

BAB 3 PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Posisi yang ditempati selama pelaksanaan kerja magang adalah sebagai *Frontend Developer* yang berada di bawah naungan *IT Development*. Dalam pelaksanaannya, kerja magang didampingi oleh mentor yang menjabat sebagai *Chief Technology Officer (CTO)*, yang selalu bersedia meluangkan waktunya untuk memberikan panduan serta penjelasan mengenai alur kerja dan kebutuhan pengguna (*user requirements*) yang berhubungan dengan pengembangan *website*. Proses koordinasi dilakukan dengan seluruh anggota melalui *WhatsApp*, lalu untuk pengelolaan kode dilakukan secara bertahap. Tahap *development* akan melalui tahap *staging* terlebih dahulu sebelum nantinya dilanjutkan ke tahap *production* yang seluruhnya dikelola melalui *GitHub*. Selain itu, rapat rutin divisi juga dilakukan setiap minggu dengan *Chief Technology Officer (CTO)*, *Chief Executive Officer (CEO)* dan *Chief Product Officer (CPO)* untuk menyampaikan perkembangan pembuatan *website E-Invitation*, serta mendiskusikan kendala teknis yang dihadapi selama proses pengembangan proyek berlangsung.

3.2 Tugas yang Dilakukan

Tugas yang didapatkan selama pelaksanaan program kerja magang ini adalah membuat *website E-Invitation* untuk memenuhi kebutuhan *company profile* perusahaan sekaligus *website* pemesanan undangan. *Website* Minyma E-Invitation berisi informasi perusahaan serta jasa pembuatan undangan digital. Tabel 3.1 merupakan daftar aktivitas yang dilakukan tiap minggu selama pelaksanaan magang.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Mempelajari dasar NextJS dan Tailwind CSS, melakukan riset desain landing page, serta mulai mengembangkan halaman Company Profile berdasarkan pembagian tugas tim.
2	Mengembangkan halaman indeks dengan penyesuaian komponen, tata letak, animasi, serta penambahan halaman 404 untuk meningkatkan tampilan dan pengalaman pengguna.
3	Mengoptimalkan halaman <i>Template</i> dan <i>Contact</i> melalui penerapan <i>Client-Side Render</i> (CSR), perbaikan fitur <i>Search</i> , penyesuaian komponen dan animasi, serta peningkatan navigasi dan responsivitas tampilan.
4	Melakukan pengembangan dan penyempurnaan fitur <i>Pricing</i> dan <i>Payment</i> , termasuk implementasi <i>form</i> pembayaran interaktif, integrasi <i>card</i> validator, serta navigasi dinamis berdasarkan pilihan pengguna. Selain itu, dilakukan peningkatan animasi, responsivitas tampilan, dan fungsionalitas filter konten untuk memperkuat pengalaman pengguna di halaman utama.
5	Melakukan peningkatan fitur navigasi, penyempurnaan antarmuka pengguna (UI), serta penyesuaian tampilan responsif. Menambahkan fitur filter warna dan memperbaiki sistem <i>filtering</i> . Mengimplementasikan <i>library</i> "Chroma-js" pada halaman <i>TemplateDetails</i> untuk pemilihan dan penyesuaian warna elemen menggunakan <i>query</i> . Mengoptimalkan beberapa komponen agar lebih fungsional dan interaktif.
6	Melakukan berbagai penyesuaian dan optimalisasi tampilan antarmuka untuk meningkatkan pengalaman pengguna di berbagai perangkat, memperbaiki fungsi yang belum berjalan dengan baik, serta menambahkan fitur pendukung dan integrasi seperti sistem pengiriman email otomatis dengan <i>nodemailer</i> untuk mendukung proses bisnis.
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilakukan tiap minggu selama magang (lanjutan)

Minggu ke -	Pekerjaan yang dilakukan
7	Mengembangkan dan menyempurnakan alur sistem pembayaran dengan menambahkan halaman konfirmasi, validasi data, serta pengelolaan keranjang belanja menggunakan Context API sebagai <i>state management</i> global. Selain itu, dilakukan integrasi sistem pengiriman <i>invoice</i> melalui email, mengoptimalkan proses pemrosesan item agar lebih fleksibel dan mendukung pembelian lebih dari satu produk.
8	Mengembangkan fitur riwayat pesanan yang dilengkapi dengan sistem penyimpanan lokal, countdown otomatis untuk batas waktu pembayaran, serta pengelolaan status pesanan. Mengembangkan sistem notifikasi otomatis via API <i>emailService</i> untuk pesanan yang kedaluwarsa. Mengembangkan Halaman detail pesanan dengan opsi cetak dan unduh <i>invoice</i> dalam format PDF.
9	Melakukan penyempurnaan pada sistem pembayaran dengan validasi keranjang, sinkronisasi penghapusan item melalui <i>CartContext</i> , dan penyesuaian beberapa komponen di tampilan status serta tombol di halaman riwayat. Diperbaiki juga tampilan UI seperti navigasi produk, <i>section Featured Template</i> , serta animasi geser pada <i>Pricing</i> versi <i>mobile</i> . Sistem notifikasi email juga dioptimalkan agar tidak terkirim berulang.
10	Melakukan penataan ulang data template ke dalam <i>file</i> template data agar konsisten di seluruh halaman, serta dibuat komponen <i>TemplateCard</i> dan <i>TemplateSection</i> dengan animasi menggunakan <i>framer-motion</i> . Melakukan beberapa penyesuaian UI agar sesuai dengan desain UI/UX.
11	Melakukan redesign halaman <i>Home</i> menyesuaikan desain UI/UX. Mengembangkan beberapa komponen seperti <i>FAQ</i> , <i>Feature</i> , <i>ContactUs</i> , dan <i>CustomerReview</i> lengkap dengan fungsi <i>dropdown</i> interaktif serta animasi menggunakan <i>Framer Motion</i> . Menyesuaikan bagian <i>Pricing</i> dan <i>Template</i> agar lebih konsisten secara visual, termasuk integrasi beberapa <i>font</i> menyesuaikan dengan <i>font</i> yang digunakan UI/UX.
Lanjut pada halaman berikutnya	

Tabel 3.1 Pekerjaan yang dilakukan tiap minggu selama magang (lanjutan)

Minggu ke -	Pekerjaan yang dilakukan
12	Melakukan penyesuaian tampilan <i>mobile</i> pada berbagai komponen seperti <i>Customer Review</i> , <i>Template</i> , dan <i>Pricing</i> , serta pengoptimalan filter kategori dengan mengubahnya menjadi <i>scroll horizontal</i> dan mengatur ulang tata letaknya. Memperbaiki konsistensi <i>layout</i> di halaman lainnya dan mulai mengembangkan fitur E- Invitation dengan sistem <i>routing</i> , animasi, dan navigasi antar template.
13	Membuat halaman Blog dengan menyesuaikan desain UI/UX, termasuk penyesuaian komponen serta implementasi fitur-fitur fungsional.
14	Membuat halaman Blog Detail yang dapat diakses dari halaman blog untuk memuat informasi seputar blog dengan menyesuaikan tampilan dari UI/UX.
15	Membuat halaman Event Detail pada tampilan GuestMode dengan menyesuaikan desain UI/UX, termasuk fitur-fitur fungsional seperti menambahkan tamu atau melakukan filter.
16	Menyambungkan halaman GuestMode dengan Backend dengan saling berkoordinasi dengan divisi backend yang bersangkutan.

3.3 Uraian Pelaksanaan Magang

Pelaksanaan magang dilakukan dengan mengerjakan tugas yang diberikan oleh mentor. Pelaksanaan magang dijalani kurang lebih selama enam bulan, peserta magang diberikan tugas utama untuk mengembangkan website *E-Invitation* yang terbagi menjadi dua bagian, yaitu pembuatan Company Profile yang berfungsi sebagai tampilan utama dari website yang akan menampilkan jasa pembuatan undangan dan informasi dari company dan Internal Dashboard untuk mengelola user dan keperluan ops lainnya, misalnya autentikasi dari website Minyma E-Invitation. Pengembangan website ini dimulai dengan merancang alur kerja atau flow dari sistem secara menyeluruh untuk memahami proses yang akan dijalankan oleh pengguna. Setelah itu, dilakukan identifikasi kebutuhan pengguna (user requirement) sebagai dasar dalam merancang fitur-fitur yang desain antarmukanya

dikembangkan oleh tim UI/UX yang disesuaikan dengan kebutuhan pengguna dan identitas visual perusahaan, sebelum akhirnya masuk ke tahap implementasi teknis.

3.3.1 User Requirements

User requirements merupakan proses analisis yang bertujuan untuk mengidentifikasi kebutuhan pengguna terhadap *website E-Invitation* yang akan dikembangkan. Melalui tahap ini, fitur yang dikembangkan dapat disesuaikan dengan peran dan kebutuhan masing-masing pengguna sehingga meningkatkan efisiensi kerja, produktivitas, serta mengurangi biaya yang tidak diperlukan. Adapun kebutuhan pengguna yang dibedakan berdasarkan *Role* masing-masing yang memiliki akses dan fungsi tersendiri di dalam *website E-Invitation*.

A. Admin

Role Admin ditujukan bagi pihak pengelola sistem yang memiliki kewenangan penuh terhadap pengaturan dan pengawasan seluruh aktivitas dalam platform E-Invitation. Admin berperan penting dalam memastikan sistem berjalan optimal. Adapun kebutuhan untuk peran ini antara lain:

1. Admin dapat melihat dan mengelola data seluruh pengguna.
2. Admin dapat mengelola permintaan pembuatan undangan.
3. Admin dapat mengatur konten dan tampilan pada halaman Company Profile.
4. Admin memiliki akses ke dashboard internal untuk keperluan operasional.
5. Admin dapat mengatur sistem, termasuk autentikasi dan konfigurasi teknis lainnya.

B. User

Role User merupakan peran bagi pengguna umum yang menggunakan platform E-Invitation untuk membuat permintaan pembuatan undangan digital. Akses pada role ini difokuskan pada fitur-fitur yang berkaitan langsung dengan proses pembuatan undangan. Berikut kebutuhan dari role User:

1. User dapat melakukan pendaftaran akun menggunakan email.

2. User dapat masuk ke dalam platform untuk mengakses fitur yang tersedia.
3. User dapat memilih desain undangan dari template yang disediakan.
4. User dapat mengisi informasi detail terkait acara yang ingin dibuat.
5. User dapat menyimpan, mengunduh, dan membagikan undangan yang telah dibuat.

3.3.2 Analisis Awal dan Desain Sistem untuk Website Minyma E-Invitation

Sebelum proses pengembangan *website* Minyma E-Invitation dimulai, dilakukan serangkaian analisis yang bertujuan untuk memahami kebutuhan *website* serta memastikan alur kerja layanan berjalan secara efisien dan sesuai ekspektasi. Analisis ini melibatkan kegiatan diskusi dan rapat bersama seluruh divisi dan mentor untuk mengumpulkan informasi mengenai bagaimana sistem pemesanan undangan digital akan bekerja.

Beberapa hal yang dibahas mencakup *flow payment*, identifikasi *flow admin* dan *user* untuk kebutuhan dari masing-masing peran, serta pembuatan *use case* diagram, flowchart alur *website* dan desain *prototype* sebagai dasar perancangan antarmuka pengguna. Analisis ini akan menjadi fondasi penting dalam proses pengembangan *website* dalam memenuhi kebutuhan fungsional, dan memberikan pengalaman pengguna yang optimal. Penjelasan lebih lanjut mengenai hasil analisis dan desain sistem akan dijabarkan pada bagian berikut.

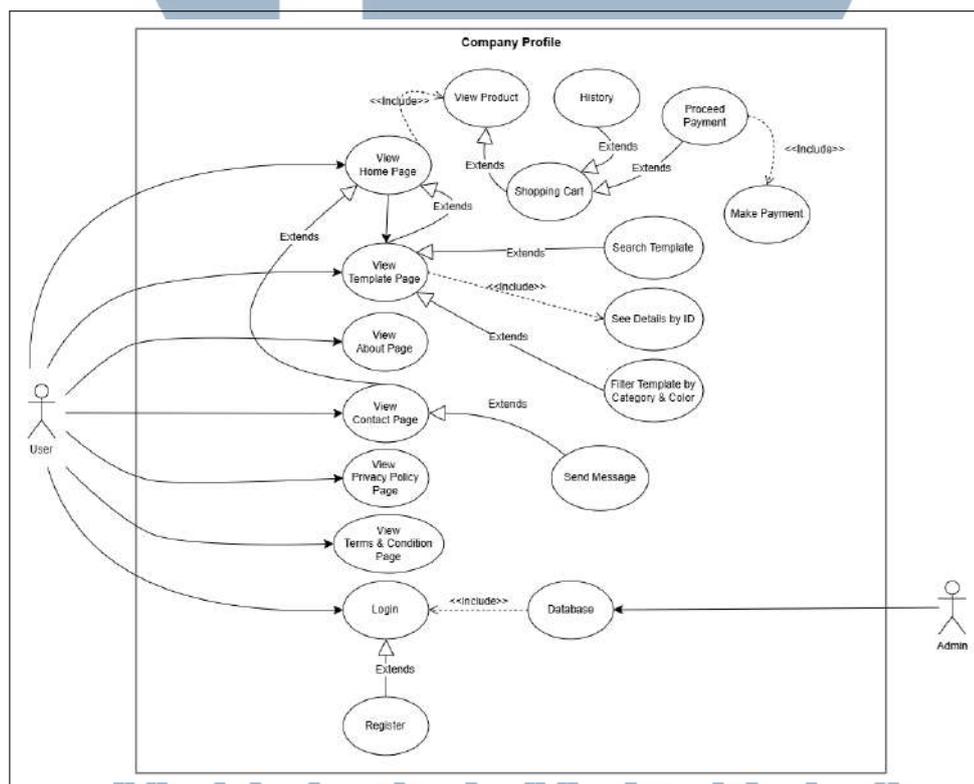
A Use Case Diagram

Use Case Diagram ini dibuat untuk menggambarkan interaksi antara *user* dengan sistem website Minyma E-Invitation. Diagram ini memberikan visualisasi dengan berbagai fitur utama yang dapat diakses oleh pengguna, serta menunjukkan keterkaitan antar fitur dalam sistem.

Gambar 3.1 menunjukkan alur interaksi pengguna dalam mengakses berbagai fitur yang tersedia. Pengguna dapat mengunjungi beberapa halaman seperti Home, Template, About, Contact, Privacy Policy, dan Terms & Condition. Dari halaman *Template*, pengguna dapat menelusuri berbagai desain undangan dengan fitur pencarian, serta filter berdasarkan kategori dan warna. Pengguna juga bisa melihat detail atau isi dari template berdasarkan ID dari masing-masing template.

Selain itu, pengguna dapat mengelola *shopping cart* untuk paket yang dipilih, meninjau riwayat transaksi, hingga melakukan pembayaran melalui proses yang terintegrasi. Fitur pengiriman pesan juga tersedia melalui halaman *Contact*, yang memungkinkan pengguna untuk berkomunikasi langsung dengan pihak Minyma E-Invitation. Proses autentikasi seperti *register* dan *login* juga menjadi bagian dari interaksi awal pengguna ke sistem. Sementara itu, admin terhubung langsung dengan sistem melalui akses ke *database*, yang memungkinkan pengelolaan data secara keseluruhan.

Relasi *include* pada diagram digunakan untuk menunjukkan bahwa suatu proses selalu melibatkan proses lainnya, jika proses utama berjalan maka proses yang terhubung dengan *include* juga akan ikut berjalan. Sementara itu, untuk relasi *extend* pada diagram menggambarkan proses tambahan opsional yang terjadi dalam kondisi tertentu saja, dan tidak selalu berjalan.



Gambar 3.1. Use case diagram website Minyma E-Invitation

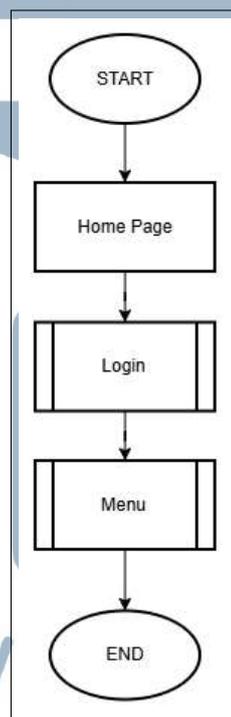
B Flow Website

Flow Website menggambarkan seluruh alur proses yang berlangsung di dalam sistem. Dengan adanya representasi visual ini, hubungan antar komponen

dalam sistem dapat terlihat dengan jelas dan terstruktur. Selain itu, *flow* ini juga membantu tim pengembang dalam memahami alur perancangan serta fitur-fitur yang dibutuhkan pada *website* Minyma E-Invitation. Berikut adalah penjelasan alur interaksi pengguna beserta fitur-fitur yang tersedia, untuk memberikan pemahaman yang lebih jelas mengenai *website* secara keseluruhan.

B.1 Alur Interaksi Pengguna

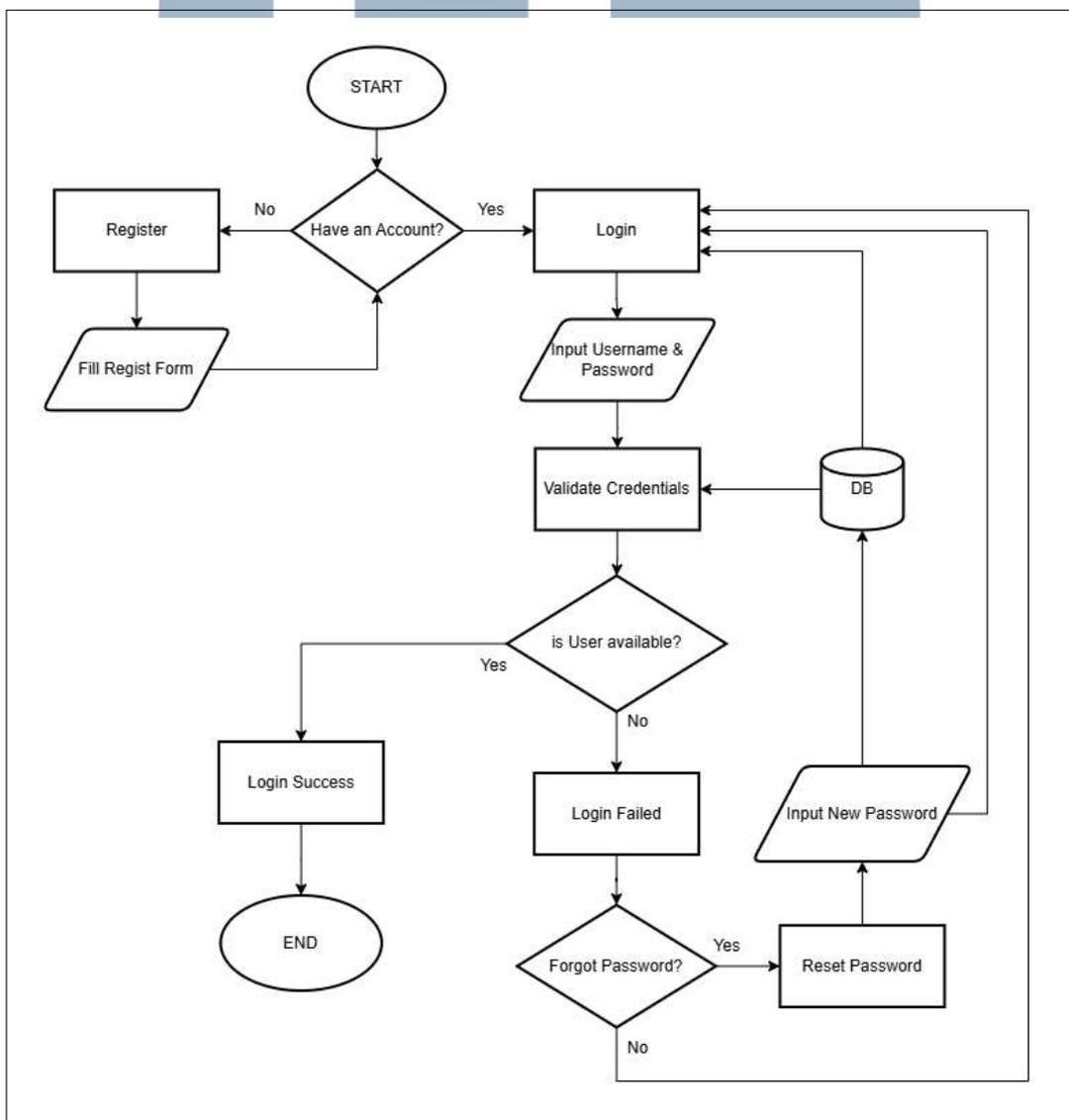
Alur interaksi pengguna menggambarkan tahapan sistematis yang dilalui oleh pengguna saat menggunakan *website*. Proses ini dimulai dari saat pengguna mengakses halaman utama, lalu melanjutkan ke proses *login* untuk bisa mengakses beberapa fitur yang lebih lengkap. Setelah berhasil masuk, pengguna dapat menjelajahi menu yang tersedia, hingga menyelesaikan transaksi sesuai kebutuhan pengguna. Gambar 3.2 adalah ilustrasi dari alur proses yang dilalui pengguna.



Gambar 3.2. Flowchart alur navigasi pengguna

Selanjutnya, penjelasan mengenai proses *login* pengguna, yang menjadi pintu awal sebelum dapat mengakses berbagai fitur *website*. Saat memulai, sistem terlebih dahulu akan menanyakan apakah pengguna sudah memiliki akun. Jika belum, pengguna akan diarahkan ke proses pendaftaran dengan mengisi *form*

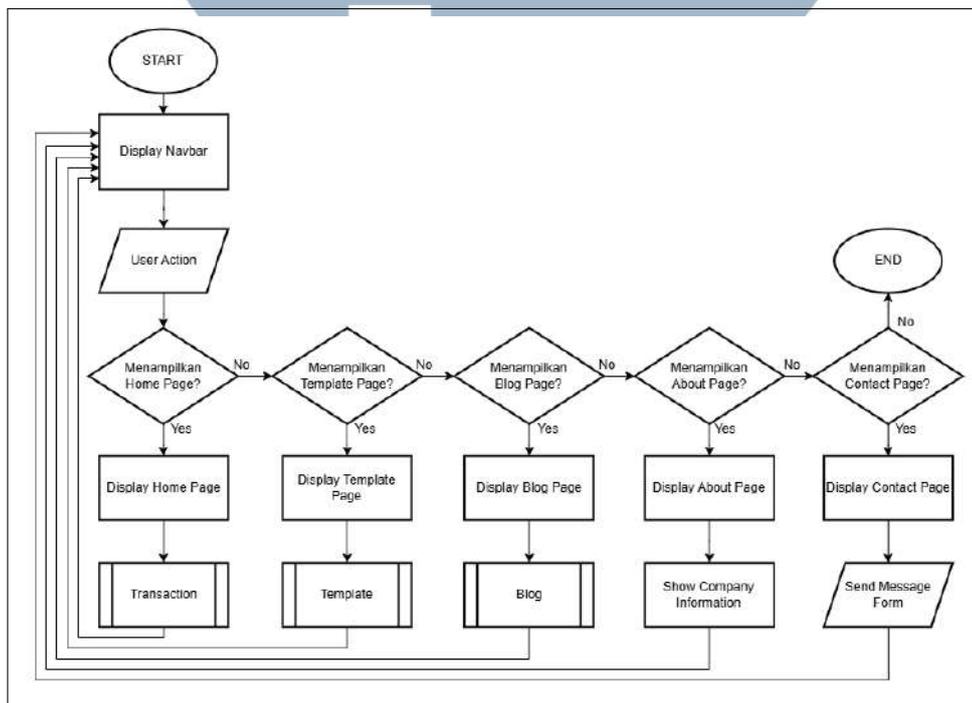
registrasi. Namun, jika pengguna sudah terdaftar, maka bisa dilanjut ke tahap login dengan memasukkan *username* dan *password*. Setelah data dikirim, sistem akan melakukan proses pengecekan apakah informasi pengguna tersedia di dalam *database*. Jika tidak ditemukan, maka akan mendapatkan peringatan *Login Failed*. Pada tahap ini, pengguna diberikan pilihan apakah ingin mereset kata sandi. Jika iya, pengguna akan diarahkan ke proses reset *password* dan diminta untuk membuat kata sandi baru. Sebaliknya, jika informasi pengguna ditemukan dalam *database*, maka pengguna akan berhasil masuk dan diarahkan ke halaman utama *website*. Proses lengkap dari tahapan *login* dapat dilihat pada Gambar 3.3.



Gambar 3.3. Flowchart proses login pengguna

Setelah berhasil login, pengguna akan diarahkan ke halaman utama. Pada tahap ini, pengguna diberikan beberapa pilihan menu seperti *Home*, *Template*, *Blog*, *About*, dan *Contact*. Setiap menu memiliki alur fungsi masing-masing, mulai dari menampilkan daftar template undangan, informasi perusahaan, hingga formulir yang bisa digunakan untuk mengirimkan pesan secara langsung. Untuk menu *Home*, menyediakan akses ke proses transaksi yang terpisah dari menu lainnya untuk menjaga kejelasan dan struktur sistem.

Sementara itu, menu *Template* memiliki alur tersendiri yang menampilkan halaman yang berisi berbagai fitur seperti pencarian, *filter*, dan detail *template*. Sedangkan pada menu *Blog* memiliki alur proses terpisah, yang memungkinkan pengguna menjelajahi kumpulan artikel berdasarkan kategori atau *tag* tertentu dan melihat detail blog secara lebih lengkap. Seluruh proses navigasi ini digambarkan secara rinci dalam Gambar 3.4, yang menunjukkan bagaimana pengguna dapat berinteraksi dengan berbagai fitur *website* berdasarkan menu yang dipilih.

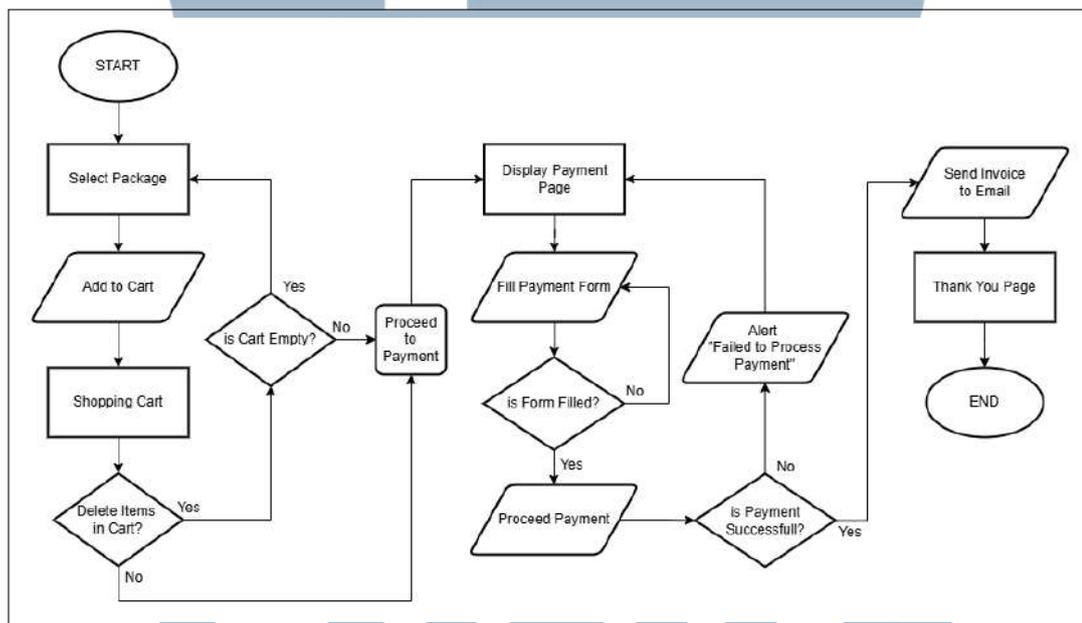


Gambar 3.4. Flowchart proses akses menu

Salah satu proses penting dalam *website* ini adalah transaksi. Proses ini dimulai dari pemilihan paket sesuai keinginan pengguna, menambahkannya ke keranjang. Di keranjang, pengguna memiliki opsi untuk menghapus paket yang sudah dipilih. Jika pengguna menghapus paket, sistem akan mengecek

apakah keranjang kosong. Jika kosong, pengguna diarahkan untuk memilih ulang paket. Jika tidak, atau jika tidak ada penghapusan, proses langsung dilanjutkan ke pembayaran.

Setelah itu, pengguna akan diarahkan ke halaman pembayaran dan diminta mengisi *form* pembayaran. Jika *form* belum lengkap, sistem akan meminta untuk melengkapi. Jika *form* sudah lengkap, pengguna bisa memproses pembayaran. Jika berhasil, invoice akan dikirim ke email pengguna dan halaman "Thank You" akan ditampilkan. Namun, jika gagal, sistem akan menampilkan notifikasi peringatan dan pengguna akan diarahkan kembali ke halaman pembayaran untuk mengulangi proses. Seluruh alur transaksi ini divisualisasikan dalam Gambar 3.5 Flowchart proses transaksi berikut.

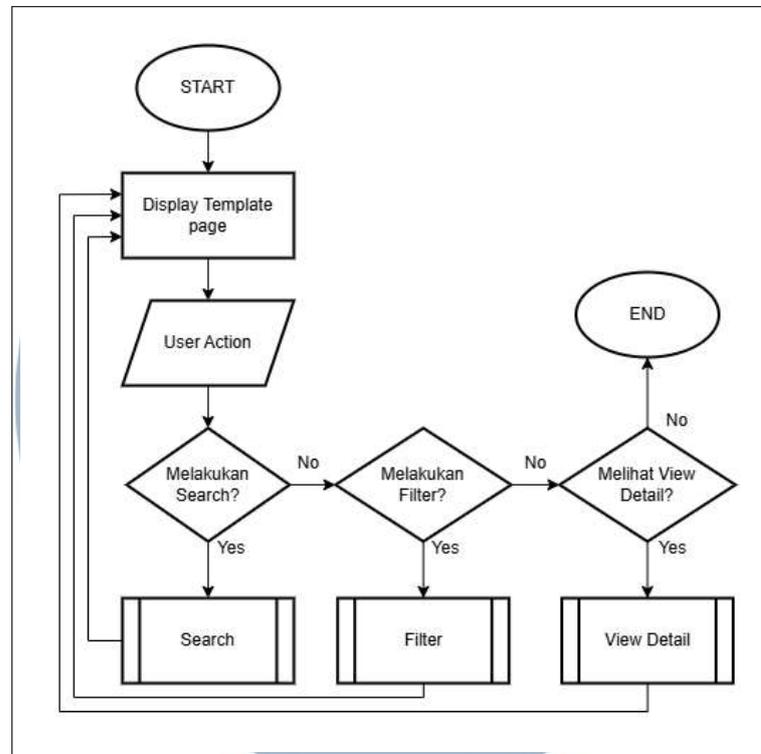


Gambar 3.5. Flowchart proses transaksi

B.2 Alur Sistem pada Halaman *Template*

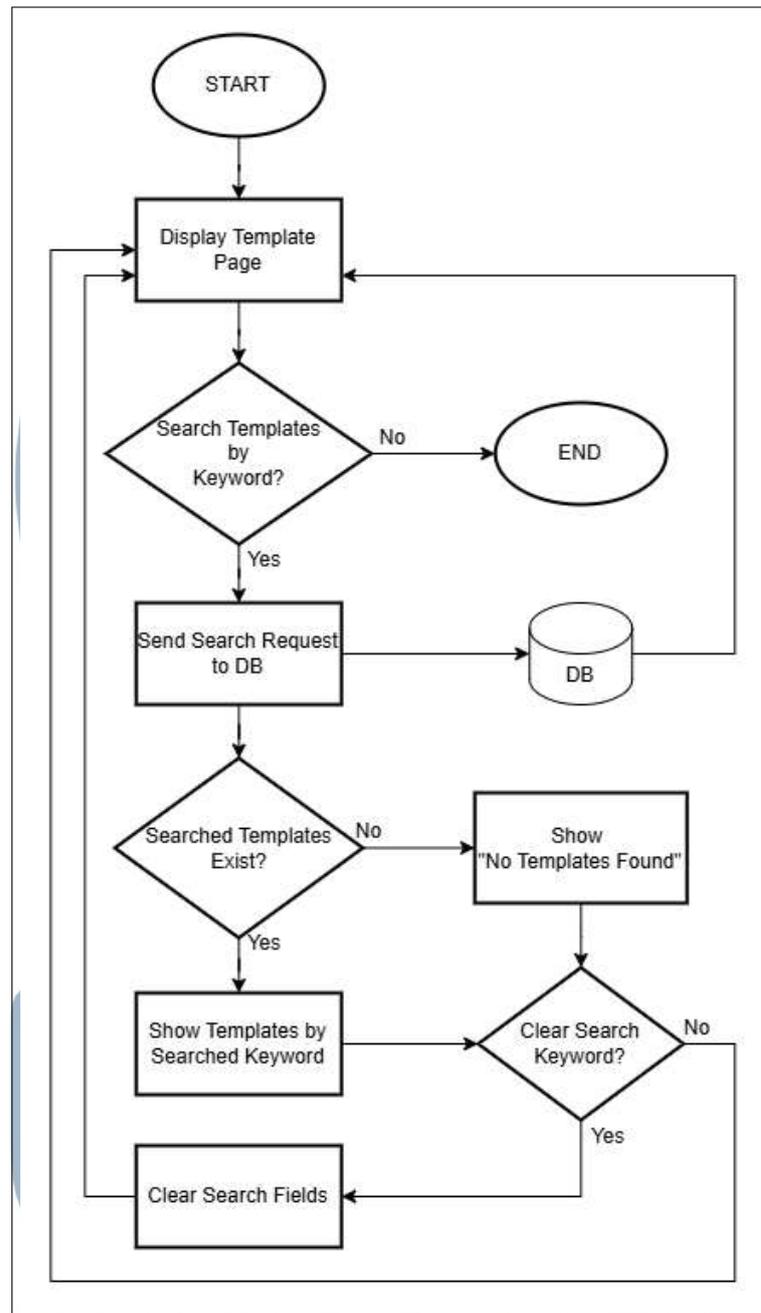
Alur sistem pada halaman *Template* menggambarkan bagaimana sistem merespons interaksi pengguna ketika mengakses fitur yang tersedia. Setelah halaman ditampilkan, sistem akan menunggu aksi dari pengguna. Jika pengguna tidak melakukan apa pun, sistem tetap berada pada halaman tersebut. Namun, jika pengguna melakukan aksi seperti *search template*, *filter template*, atau melihat detail suatu *template*, maka sistem akan menjalankan proses terpisah sesuai aksi yang dilakukan pengguna. Gambar 3.6 menggambarkan alur dari sistem saat

pengguna berinteraksi dengan halaman Template.



Gambar 3.6. Flowchart *User Action* pada halaman template

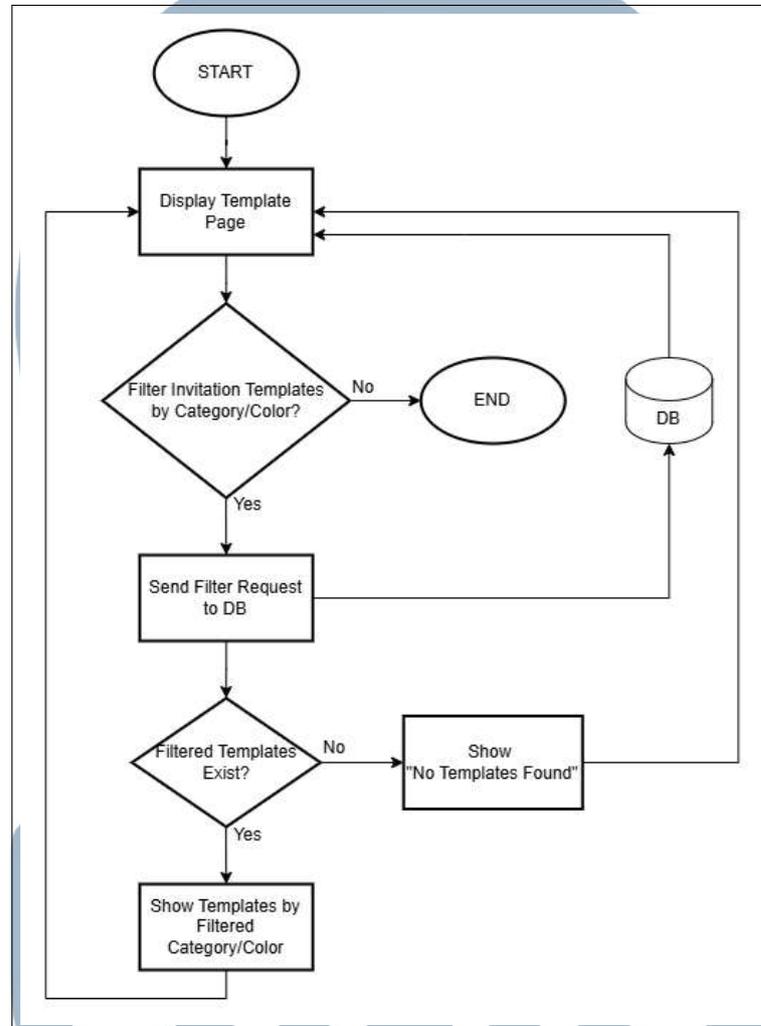
Setelah pengguna berada pada halaman Template, salah satu interaksi yang dapat dilakukan adalah mencari template berdasarkan kata kunci tertentu. Proses ini dimulai ketika pengguna memasukkan kata kunci pada kolom pencarian. Sistem kemudian mengirimkan permintaan ke *database* untuk mencari template yang sesuai dengan kata kunci tersebut. Jika tidak ditemukan hasil yang relevan, maka sistem akan menampilkan pesan “*No Templates Found*”. Namun, jika ditemukan, sistem akan menampilkan daftar template yang sesuai. Pengguna juga dapat menghapus kata kunci untuk kembali ke tampilan awal dan menampilkan semua template. Alur proses pencarian bisa dilihat pada Gambar 3.7 yang menunjukkan bagaimana sistem merespons aksi pencarian dari pengguna.



Gambar 3.7. Flowchart proses search template

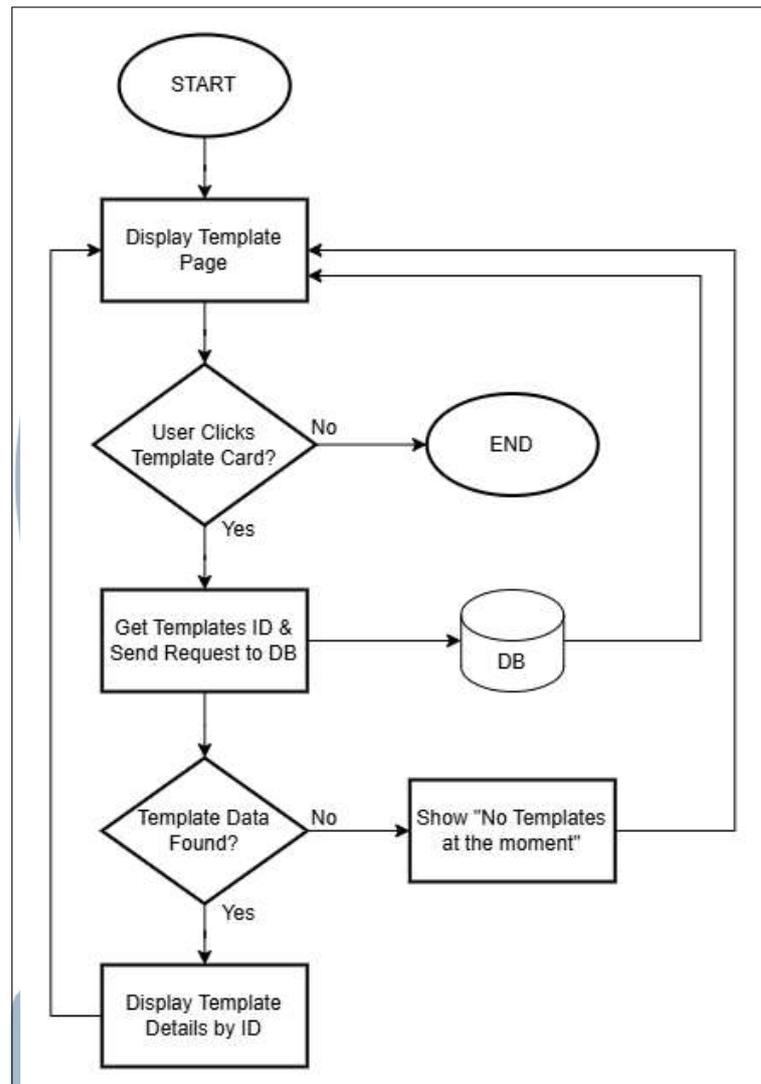
Selain fitur pencarian, pengguna juga dapat memanfaatkan fitur filter untuk membantu menyaring template berdasarkan kategori atau warna. Proses ini diawali dengan menampilkan seluruh template. Jika pengguna memilih filter tertentu, sistem akan mengirimkan permintaan ke *database* untuk menampilkan template yang sesuai dengan filter yang dipilih. Jika tidak ditemukan, sistem akan menampilkan pesan “*No Templates Found*”. Jika ditemukan, maka sistem

akan menampilkan daftar template sesuai dengan filter tersebut. Gambar 3.8 menampilkan alur sistem dalam menangani proses filter template berdasarkan pilihan pengguna.



Gambar 3.8. Flowchart proses filter template

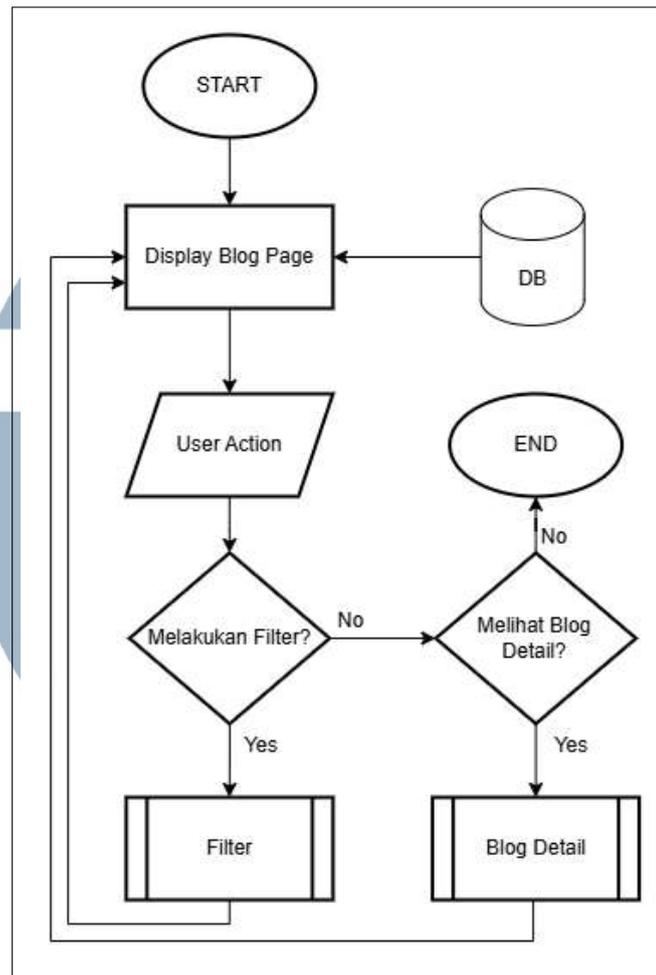
Apabila pengguna ingin melihat desain sebuah template secara detail, pengguna bisa mengklik salah satu kartu template yang tersedia. Kemudian sistem akan mengambil ID dari template tersebut dan mengirimkan permintaan ke *database* untuk mendapatkan informasinya. Setelah data diterima, sistem memverifikasi apakah template yang dimaksud tersedia. Jika template tidak tersedia atau terjadi kesalahan saat pengambilan data, sistem akan menampilkan pesan “*No Templates at the moment*”. Namun, jika data berhasil diperoleh, maka sistem akan menampilkan halaman detail template beserta desainnya. Proses sistem dalam menangani permintaan untuk melihat detail template bisa dilihat pada Gambar 3.9.



Gambar 3.9. Flowchart proses melihat detail template

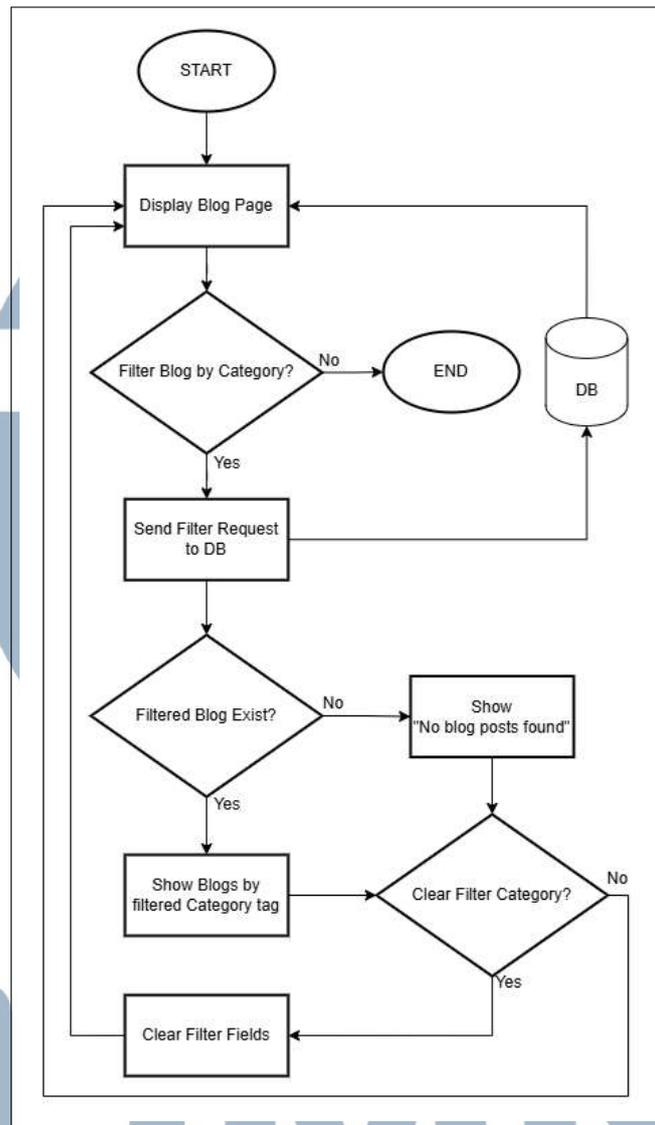
B.3 Alur Sistem pada Halaman Blog

Alur sistem pada halaman Blog menggambarkan bagaimana sistem merespons interaksi *user* ketika mengakses halaman ini. Pada halaman ini, *user* dapat melihat daftar artikel yang berkaitan dengan undangan digital. Selain itu, pengguna juga dapat memanfaatkan fitur *filter* untuk menyaring artikel berdasarkan kategori *tag*. *User* juga bisa melihat detail dari setiap artikel yang ada. Setiap aksi yang dilakukan *user* akan diarahkan ke proses yang berbeda sesuai dengan aksi yang dilakukan. Gambar 3.10 memperlihatkan alur dari sistem saat *user* berinteraksi pada halaman Blog.



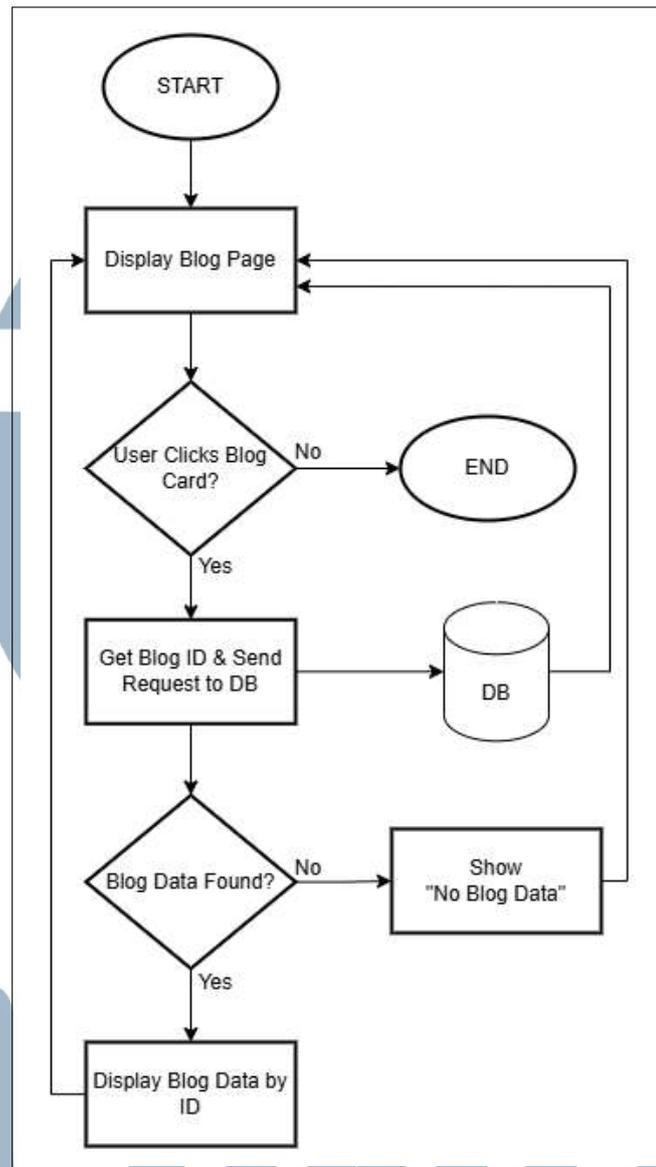
Gambar 3.10. Flowchart *User Action* pada halaman Blog

Setelah *user* berada pada halaman Blog, salah satu aksi yang bisa dilakukan *user* adalah melakukan *filter* berdasarkan kategori *tag* yang tersedia. Proses *filter* dimulai dengan menampilkan seluruh artikel. Jika pengguna memilih *tag* tertentu, sistem akan mengirimkan permintaan ke *database* untuk menampilkan artikel yang sesuai berdasarkan *tag* yang dipilih. Jika setelah melakukan *filter* tidak ada hasil, maka sistem akan menampilkan "No blog posts found". Jika ditemukan, maka sistem akan menampilkan daftar artikel berdasarkan kategori *tag*. Setelah itu, *user* juga bisa menghapus semua *tag* untuk menampilkan semua artikel seperti tampilan awal. Gambar 3.11 menampilkan alur sistem dalam menangani proses *filter* artikel.



Gambar 3.11. Flowchart proses filter blog

Apabila *user* ingin melihat isi dari sebuah artikel secara detail, *user* bisa mengklik salah satu *card* artikel yang tersedia. Setelah itu, sistem akan mengambil ID dari artikel tersebut dan mengirimkan permintaan ke *database* untuk menampilkan data yang bersangkutan. Jika data tidak ditemukan, sistem akan menampilkan "No Blog Data". Namun, jika data ditemukan, sistem akan menampilkan halaman detail dari artikel beserta informasi dari artikel tersebut berdasarkan ID artikel. Proses sistem dalam menangani permintaan untuk melihat detail artikel bisa dilihat pada Gambar 3.12.

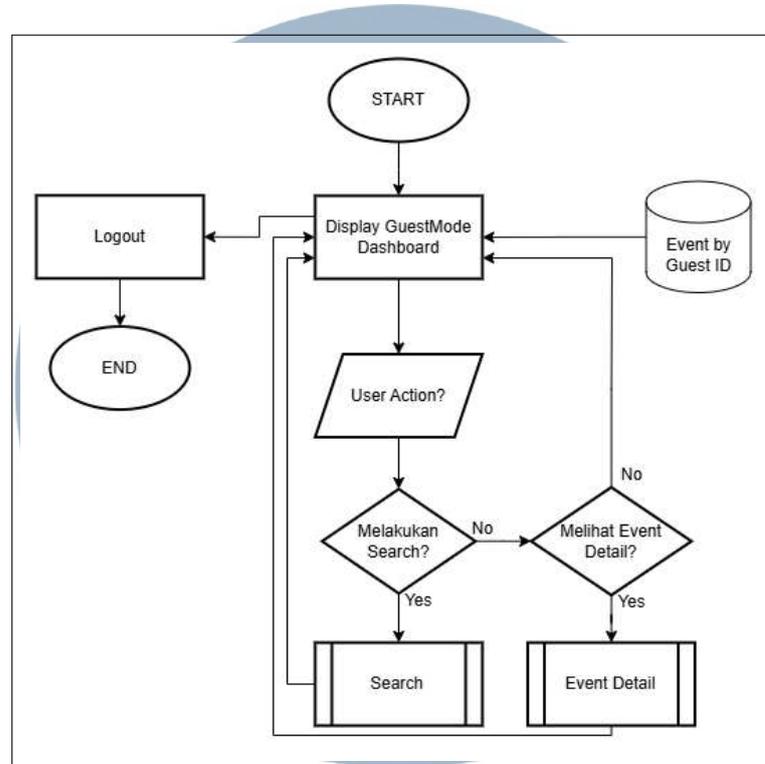


Gambar 3.12. Flowchart proses blog detail

B.4 Alur Sistem pada Halaman *GuestMode*

Alur sistem pada halaman *GuestMode* menggambarkan bagaimana sistem merespons setiap interaksi pengguna saat mengakses halaman ini. Pada halaman ini, sistem akan menampilkan daftar *event* yang sedang diselenggarakan oleh *user*, yang ditampilkan berdasarkan ID *user* yang sedang *login*. Setelah itu, *user* bisa melakukan pencarian *event* berdasarkan nama atau memilih untuk melihat detail dari *event* tersebut. Setiap aksi tersebut akan diarahkan ke alur proses yang sesuai. Jika *user* sudah tidak melakukan aksi lain, *user* bisa *logout* untuk mengakhiri

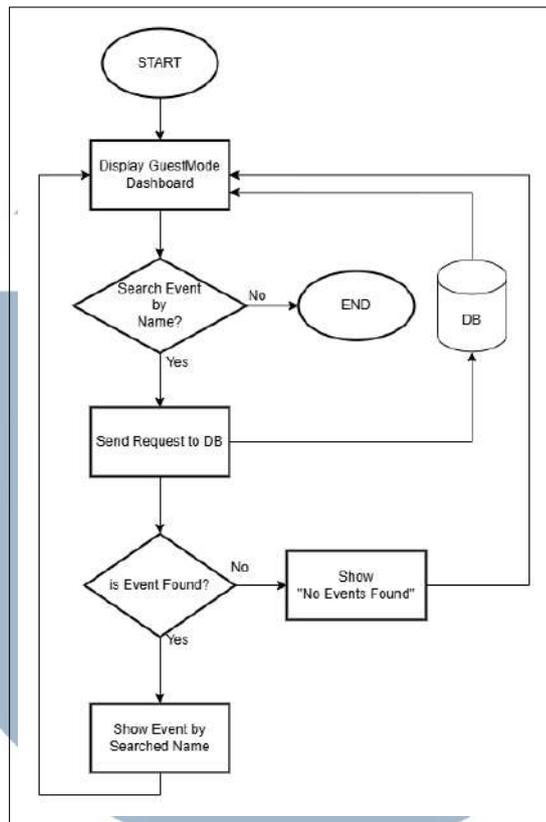
sesi. Gambar 3.13 menampilkan alur sistem pengguna saat berinteraksi di halaman *GuestMode*.



Gambar 3.13. Flowchart *User Action* pada halaman *GuestMode*

Setelah *user* berada di halaman *GuestMode*, *user* bisa melakukan pencarian *event* berdasarkan nama *event*. Proses pencarian dilakukan ketika *user* memasukkan nama *event* pada kolom pencarian. Kemudian sistem akan mengirim permintaan ke *database* untuk menampilkan *event* sesuai pencarian. Jika tidak ada hasil yang ditemukan, sistem akan menampilkan "No Events Found". Namun, jika ditemukan, sistem akan menampilkan daftar *event* berdasarkan hasil pencarian. Alur proses pencarian bisa dilihat pada Gambar 3.14 yang memperlihatkan bagaimana sistem merespons aksi pencarian *user*.

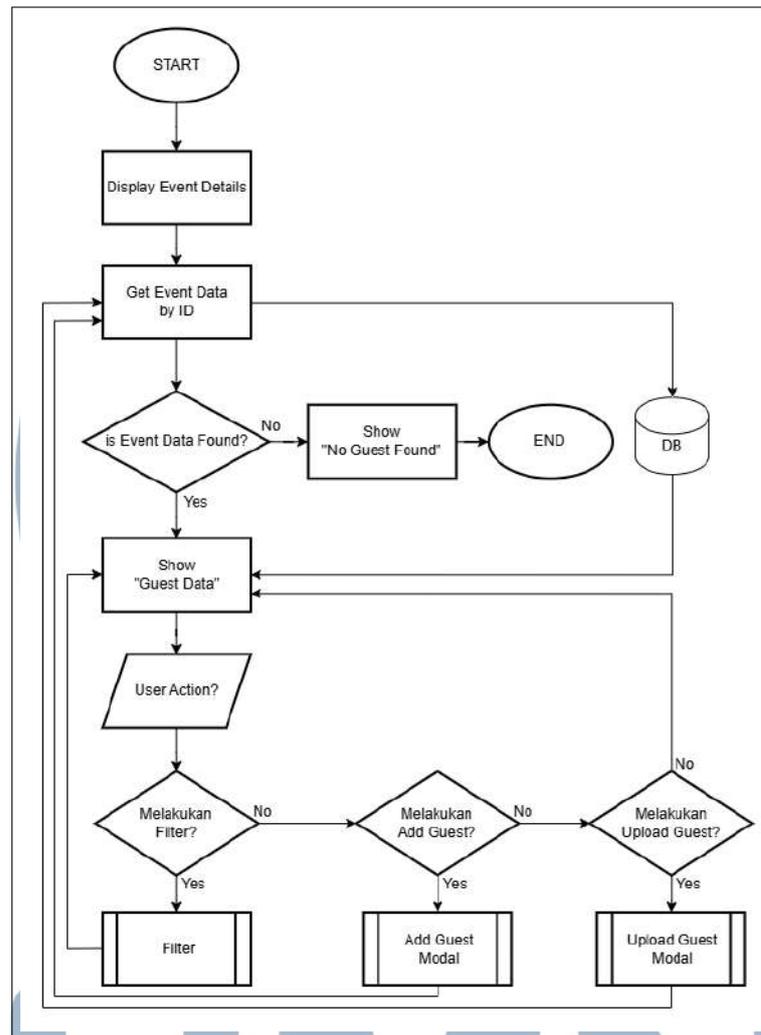
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.14. Flowchart proses search event

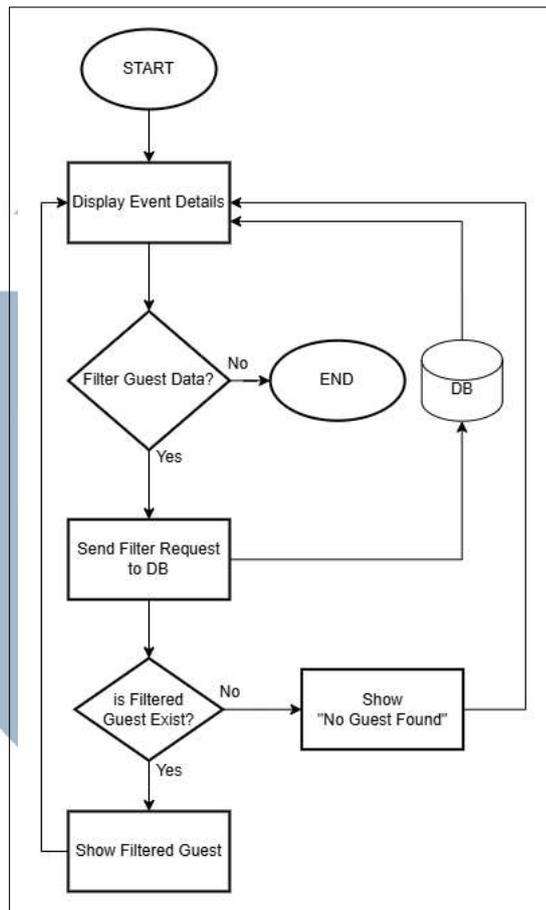
Selain fitur pencarian, *user* juga dapat melihat detail dari setiap *event*. Proses ini dimulai dengan menampilkan halaman dari *event details*. Setelah halaman ditampilkan, sistem akan mengambil data *event* dari *database* berdasarkan ID *event* yang dipilih. Jika data *event* tidak ditemukan, sistem akan menampilkan "No Guest Found". Namun, jika data ditemukan, sistem akan menampilkan daftar *Guest* dari *event* tersebut. Selanjutnya, *user* bisa melakukan beberapa aksi, seperti melakukan filter daftar tamu, menambahkan tamu secara manual, atau mengunggah data tamu dalam bentuk *file*. Setiap aksi akan diarahkan ke dalam proses yang berbeda sesuai dengan aksi yang dilakukan. Gambar 3.15 menggambarkan bagaimana sistem merespons dan menampilkan detail dari suatu *event*.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



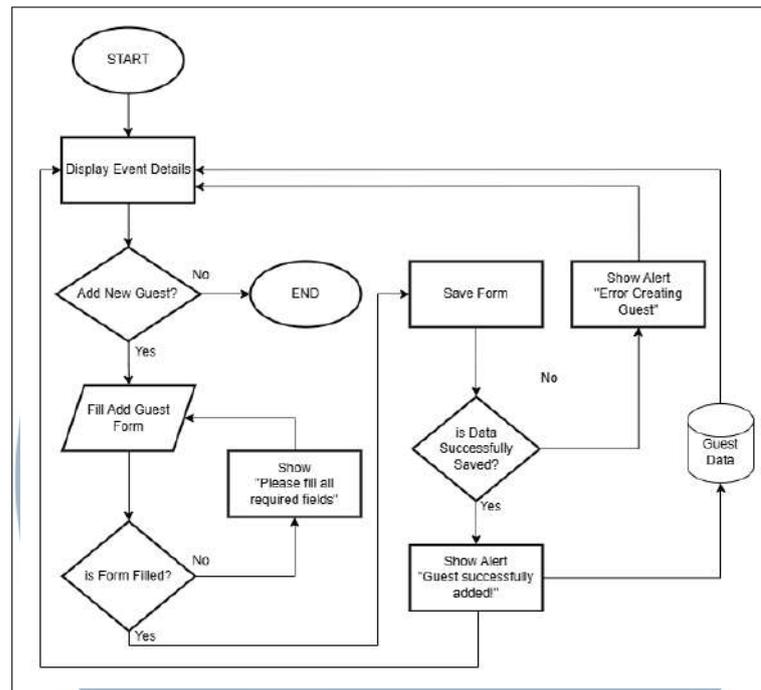
Gambar 3.15. Flowchart proses event detail

Setelah *user* berada di halaman *Event Details*, *user* bisa memfilter daftar tamu. Proses *filter* dilakukan ketika *user* memasukkan kriteria pencarian pada kolom yang tersedia. Setelah itu, sistem akan mengirim permintaan ke *database* untuk menampilkan nama tamu sesuai kriteria pencarian. Jika data tamu tidak ditemukan, sistem akan menampilkan "No Guest Found". Namun, jika data tamu ditemukan, sistem akan menampilkan daftar tamu sesuai dengan kriteria pencarian tersebut. Proses respons sistem terhadap aksi *filter* daftar tamu bisa dilihat pada Gambar 3.16



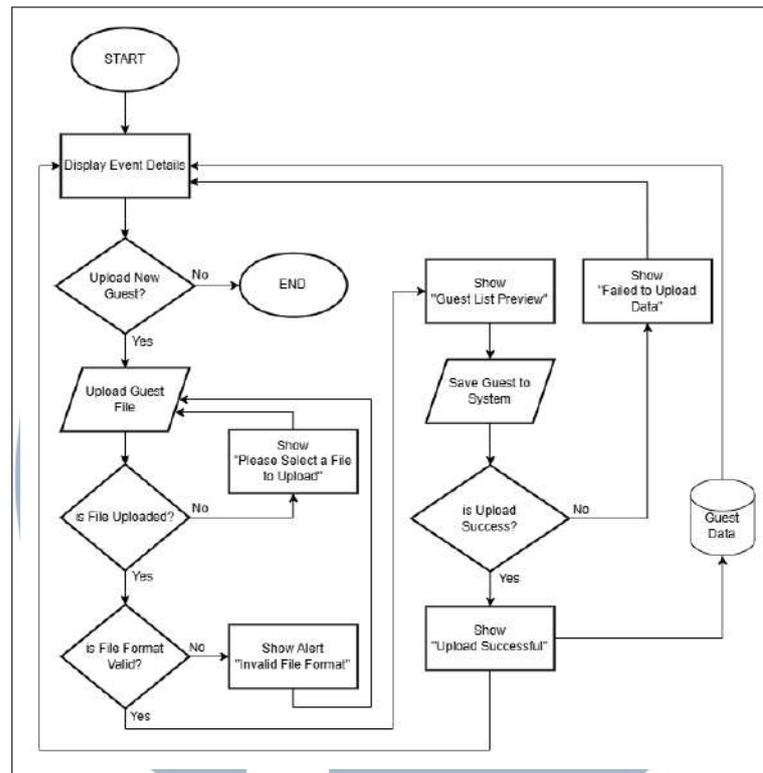
Gambar 3.16. Flowchart proses filter guest

Selain fitur *filter*, *user* juga dapat menambahkan tamu secara manual dengan fitur "Add Guest". Proses ini dimulai ketika *user* ingin menambahkan tamu baru ke dalam daftar tamu pada *event*. Setelah itu, *user* akan diminta untuk mengisi *form* tambah tamu. Jika *form* belum lengkap, maka sistem akan menampilkan "Please fill all required fields" dan menampilkan kembali tampilan *form*. Jika *form* sudah lengkap, *user* bisa menyimpan *form* data tamu tersebut. Jika data gagal tersimpan, sistem akan menampilkan "Error Creating Guest". Namun, jika data berhasil disimpan, sistem akan mengirimkan informasi pada *form* ke *database* untuk disimpan lalu menampilkan *alert* "Guest Successfully added!". Setelah itu, tampilan akan kembali menampilkan halaman detail *event* dengan data tamu yang sudah diperbarui. Gambar 3.17 menampilkan bagaimana proses dari sistem tambah tamu.



Gambar 3.17. Flowchart proses add guest

Apabila *user* ingin menambahkan tamu dalam jumlah besar tanpa harus memasukkannya satu per satu secara manual, *user* bisa menggunakan fitur "Upload Guest". Proses ini dimulai ketika *user* ingin menambahkan tamu baru ke dalam daftar tamu dalam jumlah yang banyak. *User* akan diminta mengunggah *file* dengan jenis "xlsx" atau "csv" ke kolom yang sudah disediakan. Jika *file* belum terunggah atau *upload file* gagal, sistem akan menampilkan "Please Select a File to Upload". Lalu, jika berhasil, sistem akan melanjutkan ke pengecekan format *file*. Jika format tidak sesuai, sistem akan menampilkan alert "Invalid File Format". Namun, jika format *file* sesuai, sistem akan memunculkan "Guest List Preview" terlebih dahulu sebelum menyimpan data. Jika data yang ditampilkan pada *preview* sudah sesuai, *user* bisa menyimpan data. Jika data gagal tersimpan, sistem akan menampilkan "Failed to Upload Data" dan akan langsung mengembalikan *user* ke tampilan detail *event*. Namun, jika data berhasil tersimpan, informasi data akan dikirimkan ke *database* dan sistem akan menampilkan "Upload Successful". Gambar 3.18 menggambarkan alur proses dari fitur *upload guest*.



Gambar 3.18. Flowchart proses upload guest

C List Fitur

Ada beberapa masukan yang diberikan oleh mentor selama pengembangan website Minyma E-Invitation ini. Berikut adalah *List* fitur yang dikembangkan secara keseluruhan.

Tabel 3.2. List fitur dari *website* Minyma E-Invitation

Nomor -	List Fitur
1	Terdapat 2 akses berbeda antara <i>user</i> dan <i>admin</i> .
2	Terdapat tampilan <i>GuestMode</i> yang dapat diakses <i>user</i> untuk mengelola <i>event</i> .
3	<i>Customizable Template</i> sesuai keinginan <i>user</i> .
4	Menampilkan Map lokasi perusahaan.
5	Navbar yang disesuaikan berdasarkan ukuran layar.
Lanjut pada halaman berikutnya	

Tabel 3.2 List Fitur dari Website Minyma E-Invitation (lanjutan)

Nomor -	List Fitur
6	Memberikan <i>loading animation</i> untuk memberikan informasi bahwa konten sedang di <i>load</i> .
7	Memberikan <i>Back to Top Button</i> untuk otomatis kembali ke bagian atas.
8	Memberikan fungsi <i>Add To Cart</i> pada bagian <i>Pricing</i> .
9	Implementasi <i>Shopping Cart</i> untuk menampung Item dari fungsi <i>Add to Cart</i> .
10	Terdapat fitur <i>filter</i> template berdasarkan Kategori dan Warna.
11	<i>Template</i> undangan menyesuaikan berdasarkan ID.
12	Terdapat fitur <i>search</i> untuk pencarian template.
13	Menyesuaikan <i>responsive website</i> pada tampilan desktop dan <i>mobile</i> .
14	Blog menyesuaikan berdasarkan ID
15	Implementasi <i>Nodemailer</i> untuk mengirim <i>invoice</i> ke <i>email</i> .
16	Memberikan <i>pop up</i> ketika <i>invoice</i> berhasil terkirim atau tidak.
17	<i>Downloadable invoice</i> .

D Desain Wireframe

Wireframe dirancang sebagai acuan awal untuk memvisualisasikan struktur dan penempatan elemen pada setiap halaman *website* sebelum masuk ke tahap pengembangan desain *mockup*. Dengan adanya *wireframe*, proses pengembangan dapat berjalan lebih efisien karena sudah memiliki gambaran awal mengenai tata letak dari setiap elemen yang akan ditampilkan. Berikut adalah tampilan *wireframe* untuk setiap halaman yang ada pada *website*.

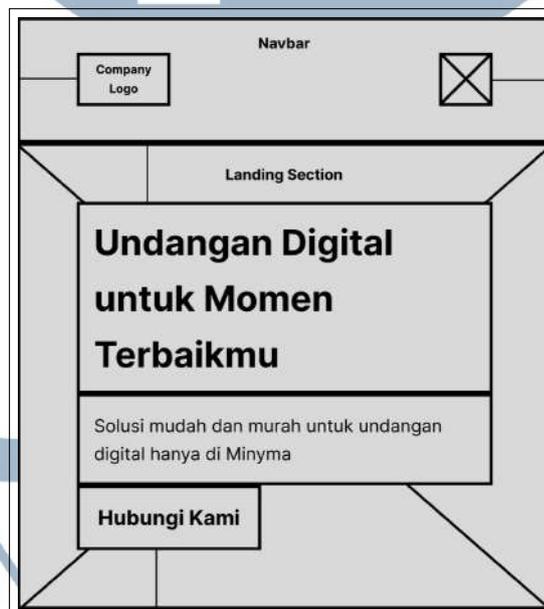
D.1 Halaman Home

Pada halaman *Home*, *wireframe* dirancang menjadi tujuh bagian utama. Dimulai dari *hero section*, sebagai tampilan awal *landing page*. Lalu dilanjutkan dengan *service section* yang menampilkan layanan utama yang ditawarkan. Bagian *customer review* menampilkan ulasan dan testimoni dari pengguna sebelumnya.

Kemudian, pada bagian *template section* menampilkan beberapa contoh template yang tersedia. Bagian berikutnya adalah *pricing section*, yang menyajikan pilihan paket harga. Setelah itu terdapat *FAQ section*, yang menyajikan jawaban atas kumpulan pertanyaan umum. Terakhir, halaman *contact section* yang menampilkan navigasi langsung ke halaman *contact* untuk dapat menghubungi tim pengembang atau mengirim pertanyaan secara langsung.

D.1.1 *Hero Section*

Gambar 3.19 merupakan tampilan *wireframe Hero Section* pada halaman *Home*. Judul ditempatkan di bagian atas yang bertujuan untuk menyampaikan pesan inti secara langsung. Deskripsi singkat diletakkan di bagian bawah judul yang berfungsi sebagai pelengkap untuk menjelaskan keunggulan layanan yang ditawarkan. Lalu pada bagian bawah *Hero Section* terdapat sebuah *button* “Hubungi Kami” yang berfungsi sebagai akses cepat untuk menghubungi pihak pengelola jika membutuhkan layanan lebih lanjut.



Gambar 3.19. Tampilan *wireframe Hero Section* pada halaman *home*

D.1.2 *Service Section*

Gambar 3.20 menampilkan *wireframe Service Section* pada halaman *Home*. Di bagian atas, terdapat judul yang menandakan bahwa area ini menampilkan

layanan utama yang ditawarkan. Di bawahnya, ditampilkan daftar fitur yang ditawarkan beserta fungsi *dropdown* untuk menampilkan informasi lebih detail dari layanan tersebut.

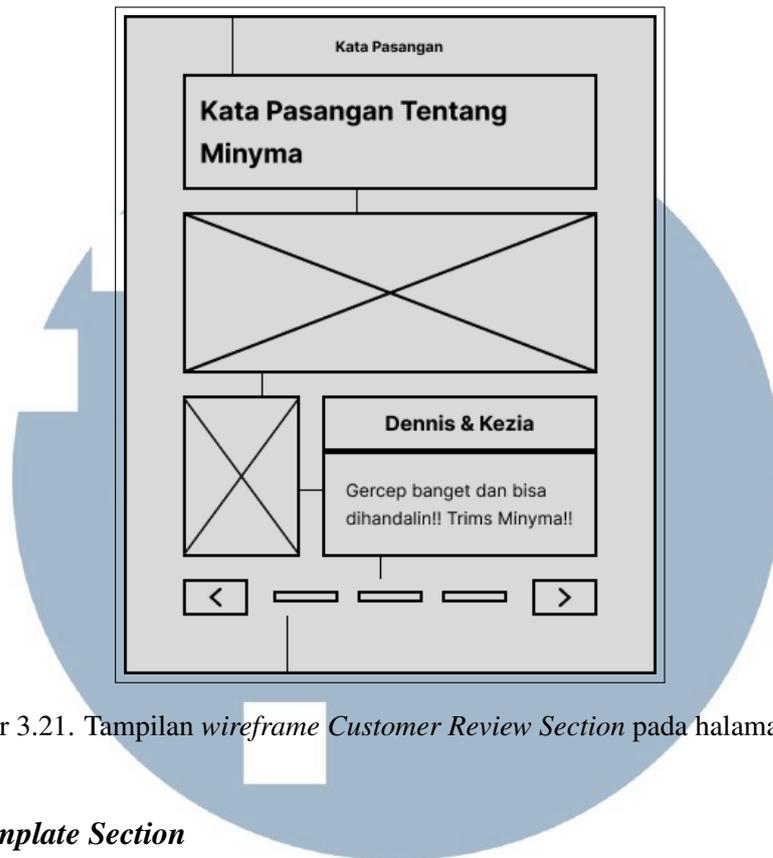


Gambar 3.20. Tampilan *wireframe Service Section* pada halaman *home*

D.1.3 *Customer Review Section*

Gambar 3.21 memperlihatkan *wireframe Customer Review Section* pada halaman *Home*. Di bagian paling atas terdapat teks yang berfungsi sebagai judul *section* ini. Video iklan diletakkan di bawah judul yang berfungsi sebagai media iklan visual untuk memperkenalkan layanan secara singkat dan menarik. Di bawahnya, ditampilkan *cover* dari hasil undangan yang disejajarkan dengan nama dan ulasan dari pelanggan terhadap kualitas layanan. Pada bagian bawah *section* ini terdapat tombol navigasi *next* dan *prev* yang berfungsi untuk melihat ulasan dari pelanggan lainnya.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

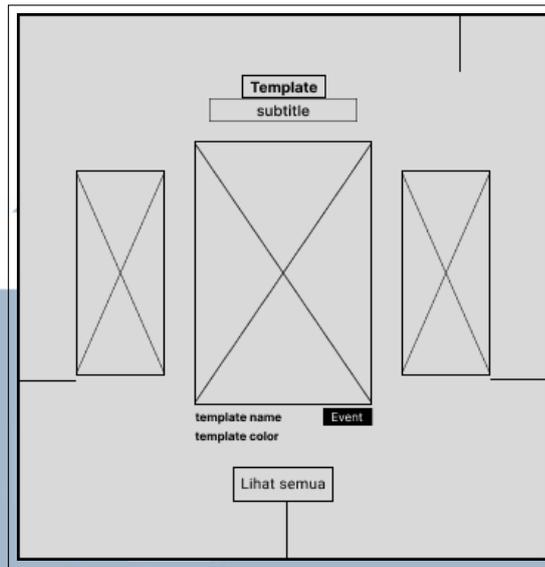


Gambar 3.21. Tampilan *wireframe* *Customer Review Section* pada halaman *home*

D.1.4 *Template Section*

Gambar 3.22 menampilkan *wireframe* *Template Section* pada halaman *Home*. Di bagian atas terdapat judul utama yang menunjukkan bahwa bagian ini menampilkan berbagai pilihan template, diikuti oleh teks subjudul sebagai penjelasan singkat. Di bawahnya diletakkan sebuah *card* yang menampilkan *preview* dari template yang dilengkapi dengan *cover* template, nama, warna, dan jenis acara (*event*). Di bagian bawah, terdapat tombol "Lihat Semua" yang berfungsi untuk menjelajahi seluruh koleksi template yang tersedia.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

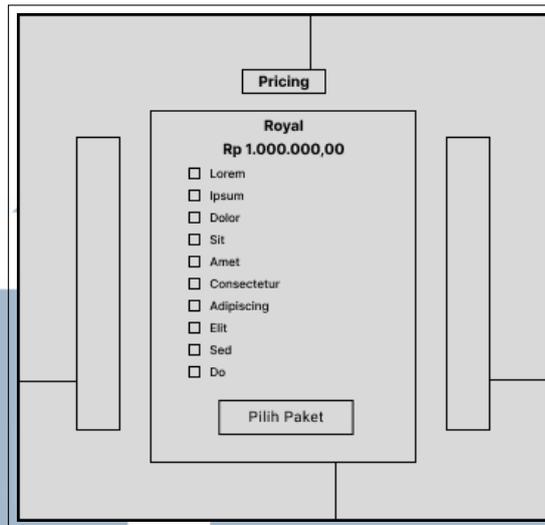


Gambar 3.22. Tampilan *wireframe Template Section* pada halaman *home*

D.1.5 Pricing Section

Gambar 3.23 menampilkan *wireframe Pricing Section* yang menyajikan informasi mengenai paket layanan yang ditawarkan. Di bagian atas terdapat judul “Pricing” sebagai penanda area ini. Di bawahnya ditampilkan sebuah *card* yang berfungsi untuk menampilkan jenis dan harga dari paket. *Card* ini juga menampilkan daftar fitur yang tersedia pada masing-masing paket. Pada sisi kanan dan kiri *card* utama, terdapat sebagian potongan *card* lain yang berfungsi sebagai penanda visual bahwa masih ada pilihan paket lainnya. Pada bagian bawah, terdapat tombol “Pilih Paket” yang berfungsi untuk memasukkan paket yang dipilih ke dalam *cart*.

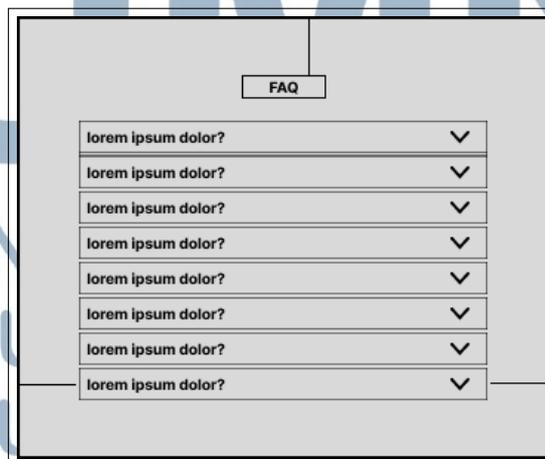
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.23. Tampilan *wireframe Pricing Section* pada halaman *home*

D.1.6 *FAQ Section*

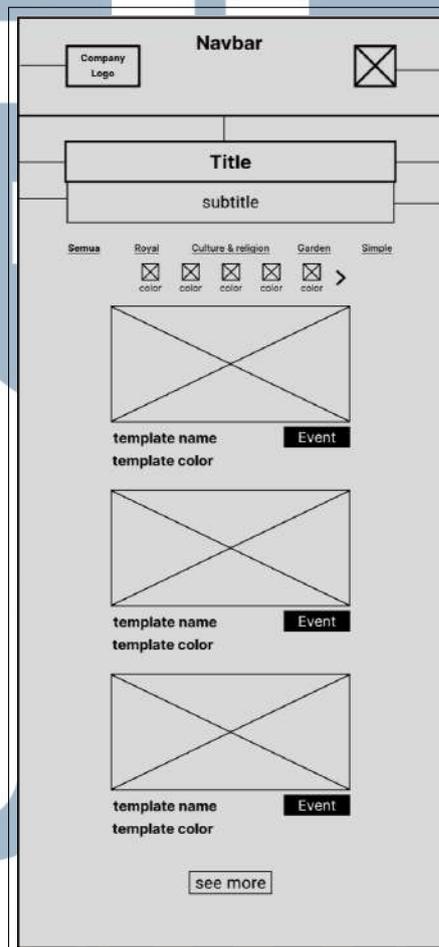
Gambar 3.24 menggambarkan *wireframe FAQ Section* pada halaman *Home*. *Section* ini berfungsi sebagai area bantuan bagi pengguna untuk menemukan jawaban atas pertanyaan umum yang sering diajukan. Bagian ini diawali dengan judul “*FAQ*” sebagai penanda visual yang memudahkan pengguna mengetahui isi dari bagian ini. Di bawahnya, terdapat beberapa baris pertanyaan yang disertai fungsi *dropdown*. Fitur *dropdown* berfungsi untuk menampilkan informasi secara detail terhadap pertanyaan-pertanyaan yang umum ditanyakan.



Gambar 3.24. Tampilan *wireframe FAQ Section* pada halaman *home*

D.2 Halaman *Template*

Pada halaman ini, menampilkan berbagai macam template yang tersedia. Setiap template ditampilkan dengan detail informatif seperti palet warna, kategori, dan elemen visual lainnya yang mendukung tema dari tiap template. Halaman ini juga dirancang untuk memberikan gambaran dari setiap template, sekaligus menyediakan fitur untuk melihat lebih detail dari masing-masing desain. Berikut adalah tampilan *wireframe* dari halaman *Template*.



Gambar 3.25. Tampilan *wireframe* halaman *template*

Gambar 3.25 adalah tampilan *wireframe* halaman *Template*. Pada bagian paling atas, terdapat teks utama yang merupakan judul dari halaman. Tepat di bawahnya, terdapat sebuah teks singkat yang berfungsi sebagai slogan untuk menarik perhatian. Lalu di bawahnya terdapat fitur filter berdasarkan kategori dan warna yang berfungsi untuk menyortir template berdasarkan kategori dan warna

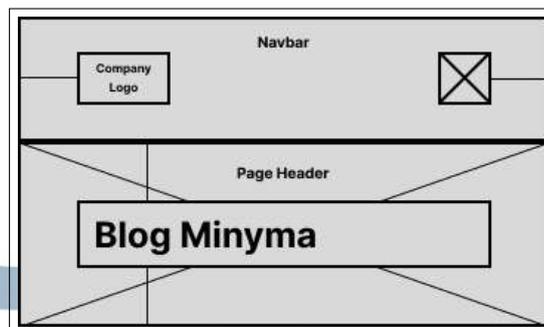
yang diinginkan. Di bawahnya merupakan bagian untuk menampilkan template-template yang ada. Lalu di bawahnya terdapat tombol aksi “*See More*” yang bertujuan untuk menampilkan lebih banyak *template*.

D.3 Halaman Blog

Halaman blog ini menampilkan berbagai macam artikel informasi dan tips yang berkaitan dengan layanan yang ditawarkan. Setiap blog ditampilkan dengan topik yang berbeda, dikelompokkan berdasarkan kategori dan tag tertentu. Halaman ini dirancang untuk memberikan informasi yang relevan, guna membantu pengguna menemukan berbagai panduan yang mendukung kebutuhan masing-masing pengguna.

D.3.1 *Hero Section*

Gambar 3.26 adalah tampilan *wireframe Hero Section* pada halaman blog. Pada bagian ini, terdapat teks yang ditempatkan di tengah sebagai judul, yang bertujuan untuk memberitahu isi dari halaman blog ini. Lalu pada bagian latar belakang, diberikan elemen visual berupa gambar yang membuat tampilan menjadi lebih menarik.



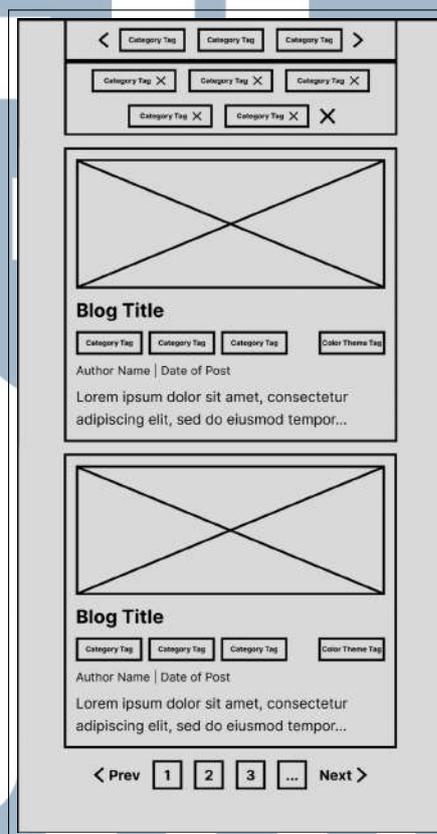
Gambar 3.26. Tampilan *wireframe Hero Section* pada halaman Blog

D.3.2 *Feature Section*

Gambar 3.27 menampilkan *wireframe Feature Section* pada halaman Blog. Di bagian atas terdapat fitur filter kategori tag yang bertujuan untuk menyortir blog berdasarkan kategori tag yang dipilih. Di sisi kiri dan kanan terdapat ikon panah, yang berfungsi sebagai navigasi untuk melihat kategori lainnya. Tepat di bawahnya,

ditampilkan kategori tag yang dipilih sebagai penanda, lalu terdapat ikon 'x' yang berfungsi untuk menghapus pilihan kategori tag.

Pada bagian tengah halaman blog, menampilkan daftar blog lengkap dengan judul, kategori, tema warna, nama penulis, tanggal publikasi, serta deskripsi singkatnya yang mendukung isi dari masing-masing blog. Lalu pada bagian bawah halaman blog, terdapat fitur *pagination* disertai tombol navigasi *next* dan *prev* di sisi kiri dan kanan yang berfungsi untuk menjelajahi blog lainnya di halaman yang berbeda.



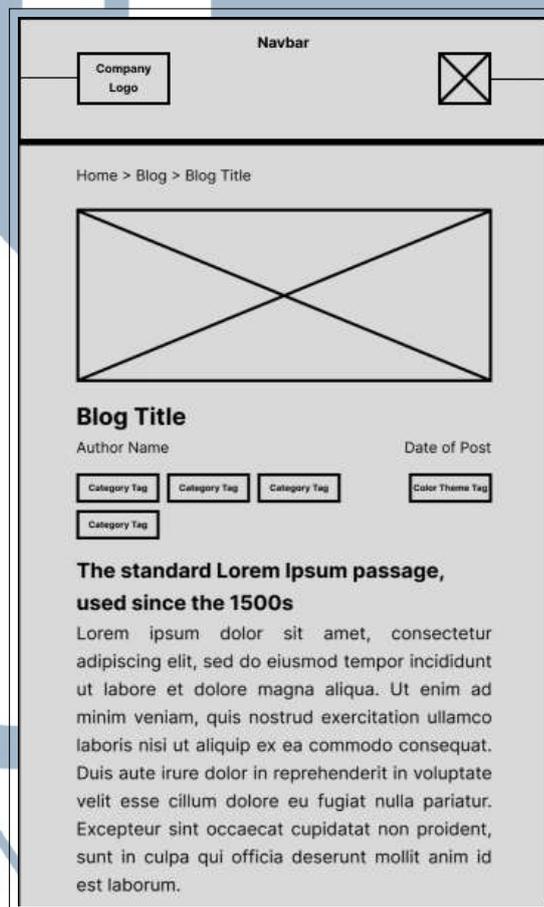
Gambar 3.27. Tampilan *wireframe Feature Section* pada halaman Blog

D.4 Halaman Blog Detail

Pada halaman ini, menampilkan isi lengkap dari blog yang dipilih oleh pengguna secara detail. Setiap blog menampilkan informasinya secara lengkap, seperti judul, kategori, hingga deskripsi dan isi blog. Halaman ini bertujuan untuk memberikan informasi lebih komprehensif dan sesuai dengan kebutuhan pengguna, berdasarkan kategori blog yang dipilih.

D.4.1 Content Section

Gambar 3.28 menunjukkan *wireframe Content Section* pada halaman Blog Detail. Di bagian atas terdapat navigasi yang bertujuan untuk menunjukkan posisi pengguna saat ini dalam *website*, sekaligus berfungsi untuk kembali ke halaman sebelumnya. Tepat di bawahnya, terdapat gambar yang mendukung isi blog. Pada bagian bawah gambar, terdapat informasi seperti judul, penulis, tanggal publikasi, beserta kategori tag yang bertujuan untuk memberikan informasi dari blog ini. Lalu pada bagian bawah, terdapat isi blog berupa informasi yang dapat membantu kebutuhan pengguna.

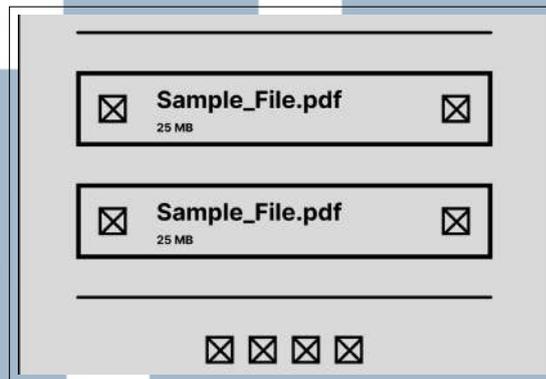


Gambar 3.28. Tampilan *wireframe Content Section* pada halaman Blog Detail

D.4.2 File Attachment Section

Gambar 3.29 memperlihatkan *wireframe File Attachment Section* pada halaman Blog Detail. Di bagian tengah terdapat lampiran file yang berkaitan dengan

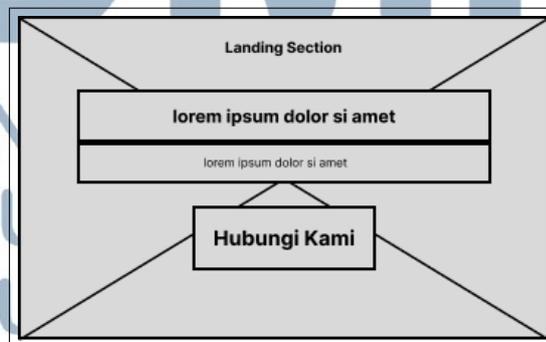
isi blog dan memiliki fungsi untuk diunduh oleh pengguna. Pada bagian atas dan bawah lampiran file ditambahkan elemen garis dekoratif sebagai pemisah visual dari konten lainnya. Di bagian bawah *section* terdapat ikon berbagai platform media sosial yang berfungsi untuk membagikan halaman blog ini.



Gambar 3.29. Tampilan *wireframe File Attachment Section* pada halaman Blog Detail

D.5 Contact Section

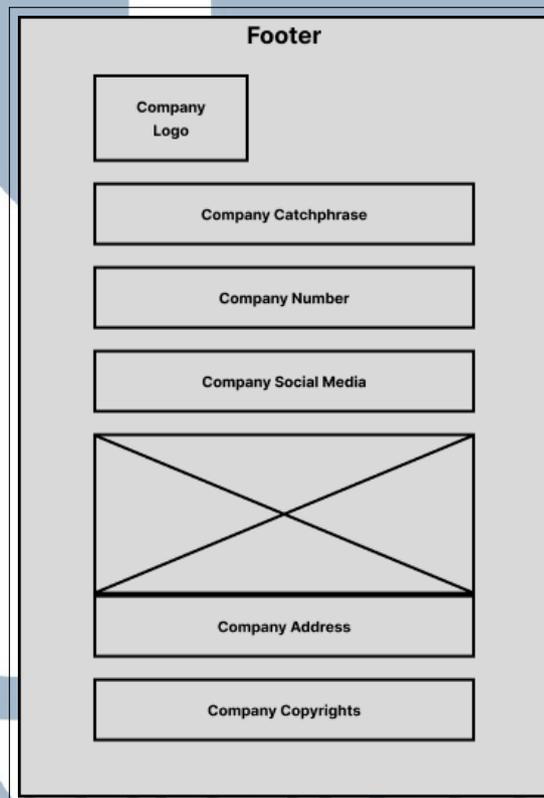
Gambar 3.30 menunjukkan *wireframe Contact Section*. Di bagian paling atas, terdapat teks utama yang berfungsi sebagai pengantar atau kalimat ajakan untuk mendorong interaksi pengguna. Di bawahnya, terdapat deskripsi tambahan yang memberikan penjelasan lebih lanjut untuk memperkuat pesan utama. Latar belakang *section* ini diberi elemen visual berupa gambar untuk menambah kesan estetik dan menarik. Tepat di bawah teks, terdapat tombol aksi “Hubungi Kami” yang bertujuan untuk menarik perhatian pengguna dan mendorong mereka melakukan pemesanan atau menghubungi tim layanan.



Gambar 3.30. Tampilan *wireframe Contact Section*

D.6 Footer Section

Gambar 3.31 merupakan *wireframe Footer Section*. Di bagian atas footer, terdapat logo perusahaan yang berfungsi sebagai identitas visual. Tepat di bawahnya, ditampilkan kalimat singkat yang bertujuan untuk menarik perhatian pengunjung terkait *website* perusahaan. Selanjutnya, terdapat informasi kontak perusahaan serta media sosial yang memudahkan pengunjung untuk menghubungi tim layanan. Di bawahnya, ditampilkan sebuah *maps* sehingga mempermudah pengguna dalam menemukan lokasi kantor. Di bagian paling bawah, terdapat informasi hak cipta sebagai bentuk perlindungan hukum dan penanda tahun publikasi konten.



Gambar 3.31. Tampilan *wireframe Footer Section*

E Desain *Mockup*

Desain *mockup* merupakan pengembangan dari desain *wireframe* yang sudah dibuat sebelumnya. Desain ini dimulai dengan mencari aset yang akan digunakan untuk isi *website*, serta penambahan elemen visual seperti warna, ikon,

dan tipografi. Desain *mockup* ini dibuat dengan menggunakan aplikasi Figma. Tujuan dari pembuatan desain *mockup* ini adalah untuk memberikan gambaran nyata kepada tim pengembang maupun pengguna mengenai bagaimana sistem akan terlihat dan berfungsi, sehingga mempermudah dalam tahap implementasi selanjutnya.

E.1 Halaman *Home*

Halaman *Home* dirancang untuk menampilkan layanan secara menyeluruh kepada pengguna. Di dalamnya terdapat beberapa *section* seperti *hero*, *service*, *customer service*, *template*, *pricing*, dan *FAQ*. Setiap *section* disusun dengan tampilan visual yang mudah dipahami, guna membantu pengguna mengenal fitur-fitur yang ada. Berikut merupakan tampilan halaman *home*.

E.1.1 *Hero Section*

Gambar 3.32 menampilkan tampilan *mockup Hero Section* pada halaman *home*. Tampilan dirancang sama seperti *wireframe*, hanya saja ada penambahan gambar di bagian bawah. Bagian lainnya tetap sama, seperti judul pada bagian atas, lalu deskripsi singkat di bawah judul. Tepat di bawahnya terdapat tombol “Hubungi Kami”.



Gambar 3.32. Tampilan *mockup Hero Section* pada halaman *home*

E.1.2 Service Section

Gambar 3.33 menunjukkan *mockup Service Section* pada halaman *home*. Pada bagian ini, terdapat penyempurnaan dengan penambahan deskripsi singkat di bawah judul, yang bertujuan untuk memperkuat pesan layanan yang ditawarkan.



Gambar 3.33. Tampilan *mockup Service Section* pada halaman *home*

E.1.3 Customer Review Section

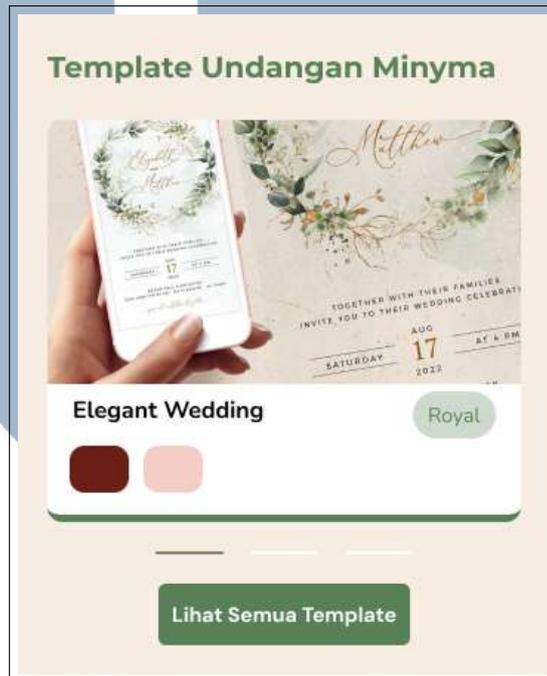
Gambar 3.34 menampilkan *mockup Customer Review Section* pada halaman *home*. Terdapat beberapa penyesuaian, seperti bagian video diganti dengan menampilkan undangan dari *customer*. Lalu, *layout* bagian bawah *section* diubah menjadi hanya menampilkan ulasan beserta nama *customer*.



Gambar 3.34. Tampilan *mockup Customer Review Section* pada halaman *home*

E.1.4 *Template Section*

Gambar 3.35 adalah tampilan *mockup Template Section* pada halaman *home*. Dilakukan beberapa penyesuaian, seperti menghilangkan subjudul yang terletak di bawah judul. Selain itu, bagian tampilan *card* disederhanakan dan ditambahkan indikator yang menunjukkan bahwa bagian ini dapat digeser secara horizontal.



Gambar 3.35. Tampilan *mockup Template Section* pada halaman *home*

E.1.5 *Pricing Section*

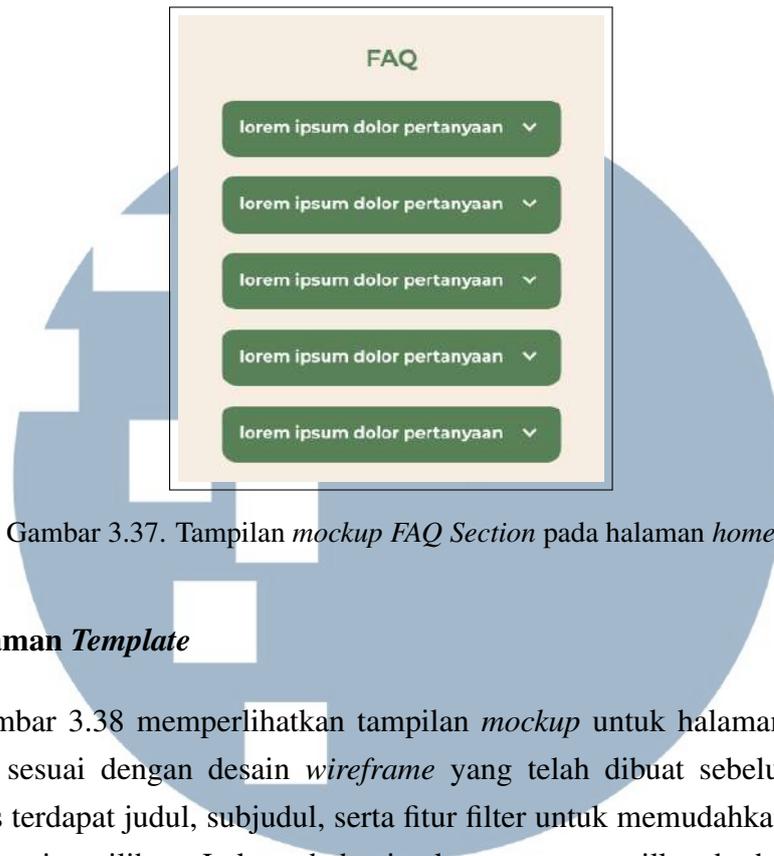
Gambar 3.36 menampilkan *mockup Pricing Section* pada halaman *home*. Tampilan pada *section* ini dirancang sesuai dengan desain *wireframe*, hanya saja dilakukan beberapa penyesuaian. Pada bagian daftar *card* yang tersedia, sekarang ditampilkan secara vertikal. Lalu bagian tombol “Pilih Paket” diganti menjadi tombol “Hubungi Kami” yang berfungsi untuk menghubungi langsung pihak pengelola.



Gambar 3.36. Tampilan *mockup Pricing Section* pada halaman *home*

E.1.6 *FAQ Section*

Gambar 3.37 menunjukkan *mockup FAQ Section* pada halaman *home*. Tampilan pada halaman ini dirancang berdasarkan desain *wireframe* yang telah dibuat, dengan judul pada bagian atas dan bagian baris pertanyaan yang disertai fitur *dropdown* untuk menampilkan informasi detail dari pertanyaan yang umum ditanyakan.

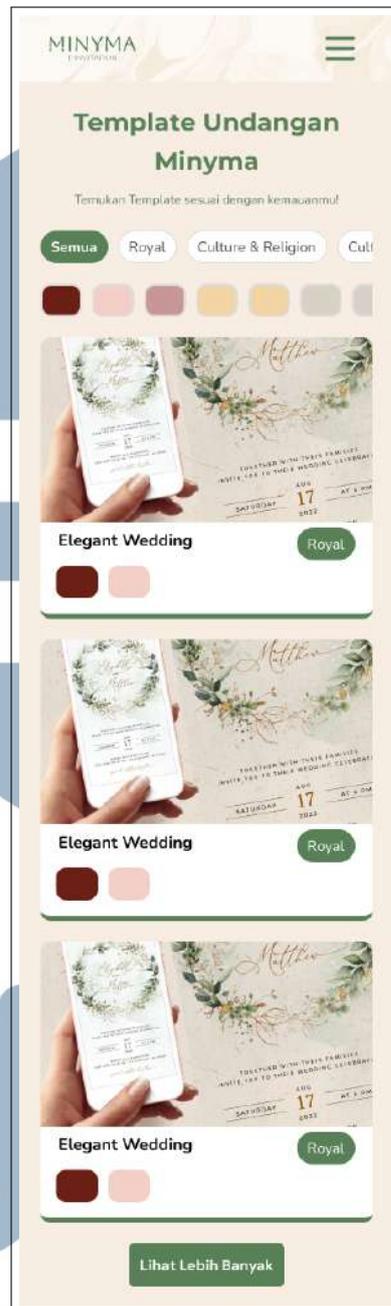


Gambar 3.37. Tampilan *mockup* FAQ Section pada halaman *home*

E.2 Halaman *Template*

Gambar 3.38 memperlihatkan tampilan *mockup* untuk halaman *Template*. Dirancang sesuai dengan desain *wireframe* yang telah dibuat sebelumnya. Di bagian atas terdapat judul, subjudul, serta fitur filter untuk memudahkan pengguna dalam menyaring pilihan. Lalu pada bagian konten menampilkan berbagai pilihan *template* yang tersedia, beserta tombol yang berfungsi untuk menampilkan lebih banyak *template*.

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.38. Tampilan *mockup* halaman *template*

E.3 Halaman Blog

Halaman Blog dirancang untuk menampilkan artikel informasi dan tips. Terdiri dari beberapa *section*, yaitu *hero*, dan *feature*. Berikut merupakan tampilan halaman blog.

E.3.1 *Hero Section*

Gambar 3.39 merupakan tampilan *mockup Hero Section* pada halaman blog. Tampilan dibuat berdasarkan desain *wireframe* yang telah dibuat, dengan penempatan judul di bagian tengah. Lalu, bagian latar belakang diberikan elemen visual berupa gambar.



Gambar 3.39. Tampilan *mockup Hero Section* pada halaman blog

E.3.2 *Feature Section*

Gambar 3.40 menunjukkan tampilan *mockup Feature Section* pada halaman blog. Pada *section* ini, dilakukan penyesuaian pada bagian yang menampilkan kategori tag yang sudah dipilih. Sekarang pengguna harus memilih tag terlebih dahulu, barulah tag yang dipilih akan muncul di bagian bawah filter.

U M I N
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



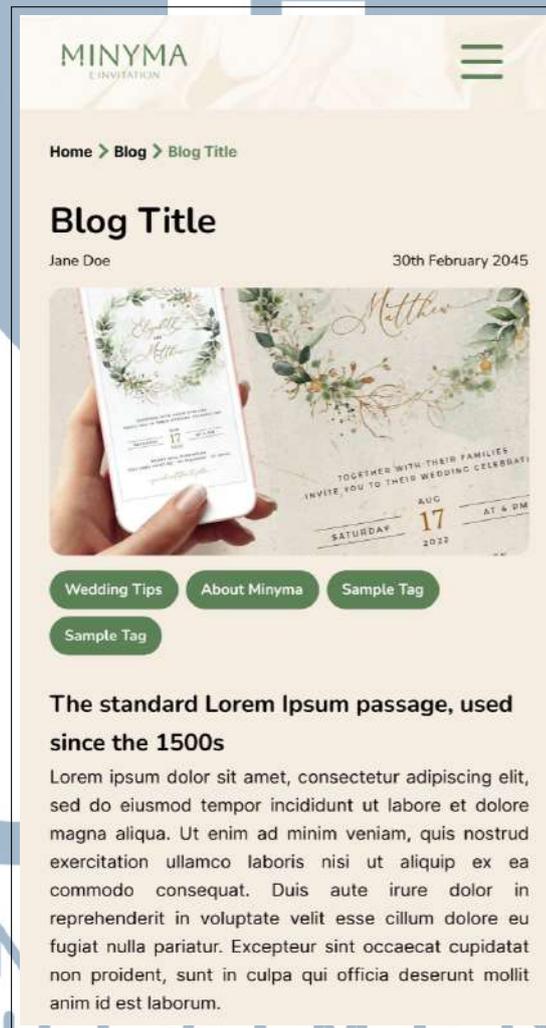
Gambar 3.40. Tampilan *mockup Feature Section* pada halaman blog

E.4 Halaman Blog Detail

Halaman Blog Detail dirancang untuk menampilkan secara detail isi dari masing-masing artikel yang tersedia. Halaman ini terdiri dari beberapa *section*, seperti *content section*, dan *file attachment section*. Berikut merupakan tampilan dari halaman blog detail.

E.4.1 Content Section

Gambar 3.41 adalah tampilan *mockup Content Section* pada halaman blog detail. Pada *section* ini, terdapat beberapa penyesuaian dari rancangan *wireframe* sebelumnya. Salah satunya adalah penempatan judul artikel, informasi penulis, dan tanggal publikasi yang sekarang ditempatkan di atas gambar. Selain itu, pada bagian kategori tag dihilangkan pada bagian color tag, sehingga hanya kategori saja yang ditampilkan.

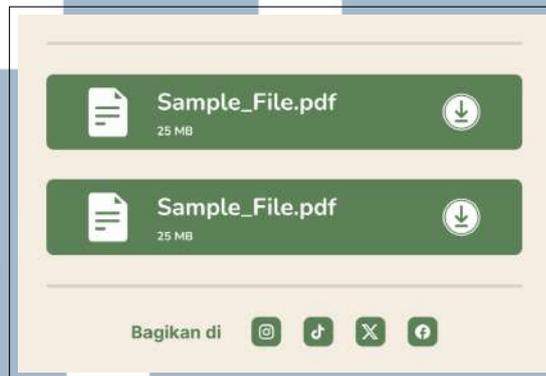


Gambar 3.41. Tampilan *mockup Content Section* pada halaman blog detail

E.4.2 File Attachment Section

Gambar 3.42 menampilkan *mockup File Attachment Section* pada halaman blog detail. Tampilan *section* ini dibuat berdasarkan desain *wireframe* yang telah

dibuat, hanya saja dilakukan penyesuaian pada bagian bawah *section*. Ditambahkan teks “Bagikan di” yang diletakkan sejajar dengan deretan ikon media sosial, sebagai ajakan agar pengunjung dapat membagikan artikel ke berbagai platform media sosial yang tersedia.



Gambar 3.42. Tampilan *mockup File Attachment Section* pada halaman blog detail

E.5 *Contact Section*

Gambar 3.43 adalah tampilan *mockup Contact Section*. Tampilan pada *section* ini dirancang sesuai desain dari *wireframe* yang telah dibuat. Pada bagian atas ditampilkan teks utama, diikuti oleh deskripsi singkat untuk memperkuat teks utama. Di bagian bawah, terdapat tombol “Hubungi Kami”. Lalu pada bagian latar belakang diberikan elemen visual berupa gambar.

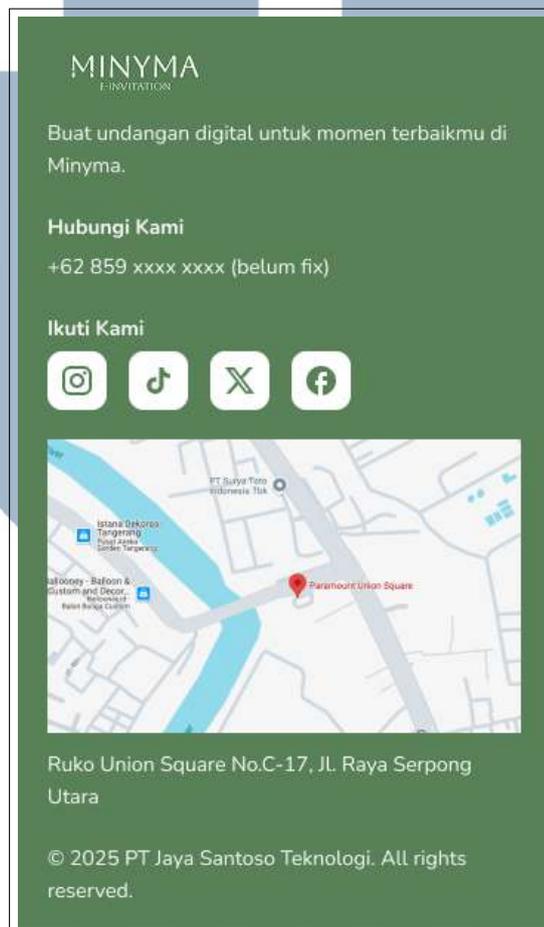


Gambar 3.43. Tampilan *mockup Contact Section*

E.6 *Footer Section*

Gambar 3.44 merupakan tampilan *mockup Footer Section*. Tampilan dirancang berdasarkan desain dari *wireframe* yang sudah dibuat. Bagian atas

menampilkan logo perusahaan, dengan kalimat ajakan untuk menarik perhatian pengunjung. Di bawahnya, tersusun informasi kontak, media sosial perusahaan, beserta peta dan alamat perusahaan sebagai penunjuk lokasi perusahaan. Di bagian paling bawah ditampilkan informasi hak cipta perusahaan.



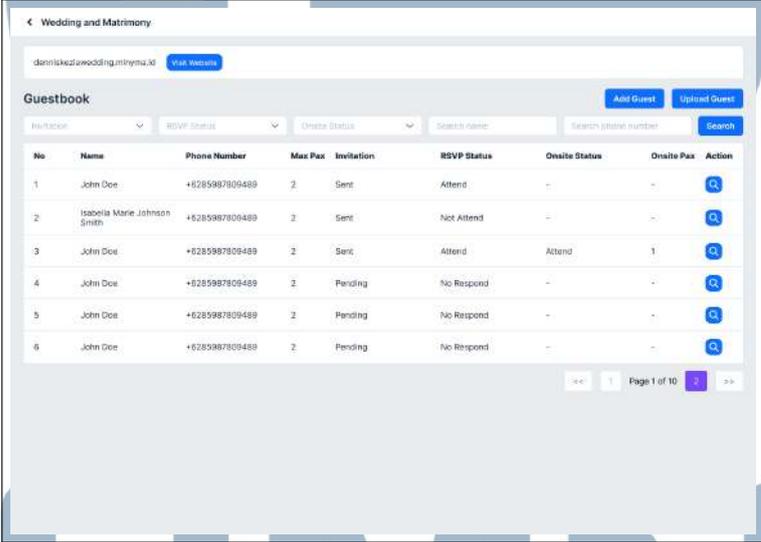
Gambar 3.44. Tampilan *mockup Footer Section*

E.7 Halaman *GuestMode*

Halaman *GuestMode* dirancang untuk menampilkan daftar *event* yang sedang diselenggarakan oleh *user*, dengan data yang ditampilkan berdasarkan ID *user* yang sedang *login*. Pada halaman ini, *user* dapat mengelola daftar tamu pada setiap *event* yang sedang diselenggarakan. Mulai dari melakukan *filter* tamu berdasarkan kriteria tertentu, menambahkan tamu secara manual, serta mengunggah data tamu dalam jumlah besar.

E.7.1 Event Detail Section

Gambar 3.45 adalah tampilan *mockup Event Detail Section* pada halaman *GuestMode*. Pada *section* ini, link *website* dari *event* yang diselenggarakan dicantumkan pada bagian atas. Di bagian inti halaman, terdapat daftar tamu dari *event* tersebut. Pada bagian daftar tamu ini, *user* bisa melakukan filter berdasarkan kriteria tertentu. Selain itu, tersedia dua opsi penambahan data tamu, yaitu secara manual dari fitur "Add Guest" atau menambahkan tamu dalam jumlah yang banyak dengan fitur "Upload Guest". Di bagian bawah tabel, terdapat elemen paginasi yang memungkinkan pengguna untuk berpindah halaman tabel ke halaman selanjutnya.

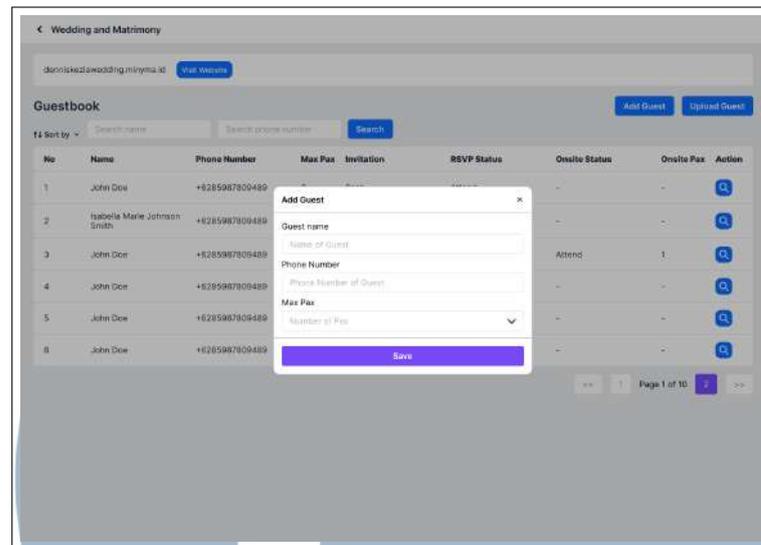


No	Name	Phone Number	Max Pax	Invitation	RSVP Status	Onsite Status	Onsite Pax	Action
1	John Doe	+6285987809489	2	Sent	Attend	-	-	
2	Isabella Marie Johnson Smith	+6285987809489	2	Sent	Not Attend	-	-	
3	John Doe	+6285987809489	2	Sent	Attend	Attend	1	
4	John Doe	+6285987809489	2	Pending	No Respond	-	-	
5	John Doe	+6285987809489	2	Pending	No Respond	-	-	
6	John Doe	+6285987809489	2	Pending	No Respond	-	-	

Gambar 3.45. Tampilan *mockup Event Detail Section*

E.7.2 Add Guest Modal

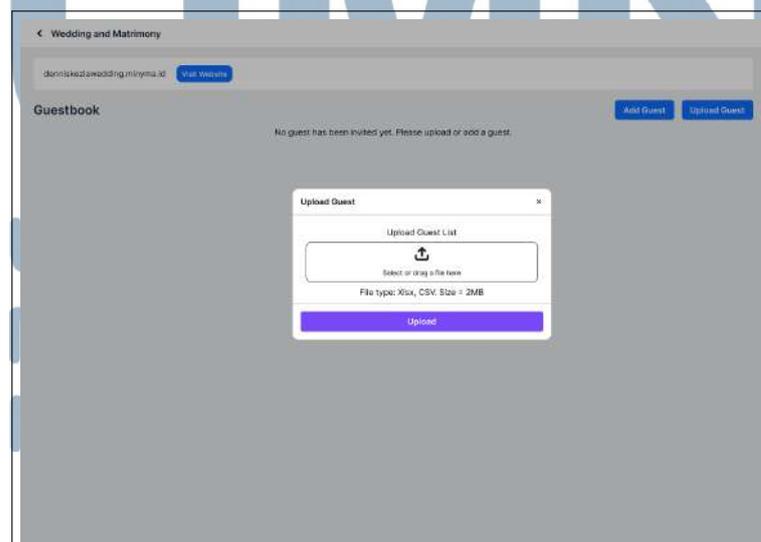
Gambar 3.46 adalah tampilan *mockup Add Guest Modal* pada *Event Detail Section*. Pada bagian ini, *user* akan diminta mengisi *form* yang digunakan untuk menambahkan data tamu secara manual. Jika *user* memilih jumlah "Max Pax" lebih dari satu, maka akan memunculkan bagian *companions* untuk menambahkan nama rekan tamu. Pada bagian bawah modal, terdapat tombol "Save" untuk menyimpan informasi yang sudah dimasukkan pada *form* ke *database*. Setelah proses penyimpanan berhasil, daftar tamu pada *event detail* akan diperbarui.



Gambar 3.46. Tampilan *mockup Add Guest Modal*

E.7.3 Upload Guest Modal

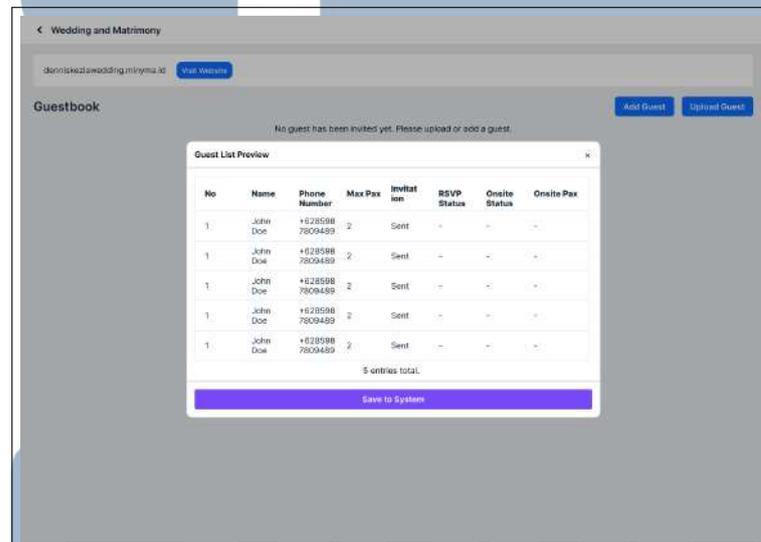
Gambar 3.47 adalah tampilan *mockup Upload Guest* pada *Event Detail Section*. Pada bagian ini, *user* dapat mengunggah data tamu ke kolom yang tersedia untuk menambahkan tamu dalam jumlah banyak. Sistem akan memvalidasi *file* yang diunggah baik dari segi format maupun ukuran. Jika *file* tidak sesuai dengan ketentuan yang ditentukan, sistem akan memunculkan *alert*. Namun, jika *file* valid, sistem akan melanjutkan ke tahap berikutnya untuk meninjau data yang akan disimpan.



Gambar 3.47. Tampilan *mockup Upload Guest Modal*

E.7.4 Guestlist Preview Modal

Gambar 3.48 adalah tampilan *mockup* *Guestlist Preview Modal* pada *Event Detail Section*, sebagai lanjutan dari proses *Upload Guest Modal*. Pada tampilan ini, sistem menampilkan isi dari data yang diunggah sebelumnya agar *user* dapat memastikan data sudah sesuai. Jika data sudah sesuai, *user* dapat melanjutkan proses dengan memilih tombol "Save to System" untuk menyimpan data. Jika data gagal tersimpan, sistem akan menampilkan *alert* "Failed to Upload Data". Sebaliknya, jika data berhasil tersimpan, sistem akan menampilkan "Upload Successful" dan data akan langsung disimpan ke dalam *database*. Setelah itu, data tamu pada *event detail* akan diperbarui.



Gambar 3.48. Tampilan *mockup* *Guestlist Preview Modal*

F Implementasi

Proses implementasi dilakukan dengan menentukan *framework* yang akan digunakan untuk merealisasikan desain *mockup* ke dalam bentuk *website*. *Website* dibuat dengan menggunakan bahasa pemrograman *Javascript* dengan *framework* *NextJS* untuk bagian *frontend*.

F.1 Halaman *Home*

Pada halaman *home*, terdapat beberapa elemen menu yang mengalami penyesuaian saat proses implementasi. Penyesuaian tersebut mencakup aspek

visual seperti ukuran font, serta penyesuaian pada fungsi-fungsi yang digunakan supaya sesuai dengan rancangan yang telah dibuat.

F.1.1 *Hero Section*

Pada bagian *hero* dilakukan penyesuaian pada ukuran *font* agar tampilannya lebih proporsional.

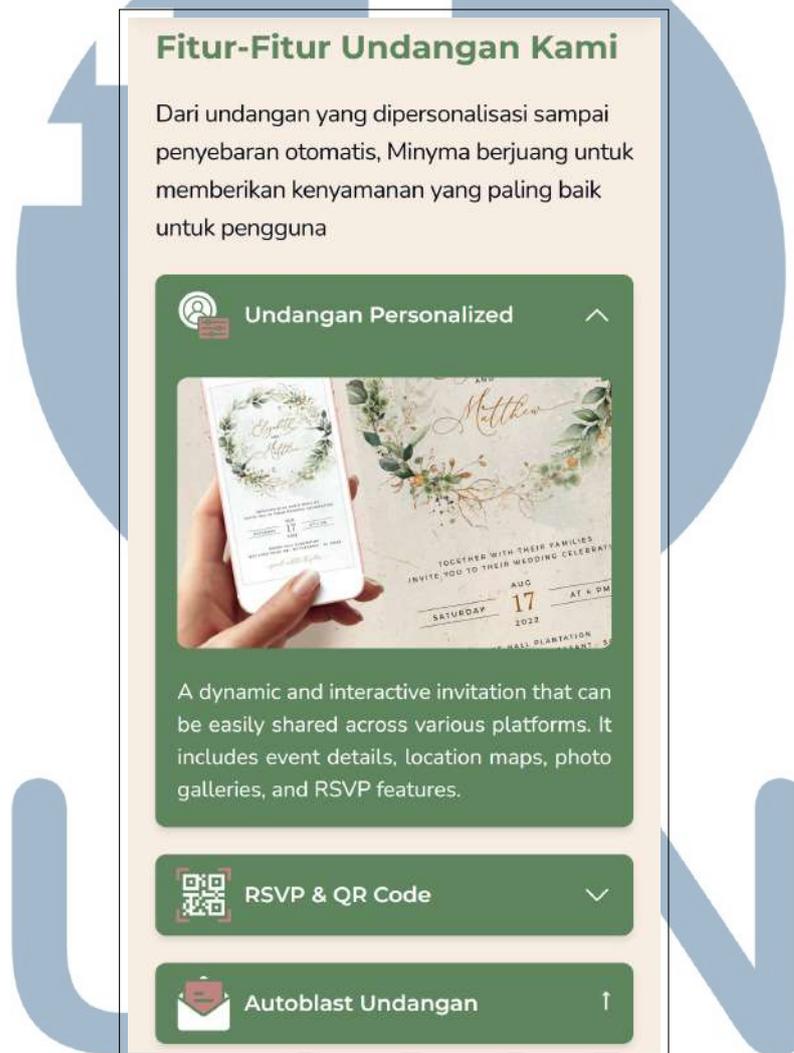


Gambar 3.49. Hasil implementasi *Hero Section* pada halaman *home*

Gambar 3.49 menampilkan hasil implementasi *Hero Section* pada halaman *home*. Pada bagian header, pengguna dapat menekan ikon *hamburger* untuk menampilkan menu navigasi yang digunakan dalam memilih halaman yang ingin diakses. Tombol “Hubungi Kami” berfungsi untuk mengarahkan pengguna ke halaman *Contact* untuk menghubungi pihak pengelola.

F.1.2 Service Section

Pada bagian *service* tidak ada perubahan dari hasil desain *mockup*. Namun, fitur seperti *dropdown* telah berhasil diimplementasikan dengan baik dan berfungsi.



Gambar 3.50. Hasil implementasi *Service Section* pada halaman *home*

Gambar 3.50 merupakan hasil implementasi *Service Section* pada halaman *home*. Pada bagian ini, ditampilkan berbagai layanan utama yang ditawarkan, masing-masing disertai dengan deskripsi singkat yang menjelaskan fungsi dan manfaat dari layanan tersebut. Penempatan elemen-elemennya dirancang untuk mempermudah dalam memberikan informasi secara ringkas dan jelas kepada pengguna.

F.1.3 *Customer Review Section*

Pada bagian *Customer Review* terdapat beberapa penyesuaian yang dibuat dari desain *mockup* sebelumnya. Seperti penyesuaian tata letak pada bagian ulasan, dan ditambahkan teks “Geser untuk melihat review lainnya” untuk memberi tahu pengguna bahwa bagian tersebut dapat digeser untuk melihat lebih banyak testimoni.



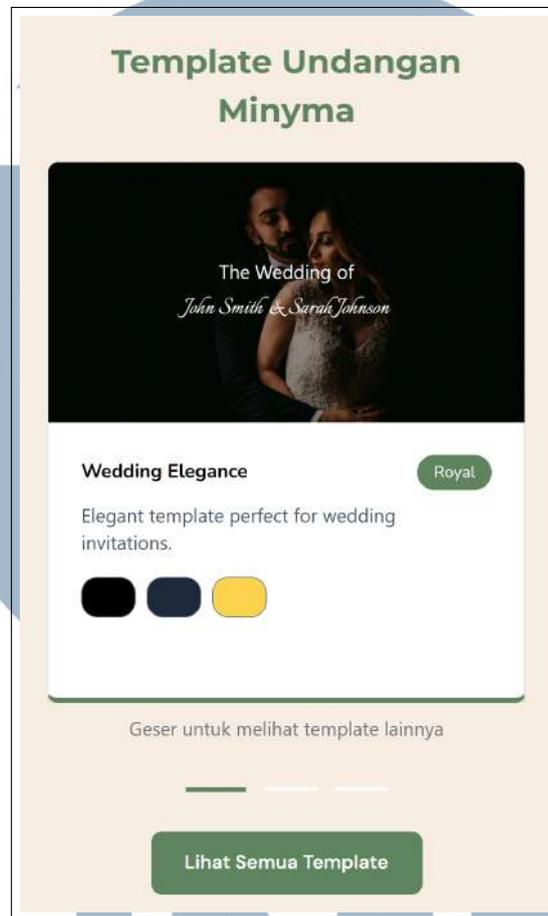
Gambar 3.51. Hasil implementasi *Customer Review Section* pada halaman *home*

Gambar 3.51 merupakan hasil implementasi *Customer Review Section* pada halaman *home*. Pada bagian ini, ditampilkan beberapa ulasan testimoni dari pelanggan yang telah menggunakan layanan Minyma E-Invitation. Setiap testimoni disajikan dengan nama dan tanggapan dari pelanggan tersebut.

F.1.4 *Template Section*

Pada bagian *Template* dilakukan beberapa penyesuaian bentuk pada bagian warna dan kategori. Selain itu, ditambahkan teks deskripsi yang ditempatkan di antara nama dan warna *template*. Ditambahkan juga teks “Geser untuk melihat

template lainnya” pada bagian atas indikator untuk memberi tahu bahwa bagian *template* dapat digeser untuk melihat pilihan *template* lainnya.



Gambar 3.52. Hasil implementasi *Template Section* pada halaman *home*

Gambar 3.52 merupakan hasil implementasi *Template Section* pada halaman *home*. Pada bagian ini, ditampilkan beberapa pilihan *template* undangan yang tersedia, lengkap dengan nama, deskripsi, kategori, dan palet warna yang menyesuaikan masing-masing *template*. Setiap *template* disajikan dalam elemen berbentuk *card*, yang dirancang untuk menyajikan informasi secara ringkas dan informatif. Tampilan ini memudahkan pengguna dalam melihat detail penting dari setiap *template* yang tersedia.

E.1.5 Pricing Section

Terdapat beberapa penyesuaian yang dilakukan pada bagian *Pricing* dari desain *mockup* sebelumnya. Pada bagian bawah judul ditambahkan indikator *dots*

untuk menunjukkan jumlah *card* yang tersedia. Selain itu, desain *pricing card* juga diperbarui, baik dari segi bentuk maupun interaksinya. Pengguna kini dapat menggeser *card* ke kiri dan kanan untuk menjelajahi berbagai pilihan paket, dengan bantuan ikon panah yang diletakkan di kedua sisi *card* untuk memudahkan navigasi.

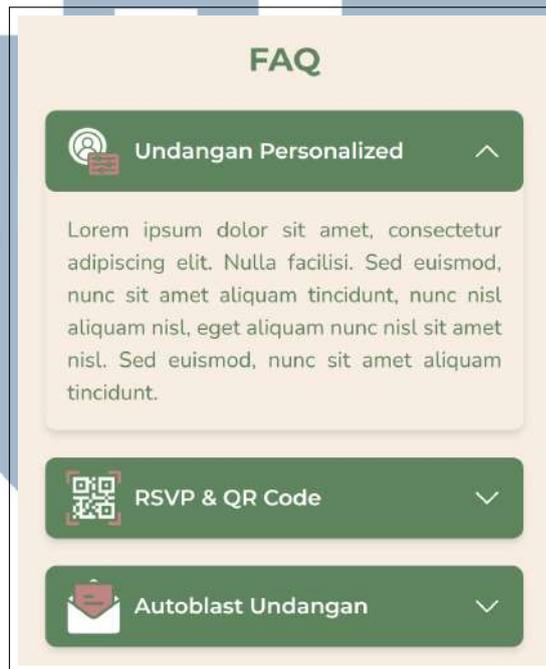


Gambar 3.53. Hasil implementasi *Pricing Section* pada halaman *home*

Gambar 3.53 merupakan hasil implementasi *Pricing Section* pada halaman *home*. Pada *section* ini, ditampilkan berbagai pilihan paket layanan, masing-masing ditampilkan dalam elemen *card* yang berisi fitur-fitur yang ditawarkan dalam setiap paket. Penggunaan elemen *card* ini bertujuan agar pengguna dapat dengan mudah membandingkan paket yang tersedia dan memilih sesuai kebutuhannya.

F.1.6 FAQ Section

Bagian *FAQ* dibuat berdasarkan hasil desain *mockup* sebelumnya. Namun, pada tahap implementasi, fitur interaktif seperti *dropdown* berhasil diterapkan dengan baik dan berfungsi sebagaimana mestinya.

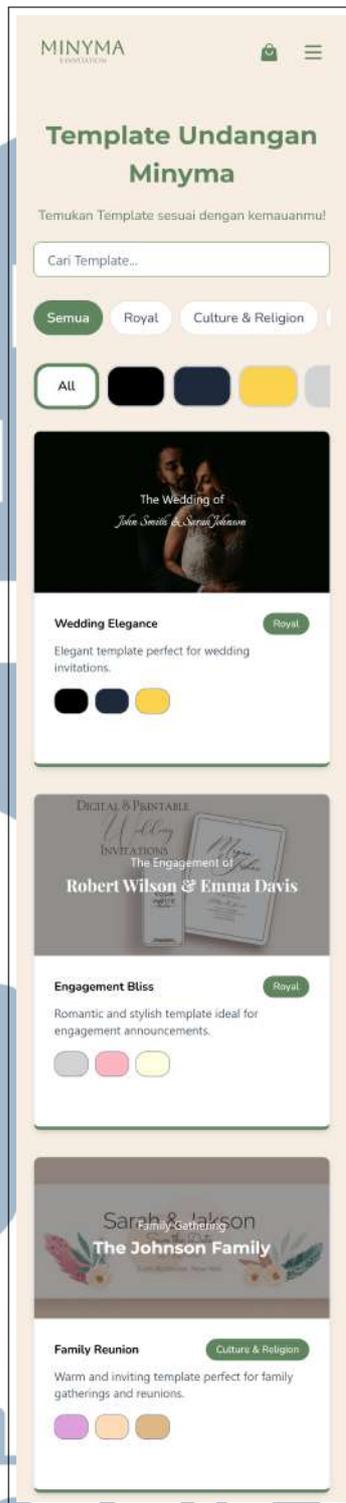


Gambar 3.54. Hasil implementasi *FAQ Section* pada halaman *home*

Gambar 3.54 merupakan hasil implementasi *FAQ Section* pada halaman *home*. Pada bagian ini, ditampilkan kumpulan pertanyaan yang sering diajukan oleh pengguna, lengkap dengan jawaban dari masing-masing pertanyaan. Bagian ini dirancang untuk membantu pengguna memahami layanan yang ditawarkan tanpa harus menghubungi pihak pengelola.

F.2 Halaman *Template*

Halaman *template* dibuat berdasarkan desain *mockup* sebelumnya, hanya saja dilakukan beberapa penyesuaian seperti bagian filter yang bisa digeser untuk melihat beberapa warna dan kategori lain yang tersedia. Lalu pada bagian *card* ditambahkan deskripsi singkat dari *template* tersebut. Selain itu, tombol “Lihat Lebih Banyak” pada bagian bawah juga dihilangkan.



Gambar 3.55. Hasil implementasi halaman *template*

Gambar 3.55 merupakan hasil implementasi halaman *template*. Pada halaman ini, ditampilkan semua koleksi *template* yang tersedia, lengkap dengan

deskripsi, kategori, dan palet warna dari masing-masing *template* yang ada. Selain itu, halaman ini dilengkapi dengan fitur pencarian yang memudahkan pengguna dalam menemukan *template* yang diinginkan, serta fitur filter yang memungkinkan pengguna untuk menyaring *template* berdasarkan kategori ataupun palet warna sesuai preferensi yang diinginkan.

F.3 Halaman Blog

Pada halaman blog, beberapa elemen mengalami penyesuaian pada proses implementasi. Penyesuaian yang dibuat seperti ukuran *font*, penambahan elemen pada beberapa bagian serta penyesuaian pada fungsi-fungsi yang digunakan supaya sesuai dengan rancangan pada desain *mockup*.

F.3.1 *Hero Section*

Pada bagian *hero* dilakukan penyesuaian terhadap ukuran dan tata letak *font* agar tampilannya lebih proporsional.

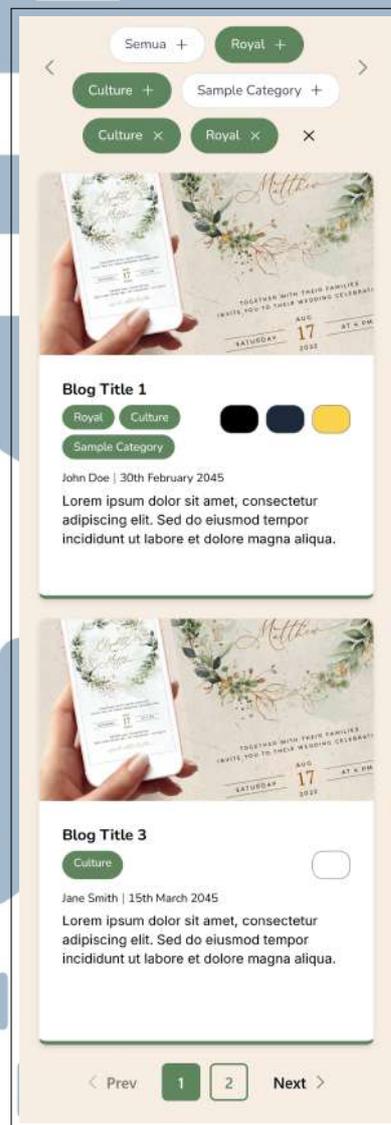


Gambar 3.56. Hasil implementasi *Hero Section* pada halaman blog

Gambar 3.56 merupakan hasil implementasi *Hero Section* pada halaman blog. Pada bagian ini ditampilkan latar belakang visual beserta teks di atasnya yang berfungsi untuk memberikan gambaran umum mengenai konten yang tersedia di halaman ini.

F.3.2 Feature Section

Pada bagian *Feature* dilakukan beberapa penyesuaian dari desain *mockup* sebelumnya. Fitur kategori tag pada bagian atas telah diimplementasikan. Saat pengguna memilih salah satu tag, maka tag yang dipilih secara otomatis akan ditampilkan di bagian bawah filter. Selain itu, ditambahkan juga palet warna dari masing-masing blog.



Gambar 3.57. Hasil implementasi *Feature Section* pada halaman blog

Gambar 3.57 merupakan hasil implementasi *Feature Section* pada halaman blog. Pada bagian ini, ditampilkan semua blog yang tersedia dalam elemen *card*, lengkap dengan nama penulis, tanggal rilis, kategori, dan deskripsi dari blog. Selain

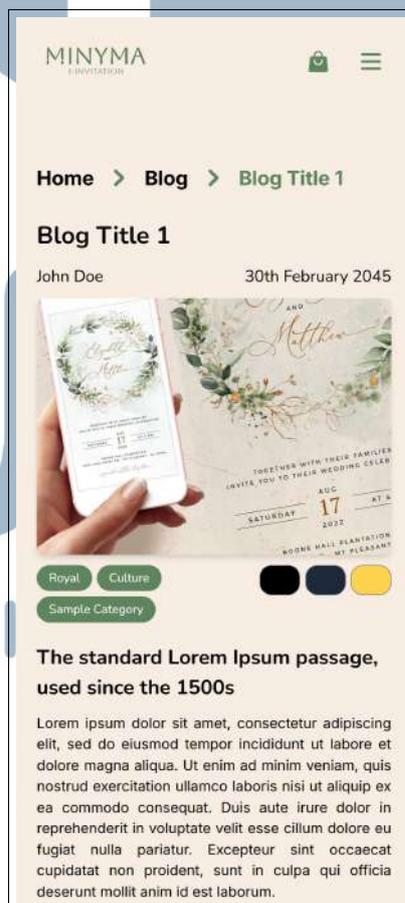
itu, halaman ini juga dilengkapi dengan fitur filter yang memungkinkan pengguna untuk menyaring blog berdasarkan kategori yang diinginkan.

F.4 Halaman Blog Detail

Pada halaman blog detail, dilakukan beberapa penyesuaian selama proses implementasi. Beberapa penyesuaian mencakup pengaturan ulang ukuran *font*, penambahan elemen pada beberapa bagian, serta penyesuaian pada fungsi-fungsi agar bekerja sesuai dengan rancangan pada desain *mockup*.

F.4.1 Content Section

Bagian ini, dibuat sesuai dengan desain *mockup* sebelumnya. Namun, dilakukan beberapa penyesuaian seperti menambahkan palet warna di samping kategori.

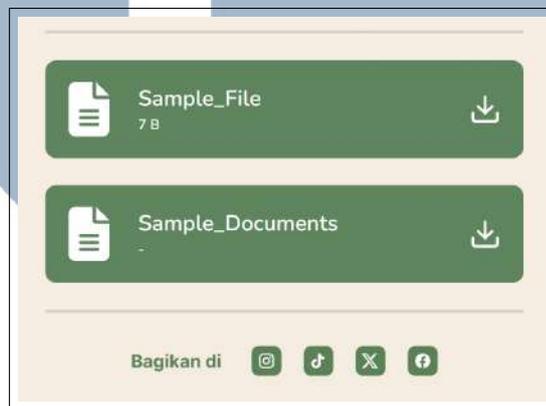


Gambar 3.58. Hasil implementasi *Content Section* pada halaman blog detail

Gambar 3.58 merupakan hasil implementasi *Content Section* pada halaman blog detail. *Section* ini, menampilkan informasi lengkap dari blog, mulai dari judul, nama penulis, tanggal rilis, kategori, hingga deskripsi blog tersebut. Halaman ini dirancang agar pengguna dapat memperoleh informasi yang berkaitan dengan layanan yang ditawarkan.

F.4.2 *File Attachment Section*

Pada bagian *File Attachment* dilakukan beberapa penyesuaian pada ikon dan ukuran *font*. Lalu, fitur *download* pada *file* juga sudah diimplementasikan. Selain itu, semuanya masih sesuai desain *mockup* sebelumnya.



Gambar 3.59. Hasil implementasi *File Attachment Section* pada halaman blog detail

Gambar 3.59 merupakan hasil implementasi *File Attachment Section* pada halaman blog detail. Bagian ini mencantumkan lampiran *file* yang berkaitan dengan isi blog.

F.5 *Contact Section*

Pada bagian *Contact*, tampilan diimplementasikan sesuai desain *mockup* yang telah dirancang sebelumnya. Hanya ditambahkan beberapa penyesuaian seperti penambahan efek *overlay* pada bagian latar.

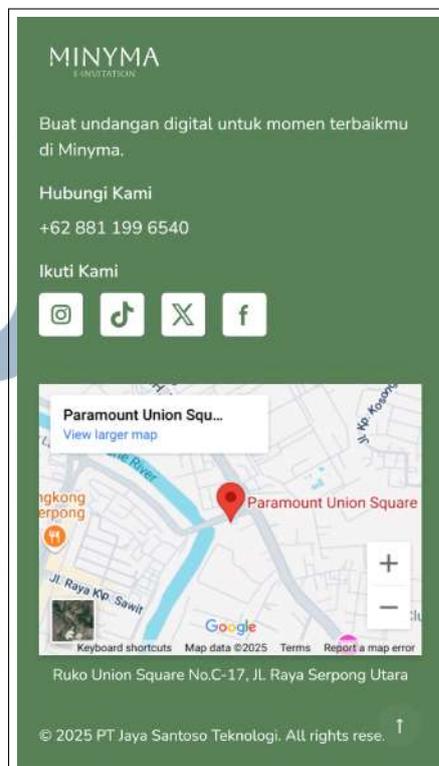


Gambar 3.60. Hasil implementasi *Contact Section*

Gambar 3.60 merupakan hasil implementasi *Contact Section*. Pada bagian ini ditampilkan teks yang berfungsi sebagai kalimat ajakan singkat beserta tombol “Hubungi Kami” untuk menghubungi pihak pengelola.

F.6 Footer Section

Pada bagian *Footer* dilakukan beberapa penyesuaian pada ukuran *font* dan bentuk elemen visualnya.



Gambar 3.61. Hasil implementasi *Footer Section*

Gambar 3.61 merupakan hasil implementasi *Footer Section*. Pada bagian ini ditampilkan informasi dari perusahaan, seperti nomor telepon, tautan menuju akun media sosial, serta alamat lengkap perusahaan. Selain itu, disediakan juga peta lokasi dari perusahaan untuk memudahkan pengguna yang ingin mengetahui lokasi perusahaan.

F.7 Halaman GuestMode

Pada halaman *GuestMode*, dilakukan beberapa penyesuaian tampilan dan fungsionalitas untuk meningkatkan pengalaman *user*. Penyesuaian ini mencakup sejumlah fitur yang telah berhasil diimplementasikan dan dapat berjalan.

F.7.1 Event Detail Section

Bagian ini dibuat sesuai dengan desain *mockup* yang telah dibuat sebelumnya. Namun, dilakukan penyesuaian pada fitur *filter* seperti penambahan elemen *dropdown* pada masing-masing kategori *tag*.

Invitation	RSVP Status	Onsite Status	Name	Phone Number	Action
Sent	Attend	Attend			
Pending	Not Attend	No Data			
	No Respond				
132	Amela Citra	1	Pending	Attend	
133	Jeswinica Santoso	+6281281127122	1	Pending	Attend
134	Juani Laoly	+6282129624254	1	Pending	Attend
137	Agustina	+6281383205656	2	Pending	Attend
139	Hengky Sudiarto	+6287888750555	2	Pending	Attend
141	Usay Viri Octasia	+6287854948999	1	Pending	Attend
142	Praptek	+6281310996136	2	Pending	Attend
144	Prahonggo sutejo	+6281210867023	2	Pending	Attend
146	Irene Lintong	+6285340278672	2	Pending	Attend

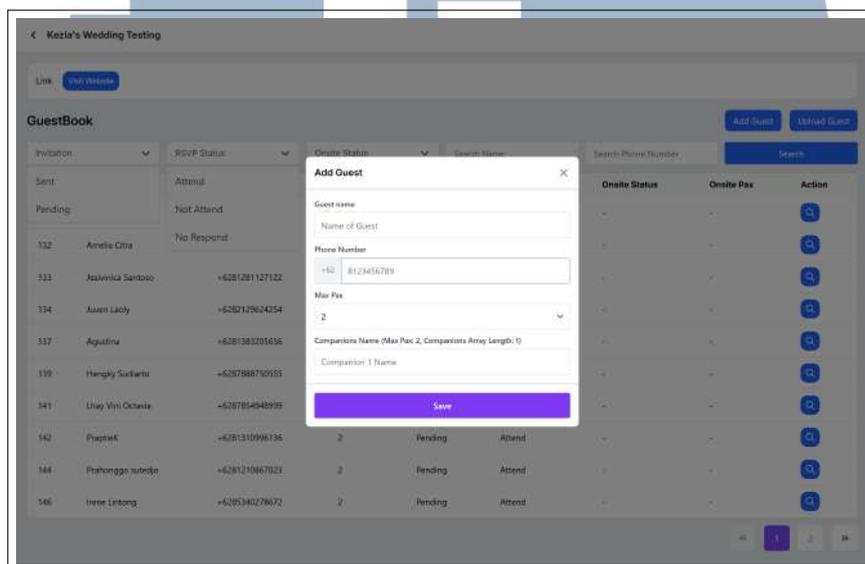
Gambar 3.62. Hasil implementasi *Event Detail Section*

Gambar 3.62 merupakan hasil implementasi *Event Detail Section* pada halaman *GuestMode*. Pada bagian ini, sistem menampilkan daftar tamu dari *event* yang dapat difilter berdasarkan kategori *tag* yang tersedia. Selain itu, *user* juga bisa menambahkan tamu secara manual dengan menekan tombol "Add Guest". Lalu jika *user* ingin mengunggah banyak tamu sekaligus, bisa dengan menekan tombol

”Upload Guest” dan nanti disediakan kolom untuk mengunggah *file* berisi data tamu. Elemen paginasi pada bagian bawah juga sudah berfungsi untuk berpindah antar halaman daftar tamu supaya mempermudah pengguna.

F.7.2 Add Guest Modal

Pada bagian *Add Guest Modal* dilakukan beberapa penyesuaian pada bagian *companions* yang akan terbuka jika *user* memasukkan *max pax* lebih dari satu.

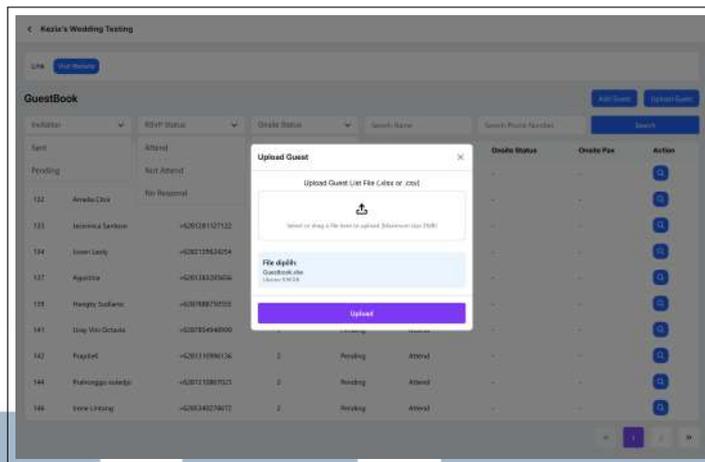


Gambar 3.63. Hasil implementasi *Add Guest Modal*

Gambar 3.63 merupakan hasil implementasi *Add Guest Modal* pada halaman *GuestMode*. Pada bagian ini ditampilkan sebuah *form* yang memungkinkan *user* untuk menambahkan data tamu secara manual. Lalu *user* juga bisa menyesuaikan ”*Max Pax*” untuk menambahkan ”*companions*”.

F.7.3 Upload Guest Modal

Pada bagian *Upload Guest Modal* dilakukan beberapa penyesuaian tampilan pada bagian bawah kolom unggah. Ditambahkan elemen yang menampilkan informasi *file* yang telah diunggah seperti format dan ukuran *file*.

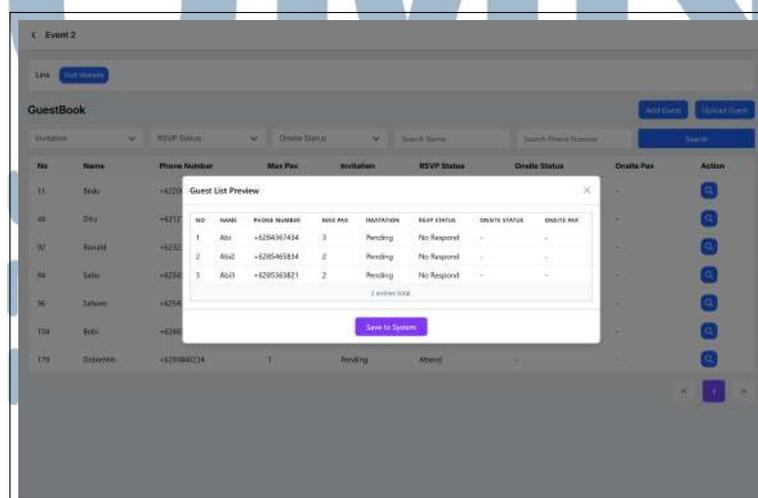


Gambar 3.64. Hasil implementasi *Upload Guest Modal*

Gambar 3.64 merupakan hasil implementasi *Upload Guest Modal* pada halaman *GuestMode*. Pada bagian ini disediakan kolom untuk mengunggah *file* yang sudah ditentukan format dan ukuran maksimalnya. Kolom upload yang tersedia memungkinkan *user* untuk menambahkan data tamu dalam jumlah banyak. Jika *file* sudah diunggah, maka informasi file akan terbaca pada bagian bawah kolom.

F.7.4 Guestlist Preview Modal

Bagian *Guestlist Preview Modal* dibuat sesuai dengan desain *mockup* sebelumnya. Seluruh elemen dan tampilan mengikuti rancangan awal tanpa adanya modifikasi, penyesuaian tambahan, atau perubahan yang dilakukan.



Gambar 3.65. Hasil implementasi *Guestlist Preview Modal*

Gambar 3.65 merupakan hasil implementasi *Guestlist Preview Modal* pada halaman *GuestMode*. Bagian ini menampilkan isi data tamu yang diunggah sebelumnya untuk ditinjau ulang. Jika data sudah sesuai, *user* dapat menekan tombol *save* untuk menyimpan data ke dalam sistem.

3.4 Spesifikasi Sistem

Berikut adalah spesifikasi dari *software* yang digunakan dalam pembuatan *website* Minyma E-Invitation.

1. *Github*
2. *Brave*
3. *Google Chrome*
4. *Visual Studio Code*
5. *Whatsapp*
6. *Figma*
7. *Diagrams.net*

Berikut merupakan spesifikasi *hardware* yang digunakan.

1. *Operating System: Windows*
2. *Random Access Memory (RAM): 16 GB*
3. *Storage: 1 TB (SSD)*
4. *Processor: Intel Core i7-13700HX*

3.5 Kendala dan Solusi yang Ditemukan

Kendala yang dialami selama pelaksanaan magang serta solusinya adalah sebagai berikut.

3.5.1 Kendala

Berikut adalah kendala yang dialami selama magang.

1. Setiap tugas yang diberikan memiliki tenggat waktu yang bervariasi.
2. Sulitnya menemukan waktu yang cocok untuk berdiskusi dengan rekan dari tim *Back End*.
3. Setiap bagian pada sistem *Back End* ditangani oleh individu yang berbeda-beda.
4. Rumit dalam menghubungkan bagian *Front End* dengan logika sistem *Back End* yang cukup kompleks.

3.5.2 Solusi

Berikut merupakan solusi untuk kendala yang dialami.

1. Menentukan prioritas berdasarkan tenggat waktu dan bobot, sehingga tugas yang lebih ringan dan memiliki batas waktu lebih cepat akan dikerjakan terlebih dahulu.
2. Menetapkan waktu diskusi bersama rekan *Back End* sesuai dengan waktu yang memungkinkan bagi kedua pihak.
3. Berkoordinasi dengan tim *Back End* untuk mengetahui siapa yang menangani bagian tertentu, agar alur kerja menjadi lebih jelas dan terorganisir.
4. Melakukan komunikasi rutin agar proses integrasi berjalan dengan lancar dan sesuai harapan.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A