dan menampilkan halaman beranda administrator. Namun apabila username atau password salah, pengguna akan tetap berada di halaman login admin. Pada Gambar 5.4 berikut adalah hasil halaman apabila pengguna berhasil melakukan login.



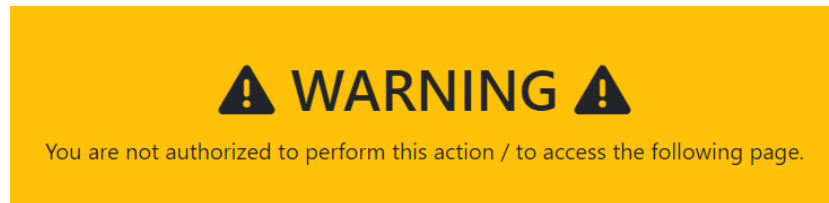
Gambar 5.4 Halaman Beranda Administrator

5.1.2 Rute Not Authorized

Sistem dapat mencegah pengguna umum untuk langsung mengakses permintaan rute GET yang hanya bisa dilakukan apabila pengguna sudah melakukan login sebagai administrator untuk mencegah adanya aktivitas yang tidak diinginkan dari pihak yang tidak bertanggung jawab. Apabila pengguna umum mencoba untuk mengakses rute yang hanya bisa diakses sebagai administrator, maka akan dialihkan ke rute “/not-authorized” seperti pada Gambar 5.5 dengan menampilkan tampilan seperti Gambar 5.6. dibawah ini.

```
router.get("/not-authorized", (req, res) => {
  res.render("admin/unauthorized");
});
```

Gambar 5.5 Rute GET Not Authorized



Gambar 5.6 Halaman Not Authorized

5.1.3 Rute Blog Create

Admin dapat membuat berita baru pada halaman blog dengan memilih menu “Blog” pada “Create” yang ada pada tampilan halaman beranda admin. Gambar 5.7 dibawah adalah ilustrasi menu tersebut.



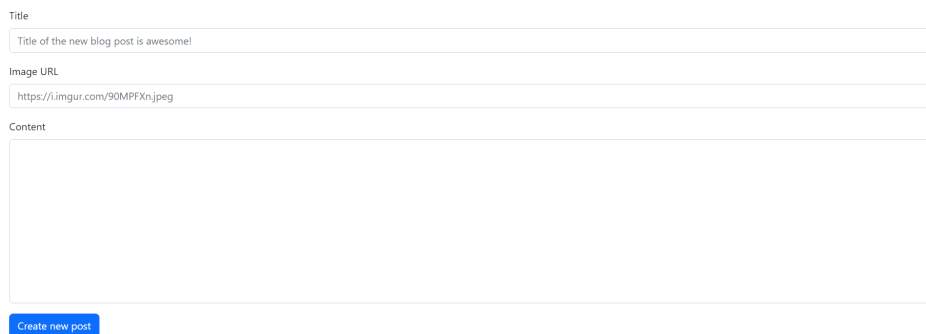
Gambar 5.7 Menu Create Blog

Setelah memilih Create Blog, pengguna akan mengakses permintaan GET ke rute “/blog-create” untuk dapat mengakses halaman Create Blog seperti pada Gambar 5.8 dan Gambar 5.9 berikut ini.

```
router.get("/blog-create", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next();
  }
  else {
    res.render("admin/blog-create");
  }
});
```

Gambar 5.8 Rute GET Blog Create

Create new Article for Blog



Gambar 5.9 Halaman Blog Create

Selanjutnya, admin dapat memasukkan judul dari berita, link berisi foto berita, dan isi dari berita. Setelah selesai, admin dapat menekan tombol “Create new post” untuk membuat permintaan POST ke rute “/blog-create” seperti pada Gambar 5.10 dibawah ini.

```
router.post("/blog-create", (req, res) => {  
  const newpost = {  
    title: req.body.createBlogTitle,  
    image: req.body.createBlogImage,  
    content: req.body.createBlogContent  
  }  
  article.unshift(newpost);  
  res.redirect("/")  
});
```

Gambar 5.10 Rute POST Blog Create

Kemudian program akan menyimpan berita yang telah dibuat kedalam API Blog dan menampilkannya di halaman blog untuk dapat dilihat oleh pengguna umum.

5.1.4 Rute Gallery Create

Admin dapat membuat foto baru pada halaman gallery dengan memilih menu “Gallery” pada “Create” yang ada pada tampilan halaman beranda admin. Gambar 5.11 dibawah adalah ilustrasi menu tersebut.



Gambar 5.11 Menu Create Gallery

Setelah memilih Create Gallery, pengguna akan mengakses permintaan GET ke rute “/gallery-create” untuk dapat mengakses halaman Create Gallery seperti pada Gambar 5.12 dan Gambar 5.13 berikut ini.

```
router.get("/gallery-create", (req, res, next) => {  
  if (!req.user) {  
    res.redirect("/not-authorized");  
    next()  
  }  
  else {  
    res.render("admin/gallery-create");  
  }  
});
```

Gambar 5.12 Rute GET Gallery Create



Gambar 5.13 Halaman Gallery Create

Selanjutnya, admin dapat memasukkan link berisi foto, dan karikatur pendek. Setelah selesai, admin dapat menekan tombol “Post a Picture” untuk membuat permintaan POST ke rute “/gallery-create” seperti pada Gambar 5.14 dibawah ini.

```
router.post("/gallery-create", (req, res) => {  
  const newpic = {  
    url: req.body.createGalleryImage,  
    caption: req.body.createGalleryCaption  
  }  
  gallery.unshift(newpic);  
  res.redirect("/");  
});
```

Gambar 5.14 Rute POST Gallery Create

Kemudian program akan menyimpan foto yang telah dibuat ke dalam API Gallery dan menampilkannya di halaman gallery untuk dapat dilihat oleh pengguna umum.

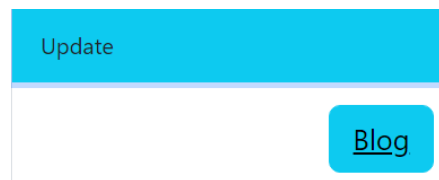
5.1.5 Rute Event Create

Admin dapat membuat acara baru pada halaman event dengan memilih menu “Event” pada “Create” yang ada pada tampilan halaman beranda admin. Gambar 5.15 dibawah adalah ilustrasi menu tersebut.

Kemudian program akan menyimpan acara yang telah dibuat ke dalam API Event dan menampilkannya di halaman event untuk dapat dilihat oleh pengguna umum.

5.1.6 Rute Blog Update

Admin dapat mengubah berita yang sudah ditampilkan pada halaman blog dengan memilih menu “Blog” pada “Update” yang ada pada tampilan halaman beranda admin. Gambar 5.19 dibawah adalah ilustrasi menu tersebut.



Gambar 5.19 Menu Update Blog

Setelah memilih Update Blog, pengguna akan mengakses permintaan GET ke rute “/blog-update” untuk dapat mengakses halaman Update Blog dan memilih berita yang ingin diubah seperti pada Gambar 5.20 dan Gambar 5.21 berikut ini.

```
router.get("/blog-update", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next();
  }
  else {
    res.render("admin/blog-update", { articles: article });
  }
})
```

Gambar 5.20 Rute GET Blog Update



Gambar 5.21 Halaman Blog Update

Selanjutnya, admin dapat memilih judul berita untuk diubah. Setelah selesai, admin dapat menekan tombol “Continue” untuk membuat

permintaan POST ke rute “/blog-update” seperti pada Gambar 5.22 dibawah ini.

```
router.post("/blog-update", (req, res) => {
  changeIndex = Number(req.body.articleNumber);
  res.redirect("/blog-update-2")
})
```

Gambar 5.22 Rute POST Blog Update

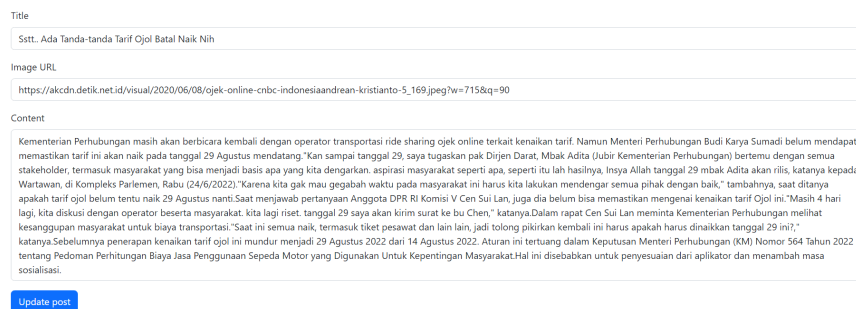
Selanjutnya, pengguna akan diarahkan ke rute GET “/blog-update-2” untuk dapat mengubah rincian berita yang diinginkan seperti pada Gambar 5.23 berikut.

```
router.get("/blog-update-2", (req, res, next) => {
  if(!req.user) {
    res.redirect("/not-authorized");
    next();
  }
  else {
    res.render("admin/blog-update-2", {
      inheritBlogTitle: article[changeIndex].title,
      inheritBlogImage: article[changeIndex].image,
      inheritBlogContent: article[changeIndex].content
    })
  }
})
```

Gambar 5.23 Rute GET Blog Update 2

Setelah mengakses rute tersebut, pengguna akan diarahkan ke halaman blog update 2 dengan ilustrasi seperti Gambar 5.24 berikut ini.

Edit / Update Article for Blog



Title

Sitt. Ada Tanda-tanda Tarif Ojol Batal Naik Nih

Image URL

https://akcdn.detik.net.id/visual/2020/06/08/ojek-online-cnbc-indonesiaandrea-kristianto-5_169.jpeg?w=715&q=90

Content

Kementerian Perhubungan masih akan berbicara kembali dengan operator transportasi ride sharing ojek online terkait kenaikan tarif. Namun Menteri Perhubungan Budi Karya Sumadi belum mendapat memastikan tarif ini akan naik pada tanggal 29 Agustus mendatang. "Kan sampai tanggal 29, saya tugaskan pak Dirjen Darat, Mbak Adita (Jubir Kementerian Perhubungan) bertemu dengan semua stakeholder, termasuk masyarakat yang bisa menjadi basis apa yang kita dengarkan, aspirasi masyarakat seperti apa, seperti itu lah hasilnya, Insya Allah tanggal 29 mbak Adita akan rilis, katanya kepada Wartawan, di Kompleks Parlemen, Rabu (24/6/2022). "Karena kita gak mau gegabah waktu pada masyarakat ini harus kita lakukan mendengar semua pihak dengan baik," tambahnya, saat ditanya apakah tarif ojol belum tentu naik 29 Agustus nanti. Saat menjawab pertanyaan Anggota DPR RI Komisi V Cen Sui Lan, juga dia belum bisa memastikan mengenai kenaikan tarif Ojol ini. "Masih 4 hari lagi, kita diskusi dengan operator beserta masyarakat, kita lagi riset, tanggal 29 saya akan kirim surat ke bu Chen," katanya. Dalam rapat Cen Sui Lan meminta Kementerian Perhubungan melihat kesianggunaan masyarakat untuk biaya transportasi. "Saat ini semua naik, termasuk tiket pesawat dan lain lain, jadi tolong pikirkan kembali ini harus apakah harus dinaikkan tanggal 29 ini?," katanya. Sebelumnya penerapan kenaikan tarif ojol ini mundur menjadi 29 Agustus 2022 dari 14 Agustus 2022. Aturan ini tertuang dalam Keputusan Menteri Perhubungan (KM) Nomor 564 Tahun 2022 tentang Pedoman Perhitungan Biaya Jasa Penggunaan Sepeda Motor yang Digunakan Untuk Kepentingan Masyarakat. Hal ini disebabkan untuk penyesuaian dari aplikasi dan menambah masa sosialisasi.

Update post

Gambar 5.24 Halaman Blog Update 2

Selanjutnya, pengguna dapat menekan tombol “Update post” untuk menyimpan perubahan terhadap berita. Tombol tersebut juga akan melakukan permintaan POST kepada rute “/blog-update-2” seperti pada Gambar 5.25. Rute ini akan melakukan perubahan terhadap isi konten berita dan menyimpan perubahan tersebut pada API Blog.


```

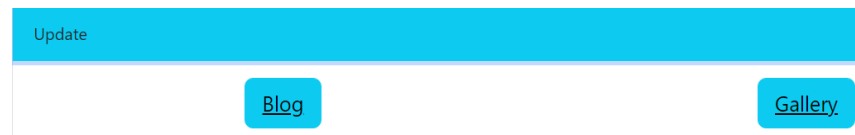
router.post("/blog-update-2", (req, res) => {
  const newpost = {
    title: req.body.createBlogTitle,
    image: req.body.createBlogImage,
    content: req.body.createBlogContent
  }
  article[changeIndex] = newpost;
  res.redirect("/")
})

```

Gambar 5.25 Rute POST Blog Update 2

5.1.7 Rute Gallery Update

Admin dapat mengubah foto yang sudah ditampilkan pada halaman gallery dengan memilih menu “Gallery” pada “Update” yang ada pada tampilan halaman beranda admin. Gambar 5.26 dibawah adalah ilustrasi menu tersebut.



Gambar 5.26 Menu Update Gallery

Setelah memilih Update Gallery, pengguna akan mengakses permintaan GET ke rute “/gallery-update” untuk dapat mengakses halaman Update Gallery dan memilih foto yang ingin diubah seperti pada Gambar 5.27 dan Gambar 5.28 berikut ini.

```

router.get("/gallery-update", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next()
  }
  else {
    res.render("admin/gallery-update", { gallery: gallery });
  }
})

```

Gambar 5.27 Rute GET Gallery Update

Edit / Update Images for Gallery

Select the Images you would like to edit / update

Celestials 12 months illustrated

Continue

Gambar 5.28 Halaman Gallery Update

Selanjutnya, admin dapat memilih foto berdasarkan karikatur foto untuk diubah. Setelah selesai, admin dapat menekan tombol “Continue” untuk membuat permintaan POST ke rute “/gallery-update” seperti pada Gambar 5.29 dibawah ini.

```
router.post("/gallery-update", (req, res) => {
  changeIndex = Number(req.body.arrayIndex);
  res.redirect("/gallery-update-2")
})
```

Gambar 5.29 Rute POST Gallery Update

Selanjutnya, pengguna akan diarahkan ke rute GET “/gallery-update-2” untuk dapat mengubah foto dan karikatur foto yang diinginkan seperti pada Gambar 5.30 berikut.

```
router.get("/gallery-update-2", (req, res, next) => {
  if(!req.user) {
    res.redirect("/not-authorized")
    next()
  }
  else {
    res.render("admin/gallery-update-2", {
      inheritGalleryImage: gallery[changeIndex].url,
      inheritGalleryCaption: gallery[changeIndex].caption
    })
  }
})
```

Gambar 5.30 Rute GET Gallery Update 2

Setelah mengakses rute tersebut, pengguna akan diarahkan ke halaman gallery update 2 dengan ilustrasi seperti Gambar 5.31 berikut ini.

Edit / Update Images for Gallery

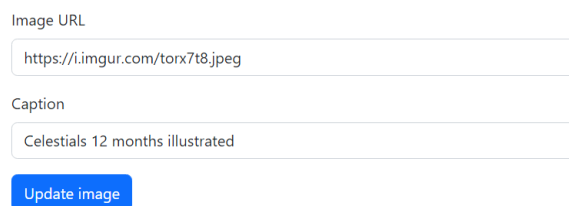


Image URL

Caption

Gambar 5.31 Halaman Gallery Update 2

Selanjutnya, pengguna dapat menekan tombol “Update image” untuk menyimpan perubahan terhadap foto. Tombol tersebut juga akan melakukan permintaan POST kepada rute “/gallery-update-2” seperti pada Gambar 5.32. Rute ini akan melakukan perubahan terhadap foto dan

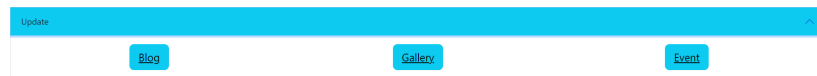
karikatur foto, kemudian menyimpan perubahan tersebut pada API Gallery.

```
router.post("/gallery-update-2", (req, res) => {
  const newpic = {
    url: req.body.createGalleryImage,
    caption: req.body.createGalleryCaption
  }
  gallery[changeIndex] = newpic;
  res.redirect("/");
})
```

Gambar 5.32 Rute POST Gallery Update 2

5.1.8 Rute Event Update

Admin dapat mengubah acara yang sudah ditampilkan pada halaman event dengan memilih menu “Event” pada “Update” yang ada pada tampilan halaman beranda admin. Gambar 5.33 dibawah adalah ilustrasi menu tersebut.



Gambar 5.33 Menu Update Event

Setelah memilih Update Event, pengguna akan mengakses permintaan GET ke rute “/event-update” untuk dapat mengakses halaman Update Event dan memilih acara yang ingin diubah seperti pada Gambar 5.34 dan Gambar 5.35 berikut ini.

```
router.get("/event-update", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next()
  }
  else {
    res.render("admin/event-update", { event: events });
  }
})
```

Gambar 5.34 Rute GET Event Update

Edit / Update Events

Select the event you would like to edit / update

Continue

Gambar 5.35 Halaman Event Update

Selanjutnya, admin dapat memilih acara berdasarkan judul acara untuk diubah. Setelah selesai, admin dapat menekan tombol “Continue” untuk membuat permintaan POST ke rute “/event-update” seperti pada Gambar 5.36 dibawah ini.

```
router.post("/event-update", (req, res) => {
  changeIndex = Number(req.body.eventIndex);
  console.log(changeIndex);
  res.redirect("/event-update-2")
})
```

Gambar 5.36 Rute POST Event Update

Selanjutnya, pengguna akan diarahkan ke rute GET “/event-update-2” untuk dapat mengubah rincian acara yang diinginkan seperti pada Gambar 5.37 berikut.

```
router.get("/event-update-2", (req, res, next) => {
  if(!req.user) {
    res.redirect("/not-authorized")
    next()
  }
  else {
    res.render("admin/event-update-2", {
      eventTitle: events[changeIndex].eventTitle,
      eventAddress: events[changeIndex].eventAddress,
      eventTime: events[changeIndex].eventTime,
      eventDate: events[changeIndex].eventDate,
      eventPoster: events[changeIndex].eventImage,
      eventRegis: events[changeIndex].regis,
      eventDescription: events[changeIndex].eventDescription
    })
  }
})
```

Gambar 5.37 Rute GET Event Update 2

Setelah mengakses rute tersebut, pengguna akan diarahkan ke halaman event update 2 dengan ilustrasi seperti Gambar 5.38 berikut ini.

Create new Event for Event

Event Title

Event Title goes here

Address

Event will be held on this

Time

@ 8 AM to 10 PM

Month

Date

Day

Event Poster URL

https://img.freepress.com/free-press/news-event-poster-template-with-abstract-shapes_13317-1338.jpg?w=7000

Event Registration URL

<https://www.eventbrite.com/Indonesia-jakarta-practice/>

Content

Lipsum ipsum dolor sit amet, consectetur adipiscing elit. Morbi et nisi justo. Pellentesque et amet rhoncus libero, sed vulgatum id. Nullam viverra con nulla consequat. Vivamus accumsan semper eros, eu lectus nunc diam. Consectetur accumsan eget nec ut dapibus. Nullam gravida ultricies odio, in tristique lorem condimentum vitae. Vestibulum varius dapibus leo, sagittis aliquam erat sit amet. Nam quisque, pellentesque sit amet, mauris massa. Duisque porta officio. Donec in ornetiam. Sed malesuada vel lorem sed aliquet. Maecenas id neque enim. Crasport magna nec, dignissim risus. Sed primar volutpat magna, non ornare massa (in)consectetur.

Gambar 5.38 Halaman Event Update 2

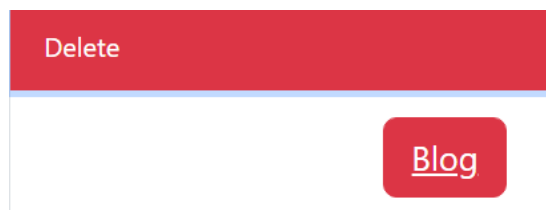
Selanjutnya, pengguna dapat menekan tombol “Update following Event” untuk menyimpan perubahan terhadap acara. Tombol tersebut juga akan melakukan permintaan POST kepada rute “/event-update-2” seperti pada Gambar 5.39. Rute ini akan melakukan perubahan terhadap rincian acara dan menyimpan perubahan tersebut pada API Event.

```
router.post("/event-update-2", (req, res) => {  
  let month = req.body.createEventMonth.slice(5, 7);
```

Gambar 5.39 Rute POST Gallery Update 2

5.1.9 Rute Blog Delete

Admin dapat menghapus berita yang sudah ditampilkan pada halaman blog dengan memilih menu “Blog” pada “Delete” yang ada pada tampilan halaman beranda admin. Gambar 5.40 dibawah adalah ilustrasi menu tersebut.



Gambar 5.40 Menu Delete Blog

Setelah memilih Delete Blog, pengguna akan mengakses permintaan GET ke rute “/blog-delete” untuk dapat mengakses halaman Delete Blog dan memilih berita berdasarkan judul berita yang ingin dihapus seperti pada Gambar 5.41 dan Gambar 5.42 berikut ini.

```
router.get("/blog-delete", (req, res, next) => {  
  if (!req.user) {  
    res.redirect("/not-authorized");  
    next();  
  }  
  else {  
    res.render("admin/blog-delete", { articles: article });  
  }  
})
```

Gambar 5.41 Rute GET Blog Delete

Delete Article for Blog

Select the article you would like to Delete

Sstt.. Ada Tanda-tanda Tarif Ojol Batal Naik Nih

Delete Post

Gambar 5.42 Halaman Blog Delete

Selanjutnya, admin dapat menekan tombol “Delete Post” untuk membuat permintaan POST ke rute “/blog-delete” seperti pada Gambar 5.43 dibawah ini.

```
router.post("/blog-delete", (req, res) => {  
  changeIndex = Number(req.body.articleNumber);  
  article.splice(changeIndex, 1);  
  res.redirect("/");  
})
```

Gambar 5.43 Rute POST Blog Delete

Setelah melakukan langkah diatas, maka program akan menghapus berita dari API dan menunjukkan berita lainnya yang tersisa pada halaman blog.

5.1.10 Rute Gallery Delete

Admin dapat menghapus foto yang sudah ditampilkan pada halaman gallery dengan memilih menu “Gallery” pada “Delete” yang ada pada tampilan halaman beranda admin. Gambar 5.44 dibawah adalah ilustrasi menu tersebut.



Gambar 5.44 Menu Delete Gallery

Setelah memilih Delete Gallery, pengguna akan mengakses permintaan GET ke rute “/gallery-delete” untuk dapat mengakses halaman Delete Gallery dan memilih foto berdasarkan karikatur foto yang ingin dihapus seperti pada Gambar 5.45 dan Gambar 5.46 berikut ini.

```
router.get("/gallery-delete", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next()
  }
  else {
    res.render("admin/gallery-delete", { gallery: gallery });
  }
})
```

Gambar 5.45 Rute GET Gallery Delete

Delete Images for Gallery

Select the Images you would like to delete

Space Illustration.

Delete Image

Gambar 5.46 Halaman Gallery Delete

Selanjutnya, admin dapat menekan tombol “Delete Image” untuk membuat permintaan POST ke rute “/gallery-delete” seperti pada Gambar 5.47 dibawah ini.

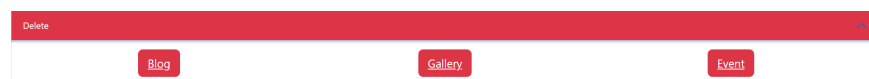
```
router.post("/gallery-delete", (req, res) => {
  changeIndex = Number(req.body.arrayIndex);
  gallery.splice(changeIndex, 1);
  res.redirect("/");
})
```

Gambar 5.47 Rute POST Gallery Delete

Setelah melakukan langkah diatas, maka program akan menghapus foto dari API dan menunjukkan foto lainnya yang tersisa pada halaman gallery.

5.1.11 Rute Event Delete

Admin dapat menghapus acara yang sudah ditampilkan pada halaman event dengan memilih menu “Event” pada “Delete” yang ada pada tampilan halaman beranda admin. Gambar 5.48 dibawah adalah ilustrasi menu tersebut.



Gambar 5.48 Menu Delete Event

Setelah memilih Delete Event, pengguna akan mengakses permintaan GET ke rute “/event-delete” untuk dapat mengakses halaman Delete Event

dan memilih acara berdasarkan judul acara yang ingin dihapus seperti pada Gambar 5.49 dan Gambar 5.50 berikut ini.

```
router.get("/event-delete", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next()
  }
  else {
    res.render("admin/event-delete", { event: events });
  }
})
```

Gambar 5.49 Rute GET Event Delete

Delete Events

Select the event you would like to delete

Delete Event

Gambar 5.50 Halaman Event Delete

Selanjutnya, admin dapat menekan tombol “Delete Event” untuk membuat permintaan POST ke rute “/event-delete” seperti pada Gambar 5.51 dibawah ini.

```
router.post("/event-delete", (req, res) => {
  changeIndex = Number(req.body.arrayIndex);
  events.splice(changeIndex, 1);
  res.redirect("/");
})
```

Gambar 5.51 Rute POST Event Delete

Setelah melakukan langkah diatas, maka program akan menghapus acara dari API dan menunjukkan acara lainnya yang tersisa pada halaman event.

5.1.12 Rute Project Edit

Admin dapat mengubah proyek yang ingin ditampilkan pada halaman beranda dengan memilih menu “Edit Project” yang ada pada tampilan halaman beranda admin. Gambar 5.52 dibawah adalah ilustrasi menu tersebut.

Edit Project

Gambar 5.52 Menu Project Edit

Setelah memilih Edit Project, pengguna akan mengakses permintaan GET ke rute “/ongoingProjectEdit” untuk dapat mengakses halaman Edit Project dan langsung mengubah nama proyek yang ingin diubah seperti pada Gambar 5.53 dan Gambar 5.54 berikut ini.

```
router.get("/ongoingProjectEdit", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next()
  }
  else {
    res.render("admin/ongoing-project-edit", {
      Project1: ongoingProjects[0],
      Project2: ongoingProjects[1],
      Project3: ongoingProjects[2],
      Project4: ongoingProjects[3],
      Project5: ongoingProjects[4],
      Project6: ongoingProjects[5],
      projectLink1: ongoingProjectLinks[0],
      projectLink2: ongoingProjectLinks[1],
      projectLink3: ongoingProjectLinks[2],
      projectLink4: ongoingProjectLinks[3],
      projectLink5: ongoingProjectLinks[4],
      projectLink6: ongoingProjectLinks[5] } );
  }
})
```

Gambar 5.53 Rute GET Project Edit

Edit contents for Project

Ongproj 1 #	Ongproj 2 #
Ongproj 3 #	Ongproj 4 #
Ongproj 5 #	Ongproj 6 #

Save Changes

Gambar 5.54 Halaman Event Delete

Selanjutnya, admin dapat mengubah nama proyek dan memasukkan link yang menuju proyek tersebut jika diperlukan. Dengan menekan tombol “Save Changes”, sistem akan membuat permintaan POST ke rute “/ongoingProjectEdit” seperti pada Gambar 5.55 dibawah ini.

```

router.post("/ongoingProjectEdit", (req, res) => {
  ongoingProjects[0] = req.body.proj1
  ongoingProjects[1] = req.body.proj2
  ongoingProjects[2] = req.body.proj3
  ongoingProjects[3] = req.body.proj4
  ongoingProjects[4] = req.body.proj5
  ongoingProjects[5] = req.body.proj6
  ongoingProjectLinks[0] = req.body.projLink1
  ongoingProjectLinks[1] = req.body.projLink2
  ongoingProjectLinks[2] = req.body.projLink3
  ongoingProjectLinks[3] = req.body.projLink4
  ongoingProjectLinks[4] = req.body.projLink5
  ongoingProjectLinks[5] = req.body.projLink6
  res.redirect("/")
})

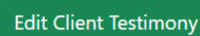
```

Gambar 5.55 Rute POST Project Edit

Setelah melakukan langkah diatas, maka program akan menyimpan perubahan nama dan link proyek dari API dan menampilkan perubahannya pada halaman beranda.

5.1.13 Rute Testimony Edit

Admin dapat mengubah testimoni yang ingin ditampilkan pada halaman beranda dengan memilih menu “Edit Client Testimony” yang ada pada tampilan halaman beranda admin. Gambar 5.56 dibawah adalah ilustrasi menu tersebut.



Gambar 5.56 Menu Testimony Edit

Setelah memilih Edit Testimony, pengguna akan mengakses permintaan GET ke rute “/testimonyEdit” untuk dapat mengakses halaman Edit Testimony dan langsung mengubah nama, ulasan, dan nama klien yang ingin diubah seperti pada Gambar 5.57 dan Gambar 5.58 berikut ini.

```

router.get("/testimonyEdit", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next()
  }
  else {
    res.render("admin/testimony-edit", {
      clientImage1: testimony[0].cliPic,
      clientName1: testimony[0].cliName,
      clientReview1: testimony[0].cliRev,
      clientImage2: testimony[1].cliPic,
      clientName2: testimony[1].cliName,
      clientReview2: testimony[1].cliRev,
      clientImage3: testimony[2].cliPic,
      clientName3: testimony[2].cliName,
      clientReview3: testimony[2].cliRev
    });
  }
})

```

Gambar 5.57 Rute GET Testimony Edit

Edit Testimonials Section

Testimony 1

Insert Link to Image:
https://img.freepik.com/premium-photo/indonesian-man-smiling_86639-992.jpg

Name 1:
 Chandia Kilau

Review 1:
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi accumsan ex semper neque aliquam tempus.

Testimony 2

Insert Link to Image:
https://img.freepik.com/premium-photo/asian-woman-white-t-shirt-stand-smiling-with-braces-white-background_2034-2735.jpg

Name 2:
 Kayla Ruffiani

Review 2:
 Vestibulum urna sapien, tristique quis euismod sed, elementum id erat. Nunc velit orci, congue ut sceler.

Testimony 3

Insert Link to Image:
https://www.accenture.com/t2022061470311062_w_/d4-en/_acmedia/Accenture/Redesign/Assets/DocCom/Images/Local/General/143/Accenture-Jayent-400x400.jpg

Name 3:
 John Farquah

Review 3:
 Aliquam eu sollicitudin leo. Aliquam erat volutpat. Etiam pharetra, massa nec ultrices sollicitudin enim.

Save Changes

Gambar 5.58 Halaman Testimony Edit

Selanjutnya, admin dapat mengubah nama klien, link foto klien, dan ulasan klien yang ingin diubah jika diperlukan. Dengan menekan tombol “Save Changes”, sistem akan membuat permintaan POST ke rute “/testimonyEdit” seperti pada Gambar 5.59 dibawah ini.

```

router.post("/testimonyEdit", (req, res) => {
  testimony[0].cliPic = req.body.clientImage1
  testimony[0].cliName = req.body.clientName1
  testimony[0].cliRev = req.body.clientReview1
  testimony[1].cliPic = req.body.clientImage2
  testimony[1].cliName = req.body.clientName2
  testimony[1].cliRev = req.body.clientReview2
  testimony[2].cliPic = req.body.clientImage3
  testimony[2].cliName = req.body.clientName3
  testimony[2].cliRev = req.body.clientReview3

  res.redirect("/")
})

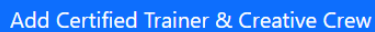
```

Gambar 5.59 Rute POST Testimony Edit

Setelah melakukan langkah diatas, maka program akan menyimpan perubahan nama, link foto, dan ulasan klien dari API dan menampilkan perubahannya pada halaman beranda.

5.1.14 Rute Add Certified Trainer & Creative Crew

Admin dapat menambah profil yang ingin ditampilkan pada halaman beranda dengan memilih menu “Add Certified Trainer & Creative Crew” yang ada pada tampilan halaman beranda admin. Gambar 5.60 dibawah adalah ilustrasi menu tersebut.



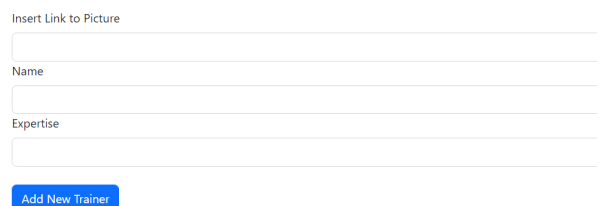
Gambar 5.60 Menu Add Certified Trainer & Creative Crew

Setelah memilih Add Certified Trainer & Creative Crew, pengguna akan mengakses permintaan GET ke rute “/trainerAdd” untuk dapat mengakses halaman Add Certified Trainer & Creative Crew dan langsung menambah profil dengan memasukkan nama, link foto, dan keahlian seperti pada Gambar 5.61 dan Gambar 5.62 berikut ini.

```
router.get("/trainerAdd", (req, res, next) => {  
  if (!req.user) {  
    res.redirect("/not-authorized");  
    next()  
  }  
  else {  
    res.render("admin/trainerAdd")  
  }  
})
```

Gambar 5.61 Rute GET Add Certified Trainer & Creative Crew

Add New Certified Trainer & Creative Crew



Gambar 5.62 Halaman Add Certified Trainer & Creative Crew

Selanjutnya, admin dapat menambah nama, link foto, dan keahlian kru yang ingin ditambah jika diperlukan. Dengan menekan tombol “Add New Trainer”, sistem akan membuat permintaan POST ke rute “/trainerAdd” seperti pada Gambar 5.63 dibawah ini.

```

router.post("/trainerAdd", (req, res) => {
  const trainerTemp = {
    image: req.body.trainerPic,
    name: req.body.trainerName,
    expertise: req.body.trainerExpertise
  }

  certifiedTrainer.unshift(trainerTemp)
  res.redirect("/")
})


```

Gambar 5.63 Rute POST Add Certified Trainer & Creative Crew

Setelah melakukan langkah diatas, maka program akan menambahkan dan menyimpan nama, link foto, dan keahlian kru pada API dan menampilkan perubahannya pada halaman beranda.

5.1.15 Rute Remove Certified Trainer & Creative Crew

Admin dapat menghapus profil yang ingin ditampilkan pada halaman beranda dengan memilih menu “Remove Certified Trainer & Creative Crew” yang ada pada tampilan halaman beranda admin. Gambar 5.64 dibawah adalah ilustrasi menu tersebut.



Gambar 5.64 Menu Remove Certified Trainer & Creative Crew

Setelah memilih Add Certified Trainer & Creative Crew, pengguna akan mengakses permintaan GET ke rute “/trainerRemove” untuk dapat mengakses halaman Remove Certified Trainer & Creative Crew dan langsung menghapus profil dengan memilih nama kru seperti pada Gambar 5.65 dan Gambar 5.66 berikut ini.

```

router.get("/trainerRemove", (req, res, next) => {
  if (!req.user) {
    res.redirect("/not-authorized");
    next()
  }
  else {
    res.render("admin/trainerRemove", {
      profile: certifiedTrainer
    })
  }
})

```

Gambar 5.65 Rute GET Remove Certified Trainer & Creative Crew

Remove Certified Trainer & Creative Crew

Select Trainer Profile

Person Name

Delete Profile

Gambar 5.66 Halaman Remove Certified Trainer & Creative Crew

Selanjutnya, admin dapat memilih nama kru dan menghapusnya dengan menekan tombol “Delete Profile”. Selanjutnya sistem akan membuat permintaan POST ke rute “/trainerRemove” seperti pada Gambar 5.67 dibawah ini.

```
router.post("/trainerRemove", (req, res) => {  
  changeIndex = Number(req.body.trainerProfile);  
  certifiedTrainer.splice(changeIndex, 1);  
  res.redirect("/");  
})
```

Gambar 5.67 Rute POST Remove Certified Trainer & Creative Crew

Setelah melakukan langkah diatas, maka program akan menghapus profil kru yang dipilih dari API dan menampilkan sisanya pada halaman beranda.

```
<!-- Trainer Section ----->  
  
<% if (trainer.length !=0) { %>  
  <section id="Trainer_Section">  
    <div class="container-fluid evenBG centerItems">  
      <h2>Certified Trainers & Creative Crew</h2>  
      <div class="scrollable">  
        <% for(let i=0; i < trainer.length; i++){ %>  
          <!-- <div class="col-lg-3 col-md-6 col-sm-12"> -->  
          <div class="card text-bg-nazmaOrange trainerCard">  
              
            <div class="card-body">  
              <p class="card-text">  
                <%= trainer[i].name %><br /><strong>  
                  <%= trainer[i].expertise %>  
                </strong>  
              </p>  
            </div>  
          </div>  
          <!-- </div> -->  
        <% } %>  
      </div>  
    </section>  
  <% } %>
```

Gambar 5.68 Alur pada bagian Trainer

Apabila API tidak memiliki profil untuk ditunjukkan pada bagian Add Certified Trainer & Creative Crew, maka bagian pada website tersebut akan disembunyikan dari halaman beranda.

5.2 Pengujian

Pada tahap pengujian ini, penulis membandingkan hasil pengujian yang diharapkan pada Bab IV dengan hasil yang didapat setelah melakukan pengembangan pada backend website. Tolak ukur pengujian yang digunakan penulis adalah Black Box Testing dan System Usability Scale. Hasil dari pengujian website dapat dilihat pada Tabel 5.1 dan Tabel 5.2 berikut.

Tabel 5.1 Hasil Pengujian Black Box

No	Skenario Pengujian	Kasus Pengujian	Hasil yang diharapkan	Hasil Pengujian
1	Masuk sebagai Administrator	Memasukkan username dan password, kemudian menekan tombol <i>sign in</i>	Sistem akan mengarahkan user ke halaman administrator apabila username dan password benar, dan akan tetap berada di halaman login apabila username atau password salah	Sistem berhasil melakukan fungsi <i>sign in</i> menggunakan username, password, dan <i>environment variable</i> sebagai otentikasi device yang melakukan login
2	Membuat berita baru pada blog	Memasukkan judul, foto, dan isi berita, kemudian menekan tombol submit	Sistem akan menambahkan judul, foto, dan isi berita menjadi berita baru, dan berita akan terlihat pada halaman blog	Sistem gagal menambahkan file foto, namun berhasil menambahkan foto menggunakan link foto. Berita terbaru berhasil dimunculkan pada halaman blog
3	Membuat foto baru pada gallery	Memasukkan foto dan karikatur foto, kemudian menekan tombol submit	Sistem akan menambahkan foto dan karikatur terbaru pada halaman gallery di paling kiri atas.	Sistem gagal menambahkan file foto, namun berhasil menampilkan foto menggunakan alamat link foto. Foto terbaru berhasil dimunculkan pada halaman gallery
4	Membuat acara pada event	Memasukkan judul acara, alamat, waktu, bulan, tanggal,	Acara akan muncul pada halaman event pada urutan paling atas dengan rincian	Sistem gagal menambahkan file foto poster, namun berhasil

No	Skenario Pengujian	Kasus Pengujian	Hasil yang diharapkan	Hasil Pengujian
		foto poster, link pendaftaran, dan deskripsi acara.	acara sesuai dengan data yang sudah dimasukkan	menampilkan foto poster menggunakan link foto poster. Acara terbaru berhasil dimunculkan pada halaman event.
5	Mengubah berita pada blog	Memilih berita mana yang ingin diubah, kemudian dapat mengubah judul, foto, dan isi berita.	Isi berita pada blog akan berubah tanpa mengubah urutan berita.	Sistem berhasil mengubah isi berita menggunakan judul berita untuk memilih objek berita dalam API blog
6	Mengubah foto pada gallery	Memilih foto mana yang ingin diubah berdasarkan karikturnya, kemudian dapat mengubah foto dan/atau karikatur dari foto tersebut.	Foto yang diubah akan tetap berada pada urutannya	Sistem berhasil mengubah link foto dan karikatur foto menggunakan karikatur foto untuk memilih objek foto yang ingin diubah dalam API gallery
7	Mengubah acara pada event	Memilih acara dari API event berdasarkan judul acara, kemudian dapat mengubah seluruh isi acara dan menyimpan perubahannya.	Acara yang diubah akan tetap berada pada urutannya.	Sistem berhasil merubah rincian acara menggunakan judul acara untuk memilih objek acara dalam API event yang akan diubah.
8	Menghapus berita pada blog	Memilih berita berdasarkan judul dan menghapusnya.	Berita akan hilang dari halaman blog	Sistem berhasil menghapus objek berita dalam API blog menggunakan judul berita untuk memilih berita yang perlu dihapus
9	Menghapus foto dari gallery	Memilih foto berdasarkan karikatur untuk menghapusnya.	Foto akan hilang dari halaman gallery	Sistem berhasil menghapus foto dari objek foto dalam API gallery menggunakan karikatur foto untuk memilih foto yang perlu dihapus

No	Skenario Pengujian	Kasus Pengujian	Hasil yang diharapkan	Hasil Pengujian
10	Menghapus acara dari event	Memilih acara berdasarkan judulnya dan menghapusnya	Acara akan hilang dari halaman event	Sistem berhasil menghapus acara dari objek acara dalam API event menggunakan judul acara untuk memilih acara yang perlu dihapus
11	Mengubah tampilan project	Mengubah proyek yang ingin ditampilkan berupa teks	Dapat mengubah dan menampilkan 6 proyek untuk ditunjukkan pada halaman beranda	Sistem berhasil mengubah 6 nama dan link proyek yang ditampilkan pada halaman beranda
12	Mengubah testimoni client	Mengubah testimoni client yang ingin ditampilkan berupa testimoni, foto client, dan nama client	Dapat mengubah dan menampilkan 3 testimoni client yang ingin ditampilkan pada halaman beranda	Sistem berhasil mengubah dan menampilkan 3 testimoni klien. Sistem berhasil mengubah foto klien menggunakan link foto
13	Menambah kru yang ingin ditampilkan	Menambahkan kru yang ingin ditampilkan berupa foto, nama, dan keahliannya	Dapat menambahkan kru yang ingin ditampilkan pada halaman beranda tanpa batas maksimal	Sistem berhasil menambahkan profil kru tanpa batas maksimal. Foto kru ditampilkan menggunakan link foto.
14	Menghapus kru yang ingin ditampilkan	Menghapus kru yang ingin ditampilkan berdasarkan nama kru	Dapat menghapus kru dari halaman beranda	Sistem berhasil menghapus profil kru dari tampilan menggunakan nama kru untuk memilih profil kru yang ingin dihapus.

Tabel 5.2 Hasil Pengujian System Usability Scale

No	Pertanyaan	Skor
1	Saya berpikir akan menggunakan backend website ini lagi	4.7
2	Saya merasa backend website ini rumit untuk digunakan	1.7

No	Pertanyaan	Skor
3	Saya merasa backend website ini mudah digunakan	4.5
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan backend website ini	2.5
5	Saya merasa fitur-fitur backend website ini berjalan dengan semestinya	5
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada backend website ini)	2.4
7	Saya merasa orang lain akan memahami cara menggunakan backend website ini dengan cepat	4.5
8	Saya merasa backend website ini membingungkan	2.2
9	Saya merasa tidak ada hambatan dalam menggunakan backend website ini	4.6
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan backend website ini	2.1

Pengujian System Usability Scale dilakukan dengan menggunakan kuesioner berisi 10 pertanyaan pada Tabel 5.2 dan diisi oleh 9 orang responden dari karyawan/karyawati NaZMa Office.

Perhitungan skor dilakukan dengan cara memisahkan skor pertanyaan dengan nomor ganjil (X) dari nomor genap (Y). Berikut merupakan kalkulasi hasil System Usability Scale yang dilakukan.

$$X = (\text{rata-rata skor pertanyaan dengan nomor ganjil}) - 5$$

$$Y = 25 - (\text{rata-rata skor pertanyaan dengan nomor genap})$$

$$\text{SUS Score} = (X + Y) * 2.5$$

Pada Tabel 5.3 berikut ini, dapat dilihat interpretasi dari hasil akhir System Usability Scale.

Tabel 5.3 Rentang Nilai System Usability Scale

SUS Score	Grade	Adjective Rating
> 80.3	A	Excellent
68 – 80.3	B	Good
68	C	Okay

SUS Score	Grade	Adjective Rating
51 – 68	D	Poor
< 51	E	Awful

Hasil SUS yang didapat dari pengujian backend website NaZMa Office adalah sebagai berikut.

$$X = (23.5) - 5$$

$$Y = 25 - (11.1)$$

$$\text{SUS Score} = (18.5 + 13.9) * 2.5$$

$$\text{SUS Score} = 81.1$$

Dengan ini, maka didapat bahwa aplikasi yang dikembangkan mendapat skor A dengan hasil EXCELLENT.

5.3 Pembahasan

Melalui hasil pengujian fungsionalitas pada Tabel 5.1 diperoleh hasil bahwa backend website NaZMa Office yang dikembangkan menggunakan NodeJS dengan Express ini memiliki tingkat keberhasilan sebesar 78.57% pada metode pengujian Black Box dan nilai A pada metode pengujian System Usability Scale. Tingkat keberhasilan tersebut diperoleh dari berhasilnya 11 dari 14 pengujian pada metode Black Box yang dilakukan dan skor sebesar 81.1 pada metode pengujian System Usability Scale yang dilakukan. Pengujian sistem yang gagal terjadi karena sistem tidak dapat menyimpan berkas foto, namun implementasi sistem tetap berhasil karena menggunakan alternatif berupa link dari foto. Maka dari itu semua foto yang ingin digunakan pada sistem harus di hosting terlebih dahulu pada web hosting foto, kemudian link foto dapat digunakan untuk ditampilkan pada website.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Kesimpulan dari pelaksanaan kerja praktik yang penulis lakukan adalah sebagai berikut.

1. Backend website dapat dibuat berupa API menggunakan NodeJS dengan Express. Namun API yang dibuat hanya dapat menampung data/informasi berbasis teks/string.
2. Perusahaan dapat menyampaikan konten dan informasi kepada pengunjung website dengan melakukan login pada sisi administrator website dan melakukan fungsi CRUD pada API yang telah dibuat untuk menampilkan hasil konten dan informasi pada website yang dapat diakses oleh publik.

6.2 Saran

1. Pada API Blog sebaiknya dapat menggunakan modul *lodash* dari NodeJS untuk membuat rute URL setiap artikel menggunakan judul artikel tersebut.
2. Backend sebaiknya dikembangkan menggunakan sistem manajemen basis data agar berkas yang tidak berbasis teks/string tidak perlu ditempatkan pada aplikasi pihak ketiga untuk dapat digunakan.
3. API yang telah dibuat dapat dikembangkan lebih lanjut dan dirilis secara *open source* sehingga siapapun yang ingin mengembangkan lebih lanjut cukup mendapatkan akses untuk melakukan permintaan POST dan GET terhadap API yang sudah dibuat penulis.

REFERENSI

- [1] Novia R. Risnu. "Company Profile CV. Nabila Zafira Mahalia (NaZMa Office)." Yogyakarta, 2022
- [2] Prayitno, Agus. "Pemanfaatan sistem informasi perpustakaan digital berbasis website untuk para penulis." *Indonesian Journal on Software Engineering (IJSE)* vol. 1, no. 1, 2015
- [3] Firdaus, A., Widodo, S., Sutrisman, A., Gading, S., & Mardiana, R, "Rancang Bangun Sistem Informasi Perpustakaan Menggunakan Web Service Pada Jurusan Teknik Komputer Polsri." *INFORMANIKA*, vol. 5, no 2, 2019.
- [4] N. Wang, X. Chen, G. Song, and H. Parsaei, "Using node-HTTP-proxy for remote experiment data transmission traversing firewall," *International Journal of Online Engineering*, vol. 11, no. 2, 2015, doi: 10.3991/ijoe.v11i2.4443.
- [5] A. Mardan, *Express.js Guide*, Lean Publishing, 2014.
- [6] Arhandi, P. P, "Pengembangan Sistem Informasi Perijinan Tenaga Kesehatan dengan Menggunakan Metode Back End dan Front End." *Jurnal Teknologi Informasi: Teori, Konsep, Dan Implementasi*, vol. 7, no. 1, 2016.
- [7] "Analisis metode waterfall Untuk Pengembangan Sistem informasi." [Online]. Available:
https://www.researchgate.net/publication/346397070_Analisis_Metode_Waterfall_Untuk_Pengembangan_Sistem_Informasi. [Accessed: 25-Oct-2022].
- [8] S. Nidhra, "Black box and white box testing techniques - A literature review," *International Journal of Embedded Systems and Applications*, vol. 2, no. 2, pp. 29–50, 2012.

- [9] R. L. Wakefield, K. L. Wakefield, J. Baker, and L. C. Wang, “How website socialness leads to website use,” *Eur. J. Inf. Syst.*, vol. 20, no. 1, pp. 118–132, Jan. 2011, doi: 10.1057/ejis.2010.47.

Lampiran A. TOR (Term of Reference)

1. Latar Belakang

Pada dunia bisnis modern, internet sudah menjadi pintu keluar masuk untuk berbagai macam informasi dan konten. Bahkan akan terasa aneh dan asing apabila sebuah bisnis tidak memanfaatkan berbagai macam platform yang umumnya akan digunakan oleh orang banyak untuk menarik perhatian dari calon pelanggan. Website adalah salah satu platform yang dapat diakses oleh semua orang. Website ini bila dimanfaatkan dengan baik dapat menjadi media bisnis untuk memperkenalkan jasa kepada calon pelanggan.

Sebagai calon pelanggan, kita biasanya akan merasa aneh apabila sebuah bisnis tidak memiliki website resmi bukan? Calon pelanggan akan merasa apakah perusahaan yang ingin dilibatkan merupakan perusahaan yang kredibel atau tidak apabila sebuah perusahaan tidak memiliki website, khususnya untuk perusahaan IT.

CV. Nabila Zafira Mahalia atau dapat disebut juga dengan NaZMa Office adalah perusahaan swasta yang bergerak dibidang IT dan *Management Consulting* yang dipersiapkan secara profesional baik dari segi kualitas kelembagaan, Sumber Daya Manusia, maupun produk atau layanan yang disediakan, dengan mengedepankan kepuasan pelanggan.

CV.NaZMa Office sebagai perusahaan yang bergerak di bidang IT masih belum memiliki website perusahaan untuk mendukung pemasaran serta penyampaian informasi terkait. Selama ini, NaZMa Office mengandalkan media sosial umum untuk menunjukkan konten-konten informatifnya. Karena itu, penulis melakukan kerja praktik proyek pembuatan website resmi perusahaan. Pembuatan website dilakukan dengan arahan dari pihak perusahaan NaZMa Office. Diharapkan setelah website ini selesai lalu dilakukan hosting dan publikasi dapat menyelesaikan masalah perusahaan dalam penyampaian informasi.

Proyek yang dikerjakan oleh penulis merupakan backend dari website NaZMa Office. Tujuan pembuatan backend website ini adalah untuk

mengembangkan fitur create, read, update, dan delete pada admin untuk dapat menyampaikan informasi kepada pengunjung website melalui halaman blog, event, atau gallery.

2. Tujuan Pekerjaan

Tujuan dari kerja praktik di CV. NaZMa Office adalah :

1. Membuat API untuk backend website NaZMa Office yang memiliki fitur create, read, update, dan delete.
2. Membuat backend pada website NaZMa Office untuk menyampaikan konten dan informasi.

3. Lingkup Pekerjaan

Ruang lingkup penulis dalam pelaksanaan kerja praktik sebagai berikut:

1. Kerja praktik dilaksanakan dengan rentang waktu 40 hari dari tanggal 18 Juli 2022 s/d 9 September 2022.
2. Pembuatan website dilakukan secara tim yang beranggotakan 2 orang.
3. Pengerjaan tugas harian dari kantor dikerjakan secara individu.
4. Pembuatan website pada backend menggunakan NodeJS dan ExpressJS.
- 5.

4. Metodologi

Metodologi pencapaian pekerjaan selama aktivitas kuliah praktik dicapai dengan memanfaatkan Waterfall Software Development Life Cycle atau SDLC Waterfall. Penulis mengumpulkan informasi tentang apa saja yang dibutuhkan dan diinginkan atasan terkait proyek website yang akan dikerjakan melalui meeting, kemudian mengimplementasikan permintaan tersebut.

Setelah itu, penulis akan mempresentasikan hasil implementasi permintaan atasan untuk memverifikasi apakah permintaan yang diimplementasikan sudah sesuai sebagaimana yang diinginkan atasan. Apabila ada implementasi yang salah, maka akan dibenarkan.

5. Hasil Pekerjaan

Pekerjaan yang dilakukan penulis saat pelaksanaan kerja praktik adalah:

1. Membuat perancangan backend website NaZMa Office.
2. Membuat hasil pengujian menggunakan black box.

6. Jadwal Kerja

Berikut merupakan jadwal pelaksanaan Kerja Praktik di NaZMa Office :

Hari Kerja : Senin s.d Jum'at

Jam Kerja : 08.00 – 16.00 WIB

Waktu Mulai : 18 Juli 2022

Waktu Selesai : 9 September 2022

Rincian dan jadwal Kerja Praktik yang dilakukan oleh penulis selama 40 hari terlampir pada Presensi dan *Logsheets* Praktek Kerja Lapangan.

7. Kerahasiaan dan Perlindungan Informasi

PERJANJIAN KERAHASIAAN (NON DISCLOSURE AGREEMENT)

ANTARA

CV NABILA ZAFIRA MAHALIA

DENGAN

MAHASISWA TEKNIK INFORMATIKA

INSTITUT TEKNOLOGI SUMATERA

DALAM KERJASAMA

**PEMBUATAN BACKEND PADA WEBSITE NAZMA OFFICE
MENGUNAKAN NODEJS DENGAN EXPRESS**

Perjanjian kerahasiaan (Non Disclosure Agreement) ini (untuk selanjutnya disebut sebagai “Perjanjian Kerahasiaan” dibuat dan ditandatangani pada tanggal 10 Oktober 2022 oleh dan antara:

I. PIHAK CV NAZMA OFFICE berkedudukan Jl. Selokan Mataram Pogung Dalangan SIA XVI No. 16 RT 11/RW 50 Sinduadi, Mlati, Sleman, Daerah Istimewa Yogyakarta, selanjutnya disebut **PIHAK PERTAMA**.

II. MAHASISWA TEKNIK INFORMATIKA ITERA, adalah mahasiswa Teknik Informatika Angkatan 2019 di Jl. Ratu Dibalau, Way Huwi, Kec. Jati Agung, Kabupaten Lampung Selatan. Dalam gal ini diwakili oleh **GEIZKA ROZILIA RUICOSTA** sebagai mahasiswa, oleh karena itu sah bertindak untuk dan atas nama mahasiswa Teknik Informatika **ITERA**, untuk selanjutnya disebut **“PIHAK KEDUA”**. **PIHAK PERTAMA** dan **PIHAK KEDUA** selanjutnya disebut sebagai **“PARA PIHAK”**. **PARA PIHAK** dengan ini menjelaskan dan menyatakan sebagai berikut :

- a. Bahwa **PIHAK PERTAMA** adalah kantor pemerintahan yang membidangi urusan komunikasi dan informatika
- b. Bahwa **PIHAK KEDUA** adalah mahasiswa
- c. Bahwa **PIHAK PERTAMA** bermaksud untuk mengungkapkan suatu informasi yang bersifat rahasia kepada **PIHAK KEDUA** berhubungan dengan teknologi, produk, layanan-layanan **PIHAK PERTAMA** secara umum. (untuk selanjutnya disebut sebagai "Pokok Permasalahan").
- d. Bahwa **PARA PIHAK** menjamin bahwa informasi yang diberikan dan disampaikan baik secara lisan, tertulis, grafik atau yang disampaikan melalui media elektronik atau informasi dalam bentuk lainnya selama berlangsung nya pembicaraan dan/atau selama pelaksanaan kerjasama dijaga kerahasiaannya dari pihak ketiga dari **PARA PIHAK** dan akan dipergunakan hanya untuk kepentingan **PARA PIHAK**.

Berdasarkan hal-hal tersebut diatas **PARA PIHAK** bersedia untuk memberikan dan menerima informasi tersebut berdasarkan ketentuan sebagai berikut:

1. Informasi berarti informasi yang tertulis atau dokumentasi yang disimpan dan dapat dibaca oleh mesin, atau informasi lainnya atau data dalam bentuk apapun yang :

a) Berhubungan dengan Pokok Permasalahan dalam Perjanjian Kerjasama, b) Diterima oleh **PARA PIHAK**, dan

c) Ditentukan sebagai sesuatu rahasia, atau memiliki suatu nilai yang serupa atau dimana pihak yang mengungkapkan informasi tersebut menyatakannya secara tertulis pada saat menyampaikannya kepada pihak yang menerima, agar diperlakukan sebagai sesuatu milik atau bilamana sifat dari data atau informasi adalah serupa dengan itu maka 44 pihak yang menerima harus memperlakukan data atau informasi tersebut sebagai sesuatu yang rahasia. Pengungkapan suatu informasi secara lisan akan dianggap sebagai milik pihak yang mengungkapkan bilamana pihak yang mengungkapkan tersebut secara lisan menyatakan bahwa informasi yang diungkapkan tersebut merupakan miliknya ataupun sesuatu yang rahasia pada saat pengungkapan atau jika sifat dari informasi tersebut adalah serupa maka pihak yang menerima pengungkapan informasi tersebut juga harus memperlakukannya sebagai suatu informasi yang rahasia pula.

2. **PIHAK KEDUA** akan mempergunakan Informasi milik **PIHAK PERTAMA** tersebut hanya untuk tujuan pelaksanaan Perjanjian Kerjasama.
3. Perjanjian Kerahasiaan ini berlaku sejak tanggal sebagaimana tersebut di atas, semua informasi-informasi rahasia yang telah diberikan oleh **PARA PIHAK** akan selalu menjadi rahasia dan mengikat **PARA PIHAK** tanpa memandang keputusan dan/atau berakhirnya Perjanjian Kerjasama ("Periode Perjanjian Kerahasiaan").
4. **PIHAK KEDUA** tidak akan mengungkapkan kepada pihak ketiga atas sesuatu informasi rahasia yang telah diterimanya berdasarkan Perjanjian Kerahasiaan ini baik secara keseluruhan maupun sebagian selama Periode Perjanjian Kerahasiaan ini dan **PIHAK KEDUA** menyatakan bahwa kewajiban tentang kerahasiaan tersebut akan tetap berlaku dalam hal pengakhiran Perjanjian Kerahasiaan ini.

5. **PIHAK KEDUA** akan :

a. Mempergunakan paling tidak dengan tingkat usaha perlindungan terhadap informasi tersebut sebagaimana pihak yang menerima akan melakukan nya untuk menjaga informasi rahasia miliknya sepanjang bahwa tingkat perlindungan yang diberikan cukup layak untuk mencegah adanya pengungkapan yang tidak tepat atau penggunaan yang tidak sah atas informasi tersebut;

b. Membatasi akses terhadap Informasi tersebut kepada para rekan mahasiswa yang tidak berkepentingan untuk mengetahui Informasi tersebut dan memberitahukan kepada para rekan mahasiswa yang telah memperoleh informasi tersebut mengenai kewajiban-kewajiban mereka menurut Perjanjian Kerahasiaan ini, dan Atas penemuan sesuatu pengungkapan yang tidak tepat atau penggunaan yang tidak sah atas Informasi tersebut maka dengan segera akan melakukan usaha-usaha yang layak untuk mencegah sesuatu pengungkapan atau penggunaan yang tidak benar lebih lanjut terhadap informasi tersebut.

c. Apabila **PIHAK KEDUA** terbukti melakukan pelanggaran hukum dengan mengungkapkan informasi rahasia yang telah diterima nya maka **PIHAK KEDUA** dengan segera harus memberitahukan nya kepada **PIHAK PERTAMA** melalui pemberitahuan resmi secara tertulis. Jika **PIHAK PERTAMA** menolak untuk mempermasalahkan pengungkapan tersebut, maka **PIHAK KEDUA** dapat melanjutkan untuk mengungkapkan informasi tersebut atas pilihannya sendiri.

d. Atas permintaan dari **PIHAK PERTAMA** atau setelah berakhirnya perjanjian kerjasama, **PIHAK KEDUA** akan

berhenti untuk mempergunakan informasi yang diterimanya dari **PIHAK PERTAMA** dan akan memusnahkan semua Informasi tersebut, termasuk segala salinan dari Informasi tersebut, dan akan melengkapinya dengan suatu pemberitahuan resmi secara tertulis mengenai pemusnahan terhadap Informasi tersebut kepada **PIHAK PERTAMA** atau atas permintaan **PIHAK PERTAMA**, dan **PIHAK KEDUA** akan mengembalikan informasi tersebut kepada **PIHAK PERTAMA**

e. Semua informasi maupun data dalam bentuk apapun, termasuk di dalamnya tidak terbatas pada dokumen-dokumen, gambar-gambar, spesifikasi- spesifikasi, prototipe-prototipe, contoh-contoh, dan hal lain-lain yang serupa yang di dapat kan dalam Perjanjian 46 Kerahasiaan ini oleh **PARA PIHAK** akan tetap menjadi milik dari **PIHAK PERTAMA** dan semua Hak atas Kekayaan Intelektual terhadap informasi tersebut akan tetap menjadi milik **PIHAK PERTAMA**.

6. Tidak satupun yang terdapat dalam Perjanjian Kerahasiaan ini yang merupakan:

a. Sebagai kewajiban bagi **PIHAK PERTAMA** untuk mengungkapkan atau bagi **PIHAK KEDUA** untuk menerima Informasi tersebut, kecuali jika informasi tersebut diperlukan untuk pelaksanaan Perjanjian Kerjasama.

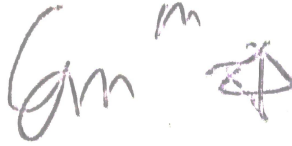
b. Memberikan jaminan kepada salah satu pihak suatu izin, baik yang dinyatakan secara langsung ataupun yang tidak secara langsung, untuk sesuatu hak paten, hak cipta, rahasia dagang atau hak kekayaan intelektual lainnya yang pada saat ini atau selanjutnya dimiliki, diperoleh atau diberikan ijin oleh pihak lainnya.

7. **PIHAK PERTAMA** menjamin bahwa dia memiliki hak untuk memindahkan atau sebaliknya mengungkapkan kepada **PIHAK KEDUA** suatu informasi yang diungkapkan kepada **PIHAK KEDUA** dalam Perjanjian Kerahasiaan ini. **PIHAK PERTAMA** tidak akan memberikan jaminan lainnya, baik yang dinyatakan secara langsung ataupun yang tidak langsung, berkaitan dengan informasi yang disampaikan menurut Perjanjian Kerahasiaan ini. Perjanjian Kerahasiaan ini berisikan keseluruhan kesepakatan antara **PARA PIHAK** dan tidak dapat diubah dengan cara apapun kecuali dengan perubahan secara tertulis yang ditandatangani oleh **PARA PIHAK**.

Demikian Perjanjian Kerahasiaan ini ditandatangani, sebagaimana dimaksud pada awal Perjanjian Kerahasiaan ini.

Terms of reference ini telah dibaca dan disetujui oleh:

Pihak Mahasiswa



Geizka Rozilia Ruicosta

NIM. 119140114

Pihak Instansi



NaZMa
Nabila Zafira Mahalia. CV

Wiji Nurastuti, SE, MT

Lampiran B. Presensi dan *Logsheets* Praktek Kerja Lapangan

Yang bertanda tangan di bawah ini,

Nama : Wiji Nurastuti, SE, MT, AWP

Jabatan : Direktur Operasional.

Perusahaan/Instansi : CV.Nabila Zafira Mahalia (NaZMa Office)

Menyatakan mahasiswa berikut:

Nama : Geizka Rozilia Ruicosta

NIM : 119140114

Topik Kerja Praktek : Pembuatan Backend Website NaZMa Office

Telah melaksanakan Kerja Praktek dengan daftar kehadiran sebagai berikut :

Hari Ke-	Hari / Tanggal	Kegiatan	Hari Ke-	Hari / Tanggal	Kegiatan
1	18 Juli 2022	Mengumpulkan User Story	15	5 Agustus 2022	Membuat API gallery-update
2	19 Juli 2022	Mengumpulkan User Story	16	8 Agustus 2022	Menguji API gallery-update
3	20 Juli 2022	Membuat daftar kebutuhan backend	17	9 Agustus 2022	Membuat API gallery-delete
4	21 Juli 2022	Membuat daftar kebutuhan backend	18	9 Agustus 2022	Menguji API gallery-delete
5	22 Juli 2022	Memastikan daftar kebutuhan backend dengan client	19	10 Agustus 2022 17 Agustus 2022	Membuat API event-create
6	25 Juli 2022	Membuat API blog-create	20	12 Agustus 2022	Menguji API event-create
7	26 Juli 2022	Menguji API blog-create	21	15 Agustus 2022	Membuat API event-update
8	27 Juli 2022	Membuat API blog-update	22	16 Agustus 2022	Menguji API event-update
9	28 Juli 2022	Menguji API blog-update	23	17 Agustus 2022	Membuat API event-delete
10	29 Juli 2022	Membuat API blog-delete	24	18 Agustus 2022	Menguji API event-delete
11	1 Agustus 2022	Menguji API blog-delete	25	19 Agustus 2022	Menguji API event keseluruhan
12	2 Agustus 2022	Menguji API blog keseluruhan	26	22 Agustus 2022	Membuat session login
13	3 Agustus 2022	Membuat API gallery-create	27	23 Agustus 2022	Membuat session login
14	4 Agustus 2022	Menguji API gallery-create	28	24 Agustus 2022	Membuat session signout

Hari Ke-	Hari / Tanggal	Kegiatan	Hari Ke-	Hari / Tanggal	Kegiatan
29	25 Agustus 2022	Membuat API edit testimoni	35	2 September 2022	Membuat API delete crew
30	26 Agustus 2022	Menguji API edit testimoni	36	5 September 2022	Menguji API delete crew
31	29 Agustus 2022	Membuat API edit ongoing project	37	6 September 2022	Menguji backend keseluruhan
32	30 Agustus 2022	Menguji API ongoing project	38	7 September 2022	Menghubungkan backend dengan frontend
33	31 Agustus 2022	Membuat API add crew	39	8 September 2022	Memperbaiki bug
34	1 September 2022	Menguji API add crew	40	9 September 2022	Finalisasi

Daerah Istimewa Yogyakarta, 10 Oktober 2022

Mengetahui,
Direktur Operasional



NaZMa
Nabila Zafira Mahalia. CV

Wiji Nurastuti, SE, MT, AWP

Lampiran C. Dokumen Teknik

SURAT PERNYATAAN KERAHASIAAN

Yang Bertanda tangan dibawah ini,

Nama : Wiji Nurastuti, SE, MT

Jabatan di Instansi : Direktur Operasional

Nama Perusahaan : CV. NABILA ZAFIRA MAHALIA

Dengan ini menyatakan semua informasi yang digunakan mahasiswa Kerja Praktik di CV. Nabila Zafira Mahalia (NaZMa Office) bersifat RAHASIA. Informasi yang dimaksud adalah:

1. Informasi bersifat confidential mencakup dokumen perencanaan, hak akses, dokumentasi dan konfigurasi.
2. Informasi mengenai survey dan Software Requirement Specification (SRS)

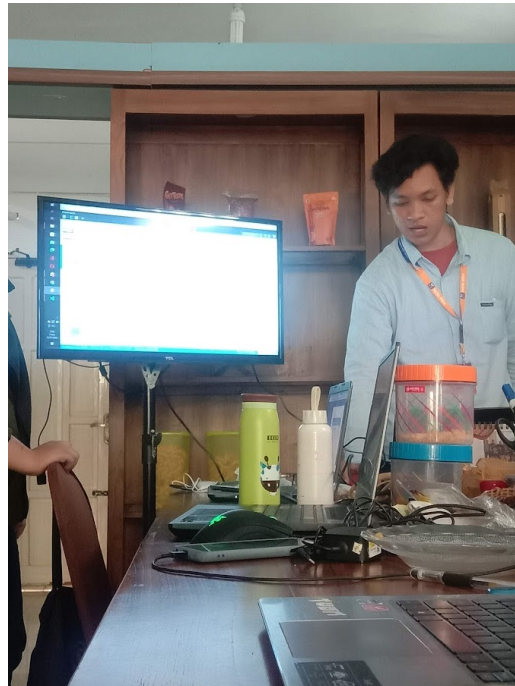
Daerah Istimewa Yogyakarta, 10 Oktober 2022



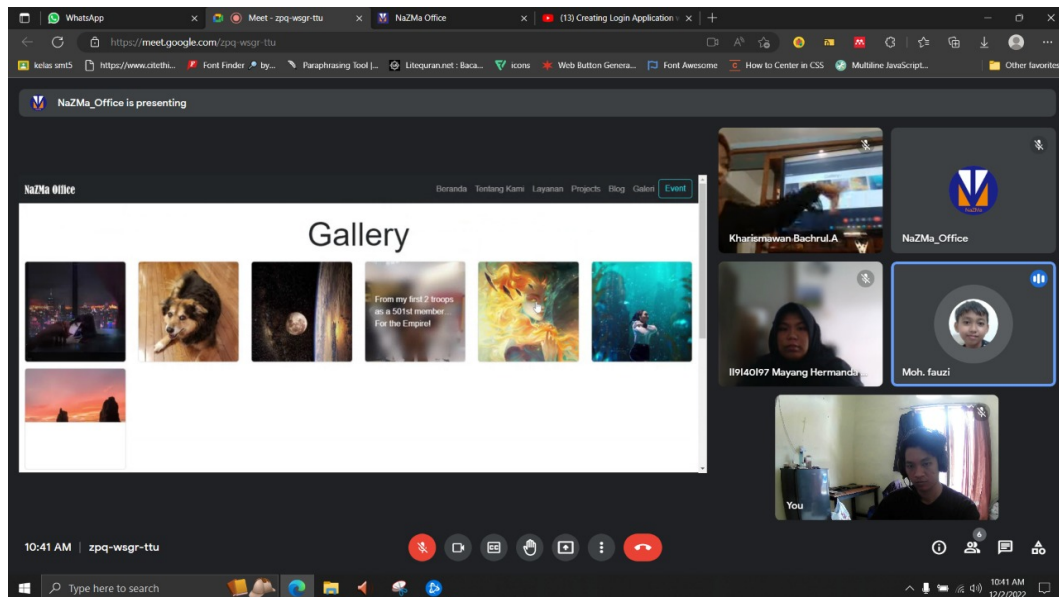
NaZMa
Nabila Zafira Mahalia. CV

Wiji Nurastuti, SE, MT

Lampiran D. Dokumentasi Kegiatan



Gambar D.1 Presentasi Tampilan Website



Gambar D.2 Presentasi Fitur Backend Website

Lampiran E. Penilaian Kerja Praktik

Nama : Geizka Rozilia Ruicosta

NIM : 119140114

Tempat Kerja Praktik : CV. Nabila Zafira Mahalia (NaZMa Office)

Periode : 18 Juli 2022 - 9 September 2022

NO.	ASPEK PENILAIAN	NILAI*
1.	Pemahaman organisasi tempat kerja praktik	87
2.	Kemampuan menerima penugasan dan penyelesaiannya	90
3.	Kemampuan berkomunikasi dan mempresentasikan penyelesaian tugas	92
4.	Kemampuan menulis laporan akademik	88
5.	Kemampuan beradaptasi dengan lingkungan pekerjaan	87
NILAI AKHIR**		88,8

*isikan dengan angka numerik (0-100)

**nilai akhir = total nilai/5

Daerah Istimewa Yogyakarta, 10 Oktober 2022

Pembimbing



Nabila Zafira Mahalia. CV

(Wiji Nurastuti, SE, MT, AWP)

E-mail: itmcnazma@gmail.com