Text

Description automatically generatedLogo

Description automatically generated

LAPORAN PROJECT WORK

**Pengembangan Aplikasi RECYLOOP: Sistem Informasi Pengelolaan dan Penjemputan Sampah Terintegrasi Berbasis Web dengan Fitur Reward System.**

MOHAMMAD ALHABSYI MAHESA

24018/2067.063

**GURU PEMBIMBING :**

Yanuar Setyoningsih, S.Pd

Venny Meida Hersianty, S.Tr.Kom

**BIDANG KEAHLIAN TEKNOLOGI INFORMASI DAN KOMUNIKASI**

**PROGRAM KEAHLIAN PENGEMBANGAN PERANGKAT LUNAK DAN GIM**

**KONSENTRASI KEAHLIAN REKAYASA PERANGKAT LUNAK**

**SMK PGRI 3 MALANG**

**2026**

# **LEMBAR PERSETUJUAN**

Judul : Pengembangan Aplikasi RECYLOOP: Sistem Informasi Pengelolaan dan Penjemputan Sampah Terintegrasi Berbasis Web dengan Fitur Reward System

Oleh : MOHAMMAD ALHABSYI MAHESA

Nomor Induk : 24018/2067.063

|  |  |
| --- | --- |
| **Mengetahui** | |
| **Pembimbing 1**  Yanuar Setyoningsih, S.Pd | **Pembimbing 2**  Venny Meida Hersianty, S.Tr.Kom |
| **Kepala Bengkel**  Eko Mulyanto, S.Pd | **Guru Wali**  Sri Ira Herawati, S.Pd |
| **Menyetujui :**  **Kepala Bidang TIK**  Yanri Nur Wibowo, ST | |

# **LEMBAR PENGESAHAN**

Judul : Pengembangan Aplikasi RECYLOOP: Sistem Informasi Pengelolaan dan Penjemputan Sampah Terintegrasi Berbasis Web dengan Fitur Reward System

Oleh : MOHAMMAD ALHABSYI MAHESA

Nomor Induk : 24018/2067.063

Tanggal Ujian :

|  |  |
| --- | --- |
| **Pembimbing** | |
| **Pembimbing 1**  Yanuar Setyoningsih, S.Pd | **Pembimbing 2**  Venny Meida Hersianty, S.Tr.Kom |
| **Penguji** | |
| **Penguji 1**  (………………………………) | **Penguji 2**  (...................................) |

# **UCAPAN TERIMAKASIH**

Puja dan puji syukur ke hadirat Allah SWt, yang telah membari rahmat, hidayah, dan kesehatan sehingga penulis bisa menyelesaikan Laporan Tugas Akhir ini. Laporan ini di buat sesuai dengan pemahaman sang penulis dapat waktu pembelajaran di SMK PGRI 3 Malang.

Serta tidak lupa sholawat serta salam yang penulis limpahkan kepada baginda Nabi besar Muhammad Saw. Dan tak lupa penulis ucapkan beribu – ribu terimakasih kepada semua pihak yang telah membimbing penulis sampai selesai nya Laporan Tugas Akhir ini selesai. Banyak pihak telah membantu penulis untuk menyelesaikan Laporan ini oleh karena itu penulis mengucapkan terimakasih kepada :

1. Tuhan Yang Maha esa
2. Kedua orang tua yang telah membimbing dan juga mendoakan penulis agar dapat menyelesaikan Laporan ini.
3. Bapak Moch. Lukman Hakim, ST.MM selaku kepala sekolah SMK PGRI 3 Malang.
4. Bapak Yanri Nur Wibowo, ST selaku Kepala Bidang (Kabid) Teknik Informasi dan Komunikasi (TIK).
5. Bu Sri Ira Herawati, S.Pd selaku wali kelas yang telah memberikan nasehat agar anak didik nya menjadi Orang yang berguna bagi nusa dan bangsa serta orang lain.
6. Ibu Yanuar Setyoningsih S.Pd dan Ibu Venny Meida Hersianty, S.Tr.Kom yang telah memberikan ilmu nya dan membimbing hingga laporan ini selesai.
7. Seluruh teman-teman yang telah membantu menyelesaikan Laporan Tugas Akhir ini.

# **KATA PENGANTAR**

Pengumpulan Laporan Project Work tentang Pengembangan Aplikasi RECYLOOP: Sistem Informasi Pengelolaan dan Penjemputan Sampah Terintegrasi Berbasis Web yang menjadi latar belakang sebagai syarat kelulusan yang ditetapkan oleh pihak sekolah. Dan juga pengumpulan Laporan Tugas Akhir ini sebagai pemenuhan dari Tugas Akhir.

Malang, 29 Januari 2026

(Mohammad Alhabsyi Mahesa)

24018/2067.063

**DAFTAR ISI**

[**LEMBAR PERSETUJUAN** ii](#_Toc182388609)

[**LEMBAR PENGESAHAN** iii](#_Toc182388610)

[**UCAPAN TERIMAKASIH** iv](#_Toc182388611)

[**KATA PENGANTAR** vi](#_Toc182388612)

[**BAB I** 1](#_Toc182388613)

[**1.1** **Latar Belakang** 1](#_Toc182388614)

[**1.2** **Rumusan Masalah** 2](#_Toc182388615)

[**1.3** **Tujuan dan Manfaat Penelitian** 2](#_Toc182388616)

[**1.4** **Batasan Masalah** 3](#_Toc182388617)

[**1.5** **Metode Penelitian** 4](#_Toc182388618)

[**1.5.1** **Pengumpulan Data** 4](#_Toc182388619)

[**1.5.2** **Penyesuaian Data** 4](#_Toc182388620)

[**1.6** **Sistematika Penulisan** 4](#_Toc182388621)

[**1.6.1** **BAB I Pendahuluan** 4](#_Toc182388622)

[**1.6.2** **BAB II Tinjauan Pustaka** 4](#_Toc182388623)

[**1.6.3** **BAB III Proses Perancangan** 5](#_Toc182388624)

[**1.6.4** **BAB IV Hasil dan Pembahasan** 5](#_Toc182388625)

[**1.6.5** **BAB V Kesimpulan dan Saran** 5](#_Toc182388626)

[**1.7** **Rencana Kegiatan** 6](#_Toc182388627)

[**BAB II** 7](#_Toc182388628)

[**2.1 PHP** 7](#_Toc182388629)

[**2.1.1** **Definisi PHP** 7](#_Toc182388630)

[**2.2 Visual Studio Code** 8](#_Toc182388631)

[**2.2.1**  **Definisi Visual Studio Code** 8](#_Toc182388632)

[**2.3 Laragon** 10](#_Toc182388633)

[**2.3.1** **Definisi Laragon** 10](#_Toc182388634)

[**2.4 DFD** 11](#_Toc182388635)

[**2.4.1** **Definisi DFD** 11](#_Toc182388636)

[**2.4.2 Simbol DFD** 12](#_Toc182388637)

[**2.5 Draw.io** 14](#_Toc182388638)

[**2.5.1** **Definisi Draw.io** 15](#_Toc182388639)

[**2.6 Bootstrap** 16](#_Toc182388640)

[**2.6.1** **Definisi Bootstrap** 16](#_Toc182388641)

[**2.7 HTML** 17](#_Toc182388642)

[**2.7.1** **Definisi HTML** 17](#_Toc182388643)

[**2.8 CSS** 18](#_Toc182388644)

[**2.8.1** **Definisi CSS** 18](#_Toc182388645)

[**2.9 Java Script** 19](#_Toc182388646)

[**2.9.1** **Definisi Java Script** 20](#_Toc182388647)

[**2.10 Flowchart** 21](#_Toc182388648)

[**2.10.1 Definisi Flowchart** 21](#_Toc182388649)

[**2.10.2 Simbol Flowchart** 22](#_Toc182388650)

[**2.11 ERD** 25](#_Toc182388651)

[**2.11.1 Definisi ERD** 25](#_Toc182388652)

[**2.11.2 Simbol ERD** 26](#_Toc182388653)

[**BAB III** 28](#_Toc182388654)

[**3.1** **DFD** 28](#_Toc182388655)

[**3.1.1** **DFD Level 0** 28](#_Toc182388656)

[**3.1.2** **DFD Level 1** 29](#_Toc182388657)

[**3.2** **ERD** 30](#_Toc182388658)

[**3.3 FLOWCHART** 32](#_Toc182388659)

[**3.4** **DESAIN TABEL** 34](#_Toc182388660)

[**3.5** **Desain Tampilan** 36](#_Toc182388661)

[**3.5.1** **Desain Tampilan Login** 36](#_Toc182388662)

[**3.5.2** **Desain Tampilan Superadmin** 37](#_Toc182388663)

[**3.5.3** **Desain Tampilan Admin** 38](#_Toc182388664)

[**BAB IV** 39](#_Toc182388665)

[**4.1** **Hasil Menjalankan Aplikasi** 39](#_Toc182388666)

[**4.1.1 Tampilan Login** 39](#_Toc182388667)

[**4.1.2 Tampilan Akses Admin** 40](#_Toc182388668)

[**4.1.3 Tampilan Akses Superadmin** 43](#_Toc182388669)

[**BAB V** 47](#_Toc182388670)

[**5.1 Kesimpulan** 47](#_Toc182388671)

[**5.2 Saran** 48](#_Toc182388672)

**DAFTAR GAMBAR**

[**Gambar 2. 1 PHP** 16](#_Toc182291453)

[**Gambar 2. 2 VSCODE** 17](#_Toc182291454)

[**Gambar 2. 3 Laragon** 19](#_Toc182291455)

[**Gambar 2. 4 DFD** 20](#_Toc182291456)

[**Gambar 2. 5 Simbol Terminator** 21](#_Toc182291457)

[**Gambar 2. 6 Simbol Process** 22](#_Toc182291458)

[**Gambar 2. 7 Simbol Data Store** 22](#_Toc182291459)

[**Gambar 2. 8 Simbol Flow** 23](#_Toc182291460)

[**Gambar 2. 9 Draw.io** 23](#_Toc182291461)

[**Gambar 2. 10 Bootstrap** 25](#_Toc182291462)

[**Gambar 2. 11 HTML** 26](#_Toc182291463)

[**Gambar 2. 12 CSS** 27](#_Toc182291464)

[**Gambar 2. 13 Java Script** 28](#_Toc182291465)

[**Gambar 2. 14 Flowchart** 30](#_Toc182291466)

[**Gambar 2. 15 Simbol Flow** 31](#_Toc182291467)

[**Gambar 2. 16 Input/ output** 32](#_Toc182291468)

[**Gambar 2. 17 Decision** 32](#_Toc182291469)

[**Gambar 2. 18 Process** 33](#_Toc182291470)

[**Gambar 2. 19 On-page Reference** 33](#_Toc182291471)

[**Gambar 2. 20 ERD** 34](#_Toc182291472)

[**Gambar 2. 21 Atribut Multinilai** 35](#_Toc182291473)

[**Gambar 2. 22 Relasi** 36](#_Toc182291474)

[**Gambar 2. 23 Entity** 36](#_Toc182291475)

[**Gambar 3. 1 Data flow Diagram Level 0** 38](#_Toc182291987)

[**Gambar 3. 2 DFD Level 1** 39](#_Toc182291988)

[**Gambar 3. 3 Entity Relationship Diagram** 40](#_Toc182291989)

[**Gambar 3. 4 Flowchart Admin** 41](#_Toc182291990)

[**Gambar 3. 5 Flowchart Superadmin** 42](#_Toc182291991)

[**Gambar 3. 6 Desain Tampilan Login** 45](#_Toc182291992)

[**Gambar 3. 7 Desain Tampilan Superadmin** 46](#_Toc182291993)

[**Gambar 3. 8 Desain Tampilan Admin** 47](#_Toc182291994)

[**Gambar 4. 1 Tampilan Login** 48](#_Toc182296373)

[**Gambar 4. 2 Tampilan Dashboard** 49](#_Toc182296374)

[**Gambar 4. 3 Tampilan Order** 50](#_Toc182296375)

[**Gambar 4. 4 Tambah Order** 50](#_Toc182296376)

[**Gambar 4. 5 Tampilan Order Item** 50](#_Toc182296377)

[**Gambar 4. 6 Tambah Item Menu** 51](#_Toc182296378)

[**Gambar 4. 7 Bayar** 51](#_Toc182296379)

[**Gambar 4. 8 Cetak Struk** 51](#_Toc182296380)

[**Gambar 4. 9 Tampilan Superadmin** 52](#_Toc182296381)

[**Gambar 4. 10 Daftar Menu** 53](#_Toc182296382)

[**Gambar 4. 11 Edit Menu** 53](#_Toc182296383)

[**Gambar 4. 12 Tambah Menu** 53](#_Toc182296384)

[**Gambar 4. 13 Hapus Menu** 54](#_Toc182296385)

[**Gambar 4. 14 Kategori Menu** 54](#_Toc182296386)

[**Gambar 4. 15 Tambah Kategori Menu** 55](#_Toc182296387)

[**Gambar 4. 16 Edit Kategori Menu** 55](#_Toc182296388)

**DAFTAR TABEL**

[**Tabel 1.1 Rencana Kegiatan** 15](#_Toc182249267)

[**Tabel 3. 1 Tabel User** 42](#_Toc182249277)

[**Tabel 3. 2 Daftar Menu** 42](#_Toc182249278)

[**Tabel 3. 3 Kategori Menu** 43](#_Toc182249279)

[**Tabel 3. 4 Order** 43](#_Toc182249280)

[**Tabel 3. 5 List Order** 43](#_Toc182249281)

[**Tabel 3. 6 Bayar** 44](#_Toc182249282)

# **BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang**

Di era digital yang semakin maju, pemanfaatan teknologi informasi menjadi kunci utama dalam mentransformasi berbagai sektor kehidupan, termasuk dalam upaya pelestarian lingkungan. Kesadaran masyarakat akan pentingnya pengelolaan limbah yang terintegrasi kini menjadi perhatian serius seiring dengan meningkatnya volume sampah rumah tangga. Proses birokrasi dan akses yang mudah bagi masyarakat untuk berpartisipasi dalam menjaga kebersihan lingkungan sangat dibutuhkan untuk menciptakan ekosistem hidup yang lebih sehat dan berkelanjutan..

Masalah sampah rumah tangga seringkali menjadi kendala besar akibat kurangnya sarana pembuangan yang efektif dan sistem penjemputan yang belum terorganisir dengan baik. Banyak masyarakat yang memiliki niat untuk memilah sampah, namun terkendala oleh jarak menuju tempat pembuangan akhir atau ketidakpastian jadwal petugas kebersihan. Fenomena ini menyebabkan tumpukan sampah liar yang berdampak buruk bagi kesehatan dan estetika lingkungan. Padahal, jika dikelola dengan sistem yang tepat, sampah rumah tangga seperti anorganik memiliki nilai ekonomis yang dapat memberikan keuntungan timbal balik bagi masyarakat melalui sistem *reward*.

Oleh karena itu, penulis berencana untuk membangun aplikasi pengelolaan sampah berbasis web bernama RECYLOOP dengan menggunakan teknologi Next.js, Tailwind CSS, dan Supabase. Aplikasi ini dirancang untuk menghubungkan warga, kurir, dan admin secara *real-time*, sehingga proses penjemputan sampah menjadi lebih transparan, efisien, dan mampu meningkatkan partisipasi masyarakat melalui konversi sampah menjadi poin penghargaan yang dapat ditukarkan.

* 1. **Rumusan Masalah**

Berdasarkan latar belakang yang telah dijelaskan di atas, maka dapat dirumuskan beberapa masalah penting sebagai berikut:

1. Bagaimana cara merancang dan membangun aplikasi pengelolaan sampah berbasis web RECYLOOP menggunakan *framework* Next.js, Tailwind CSS, dan Supabase?
2. Bagaimana mengimplementasikan sistem koordinasi *real-time* antara warga (User) dan petugas lapangan (Kurir) dalam proses penjemputan sampah?
3. Bagaimana merancang mekanisme konversi berat sampah menjadi poin *reward* yang transparan dan akurat bagi pengguna?
4. Bagaimana sistem dapat memfasilitasi Admin dalam melakukan validasi data dan monitoring laporan pengelolaan sampah secara menyeluruh?
   1. **Tujuan dan Manfaat Penelitian**

Tujuan dari dibuatnya aplikasi ini adalah :

1. Membangun sistem informasi pengelolaan sampah berbasis web yang responsif dan modern menggunakan Next.js dan Supabase**.**
2. Menyediakan platform yang memudahkan warga dalam melakukan permintaan penjemputan sampah secara terjadwal dan efisien.
3. Mengotomatisasi proses penghitungan poin *reward* berdasarkan berat sampah untuk memotivasi masyarakat dalam memilah sampah.
4. Memfasilitasi admin dan kurir dengan alat monitoring yang transparan dalam mengelola data setoran sampah dan validasi di lapangan.

Manfaat dari dibuatnya aplikasi ini adalah :

1. Bagi Masyarakat: Mempermudah proses penyaluran sampah anorganik tanpa harus keluar rumah dan mendapatkan keuntungan ekonomis berupa poin.
2. Bagi Petugas (Kurir): Membantu mengorganisir rute penjemputan berdasarkan data lokasi (GPS) yang akurat dari pengguna sehingga menghemat waktu dan biaya operasional.
3. Bagi Lingkungan: Mendorong terciptanya budaya pilah sampah dari sumbernya (rumah tangga) yang berkontribusi langsung pada pengurangan beban sampah di TPA.
4. Bagi Akademik: Sebagai implementasi penggunaan teknologi *Real-time Database* dan *Framework* modern dalam memecahkan masalah sosial di lingkungan sekitar.
   1. **Batasan Masalah**

Agarpembahasan dalam laporan ini lebih terarah dan tidak menyimpang dari tujuan yang telah ditetapkan, maka penulis memberikan batasan masalah sebagai berikut:

1. Sistem ini merupakan aplikasi berbasis web yang memerlukan koneksi internet stabil untuk mengakses seluruh fitur dan sinkronisasi data secara *real-time*.
2. Pengelolaan sampah pada aplikasi ini dibatasi hanya untuk jenis sampah anorganik (seperti plastik, kertas, dan logam) yang memiliki nilai untuk didaur ulang.
3. Fitur penjemputan sampah (*pickup*) hanya berlaku untuk wilayah cakupan tertentu yang telah ditentukan oleh Admin dan didukung oleh ketersediaan Kurir di wilayah tersebut.
4. Sistem pembayaran atau penukaran poin tidak mencakup transaksi perbankan secara langsung di dalam web, melainkan hanya simulasi konversi poin menjadi nilai saldo yang dapat divalidasi oleh Admin.
5. Perancangan aplikasi ini menggunakan *framework* Next.js, Tailwind CSS, dan Supabase sebagai *backend-as-a-service.*
   1. **Metode Penelitian**
      1. **Pengumpulan Data**

Dalam upaya mengumpulkan data dan informasi yang akurat untuk mendukung pengembangan aplikasi RECYLOOP, penulis menggunakan beberapa teknik pengumpulan data sebagai berikut:

1. Studi Pustaka (*Library Research*):Penulis melakukan pencarian dan mempelajari literatur, artikel kimiah serta dokumentasi teknis yang berkaitan dengan pengembangan web menggunakan Next.js dan manajemen basis data menggunakan Supabase.
2. Observasi Berbasis Internet (Online Research):Penulis melakukan analisis terhadap platform pengelolaan sampah yang sudah ada (seperti Mallsampah dan Banksampah.id) sebagai referensi dalam menentukan fitur, alur kerja (workflow), serta antarmuka pengguna (UI/UX) yang efektif.
3. Dokumentasi**:** Mengumpulkan data mengenai klasifikasi jenis sampah anorganik dan standar konversi nilai ekonomi sampah untuk dijadikan dasar perhitungan poin dalam system.
   * 1. **Perancangan Sistem**

Tahap perancangan dilakukan secara mendalam untuk memastikan aplikasi RECYLOOP memiliki struktur yang solid dan antarmuka yang intuitif. Proses ini mencakup pembuatan desain antarmuka (*UI/UX Design*) menggunakan Figma untuk menghasilkan *high-fidelity prototype* bagi seluruh peran pengguna, penyusunan diagram alir (*Flowchart*) dan *Data Flow Diagram* (DFD) guna memetakan pergerakan data di dalam sistem, serta perancangan skema basis data pada Supabase yang mendefinisikan relasi antar tabel untuk profil pengguna, riwayat setoran sampah, hingga sistem manajemen *reward*.

* + 1. **Implementasi Sistem**

Tahap implementasi dilakukan dengan menerjemahkan rancangan antarmuka dari Figma ke dalam kode program menggunakan framework Next.js dan Tailwind CSS untuk menghasilkan *frontend* yang responsif. Proses ini dilanjutkan dengan integrasi *backend* menggunakan Supabase sebagai basis data dan layanan autentikasi, yang mencakup pembuatan logika CRUD untuk manajemen setoran sampah serta penerapan *Role-Based Access Control* (RBAC) guna memastikan pembatasan hak akses dan sinkronisasi data poin secara *real-time* sesuai dengan peran masing-masing pengguna (User, Kurir, dan Admin).

* + 1. **Pengujian Sistem**

Tahap pengujian dilakukan menggunakan metode Black Box Testing untuk memvalidasi fungsionalitas aplikasi RECYLOOP secara menyeluruh tanpa harus melihat detail kode program. Proses ini berfokus pada pengujian skenario utama seperti keberhasilan pendaftaran akun, akurasi kalkulasi poin berdasarkan berat sampah yang diinput kurir, serta memastikan sistem keamanan hak akses (*role-access*) antara User, Kurir, dan Admin berjalan sesuai dengan kebutuhan fungsional yang telah direncanakan.

* 1. **Sistematika Penulisan**

**BAB I Pendahuluan**

Pendahuluan berisi tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metode penelitian yang digunakan, serta sistematika penulisan laporan.

**BAB II Tinjauan Pustaka**

Menjelaskan tentang teori-teori yang mendukung dan yang digunakan dalam system.

**BAB III Proses Perancangan**

Berisi tentang rancangan system dan desain yang berada dalam system.

**BAB IV Hasil dan Pembahasan**

Mengimplementasikan semua data dan menyusun rancangan sistem dari *web*.

**BAB V Kesimpulan dan Saran**

Membuat kesimpulan dan mempertimbangkan saran yang ada.

* 1. **Rencana Kegiatan**

**Tabel 1.1 Rencana Kegiatan**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Kegiatan** | **Minggu Ke** | | | | | | | | | | | |
| Januari | | | | Februari | | | | Maret | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| Pengajuan Judul |  |  |  |  |  |  |  |  |  |  |  |  |
| Laporan TA |  |  |  |  |  |  |  |  |  |  |  |  |
| Bab 1 |  |  |  |  |  |  |  |  |  |  |  |  |
| Bab 2 |  |  |  |  |  |  |  |  |  |  |  |  |
| Bab 3 |  |  |  |  |  |  |  |  |  |  |  |  |
| Bab 4 |  |  |  |  |  |  |  |  |  |  |  |  |
| Bab 5 |  |  |  |  |  |  |  |  |  |  |  |  |
| Pengerjaan Project |  |  |  |  |  |  |  |  |  |  |  |  |

# **BAB II**

**TINJAUAN PUSTAKA**

## **2.1 Next.JS**



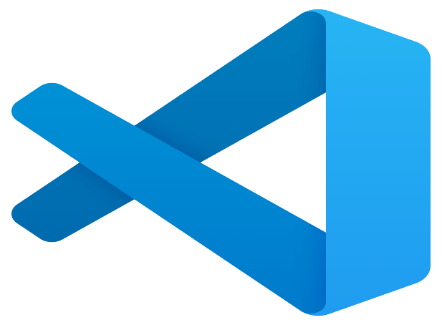
**Gambar 2. 1 Next.JS**

( Sumber : [https://nextjs.org/docsl](https://nextjs.org/docs) )

Next.js adalah sebuah *framework* berbasis React yang dikembangkan oleh Vercel untuk membangun aplikasi web yang sangat cepat dan ramah SEO (*Search Engine Optimization*). Next.js memungkinkan pengembang untuk melakukan *Server-Side Rendering* (SSR) dan *Static Site Generation* (SSG), yang berarti halaman web dapat dimuat lebih cepat karena proses perenderan dilakukan di sisi server sebelum sampai ke browser pengguna.

Berbeda dengan pengembangan web tradisional, Next.js menyediakan fitur-fitur modern seperti *file-based routing* yang memudahkan pengaturan halaman, serta optimalisasi gambar otomatis yang sangat mendukung performa aplikasi. Dengan menggunakan Next.js, pengembangan aplikasi RECYLOOP menjadi lebih efisien karena didukung oleh ekosistem JavaScript yang luas serta kemudahan dalam integrasi dengan berbagai layanan *backend* modern seperti Supabase.

## **2.2 Visual Studio Code**



**Gambar 2. 2 VSCODE**

(Sumber : <https://id.wikipedia.org/wiki/Visual_Studio_Code> )

Visual Studio Code adalah aplikasi code editor buatan Microsoft yang dapat dijalankan di semua perangkat desktop secara gratis. Kelengkapan fitur dan ekstensi membuat *code editor* ini menjadi pilihan utama para pengembang. Visual Studio Code bahkan mendukung hampir semua sistem operasi seperti *Windows, Mac OS, Linux*, dan lain sebagainya.

Berdasarkan survey dari *Stack Overflow*, Visual Studio Code merupakan editor terpopuler di kalangan developer profesional. Dari 21 aplikasi text editor saingannya, Visual Studio Code berada di peringkat satu dengan persentase user mencapai 71.07%.

Bukan tanpa alasan, Visual Studio Code dibuat se-ringan dan se-nyaman mungkin sehingga pengguna tidak terlalu membutuhkan perangkat berspesifikasi tinggi. Aplikasi ini juga bisa dijalankan untuk membuat atau mengedit kode sumber berbagai programming language. Sebut saja seperti *Node.js*, *JavaScript*, *TypeScript,* dan masih banyak lagi.

## **2.3 Tailwind CSS**



**Gambar 2. 3 Tailwind CSS**

(Sumber : <https://tailwindcss.com/docs/installation/using-vite> )

Tailwind CSS adalah sebuah *framework* CSS yang menggunakan pendekatan *utility-first* untuk membangun antarmuka pengguna secara cepat tanpa harus meninggalkan file HTML atau komponen JavaScript. Berbeda dengan *framework* seperti Bootstrap yang menyediakan komponen siap pakai (seperti tombol atau kartu), Tailwind CSS menyediakan kelas-kelas utilitas tingkat rendah (*low-level*) seperti *flex*, pt-4, *text-center*, dan *rotate*-90 yang dapat dikombinasikan untuk membangun desain kustom yang unik dan fleksibel.

Penggunaan Tailwind CSS dalam proyek RECYLOOP memberikan keuntungan berupa ukuran file CSS yang sangat kecil dan performa yang optimal karena hanya kelas yang digunakan saja yang akan disertakan dalam hasil akhir produksi. Selain itu, Tailwind memudahkan pembuatan desain yang responsif di berbagai ukuran layar, mulai dari perangkat *mobile* hingga desktop, hanya dengan menggunakan prefiks kelas tertentu, sehingga mempercepat proses pengembangan antarmuka yang modern dan konsisten.

## **2.4 Supabase**



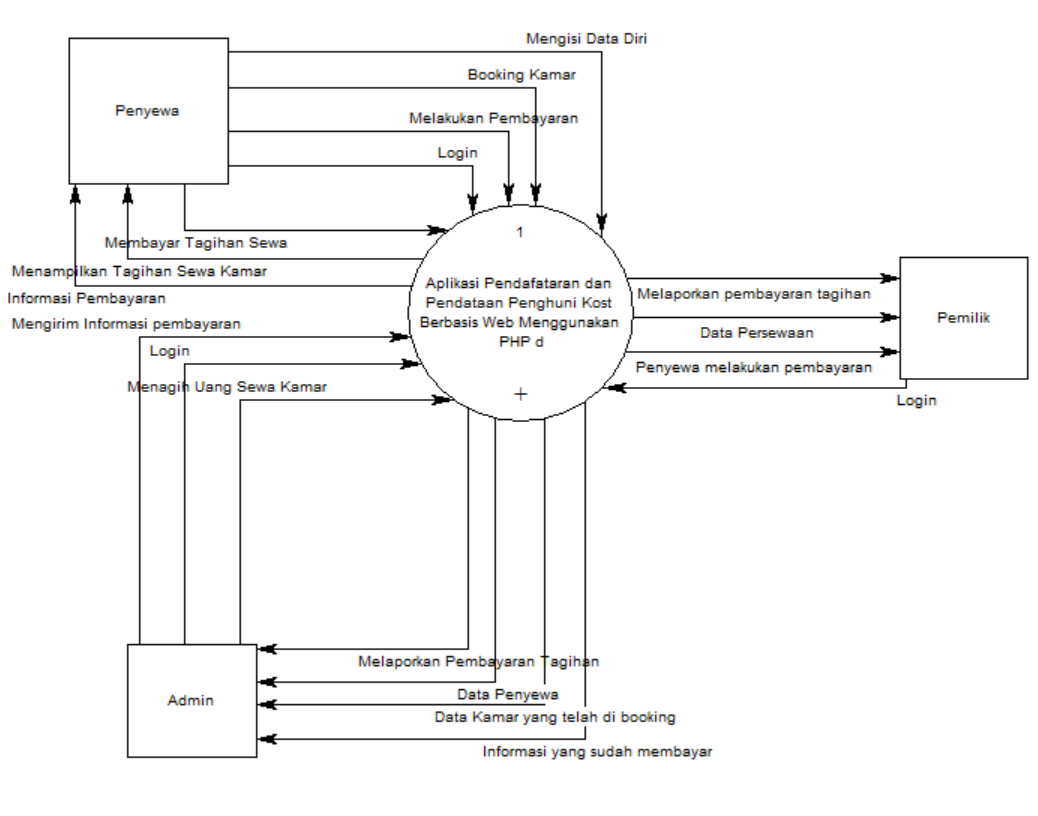
**Gambar 2. 4 Supabase**

Sumber : <https://supabase.com/docs> )

Supabase adalah platform *open-source* yang menyediakan berbagai layanan *backend* untuk mempercepat pengembangan aplikasi tanpa harus membangun infrastruktur server secara manual. Sering disebut sebagai alternatif dari Firebase, Supabase dibangun di atas PostgreSQL, salah satu sistem basis data relasional paling kuat di dunia. Layanan utama yang disediakan oleh Supabase meliputi penyimpanan data (*Database*), sistem autentikasi pengguna (*Auth*), penyimpanan file (*Storage*), hingga fungsi *Edge Functions* yang berjalan secara otomatis.

Dalam pengembangan aplikasi RECYLOOP, Supabase berperan penting dalam menangani fitur *real-time*, di mana perubahan data seperti status penjemputan sampah atau penambahan poin dapat langsung diperbarui di layar pengguna tanpa perlu memuat ulang halaman. Selain itu, fitur autentikasinya memudahkan penulis dalam mengelola keamanan akun serta pembatasan hak akses antara User, Kurir, dan Admin secara terintegrasi.

## **2.5 DFD**

****

**Gambar 2. 5 DFD**

### **2.5.1 Definisi DFD**

Diagram konteks adalah diagram tingkat tinggi, artinya diagram tersebut tidak membahas seluk-beluk sistem secara terperinci. Sebaliknya, diagram tersebut memetakan keseluruhan sistem dengan cara yang sederhana, jelas, dan mudah dipahami.

Dalam diagram alir data juga tidak mempunyai kontrol terhadap *flow*-nya, sehingga tidak adanya aturan terkait keputusan atau pengulangan. Bentuk penggambaran berupa data *flowchart* dengan skema yang lebih spesifik. Menurut Kenneth Kozar, tujuan dari adanya DFD sendiri adalah sebagai penyedia atau menjembatani antara pengguna dengan sistem.

*Data flow diagram*berbeda dengan UML (*Unified Modelling Language*), dimana hal mendasar yang menjadi pembeda antara kedua skema tersebut terletak pada *flow*dan *objective*penyampaian informasi di dalamnya.

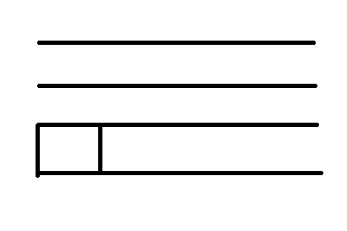
### **2.5.2 Simbol DFD**

**Gambar 2. 6 Simbol Terminator**

**Terminator**: Kesatuan diluar sistem (external entity) yang memberikan input ke sistem atau menerima *output* dari sistem berupa data.

**Gambar 2. 7 Simbol Process**

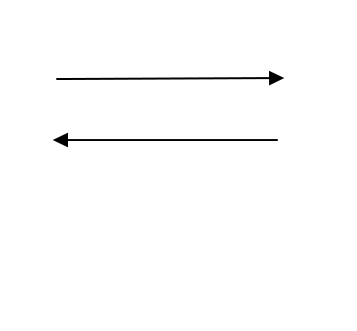
**Process**: Aktivitas yang mengolah input menjadi output.



**Gambar 2. 8 Simbol Data Store**

(Sumber: <https://www.nesabamedia.com/pengertian-dfd/> )

**Data Store**: Penyimpanan data pada database, biasanya berupa tabel.



**Gambar 2. 9 Simbol Flow**

(Sumber: <https://www.nesabamedia.com/pengertian-dfd/> )

**Data Flow**: Aliran data pada sistem (antar proses, antara terminator & proses, serta antara proses & data store).

## **2.6 Draw.io**



**Gambar 2. 10 Draw.io**

**(Sumber :** <https://www.pinterest.com/pin/1034139133154653802/> **)**

Draw.io adalah perangkat lunak gratis untuk menggambar grafik yang dapat digunakan untuk membuat berbagai jenis diagram, seperti diagram alir, *wireframe*, diagram UML, bagan organisasi, dan diagram jaringan. Draw.io juga dikenal sebagai diagrams.net.

Draw.io dapat digunakan untuk membuat desain antarmuka, seperti diagram *Unified Modeling Language* (UML). Diagram UML dapat membantu pengguna membuat model dan desain sistem yang kuat dan mudah dipahami.

Meskipun tidak ada aplikasi Draw.io, pengguna dapat menambahkan Draw.io ke layar beranda tablet Android atau iPad. Setelah ditambahkan, Draw.io dapat dibuka seolah-olah merupakan aplikasi asli.

## **2.7 Typescript**



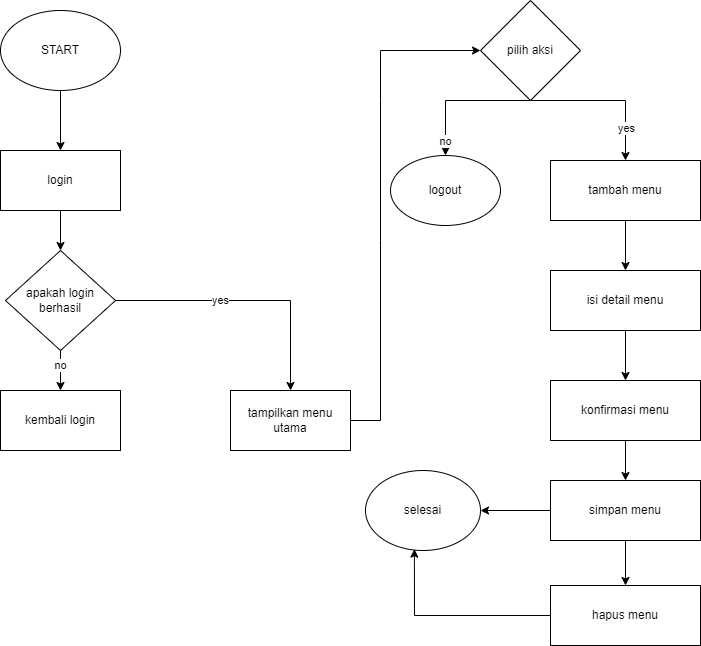
**Gambar 2. 11 Typescript**

(Sumber: <https://www.typescriptlang.org/docs/> )

TypeScript adalah bahasa pemrograman berbasis *open-source* yang dikembangkan oleh Microsoft sebagai pengembangan dari JavaScript dengan menambahkan fitur *static typing*. Sering disebut sebagai *superset* dari JavaScript, artinya setiap kode JavaScript yang valid juga merupakan kode TypeScript yang valid. Fitur utama TypeScript adalah kemampuannya untuk mendefinisikan tipe data secara eksplisit, yang membantu pengembang dalam mendeteksi kesalahan penulisan kode (*bug*) lebih dini pada saat proses pengembangan sebelum aplikasi dijalankan.

Dalam pengembangan aplikasi RECYLOOP, penggunaan TypeScript sangat krusial untuk memastikan konsistensi data yang mengalir dari backend Supabase ke antarmuka pengguna. Dengan adanya *interface* dan *type*, penulis dapat meminimalisir kesalahan logika pada fitur-fitur kompleks seperti perhitungan poin sampah dan manajemen status penjemputan. Selain itu, fitur *IntelliSense* pada TypeScript meningkatkan efisiensi pengodingan melalui saran pelengkapan kode otomatis yang akurat.

## **2.8 Flowchart**

****

**Gambar 2. 12 Flowchart**

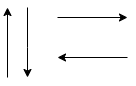
**2.8.1 Definisi Flowchart**

Flowchart atau bagan alur adalah diagram yang menampilkan langkah-langkah dan keputusan untuk melakukan sebuah proses dari suatu program. Setiap langkah digambarkan dalam bentuk diagram dan dihubungkan dengan garis atau arah panah.

Flowchart berperan penting dalam memutuskan sebuah langkah atau fungsionalitas dari sebuah proyek pembuatan program yang melibatkan banyak orang sekaligus.

Selain itu dengan menggunakan bagan alur proses dari sebuah program akan lebih jelas, ringkas, dan mengurangi kemungkinan untuk salah penafsiran. Penggunaan flowchart dalam dunia pemrograman juga merupakan cara yang bagus untuk menghubungkan antara kebutuhan teknis dan non-teknis.

### **2.8.2 Simbol Flowchart**



**Gambar 2. 13 Simbol Flow**

Simbol ini digunakan untuk menghubungkan bagian-bagian flowchart yang terpisah. Biasanya direpresentasikan dengan garis lurus atau panah.



**Gambar 2. 14 Input/ output**

Simbol ini digunakan untuk menunjukkan input atau output data dalam proses. Biasanya direpresentasikan dengan bentuk paralelogram.



**Gambar 2. 15 Decision**

Simbol ini menunjukkan titik keputusan dalam alur kerja yang memerlukan pilihan ya atau tidak. Biasanya direpresentasikan dengan bentuk diamond.

****

**Gambar 2. 16 Process**

Flowchart proses adalah cara penggambaran rekayasa industrial dengan cara merinci dan menganalisis langkah-langkah selanjutnya dalam suatu prosedur atau sistem.Simbol yang menyatakan suatu proses yang di lakukan oleh komputer

****

**Gambar 2. 17 On-page Reference**

Simbol untuk keluar – masuk atau penyambungan proses dalam lembar kerja yang sama.

## **2.9 ERD**

**Gambar 2. 18 ERD**

**2.9.1 Definisi ERD**

ERD adalah salah satu bentuk diagram yang digunakan untuk keperluan bisnis dalam proses visualisasi atau gambaran tentang objek maupun data yang berhubungan dengan bisnis yang terbilang cukup rumit.

Adanya diagram tersebut diharapkan dapat memberikan pemahaman dan memberikan kemudahan dalam upaya pengembangan proyek bisnis yang hendak dilakukan. Selain itu adanya diagram ER juga dapat membantu supaya database system dapat terlihat lebih rapi dan terorganisir.

Pada umumnya diagram ER atau Entity Relationship Diagram kerap digunakan oleh perusahaan-perusahaan untuk memberikan kemudahan dalam mengelola dan mengatur data-data yang dimiliki supaya lebih keterkaitan antara entitas satu dan lainnya dapat dengan mudah dipahami dan dimengerti.

**2.9.2 Simbol ERD**



**Gambar 2. 19 Atribut Multinilai**

Atribut multinilai dapat memiliki lebih dari satu nilai. Misalnya, entitas karyawan dapat memiliki beberapa nilai keterampilan.



**Gambar 2. 20 Relasi**

Relasi adalah Tindakan, yang diwakili oleh bentuk intan, menunjukkan bagaimana dua entitas berbagi informasi dalam database.



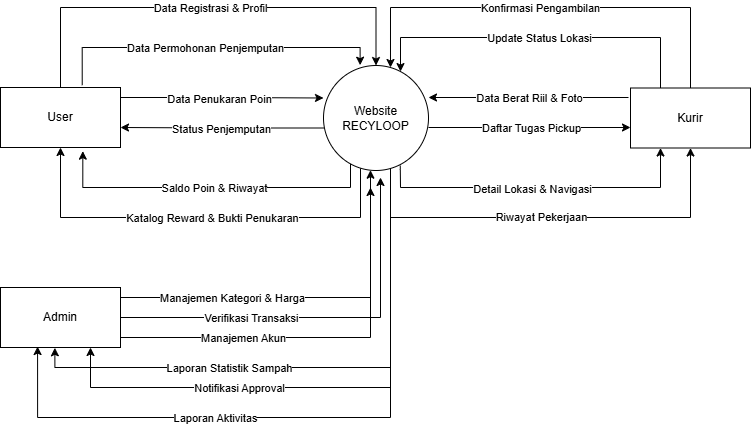
**Gambar 2. 21 Entity**

Entitas, yang diwakili oleh persegi panjang. Entitas adalah objek atau konsep yang ingin Anda simpan informasinya.

# **BAB III**

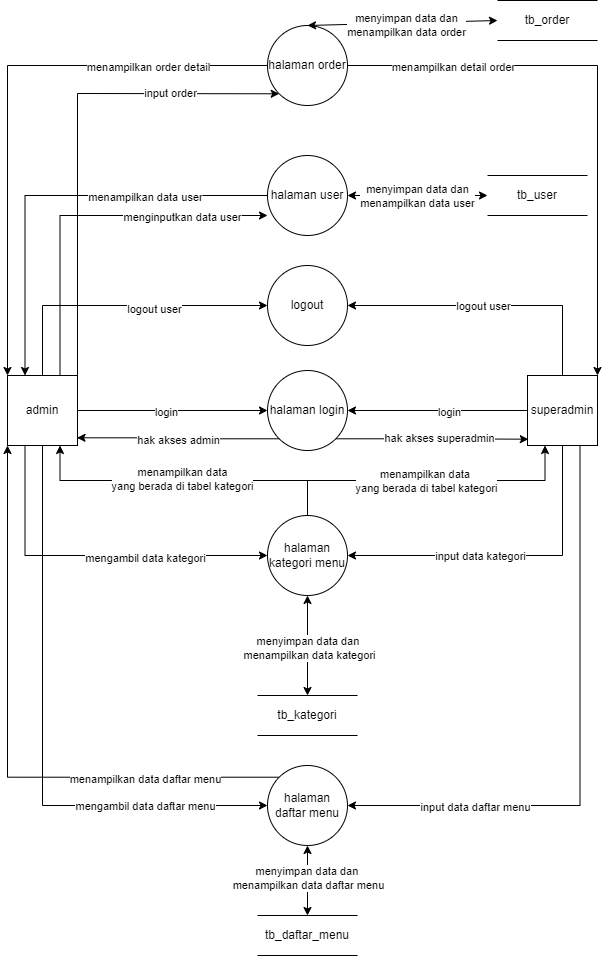
**PROSES PERANCANGAN**

* 1. **DFD**
     1. **DFD Level 0**

****

**Gambar 3. 1 Data flow Diagram Level 0**

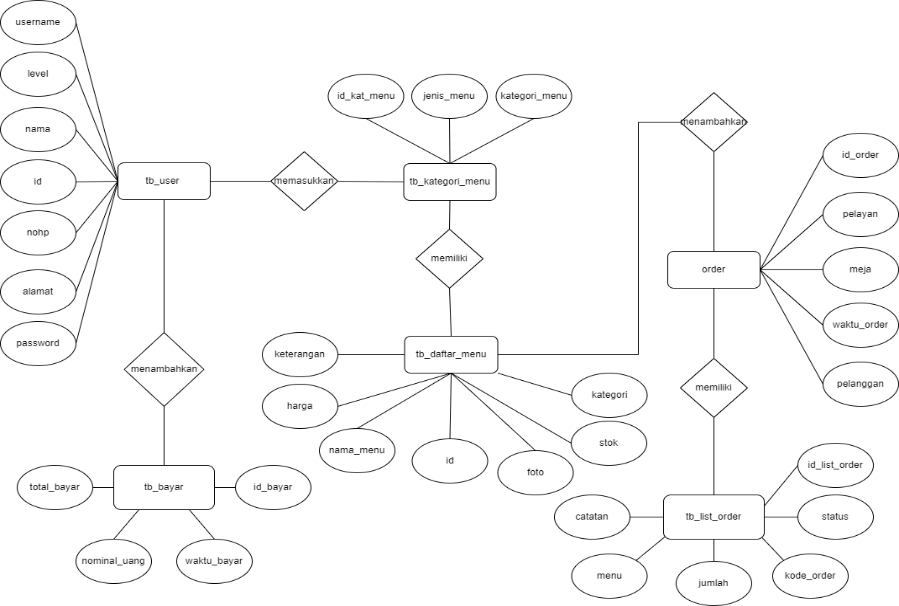
Pada DFD level 0 saya menggunakan 3 entitas yaitu User, Admin, dan Kurir dan Gambar 3.1 diatas beserta juga alur kerja dari proses tersebut.

* + 1. **DFD Level 1**

**Gambar 3. 2 DFD Level 1**

Pada DFD level 1 Alur data dalam sistem ini bergerak dari pengguna (admin dan superadmin) ke halaman-halaman utama (login, kategori menu, daftar menu, order, dan logout), dengan penyimpanan dan pengambilan data dari beberapa tabel basis data (tb\_user, tb\_kategori, tb\_daftar\_menu, tb\_order). Sistem ini juga memiliki mekanisme logout untuk mengakhiri sesi akses.

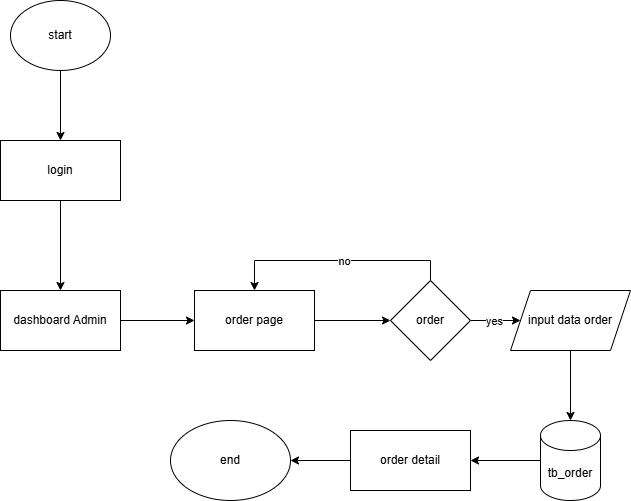
* 1. **ERD**

****

**Gambar 3. 3 Entity Relationship Diagram**

ERD ini menggambarkan hubungan antar-entitas utama dalam sistem, seperti pengguna, kategori menu, subkategori, daftar menu, pembeli, order, dan detail order. Diagram ini membantu dalam memahami struktur data dan alur pengelolaan data di dalam sistem, serta memberikan panduan bagi pengembangan basis data untuk mendukung fungsi sistem secara keseluruhan.

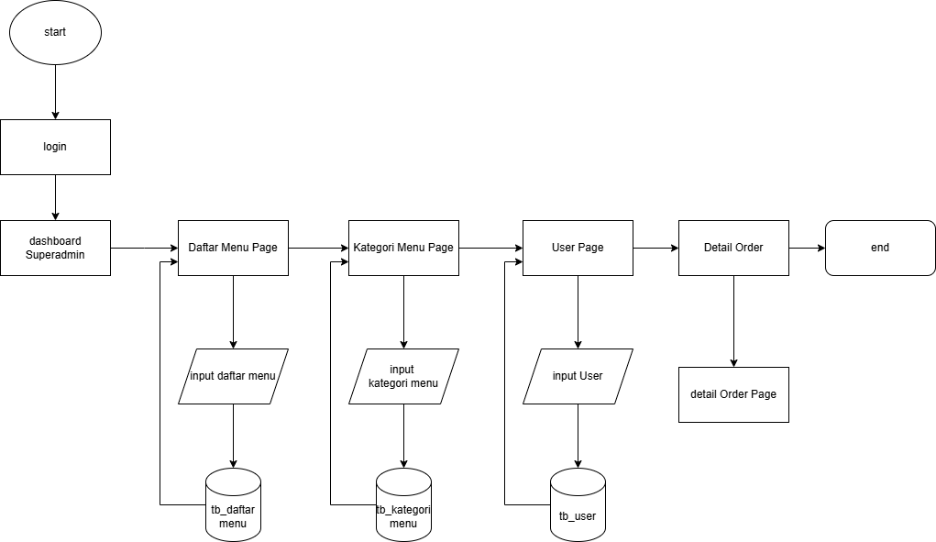
## **3.3 FLOWCHART**

**3.3.1 FLOWCHART ADMIN**

**Gambar 3. 4 Flowchart Admin**

Flowchart ini menggambarkan alur kerja admin dalam mengelola sistem, mulai dari login, pengecekan akun, input data diri, hingga mengelola pesanan di *order page*. Flowchart ini membantu menggambarkan langkah-langkah yang harus diikuti admin untuk mengakses dan mengelola data dalam sistem dengan alur yang jelas.

**3.3.2 FLOWCHART SUPERADMIN**



**Gambar 3. 5 Flowchart Superadmin**

Flowchart ini memperlihatkan bahwa Superadmin memiliki akses penuh terhadap fitur-fitur penting seperti pengelolaan penawaran, kategori menu, pengguna, laporan, dan detail pesanan. Dengan mengikuti alur dari flowchart ini, Superadmin dapat mengoptimalkan pengelolaan sistem secara efisien.

* 1. **DESAIN TABEL**

**Tabel 3. 1 Tabel User**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** | **Atribut** |
| Id | int (5) | Primary Key | AUTO\_INCREMENT |
| Nama | Varchar(100) | Not Null |  |
| Username | varchar (100) | Not Null |  |
| Password | Varchar (100) | Not Null |  |
| Level | int (1) | Not Null |  |
| Nohp | Int(15) | Not Null |  |
| Alamat | Varchar(100) | Not Null |  |

**Tabel 3. 2 Daftar Menu**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** | **Atribut** |
| Id | int (10) | Primary Key | AUTO\_INCREMENT |
| Foto | varchar (200) | Not Null |  |
| Nama\_menu | Varchar(100) | Not Null |  |
| Keterangan | varchar(100) | Not Null |  |
| Kategori | Varchar(200) | Not Null |  |
| Harga | Int(100) | Not Null |  |
| Stok | Varchar(50) | Not Null |  |

**Tabel 3. 3 Kategori Menu**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** | **Atribut** |
| Id\_kat\_menu | int (5) | Primary Key | AUTO\_INCREMENT |
| jenis\_menu | Varchar (50) | Not Null |  |
| Kategori\_menu | varchar(100) | Not Null |  |

**Tabel 3. 4 Order**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** | **Atribut** |
| id\_order | int (10) | Primary Key | AUTO\_INCREMENT |
| Pelanggan | Varchar (200) | Not Null |  |
| Meja | int (10) | Not Null |  |
| Pelayan | int(10) | Not Null |  |
| Waktu\_order | timestamp | Not Null |  |

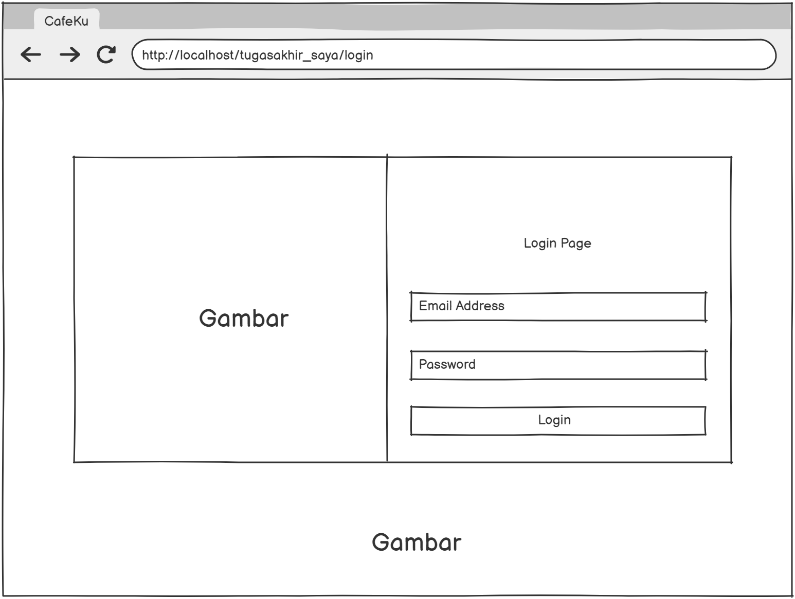
**Tabel 3. 5 List Order**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** | **Atribut** |
| **id\_list\_order** | **int(10)** | Primary Key | AUTO\_INCREMENT |
| **Menu** | **int(10)** | Not Null |  |
| **Kode\_order** | **int(19)** | Not Null |  |
| **Jumlah** | **int(10)** | Not Null |  |
| **Catatan** | **Varchar (300)** | Not Null |  |
| **Status** | **int(10)** | Not Null |  |

**Tabel 3. 6 Bayar**

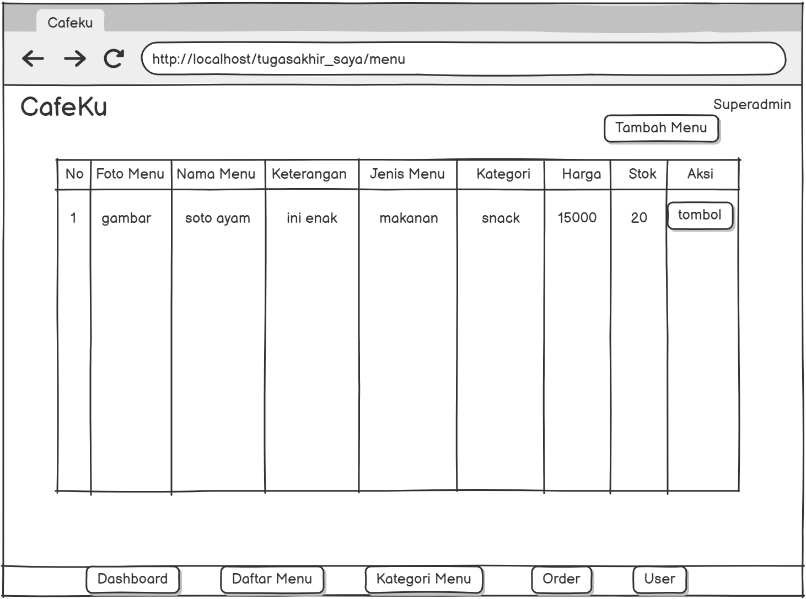
|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** | **Atribut** |
| **id\_bayar** | **Bigint(19)** | Primary Key | AUTO\_INCREMENT |
| **Nominal\_uang** | **Bigint(19)** | Not Null |  |
| **Total\_bayar** | **Bigint(19)** | Not Null |  |
| **Waktu\_bayar** | **timestamp** | Not Null |  |

* 1. **Desain Tampilan** 
     1. **Desain Tampilan Login**

****

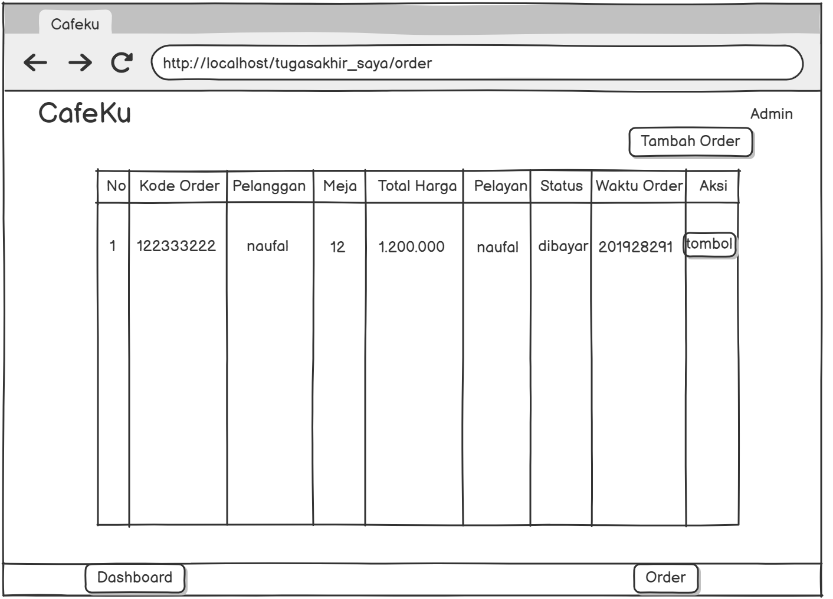
**Gambar 3. 6 Desain Tampilan Login**

* + 1. **Desain Tampilan Superadmin**

****

**Gambar 3. 7 Desain Tampilan Superadmin**

* + 1. **Desain Tampilan Admin**

****

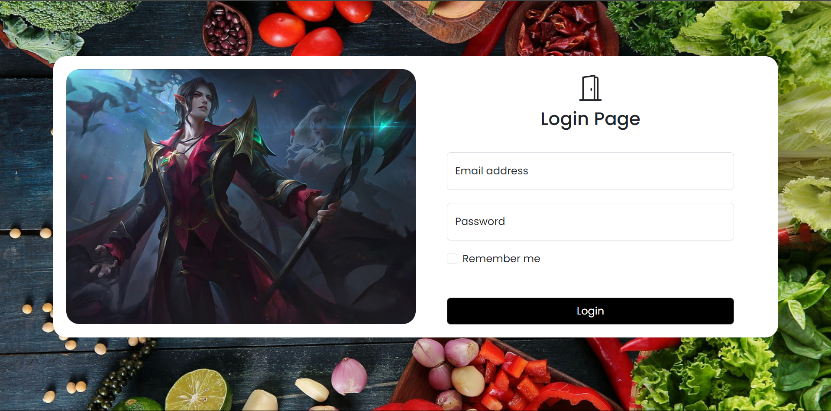
**Gambar 3. 8 Desain Tampilan Admin**

**BAB IV**

**HASIL DAN PEMBAHASAN**

* 1. **Hasil Menjalankan Aplikasi**

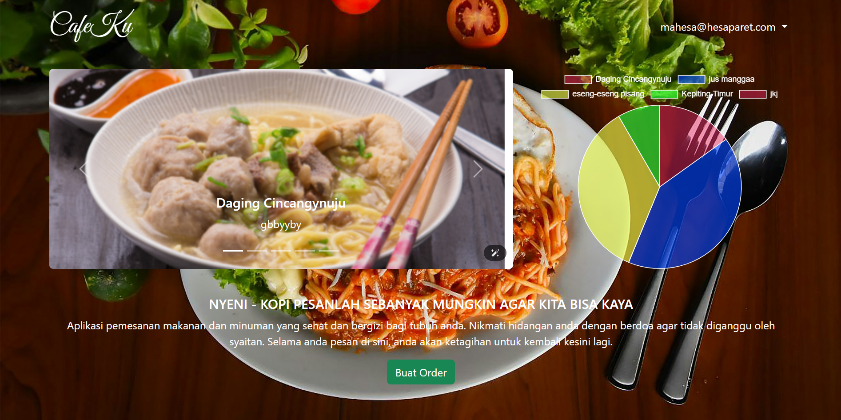
### **4.1.1 Tampilan Login**

**** Tampilan login merupakan halaman *website* yang harus di lewati oleh seluruh user, untuk mendefinisikan akses yang dapat masuk kedalaman halaman sesuai hak akses. Dan sudah memiliki akun yang telah terdaftar pada system.

**Gambar 4. 1 Tampilan Login**

### **4.1.2 Tampilan Akses Admin**

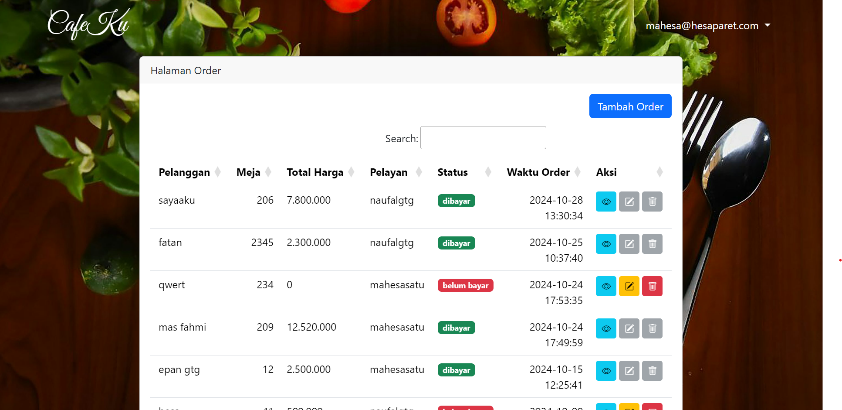
**4.1.2.1 Dashboard**

Tampilan Dashboard Admin ini menampilkan menu-menu makanan/minuman yang tersedia di dalam cafe ini, dan juga jumlah data makanan/minuman yang sudah di pesan, disini Admin hanya bisa mengakses order.

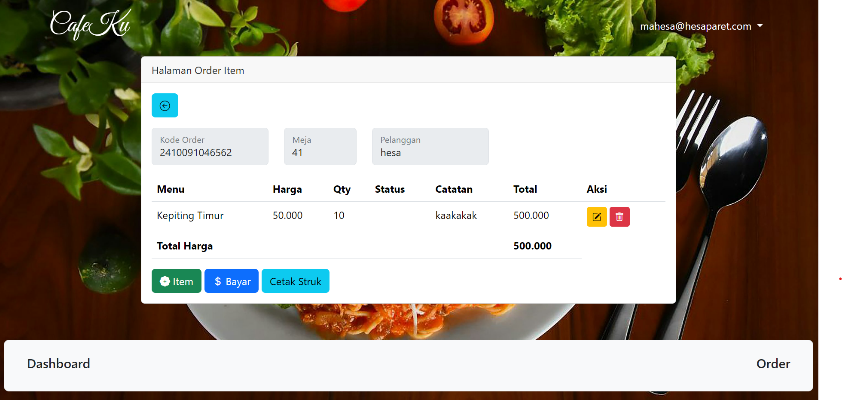
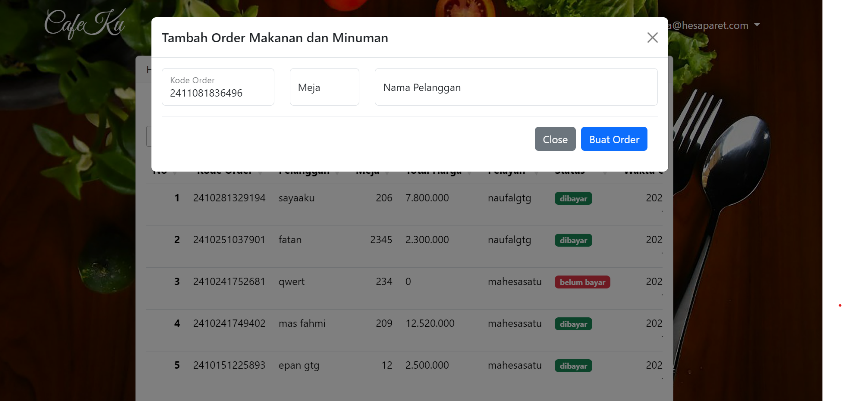
**Gambar 4. 2 Tampilan Dashboard**

**4.1.2.2 Order**

Tampilan Order menampilkan orderan dari para pelanggan, disini admin bisa menambah orderan dari pelanggan dan melihat siapa saja orderan yang belum di bayar maupun sudah di bayar, juga Admin bisa mengetahui menu-menu makanan/minuman apa yang sudah di pesan oleh pelanggan di halaman order item jika menekan tombol view dan bisa menambah item, bayar juga cetak struk.

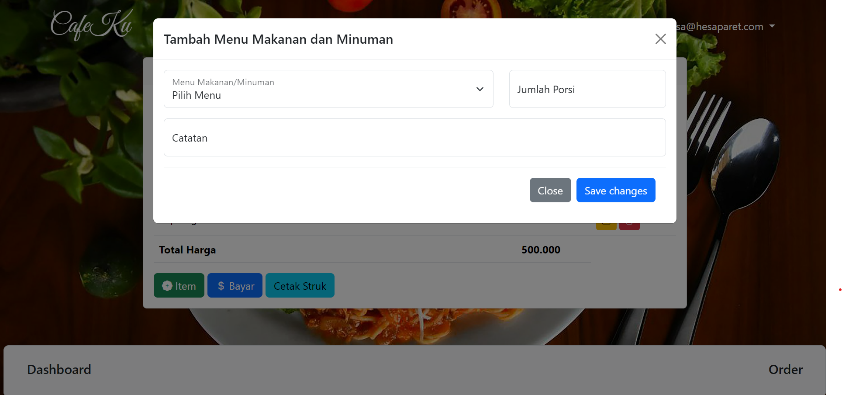


**Gambar 4. 3 Tampilan Order**

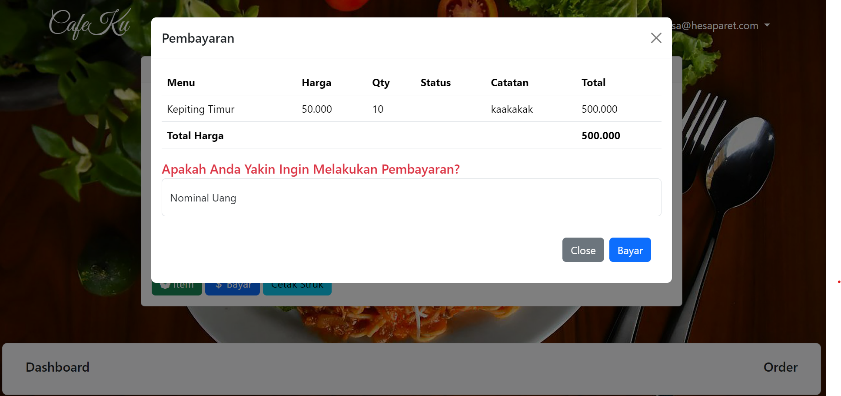
****

**Gambar 4. 4 Tambah Order**

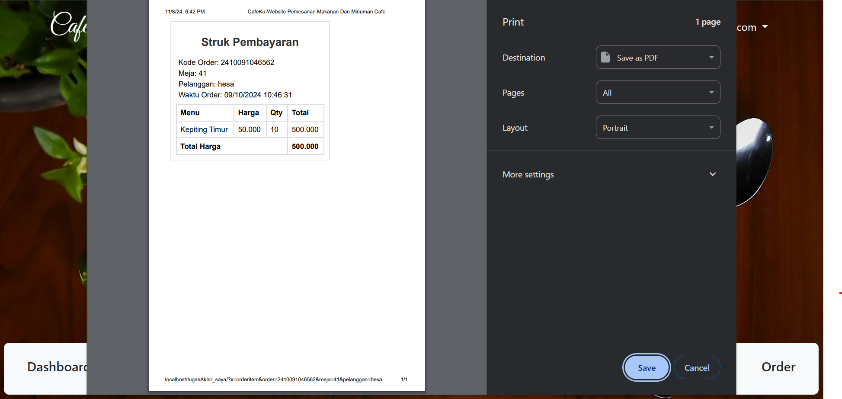
**Gambar 4. 5 Tampilan Order Item**



**Gambar 4. 6 Tambah Item Menu**

****

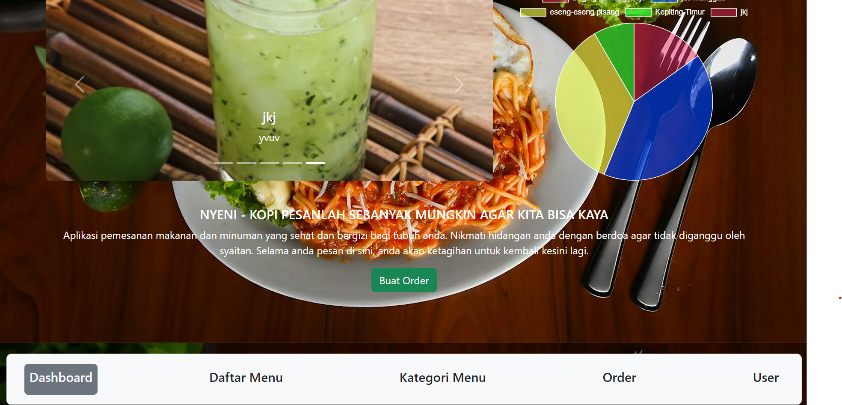
**Gambar 4. 7 Bayar**

****

**Gambar 4. 8 Cetak Struk**

### **4.1.3 Tampilan Akses Superadmin**

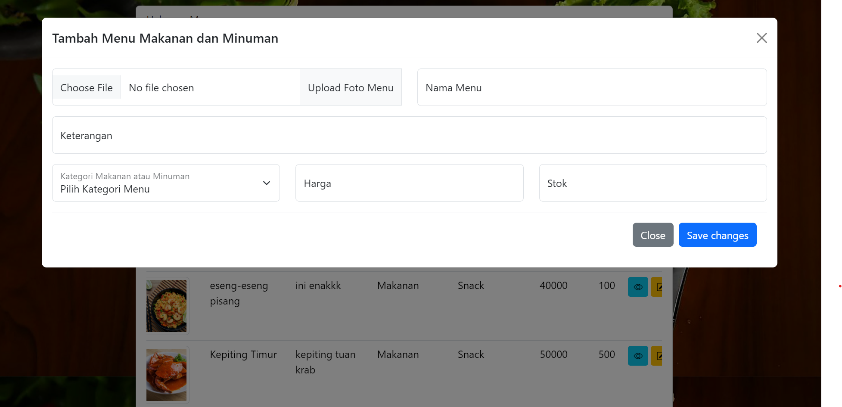
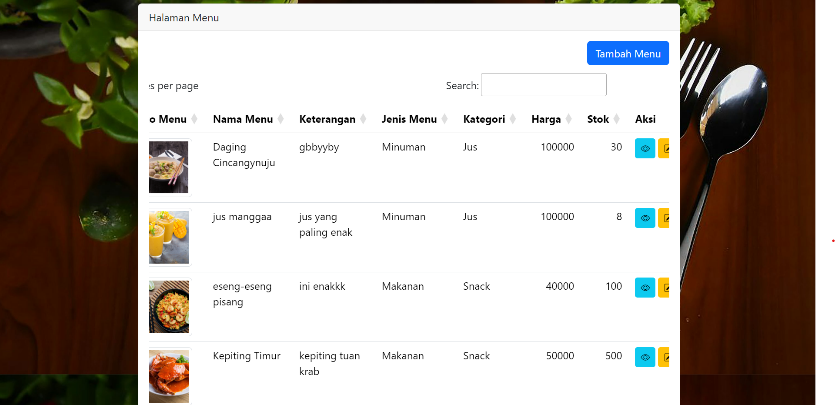
**4.1.3.1 Dashboard**

****Berbeda dengan tampilan Admin, disini Superadmin bisa mengakses secara keseluruhan seperti daftar menu,kategori menu,order bahkan juga user.

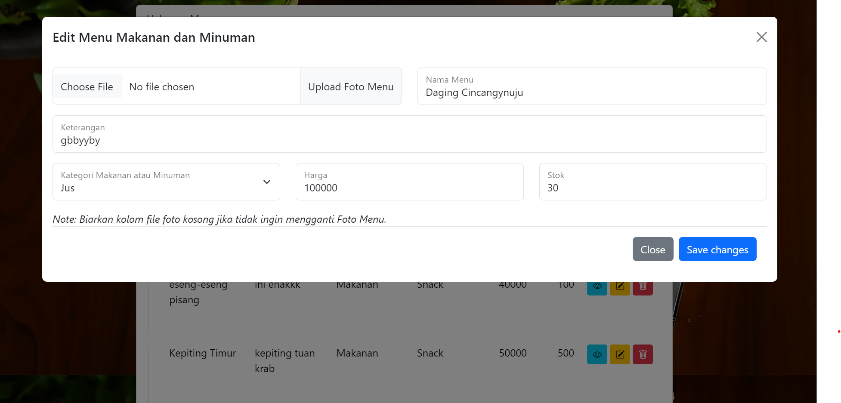
**Gambar 4. 9 Tampilan Superadmin**

**4.1.3.2 Daftar Menu**

Tampilan halaman Daftar Menu yang menampilakan seluruh menu-menu yang ada di cafe, kita bisa menambah menu, mengedit menu dan menghapusnya disini kita juga bisa melihat dari nama menunya, jenis, harga, stok dll.

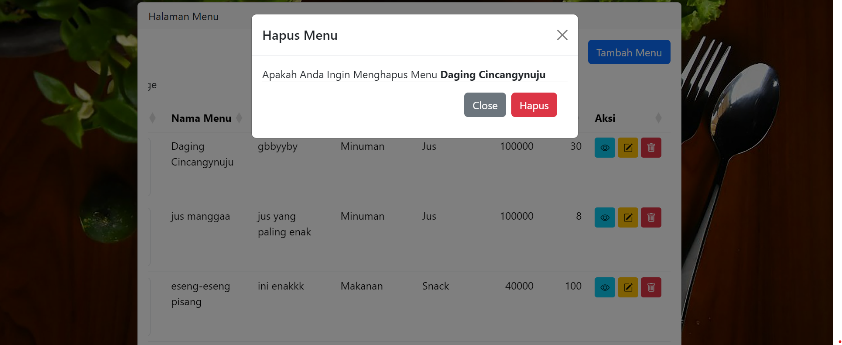
****

**Gambar 4. 10 Daftar Menu**

****

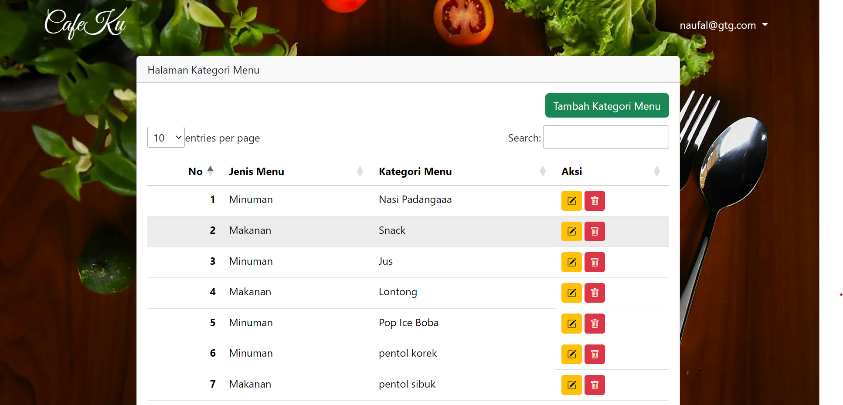
**Gambar 4. 11 Edit Menu**

**Gambar 4. 12 Tambah Menu**

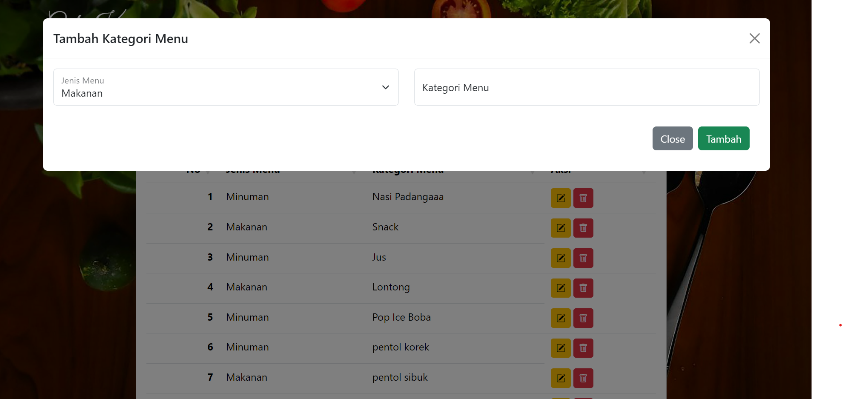


**Gambar 4. 13 Hapus Menu**

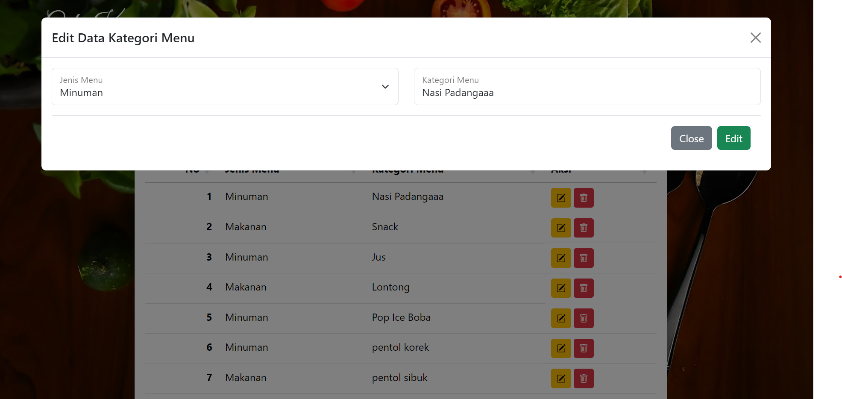
**4.1.3.3 Kategori Menu**

****Tampilan halaman Kategori Menu menampilakan kategori seluruh menu-menu yang ada di cafe, kita bisa menambah kategori menu, mengedit dan menghapusnya disini kita juga bisa melihat kategori menu apa saja yang ada di cafe ini.

**Gambar 4. 14 Kategori Menu**



**Gambar 4. 15 Tambah Kategori Menu**



**Gambar 4. 16 Edit Kategori Menu**

**BAB V**

**KESIMPULAN DAN SARAN**

## **5.1 Kesimpulan**

Aplikasi pemesanan cafe/restoran berbasis web dengan menggunakan bootstrap dan php dapat di ambil kesimpulan bahwa :

1. Dalam Pembuatan *website* Sistem Akademik ini pertama kali yang harus di lakukan adalah melakukan analisa terhadap kebutuhan sistem yang akan dibuat, kedua membuat DFD sesuai dengan kebutuhan sitem yang akan di buat, ketiga buat design database sistem yang sesuai dengan kebutuhan, keempat buata flowchart atau alur jalan dari sistem sesuai dengan kebutuhan sistem, kelima kita buat struktur yang akan ada pada setiap tabel pada database, ke enam buat mockup tampilan untuk setiap yang akses sistem.
2. aplikasi pemesanan cafe/restoran berbasis web bertujuan untuk memudahkan pelanggan dalam melakukan reservasi dan pemesanan makanan secara online, serta meningkatkan efisiensi operasional restoran. Manfaatnya meliputi peningkatan kepuasan pelanggan, pengurangan waktu tunggu, pengelolaan stok yang lebih baik, serta potensi peningkatan penjualan dan personalisasi layanan. Aplikasi ini tidak hanya mempermudah pelanggan tetapi juga membantu restoran dalam mengelola bisnisnya secara lebih efektif dan efisien.

## **5.2 Saran**

Di karenakan setiap system buatan manusia pasti memiliki kekurangaan maka dari itu penulis ingin memberikan saran yang di rasa dapat melengkapi kekurangan dari system tersebut. Fitur yang dapat di tambahkan sebagai berikut :

1. Menambahkan fitur nontifikasi untuk setiap aksi yang ada.
2. Mengganti desain interface website agar lebih menarik dan interaktif

# **Daftar Pustaka**

TeknoMia, *Cara Membuat Nomor Halaman Berbeda Dalam 1 Dokumen Word.* (Online)( https://www.youtube.com/watch?v=ACcvrvKjfL0 ), diakses 18 November 2024.

Febriyafadly, *Membuat DFD Dengan Power Designer*

(Online)( <https://febriyafadly.wordpress.com/2012/11/15/membuat-dfd-menggunakan-powerdesigner/> ), diakses 8 November 2024.

Takagi Fujimaru, *Laragon Alternatif XAMPP Terbaik Windows*

(Online) ( <https://codepolitan.com/blog/laragon-alternatif-xampp-terbaik-di-windows-5ae1bfaeb07be> ), diakses 3 November 2024.

Unkown, *Contoh Biodata Penulis dan Cara Membuatnya*

(Online) ( <https://penerbitdeepublish.com/contoh-biodata-penulis/> ), diakses 3 November 2024.

# **Biodata Penulis**

Penulis ini dilahirkan di Malang, 29 Mei 2008, penulis yang biasa dipanggil dengan nama “naufal” ini memiliki nama lengkap Mukhammad Naufal Faisal Mubarok, dan merupakan anak ke satu dari 2 bersaudara. Penulis telah menempuh Pendidikan formal yaitu di PAUD Darul Ilmi, TK Darul Ilmi, Min 2 Kota Malang, dan ponpes Al-Munawwariyyah. Setelah lulus dari ponpes Al-Munawwariyyah pada tahun 2021. Penulis memutuskan melanjutkan Pendidikan di SMK PGRI 3 Malang di Bidang Keahlian Teknik Informasi dan Komunikasi, Kompetensi Keahlian Rekayasa Perangkat Lunak (RPL) pada tahun 2024 dan terdaftar dengan nomor induk 24033/2082.063. Di SMK penulis aktif.

