BAB 3
PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Organisasi

Dalam rangkaian program magang, penugasan dilaksanakan pada unit proyek Invoices dengan peran sebagai Front End Developer. Sepanjang periode magang, pengawasan dan bimbingan dilakukan oleh Bapak Iwan Hasim selaku Kepala IT. Tugas pokok selama magang ini meliputi perancangan serta pembangunan situs Mixternal yang dilengkapi dengan fitur-fitur manajemen inventori dan faktur sesuai keperluan perusahaan. Interaksi terkait pengembangan situs web dilakukan melalui komunikasi langsung di kantor atau melalui Pesan WhatsApp untuk pertanyaan yang memerlukan klarifikasi dari Manajemen Proyek. Kemajuan pengembangan website Mixternal dipaparkan setidaknya seminggu sekali dalam sesi presentasi bersama manajemen proyek dan pembimbing lapangan.

3.2 Tugas yang Dilakukan

Selama menjalani masa magang di PT Mobile Data Indonesia, tugas yang diberikan melibatkan perancangan dan pembangunan website Mixternal pada PT Mobile Data Indonesia. Tugas yang dikerjakan adalah sebagai berikut.

1. Membangun front-end yang responsif.
2. Membuat mockup untuk pembuatan UI UX dan front endnya
3. Mengelola operasi CRUD (Create, Read, Update, Delete).

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Melakukan perkenalan divisi dan mempelajari proses bisnis.
2	Diskusi terkait pembuatan *website* inventaris serta mengumpulkan daftar kebutuhan apa saja yang nanti akan dibutuhkandi dalam *website*
3	Melakukan diskusi dengan tim IT perusahaan terkait kebutuhan *website* dengan kesiapan infrastruktur yang ada.
4	Melanjutkan pembuatan *flowchart website* inventaris.
5	Melanjutkan pembuatan *mockup website* rekrutmen berdasarkan hasil diskusi bersama.
6	Melakukan revisi mockup berdasarkan hasil *review user*.
7	Membuat UI untuk di implementasikan ke website
8	Menyelesaikan pembuatan UI UX dan mulai mengerjakan frontend.
9	Melanjutkan pembuatan frontend bagian landing page dan peminjaman barang.
10	Melanjutkan pembuatan frontedn bagian landing page dan peminjaman barang.
11	Mendiskusikan perubahan project dikarenakan ada *urgensi* kebutuhan dari kantor, dibagian divisi finance.
12	Mulai merancang flowchart dan mockup.
13	Memulai pembuatan UI UX untuk di develop kedalam website
14	Memulai pembuatan frontend di bagian login page dan landing page.
15	Melanjutkan pembuatan frontend di bagian landing page yaitu page invoice dan pembuatan page payments.
16	Melanjutkan pembuatan frontend di bagian page *clients* dan page *reports*.
17	Penyerahan hasil tampilan website ke divisi finance dan mendemokannya.
18	Melakukan integrasi dengan backend.
19	Melakukan integrasi dengan backend.
Lanjut pada halaman berikutnya	

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

Minggu Ke -	Pekerjaan yang dilakukan
20	Melakukan integrasi dengan backend.
21	Melaukan testing dan sudah di *deploy*

3.3 Uraian Pelaksanaan Magang

Pelaksanaan magang dilakukan dengan mengerjakan tugas-tugas yang diberikan oleh mentor. Selama kurang lebih 6 bulan menjalani magang, peserta diberikan beberapa tugas utama untuk menunjang pengembangan ulang *website* invoice perusahaan yang terbagi menjadi dua bagian, yaitu pembuatan *website* inventaris yang dikerjakan kurang dari setengah jalan, lalu di pindahkan fokusnya ke invoice, karena kebutuhan kantor yang mendesak.

3.3.1 Invoicesing

Invoice merupakan dokumen penting dalam perdagangan yang berfungsi sebagai bukti penjualan produk atau jasa. Dalam konteks bisnis, invoice memiliki fungsi penting sebagai catatan transaksi, arsip, dan dasar perhitungan pajak. Agar dapat efektif dalam pengelolaan keuangan perusahaan, pembuatan invoice harus mengikuti prosedur yang sistematis1. Selain itu, invoice juga berkontribusi dalam memastikan transparansi antara pemasok dan pelanggan, yang membantu mengurangi risiko perselisihan pembayaran. [6]

UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.4 Perancangan Sistem Website Internal

Sistem web Mixternal dirancang dengan dua komponen utama: front-end, yang berfungsi sebagai antarmuka pengguna, dan back-end, yang mengurus pengolahan serta penyimpanan data situs. Pemisahan ini memungkinkan pengembangan sistem yang lebih terorganisir, di mana front-end mengedepankan aspek tampilan dan pengalaman pengguna, sementara back-end menangani pengolahan serta penyimpanan data. Pendekatan ini meningkatkan kemudahan akses ke situs web Mixternal PT Mobile Data Indonesia.

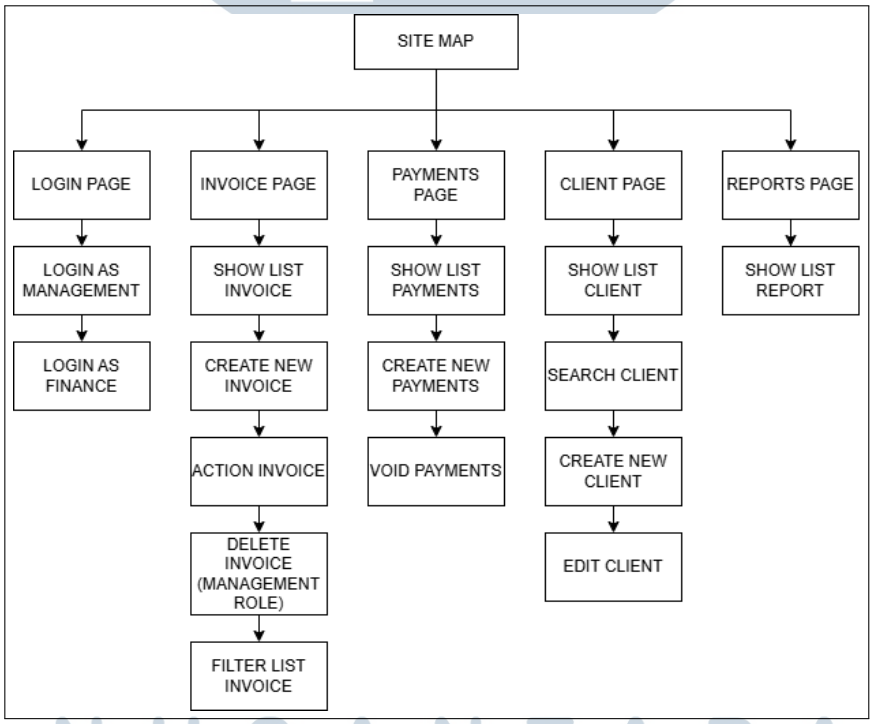
3.4.1 Flowchart

Pembuatan flowchart dalam proyek ini dilakukan menggunakan *draw*.io , sebuah alat visual berbasis web yang mendukung perancangan diagram alur secara fleksibel dan presisi. Tool ini dipilih karena kemampuannya dalam menyusun struktur navigasi sistem secara modular, serta kemudahan dalam menyesuaikan elemen diagram sesuai kebutuhan pengembangan antarmuka pengguna. Flowchart yang dihasilkan berfungsi sebagai panduan utama dalam membangun logika interaksi dan struktur halaman website secara sistematis. Flowchart yang ditampilkan ini memberikan gambaran visual tentang alur navigasi dan interaksi pengguna di situs web Mixternal, dengan fokus pada elemen pengembangan antarmuka pengguna frontend. Untuk membuat desain dan implementasi tampilan yang intuitif dan mudah digunakan, diagram ini memetakan struktur halaman dan konektivitas antara fitur utama. Diagram ini bermanfaat bagi tim frontend dalam memastikan setiap bagian dari antarmuka pengguna dirancang mengikuti jalur interaksi pengguna yang efektif dan masuk akal. Metode ini memungkinkan pengembangan frontend yang lebih terstruktur dan sesuai dengan keperluan pengguna akhir. [7]

UNIVERSITAS
MULTIMEDIA
NUSANTARA

1. Flowchart Site Map

Flowchart sitemap ini merupakan struktur utama website, terdapat dari lima modul utama yaitu ada login, invoice, create payment, client, dan report. Pengguna memuli dari login, terdapat dua role yaitu management dan finance. Role management memiliki fitur yang sama seperti role finance tetapi memiliki kelebihan yaitu dia dapat menghapus sebuah invoice. Setelah masuk mereka dapat mengelola invoice dengan fitur yang sudah dibuat seperti menampilkan daftar invoice, membuat invoice, mengedit invoice, mengdownload invoice dan menghapus invoice (khusus role management). Modul pembayaran merupakan fitur yang memungkinkan pengguna untuk memasukkan bukti pembayaran dari client dan memberikan tanda invoice mana yang dibayarkan oleh si client. Lalu ada modul client. Dalam modul klien, pengguna dapat membuat list klien, merubah data klien dan mencari data klien. Website ini dibuat dengan alur yang sudah tertata sesuai dengan keinginan finance, jadi memastikan pengguna dapat mengelola dengan efisien melalui navgigasi yang sudah dibuat.

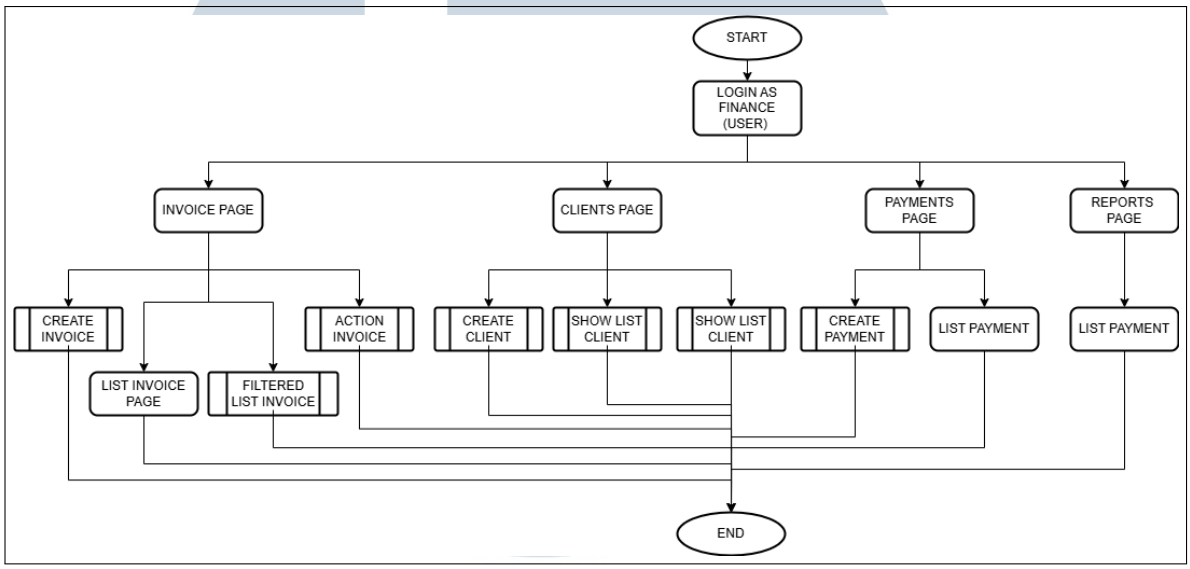


Gambar 3.1. Gambar Overall Flow Website

NUSANTARA

2. Flowchart overall website

3.2 menampilkan flowchart dari alur website keseluruhan. Di dalam website, terdapat lima modul utama yang dapat diakses pengguna: login, invoice, create payment, client, dan report. Setiap modul ini dilengkapi dengan sidebar untuk memudahkan pengguna yang ingin berpindah ke modul lain.

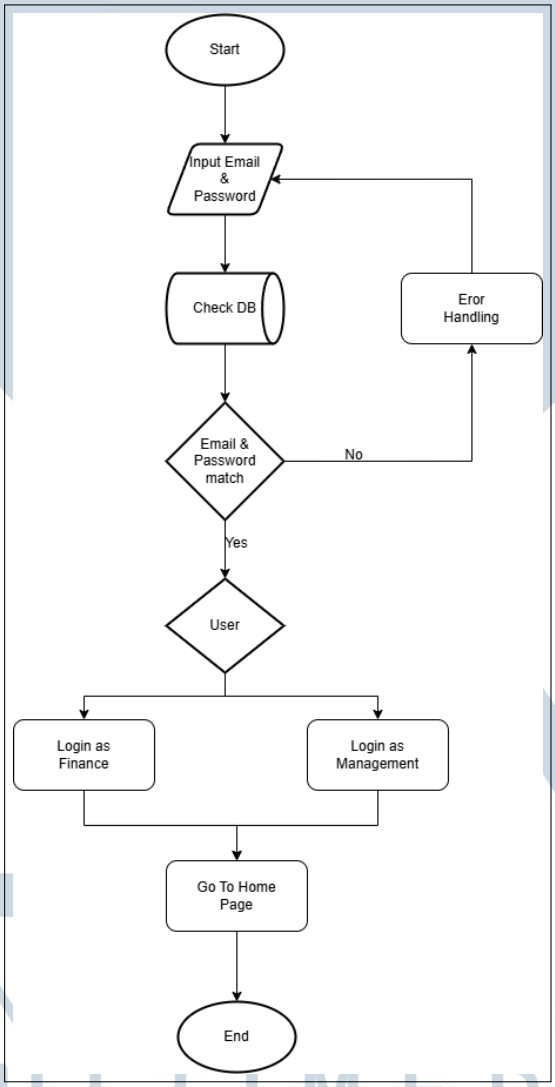


Gambar 3.2. Gambar Overall Flow Website

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3. Flowchart Login Page

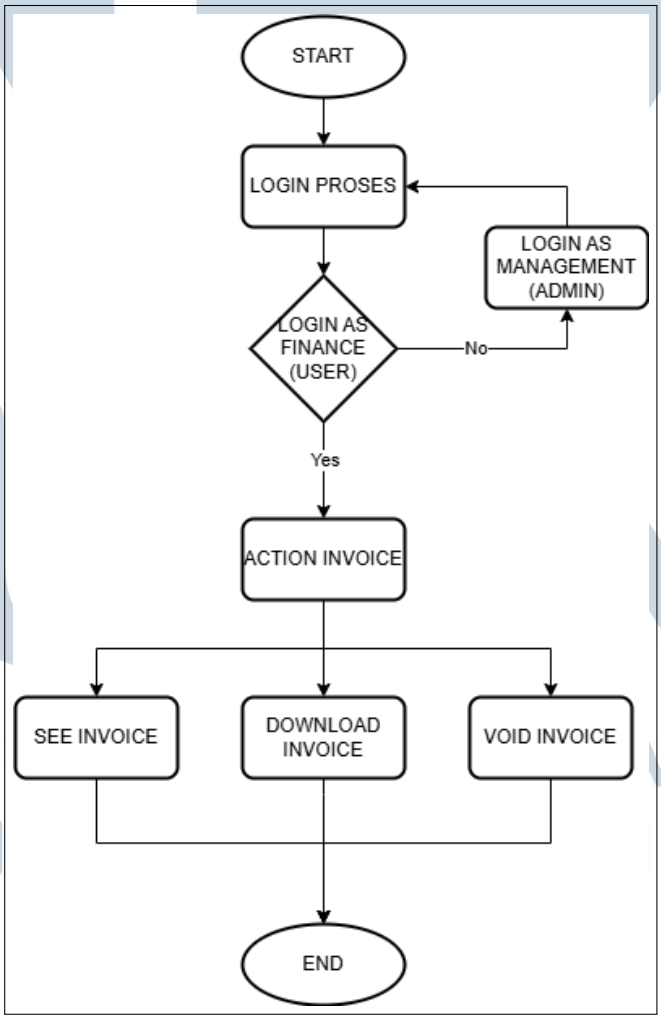
Modul login akan muncuk saat pertama kali membuka website. Pada halaman login, pengguna bisa mengakses sesuai dengan username dan password yang sudah dibuatkan oleh pihak kantor. Ketika pengguna salah menginput username atau pasword, maka akan tulisan gagal atau eror. Flowchart login dapat dilihat digambar 3.3



Gambar 3.3. Gambar Flowchart Login Page

4. Flowchart Invoice page

Proses flowchart modul invoice secara keseluruhan dapat dilihat pada Gambar 3.4 invoice memungkinkan pengguna untuk megelola sesuai dengan peran yang mereka miliki di dalam website. Setelah login, pengguna akan masuk ke modul utama invoice, di mana mereka dapat memilih tindakan sesuai dengan akses yang diberikan. pengguna dengan peran Finance dapat melihat, mengunduh, dan mengvoid invoice. Dan untuk peran management memiliki akses yang sama namun yang membedakan hanyalah opsi tambahan untuk menghapus invoice. Setiap aksi yang dilakukan di modul ini memastikan bahwa sang pengelola berjalan sesuai dengan prosedur yang ditetapkan.
Proses flowchart modul invoice dapat dilihat pada gambar 3.4



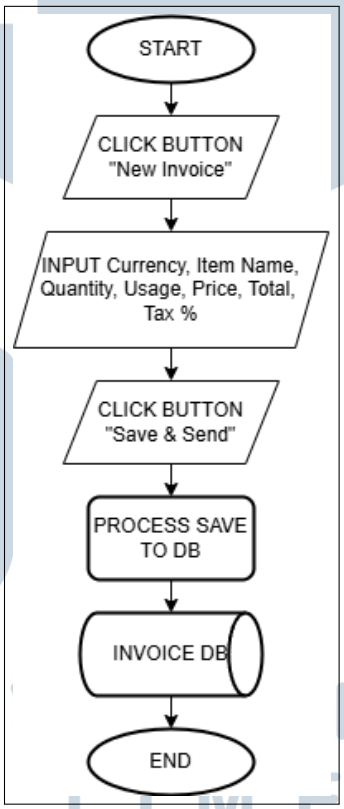
Gambar 3.4. Gambar Flowchart Invoice Page

5. Flowchart Create Invoice

Modul create invoice memungkinkan pengguna untuk membuat faktur baru sesuai dengan kebutuhan pengguna. Alur dimulai dari create invoice, fitur ini terdapat dari modul invoice.

Pengguna harus memasukkan data yang diperlukan didalam invoice form. Setelah semua informasi sudah diisi dengan benar, mereka dapat meyimpan data client dengan cara klik invoice, dimana data akan disimpan dalam database.

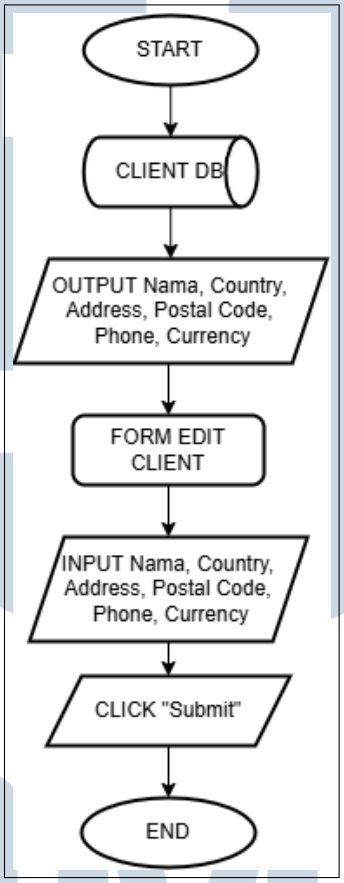
Jika proses berhasil semua, sistem akan mengarahkan pengguna ke end create invoice, menandakan bahwa semua prosedur sudah berhasil dilakukan dengan benar. Flochart ini dapat menggambarkan jelas mengenai langkah-langkah yang diperlukan untuk membuat faktur secara sistematis dan memastikan bahwa data yang dimasukkan ke dalam form sudah benar semua.



Gambar 3.5. Gambar Flowchart Create Invoice

6. Flowchart Fitur Filter List Invoice

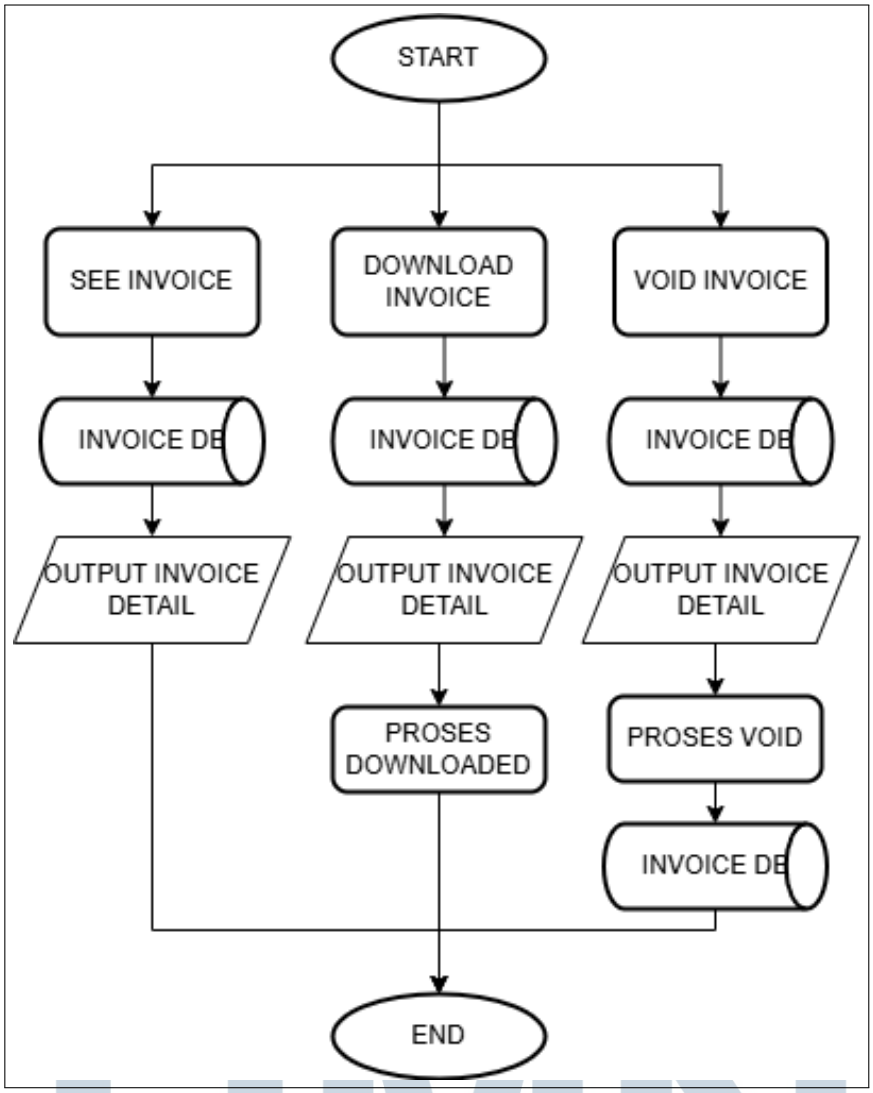
Pada gambar 3.6 menunjukkan alur atau *flowchart* yang menjelaskan fitur *filter* yang ada pada *website InvoiceApp*.



Gambar 3.6. Gambar Flowchart Create Fitur Filter List Invoice

7. Flowchart Detail Action Invoice

Pada gambar 3.7 menunjukkan alur atau *flowchart* yang menjelaskan fitur *action* yang ada pada *website InvoiceApp*. Fitur action ini terdiri dari *see invoice, download invoice* dan *void invoice*.



Gambar 3.7. Gambar Flowchart Detail Action Invoice

8. Flowchart Payment page

Modul Payment Page berfungsi sebagai pusat pengelolaan keuangan yang masuk. Modul ini memungkinkan pengguna untuk melihat, mengelola, dan memproses pembayaran.

Setelah pengguna mengakses modul Payment Page, mereka dapat melakukan beberapa tindakan yang sudah disediakan oleh sistem, termasuk:

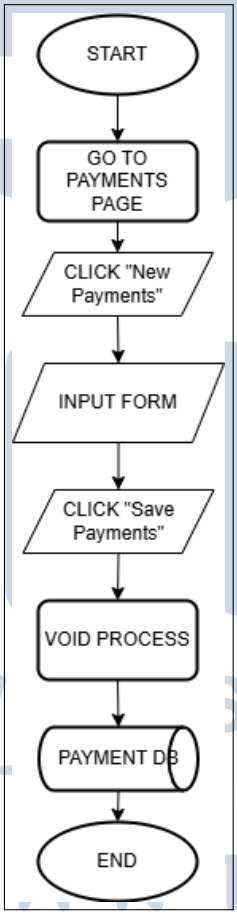
- View Payment Details : Pengguna dapat melihat detail pembayaran, termasuk juga status transaksi dan informasi terkait penerima dan pengirim dana.
- Make payment : Pengguna dapat mengisi form yang sudah diesediakan oleh

sistem, untuk mendakan client mana yang sudah membayar tagihan dari pihak kantor.

- Cancel Payment : Fitur ini digunakan untuk membatalkan pembayaran jika ditemukan keselahan.
- Generate Payment Report : Fitur ini bisa digunakan untuk melihat laporan pembayaran yang berisi riwayat transaksi serta analisi keuangan.

Setiap aksi yang dilakukan untuk pembaruan status pembayaran yang tercatat dalam sistem. Dengan demikian, Payment Page memastikan pengelolaan transaksi yang sistematis, dan sesuai dengan standar operasional yang berlaku dan untuk memudahkan finance untuk melakukan pencatatan uang masuk.

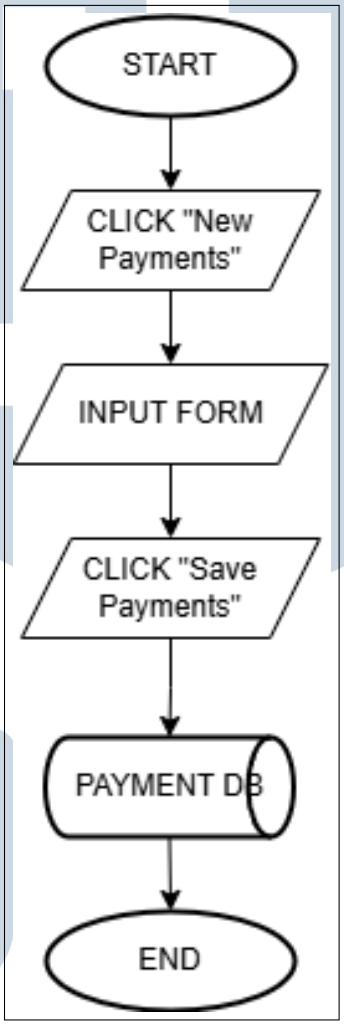
Proses flowchart modul invoice dapat dilihat pada gambar 3.8



Gambar 3.8. Gambar Flowchart Payments Page

9. Flowchart Create Payment

Pada gambar 3.9 menunjukkan alur atau *flowchart* yang menjelaskan fitur *create payment* yang ada pada *website InvoiceApp*. Fitur *create payment* berguna untuk menambahkan *list payment* ke dalam *database*.

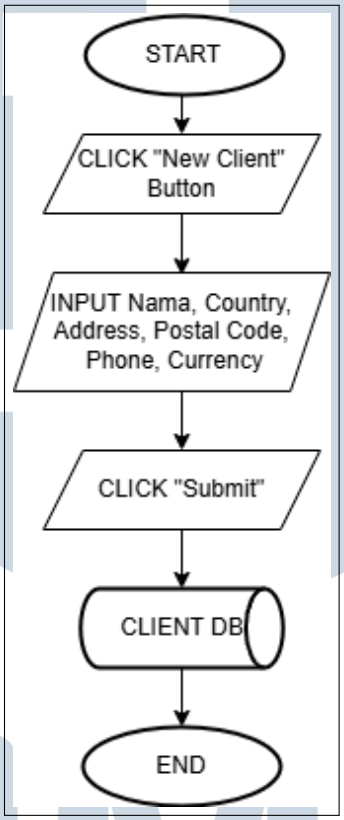


Gambar 3.9. Gambar Flowchart Create Payment

UNIVERSITAS MULTIMEDIA NUSANTARA

10. Flowchart Create Client

Pada gambar 3.10 menunjukkan alur atau *flowchart* yang menjelaskan fitur *create client* yang ada pada *website InvoiceApp*. Fitur *create client* berguna untuk menambahkan *list client* ke dalam *database*.

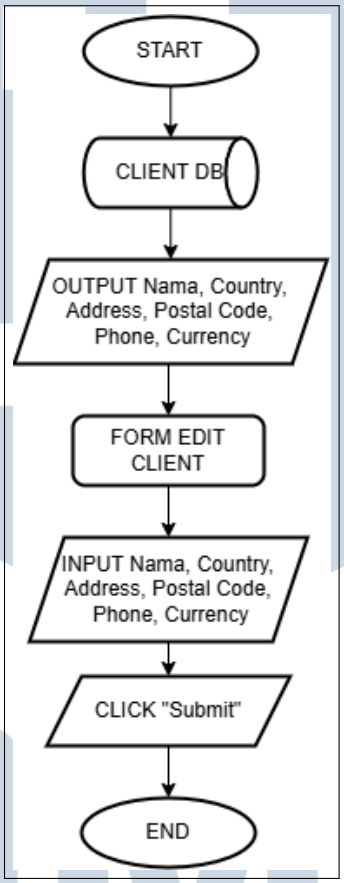


Gambar 3.10. Gambar Flowchart Create Client

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

11. Flowchart Edit Client

Pada gambar 3.11 menunjukkan alur atau *flowchart* yang menjelaskan fitur *edit client* yang ada pada *website InvoiceApp*. Fitur *edit client* berguna untuk mengubah isi dan detail *list client* yang sudah ada di dalam *database*.



Gambar 3.11. Gambar Flowchart Create Client

UNIVERSITAS MULTIMEDIA NUSANTARA

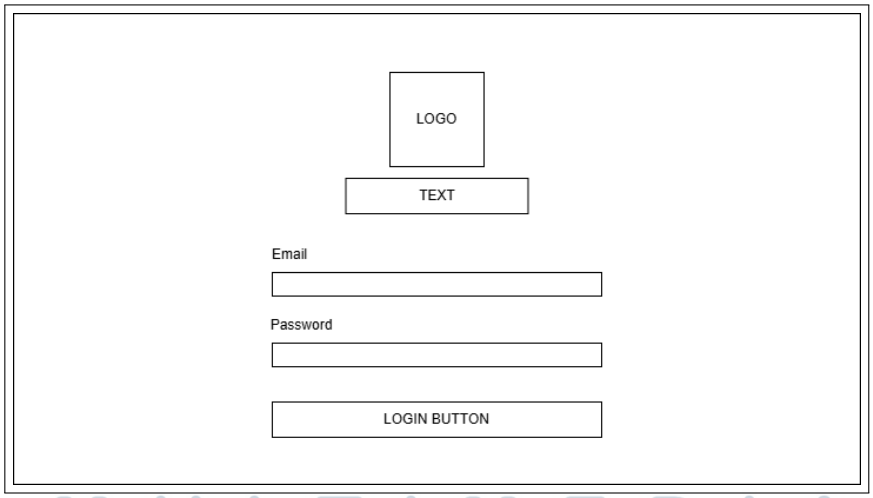
3.4.2 Mockup

Proses pembuatan mockup dilakukan sebagai tahap awal perancangan visual antarmuka website sebelum implementasi kode dilakukan. Dalam proses ini digunakan software draw.io yang mendukung pembuatan diagram dan mockup dengan pendekatan *low-fidelity*. Mockup ini bertujuan memberikan gambaran struktur halaman, peletakan komponen, serta navigasi dasar, sehingga mempermudah tim pengembang dalam memahami alur interaksi pengguna dan kebutuhan layout setiap fitur.[7]

A. Login Page Mockup

Gambar 3.12 menunjukkan desain awal atau *mockup* dari halaman login pada aplikasi *InvoiceApp*. *Mockup* ini digunakan sebagai acuan visual sebelum proses implementasi kode dilakukan.

Desain dibuat dengan tampilan yang sederhana namun fungsional, agar memudahkan pengguna dalam mengakses aplikasi. *Mockup* ini juga memperlihatkan struktur layout, warna, dan penempatan elemen yang akan digunakan dalam versi final halaman login. di webiste *InvoiceApp*

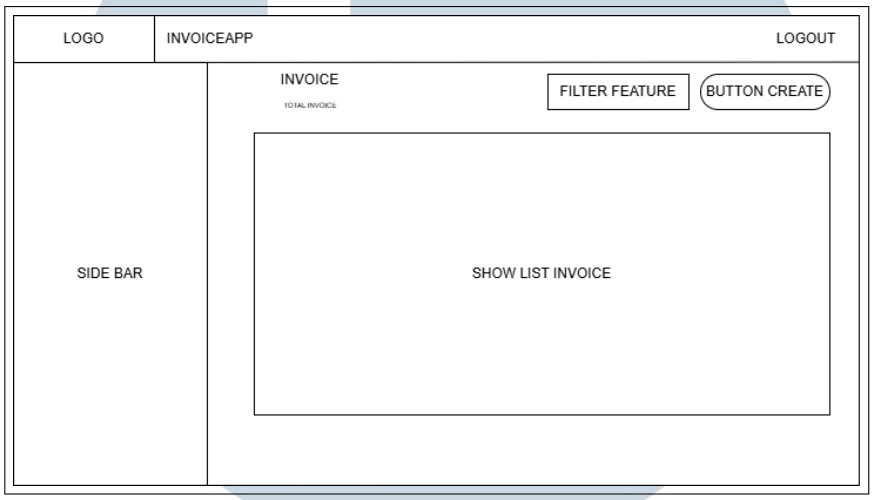


Gambar 3.12. Gambar Login Mockup

MULTIMEDIA NUSANTARA

B. Invoice Page Mockup

Gambar 3.13 menunjukkan desain awal atau *mockup* dari halaman *invoice page* pada aplikasi *InvoiceApp*. Dalam *mockup* ini, ditampilkan daftar invoice yang sudah dibuat oleh pengguna. Setiap invoice menampilkan informasi penting seperti nama klien, nomor invoice, tanggal, dan status pembayaran.

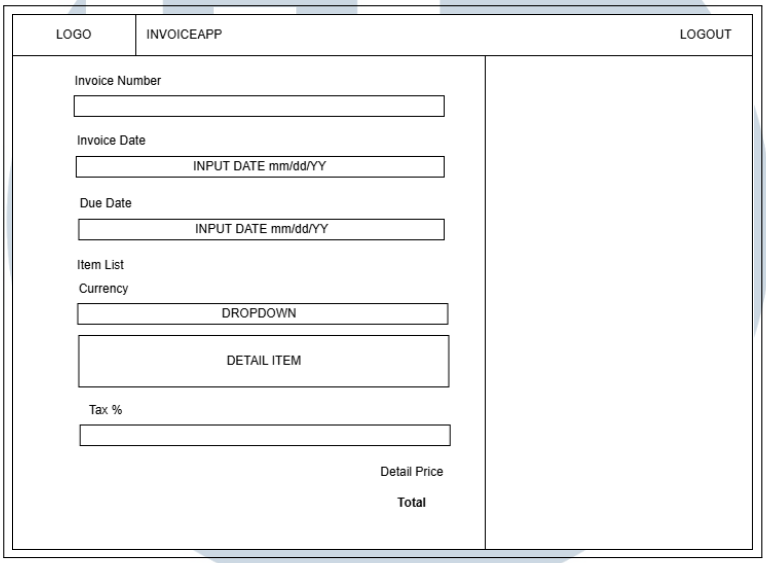


Gambar 3.13. Gambar Invoice Page Mockup

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

C. Create Invoice Mockup

Gambar 3.14 menunjukkan desain awal atau *mockup* dari halaman *create invoice page* pada aplikasi *InoviceApp*. Pada halaman ini berisi beberapa form yang nanti bisa diisi oleh pengguna.

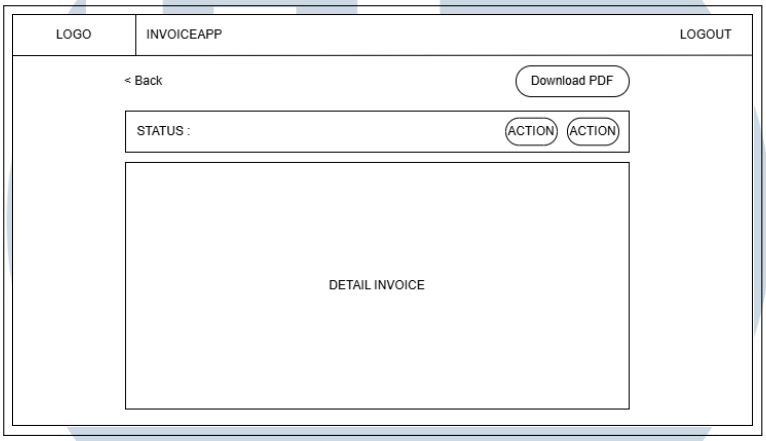


Gambar 3.14. Gambar Page Create Invoice

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

D. Detail Invoice Page Mockup

Gambar 3.15 menunjukkan desain awal atau *mockup* dari halaman *Detail Invoice Page*. Pada halaman ini berisi detail data berupa invoice yang sudah ditambahkan sebelumnya.

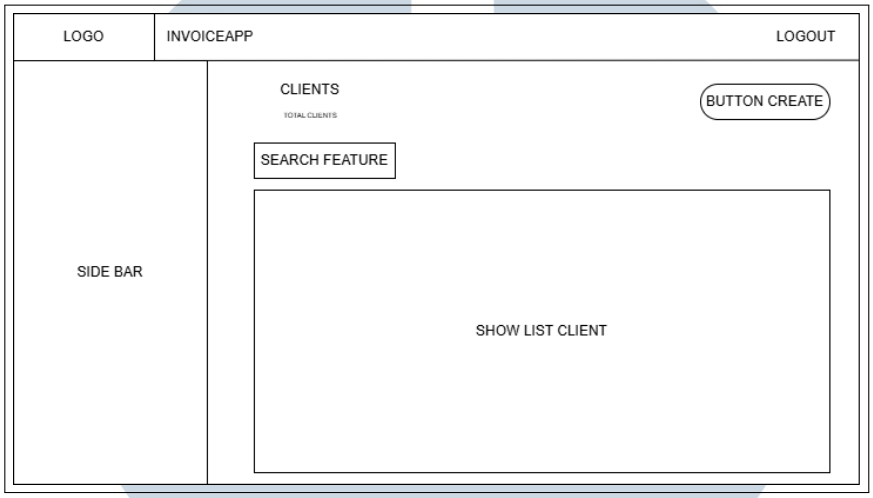


Gambar 3.15. Gambar Invoice Page Mockup

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

E. Client Page Mockup

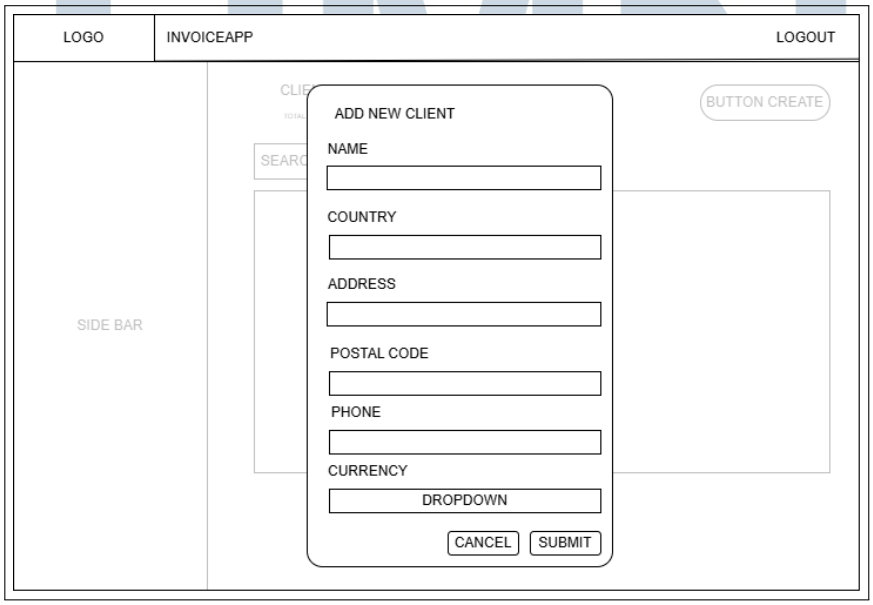
Gambar 3.16 menujukkan desain awal atau *mockup* dari halaman *Client Page*. Pada halaman ini berisi data dari client.



Gambar 3.16. Gambar Client Page Mockup

F. Create New Client Page Mockup

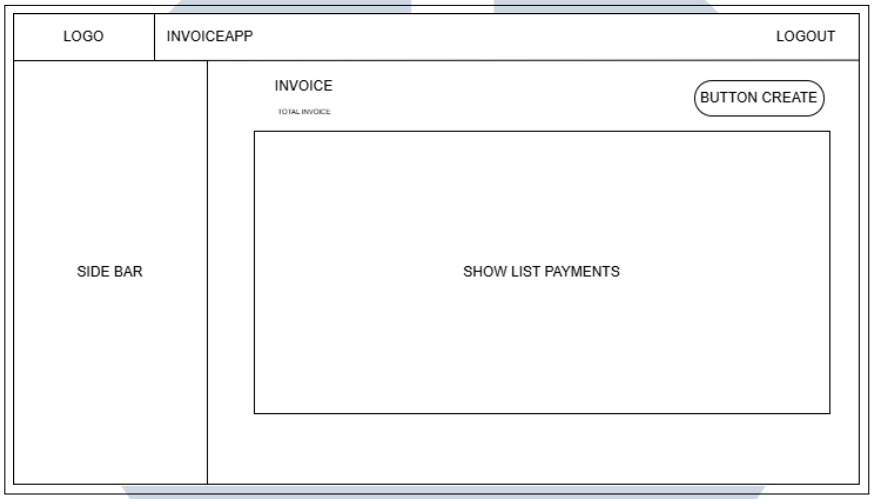
Gambar 3.19 menujukkan desain awal atau *mockup* dari halaman *Create New Client Page*. Pada halaman ini berisi form untuk user dapat memasukan data-data dari client.



Gambar 3.17. Gambar Client Page Mockup

G. Payment Page Mockup

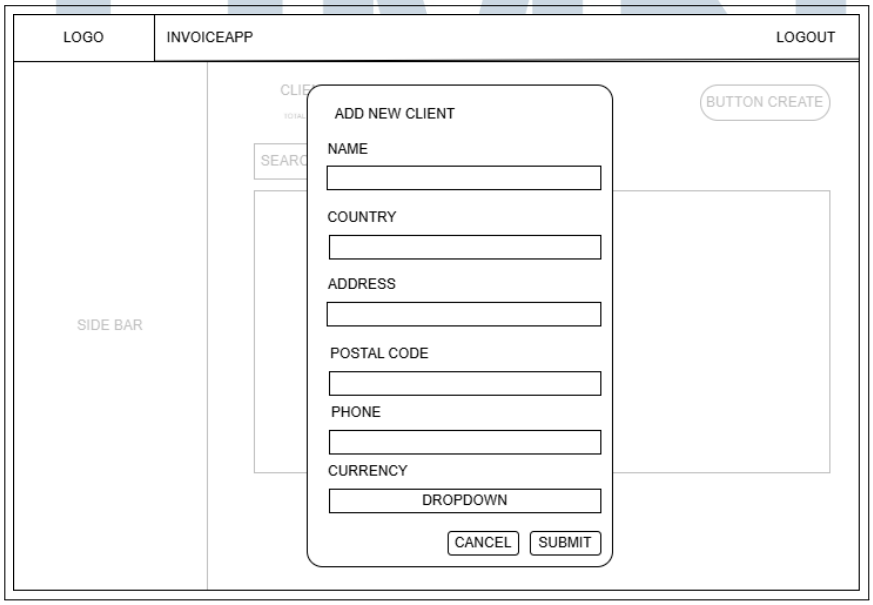
Gambar 3.18 menujukkan desain awal atau *mockup* dari halaman *Payment Page*. Pada halaman ini berisi data dari payment untuk invoice yang ada.



Gambar 3.18. Gambar Payment Page Mockup

H. Create New Payment Page Mockup

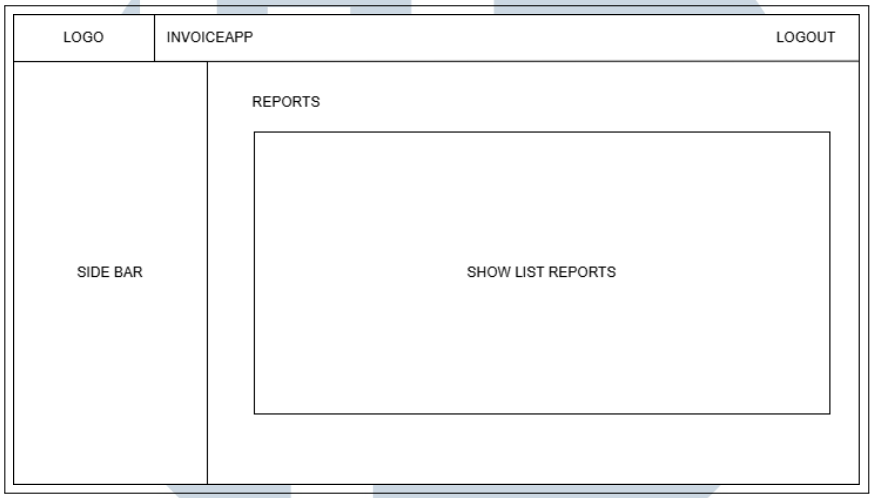
Gambar 3.19 menujukkan desain awal atau *mockup* dari halaman *Create New Payment Page Mockup*. Pada halaman ini pengguna dapat mengisi form untuk mengisi detail payment.



Gambar 3.19. Gambar Create New Payment Page Mockup

I. Create Reports Page Mockup

Gambar 3.20 menujukkan desain awal atau *mockup* dari halaman *Create Reports Page Mockup*. Pada halaman ini pengguna dapat melihat laporan untuk invoice yang sudah ada.



Gambar 3.20. Gambar Create Reports Page Mockup

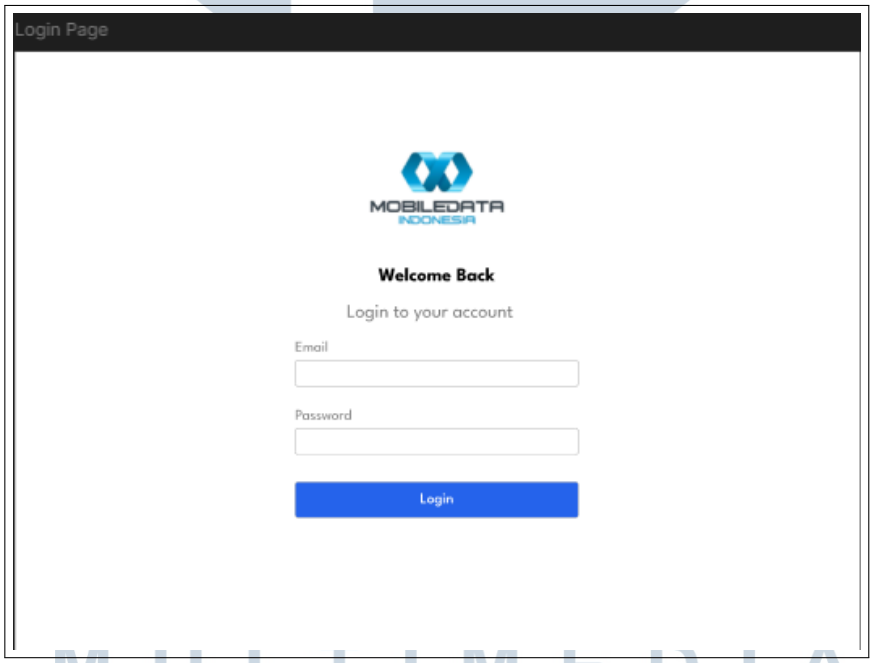
UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.5 *Prototype User Interface*

Pembuatan prototipe antarmuka pengguna dilakukan menggunakan Figma, sebuah platform desain berbasis cloud yang mendukung kolaborasi real-time dan pengembangan *UI* secara interaktif. *Figma* dipilih karena kemampuannya dalam menyusun komponen visual secara presisi, mendukung sistem desain modular, serta mempermudah proses validasi tampilan sebelum implementasi kode. Prototipe yang dihasilkan mencerminkan struktur dan fungsionalitas akhir dari website, termasuk navigasi, form input, dan elemen interaktif yang telah disesuaikan dengan kebutuhan pengguna akhir.[8]

3.5.1 Prototype Login Page

Gambar 3.21 menunjukkan hasil implementasi *login mockup* ke dalam *login prototype*, *login page* ini akan menjadi page pertama yang diakses oleh *user* saat mengakses *website*.



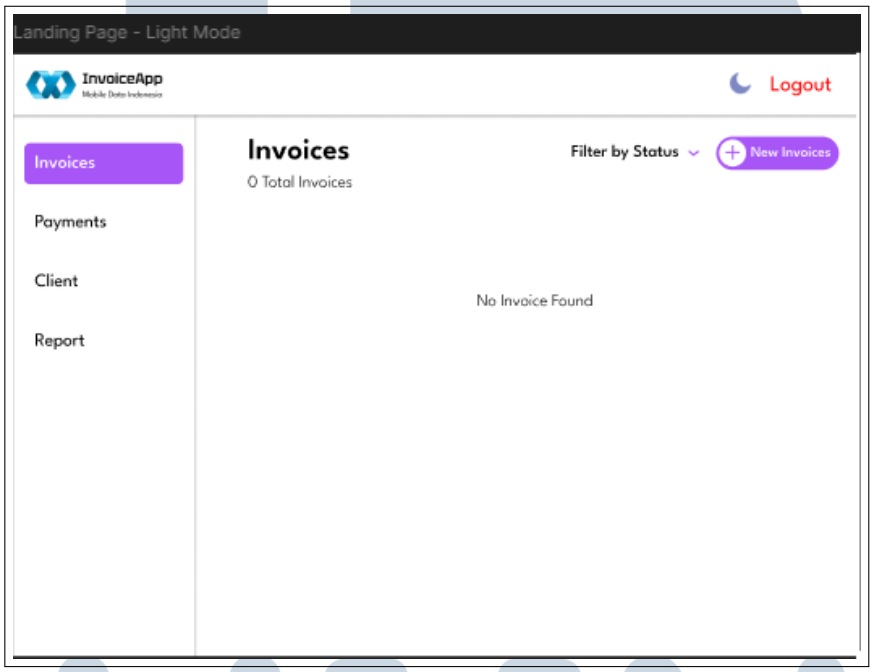
Gambar 3.21. Gambar Prototype Login Interface

MULTIMEDIA
NUSANTARA

3.5.2 *Prototype Invoice Page*

A. Prototype Invoice Page - Empty

Gambar 3.22 menunjukkan hasil implementasi dari *invoice page mockup* ke dalam *prototype invoice page*, *invoice page* ini akan menjadi page yang dapat menampung *invoice* dan *create invoice* oleh *user*.

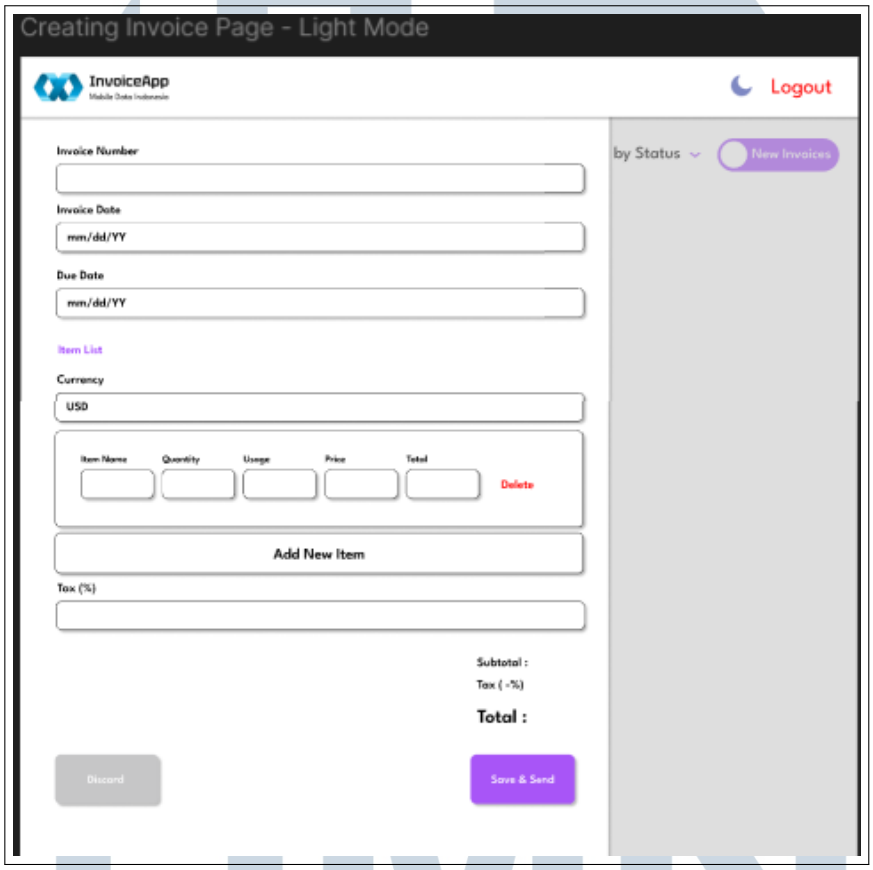


Gambar 3.22. Gambar Prototype Invoice Page

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

B. Prototype Create Invoice

Gambar 3.23 menunjukkan hasil implementasi dari *create invoice page mockup* ke dalam *prototype create invoice page*, *create invoice page* ini akan menjadi page yang bisa menambahkan invoice.

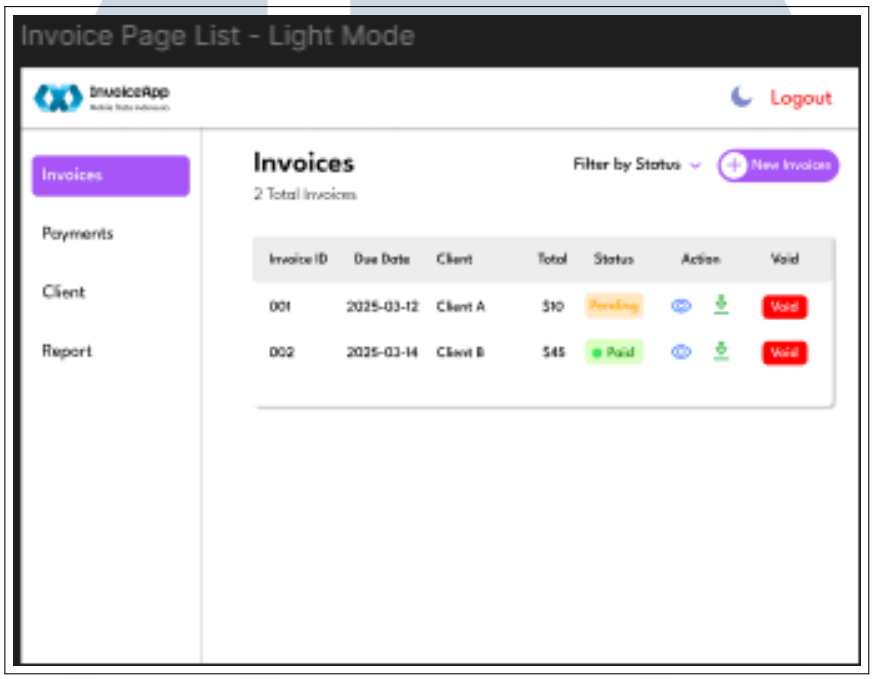


Gambar 3.23. Gambar Prototype Create Invoice Page Empty

UNIVERSITAS MULTIMEDIA NUSANTARA

C. Prototype Invoice Page

Gambar 3.24 menunjukkan hasil implementasi dari *invoice page mockup* ke dalam *prototype invoice page*, *invoice page* ini akan menjadi page yang dapat menampung *invoice* dan *create invoice* oleh *user*. pada gambar 3.24 menampilkan *prototype invoice page* yang sudah berisi *list data invoice*.

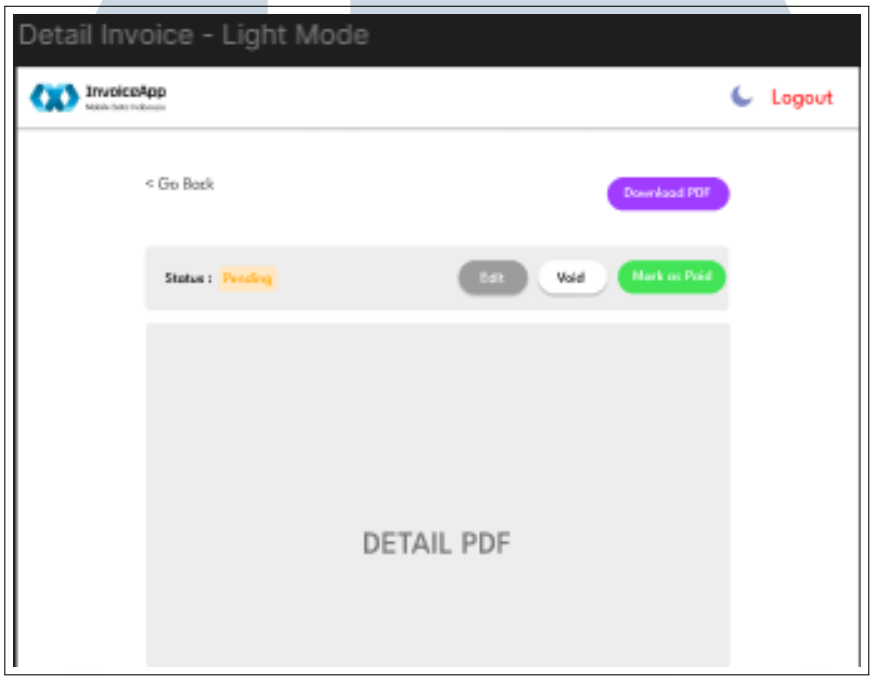


Gambar 3.24. Gambar Prototype Create Invoice Page

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

D. Prototype Detail Invoice Page - Status Paid

Gambar 3.25 menunjukkan hasil implementasi dari *detail invoice page mockup* ke dalam *prototype detail invoice page*, *detail invoice page* ini akan menjadi page yang dapat melihat *detail* dari *invoice* yang telah ditambahkan dan *detail invoice* dengan status *pending*.



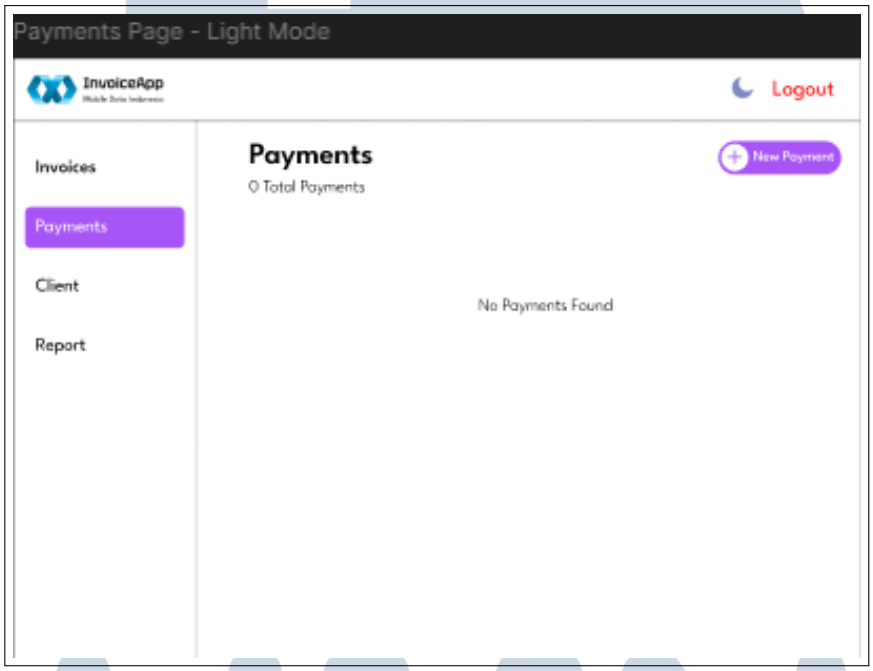
Gambar 3.25. Gambar Prototype Detail Invoice Page dengan Status Pending

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.5.3 Prototype Payment Page

A. Prototype Payment Page - Empty

Gambar 3.26 menunjukkan hasil implementasi dari *payment page mockup* ke dalam *payment page prototype*, *payment page* ini akan menjadi page yang dapat menampung *list payment* dan *create payment* oleh *user*.

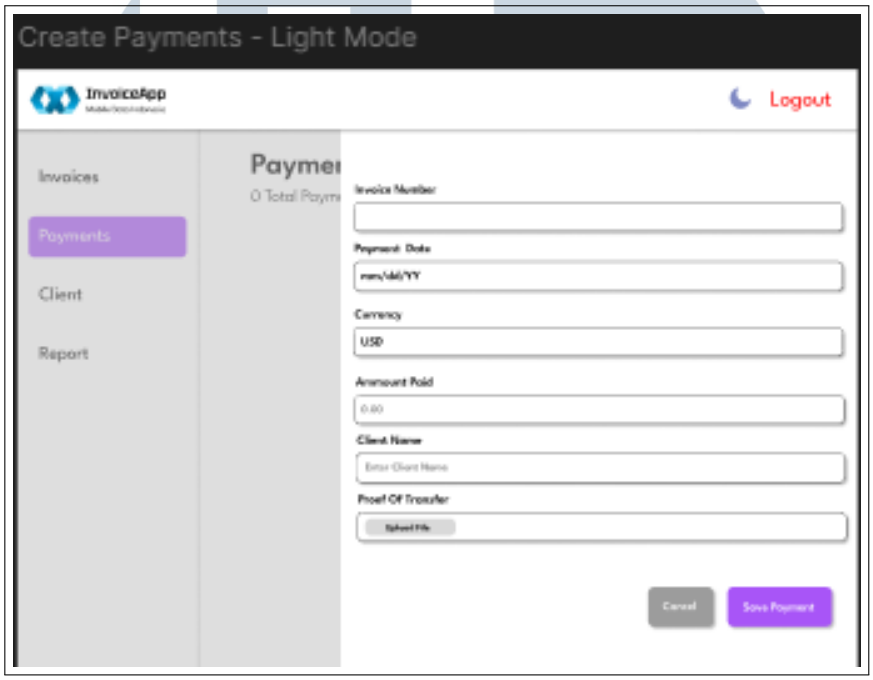


Gambar 3.26. Gambar Prototype Payment Page Interface

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

B. Prototype Create Payment

Gambar 3.27 menunjukkan hasil implementasi dari *create payment mockup* ke dalam *create payment prototype*, *create payment* akan menjadi halaman yang dapat mengisiisi formulir untuk menambahkan data *payment*.

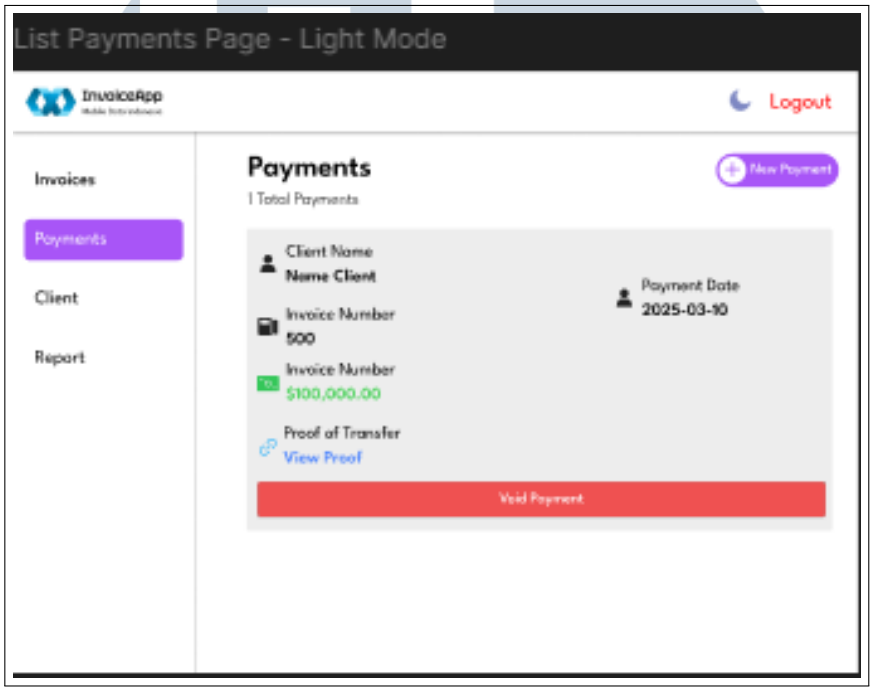


Gambar 3.27. Gambar Prototype Create Payment

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

C. Prototype List Payment

Gambar 3.28 menunjukkan hasil implementasi dari *list payment mockup* ke dalam *list payment prototype*, *list payment* menjadi halaman yang menampilkan data dari *payment*.



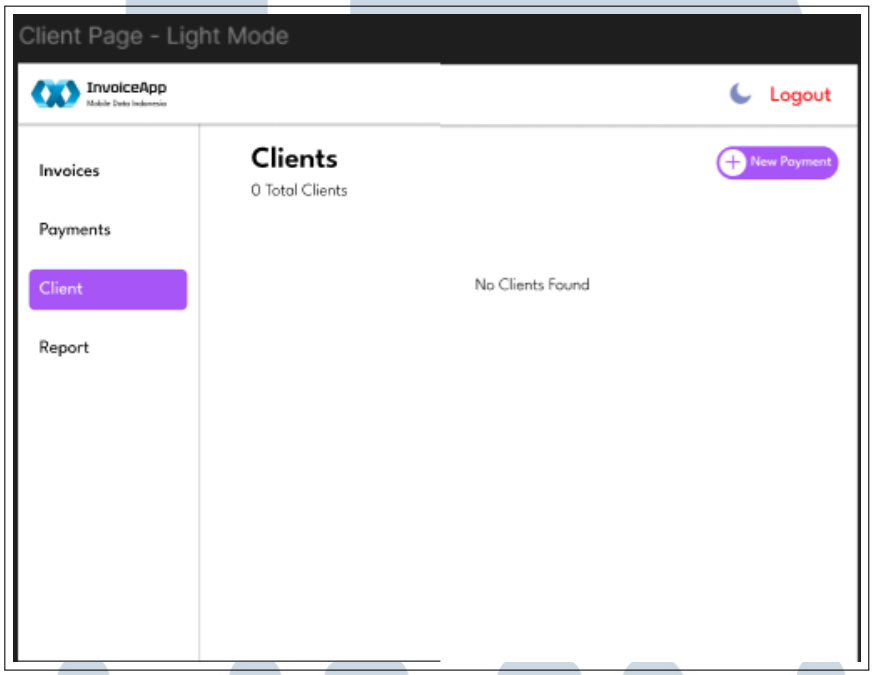
Gambar 3.28. Gambar Prototype List Payment

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.5.4 Prototype Client Page

A. Prototype Client Page - Empty

Gambar 3.29 menunjukkan hasil implementasi dari *client page mockup* ke dalam *client page prototype*, *client page* ini akan menjadi page yang dapat menampung *list client* dan *create client* oleh *user*.

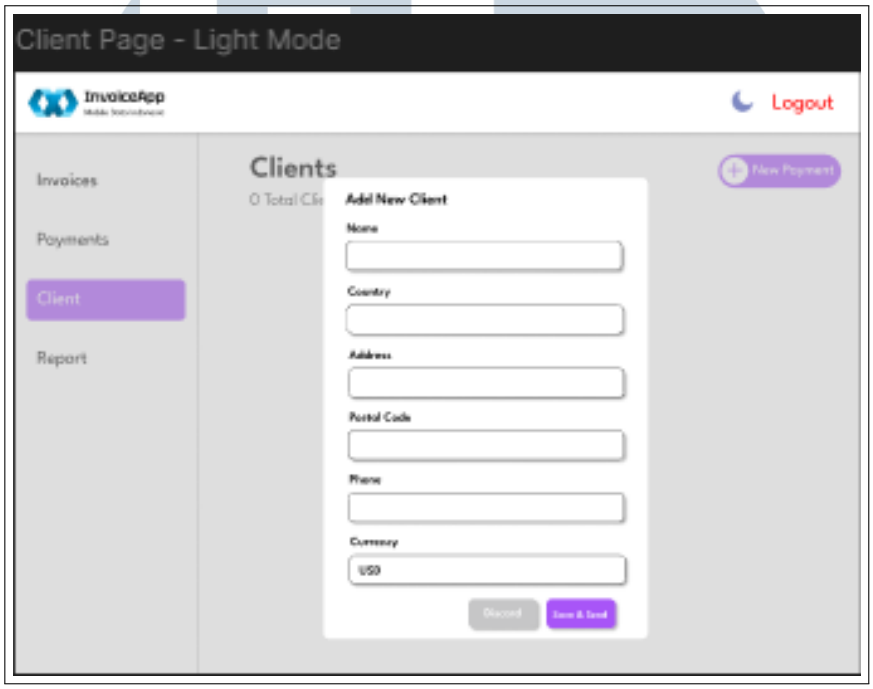


Gambar 3.29. Gambar Prototype Client Page

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

B. Prototype Create Client

Gambar 3.30 menunjukkan hasil implementasi dari *create client page mockup* ke dalam *create client page prototype*, *create client page* akan menjadi halaman yang dapat mengisi formulir untuk menambahkan data *client*.

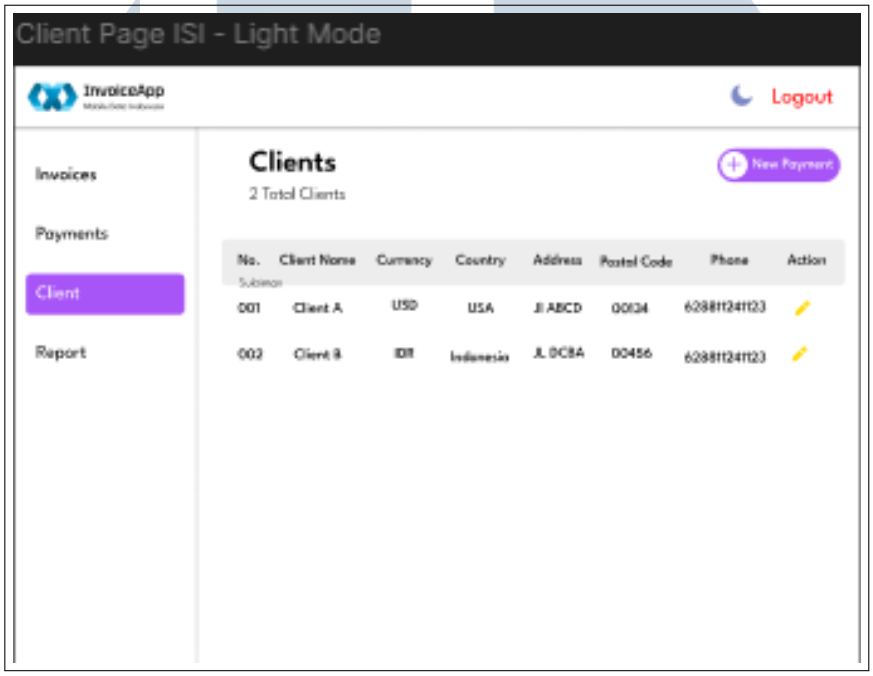


Gambar 3.30. Gambar Prototype Create Client Page

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

C. Prototype List Client Page

Gambar 3.31 menunjukkan hasil implementasi dari *list client page mockup* ke dalam *list client page prototype*, *list client page* ini akan menampilkan data dari *client* yang sudah di tambahkan melalui *create client*.



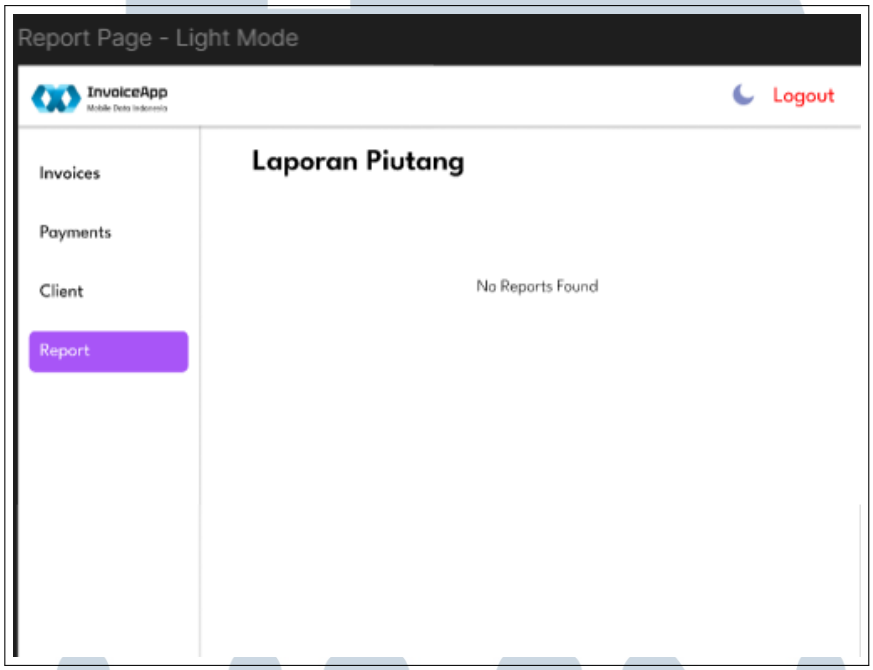
Gambar 3.31. Gambar Prototype List Client Page

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.5.5 Prototype Report Page

A. Prototype Report Page - Empty

Gambar 3.32 menunjukkan hasil implementasi dari *report page mockup* ke dalam *report page prototype*, *report page* ini akan menjadi halaman yang dapat menampung *list report*.

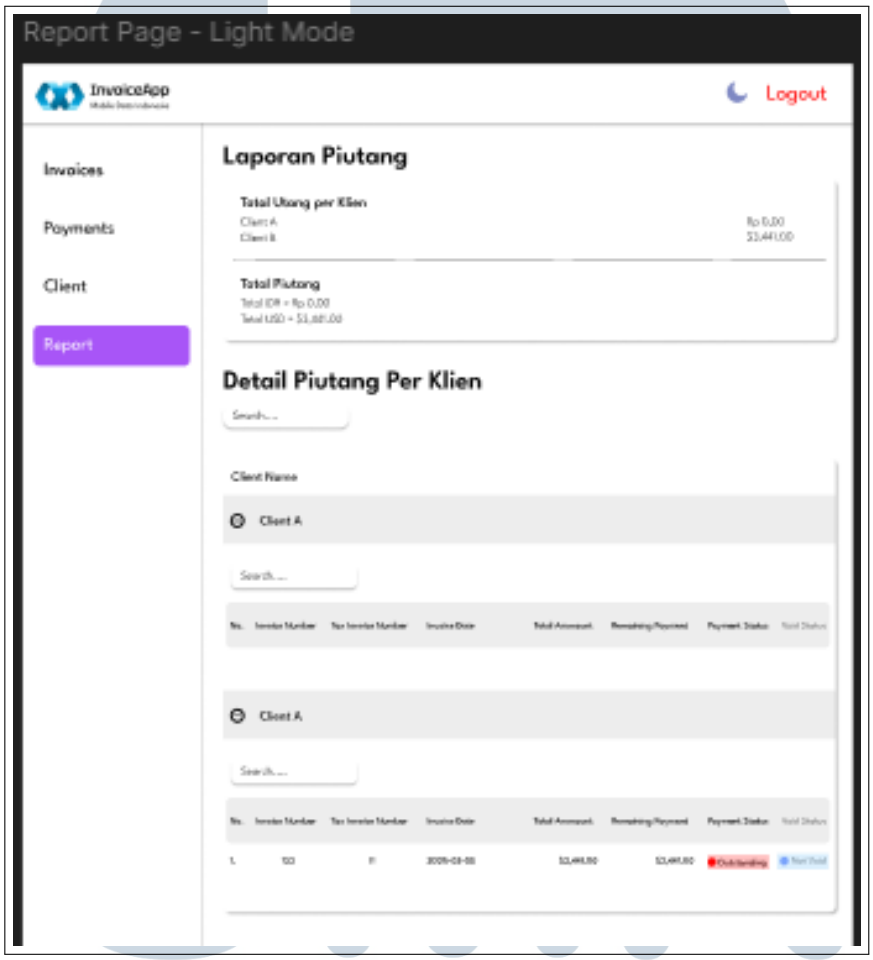


Gambar 3.32. Gambar Prototype Report Page Interface

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

B. Prototype List Report Page

Gambar 3.33 menunjukkan hasil implementasi dari *list report page mockup* ke dalam *list report page prototype*, *list report page* ini akan menjadi halaman yang menampilkan data dari *invoice* dan *client* yang dijadikan satu menjadi sebuah *report*.



Gambar 3.33. Gambar List Prototype Report Page Interface

UNIVERSITAS
MULTIMEDIA
NUSANTARA

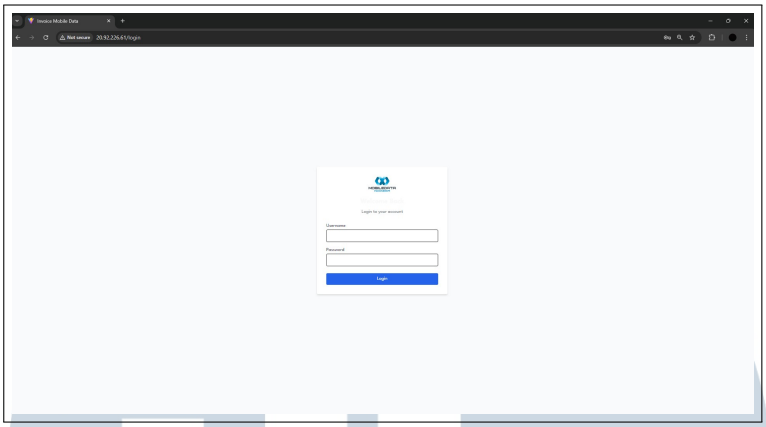
3.6 Implementasi

Tahap implementasi adalah bagian krusial dalam proses pengembangan website, di mana desain dan alur yang telah direncanakan sebelumnya diterjemahkan ke dalam bentuk website yang sebenarnya. Pada fase ini, berbagai elemen seperti antarmuka, struktur, dan fitur yang telah dirancang mulai diwujudkan melalui penulisan kode dan pengaturan teknis, guna memastikan bahwa hasil akhir tidak hanya sesuai dengan tampilan yang diinginkan, tetapi juga berjalan dengan baik sesuai fungsinya.

3.6.1 Login Page

Gambar 3.34 menampilkan hasil implementasi kode pada bagian Login Page dari aplikasi *InvoiceApp*. Halaman ini merupakan komponen penting yang menjadi gerbang utama bagi pengguna untuk dapat mengakses sistem. Login Page dirancang agar pengguna dapat masuk ke dalam aplikasi dengan menggunakan kredensial akun yang telah terdaftar sebelumnya, sehingga memastikan keamanan akses terhadap data dan fitur-fitur yang tersedia di dalam sistem.
Pada halaman ini, pengguna diminta untuk memasukkan username dan kata sandi yang sudah dibuatkan oleh managemen. Setelah berhasil melakukan proses autentikasi, sistem akan mengarahkan pengguna ke halaman utama. Dalam aplikasi ini terdapat dua jenis peran (role), yaitu *finance* dan *management*. Pengguna dengan role *finance* memiliki akses terhadap fitur-fitur yang berhubungan dengan pengelolaan transaksi keuangan, sedangkan role management lebih difokuskan untuk monitoring dan evaluasi kinerja serta pelaporan. Dan untuk role *managment* juga memiliki akses yang serupa, untuk sekarang yang membedakan hanya *management* bisa menghapus invoice sudah tidak ingin dilihat.

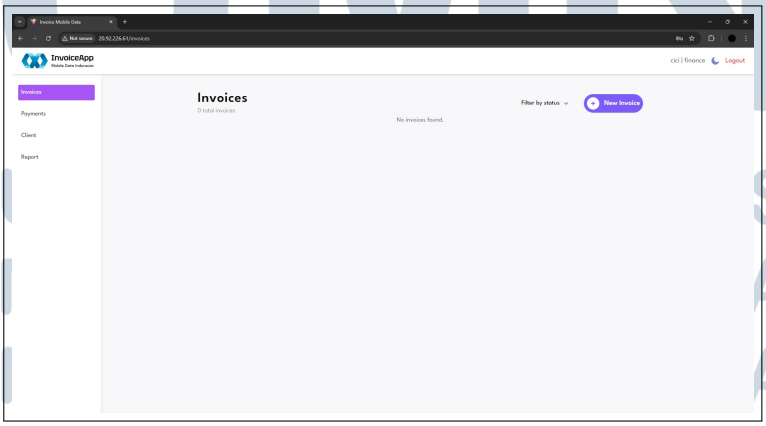
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.34. Implementasi Login Page

3.6.2 Invoice Page - Empty

Gambar 3.35 menampilkan hasil implementasi kode pada bagian invoice page dari aplikasi InvoiceApp. Invoice page ini berfungsi sebagai halaman utama (landing page) yang pertama kali ditampilkan kepada pengguna setelah berhasil melakukan proses login. Halaman ini dirancang untuk menampilkan daftar invoice yang dimiliki oleh pengguna. Namun, apabila sistem mendeteksi bahwa tidak terdapat data invoice yang tersimpan di dalam basis data (misalnya untuk pengguna baru yang belum pernah membuat invoice), maka halaman akan menampilkan tampilan kosong atau pesan khusus yang menginformasikan bahwa belum ada data yang tersedia, sebagaimana terlihat pada gambar 3.35. Tampilan ini juga bertujuan memberikan petunjuk visual kepada pengguna bahwa mereka perlu membuat invoice baru untuk mulai menggunakan fitur-fitur utama dari aplikasi.



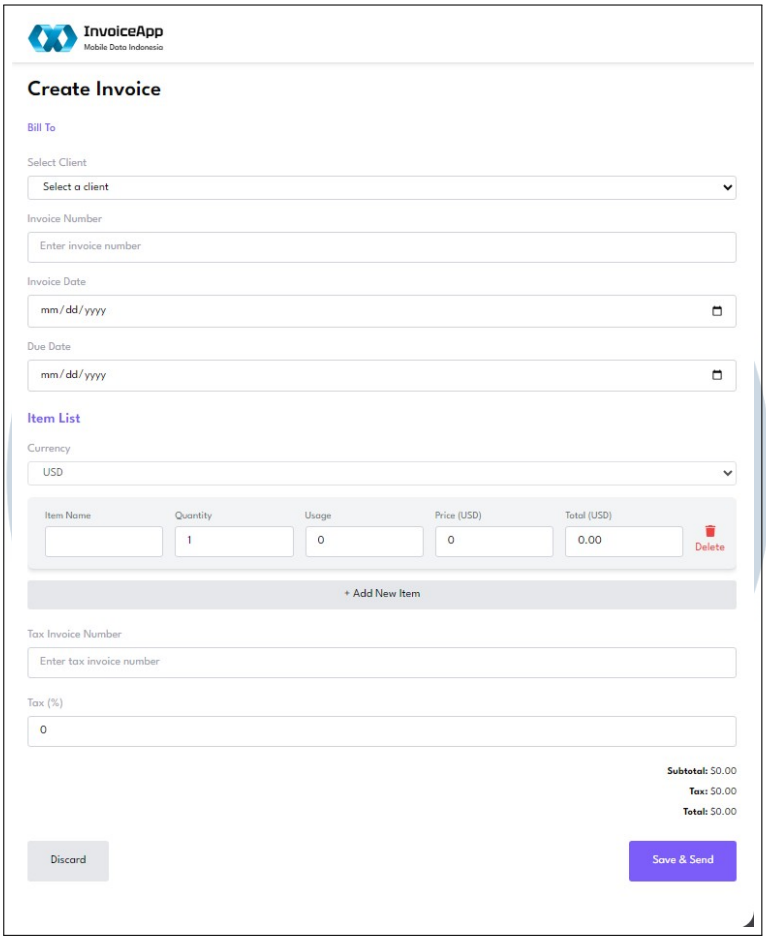
Gambar 3.35. Implementasi Invoice Page

3.6.3 Create New Invoice

Gambar 3.36 memperlihatkan hasil implementasi kode pada bagian create new invoice dalam aplikasi InvoiceApp. Fitur ini dirancang untuk memungkinkan pengguna menambahkan data invoice baru yang nantinya akan ditampilkan dan dikelola melalui halaman invoice page. Proses pembuatan invoice dilakukan melalui sebuah form input yang disediakan secara interaktif, dengan berbagai kolom isian yang wajib dilengkapi oleh pengguna agar invoice dapat tersimpan dengan benar ke dalam basis data.

Adapun elemen-elemen yang harus diisi dalam form tersebut mencakup beberapa bagian penting, yaitu: pemilihan klien melalui opsi select client, pengisian nomor invoice atau invoice number sebagai identifikasi unik setiap invoice, penentuan tanggal pembuatan invoice melalui invoice date, serta batas waktu pembayaran melalui due date. Selain itu, pengguna juga harus mengisi daftar item atau jasa yang ditagihkan dalam bagian item list, serta mencantumkan nomor faktur pajak pada kolom tax invoice number dan nilai pajaknya pada bagian tax. Seluruh data yang diisikan dalam form ini akan divalidasi sebelum dikirim dan disimpan ke dalam sistem, sehingga memastikan integritas dan kelengkapan data yang dibutuhkan untuk keperluan dokumentasi dan penagihan.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



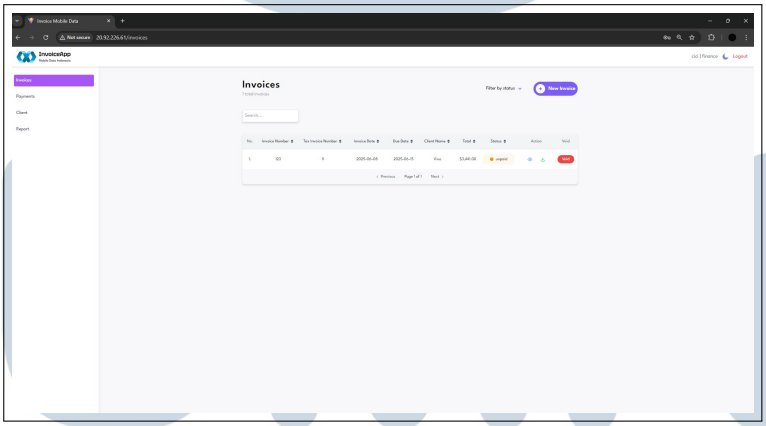
Gambar 3.36. Implementasi Create New Invoice

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.6.4 Invoice Page

Gambar 3.37 menampilkan tampilan invoice page pada saat data invoice sudah berhasil ditambahkan ke dalam basis data melalui fitur create new invoice. Tidak seperti tampilan awal yang kosong ketika belum ada data yang tersimpan, halaman ini kini telah menampilkan daftar invoice yang telah dibuat oleh pengguna. Setiap baris pada daftar tersebut merepresentasikan satu entri invoice yang berisi informasi penting, seperti nama klien, nomor invoice, tanggal pembuatan, tenggat waktu pembayaran, serta status dari masing-masing invoice (misalnya: belum dibayar, sudah dibayar, atau jatuh tempo).

Halaman ini merupakan representasi dinamis dari data yang tersimpan dalam sistem, yang ditampilkan secara real-time setelah proses input data melalui form pada fitur create new invoice. Dengan adanya daftar ini, pengguna dapat dengan mudah memantau, mengelola, dan menindaklanjuti invoice yang sudah dibuat sebelumnya. Tampilan antarmukanya dirancang agar informatif dan mudah digunakan, memungkinkan pengguna untuk melakukan tindakan lanjutan seperti melihat detail, mengedit, atau menghapus invoice tertentu sesuai kebutuhan.



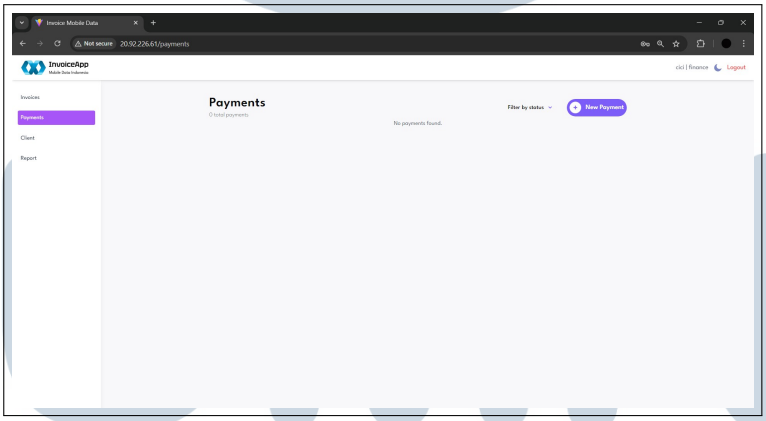
Gambar 3.37. Implementasi Invoice Page

UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.6.5 Payment Page

Gambar 3.38 menampilkan tampilan implementasi dari Payment Page pada aplikasi InvoiceApp. Halaman ini berfungsi sebagai pusat informasi yang dirancang untuk menampilkan data lengkap mengenai pembayaran *payment* yang telah ditambahkan ke dalam database oleh pengguna. Seluruh transaksi pembayaran yang berhasil diproses akan muncul di halaman ini secara terstruktur dan mudah dibaca, sehingga pengguna dapat dengan cepat memverifikasi status pembayaran dari berbagai klien.

Desain antarmuka halaman ini dibuat minimalis dan intuitif, dengan menampilkan daftar pembayaran secara real-time, serta opsi untuk menambahkan pembayaran baru melalui tombol *New Payment*. Selain itu, terdapat fitur navigasi sidebar di sisi kiri untuk memudahkan perpindahan antar halaman utama aplikasi. Dengan adanya halaman *Payment Page*, proses pengelolaan data transaksi menjadi lebih efisien dan transparan, karena semua riwayat pembayaran terdokumentasi dengan baik dan dapat diakses kapan saja. Fitur ini sangat penting dalam memastikan akurasi laporan keuangan dan memudahkan proses audit internal.



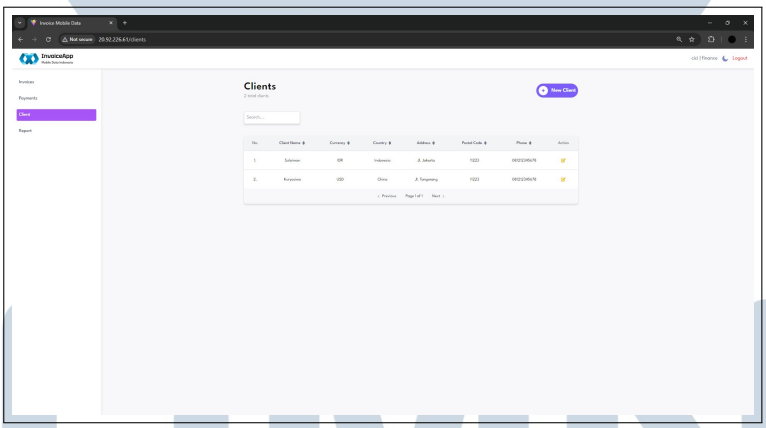
Gambar 3.38. Implementasi Payment Page

UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.6.6 Client Page

Gambar 3.39 menampilkan tampilan *client page* dalam aplikasi *InvoiceApp*. Halaman ini berfungsi sebagai pusat informasi yang secara khusus dirancang untuk menampilkan data lengkap dari setiap klien yang telah disimpan ke dalam basis data oleh pengguna. Melalui halaman ini, pengguna dapat melihat daftar klien yang pernah atau sedang menjadi bagian dari transaksi, baik sebagai penerima invoice maupun sebagai mitra kerja dalam proses bisnis.

Halaman ini dirancang dengan antarmuka yang intuitif dan terstruktur secara rapi, sehingga memudahkan pengguna dalam mencari dan mengelola data klien. Informasi yang ditampilkan diambil langsung dari database secara real-time, memastikan bahwa data yang terlihat selalu up-to-date. Dengan adanya *client page*, pengguna dapat dengan cepat mengakses data klien yang dibutuhkan saat membuat invoice baru, serta menjaga akurasi dan konsistensi data dalam sistem.



Gambar 3.39. Implementasi Client Page

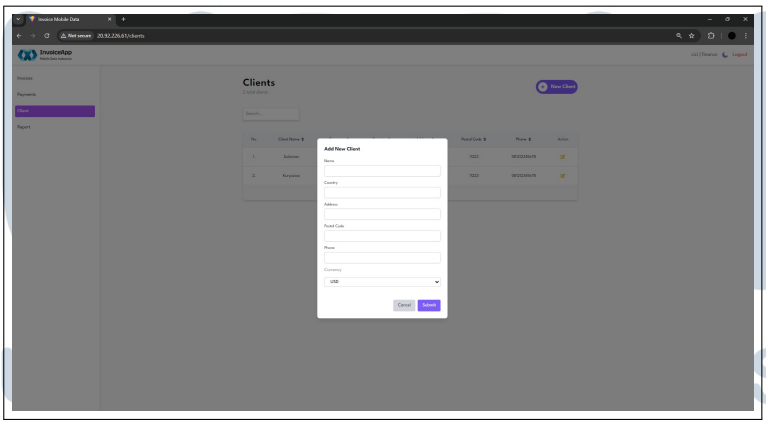
UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.6.7 Add Client

Gambar 3.40 menampilkan tampilan dari fitur *add client* dalam aplikasi *InvoiceApp*. Fitur ini merupakan komponen penting yang menyediakan sebuah *form input* bagi pengguna untuk menambahkan data klien baru ke dalam sistem. Formulir ini dirancang agar mudah digunakan dan mencakup berbagai kolom yang relevan untuk mengidentifikasi dan menyimpan informasi klien secara lengkap dan akurat ke dalam basis data.

Dalam form tersebut, pengguna diminta untuk mengisi sejumlah informasi penting yang berkaitan dengan identitas klien, seperti nama lengkap atau nama perusahaan, alamat email, nomor telepon, alamat fisik, dan informasi tambahan lainnya yang diperlukan untuk keperluan administratif dan pembuatan invoice. Setiap kolom dalam form diberi label yang jelas, dan sistem dapat dilengkapi dengan validasi data untuk memastikan bahwa input yang diberikan sesuai format dan tidak ada informasi yang terlewatkan.

Dengan adanya fitur ini, pengguna tidak hanya dimudahkan dalam menambahkan klien baru secara cepat dan efisien, tetapi juga dapat menjaga kelengkapan dan konsistensi informasi dalam sistem. Gambar 3.40 merepresentasikan bagaimana proses input data klien difasilitasi melalui desain antarmuka yang fungsional dan ramah pengguna.



Gambar 3.40. Implementasi Add Client Page

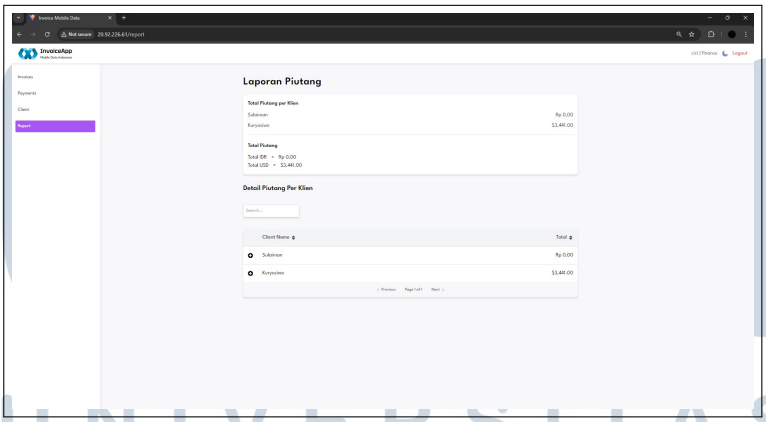
MULTIMEDIA NUSANTARA

3.6.8 Report Page

Gambar 3.41 menampilkan tampilan dari *reports page* dalam aplikasi *InvoiceApp*, yang berfungsi sebagai halaman khusus untuk menampilkan rincian data dari setiap item yang telah diinputkan oleh pengguna melalui fitur *create new invoice*. Halaman ini menyajikan data dalam bentuk daftar atau tabel yang terstruktur, sehingga memudahkan pengguna dalam memantau dan mengevaluasi seluruh transaksi atau item yang telah dimasukkan ke dalam sistem.

Selain berfungsi sebagai alat pelaporan, halaman ini juga dapat dilengkapi dengan fitur tambahan seperti pencarian, *filter* berdasarkan tanggal atau klien, serta ekspor data ke format lain seperti PDF atau Excel, untuk mempermudah dokumentasi dan analisis. Tampilan yang sistematis dan informatif ini dirancang untuk membantu pengguna dalam melakukan pengecekan data, menghitung total transaksi, atau melakukan audit terhadap aktivitas yang telah terjadi di dalam aplikasi.

Dengan demikian, gambar 3.41 merepresentasikan bagaimana data dari item-item invoice yang telah dibuat sebelumnya dapat disusun dan ditampilkan kembali dalam format laporan yang rapi, lengkap, dan mudah dipahami, sebagai bagian dari sistem pelacakan dan dokumentasi keuangan yang efisien di dalam *InvoiceApp*.



Gambar 3.41. Implementasi Report Page

UNIVERSITAS MULTIMEDIA NUSANTARA

3.7 Kendala dan Solusi yang Ditemukan

Selama menjalani periode magang, ada beberapa kendala yang memerlukan adaptasi dan penyelesaian secara bertahap. Kendala utama yang di hadapi antara lain adalah:

- Mempelajari konsep project. Pada awal magang, belum memahami sepenuhnya dengan konteks dan alur kerja dari proyek yang ingin dibuat, Konsep proyek yang cukup kompleks dan spesifik terhadap kebutuhan internal perusahaan menuntut pemahaman yang mendalam serta penyusaian terhadap tools yang dibutuhkan.
- Mempelajari hal-hal diluar IT yang harus dipelajari lebih lanjut. Selain tugas-tugas teknis sebagai bagian dari divisi IT, juga ada beberapa hal yang harus di pelajari lebih lanjut, khususnya berkaitan dengan proses bisnis dan keuangan. Hal ini menjadi tantangan tersendiri karena memerlukan pemahaman tambahan.
- Perbedaan struktur data antara frontend dan backend saat integrasi API.

Solusi yang dilakukan antara lain adalah :

- Membahas konsep bersama mentor dan teman magang. Yaitu secara aktif mengikuti sesi diskusi, baik secara formal maupun informal, untuk mendapatkan pemahaman yang lebih menyuluruh mengenai konep dari proyek yang akan dikerjakan. Bimbingan dari mentor serta kolaborasi dengan tim IT yang lain sangat membantu dalam pengerjaan proyek ini.
- Membahasnya bersama rekan kerja yang berdivisi finance. Untuk memahami aspek non-teknis, dengan berkomunikasi dengan staf dari divisi finance. Langkah ini dilakukan agar dapat memahmi konteks dan tujuan dari fitur yang dikebangkan, khususnya yang berkaitan dengan proses keuangan atau administrasi.
- *Refactoring* struktur data agar seragam dengan backend menggunakan *Axios* dan *JSON Schema Validation*.

Dengan kolaboratif dan sikap proaktif dalam mencari informasi, kendala-kendala yang dihadapi selama masa magang dapat diselesaikan secara bertahap. Hal ini juga memberikan pengalaman yang berharga dalam menghadapi situasi diluar zona nyaman dan memperluas pemahaman lintas bidang.