

## BAB 3

### PELAKSANAAN KERJA MAGANG

#### 3.1 Kedudukan dan Koordinasi

Selama pelaksanaan magang, posisi yang diisi adalah sebagai *Front-end Developer* dengan tanggung jawab utama dalam pengembangan antarmuka pengguna pada proyek Website Kartu Digital. Fokus utama peran ini adalah menerjemahkan desain visual menjadi tampilan fungsional menggunakan framework Next.js. Implementasi dilakukan agar antarmuka dapat berjalan dengan baik di berbagai perangkat serta sesuai dengan alur sistem yang telah dirancang sebelumnya. Pengembangan dilakukan dengan memperhatikan prinsip responsive design dan kesesuaian antar komponen halaman.

Dalam pelaksanaannya, arahan diberikan langsung oleh supervisor yang juga menjabat sebagai *Project Manager*. Bimbingan dan masukan teknis disampaikan selama proses pengerjaan, termasuk pengecekan terhadap hasil kerja yang telah diselesaikan. Komunikasi dilakukan secara daring melalui platform seperti Zoom dan Discord, disesuaikan dengan kebutuhan dan urgensi proyek. Dengan pendekatan ini, kendala dapat segera diidentifikasi dan diselesaikan melalui diskusi intensif bersama tim.

Koordinasi kerja dilakukan dengan pendekatan berbasis tugas menggunakan platform Notion untuk pengelolaan dan pelacakan progres. Setiap pekerjaan dicatat dan dipantau secara terstruktur guna memastikan ketercapaian target mingguan. Diskusi rutin juga diadakan untuk membahas perkembangan tugas dan mengatasi hambatan yang muncul selama pengembangan. Sistem ini memungkinkan kegiatan magang tetap berjalan secara efisien meskipun dilaksanakan dalam skema *Work From Home*.

#### 3.2 Tugas yang Dilakukan

Tugas utama dalam kegiatan magang ini berfokus pada pengembangan dan penyempurnaan *Website Kartu Digital* yang dikembangkan oleh perusahaan Eight. Website ini ditujukan sebagai platform digital untuk pembuatan dan pengelolaan undangan, kartu nama, *link in bio*, dan *one page* secara mandiri. Proyek sudah memiliki fondasi awal, sehingga fokus pengembangan diarahkan pada penambahan fitur baru, perbaikan tampilan, dan penyempurnaan sistem agar lebih sesuai

dengan kebutuhan pengguna. Seluruh proses pengembangan dilakukan dengan mengacu pada standar best practice agar hasil kode terstruktur, efisien, dan mudah dikembangkan ke depannya.

Pengerjaan lebih difokuskan pada sisi frontend menggunakan framework *Next.js*. Tugas mencakup penerjemahan desain *UI* ke dalam komponen fungsional yang responsif dan adaptif di berbagai perangkat. Beberapa fitur yang dikembangkan antara lain RSVP kehadiran dan ucapan, pengelolaan daftar tamu, serta halaman publikasi undangan digital. Setiap pengembangan fitur dilakukan dengan mempertimbangkan integrasi dengan backend, sehingga data dapat tersinkronisasi dengan baik dan sistem berjalan sesuai alur.

Optimasi performa juga menjadi bagian penting dari pengembangan. Beberapa upaya dilakukan, seperti mengurangi beban loading halaman, mengimplementasikan teknik *lazy loading*, serta melakukan *client-side caching* untuk meningkatkan kecepatan akses. Struktur komponen juga disusun secara efisien agar penggunaan memori tidak berlebihan. Tujuannya adalah menciptakan pengalaman pengguna yang ringan, cepat, dan nyaman saat mengakses halaman kartu digital.

Dokumentasi teknis dibuat untuk setiap fitur yang telah diselesaikan. Dokumentasi ini memuat informasi terkait struktur kode, alur data, dependensi library yang digunakan, serta konfigurasi sistem. Tujuannya untuk memudahkan tim lain dalam memahami cara kerja sistem dan mempercepat proses pengembangan lanjutan di masa mendatang. Dokumentasi juga menjadi referensi penting dalam proses pemeliharaan sistem.

Struktur proyek disusun secara konsisten mengikuti standar pengembangan perangkat lunak. Penamaan folder, file, serta penempatan komponen diatur dengan rapi agar memudahkan navigasi kode. Langkah ini mendukung kolaborasi dalam tim dan mempercepat proses debugging maupun pengembangan fitur baru. Dengan tugas-tugas yang dijalankan, proyek *Website Kartu Digital* diharapkan mampu memberikan nilai guna tinggi bagi pengguna, sekaligus menjadi produk yang siap dikembangkan secara berkelanjutan.

### **3.3 Uraian Pelaksanaan Magang**

Tabel 3.1. Aktivitas/Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke-	Aktivitas/Pekerjaan yang Dilakukan
1	Proyek Sports Booking diselesaikan hingga tahap akhir, termasuk pembuatan halaman detail transaksi di sisi frontend dan integrasi dengan database. Pengembangan awal sistem Point of Sales (POS) dimulai menggunakan Next.js, TailwindCSS, dan shadcn/ui, dengan Firebase sebagai basis data.
2	Pengembangan website Kartu Digital dilanjutkan dengan eksplorasi kode backend dan perbaikan pada tampilan frontend. Penyesuaian dilakukan berdasarkan struktur kode yang ada, serta dieksplorasi mekanisme editor yang sesuai kebutuhan proyek.
3	Deploy pengujian dilakukan pada server VPS untuk backend dan frontend. Aplikasi server dan client berhasil dijalankan di lingkungan produksi. Belajar konfigurasi dilakukan pada pengaturan CORS dan Nginx.
4	Pengerjaan sementara dialihkan ke proyek pembuatan dua website company profile dalam satu perusahaan menggunakan WordPress, Elementor, dan plugin tambahan. Struktur awal situs disiapkan dan disesuaikan dengan kebutuhan perusahaan.
5	Kedua website WordPress diselesaikan dengan penambahan konten visual dan teks dari klien. Desain diperbaiki agar lebih responsif dan konsisten di berbagai perangkat.
6	Optimasi website dilakukan dengan penerapan caching dan pengaktifan Cloudflare untuk meningkatkan kecepatan akses dan efisiensi tampilan halaman.
7	Proyek baru berupa game berbasis web dimulai menggunakan Phaser.js. Inisialisasi proyek, instalasi pustaka, dan penyiapan aset game dilakukan sebagai tahap awal.
8	Revisi elemen pada game dilakukan sesuai masukan klien. Penambahan konten dan penyesuaian visual diselesaikan sebagai bagian dari persiapan finalisasi proyek.
9	Pengembangan kembali diarahkan ke website Kartu Digital. Fungsi untuk menampilkan daftar tamu dan halaman RSVP disusun agar tamu dapat memberikan respon kehadiran.

Minggu Ke-	Aktivitas/Pekerjaan yang Dilakukan
10	Penyesuaian dilakukan pada proses pengambilan data berdasarkan ID undangan (foreign key) masing-masing tamu. Integrasi API untuk pembaruan status kehadiran diselesaikan menggunakan metode patch.
11	Dua proyek dikerjakan secara paralel, yaitu penyesuaian tampilan smart dashboard pada website Kartu Digital agar unik per undangan, serta inisiasi proyek SaaS Photobooth menggunakan Next.js dan Electron.js.
12	Proyek game campaign dimulai menggunakan React dan Phaser.js. Penyesuaian rute API pada fitur kelola tamu di dashboard dilakukan. Pengembangan desktop app Pickbooth dilanjutkan dengan eksperimen integrasi kamera DSLR secara lokal tanpa SaaS.
13	Pengembangan lanjutan dilakukan untuk Pickbooth dan Kartu Digital. Proyek game campaign berbasis React dan filter TikTok diinisiasi. Fitur RSVP dan ucapan diintegrasikan dengan database. Pickbooth versi desktop berhasil dipublikasikan.
14	Proses pengambilan data RSVP dan ucapan dari tamu dilakukan, serta fitur pembaruan kehadiran dan publikasi ucapan diselesaikan. Backend dikembangkan menggunakan Nest.js dan frontend menggunakan Next.js. Pengalaman pertama dalam membangun backend secara mandiri diperoleh dengan bantuan dokumentasi dan referensi daring.

### 3.4 Perancangan Website

Pada bagian ini, akan dijelaskan perancangan *Website Kartu Digital* yang dikerjakan mulai dari Metodologi Pengembangan Sistem, *User Requirement*, *Sitemap*, *Flowchart*, dan *Wireframe*

#### 3.4.1 Metodologi Pengembangan Sistem

Pada pengembangan website Kartu Digital di Eiqht, tim menerapkan metodologi Agile untuk memastikan proses pengembangan berjalan secara adaptif dan kolaboratif. Agile dipilih karena mampu menyesuaikan kebutuhan yang

dinamis selama proyek berlangsung, serta mendorong komunikasi yang efektif antar anggota tim. Pengembangan dilakukan secara remote dengan memanfaatkan tools kolaborasi seperti Zoom, Notion, dan Discord, sehingga koordinasi tetap terjaga meskipun tidak berada di lokasi yang sama.

Metodologi Agile menekankan pada pengembangan secara iteratif dan inkremental, di mana pekerjaan dibagi ke dalam beberapa siklus singkat yang disebut sprint. Setiap sprint biasanya berlangsung selama satu hingga dua minggu dan berfokus pada penyelesaian fitur atau modul tertentu yang dapat langsung diuji dan dievaluasi. Dengan cara ini, tim dapat dengan cepat merespons perubahan kebutuhan atau masukan dari stakeholder, serta melakukan perbaikan secara berkelanjutan pada sistem yang dikembangkan.

Tahapan dalam pengembangan sistem menggunakan Agile dimulai dari perencanaan sprint, di mana tim menentukan backlog atau daftar fitur yang akan dikerjakan pada periode tertentu. Setelah itu, pengembangan dan integrasi dilakukan secara bertahap, dimulai dari perancangan wireframe, implementasi frontend dengan Next.js, backend dengan Express.js, hingga integrasi API. Setiap anggota tim memiliki tanggung jawab masing-masing, namun tetap saling berkoordinasi untuk memastikan semua bagian sistem berjalan dengan baik.

Selama proses pengembangan, tim melakukan daily standup atau koordinasi harian untuk membahas progres, hambatan, dan solusi yang diperlukan. Komunikasi intensif ini membantu mengidentifikasi kendala lebih awal dan mempercepat proses penyelesaian masalah. Setelah setiap sprint berakhir, dilakukan review dan retrospektif untuk mengevaluasi hasil kerja, mengidentifikasi area yang perlu ditingkatkan, serta merencanakan langkah selanjutnya agar proses pengembangan semakin efisien.

Dengan penerapan metodologi Agile, pengembangan website Kartu Digital menjadi lebih responsif terhadap perubahan dan kebutuhan pengguna. Proses iteratif dan kolaboratif yang dijalankan mampu meningkatkan kualitas produk, mempercepat waktu rilis fitur, serta memberikan pengalaman belajar yang berharga bagi seluruh anggota tim. Hasil akhirnya adalah sistem yang lebih stabil, fungsional, dan sesuai dengan harapan stakeholder maupun pengguna akhir.

### **3.4.2 User Requirement**

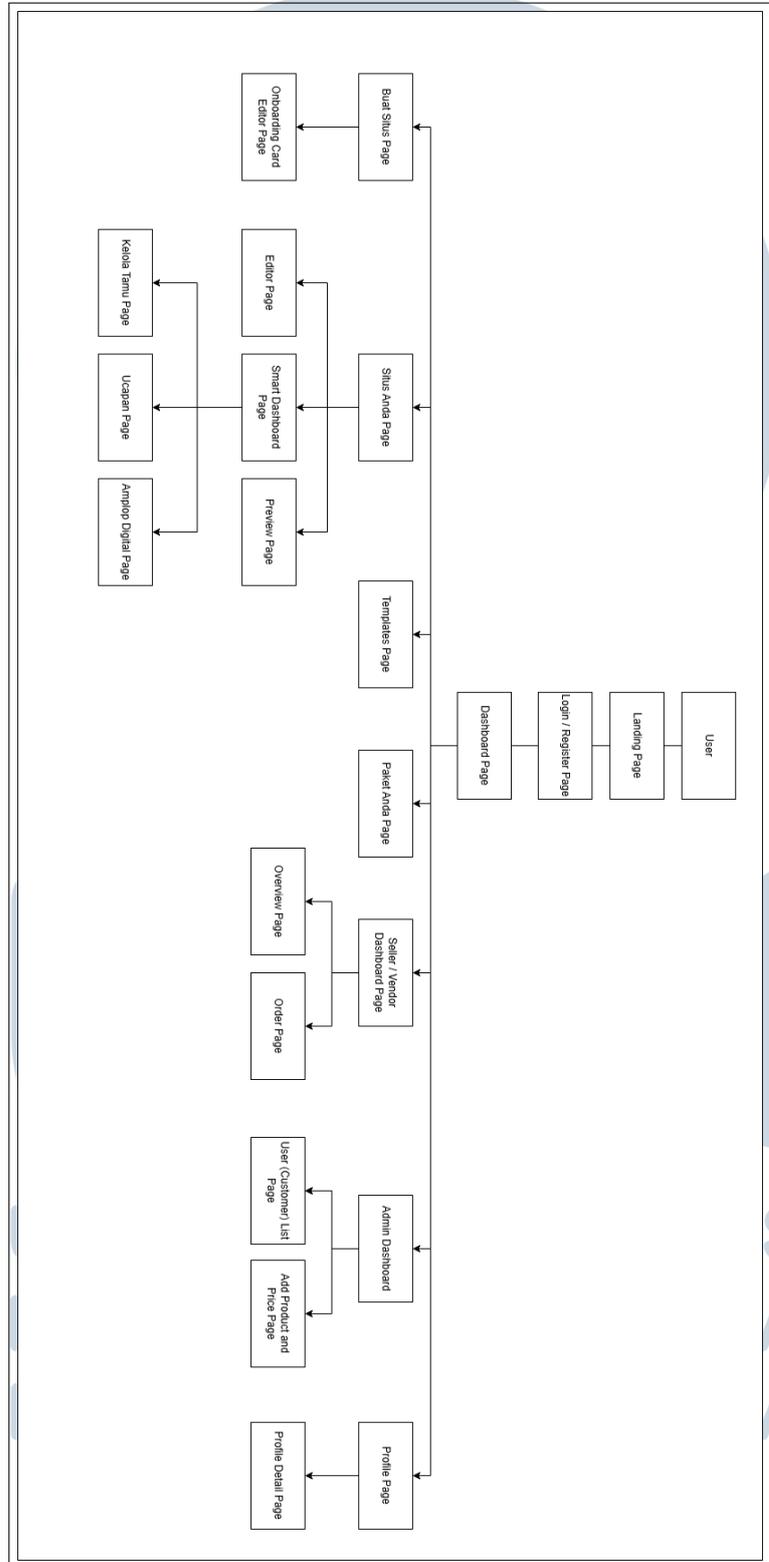
Sebelum melakukan pengembangan lanjutan dari website Kartu Digital, *Project Manager* memberikah arahan terkait tugas untuk mengerjakan lanjutan

pengembangan Kartu Digital, diantaranya sebagai berikut:

1. Mengembangkan bagian *register* untuk melakukan pendaftaran bagi user. Dari progress sebelumnya sudah terdapat frontend, dan untuk pengembangan ini hanya perlu melakukan integrasi backend api yg sudah dibuat, lalu menghubungkannya dengan frontend.
2. Mengembangkan bagian *profile* untuk menunjukkan data dari user. Disini user dapat mengubah data yang lama dan mengganti foto profil baru.
3. Mengembangkan fitur kelola tamu bagi pemilik undangan, untuk dapat mengelola tamu yang akan mengikuti acara. Dengan ada fitur auto broadcast undangan melalui whatsapp dengan link yang akan berisi undangan jadi. Selain itu, terdapat link lain yang dibagi untuk RSVP, RSVP berguna untuk mengkonfirmasi kedatangan sang tamu dan juga tempat memberi ucapan selamat.
4. Mengembangkan fitur ucapan untuk pemilik undangan dapat menangani ucapan-ucapan yang masuk melalui RSVP ucapan. Ucapan yang diberikan oleh tamu tidak langsung muncul dalam RSVP, karena pemilik undangan memiliki dapat membuat pilihan untuk memublishnya atau menghapusnya.
5. Mengembangkan fitur amplop digital, untuk mengelola masuknya uang dari acara untuk dapat dipantau dan dapat ditarik melalui website Kartu Digital.

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

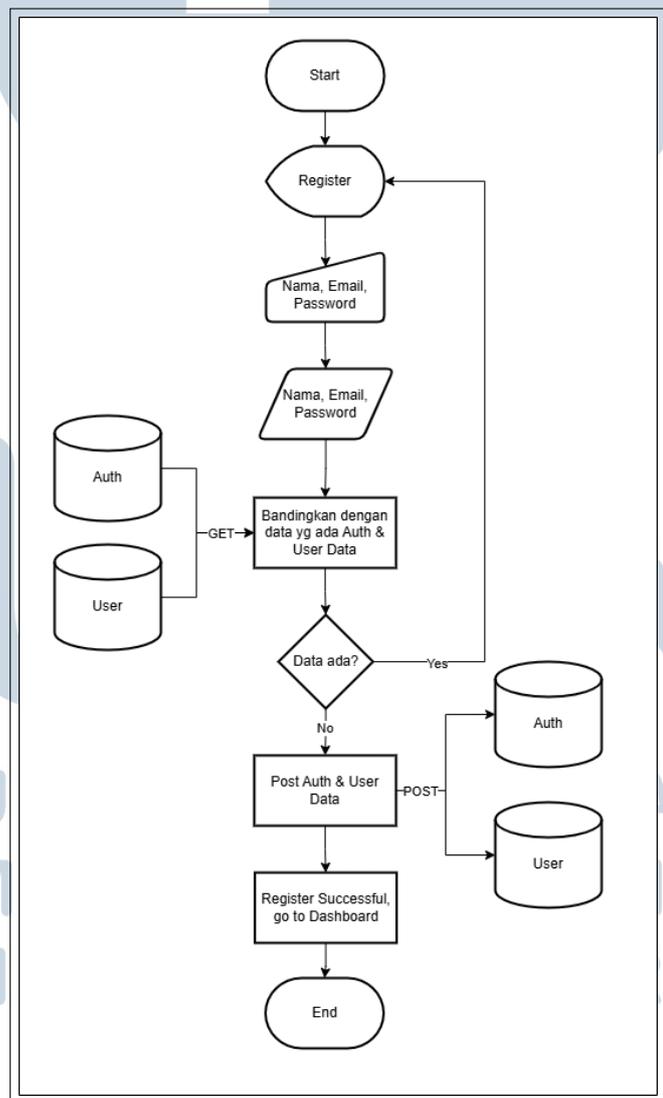
### 3.4.3 Sitemap



Gambar 3.1. Sitemap Website Kartu Digital

Gambar 3.1 merupakan gambar *sitemap* berdasarkan *Website Kartu Digital* yang sedang dikembangkan. Dari *sitemap* yang sudah ada, pada tahap awal user akan melihat *login page*, namun bila dia belum memiliki akun dia bisa memilih *create account*. Ketika user sudah berhasil *login/register*, maka user akan beralih ke halaman *dashboard*. Dari *dashboard*, user dapat mengakses halaman lainnya, yaitu *Buat Situs Page*, *Situs Anda Page*, *Templates Page*, *Paket Anda Page*, *Seller Dashboard*, *Profile Page*. Untuk bagian *Vendor Dashboard* dan juga *Admin Dashboard* adalah user yang memiliki role "ADMIN", ataupun "VENDOR".

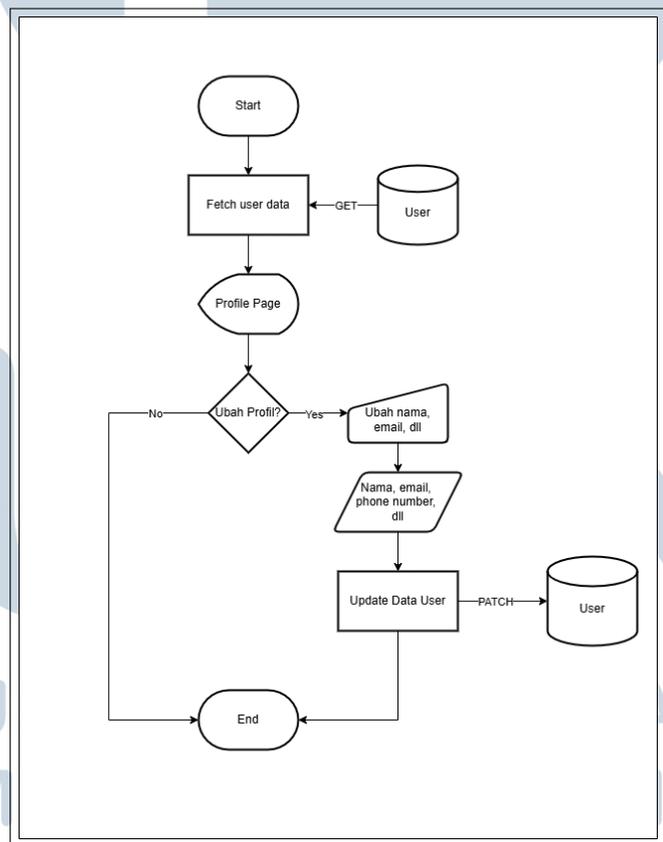
### 3.4.4 Flowchart



Gambar 3.2. Flowchart Register

Gambar 3.2 menunjukkan flowchart proses registrasi pada Website Kartu Digital. Proses dimulai ketika pengguna mengakses halaman registrasi dan mengisi formulir yang berisi data seperti nama, email, dan password. Data yang telah diinput kemudian diproses dan dibandingkan dengan data yang sudah ada pada database Auth dan User untuk memastikan tidak terjadi duplikasi akun.

Jika data pengguna sudah terdaftar, maka proses registrasi akan dihentikan dan pengguna diminta untuk melakukan registrasi ulang dengan data yang berbeda. Namun, apabila data tersebut belum ada di database, sistem akan melakukan proses penyimpanan (POST) data ke database Auth dan User. Setelah data berhasil disimpan, pengguna akan mendapatkan notifikasi bahwa registrasi telah berhasil dan secara otomatis diarahkan ke halaman dashboard. Dengan alur ini, sistem dapat memastikan setiap pengguna yang mendaftar adalah pengguna baru dan mencegah terjadinya duplikasi data pada database.



Gambar 3.3. Flowchart Profile

Gambar 3.3 menunjukkan flowchart dari fitur Profile pada Website Kartu Digital. Proses dimulai saat pengguna mengakses halaman profil. Sistem akan melakukan fetching data (pengambilan data) dari user yang sedang login melalui

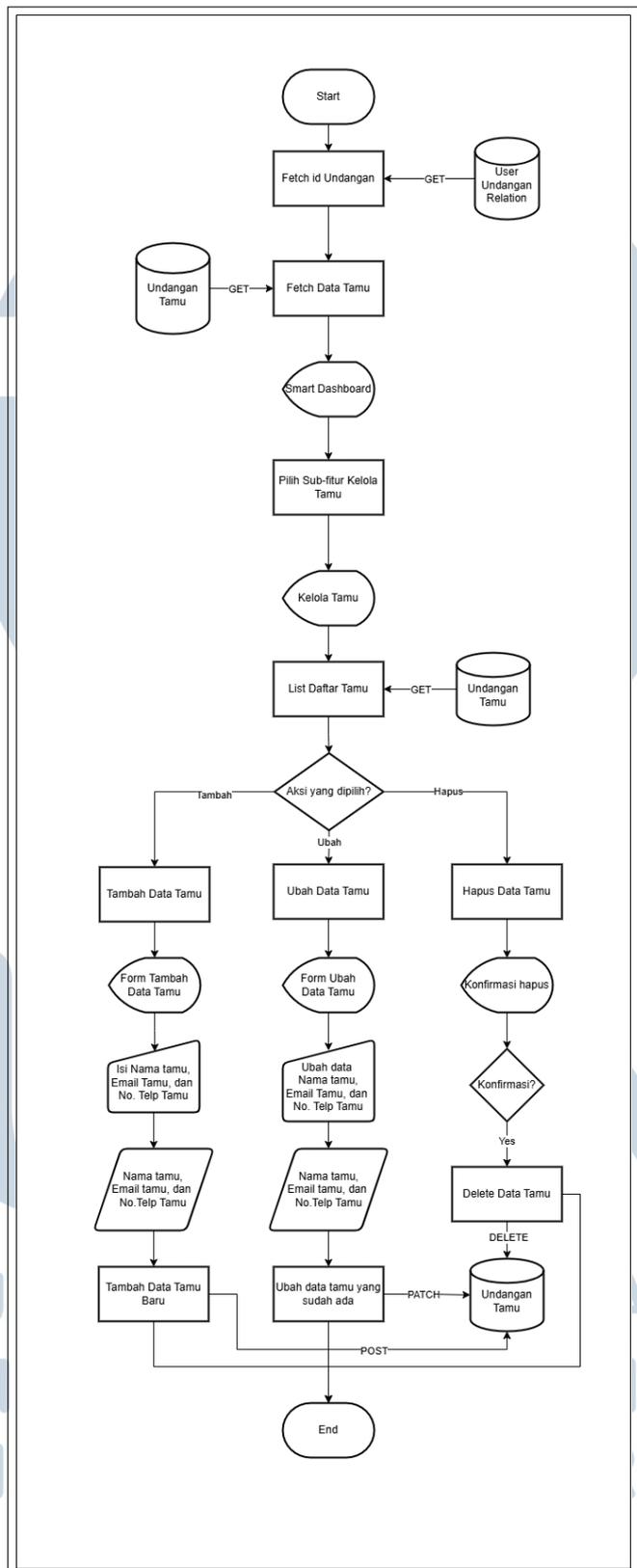
metode GET dari basis data. Data yang berhasil diambil akan ditampilkan pada halaman profil, memungkinkan pengguna untuk melihat informasi pribadinya seperti nama, email, dan nomor telepon.

Setelah halaman profil ditampilkan, pengguna memiliki dua opsi, yaitu hanya melihat-lihat data tanpa melakukan perubahan, atau memilih untuk mengubah data profil yang tersedia. Jika pengguna tidak ingin melakukan perubahan, maka proses akan langsung berakhir.

Namun, apabila pengguna memilih untuk mengubah data, maka sistem akan mengarahkan ke tahap pengisian data baru seperti nama, email, nomor telepon, dan informasi lainnya yang diinginkan. Setelah pengguna mengisi data tersebut, sistem akan memproses pembaruan data melalui metode PATCH dan menyimpannya kembali ke basis data pengguna.

Proses ini memastikan bahwa data pengguna yang tersimpan di sistem selalu up-to-date dan sesuai dengan informasi terbaru yang diberikan oleh pengguna. Setelah data berhasil diperbarui, sistem akan mengakhiri proses dan pengguna akan melihat data terbaru yang telah disimpan.





Gambar 3.4. Flowchart Kelola Tamu

Gambar 3.4 merupakan gambar *flowchart* untuk subfitur Kelola Tamu pada halaman Smart Dashboard. Sistem akan memulai dengan proses Fetch id Undangan, yang mengambil relasi antara user dan undangan menggunakan metode GET dari entitas User Undangan Relation. Setelah itu, sistem mengarahkan ke halaman Smart Dashboard, kemudian pengguna perlu klik subfitur Kelola Tamu untuk ke menu Kelola Tamu.

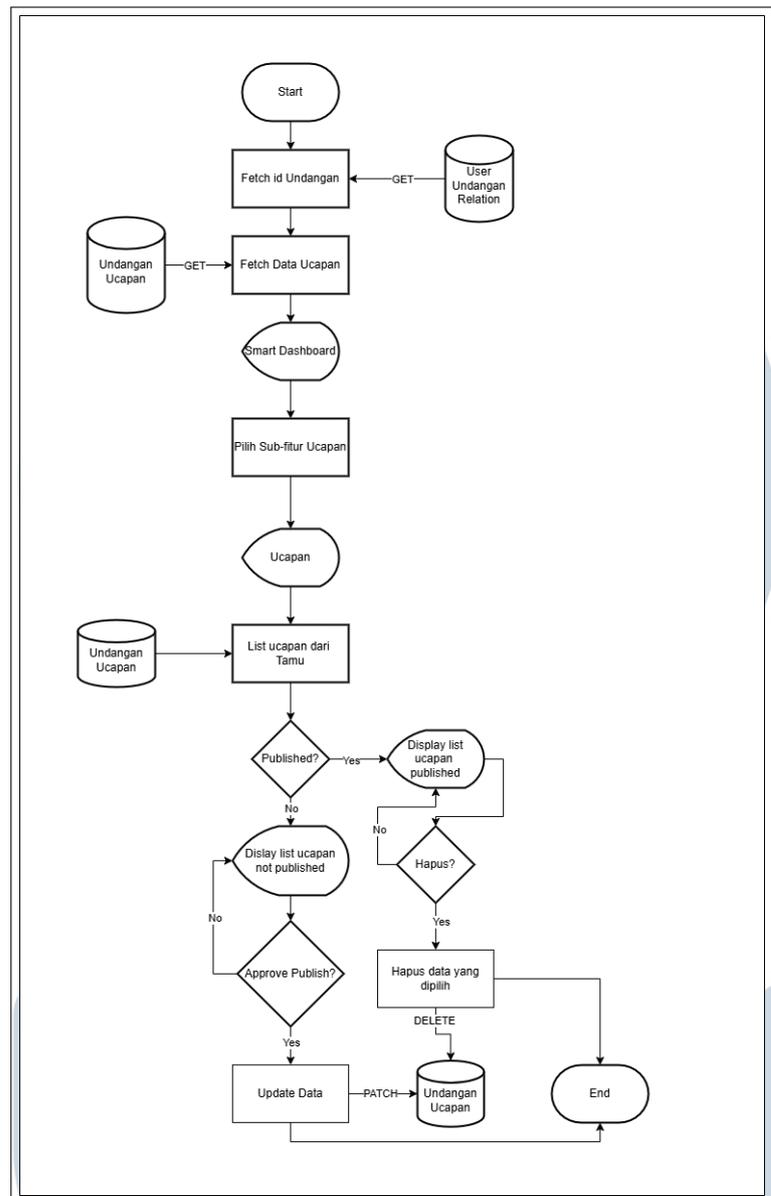
Dari menu Kelola Tamu, sistem akan menampilkan List Daftar Tamu yang mengambil data tamu dari Undangan Tamu menggunakan metode GET. Setelah daftar tamu muncul, user memiliki tiga opsi aksi yang dapat dilakukan, yaitu Tambah Data Tamu, Ubah Data Tamu, dan Hapus Data Tamu.

Jika user memilih Tambah Data Tamu, sistem akan menampilkan Form Tambah Data Tamu. User akan mengisi informasi berupa Nama tamu, Email tamu, dan No. Telp tamu. Setelah itu, data akan dikirimkan untuk proses penambahan sebagai Tambah Data Tamu Baru, dan sistem menyimpan data dengan metode POST.

Jika user memilih Ubah Data Tamu, sistem akan menampilkan Form Ubah Data Tamu. User dapat memperbarui Nama tamu, Email tamu, dan No. Telp tamu, kemudian sistem akan memperbarui data tersebut sebagai Ubah Data Tamu yang Sudah Ada, dan menyimpan pembaruan menggunakan metode UPDATE dan POST.

Jika user memilih Hapus Data Tamu, sistem akan menampilkan halaman konfirmasi. Jika user menyetujui proses konfirmasi (*Yes*), sistem akan menjalankan proses Delete Data Tamu, dan data tamu akan dihapus dari entitas Undangan Tamu menggunakan metode DELETE.

U M M 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

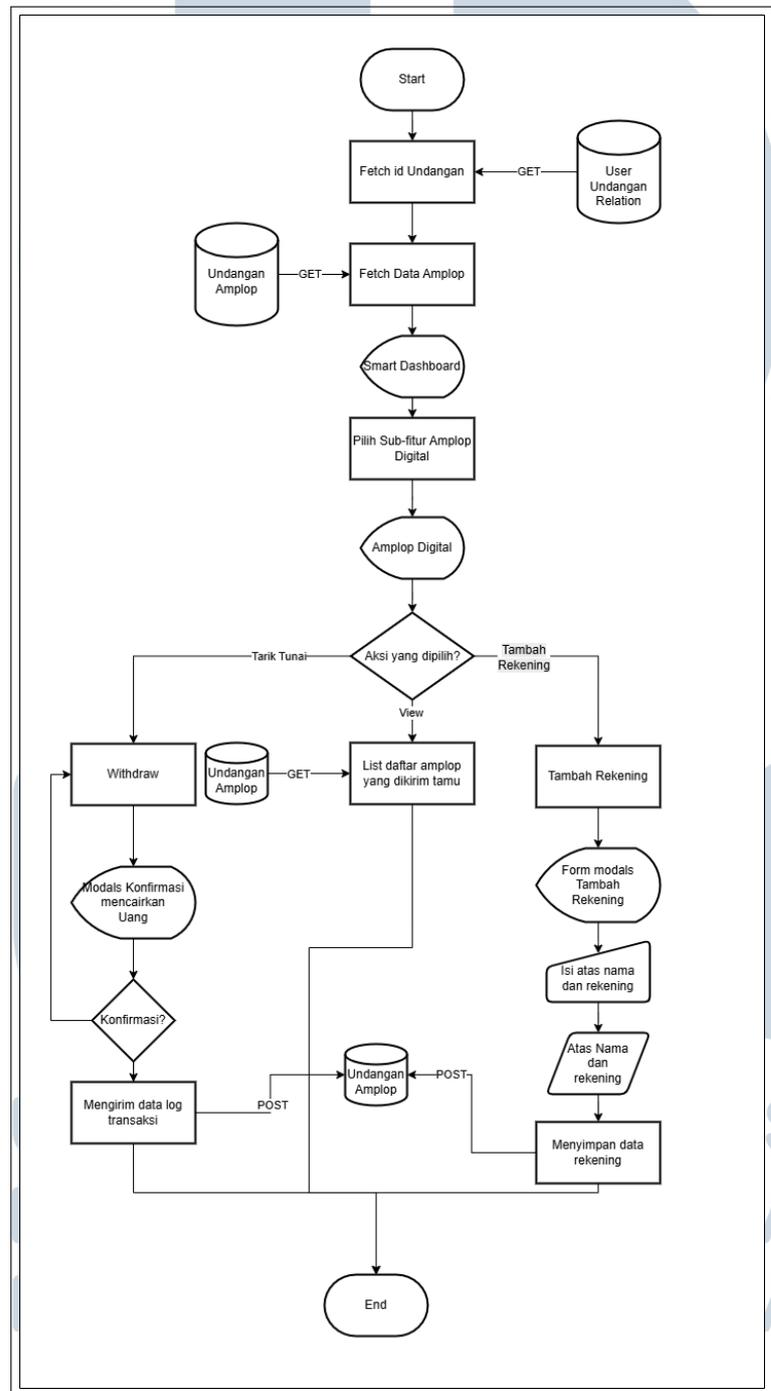


Gambar 3.5. Flowchart Ucapan

Gambar 3.5 merupakan gambar *flowchart* untuk subfitur Ucapan dari halaman Smart Dashboard. Sistem dimulai dari Fetch Id Undangan, untuk mengambil relasi user dengan undangan menggunakan metode GET dari entitas User Undangan Relation, setelah itu akan masuk ke halaman Smart Dashboard dan pengguna perlu klik subfitur Ucapan untuk ke menu Ucapan. Sistem akan menampilkan list data ucapan dari semua tamu dengan menggunakan GET.

Dalam halaman Ucapan Smart Dashoard ini user dapat melihat ucapan-ucapan yang di publish maupun tidak di publish. Ucapan yang tidak di publish,

dapat di setuju untuk di publish oleh user, dan akan melakukan update dengan method PATCH ke tabel data undangan ucapan. Ucapan yang sudah dipublish, dapat dihapus oleh user dan fengan method DELETE ke tabel data undangan ucapan.



Gambar 3.6. Flowchart Amplop Digital

Gambar 3.6 menunjukkan flowchart dari fitur Amplop Digital pada Website Kartu Digital. Proses dimulai ketika sistem mengambil ID undangan berdasarkan user yang sedang login melalui relasi dari tabel User Undangan Relation. Setelah ID berhasil diambil, sistem melanjutkan dengan mengambil data amplop dari database Undangan Amplop.

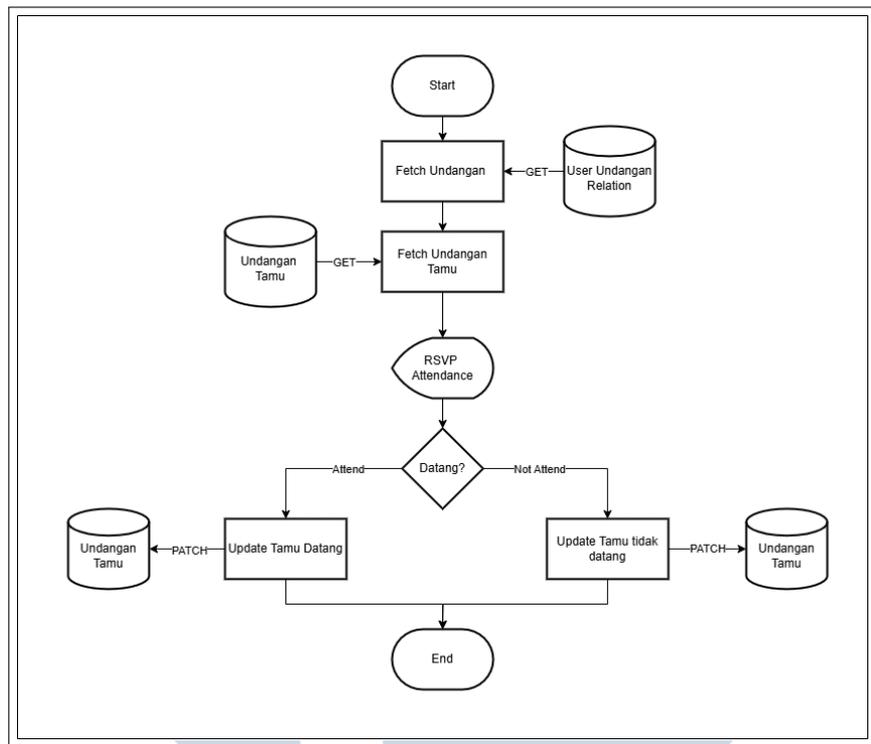
Setelah data berhasil diambil, pengguna diarahkan ke halaman Smart Dashboard, lalu pengguna perlu klik Amplop Digital untuk masuk ke menu Amplop Digital. Di halaman ini, sistem menampilkan daftar amplop digital yang telah dikirimkan oleh tamu undangan kepada pengguna. Data tersebut ditampilkan berdasarkan hasil pengambilan data menggunakan metode GET.

Pengguna memiliki dua pilihan dalam halaman ini, yaitu melakukan pencairan dana (withdraw) atau menambahkan rekening. Jika memilih withdraw, sistem akan menampilkan modal konfirmasi pencairan dana. Apabila pengguna menyetujui konfirmasi tersebut, maka sistem akan mengirimkan data log transaksi ke database Undangan Amplop dengan metode POST.

Di sisi lain, pengguna juga dapat menambahkan rekening terlebih dahulu sebelum mencairkan dana. Proses ini diawali dengan mengisi form modal yang berisi informasi seperti nama pemilik dan nomor rekening. Setelah data tersebut diisi, sistem akan menyimpan informasi rekening ke dalam database menggunakan metode POST.

Fitur ini memberikan kemudahan bagi pengguna dalam mengelola amplop digital yang diterima, memastikan bahwa proses pencairan dana dan penyimpanan data rekening dilakukan secara aman dan terstruktur.

U I M 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 3.7. Flowchart RSVP Attendance

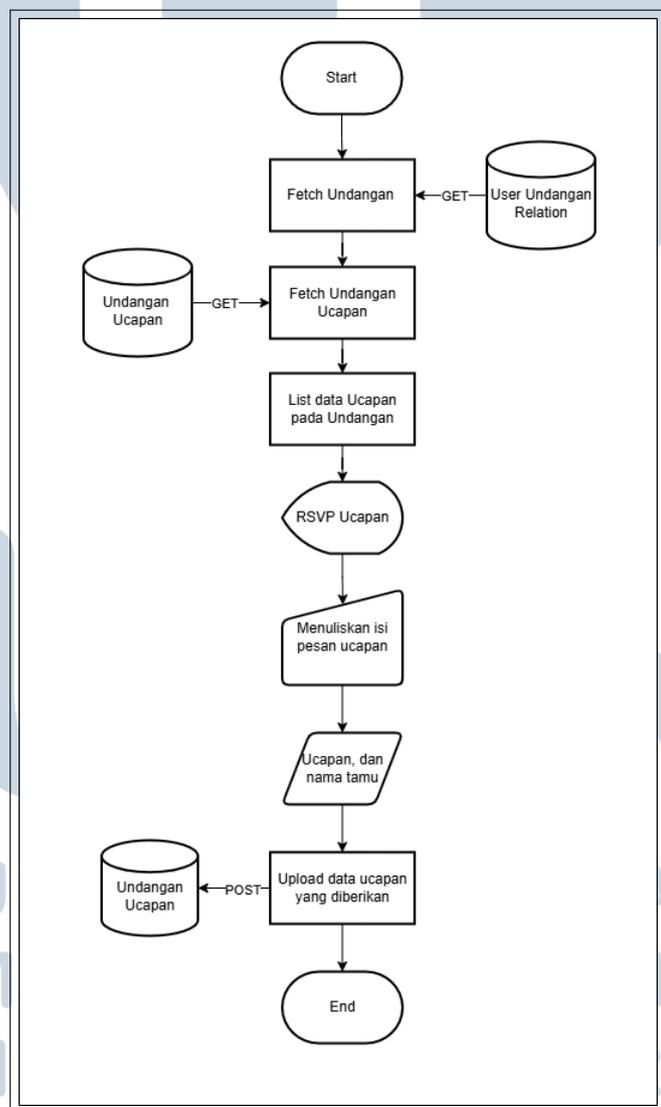
Gambar 3.7 menunjukkan *flowchart* dari fitur *RSVP Attendance* yang digunakan untuk mencatat kehadiran tamu dalam sistem *Website Kartu Digital*. Fitur ini dirancang untuk membantu pemilik undangan dalam memantau dan mencatat siapa saja yang akan menghadiri acara. Sistem memulai proses ketika data undangan diambil berdasarkan akun pengguna yang sedang login. Pengambilan data ini dilakukan dengan menggunakan relasi dari tabel *User Undangan Relation* untuk memastikan hanya data undangan milik pengguna tersebut yang diakses.

Setelah data undangan berhasil diperoleh, sistem melanjutkan dengan mengambil daftar tamu undangan dari tabel *Undangan Tamu* menggunakan metode *GET*. Data ini mencakup informasi tamu yang sebelumnya telah dimasukkan oleh pengguna saat membuat undangan. Selanjutnya, tamu diarahkan ke halaman *RSVP* yang menampilkan form konfirmasi kehadiran. Form ini berisi pilihan bagi tamu untuk menyatakan apakah akan menghadiri acara atau tidak.

Pilihan yang diberikan tamu menjadi dasar bagi sistem untuk memperbarui status kehadiran dalam basis data. Jika tamu memilih akan datang, maka sistem akan menjalankan proses *Update Tamu Datang* dan melakukan pembaruan data menggunakan metode *PATCH* pada tabel *Undangan Tamu*. Sebaliknya, jika tamu memilih tidak akan hadir, maka sistem akan menjalankan proses *Update Tamu*

Tidak Datang menggunakan metode PATCH yang sama, namun dengan status berbeda. Setiap perubahan status tersimpan secara langsung di basis data untuk keperluan pemantauan oleh pemilik undangan.

Fitur *RSVP Attendance* ini memegang peran penting dalam sistem karena memberikan informasi real-time kepada pemilik acara. Melalui fitur ini, pemilik acara dapat mempersiapkan jumlah konsumsi, tempat duduk, dan logistik lainnya dengan lebih tepat. Selain itu, sistem ini juga memudahkan tamu dalam memberikan konfirmasi secara cepat tanpa perlu komunikasi langsung. Dengan demikian, proses perencanaan acara menjadi lebih efisien dan terorganisir.



Gambar 3.8. Flowchart RSVP Wishes

Gambar 3.8 memperlihatkan *flowchart* dari fitur *RSVP Ucapan* pada sistem



Entity Relationship Diagram (ERD) pada sistem Website Kartu Digital menggambarkan struktur dan hubungan antar entitas yang digunakan dalam pengelolaan undangan digital. Berdasarkan fitur yang dikembangkan selama pelaksanaan magang, terdapat sejumlah entitas penting yang berperan langsung dalam proses registrasi pengguna, pengelolaan profil, RSVP kehadiran dan ucapan, pengelolaan tamu, serta fitur amplop digital. ERD ini menjadi dasar penting dalam memahami alur data yang mendukung sistem secara keseluruhan.

**1. Tabel users dan auths**

Fitur registrasi dan login menggunakan dua tabel utama, yaitu users dan auths. Tabel users menyimpan data profil pengguna seperti *first\_name*, *last\_name*, *phone\_number*, dan *birthdate*. Tabel auths menyimpan email dan password yang digunakan untuk proses autentikasi. Kedua tabel ini saling terhubung melalui *user\_id*, dan digunakan untuk menampilkan atau memperbarui data profil pengguna saat login.

**2. Tabel user\_undangan\_relation**

Tabel ini merepresentasikan undangan digital yang dimiliki oleh pengguna. Masing-masing pengguna dapat memiliki satu atau lebih undangan yang terhubung melalui *user\_id*. Kolom *event\_name*, *plan\_id*, dan *expired\_at* menyimpan informasi penting terkait detail acara. Tabel ini menjadi pusat relasi dari fitur lain seperti daftar tamu (*undangan\_tamu*), ucapan (*undangan\_ucapan*), dan amplop digital (*undangan\_amplop*).

**3. Tabel undangan\_tamu**

Tabel ini berfungsi menyimpan data tamu undangan dalam bentuk *JSON*, seperti nama tamu, alamat, nomor kontak, dan status kehadiran. Data tamu ini digunakan dalam dua fitur sekaligus: *Kelola Tamu* (untuk menambah/menghapus tamu) dan *RSVP Attendance* (untuk mengonfirmasi kehadiran tamu ke acara). Status kehadiran akan diperbarui di tabel ini berdasarkan input dari tamu.

**4. Tabel undangan\_ucapan**

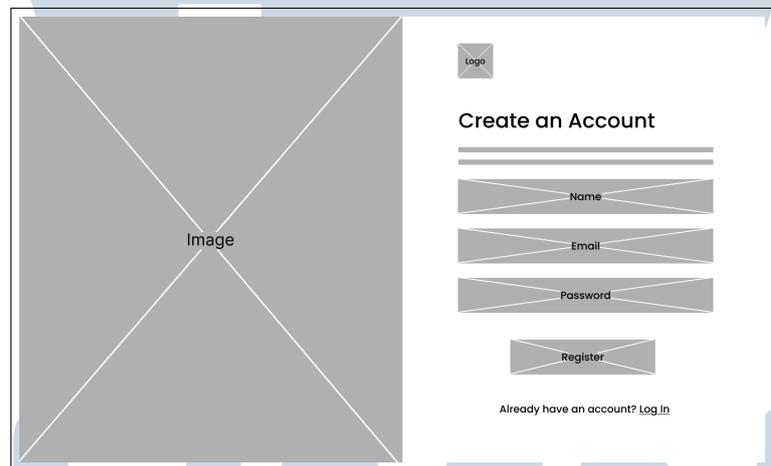
Tabel ini menyimpan pesan atau ucapan yang diberikan oleh tamu kepada pemilik acara. Data yang disimpan mencakup nama pengirim, isi pesan, dan status publikasi (*is\_publish*) untuk menentukan apakah ucapan ditampilkan di halaman publik. Fitur ini ditampilkan dalam alur flowchart *RSVP Wishes*,

di mana tamu mengisi form ucapan dan sistem menyimpannya ke dalam tabel ini.

#### 5. Tabel undangan amplop

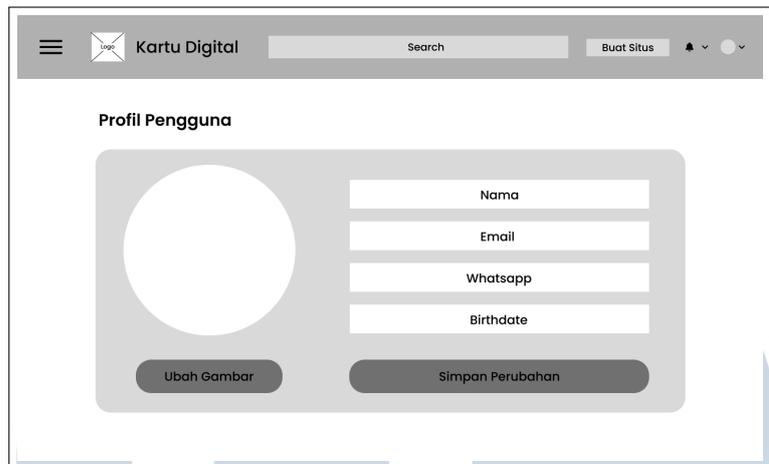
Tabel ini digunakan untuk mencatat transaksi berupa hadiah uang digital (amplop) yang dikirim oleh tamu. Data meliputi nama, pesan, jumlah nominal, serta waktu pengiriman. Fitur ini ditampilkan dalam sistem undangan sebagai bentuk apresiasi atau kontribusi dari tamu kepada pemilik acara.

#### 3.4.6 Wireframe



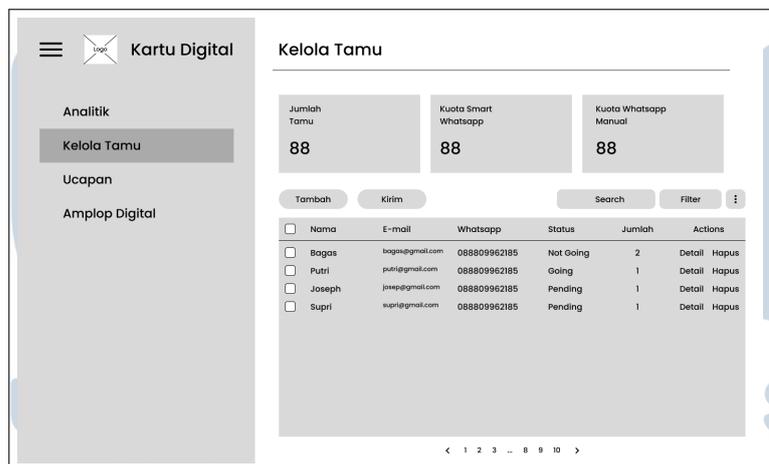
Gambar 3.10. Wireframe Register

Gambar 3.9 merupakan tampilan *wireframe* dari halaman *Register*. Halaman ini dirancang agar pengguna dapat melakukan pendaftaran akun baru dengan mudah. Komponen utama pada halaman ini adalah form input seperti nama lengkap, email, dan password yang disusun secara sederhana dan responsif. Desainnya menekankan pada keterbacaan dan kemudahan penggunaan agar proses registrasi tidak membingungkan. Tujuan dari tampilan ini adalah agar pengguna baru dapat mendaftar tanpa hambatan dan langsung diarahkan ke sistem.

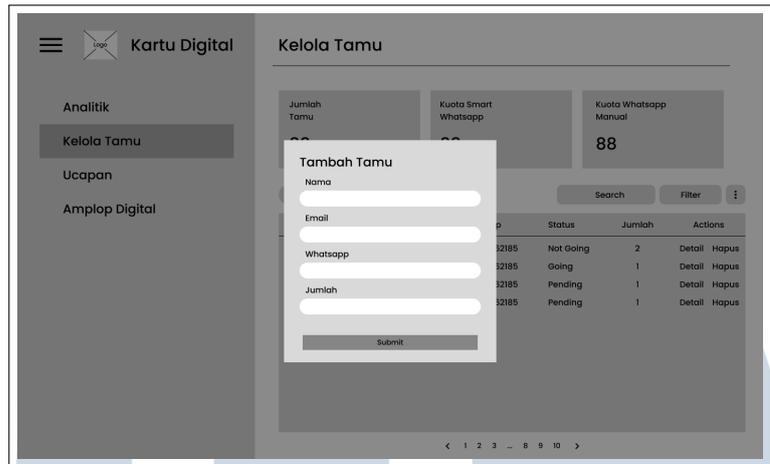


Gambar 3.11. Wireframe Profile

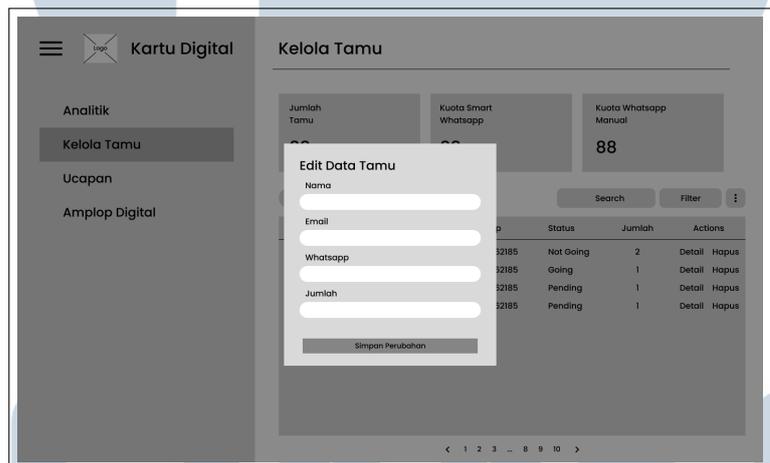
Gambar 3.10 merupakan *wireframe* dari halaman *Profile* pengguna. Pada halaman ini, pengguna dapat melihat dan mengubah data pribadi seperti nama, email, nomor telepon, dan foto profil. Desain disusun dengan antarmuka yang bersih dan terstruktur, sehingga pengguna dapat dengan mudah melakukan pembaruan data diri. Adanya fitur pengelolaan data pribadi ini penting dalam mendukung personalisasi informasi dan meningkatkan kenyamanan pengguna. Halaman ini juga menjadi pusat informasi akun bagi setiap user yang *login*.



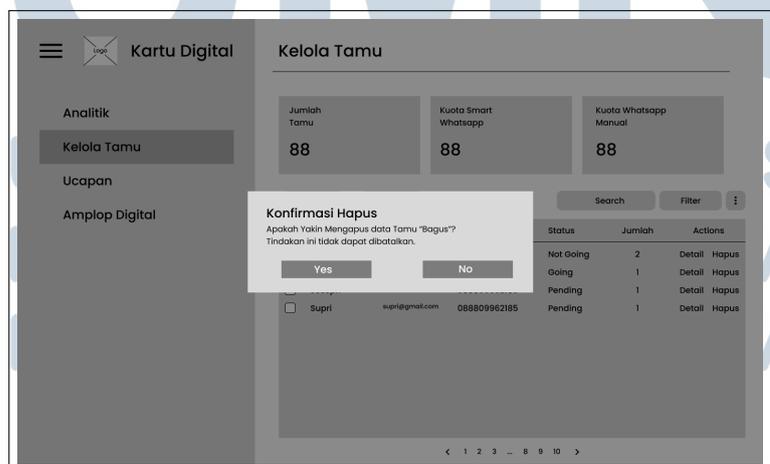
Gambar 3.12. Wireframe Kelola Tamu



Gambar 3.13. Wireframe Kelola Tamu (Tambah)

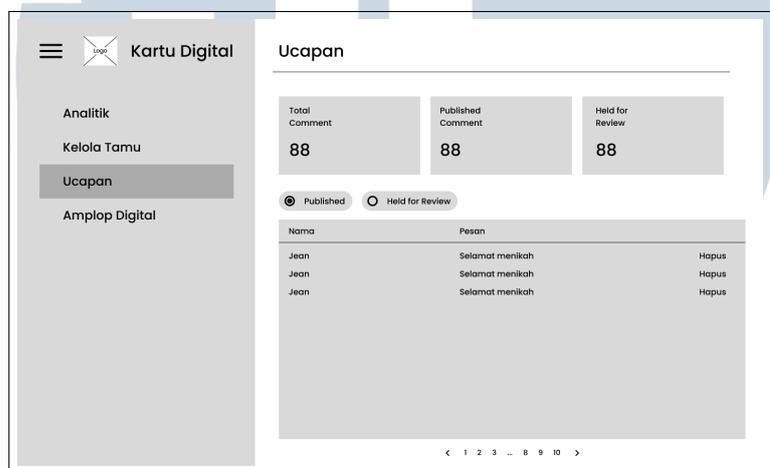


Gambar 3.14. Wireframe Kelola Tamu (Edit)

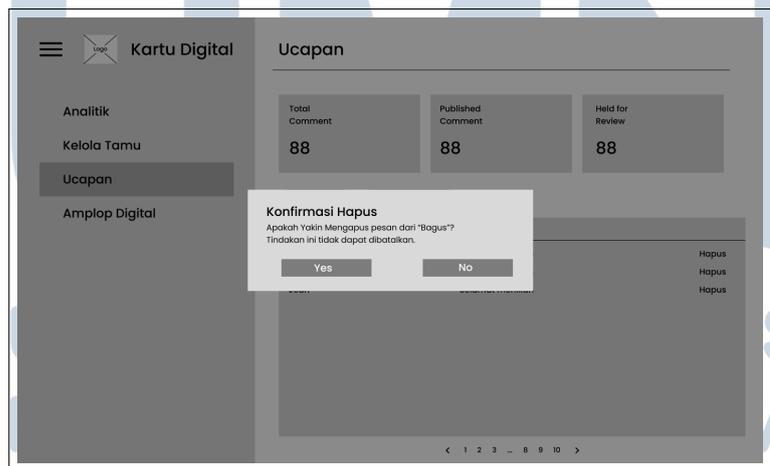


Gambar 3.15. Wireframe Kelola Tamu (Hapus)

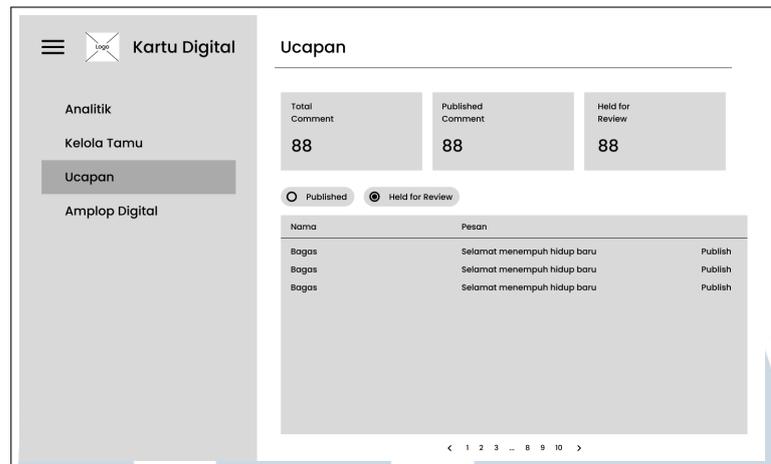
Gambar 3.11 hingga Gambar 3.14 merupakan *wireframe* dari fitur Kelola Tamu dalam halaman *Smart-Dashboard*. Tampilan pertama menunjukkan daftar tamu yang telah ditambahkan ke dalam undangan, dilengkapi dengan informasi dasar seperti nama dan kontak. Selanjutnya, terdapat halaman untuk menambahkan tamu baru melalui form input, serta halaman untuk mengedit data tamu yang sudah ada. Proses penghapusan tamu juga disertai dengan konfirmasi terlebih dahulu untuk menghindari kesalahan penghapusan. Seluruh tampilan ini dirancang untuk memberikan fleksibilitas dan kontrol penuh dalam pengelolaan data tamu undangan.



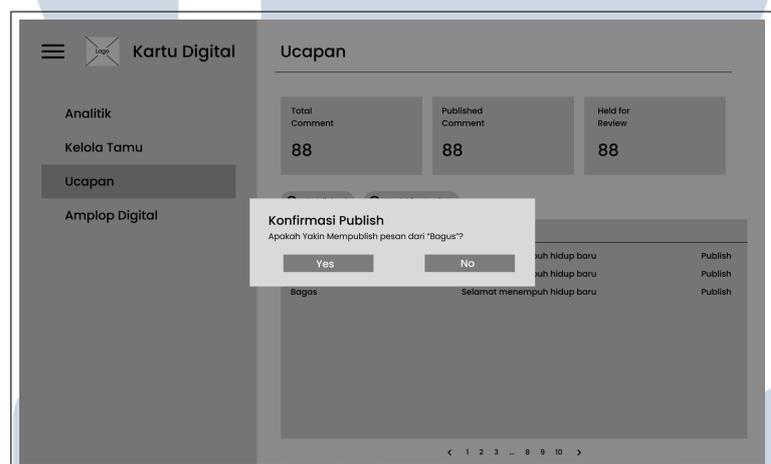
Gambar 3.16. Wireframe Ucapan



Gambar 3.17. Wireframe Ucapan (Hapus)

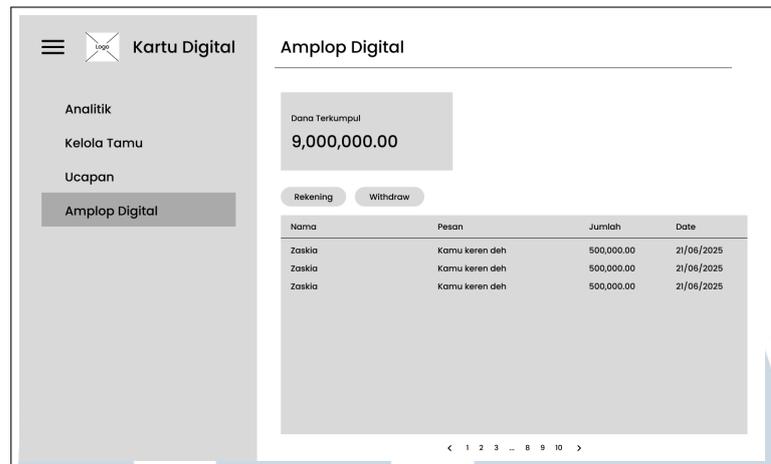


Gambar 3.18. Wireframe Ucapan (*Held for Review*)



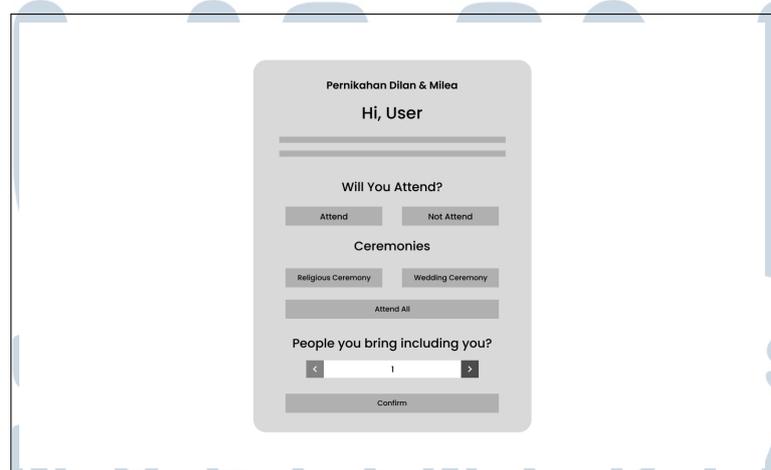
Gambar 3.19. Wireframe Ucapan (Publish)

Gambar 3.15 hingga Gambar 3.18 merupakan *wireframe* dari fitur Ucapan pada *Smart-Dashboard*. Halaman ini memungkinkan pengguna melihat daftar pesan atau ucapan yang telah dikirim oleh para tamu. Selain itu, terdapat tab untuk menampilkan pesan yang sedang menunggu persetujuan sebelum dipublikasikan. Pengguna diberikan opsi untuk menghapus atau mem-publish ucapan sesuai kebijakan penyelenggara. Dengan fitur ini, penyelenggara memiliki kendali penuh atas konten yang akan ditampilkan secara publik di undangan digital.



Gambar 3.20. Wireframe Amplop Digital

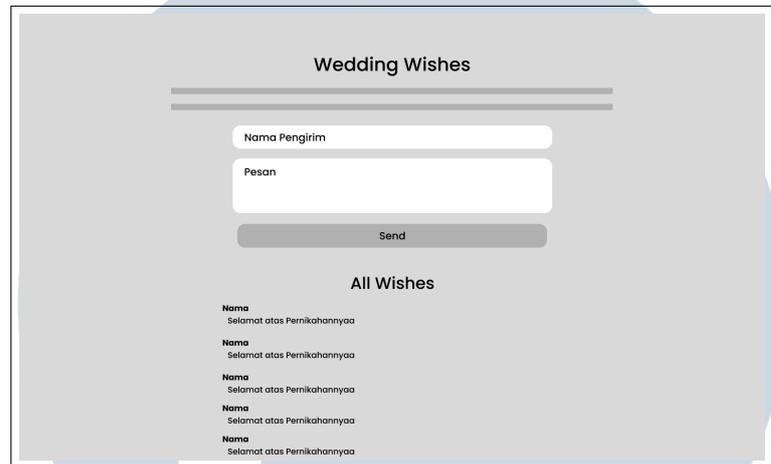
Gambar 3.19 merupakan *wireframe* dari halaman Amplop Digital pada *Smart-Dashboard* yang menampilkan riwayat transaksi yang dikirimkan tamu sebagai bentuk hadiah digital. Informasi pada halaman ini mencakup nama pengirim, nominal yang diberikan, serta metode transfer yang digunakan. Selain itu, pengguna juga dapat menambahkan informasi rekening pribadi untuk menerima dana. Tampilan dirancang agar mudah dipahami dan memberikan transparansi terhadap arus masuk dana dari para tamu. Fitur ini membantu pemilik acara dalam memantau amplop digital secara rapi dan terstruktur.



Gambar 3.21. Wireframe RSVP Attendance

Gambar 3.20 merupakan *wireframe* dari halaman *RSVP Attendance*. Di halaman ini, tamu dapat memilih kehadiran pada masing-masing sesi acara, seperti akad, resepsi, atau keduanya. Selain itu, tamu juga dapat mencantumkan jumlah

orang yang hadir bersama mereka. Desain form disusun secara sederhana agar mudah diisi dan tetap informatif. Data dari halaman ini digunakan oleh pemilik undangan untuk mempersiapkan jumlah tamu secara akurat.



The wireframe shows a form titled "Wedding Wishes" with a header line. Below the header are two input fields: "Nama Pengirim" and "Pesan". A "Send" button is positioned below the "Pesan" field. Underneath the form is a section titled "All Wishes" which contains a list of messages. Each message in the list starts with "Nama" followed by "Selamat atas pernikahannya".

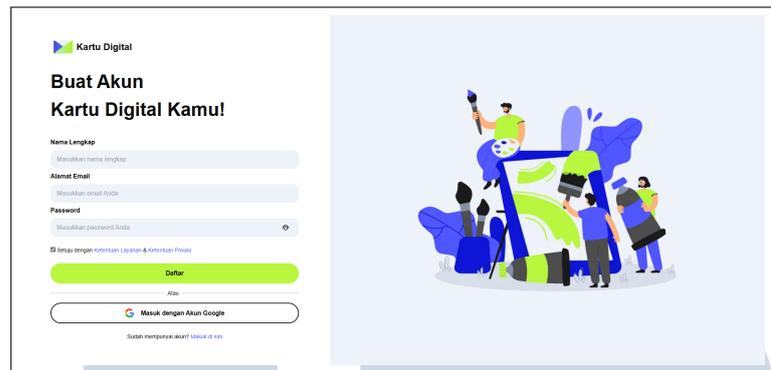
Gambar 3.22. Wireframe RSVP Wishes

Gambar 3.21 merupakan *wireframe* dari halaman *RSVP Wishes*, yaitu halaman bagi tamu yang ingin memberikan ucapan atau pesan secara digital. Tamu diminta mengisi nama dan isi pesan yang ingin disampaikan kepada pemilik acara. Setelah pesan dikirim, sistem akan memproses dan menampilkannya pada halaman utama setelah melalui persetujuan. Tampilan ini dirancang untuk memberikan kesan personal dan membangun interaksi antara tamu dan penyelenggara. Fitur ini juga mendukung pengalaman undangan digital yang lebih hangat dan bermakna.

### 3.5 Implementasi

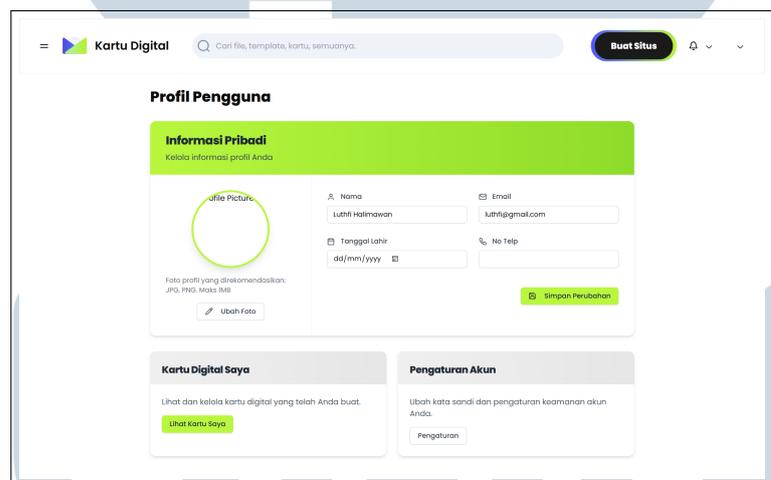
Pengembangan Kartu Digital telah diimplementasikan dan sudah masuk tahap production pada situs kartudigital.com. Berikut ini adalah hasil akhir Kartu Digital yang sudah saya kerjakan:

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



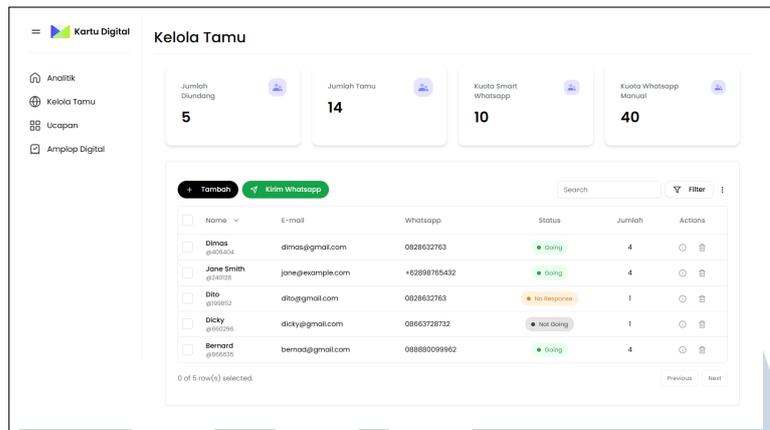
Gambar 3.23. Tampilan Register

Implementasi dari halaman *Register* yang terdapat di gambar 3.22. *User* akan diminta untuk mengisi nama lengkap, email, dan password terlebih dahulu, Setelah itu baru *register*, apabila sudah ada datanya dalam *database* maka user harus membuat yang baru dan berbeda.

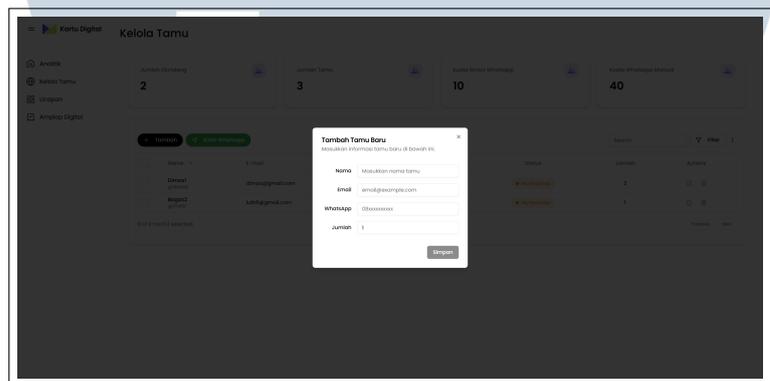


Gambar 3.24. Tampilan Profile

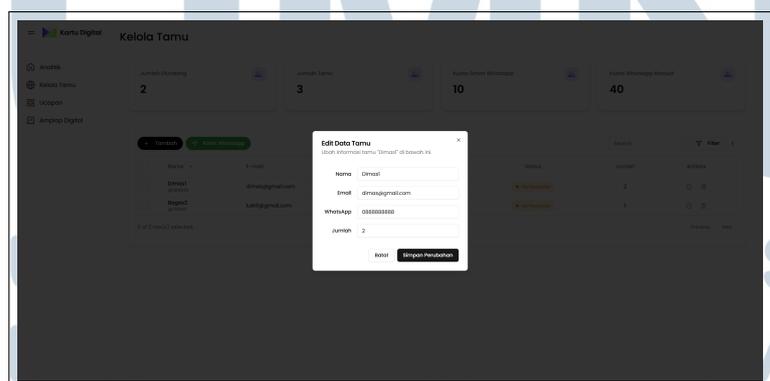
Selanjutnya adalah hasil implementasi dari *Profile* yang dapat dilihat di gambar 3.23. Data dari user yang login akan ditampilkan di halaman *profile*. *User* dapat mengubah data seperti foto profil, nama, email, tanggal lahir, dan no. telp.



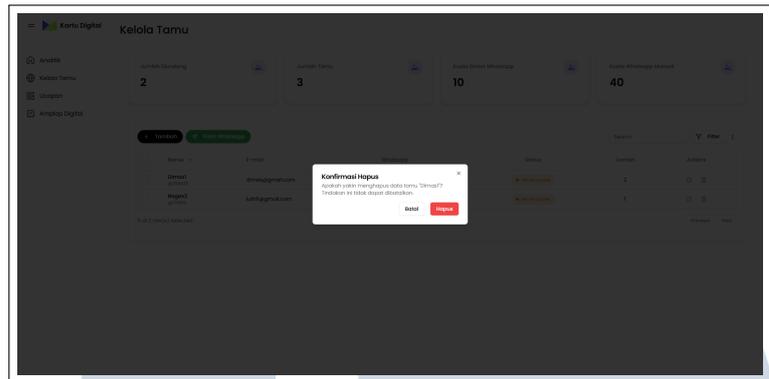
Gambar 3.25. Tampilan Kelola Tamu



Gambar 3.26. Tampilan Kelola Tamu (Tambah)

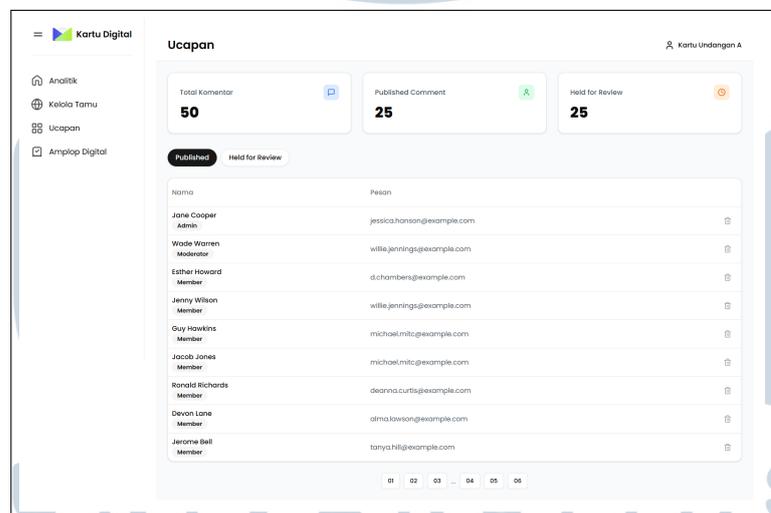


Gambar 3.27. Tampilan Kelola Tamu (Edit)

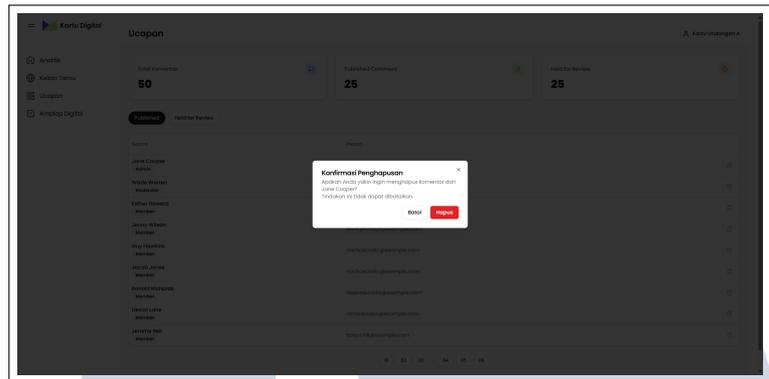


Gambar 3.28. Tampilan Kelola Tamu (Hapus)

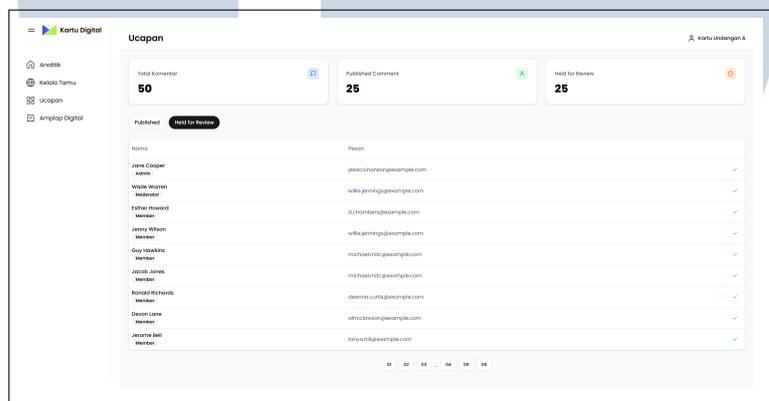
Gambar 3.24 merupakan implementasi fitur kelola tamu di dalam halaman *Smart-Dashboard*. *User* yang memiliki undangannya dapat mengatur semua hal mengenai tamu di kelola tamu ini. Ada fitur untuk menambahkan data tamu baru, seperti di gambar 3.25 dengan mengisi lengkap data nama, email, whatsapp, dan jumlah tamu yang kemungkinan akan dibawa. Selanjutnya ada edit data tamu pada gambar 3.26 untuk melakukan perubahan data terhadap data tamu yang sudah ada. Dan hapus data tamu yang ada, dapat dilihat di gambar 3.27.



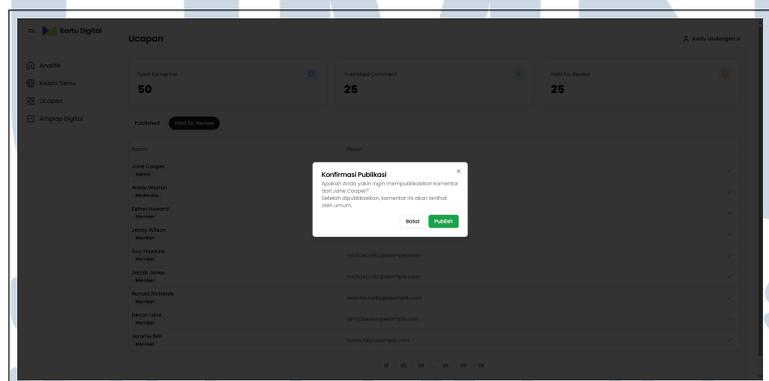
Gambar 3.29. Tampilan Ucapan



Gambar 3.30. Tampilan Ucapan (Hapus)



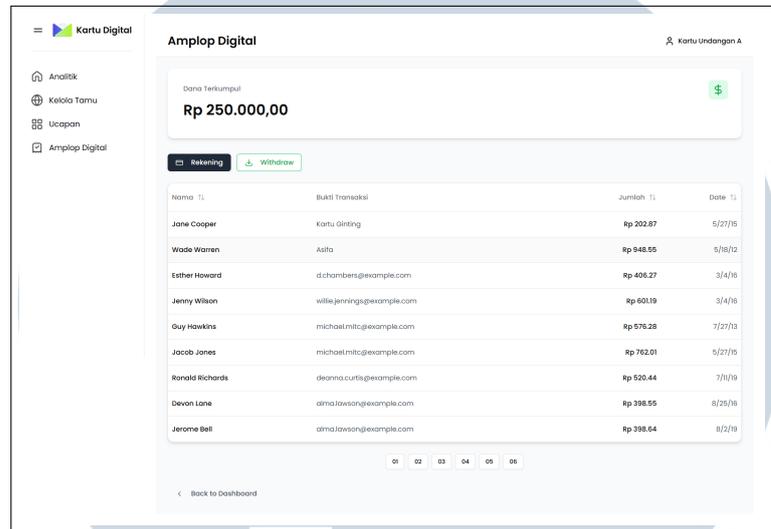
Gambar 3.31. Tampilan Ucapan (Held for Review)



Gambar 3.32. Tampilan Ucapan (Publish)

Implementasi dari halaman fitur Ucapan pada *Smart-Dashboard* terdapat pada gambar 3.28 sampai 3.31. *User* yang memiliki undangan dapat mengelola ucapan-ucapan yang dikirimkan tamu melalui RSVP Wishes. *User* dapat menghapus data (gambar 3.29) pesan dari tamu di tab Published (gambar 3.28) yang

berisi list daftar pesan yang sudah publish. Untuk data baru masuk akan berada di held for review (gambar 3.30), dan untuk menjadikannya publish perlu adanya persetujuan dari user dengan mengklik publish dan menyetujuinya (gambar 3.31).



Gambar 3.33. Tampilan Amplop Digital

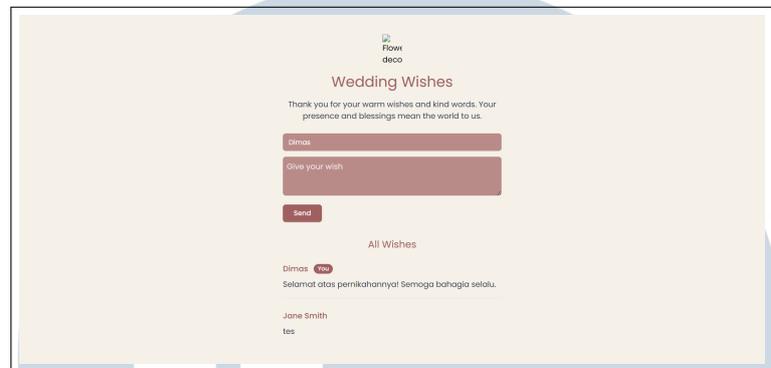
Gambar 3.32 merupakan hasil implementasi untuk halaman fitur Amplop Digital pada *Smart-Dashboard*. Halaman ini akan menunjukkan transaksi uang masuk dari tamu sebagai hadiah. Disini user dapat menambahkan rekeningnya dan langsung bisa menarik/withdraw uang ke rekening user secara langsung.



Gambar 3.34. Tampilan RSVP Attendance

Gambar 3.33 merupakan hasil implementasi dari RSVP Attendance. Halaman ini diperuntukkan untuk tamu-tamu dari undangan. Di bagian ini tamu akan mengonfirmasi kedatangan dan juga perayaan nikah agama, resepsi, atau

keduanya yang akan di datangi. Hal lain yang perlu diisi adalah tamu akan membawa berapa orang termasuk dirinya.



Gambar 3.35. Tampilan RSVP Wishes

Gambar 3.34 merupakan hasil implementasi dari *RSVP Wishes*. Bagian ini adalah lanjutan dari *RSVP Attendance*, di *RSVP Wishes* tamu bisa memberikan pesan kepada *user*. Di bagian bawah akan muncul list-list ucapan dari tamu lainnya.

### 3.6 Kendala dan Solusi yang Ditemukan

Selama proses pengembangan sistem *Website Kartu Digital*, ditemukan beberapa kendala yang berdampak terhadap efektivitas kerja tim dan keterlambatan dalam pengembangan sistem. Kendala-kendala tersebut dijelaskan sebagai berikut:

- **Tidak adanya dokumentasi progres dari tim pengembang sebelumnya**  
Proyek ini merupakan hasil serah terima (*handover*) dari tim pengembang sebelumnya, namun tidak disertai dengan dokumentasi progres pengembangan yang memadai. Informasi mengenai fitur-fitur yang telah diselesaikan, fitur yang masih dalam proses, maupun fitur yang belum dikerjakan tidak tersedia secara jelas. Akibatnya, diperlukan waktu tambahan untuk menelusuri kode secara manual dan melakukan verifikasi ulang terhadap status masing-masing bagian sistem. Kondisi ini menyebabkan kebingungan dalam menentukan prioritas tugas dan berpotensi menimbulkan duplikasi pekerjaan. Selain itu, proses integrasi lanjutan menjadi kurang efisien karena tidak adanya acuan yang terstandarisasi mengenai progres sebelumnya.
- **Minimnya riset awal terhadap kebutuhan fitur editor**  
Fitur editor yang dikembangkan bertujuan untuk memberikan keleluasaan

kepada pengguna dalam menyesuaikan tampilan undangan digital. Namun, riset awal mengenai pendekatan teknis serta kebutuhan fleksibilitas pengguna belum dilakukan secara mendalam. Alhasil, editor yang dihasilkan masih bersifat terbatas dan belum mampu menyediakan kebebasan penuh seperti mengatur posisi, ukuran elemen, maupun menambahkan komponen visual secara bebas, sebagaimana platform desain populer seperti Canva.

- **Kurangnya pemahaman tim terhadap alur sistem dan basis data**

Beberapa anggota tim pengembang belum sepenuhnya memahami alur kerja sistem serta struktur basis data yang digunakan. Hal ini berdampak pada lambatnya proses integrasi dan pengembangan fitur, serta menimbulkan kendala dalam komunikasi teknis antar anggota tim. Dalam beberapa kasus, pekerjaan harus direvisi atau diulang karena terjadinya kesalahpahaman teknis dalam implementasi.

Solusi yang diterapkan untuk mengatasi berbagai kendala selama proses pengembangan sistem *Website Kartu Digital* dijelaskan sebagai berikut:

- **Solusi terhadap tidak adanya dokumentasi progres dari tim sebelumnya**

Untuk mengatasi ketiadaan dokumentasi progres, dilakukan penelusuran kode secara menyeluruh guna memetakan fitur-fitur yang telah selesai, masih dalam proses, atau belum dikerjakan. Hasil penelusuran tersebut kemudian dirangkum dalam bentuk catatan teknis dan daftar progres internal agar dapat dijadikan acuan selama proses pengembangan berlanjut. Selain itu, dokumentasi tambahan juga disusun secara paralel selama pengembangan agar tim memiliki referensi yang lebih terstruktur ke depannya. Dengan langkah ini, pekerjaan dapat disusun lebih terarah, dan duplikasi tugas dapat dihindari.

- **Solusi terhadap minimnya riset awal fitur editor**

Menyadari terbatasnya hasil pada fitur editor, dilakukan evaluasi ulang terhadap kebutuhan pengguna dan eksplorasi terhadap pendekatan teknis yang lebih fleksibel. Referensi dari platform serupa seperti Canva atau PowerPoint digunakan untuk memahami cara pengaturan elemen secara dinamis. Selain itu, fitur prototipe dikembangkan secara bertahap untuk menguji kemungkinan pengaturan bebas terhadap posisi, ukuran, dan penambahan elemen visual. Langkah ini diharapkan dapat menjadi dasar

dalam pengembangan editor yang lebih interaktif dan fleksibel pada iterasi berikutnya.

- **Solusi terhadap kurangnya pemahaman alur sistem dan basis data**

Untuk mengurangi hambatan pemahaman sistem, dilakukan penggalian informasi secara mandiri terhadap struktur database dan alur kerja sistem melalui pembacaan kode dan percobaan langsung. Diagram ERD dan alur data juga disusun ulang agar dapat memperjelas hubungan antar entitas dan memudahkan integrasi fitur. Komunikasi rutin dengan anggota tim yang lebih memahami sistem juga dilakukan untuk klarifikasi teknis. Dengan cara ini, kesalahan implementasi dapat diminimalkan dan proses pengembangan dapat berjalan lebih efektif.

