

**PEMBANGUNAN APLIKASI PEMESANAN MENU MAKANAN  
SECARA *ONLINE (E-MENU)* DENGAN PEMBAYARAN NON-TUNAI  
MENGGUNAKAN TEKNOLOGI *QR-CODE* BERBASIS WEB DAN  
*MOBILE* PADA CAFE DECO BOCO YATAI**

**TUGAS AKHIR**

Diajukan Sebagai Salah Satu Syarat untuk Menyelesaikan Program Strata-1 pada  
Departemen Sistem Informasi Fakultas Teknologi Informasi Universitas Andalas



**DEPARTEMEN SISTEM INFORMASI  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS**

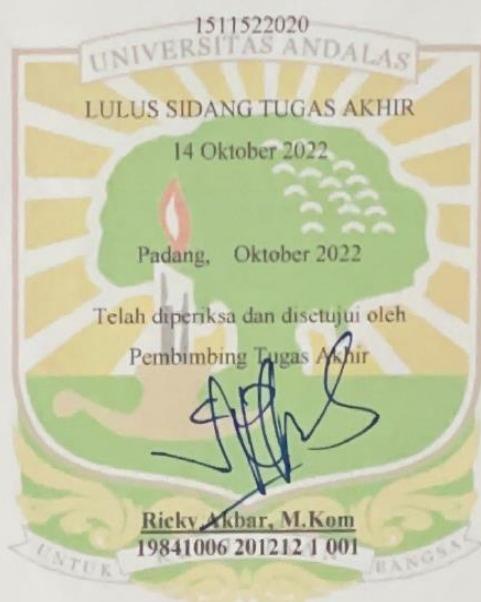
**2022**

HALAMAN PENGESAHAN

PEMBANGUNAN APLIKASI PEMESANAN MENU MAKANAN SECARA  
ONLINE (E-MENU) DENGAN PEMBAYARAN NON-TUNAI  
MENGGUNAKAN TEKNOLOGI QR-CODE BERBASIS WEB DAN  
MOBILE PADA CAFE DECO BOCO YATAI

Oleh:

Ferina Dwi Fitri



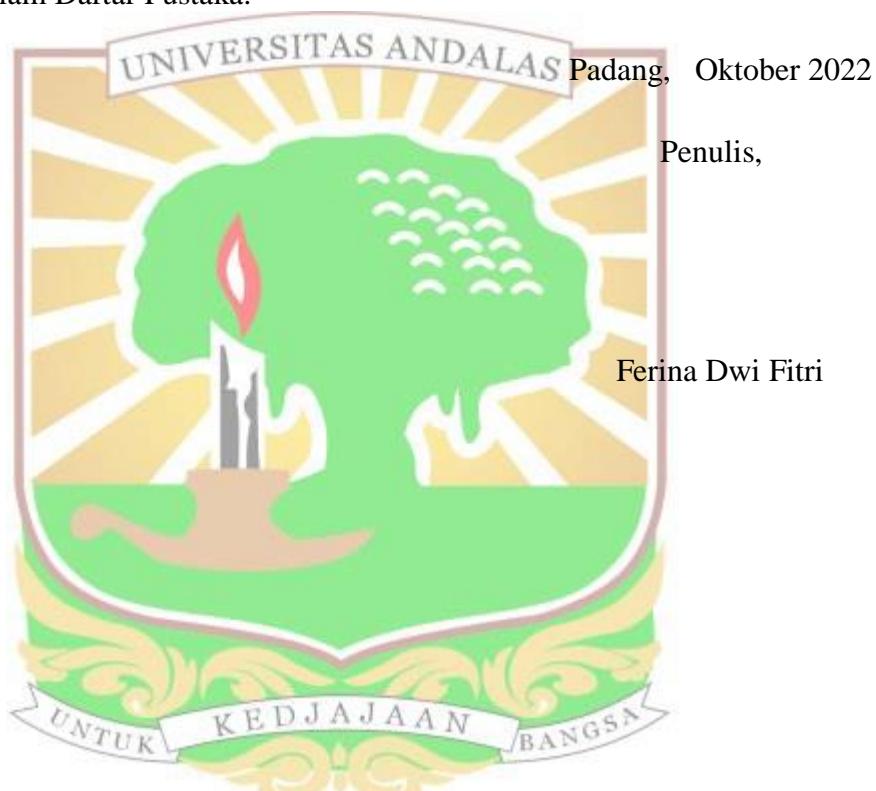
Mengetahui,

Ketua Departemen Sistem Informasi



## PERNYATAAN

Saya menyatakan bahwa laporan tugas akhir yang berjudul “Pembangunan Aplikasi Pemesanan Menu Makanan Secara *Online (E-menu)* Dengan Pembayaran Non-Tunai Menggunakan Teknologi *QR Code* Berbasis *Web* dan *Mobile* Pada *Cafe Deco Boco Yatai*” ini tidak terdapat karya yang pernah diajukan sebagai salah satu syarat menyelesaikan matakuliah tugas akhir di suatu Perguruan Tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diorbitkan oleh orang lain kecuali secara tertulis diacu dalam naskah ini dan disebut dalam Daftar Pustaka.



## KATA PENGANTAR

Puji dan syukur penulis ucapkan kehadiran Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir ini. Tugas akhir ini disusun berdasarkan penelitian yang penulis lakukan pada *Cafe Deco Boco Yatai* yang berjudul **“Pembangunan Aplikasi Pemesanan Menu Makanan Secara Online (E-menu) Dengan Pembayaran Non-Tunai Menggunakan Teknologi QR Code Berbasis Web dan Mobile Pada Cafe Deco Boco Yatai”**.

Dalam penyusunan tugas akhir ini penulis banyak mendapatkan bantuan dari berbagai pihak. Oleh sebab itu, penulis mengucapkan terima kasih atas bimbingan dan bantuan kepada:

1. Bapak Husnil Kamil, MT, selaku Ketua Departemen Sistem Informasi, Fakultas Teknologi Informasi Universitas Andalas,
2. Bapak Ricky Akbar, M.Kom, selaku pembimbing tugas akhir,
3. Karyawan Cafe Deco Boco Yatai

Penulis menyadari bahwa dalam penyusunan tugas akhir ini masih terdapat banyak kekurangan. Untuk itu penulis sangat mengharapkan kritik dan saran yang bersifat membangun dari pembaca melalui *email* [ferinadwifitri@gmail.com](mailto:ferinadwifitri@gmail.com). Akhirnya semoga laporan tugas akhir ini dapat *memberikan* manfaat bagi penulis dan pembaca.

Padang, Oktober 2022

Penulis,

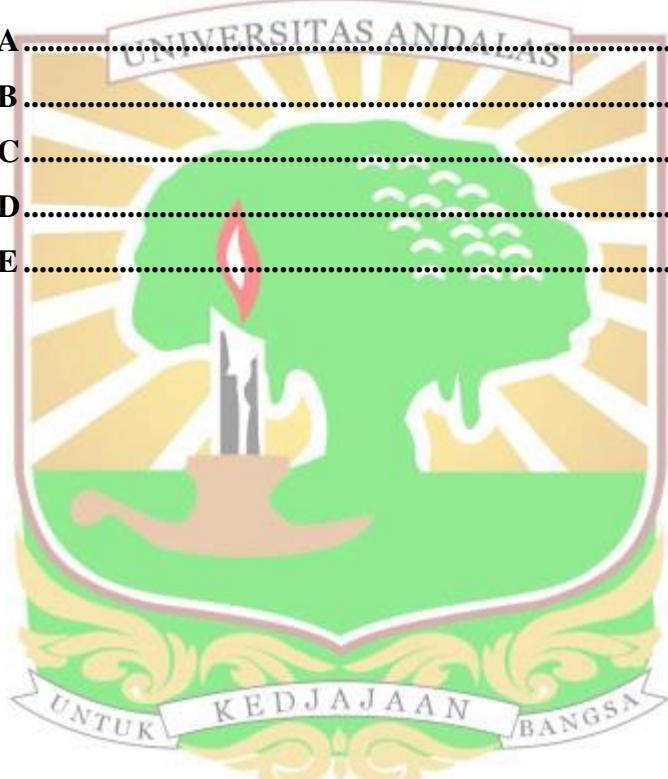
Ferina Dwi Fitri

## DAFTAR ISI

<b>HALAMAN PENGESAHAN .....</b>	Error! Bookmark not defined.
<b>PERNYATAAN.....</b>	<b>ii</b>
<b>KATA PENGANTAR.....</b>	<b>iv</b>
<b>DAFTAR ISI.....</b>	<b>v</b>
<b>DAFTAR GAMBAR.....</b>	<b>viii</b>
<b>DAFTAR TABEL.....</b>	<b>x</b>
<b>ABSTRAK .....</b>	<b>xi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah .....	4
1.4. Tujuan Penilitian.....	4
1.5. Sistematika Penulisan .....	4
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>6</b>
2.1. Konsep <i>E-menu</i> .....	6
2.2. Sistem Pembayaran non Tunai.....	6
2.3. Konsep QR Code .....	7
2.4. Perangkat Lunak Pendukung .....	9
2.4.1. Bahasa Pemrograman PHP.....	9
2.4.2. HTML & CSS .....	10
2.4.3. MySQL.....	10
2.4.4. <i>Unified Modelling Language</i> (UML).....	11
2.4.5. Android Studio .....	11
2.5. Alat Pemodelan Sistem yang Digunakan.....	12
2.5.1. BPMN ( <i>Bussiness Process Model Notation</i> ) .....	12
2.5.2. <i>Use Case Diagram</i> .....	12
2.5.3. <i>Use Case Scenario</i> .....	14

2.5.4. <i>Sequence diagram</i> .....	14
2.5.5. <i>Entity Relationship Diagram</i> .....	14
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>16</b>
3.1. Objek Penelitian.....	16
3.2. Metode Pengumpulan Data.....	16
3.3. Metode Pengembangan Sistem .....	17
3.4. <i>Flowchart</i> Metodologi Penelitian .....	18
<b>BAB IV ANALISIS DAN PERANCANGAN SISTEM .....</b>	<b>21</b>
4.1. Analisis Sistem.....	21
4.1.1. Kedudukan Sistem .....	21
4.1.2. Sistem yang Berjalan.....	21
4.1.3. Sistem yang Diusulkan.....	23
4.1.4. Analisis Kebutuhan Fungsional .....	25
4.1.5. <i>Use case diagram</i> Aplikasi .....	26
4.1.6. Deskripsi Tugas Aktor .....	28
4.1.7. <i>Use case scenario</i> Aplikasi .....	30
4.2. Perancangan Sistem .....	33
4.2.1. <i>Context Diagram</i> .....	33
4.2.2. Data Flow Diagram.....	34
4.2.3. Perancangan <i>Database</i> .....	35
4.2.4. Arsitektur Aplikasi .....	37
4.2.5. Perancangan Antarmuka .....	37
<b>BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM.....</b>	<b>41</b>
5.1. Implementasi Sistem.....	41
5.1.1. Pengkodean Program .....	42
5.1.2. Implementasi Antar Muka Aplikasi <i>Web</i> .....	44
5.1.3. Implementasi Antar Muka Aplikasi <i>Mobile</i> .....	48

5.2. Pengujian Sistem.....	50
5.2.1. Fokus Pengujian.....	50
5.2.2. Kasus Hasil Pengujian.....	52
5.3. Kesimpulan Hasil Pengujian.....	59
<b>BAB VI KESIMPULAN DAN SARAN.....</b>	<b>68</b>
6.1. Kesimpulan .....	68
6.2. Saran .....	68
<b>DAFTAR PUSTAKA .....</b>	<b>69</b>
<b>LAMPIRAN A .....</b>	<b>71</b>
<b>LAMPIRAN B .....</b>	<b>77</b>
<b>LAMPIRAN C .....</b>	<b>80</b>
<b>LAMPIRAN D .....</b>	<b>93</b>
<b>LAMPIRAN E .....</b>	<b>100</b>



## DAFTAR GAMBAR

Gambar 2.1 Contoh <i>QR Code</i> (Akbar & Kamil, 2017).....	7
Gambar 2.2 Diagram alir pembangkitan <i>QR Code</i> (Nugraha & Munir, 2011).....	8
Gambar 2.3 Diagram proses pembacaan <i>QR Code</i> (Nugraha & Munir, 2011) .....	8
Gambar 2.4 Elemen dari <i>Use Case Diagram</i> (Mulyani, 2016).....	13
Gambar 3.1 Metode <i>Waterfall</i> (Sommerville, 2011) .....	17
Gambar 3.2 <i>Flowchart</i> Penelitian .....	19
Gambar 4.1 Proses Pemesanan Menu Makanan yang Sedang Berjalan .....	22
Gambar 4.2 Proses Pembayaran yang Sedang Berjalan.....	23
Gambar 4.3 Proses Pemesanan yang Diusulkan .....	24
Gambar 4.4 Proses Pembayaran yang Disusulkan .....	25
Gambar 4.5 Use Case Diagram Perancangan Aplikasi .....	27
Gambar 4.6 <i>Context Diagram</i> .....	34
Gambar 4.7 Data Flow Diagram Level 0 .....	35
Gambar 4.8 Rancangan ERD Sistem .....	35
Gambar 4.9 Arsitektur Aplikasi .....	37
Gambar 4.10 Perancangan Antarmuka Halaman <i>Login</i> .....	38
Gambar 4.11 Perancangan Antarmuka Halaman Utama Admin .....	39
Gambar 4.12 Perancangan Antarmuka Halaman Pesanan.....	39
Gambar 4.13 Halaman <i>Login</i> .....	40
Gambar 4.14 Halaman Utama.....	40
Gambar 4.15 Halaman <i>Detail Menu</i> .....	40
Gambar 4.16 Halaman Profil <i>Member</i> .....	40
Gambar 5.1 Kode Program Koneksi <i>Database</i> Aplikasi .....	42

Gambar 5.2 Kode Program Koneksi <i>Database</i> Aplikasi .....	43
Gambar 5.3 Kode Program Tambah Pesanan .....	44
Gambar 5.4 Kode Program Tambah Pesanan (Lanjutan) .....	44
Gambar 5.5 Tampilan Halaman <i>Login</i> .....	45
Gambar 5.6 Tampilan Halaman Utama Admin .....	46
Gambar 5.7 Tampilan Halaman Koki .....	48
Gambar 5.8 Tampilan Halaman Awal .....	49
Gambar 5.9 Tampilan Halaman Utama.....	49
 The logo of Universitas Andalas, featuring a green and yellow circular design with a central emblem and the university's name in Indonesian and English.	50
Gambar 5.10 Halaman Menu.....	50
Gambar 5.11 Halaman Profil.....	50
Gambar 5.12 Halaman Awal Sebelum Tambah Pesan .....	53
Gambar 5.13 Halaman Pilih Meja.....	53
Gambar 5.14 Halaman Pilih Menu.....	54
Gambar 5.15 Halaman Pesanan Baru Telah Berhasil Ditambahkan.....	54
Gambar 5.16 <i>Input</i> Pengujian Transaksi Pembayaran (Benar) .....	55
Gambar 5.17 <i>Output</i> Pengujian Transaksi Pembayaran (Benar) .....	55
Gambar 5.18 Pengecekan Pengujian Transaksi Pembayaran (Benar) .....	56
Gambar 5.19 <i>Input</i> Pengujian Transaksi Pembayaran (Alternatif) .....	57
Gambar 5.20 <i>Output</i> Pengujian Transaksi Pembayaran (Alternatif) .....	57
Gambar 5.21 Pengecekan Pengujian Transaksi Pembayaran (Alternatif) .....	57
Gambar 5.22 Ubah Status Pesanan Baru.....	58
Gambar 5.23 Status Pesanan Selesai.....	59

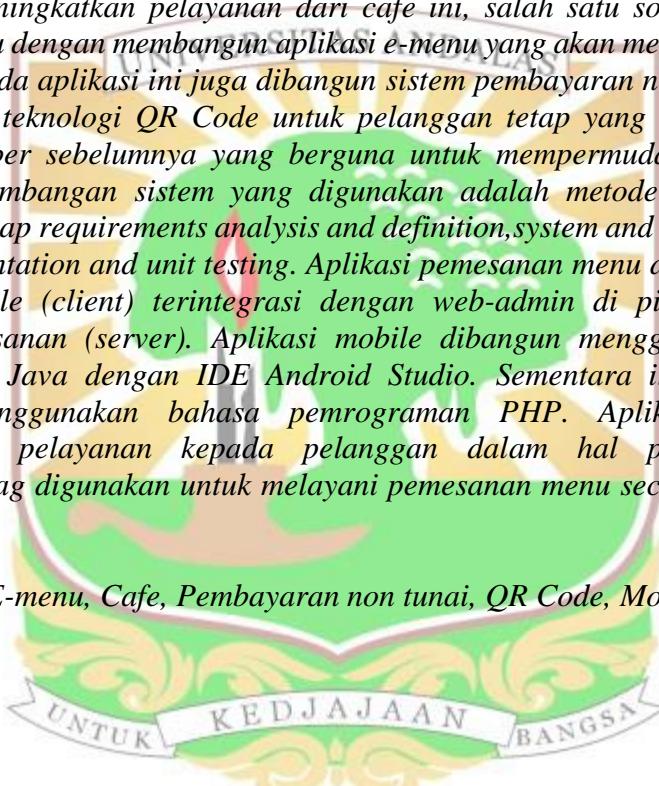
## DAFTAR TABEL

Tabel 4.1 Deskripsi Tugas Aktor .....	28
Tabel 4.2 Use Case Scenario Melakukan Pemesanan.....	30
Tabel 4.3 Use Case Scenario Mengubah Status Pesanan.....	31
Tabel 4.4 Use Case Scenario Transaksi Pembayaran .....	32
Tabel 4.5 Tabel Menu .....	36
Tabel 4.6 Tabel Transaksi .....	36
Tabel 4.7 Tabel <i>Detail</i> Transaksi .....	36
Tabel 5.1 Fokus Pengujian.....	51
Tabel 5.2 Pengujian Melakukan Pemesanan (Benar) .....	52
Tabel 5.3 Pengujian Transaksi Pembayaran <i>Cafe</i> (Benar).....	55
Tabel 5.4 Pengujian Transaksi Pembayaran (Alternatif) .....	56
Tabel 5.5 Pengujian Lihat Laporan Transaksi <i>Cafe</i> (Benar).....	58
Tabel 5.6 Tabel Hasil Pengujian .....	59

## ABSTRAK

*Cafe Deco Boco Yatai merupakan salah satu usaha tempat makan yang terletak di Jalan Muhammad Hatta kelurahan Cupak Tangah kecamatan Pauh kota Padang Sumatera Barat yang menyajikan aneka macam masakan khas Jepang. Transaksi pada cafe ini masih menggunakan proses manual atau belum terkomputerisasi, baik dalam proses pemesanan maupun pembayaran sehingga masih banyak ditemukan kendala dalam hal pelayanan kepada pelanggan. Kendala-kendala pada sistem pemesanan yang sedang berjalan antara lain adanya pemesanan rangkap, tidak urutnya pembuatan pesanan akibat bertumpuknya orderan terutama pada saat ramai pengunjung dan kendala pada proses pembayaran antara lain kesalahan dalam penghitungan dan kesulitan dalam pengembalian uang pembayaran dari pelanggan. Untuk mengatasi kendala yang ada, guna meningkatkan pelayanan dari cafe ini, salah satu solusi yang dapat diusulkan yaitu dengan membangun aplikasi e-menu yang akan mengelola pesanan pelanggan. Pada aplikasi ini juga dibangun sistem pembayaran non tunai dengan menggunakan teknologi QR Code untuk pelanggan tetap yang telah mendaftar menjadi member sebelumnya yang berguna untuk mempermudah pembayaran. Metode pengembangan sistem yang digunakan adalah metode waterfall yang terdiri dari tahap requirements analysis and definition, system and software design, serta implementation and unit testing. Aplikasi pemesanan menu di sisi pelanggan berbasis mobile (client) terintegrasi dengan web-admin di pihak cafe untuk mengelola pesanan (server). Aplikasi mobile dibangun menggunakan bahasa pemrograman Java dengan IDE Android Studio. Sementara itu aplikasi web dibangun menggunakan bahasa pemrograman PHP. Aplikasi ini dapat meningkatkan pelayanan kepada pelanggan dalam hal pemesanan dan pembayaran yang digunakan untuk melayani pemesanan menu secara langsung di cafe.*

*Kata Kunci : E-menu, Cafe, Pembayaran non tunai, QR Code, Mobile, Web.*



## BAB I

### PENDAHULUAN

#### 1.1. Latar Belakang

Salah satu hal yang paling penting dalam dunia bisnis untuk menuju kesuksesan yaitu adalah pelayanan yang baik bagi pelanggan. Dengan adanya pelayanan yang baik, tepat dan cepat akan memberikan kepuasan bagi pelanggan yang akan berujung kepada keloyalan pelanggan terhadap suatu bisnis. Dengan adanya kepuasan pelanggan tersebut akan meningkatkan penjualan dan memberikan keuntungan bagi suatu bisnis.

Pemanfaatan teknologi pada saat ini telah dirasakan di semua aspek, sehingga membawa perubahan terhadap gaya hidup pada masyarakat, termasuk pada restoran atau tempat makan. Pemanfaatan teknologi dapat digunakan untuk meningkatkan pelayanan terhadap pelanggan. Manusia pada umumnya menginginkan segala sesuatu dapat dengan mudah dikerjakan, begitu pula dengan pelanggan restoran atau tempat makan yang ingin memesan menu dengan mudah dalam artian tidak rumit dan tidak memakan waktu yang lama. Mudah dalam memesan menu yang dimaksud adalah tanpa harus mengantre dan tanpa harus menunggu pelayan yang sibuk dengan pelayan lainnya (Tompoh dkk, 2016).

*Cafe Deco Boco Yatai* merupakan salah satu usaha tempat makan yang terletak di Jalan Muhammad Hatta kelurahan Cupak Tangah kecamatan Pauh kota Padang Sumatera Barat yang menyajikan aneka macam masakan khas Jepang. *Cafe Deco Boco Yatai* mampu menerima sekitar 30-50 transaksi per hari. Daya tarik dari *Cafe* ini yaitu makanan yang dijual adalah makanan khas Jepang. *Cafe* ini ramai dikunjungi karena menu yang disajikan cukup unik dan letaknya yang dekat dengan area kampus.

Proses bisnis yang dilakukan pada *Cafe Deco Boco Yatai* pada saat ini masih menggunakan cara manual atau belum terkomputerisasi, baik dalam pemesanan maupun transaksi. Pemesanan yang dilakukan masih menggunakan media pena dan kertas untuk mencatat pesanan sehingga timbul kendala-kendala antara lain adanya pemesanan rangkap, tidak urutnya pembuatan pesanan akibat bertumpuknya orderan terutama pada saat ramai pengunjung. Selanjutnya transaksi pembayaran

pada *cafe* ini masih menggunakan cara manual yang belum terkomputerisasi. Akibatnya kemungkinan akan terjadi kesalahan dalam penghitungan dan kesulitan dalam pengembalian uang pembayaran dari pelanggan. Salah satu solusi untuk mengatasi permasalahan ini adalah dengan membangun sebuah aplikasi pemesanan menu makanan (*e-menu*) dengan pembayaran non tunai menggunakan teknologi QR Code.

Aplikasi ini dibangun melalui aplikasi *mobile* yang dapat mempermudah proses pemesanan makanan pada *cafe*, karena pemesanan menu makanan secara manual akan menemui beberapa kendala seperti yang telah disebutkan di atas. Semua fitur yang dibangun dalam aplikasi ini hanya bisa diakses di dalam area *cafe* dan digunakan untuk memesan makanan untuk makan di tempat (*dine in*) karena aplikasi mengharuskan pengguna untuk memilih meja terlebih dahulu sebelum melakukan pemesanan. Pada aplikasi ini juga terdapat fitur pembayaran non tunai dengan menggunakan teknologi *QR Code* yang dibangun khusus untuk *member* yang telah berlangganan pada *cafe* ini. Dengan adanya aplikasi ini diharapkan dapat meningkatkan pelayanan pada *cafe* dan meminimalisir terjadinya *human error* yang ada pada sistem yang sedang berjalan.

Sebagai referensi untuk *memberikan* pengetahuan dan wawasan mengenai aplikasi *e-menu* pada *cafe* ini, penulis merujuk kepada beberapa penelitian yang berkaitan dengan penelitian ini yang telah dilakukan sebelumnya. Tompoh dkk pada 2016 melakukan penelitian yang berjudul Rancang Bangun Aplikasi Pemesanan Menu Makanan Restoran Berbasis Android yang bertujuan untuk merancang dan membangun suatu aplikasi pemesanan menu makanan restoran untuk membantu pihak restoran dalam menangani pemesanan menu makanan. Dengan menggunakan *web-service* aplikasi pemesanan menu di sisi pelanggan berbasis android (*client*) dapat terintegrasi dengan aplikasi *web-admin* pihak restoran untuk mengatur pemesanan (*server*) (Tompoh dkk,2016).

Penelitian yang dilakukan Wibowo tahun 2014 dengan judul Analisis dan Perancangan Sistem *E-menu* Pada Rumah Makan Muara Kapuas menghasilkan sistem *e-menu* berbasis *web* yang dapat melakukan transaksi pemesanan menu makanan melalui media elektronik seperti *smartphone* atau *tablet* yang dapat meningkatkan kinerja rumah makan Muara Kapuas dan dapat dijadikan sebagai

salah satu strategi bisnis untuk memajukan usaha rumah makan Muara Kapuas. Dilihat dari hasil uji coba sistem atau *black box testing* sistem *e-menu* ini dapat mengantikan media kertas sebagai pencatat pesanan menu sehingga dapat meminimalisir penggunaan kertas. Selain itu proses pemesanan menu dan pembayaran pun menjadi lebih cepat, mudah, dan akurat sehingga pelanggan tidak perlu menunggu terlalu lama dan hal tersebut menjadi penilaian positif yang lebih dari pelanggan (Wibowo, 2014).

Mardela Rajesh didalam jurnalnya yang berjudul E-Restaurant: Online Restaurant Management System for Android (Rajesh, Satya, & Rao, 2015), menyatakan bahwa kesederhanaan dan kemudahan akses menu adalah hal utama yang memudahkan dalam memesan makanan di sebuah restoran. Keuntungan memanfaatkan sistem *e-menu* bagi konsumen yaitu, *e-menu* menyediakan informasi tambahan mengenai makanan dan minuman dibandingkan dengan menu berbasis kertas. Pengembang aplikasi sejenis yang telah melakukan tahap *maintenance* / perawatan menemukan fakta bahwa konsumen yang duduk di meja yang dilengkapi dengan smartphone menghabiskan uangnya 10% lebih banyak dibandingkan meja lainnya (*people buy more when they can do so instantly, without waiting for service*). Berdasarkan pendapat diatas, maka dapat disimpulkan sistem *e-menu* memberikan kemudahan bagi konsumen dalam mengakses informasi dan melakukan pemesanan menu.

Berdasarkan uraian yang telah dijelaskan di atas, maka dilakukan penelitian tugas akhir yang diberi judul **“Pembangunan Aplikasi Pemesanan Menu Makanan Secara Online (*E-menu*) Dengan Pembayaran Non-Tunai Menggunakan Teknologi QR Code Berbasis Web dan Mobile Pada Cafe Deco Boco Yatai”**. Aplikasi ini dibangun agar dapat mengatasi permasalahan yang ada pada Cafe Deco Boco Yatai.

## 1.2. Rumusan Masalah

Berdasarkan uraian yang telah dijelaskan pada sub bab sebelumnya, maka permasalahan yang dibahas dalam penelitian ini adalah bagaimana cara merancang dan membangun sebuah aplikasi *e-menu* menggunakan teknologi *QR Code* berbasis *web* dan *mobile* pada *Cafe Deco Boco Yatai*.

### 1.3. Batasan Masalah

Berdasarkan rumusan masalah yang ada di atas, maka batasan masalah dalam penelitian ini adalah :

1. Aplikasi *e-menu* ini hanya dapat digunakan untuk pemesanan makanan untuk makan di tempat (*dine in*).
2. Aplikasi *e-menu* ini mencakup pengelolaan menu makanan dan minuman, pengelolaan data pesanan, pengelolaan data pelanggan yang telah menjadi *member*, dan pembayaran menggunakan sistem transaksi non tunai dengan memanfaatkan teknologi *QR Code*, serta pengisian dan penambahan deposit *member*.
3. Bagi pelanggan yang ingin menjadi *member* harus mendaftar terlebih dahulu ke *Cafe Deco Boco Yatai*. Setelah pelanggan mendaftar baru bisa melakukan *login* pada aplikasi *mobile*.
4. Sistem yang dibangun berbasis *web* sebagai halaman admin dan digunakan untuk pengelolaan pesanan, transaksi dan pengelolaan *member*, sedangkan aplikasi *mobile* sebagai halaman pengguna yang digunakan untuk melihat daftar menu, memesan makanan, melihat biaya pesanan, dan melakukan pembayaran non tunai dengan teknologi *QR Code*.

### 1.4. Tujuan Penilitian

Adapun tujuan dari penelitian ini adalah sebagai berikut :

1. Menganalisis dan merancang aplikasi *e-menu* dengan menggunakan teknologi *QR Code* berbasis *web* dan *mobile*.
2. Membangun aplikasi *e-menu* dengan menggunakan teknologi *QR Code* pada *Cafe Deco Boco* berbasis *web* dan *mobile* dengan mengoptimalkan proses bisnis yang ada.
3. Melakukan pengujian aplikasi *e-menu* dengan menggunakan teknologi *QR Code* pada *Cafe Deco Boco* berbasis *web* dan *mobile* yang telah dibuat sehingga sesuai dengan kebutuhan.

### 1.5. Sistematika Penulisan

Sistematika penulisan laporan ini dibagi menjadi enam bab yaitu :

## BAB I : PENDAHULUAN

Bab ini berisi tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, serta sistematika penulisan laporan.

## BAB II : TINJAUAN PUSTAKA

Bab ini berisi tentang landasan teori dan informasi pendukung yang digunakan untuk penelitian ini.

## BAB III : METODOLOGI PENELITIAN

Bab ini menjelaskan tentang objek penelitian, metode pengumpulan data, metode pengembangan sistem yang digunakan, dan *flowchart* penelitian.

## BAB IV : ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi tentang pemodelan analisis sistem menggunakan *tools* seperti, *Business Process Model Notation* (BPMN), *use case diagram*, *scenario diagram*, *sequence diagram*, perancangan basis data, struktur basis data dan tabel, *class diagram*, arsitektur aplikasi dan perancangan antarmuka.

## BAB V : IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisi tentang pengimplementasian aplikasi ke dalam bahasa pemrograman berdasarkan analisis dan perancangan, serta pengujian terhadap hasil implementasi sistem.

## BAB VI : PENUTUP

Berisi tentang kesimpulan terhadap hasil penelitian dan saran untuk pengembangan sistem kedepannya.

## BAB II

### TINJAUAN PUSTAKA

Bab ini menjelaskan teori-teori yang digunakan untuk mendukung penelitian tugas akhir. Beberapa hal yang menjadi tinjauan pustaka adalah mengenai sistem pemesanan menu makanan *online (e-menu)*, sistem pembayaran non tunai menggunakan teknologi *QR Code*, perangkat lunak pendukung yang digunakan dalam pembangunan sistem, serta hal-hal yang lain terkait dengan *e-menu*.

#### **2.1. Konsep *E-menu***

*E-menu* berasal dari dua suku kata yaitu, *electronic* dan *menu*. Menu merupakan daftar makanan yang disajikan kepada tamu di ruang makan. Sedangkan elektronik merupakan alat yang dibuat berdasarkan prinsip elektronika yang memanfaatkan jaringan komputer untuk pengimplementasiannya (Hendri, 2016). Jadi, *E-menu* adalah penggunaan jaringan komputer untuk melakukan pemesanan makanan atau minuman pada sebuah rumah makan atau restoran yang dilakukan secara elektronik. Pengertian *e-menu* tersebut diambil dari pengertian *e-commerce* yaitu pembelian, penjualan dan pemasaran barang serta jasa melalui sistem elektronik. Perbedaan pada kedua istilah tersebut hanyalah pada fungsi dan radius dari jaringan komputer yang digunakan (Wibowo, 2014).

#### **2.2. Sistem Pembayaran non Tunai**

Sistem pembayaran non tunai adalah transaksi yang dilakukan tanpa menggunakan uang tunai yang beredar. Alat transaksi pembayaran non tunai ini dibedakan atas tiga jenis yaitu : alat pembayaran non tunai berbasis surat, alat pembayaran non tunai berbasis elektronik, dan alat pembayaran non tunai berbasis ponsel (*mobile based*). Alat pembayaran non tunai berbasis elektronik dibedakan lagi atas dua jenis, yaitu : alat pembayaran menggunakan kartu yang meliputi kartu dan uang elektronik. Sebagaimana diketahui bahwa uang elektronik pada hakikatnya adalah uang tunai yang dikonversi menjadi data elektronik yang disimpan pada sebuah media penyimpanan, bisa dalam media *server* atau kartu

*chip*. Di sini terjadi penciptaan uang giral dengan cara menyetorkan uang tunai yang nilainya dimasukkan pada media penyimpan uang elektronik. Uang elektronik tersebut hanya dapat digunakan sebagai alat pembayaran secara giral dengan cara dipindahkan secara elektronik untuk kepentingan transaksi pembayaran atau transfer dana. Jadi, uang elektronik merupakan alat pembayaran non tunai karena mekanisme transaksi pembayarnya menggunakan pembayaran secara tidak langsung melalui media penyimpan secara elektronik berupa *server* atau *chip*. Di dalamnya terdapat nilai uang elektronik yang berfungsi sebagai alat pembayaran giral kepada pedagang yang bukan penerbit uang elektroniknya (Usman, 2017).

### 2.3. Konsep QR Code

*QR Code* adalah image berupa matriks dua dimensi yang memiliki kemampuan untuk menyimpan data di dalamnya. *QR Code* merupakan evolusi dari kode batang (*barcode*). *Barcode* merupakan sebuah simbol penandaan objek nyata yang terbuat dari pola batang-batang berwarna hitam dan putih agar mudah untuk dikenali oleh komputer (Rahayu dkk, 2006). Contoh sebuah *QR Code* dapat dilihat pada gambar 2.1 berikut.



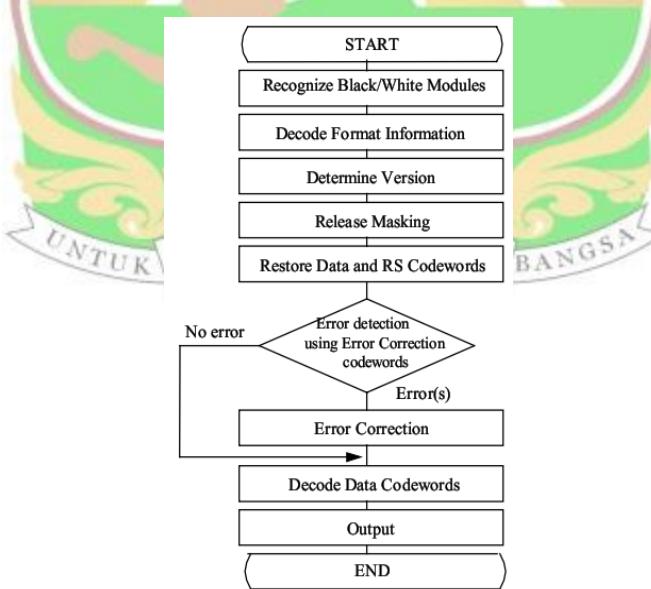
Gambar 2.1 Contoh *QR Code* (Akbar & Kamil, 2017)

*QR Code* merupakan singkatan dari Quick Response Code, atau dapat diterjemahkan menjadi kode respon cepat. *QR Code* dikembangkan oleh Denso Corporation, sebuah perusahaan Jepang yang banyak bergerak di bidang otomotif. *QR Code* ini dipublikasikan pada tahun 1994 dengan tujuan untuk pelacakan kendaraan di bagian manufaktur dengan cepat dan mendapatkan respon dengan cepat pula (Nugraha & Munir, 2011). Prosedur pembangkitan *QR Code* dari sebuah teks dapat dijelaskan dengan diagram alir pada gambar 2.2 berikut.



Gambar 2.2 Diagram alir pembangkitan *QR Code* (Nugraha & Munir, 2011)

Langkah-langkah untuk membaca *QR Code* menjadi teks aslinya merupakan *reverse* atau kebalikan dari langkah-langkah pada pembangkitan *QR Code*. Secara umum prosedur pembacaan *QR Code* dapat dijelaskan dengan diagram alir pada gambar 2.3 berikut.



Gambar 2.3 Diagram proses pembacaan *QR Code* (Nugraha & Munir, 2011)

Berikut merupakan beberapa kelebihan dari teknologi *QR Code* (Pramihapsari & Kaldera, 2012) :

1. *QR Code* memiliki kapasitas tinggi dalam data pengkodean, yaitu mampu menyimpan semua jenis data, seperti data numerik, data alphabetis, kanji, kana, hiragana, simbol, dan kode biner secara spesifik. Dibandingkan dengan *barcode* yang hanya dapat menyimpan jenis data numerik.
2. *QR Code* mampu menyimpan data jenis numerik sampadengan 7.089 karakter, data alphanumerik sampai dengan 4.296 karakter, kode binari sampai dengan 2.844 *byte*, dan huruf kanji sampai dengan 1.817 karakter.
3. *QR Code* memiliki tampilan lebih kecil daripada *barcode*. Hal ini dikarenakan *QR Code* mampu menampung data secara horizontal dan vertikal.
4. *QR Code* tahan terhadap kerusakan, sebab *QR Code* mampu memperbaiki kesalahan sampai dengan 30%. Oleh karena itu, walaupun sebagian simbol *QR Code* kotor ataupun rusak, data tetap dapat disimpan dan dibaca. Tiga tanda berbentuk persegi di tiga sudut memiliki fungsi agar simbol dapat dibaca dengan hasil yang sama dari sudut manapun sepanjang 360 derajat.

## 2.4. Perangkat Lunak Pendukung

Sub bab ini menjelaskan tentang perangkat lunak pendukung yang digunakan dalam pembangunan sistem informasi.

### 2.4.1. Bahasa Pemrograman PHP

PHP merupakan salah satu bahasa pemrograman berbasis *web* dimana sistem yang diterapkan adalah pada sisi *server side*. PHP dapat disisipkan diantara skrip-skrip bahasa HTML dan arena bahasa *server side* lainnya, dengan itu PHP akan dieksekusi secara langsung pada *server*. Sedangkan *browser* akan mengeksekusi halaman *web* tersebut melalui *server* yang kemudian akan menerima tampilan “hasil jadi” dalam bentuk HTML, sedangkan kode PHP itu sendiri tidak akan dapat terlihat (Haryana, 2008).

Ketika seorang pengguna internet membuka suatu situs yang menggunakan fasilitas *server side scripting* PHP, maka terlebih dahulu *server* yang bersangkutan akan memproses semua perintah PHP di *server* lalu mengirimkan hasilnya dalam format HTML ke *web server* pengguna internet tadi. Sehingga kode asli yang ditulis dengan PHP tidak terlihat di *browser* pengguna. PHP tergolong sebagai perangkat lunak *open source* yang diatur dalam aturan *general purpose license* (GPL). Pemrograman PHP sangat cocok dikembangkan dalam lingkungan *web*, karena PHP ini bisa dilekatkan pada *script* HTML atau sebaliknya. PHP dikhususkan untuk pengembangan *web* dinamis. Maksudnya, PHP mampu menghasilkan *website* yang secara terus menerus hasilnya bisa berubah-ubah sesuai dengan pola yang diberikan, Nugroho dalam (Haryanti & Irianto, 2011).

#### 2.4.2. HTML & CSS

HTML adalah bahasa dasar untuk membuat *mobilesite*. Struktur dan tampilan *mobilesite* ditentukan dari susunan kode HTML yang berisi tag-tag pembuka dan penutup. CSS (*Cascading Style Sheets*) adalah pelengkap dari HTML. CSS membentuk aturan tampilan atau style yang digunakan pada *mobilesite*, sehingga dapat dipisahkan antara isi *mobilesite* yang ditulis dengan HTML dan tampilan *mobile* yang ditulis dengan CSS. Biasanya CSS digunakan untuk menentukan warna, *font*, *layout*, dan berbagai macam hal yang menyangkut tampilan. Pemisahan ini membuat kita lebih leluasa untuk mengontrol tampilan tanpa mengutak atik struktur isi (Ollie, 2008).

#### 2.4.3. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*Database Management System*), *database* ini *multithread*, *multi user*. MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public Licence* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus yang bersifat khusus. Kekuatan MySQL tidak ditopang oleh sebuah komunitas, seperti Apache, yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh pemilik masing-masing, tetapi MySQL didukung penuh oleh sebuah perusahaan profesional dan

komersial, yakni MySQL AB dari Swedia. MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (*General Public License*). Di mana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (Wardani, 2012).

#### **2.4.4. *Unified Modelling Language* (UML)**

*Unified Modelling Language* (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual, Braun dalam (Haviluddin, 2011). UML merupakan alat perancangan sistem yang berorientasi pada objek. Secara filosofi kemunculan UML diilhami oleh konsep yang telah ada yaitu konsep permodelan *Object Oriented* (OO), karena konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh obyek dan digambarkan atau dinotasikan dalam simbol-simbol yang cukup spesifik maka OO memiliki proses standar dan bersifat independen. UML diagram memiliki tujuan utama untuk membantu tim pengembangan proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau membuat program. (Haviluddin, 2011).

#### **2.4.5. *Android Studio***

Menurut situs resminya (Android Studio, 2018) Android Studio adalah *Integrated Development Environment* (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA. Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas saat membuat aplikasi Android.

## 2.5. Alat Pemodelan Sistem yang Digunakan

Sub bab ini menjelaskan mengenai perangkat apa saja yang akan digunakan dalam memodelkan sistem yang akan dirancang.

### 2.5.1. BPMN (*Bussiness Process Model Notation*)

BPMN adalah suatu metode penyelarasan secara efisien suatu organisasi dengan keinginan dan kebutuhan organisasi. BPMN merupakan suatu pendekatan manajemen holistik untuk meningkatkan efektivitas dan efisiensi bisnis seiring upaya untuk mencapai inovasi, fleksibilitas, dan integrasi dengan teknologi. Berikut merupakan kelebihan dan kekurangan dari BPMN (Sari & Asniar, 2015) :

Kelebihan menggunakan BPMN adalah :

1. Dapat menggambarkan keseluruhan proses dalam satu diagram sederhana sehingga representasi proses bisnis relatif lebih cepat dipahami.
2. Mampu memodelkan aliran pesan.
3. Mampu memodelkan aliran proses secara sekuensial dari kejadian awal sampai hasil akhir.

Kekurangan menggunakan BPMN adalah :

1. Simbol-simbol yang ada pada BPMN terlalu *complicated* untuk diimplementasikan pada real transaksi di industri.
2. BPMN tidak bisa menggambarkan hasil dari proses dan model resiko, sehingga *Key Performance Indicator* (KPI) tidak bisa digambarkan menggunakan notasi BPMN.
3. Tidak bisa menggambarkan *conceptual modeling*, *business logic* dan detail dari aktivitas.

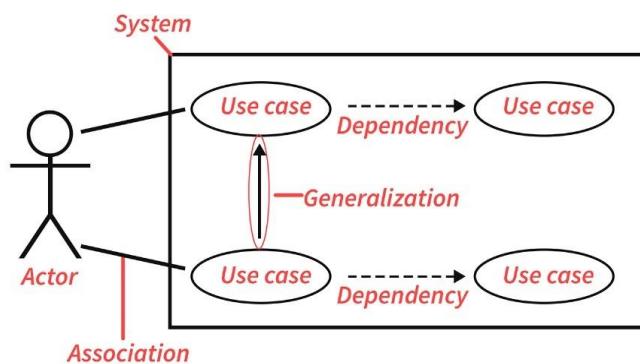
### 2.5.2. Use Case Diagram

Use Case Diagram adalah diagram yang menggambarkan aktor, use case dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah use case digambarkan sebagai *elips* horizontal dalam suatu diagram

UML use case (Ropianto, 2016). Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Hendini, 2016).

Berikut ini elemen-elemen yang digunakan pada use case diagram (Mulyani, 2016) :

- a. Sistem, merupakan batasan-batasan proses yang sudah kita deskripsikan dalam sebuah sistem.
- b. Aktor, elemen yang menjadi pemicu sistem. Aktor bisa berupa orang, mesin ataupun sistem lain yang berinteraksi dengan *use case*.
- c. *Use Case*, potongan proses yang merupakan bagian dari sistem.
- d. *Association*, menggambarkan interaksi antara use case dan aktor.
- e. *Dependency*, menggambarkan relasi (*relationship*) antara dua *use case*. Ada 2 (dua) tipe dari *dependency* yaitu, *include* dan *extends*. *Include* merupakan tipe dari *dependency* yang menghubungkan dua *use case* dimana, satu *use case* membutuhkan *use case* yang satunya sedangkan *extends* adalah tipe dari *dependency* yang menghubungkan dua *use case* dimana satu *use case* terkadang akan memanggil *use case* yang satunya, tergantung pada kondisi.
- f. *Generalization*, menggambarkan pewarisan antara dua aktor atau *use case* dimana salah satu aktor atau *use case* mewarisi *properties* ke aktor atau *use case* yang satunya. Elemen *use case diagram* dapat dilihat pada gambar 2.4 berikut.



Gambar 2.4 Elemen dari *Use Case Diagram* (Mulyani, 2016)

### 2.5.3. Use Case Scenario

Masing-masing *use case diagram* akan dijabarkan ke dalam *use case scenario*. Di dalam *use case scenario* akan diberi uraian nama *actor* yang berhubungan dengan *use case* tersebut, tujuan dari *use case*, deskripsi global tentang *use case*, pra-kondisi yang harus dipenuhi, dan pasca-kondisi yang diharapkan setelah berjalannya fungsional *use case*. Selain itu juga diberikan usulan yang berkaitan dengan tanggapan dari sistem atas suatu aksi yang diberikan oleh aktor, baik sebagai proses utama yang pasti dilakukan oleh sistem maupun proses alternatif (Stone, 2006).

### 2.5.4. Sequence diagram

*Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu. Diawali dari apa yang men-trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah *message* (Dharwiyanti & Wahono, 2003).

### 2.5.5. Entity Relationship Diagram

*Entity Relationship Diagram* (ERD) adalah diagram yang menggambarkan hubungan antara suatu entitas dengan dirinya sendiri maupun entitas lainnya. ERD digunakan untuk mengidentifikasi objek-objek yang terdapat dalam suatu sistem dan menggambarkan *file-file* dan tabel-tabel yang terdapat dalam sistem (Timotius Witono & Raphael Susanto, 2012).

ERD hanya berfokus pada data, dengan menunjukkan jaringan data yang ada untuk suatu sistem yang diberikan. ERD sangat berguna bagi aplikasi di mana data dan hubungan yang mengatur data sangatlah kompleks. ERD pada mulanya diusulkan oleh Peter Chen untuk desain sistem *database* relasional dan telah dikembangkan oleh yang lainnya. Serangkaian komponen utama diidentifikasi untuk ERD yaitu objek data, atribut, hubungan, dan berbagai tipe indikator. Tujuan utama dari ERD adalah untuk mewakili objek data dan hubungan mereka (Widianti, 2012).



## **BAB III**

### **METODOLOGI PENELITIAN**

Bab ini berisi objek penelitian, metode pengumpulan data, metode pengembangan sistem, dan *flowchart* metodologi penelitian yang digunakan.

#### **3.1. Objek Penelitian**

Objek kajian dalam penelitian adalah sistem pemesanan dan transaksi pada *Cafe* Deco Boco Yatai yang berlokasi di Jalan Dr.M.Hatta Kelurahan Cupak Tangah Kecamatan Pauh Kota Padang Provinsi Sumatera Barat. Pemilihan tempat tersebut sebagai objek penelitian ini dikarenakan menurut observasi yang dilakukan, Deco Boco Yatai merupakan *Cafe* yang ramai dikunjungi karena lokasinya yang strategis yaitu terletak di dekat wilayah kampus. Hal lain yang dipertimbangkan dalam pemilihan *Cafe* Deco Boco Yatai untuk penerapan sistem *e-menu* dan pembayaran melalui *QR Code* ini adalah pelayanannya yang masih lama jika pengunjung ramai dan transaksi pembayarannya yang masih bersifat manual dan belum terkomputerisasi.

#### **3.2. Metode Pengumpulan Data**

Pada penelitian ini metode pengumpulan data yang dilakukan adalah dengan studi lapangan dan studi literatur, dimana studi lapangan terdiri dari wawancara, observasi, dan analisis dokumen. Berikut penjelasan masing-masingnya :

a. Wawancara

Wawancara dilakukan untuk mendapatkan informasi lebih detail tentang proses bisnis yang ada pada objek penelitian. Pada penelitian ini wawancara dilakukan dengan melakukan tanya jawab dengan pemilik *Cafe* Deco Boco Yatai.

b. Observasi

Observasi atau survei dilakukan dengan cara pengamatan langsung terhadap objek penelitian yang telah ditetapkan. Seperti alur proses bisnis yang ada serta cara kerja aktor/karyawan yang terlibat.

c. Analisis Dokumen

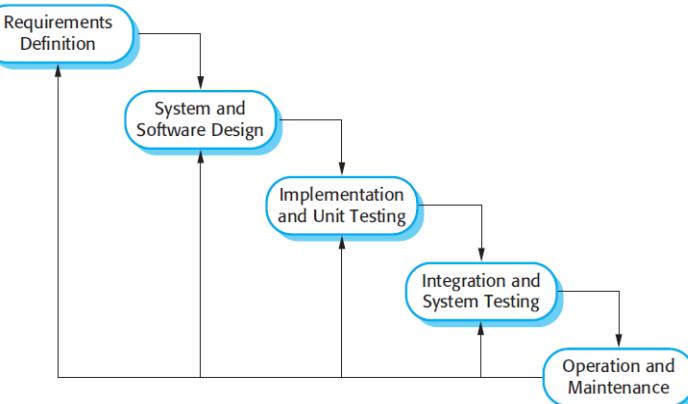
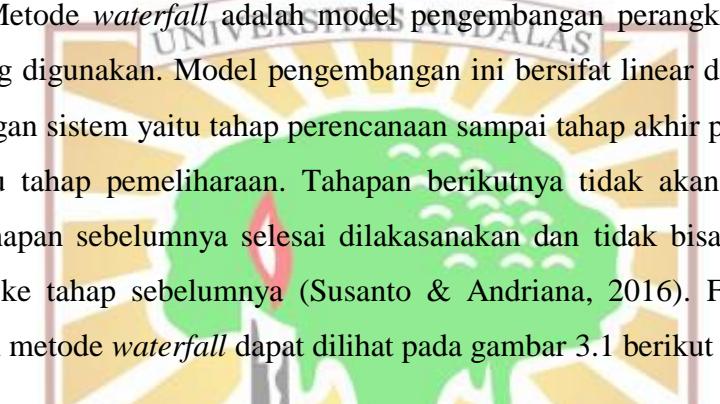
Analisis dokumen dilakukan dengan cara mempelajari dokumen-dokumen yang berkaitan dengan sistem yang akan dibangun.

d. Studi Literatur

Studi literatur dilakukan dengan mempelajari dan memahami literatur dari berbagai sumber seperti situs internet, jurnal ilmiah, dan sumber lain yang mendukung penelitian.

### 3.3. Metode Pengembangan Sistem

Metode yang digunakan untuk membangun aplikasi ini adalah metode *waterfall*. Metode *waterfall* adalah model pengembangan perangkat lunak yang paling sering digunakan. Model pengembangan ini bersifat linear dari tahap awal pengembangan sistem yaitu tahap perencanaan sampai tahap akhir pengembangan sistem yaitu tahap pemeliharaan. Tahapan berikutnya tidak akan dilaksanakan sebelum tahapan sebelumnya selesai dilaksanakan dan tidak bisa kembali atau mengulang ke tahap sebelumnya (Susanto & Andriana, 2016). Fase-fase pada pelaksanaan metode *waterfall* dapat dilihat pada gambar 3.1 berikut :



Gambar 3.1 Metode Waterfall (Sommerville, 2011)

Dalam pembangunan sistem *e-menu* menggunakan teknologi *QR Code* ini, tahapan yang dilalui hanya sampai pada tahap ketiga yaitu pada tahap

*Implementation and Unit Testing.* Penjelasan dari setiap tahapan adalah sebagai berikut :

1. *Requirements Analysis and Definition*

Pada tahap ini dilakukan analisis dan pengumpulan data yang diperlukan terhadap kebutuhan sistem dengan cara wawancara, observasi, analisis dokumen, dan studi literatur yang berkaitan dengan pembangunan sistem *e-menu*. Pada tahap ini juga akan diidentifikasi permasalahan yang terjadi pada sistem pemesanan dan transaksi *Cafe Deco Boco Yatai* , lalu dilakukan analisa dan evaluasi. Proses analisis ini digambarkan dengan BPMN (*Business Process Modelling System*).

2. *System and Software Design*

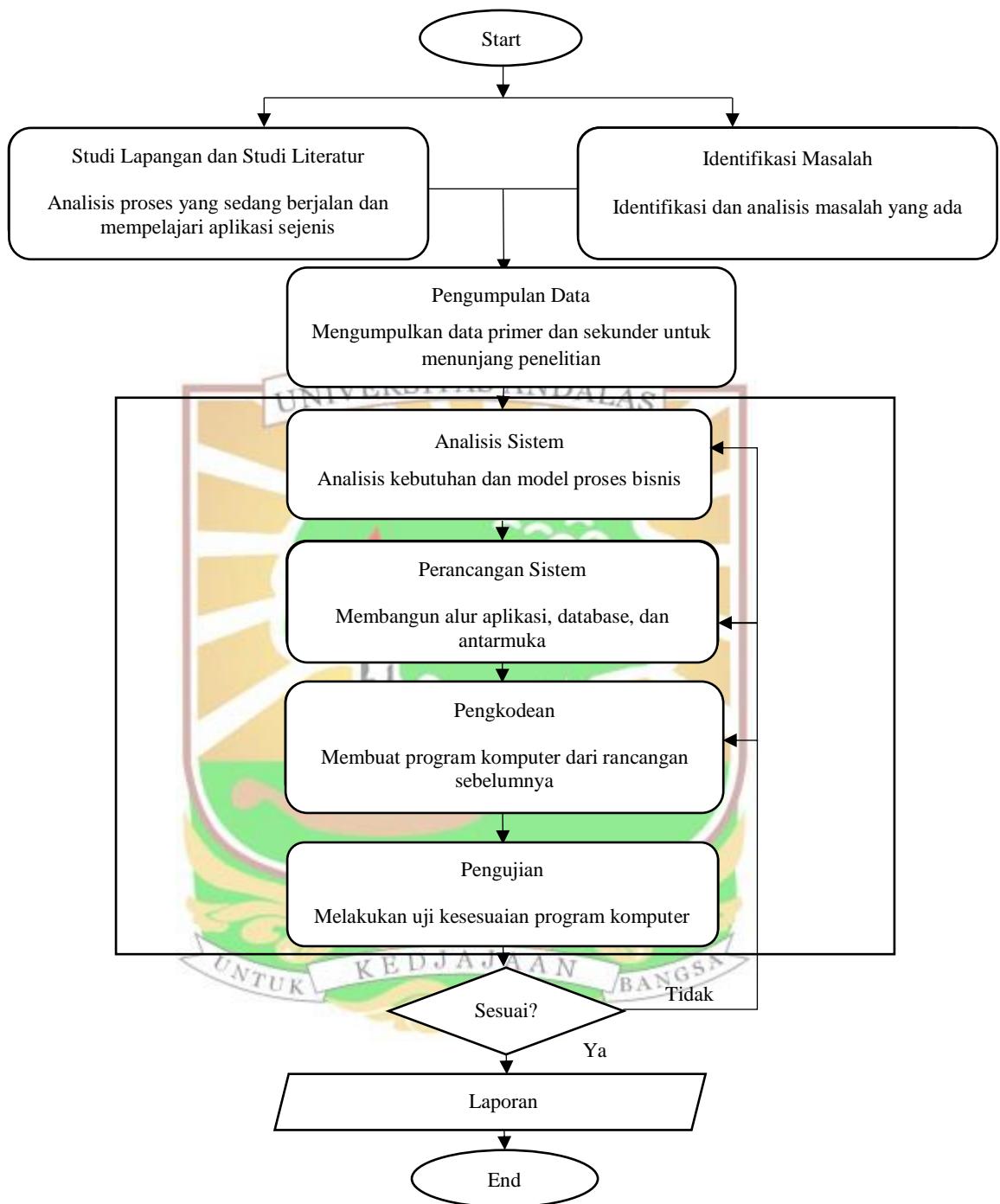
Tahapan ini mengalokasikan kebutuhan -kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Data yang didapat selama tahap analisis akan diolah sehingga menjadi rancangan sistem baru dengan fungsional yang telah dirumuskan sebelumnya. Tahapan ini terdiri dari perancangan *database* , arsitektur perangkat lunak, dan perancangan antarmuka.

3. *Implementation and Unit Testing*

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. *Software requirement* yang dihasilkan pada tahap perancangan diterjemahkan ke dalam bahasa pemrograman dalam hal ini penulis menggunakan bahasa PHP untuk aplikasi *web* dan *Java* untuk aplikasi berbasis *mobile*. Setelah tahap *coding* selesai dilanjutkan dengan tahap pengujian terhadap aplikasi yang dirancang untuk memastikan aplikasi telah memenuhi persyaratan dan menghasilkan keluaran yang sesuai dengan kebutuhan.

#### 3.4. *Flowchart* Metodologi Penelitian

*Flowchart* pada penelitian ini menjelaskan langkah-langkah yang dilakukan pada pembangunan aplikasi *e-menu* menggunakan teknologi *QR Code* berbasis *web* dan *mobile*. *Flowchart* penelitian dapat dilihat pada gambar 3.2 :



Gambar 3.2 *Flowchart* Penelitian

Berikut adalah penjelasan dari *flowchart* penelitian di atas :

1. Studi Lapangan dan Studi Literatur

Pada tahap ini dilakukan analisis proses pemesanan makanan dan transaksi yang sedang berjalan. Selanjutnya memahami aplikasi sejenis yang pernah dikembangkan sebelumnya.

## 2. Identifikasi Masalah

Pada tahap ini dilakukan identifikasi dan analisis masalah pemesanan makanan pada *Cafe Deco Boco Yatai*.

## 3. Pengumpulan Data

Mengumpulkan data primer dan data sekunder yang berguna untuk menunjang penelitian.

## 4. Analisis sistem

Pada tahap ini dilakukan analisis kebutuhan *user* dan pemodelan proses bisnis menggunakan BPMN.

## 5. Perancangan sistem

Pada tahap ini dilakukan perancangan sistem yang meliputi perancangan *database* , perancangan *user interface*, dan arsitektur IT untuk pembangunan aplikasi.

## 6. Pengkodean

Pada tahapan ini direalisasikan rancangan aplikasi dengan bahasa pemrograman PHP, aplikasi Android Studio, dan *database* MySQL.

## 7. Pengujian

Pada tahap ini dilakukan pengujian sistem terhadap aplikasi yang telah dibangun. Pengujian berguna untuk melihat apakah perangkat lunak yang dibuat dengan bahasa pemrograman telah sesuai dengan persoalan. Pengujian kesesuaian kebutuhan fungsional dengan aplikasi yang dirancang dilakukan dengan menggunakan *blackbox testing*.

## 8. Laporan

Pada tahap ini dilakukan penyusunan laporan dari tahap awal hingga tahap akhir serta kesimpulan dan saran.

## **BAB IV**

### **ANALISIS DAN PERANCANGAN SISTEM**

Pada bab ini dijelaskan hasil dari analisis dan perancangan aplikasi aplikasi pemesanan menu makanan secara *online (e-menu)* menggunakan teknologi *QR-Code* berbasis *web* dan *mobile* pada *Cafe Deco Boco Yatai*.

#### **4.1. Analisis Sistem**

Analisis sistem dilakukan untuk menganalisis sistem pemesanan menu makanan dan pembayaran pada *Cafe Deco Boco* dengan menggunakan UML (*Unified Modelling Language*). UML yang digunakan untuk analisis sistem ini adalah *use case diagram* dan *use case scenario*.

##### **4.1.1. Kedudukan Sistem**

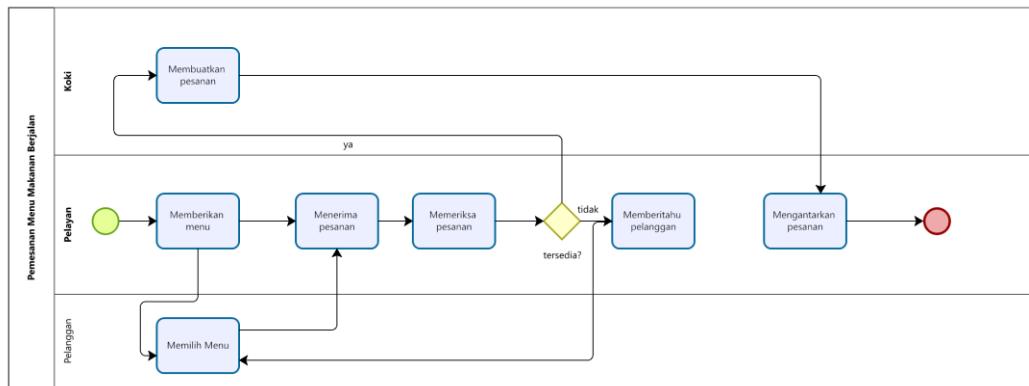
Sistem yang ada pada *Cafe Deco Boco Yatai* pada saat ini terdiri dari sistem pemesanan menu makanan dan sistem pembayaran. Sistem ini masih dijalankan secara manual dan belum terkomputerisasi, akibatnya sering terjadi kesalahan dan lambatnya pelayanan pada *cafe*. Pada penelitian ini dirancang aplikasi pemesanan menu makanan secara *online (e-menu)* untuk mempermudah pelayanan dan pengelolaan transaksi *cafe*, aplikasi yang dibuat berbasis *web* dan *mobile* dengan memanfaatkan teknologi *QR Code* untuk metode pembayaran secara non tunai khusus untuk pelanggan yang telah terdaftar sebagai *member*.

##### **4.1.2. Sistem yang Berjalan**

Proses bisnis yang sedang berjalan pada *Cafe Deco Boco* pada saat ini meliputi proses pemesanan makanan dan proses pembayaran. Proses tersebut akan digambarkan dengan BPMN seperti berikut :

a. Proses Pemesanan Menu Makanan

Proses ini dilakukan oleh tiga aktor, yaitu pelanggan, pelayan, dan koki. Model proses bisnis pemesanan menu makanan yang sedang berjalan dapat dilihat pada Gambar 4.1 berikut.

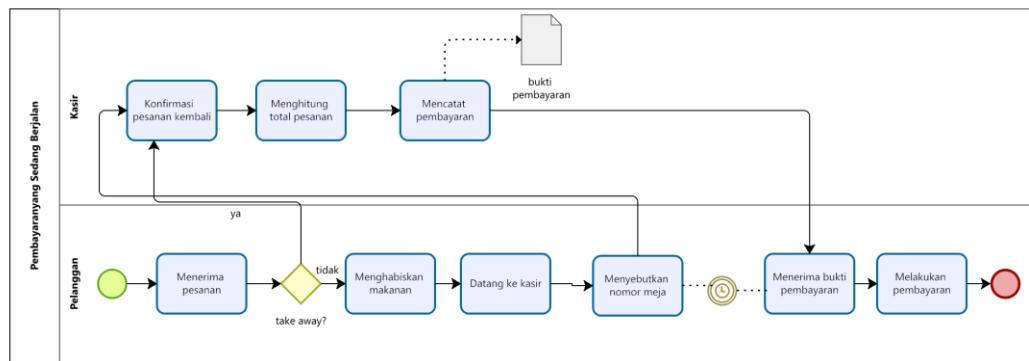


Gambar 4.1 Proses Pemesanan Menu Makanan yang Sedang Berjalan

Berikut ini penjelasan Gambar 4.1 mengenai proses pemesanan menu makanan yang sedang berjalan :

1. Pelanggan datang ke *cafe*
  2. Pelanggan memilih meja, lalu pelayan *memberikan* menu ke meja pelanggan
  3. Pelanggan mencatat sendiri apa yang ingin dipesan beserta jumlahnya
  4. Jika pelanggan telah selesai mencatat pesanan, pelanggan menyerahkannya ke pelayan
  5. Pelayan menerima dan memeriksa pesanan apakah tersedia atau tidak
  6. Jika pesanan tersedia, pelayan menyerahkan daftar pesanan kepada koki dan koki akan membuat pesanan
  7. Jika pesanan tidak tersedia, pelayan akan *memberitahu* pelanggan dan pelanggan akan memilih menu lain dan menyerahkan kembali kepada pelayan
  8. Setelah koki selesai membuat pesanan, pelanggan akan mengantarkan pesanan ke meja pelanggan
  9. Pelanggan menerima pesanan
- b. Proses Pembayaran

Proses ini dilakukan oleh dua aktor, yaitu pelanggan dan kasir. Model proses bisnis proses pembayaran dapat dilihat pada Gambar 4.2 berikut.



Gambar 4.2 Proses Pembayaran yang Sedang Berjalan

Berikut ini penjelasan Gambar 4.2 mengenai proses pembayaran yang sedang berjalan :

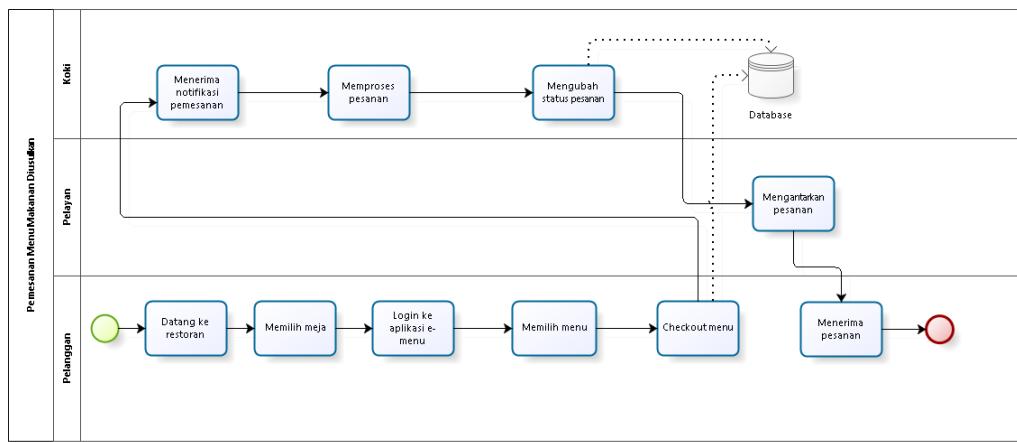
1. Pelanggan menerima pesanan
2. Setelah pelanggan menghabiskan makanan, lalu pelanggan menuju ke meja kasir
3. Pelanggan menyebutkan nomor meja
4. Kasir akan mengkonfirmasi kembali pesanan yang telah dipesan pelanggan
5. Kasir menghitung total pembayaran
6. Kasir mencatat pembayaran
7. Setelah menerima total pembayaran, pelanggan melakukan pembayaran

#### 4.1.3. Sistem yang Diusulkan

Sistem pemesanan dan pembayaran yang diusulkan dalam analisis sistem ini, dimodelkan dengan menggunakan Business Process Model and Notation (BPMN). Terdapat dua proses bisnis yang diusulkan. Berikut merupakan BPMN yang diusulkan pada pembangunan aplikasi *e-menu* pada *Cafe Deco Boco Yatai* dengan menggunakan teknologi *QR Code*.

##### a. BPMN Proses Pemesanan Menu Makanan

Pada bagian ini dijelaskan bagaimana proses pemesanan makanan yang diusulkan dalam aplikasi *e-menu* pada *Cafe Deco Boco Yatai*. Proses bisnis yang diusulkan dimodelkan dengan menggunakan BPMN seperti pada Gambar 4.3 berikut.

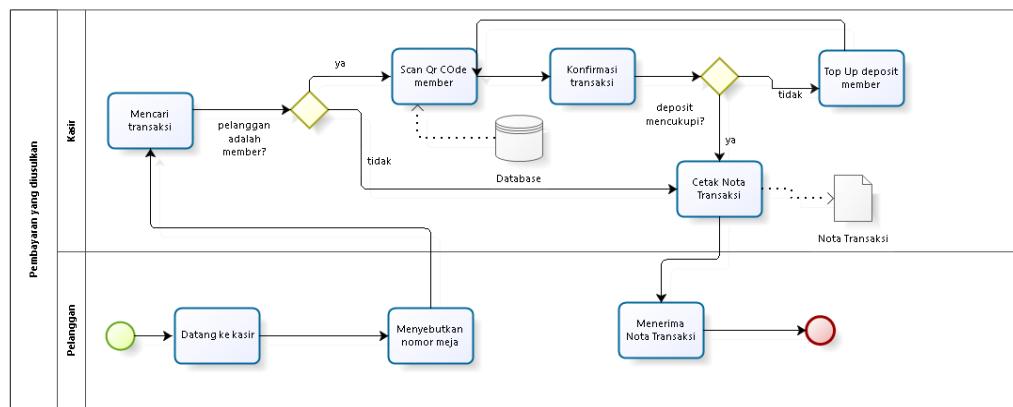


Gambar 4.3 Proses Pemesanan yang Diusulkan

Berikut merupakan penjelasan dari Gambar 4.3 tentang usulan proses pemesanan menu makanan pada pembangunan aplikasi *e-menu* pada *Cafe Deco Boco Yatai* :

1. Proses diawali dengan pelanggan datang ke *cafe*
  2. Selanjutnya pelanggan memilih meja
  3. Setelah pelanggan memilih meja, pelanggan bisa *login* ke aplikasi *e-menu* menggunakan *username* dan *password* yang telah terdaftar.
  4. Selanjutnya pelanggan memilih meja dan menu yang tersedia di aplikasi *e-menu*
  5. Setelah selesai memilih menu yang akan dipesan, pelanggan menekan tombol *checkout*
  6. Koki atau bagian dapur menerima notifikasi pemesanan
  7. Jika makanan telah selesai diproses, koki akan mengubah status pesanan menjadi selesai dan pelayan akan mengantarkan pesanan kepada pelanggan
- b. BPMN Proses Pembayaran (Khusus *Member*)

Pada bagian ini dijelaskan bagaimana proses pembayaran yang diusulkan dalam aplikasi *e-menu* pada *Cafe Deco Boco Yatai*. Proses bisnis yang diusulkan dimodelkan dengan menggunakan BPMN seperti pada Gambar 4.4 berikut.



Gambar 4.4 Proses Pembayaran yang Disusulkan

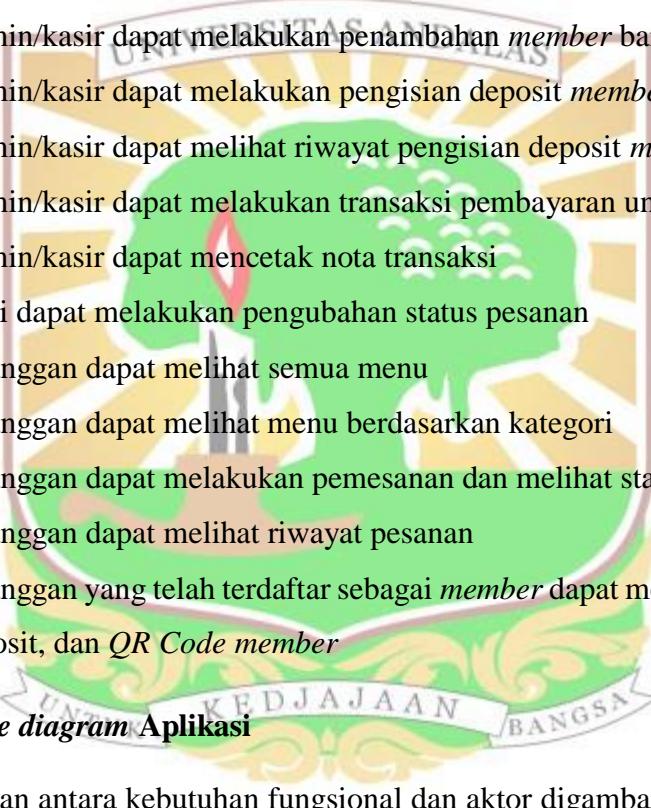
Berikut ini merupakan penjelasan dari gambar tentang usulan proses pembayaran khusus *member* di *Cafe Deco Boco Yatai* :

1. Proses diawali dengan pelanggan datang ke kasir
2. Pelanggan menyebutkan nomor meja dan kemudian kasir akan mencari transaksi
3. Jika pelanggan adalah *member* kasir akan melakukan scan *QR Code member* dan jika pelanggan belum terdaftar sebagai *member*, maka kasir akan langsung mencetak nota transaksi
4. Setelah melakukan scan *QR Code*, kasir akan melakukan konfirmasi transaksi
5. Jika saldo deposit *member* mencukupi kasir akan mencetak nota transaksi dan pelanggan menerima nota transaksi sebagai bukti pembayaran
6. Jika saldo deposit *member* tidak mencukupi maka kasir akan melakukan *top up* terlebih dahulu.

#### 4.1.4. Analisis Kebutuhan Fungsional

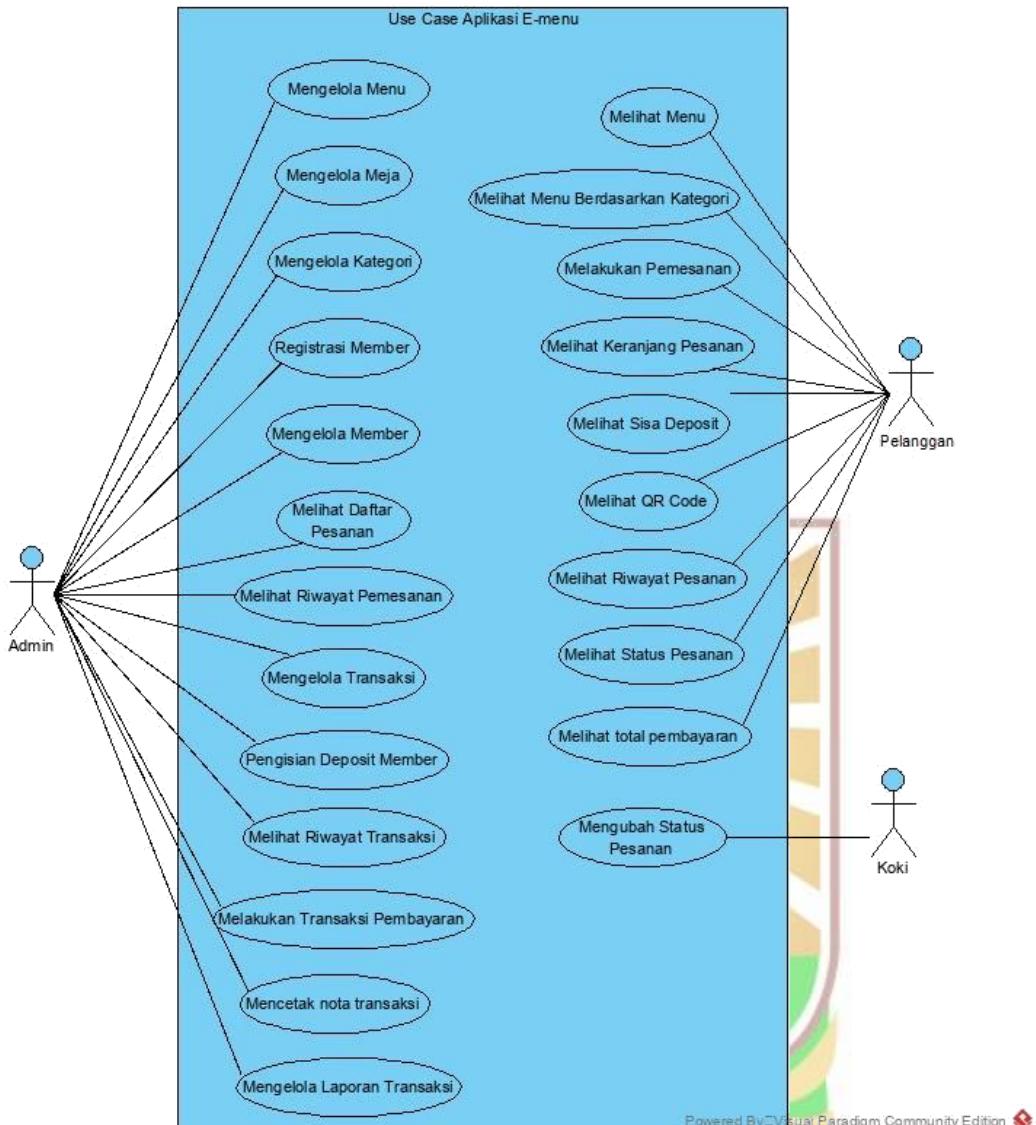
Proses pengumpulan data dan proses bisnis yang diusulkan menghasilkan kebutuhan fungsional. Berdasarkan analisis alur proses aplikasi *e-menu* dengan pemanfaatan teknologi *QR Code* yang diusulkan, maka dirumuskan beberapa kebutuhan fungsional, yaitu :

1. Admin/kasir dapat melihat seluruh data *member*, meja, menu, kategori menu.

- 
2. Admin/kasir dapat mengelola data *member*
  3. Admin/kasir dapat mengelola data meja
  4. Admin/kasir dapat mengelola data menu
  5. Admin/kasir dapat mengelola data kategori menu
  6. Admin/kasir dapat melihat daftar pesanan
  7. Admin/kasir dapat melihat riwayat pesanan
  8. Admin/kasir dapat melihat daftar transaksi
  9. Admin/kasir dapat melihat riwayat transaksi
  10. Admin/kasir dapat melihat dan mencetak laporan transaksi
  11. Admin/kasir dapat melakukan penambahan *member* baru
  12. Admin/kasir dapat melakukan pengisian deposit *member*
  13. Admin/kasir dapat melihat riwayat pengisian deposit *member*
  14. Admin/kasir dapat melakukan transaksi pembayaran untuk *member*
  15. Admin/kasir dapat mencetak nota transaksi
  16. Koki dapat melakukan pengubahan status pesanan
  17. Pelanggan dapat melihat semua menu
  18. Pelanggan dapat melihat menu berdasarkan kategori
  19. Pelanggan dapat melakukan pemesanan dan melihat status pesanan
  20. Pelanggan dapat melihat riwayat pesanan
  21. Pelanggan yang telah terdaftar sebagai *member* dapat melihat profil, sisa deposit, dan *QR Code member*

#### 4.1.5. *Use case diagram* Aplikasi

Hubungan antara kebutuhan fungsional dan aktor digambarkan dengan *use case diagram*. Rancangan *use case diagram* pembangunan aplikasi *e-menu* dapat dilihat pada Gambar 4.5.



Gambar 4.5 Use Case Diagram Perancangan Aplikasi

Berdasarkan *use case diagram* perancangan aplikasi pada Gambar 4.5 , didapatkan tiga aktor yang terlibat dalam aplikasi ini, dimana dua aktor untuk aplikasi *web*, yaitu admin dan koki, dan satu aktor untuk aplikasi *mobile* yaitu pelanggan. Masing-masing aktor harus *login* terlebih dahulu untuk bisa masuk ke aplikasi. Terdapat sepuluh fungsional pada aplikasi *web* yang memiliki hubungan sesuai dengan fungsi dari masing-masing aktor. Aktor admin memiliki sembilan fungsional, yaitu mengelola menu, mengelola meja, mengelola kategori, melakukan registrasi *member*, mengelola data *member*, melihat pesanan, mengelola transaksi, melihat riwayat transaksi dan melakukan pengisian deposit *member*. Sedangkan

aktor koki hanya memiliki satu fungsional, yaitu mengubah status pesanan. Untuk aktor pelanggan terdapat tujuh fungsional, yaitu melihat semua menu yang ada di *cafe*, melihat menu berdasarkan kategori, melakukan pemesanan, melihat keranjang pesanan, melihat riwayat pesanan, melihat *QR Code member* dan melihat sisa deposit.

#### 4.1.6. Deskripsi Tugas Aktor

Terdapat tiga aktor dari proses pemesanan dan pembayaran pada Café Deco Boco Yatai ini, yaitu admin, koki, dan pelanggan. Deskripsi peran dari masing-masing aktor dapat dilihat pada Tabel 4.1.

Tabel 4.1 Deskripsi Tugas Aktor

No	Aktor	Deskripsi Tugas
1	Admin	<ul style="list-style-type: none"> <li>• Melakukan registrasi <i>member</i> baru</li> <li>• Mengelola data <i>member</i> Admin dapat melihat data <i>member</i>, menambah data <i>member</i> meng-edit data <i>member</i> dan menghapus data <i>member</i></li> <li>• Mengelola data meja Admin dapat menambah, melihat, meng-edit dan menghapus data meja</li> <li>• Mengelola data menu makanan/minuman Admin dapat menambah, melihat, meng-edit dan menghapus data makanan/minuman</li> <li>• Mengelola data kategori menu Admin dapat menambah, melihat, meng-edit dan menghapus data kategori menu</li> <li>• Melihat pesanan</li> </ul>

Tabel 4.1 Deskripsi Tugas Aktor

		<p>Admin dapat melihat data pesanan yang baru dipesan pelanggan, data pesanan yang sedang berjalan, dan data pesanan yang telah selesai</p> <ul style="list-style-type: none"> <li>• Mengelola transaksi pembayaran</li> </ul> <p>Admin dapat melakukan transaksi pembayaran secara tunai atau non tunai khusus <i>member</i> melalui <i>scanning QR Code member</i></p> <ul style="list-style-type: none"> <li>• Melakukan penambahan deposit <i>member</i></li> </ul> <p>Admin dapat melakukan penambahan deposit <i>member</i></p>
2	Koki	<ul style="list-style-type: none"> <li>• Mengelola pesanan</li> </ul> <p>Koki dapat melihat dan mengubah status pesanan</p>
3	User	<ul style="list-style-type: none"> <li>• Melihat daftar menu</li> </ul> <p><i>User</i> yang telah <i>login</i> dapat melihat seluruh daftar menu yang ada</p> <ul style="list-style-type: none"> <li>• Melihat daftar menu berdasarkan kategori</li> </ul> <p><i>User</i> dapat melihat menu berdasarkan kategorinya</p> <ul style="list-style-type: none"> <li>• Melihat keranjang pesanan</li> <li>• Melakukan pemesanan</li> </ul> <p><i>User</i> dapat melakukan pemesanan makanan atau minuman</p> <ul style="list-style-type: none"> <li>• Melihat total pembayaran</li> <li>• Melihat sisa deposit <i>member</i></li> <li>• Menampilkan <i>QR Code member</i></li> </ul>

#### 4.1.7. Use case scenario Aplikasi

Rancangan *use case* yang telah dibuat dijelaskan dengan *use case scenario*. Pada bagian ini membahas beberapa skenario dari *use case* yang terdapat dalam aplikasi *e-menu* ini, yaitu pemesanan makanan oleh pelanggan, mengubah status pesanan, pembayaran khusus *member*. *Use Case Scenario* lainnya dapat dilihat pada lampiran A.

##### 4.1.7.1. Use Case Scenario Melakukan Pemesanan Melalui Aplikasi Mobile

*Use case scenario* melakukan pemesanan melalui aplikasi *mobile* merupakan alur proses yang dapat dilakukan oleh aktor pelanggan dengan kondisi awal pelanggan telah *login* ke aplikasi. Skenario *use case* melakukan pemesanan melalui aplikasi *mobile* dapat dilihat pada Tabel 4.2 berikut.

Tabel 4.2 Use Case Scenario Melakukan Pemesanan

<i>Use case</i>	Melakukan Pemesanan
<i>Actor</i>	Pelanggan
<i>Entry Condition</i>	<i>Actor</i> telah <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"><li><i>Actor</i> memilih halaman “Order”</li><li>Sistem menampilkan data meja</li><li><i>Actor</i> memilih memilih salah satu meja</li><li>Sistem menampilkan halaman untuk memilih menu</li><li><i>Actor</i> menekan tombol menu dan memilih menu ke</li><li>Sistem menampilkan halaman menu yang sudah dipilih</li><li><i>Actor</i> menekan tombol “Checkout” pada halaman “Detail Pesanan”</li><li>Sistem menampilkan halaman “Daftar Pesanan” dan total pembayaran</li></ol>

Tabel 4.2 Use Case Scenario Melakukan Pemesanan

<i>Scenario Alternatif</i>	9. Actor memesan menu yang tidak tersedia 10. Sistem tidak menjalankan proses pemesanan dan menampilkan notifikasi “Menu Tidak Tersedia”
<i>Exit Condition</i>	Sistem menampilkan notifikasi transaksi pesanan berhasil dan menampilkan halaman transaksi

#### 4.1.7.2. *Use case scenario* Mengubah Status Pesanan

*Use case scenario* mengubah status pesanan merupakan alur proses yang dapat dilakukan oleh aktor koki dengan kondisi awal dimana koki telah *login*. Skenario *use case* mengubah status pesanan melalui *web* dapat dilihat pada Tabel 4.3 berikut.

Tabel 4.3 Use Case Scenario Mengubah Status Pesanan

<i>Use case</i>	Mengubah Status Pesanan
<i>Actor</i>	Koki
<i>Entry Condition</i>	Actor telah <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Actor membuka menu “Pesanan”</li> <li>2. Sistem menampilkan submenu pesanan</li> <li>3. Actor memilih submenu “Pesanan Baru”</li> <li>4. Sistem membaca data pesanan di <i>database</i> dan menampilkan halaman daftar pesanan yang baru masuk</li> <li>5. Actor menekan tombol “Proses” pada pesanan yang dipilih</li> <li>6. Sistem menampilkan halaman “Pesanan Berlangsung”</li> <li>7. Actor menekan tombol “Selesai” pada pesanan yang dipilih</li> </ol>

Tabel 4.3 Use Case Scenario Mengubah Status Pesanan

	8. Sistem menampilkan halaman “Pesanan Selesai”
<i>Exit Condition</i>	Sistem menampilkan notifikasi status pesanan dan menampilkan halaman pesanan

#### 4.1.7.3. *Use case scenario* Transaksi Pembayaran

*Use case scenario* transaksi pembayaran merupakan alur proses yang dapat dilakukan oleh aktor admin dengan kondisi awal dimana admin telah *login*. Skenario *use case* dari transaksi pembayaran melalui *web* dapat dilihat pada Tabel 4.4 berikut.

Tabel 4.4 Use Case Scenario Transaksi Pembayaran

<i>Use case</i>	Transaksi Pembayaran
<i>Actor</i>	Admin
<i>Entry Condition</i>	Actor telah <i>login</i>
<i>flow of Event</i>	<ol style="list-style-type: none"> <li>1. Actor membuka menu “Transaksi”</li> <li>2. Sistem menampilkan submenu transaksi</li> <li>3. Actor memilih submenu “Order Berjalan”</li> <li>4. Sistem membaca data transaksi di <i>database</i> dan menampilkan halaman daftar <i>order</i> yang sedang berjalan</li> <li>5. Actor menekan tombol “Pembayaran” pada transaksi yang dipilih</li> <li>6. Sistem menampilkan halaman <i>scanning</i> “Pembayaran”</li> <li>7. Actor melakukan <i>scanning QR Code</i> member yang melakukan transaksi</li> </ol>

Tabel 4.4 Use Case Scenario Transaksi Pembayaran

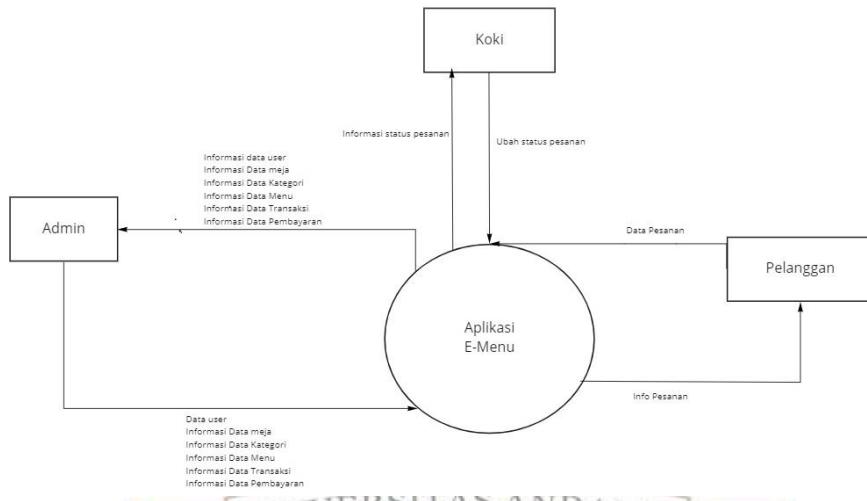
	<p>8. Sistem membaca data <i>member</i> di <i>database</i> dan menampilkan notifikasi transaksi pembayaran sukses</p> <p>9. Sistem menampilkan halaman daftar transaksi selesai</p>
<i>Scenario Alternatif</i>	<p>10. <i>Actor</i> melakukan <i>scanning QR Code</i> yang tidak terdaftar</p> <p>11. Sistem gagal membaca data <i>member</i> di <i>database</i> dan menampilkan notifikasi “Data Tidak ditemukan”</p>
<i>Exit Condition</i>	Sistem menampilkan notifikasi transaksi pembayaran berhasil dan menampilkan halaman cetak nota transaksi

## 4.2. Perancangan Sistem

Perancangan sistem meliputi *context diagram*, *data flow diagram* (DFD), perancangan *database*, struktur tabel dan basis data, arsitektur aplikasi antarmuka (*user interface*).

### 4.2.1. Context Diagram

Context diagram merupakan gambaran menyeluruh dari data flow diagram (DFD). Pada context diagram ini terdapat entitas admin dan SPKT, juga proses berupa aplikasi pengelolaan laporan kriminal, dan adanya aliran data berupa masukkan (input) dan keluaran (output). Rancangan context diagram aplikasi pengelolaan laporan kriminal pada Polsek Pauh Padang dapat dilihat pada Gambar 4.6.



Gambar 4.6 *Context Diagram*

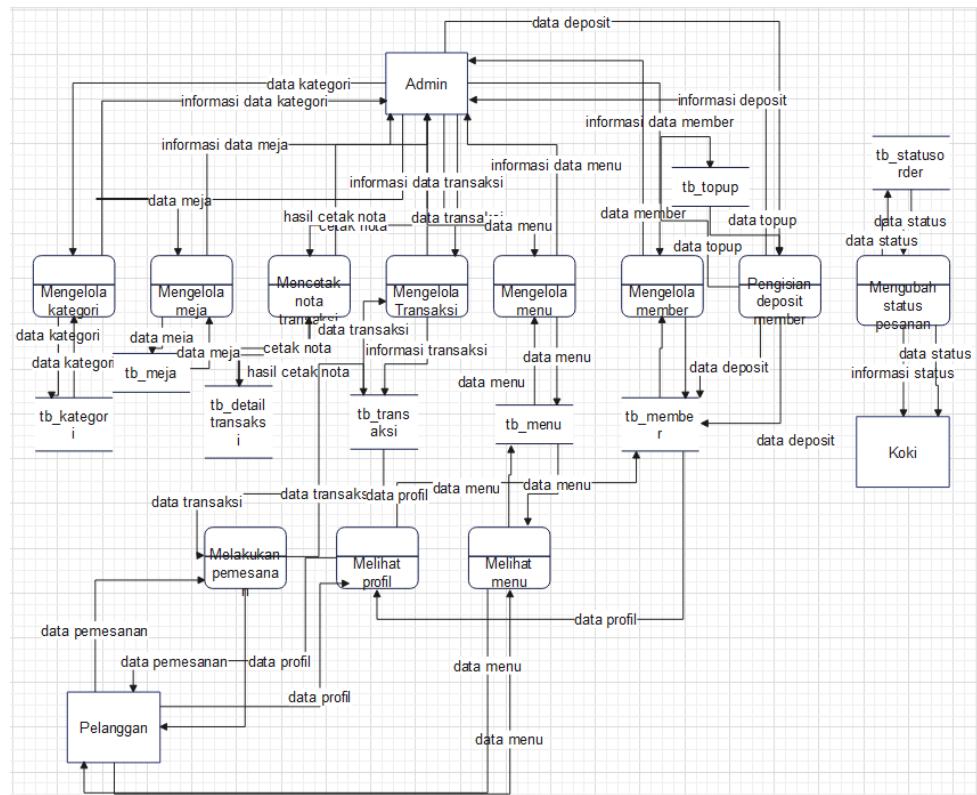
Berdasarkan Gambar 4.6 terdapat tiga entitas, yaitu admin, koki dan pelanggan. Selain itu terdapat proses berupa aplikasi pengelolaan transaksi serta adanya aliran data berupa input (masukkan) menuju proses dan output (keluaran) menuju entitas.

#### 4.2.2. Data Flow Diagram

Data flow diagram (DFD) adalah metode untuk membuat sebuah perancangan sistem yang berorientasi pada alur data sistem. Pemodelan dalam DFD terdiri dari terminator yang merupakan external entity yang memberikan input ke sistem, proses yang merupakan aktivitas mengolah input menjadi output, data flow yang merupakan aliran data pada sistem, dan data store yang merupakan penyimpanan data pada database.

##### 4.2.2.1. Data Flow Diagram Level 0

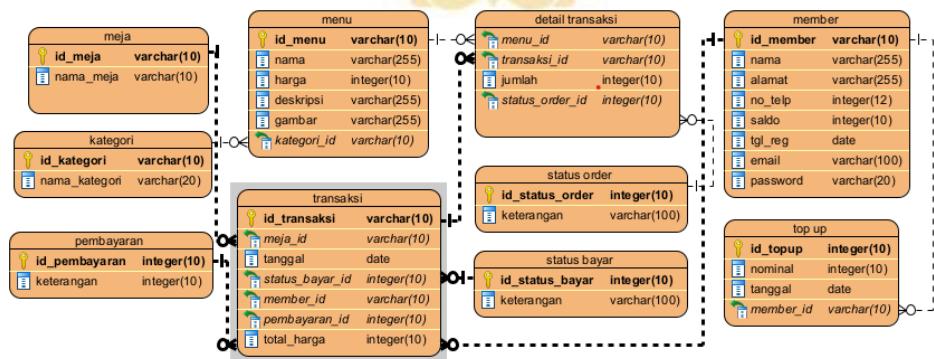
Pada DFD level 0, sistem dibagi menjadi 11 proses yang masing masing proses saling berhubungan dengan entitas maupun tabel. Perancangan DFD level 0 dapat dilihat pada Gambar 4.7.



Gambar 4.7 Data Flow Diagram Level 0

### 4.2.3. Perancangan *Database*

Perancangan *database* pada aplikasi ini terdiri dari sepuluh tabel, yaitu tabel meja, tabel menu, tabel kategori, tabel status\_order, tabel status\_bayar, tabel pembayaran, tabel *member*, tabel *topup*, tabel transaksi dan tabel *detail\_transaksi*. Perancangan dapat dilihat pada Gambar 4.8.



Gambar 4.8 Rancangan ERD Sistem

Berikut merupakan struktur *database* dari tabel menu, tabel transaksi, dan tabel detail\_transaksi yang dijelaskan pada Tabel 4.5, 4.6, dan 4.7. Untuk struktur *database* tabel lainnya dapat dilihat pada lampiran B

Tabel 4.5 Tabel Menu

Nama atribut	Tipe data	Keterangan
Id_menu	Varchar (10)	PK
Nama	Varchar (255)	
Harga	Integer (10)	
Deskripsi	Varchar (255)	
Gambar	Varchar (255)	
Kategori_id	Integer (10)	FK

Tabel 4.6 Tabel Transaksi

Nama atribut	Tipe data	Keterangan
Id_transaksi	Varchar (10)	PK
Meja_id	Varchar (10)	FK
Tanggal	Date	
Status_bayar_id	Integer (10)	FK
Pembayaran_id	Integer (10)	FK
Member_id	Integer (10)	FK
Total_harga	Integer (10)	

Tabel 4.7 Tabel *Detail Transaksi*

Nama atribut	Tipe data	Keterangan
Menu_id	Varchar (10)	PK
Transaksi_id	Varchar (10)	PK
Jumlah	Integer (10)	
Catatan	Varchar (255)	
Status_order_id	Varchar (255)	FK

#### 4.2.4. Arsitektur Aplikasi

Arsitektur aplikasi mendeskripsikan struktur dan interaksi aplikasi sebagai kelompok kemampuan yang menyediakan fungsi bisnis utama dan mengelola aset data. Arsitektur aplikasi, tujuannya adalah mendefinisikan berbagai jenis sistem aplikasi utama yang diperlukan untuk memproses data dan bisnis, tidak berhubungan dengan rancangan sistem aplikasi. Perancangan dan pembangunan arsitektur aplikasi ini menggunakan arsitektur pemrograman prosedural dengan PHP native. Pemodelan prosedural berisi perintah yang akan dilakukan oleh komputer. Pemodelan ini semua data dan kode program disatukan menjadi satu program. Arsitektur aplikasi ini dapat dilihat pada Gambar 4.9.



Gambar 4.9 Arsitektur Aplikasi

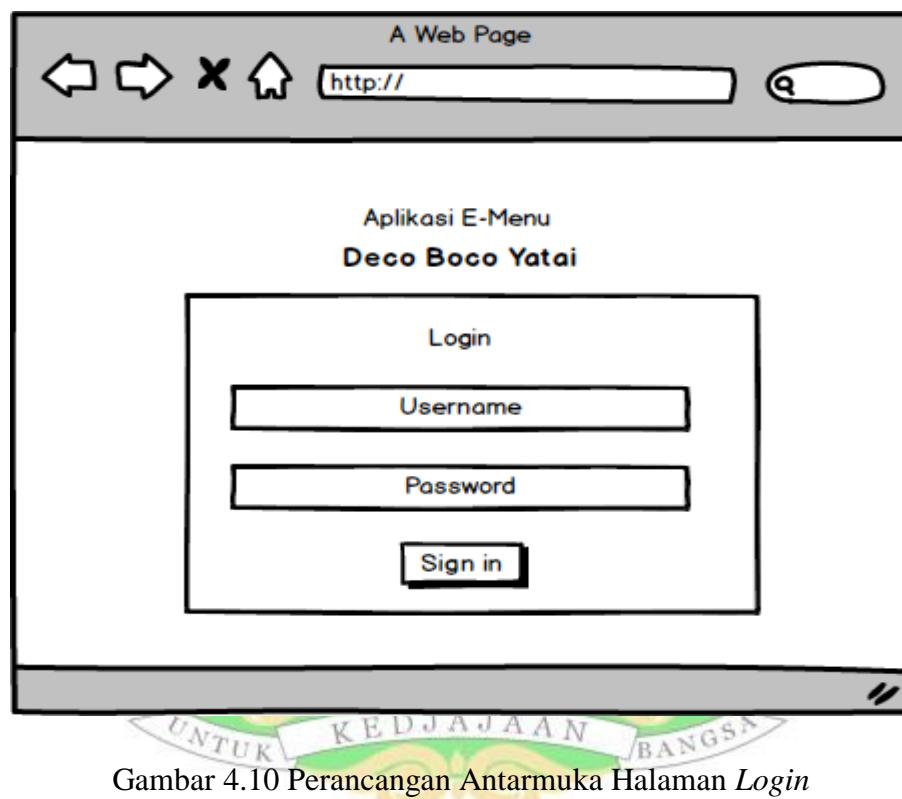
Diketahui bahwa pada aplikasi ini menggunakan API (Application Programming Interface) dalam melakukan request, request akan dilanjutkan ke server yang dalam hal ini adalah database server. Setelah diterima oleh database, dan menemukan data sesuai yang diminta, maka server akan kembali menghubungi API. Selanjutnya API meneruskan informasi dari server ke aplikasi/client.

#### 4.2.5. Perancangan Antarmuka

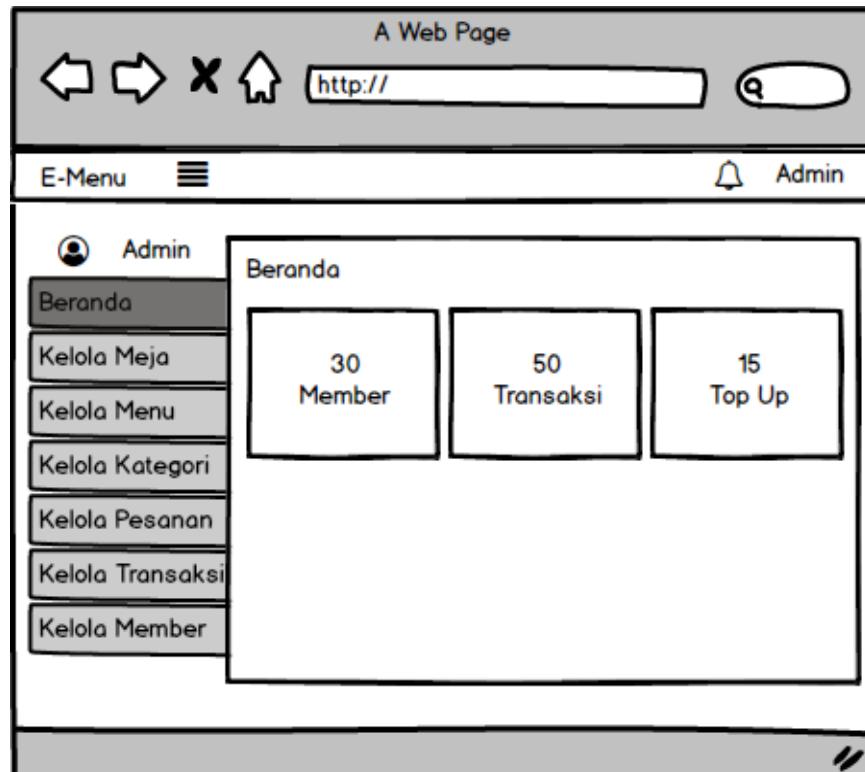
Antarmuka aplikasi *web* yang dijelaskan pada bagian ini adalah antarmuka halaman *login* aplikasi, halaman utama untuk *user* admin, halaman pesanan, dan halaman transaksi *cafe*. Halaman *login* merupakan halaman yang ditampilkan saat pertama kali mengakses aplikasi. Halaman *login* menampilkan form *login* sebelum *user* masuk ke dalam sistem. Setelah *user* berhasil *login*, sistem akan menampilkan halaman utama admin. Untuk gambaran perancangan antarmuka aplikasi *web* lainnya dapat dilihat pada lampiran D.

#### 4.2.5.1. Perancangan Antarmuka Aplikasi Web

Antarmuka aplikasi *web* yang dijelaskan pada bagian ini adalah antarmuka halaman *login* aplikasi, halaman utama untuk admin, dan halaman daftar pesanan. Halaman *login* merupakan halaman yang ditampilkan saat pertama kali mengakses aplikasi. Halaman *login* menampilkan form *login* sebelum *user* masuk ke dalam sistem. Setelah *user* berhasil *login*, sistem akan menampilkan halaman utama. Untuk gambaran perancangan antarmuka aplikasi *web* tersebut dapat dilihat pada Gambar 4.10, 4.11, dan 4.12.



Gambar 4.10 Perancangan Antarmuka Halaman *Login*



Gambar 4.11 Perancangan Antarmuka Halaman Utama Admin

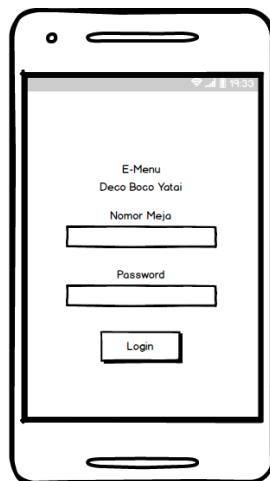
NO	ID	Meja	Menu	Jumlah	Catatan	Aksi
1	001	Meja 1	Ramen Cumi	2		<input type="button" value="Proses"/>

Gambar 4.12 Perancangan Antarmuka Halaman Pesanan

#### 4.2.5.2. Perancangan Antarmuka Aplikasi *Mobile*

Aplikasi *mobile* dalam pembangunan aplikasi *e-menu* ini ditujukan untuk semua pelanggan, baik yang telah mendaftar menjadi *member* maupun yang belum

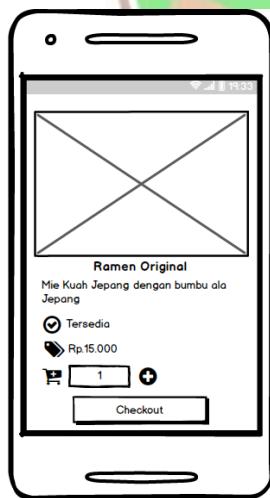
terdaftar sebagai *member*. Dengan aplikasi ini pelanggan dapat melihat menu yang ada, melakukan pemesanan menu, dan untuk *member* yang telah terdaftar dapat melakukan pembayaran melalui *QR Code*. Antarmuka aplikasi *mobile* yang dijelaskan pada bagian ini adalah antarmuka halaman *login*, halaman utama, halaman *detail* menu, dan halaman profil *member*. Perancangan antarmuka halaman-halaman tersebut dapat dilihat pada Gambar 4.13, 4.14, 4.15, dan 4.16.



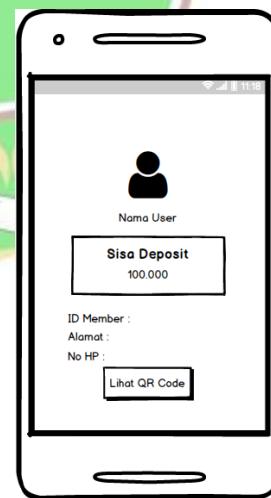
Gambar 4.13 Halaman *Login*



Gambar 4.14 Halaman Utama



Gambar 4.15 Halaman *Detail Menu*



Gambar 4.16 Halaman Profil *Member*

## BAB V

### IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisi tentang implementasi dan pengujian terhadap sistem pemesanan dan pembayaran pada Café Deco Boco Yatai yang telah dibangun.

#### 5.1. Implementasi Sistem

Implementasi aplikasi ini dilakukan dengan menggunakan perangkat keras komputer dengan spesifikasi sebagai berikut :

1. Komputer dengan *processor* Intel(R) Core(TM) i7-4510U CPU 2.59 GHz.
2. *Random Acces Memory* (RAM) 12 GB.
3. *Harddisk* Komputer dengan kapasitas 1 TB.

Untuk implementasi aplikasi *mobile* digunakan perangkat *smartphone* dengan spesifikasi sebagai berikut :

1. *Processor* Quadcore Cortex-A7, 1.3 GHz
2. Memori (RAM) 1GB
3. *Harddisk internal smartphone* 8GB

Spesifikasi perangkat lunak yang digunakan dalam implementasi sistem ini adalah sebagai berikut :

1. Sistem operasi Windows 10 Home
2. Sistem operasi Android Lollipop 5.0
3. *Web browser* yang digunakan adalah google chrome version 88.0

Berikut ini spesifikasi *system requirement* untuk implementasi sistem :

1. Sistem operasi Windows 7, 8, 8.1, 10, Mac OS X 10.5 dan 10.6+, Linux, dan Ubuntu 8+.
2. *Web server* dengan modul PHP 7.0.13 atau setelahnya
3. *Web browser* yang mendukung HTML 5
4. *Android Operating System* versi 4.4 (Kit Kat) atau setelahnya.

Bahasa pemrograman yang digunakan untuk membangun aplikasi web adalah PHP versi 7.4.2, dan *web server apache* XAMPP versi 3.2.4. *Database* yang digunakan sebagai tempat penyimpanan data adalah *database* MySQL.

### 5.1.1. Pengkodean Program

Pengkodean program ini dijelaskan secara terpisah berdasarkan fungsional dan kegunaan masing-masing bagian pada aplikasi. Pada bagian ini akan dibahas kode program koneksi *database*, tambah data menu, dan tambah pesanan. Untuk penjelasan kode program lebih lanjut dapat dilihat pada lampiran C.

#### 5.1.1.1. Kode Program Koneksi *Database*

Koneksi database merupakan bagian yang mengatur hubungan antara program dan database. Kode koneksi database terdapat dalam file db.php. Kode program koneksi *database* aplikasi dapat dilihat pada Gambar 5.1.

```
<?php
```

```
$conn = mysqli_connect('localhost', 'root', '', 'emenu');

if (!$conn) {
    die('Could not connect: ' . mysqli_connect_error());
}

mysqli_select_db($conn, "emenu");
?>
```

Gambar 5.1 Kode Program Koneksi *Database* Aplikasi

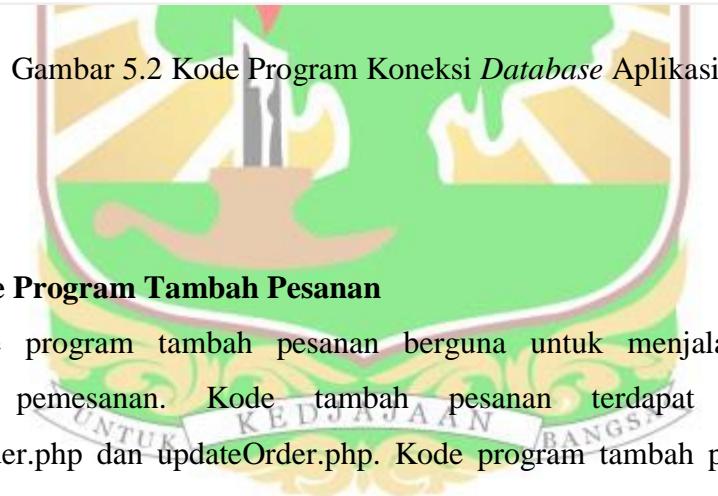
#### 5.1.1.2. Kode Program Tambah Data Menu

Kode program tambah data menu berguna untuk menjalankan fungsi tambah data menu pada café. Kode tambah data menu terdapat dalam file addDish.php. Kode program tambah data menu dapat dilihat pada Gambar 5.2.

```

<?php
ob_start();
include("db.php");
$cusine=htmlspecialchars($_POST['cusineid']);
$dishname=htmlspecialchars($_POST['dishname']);
$desc=htmlspecialchars($_POST['discription']);
$price=htmlspecialchars($_POST['price']);
// $type=htmlspecialchars($_POST['typeid']);
$dishimage = $_FILES['dishimage']['name'];
$dishflag=0;
$checkdish= mysqli_query($conn,"SELECT dishname FROM dish");
if(mysqli_num_rows($checkdish)>0)
{
    while ($row= mysqli_fetch_row($checkdish))
    {
        $dishnm=$row[0];
        if(strcasecmp($dishnm, $dishname)==0)
            $dishflag=1;
    }
}
else {
    $dishflag=0;
}
if($dishflag==0)
{
$query ="INSERT INTO `dish`(cusineid,dishname,dishimage,description,price) VALUES ('$cusine','$dishname','$d
$result = mysqli_query($conn,$query);
}
else {
    header("Location: addDish.php?add_rdata=checkdishname");
    die();
}

```



Gambar 5.2 Kode Program Koneksi Database Aplikasi

### 5.1.1.3. Kode Program Tambah Pesanan

Kode program tambah pesanan berguna untuk menjalankan fungsi melakukan pemesanan. Kode tambah pesanan terdapat dalam file startNewOrder.php dan updateOrder.php. Kode program tambah pesanan dapat dilihat pada Gambar 5.3 dan 5.4.

```

<?php
$tableid=$_POST['tableid'];
$userid=$_POST['userid'];
$response = array();
require '../db.php';

$query1="UPDATE tablemaster set tablestatus='1' WHERE id='$tableid'";
$result1 = mysqli_query($conn,$query1);

$query ="INSERT INTO `ordertable`(`tableid`, `userid`, `status`, `paymentstatus`) VALUES"
. " ('$tableid','$userid','1','unpaid')";
$result = mysqli_query($conn,$query);

$last_insert_id = mysqli_insert_id($conn);

    | $response["orderid"] = $last_insert_id ;

echo json_encode($response);

```

Gambar 5.3 Kode Program Tambah Pesanan

```

<?php
require '../db.php';
$dataraw = json_decode(file_get_contents("php://input"));
$orders = $dataraw->order;

foreach ($orders as $data) :

$orderid = $data->orderid;
$dishid = $data->dishid;
$quantity = $data->quantity;
$userid = $data->userid;
$status = "0";
$pricequery="SELECT price FROM dish where id=".$dishid;
$priceresult= mysqli_query($conn, $pricequery);
$row= mysqli_fetch_row($priceresult);
$getprice=$row[0];
$price=$getprice*$quantity;

$query = "INSERT INTO orderdetails(dishid,quantity,price,$status,orderid)
VALUES('$dishid','$quantity','$price','$status','$orderid')";
$result = mysqli_query($conn, $query);
endforeach;
if ($result) :

$data = array(
    'status' => 200,
    'message' => 'success'
);

```

Gambar 5.4 Kode Program Tambah Pesanan (Lanjutan)

### 5.1.2. Implementasi Antar Muka Aplikasi Web

Pada bagian ini dijelaskan tampilan aplikasi *web* dalam pembangunan sistem pemesanan menu makanan secara *online (e-menu)* menggunakan teknologi *QR Code* pada *Cafe Deco Boco Yatai*. Implementasi antar muka yang akan

dijelaskan yaitu pada bagian halaman *login*, admin, dan koki. Untuk penjelasan implementasi antar muka lain dapat dilihat pada lampiran E.

#### 5.1.2.1. Halaman *Login*

Halaman *login* adalah halaman yang pertama kali ditampilkan ketika *user* membuka aplikasi. Halaman login berfungsi untuk membuat aplikasi hanya dapat diakses oleh *user* yang memiliki hak untuk mengakses sistem. Aplikasi e-menu ini memiliki dua level hak akses *user*, yaitu admin dan koki. Tampilan halaman *login* dapat dilihat pada Gambar 5.5.

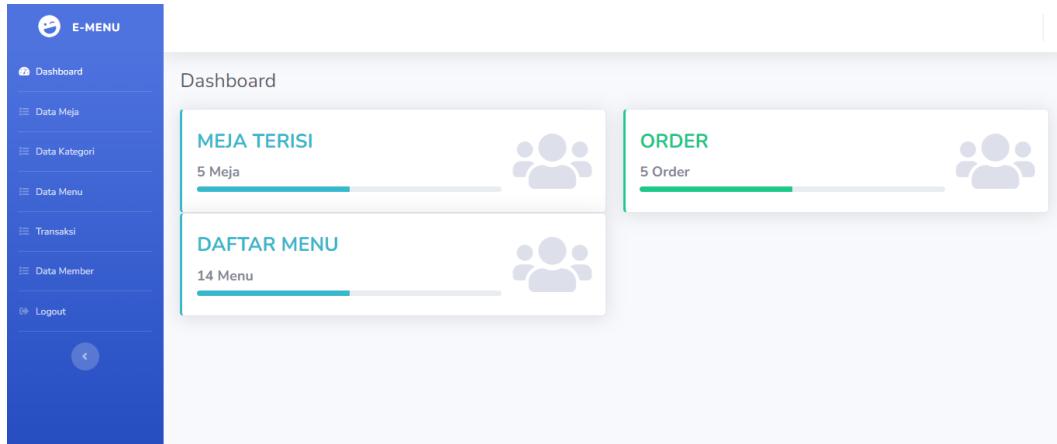


Gambar 5.5 Tampilan Halaman *Login*

Berdasarkan gambar 5.5, *user* diharuskan untuk melakukan *login* terlebih dahulu agar dapat masuk ke dalam sistem. Untuk *login* dibutuhkan *username* dan *password* yang telah terdaftar di *database*. Jika *user* berhasil *login*, sistem akan menampilkan halaman utama aplikasi sesuai role *user* yang *login*.

#### 5.1.2.2. Halaman Admin

Halaman admin dapat diakses oleh *user* dengan level hak akses admin. Admin memiliki wewenang dalam mengelola data *member*, data master menu, manajemen *user* dan melihat laporan transaksi serta pengisian deposit *member*. Pada halaman ini *user* juga bisa melihat informasi tentang *cafe*, seperti jumlah *member* yang telah terdaftar, jumlah transaksi sedang berjalan, jumlah menu makanan dan minuman, serta jumlah pengisian deposit *member*. Tampilan halaman admin dapat dilihat pada gambar 5.6.



Gambar 5.6 Tampilan Halaman Utama Admin

Berdasarkan Gambar 5.6 dapat dijelaskan bahwa halaman admin memiliki delapan menu utama dan satu menu *sign out* yang diuraikan sebagai berikut:

1. Menu Beranda

Menu beranda adalah halaman utama yang menampilkan informasi dan pemberitahuan tentang *cafe*. Informasi tersebut berupa jumlah transaksi yang sedang berlangsung, jumlah transaksi yang telah selesai, jumlah *member*, dan jumlah menu yang ada pada *cafe*.

2. Menu Kelola Meja

Menu kelola meja menampilkan halaman pengelolaan data meja *cafe*. Pada menu ini *user* dapat melakukan beberapa fungsional seperti penambahan meja baru, memperbarui data meja, dan menghapus data meja.

3. Menu Kelola Menu (Makanan dan Minuman)

Menu kelola menu makanan dan minuman ini menampilkan halaman pengelolaan data menu makanan dan minuman yang ada tersedia di *cafe*. Pada menu ini *user* dapat melakukan beberapa fungsional seperti penambahan menu baru, perbarui data menu dan hapus data menu.

4. Menu Kelola Kategori

Menu kelola kategori menampilkan halaman pengelolaan data kategori atau jenis dari menu makanan ataupun minuman yang ada pada *cafe*. Pada menu ini *user* dapat melakukan beberapa fungsional seperti penambahan kategori, perbarui data kategori dan hapus data kategori.

## 5. Menu Transaksi

Menu transaksi memiliki dua submenu, yaitu order berjalan dan riwayat transaksi. Submenu order berjalan berfungsi untuk pengelolaan order pesanan yang masih berlangsung, sedangkan submenu riwayat transaksi berfungsi untuk melihat transaksi-transaksi yang sudah selesai/berakhir.

## 6. Menu *Member*

Menu *member* terdiri dari empat submenu, yaitu data *member*, tambah *member*, *top up*, dan riwayat *top up*. Submenu data *member* menampilkan halaman pengelolaan data *member*. Pengelolaan data *member* yang dapat dilakukan meliputi perbarui data *member* dan hapus data *member*. Submenu tambah *member user* dapat menambahkan data *member* baru. Pada submenu *top up user* dapat melakukan pengisian deposit *member*. Sedangkan pada submenu riwayat *top up*, *user* dapat melihat daftar pengisian deposit *member* yang telah dilakukan.

## 7. Menu Laporan

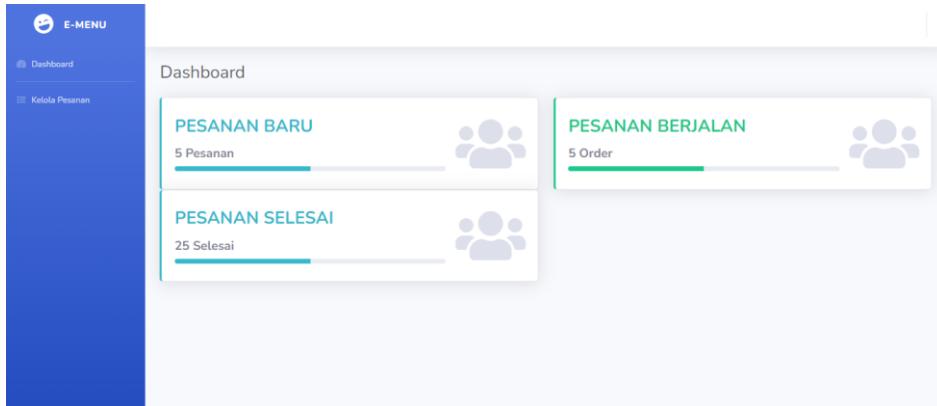
Menu laporan berfungsi untuk menampilkan data laporan terkait transaksi dan dapat diconvert kedalam Microsoft Excel serta dapat melakukan pencetakan.

## 8. Menu *Logout*

Menu *logout* berfungsi untuk menutup halaman utama dan menampilkan halaman *login*. Menu ini dapat diakses *user* pada bagian *navbar* di aplikasi.

### 5.1.2.1. Halaman Koki

Halaman koki dapat diakses oleh *user* dengan level hak akses koki. Hak akses untuk masuk ke dalam sistem dan bisa menggunakan fungsional bisa juga dilakukan oleh pegawai yang bertugas di dapur. Tampilan halaman koki dapat dilihat pada Gambar 5.7.



Gambar 5.7 Tampilan Halaman Koki

Berdasarkan Gambar 5.7 dapat dijelaskan bahwa halaman koki memiliki dua menu utama, tiga submenu, dan memiliki satu menu *sign out* yang diuraikan sebagai berikut :

1. Menu Beranda

Menu beranda adalah halaman utama yang menampilkan informasi mengenai pesanan pada *cafe*. Informasi tersebut berupa jumlah pesanan baru yang masuk, jumlah pesanan berjalan atau pesanan yang sedang diproses, dan jumlah pesanan selesai.

2. Menu Kelola Pesanan

Menu kelola pesanan memiliki tiga submenu, yaitu pesanan baru, pesanan berjalan, dan pesanan selesai. Submenu pesanan baru menampilkan data pesanan yang baru masuk. Submenu pesanan berjalan menampilkan data pesanan baru yang telah diproses dan sedang dimasak. Sedangkan submenu pesanan selesai berisi data pesanan yang telah selesai diproses dan telah disajikan kepada pelanggan

3. Menu *Logout*

Menu *logout* berfungsi untuk menutup halaman utama dan menampilkan halaman *login*. Menu ini dapat diakses *user* pada bagian *navbar* di aplikasi.

### 5.1.3. Implementasi Antar Muka Aplikasi *Mobile*

Aplikasi *mobile* dapat digunakan oleh *user* yang memiliki level hak akses *member* dan pelanggan biasa yang belum menjadi *member*. Pada aplikasi *mobile*

ini terdapat enam fungsional, yaitu melihat menu, melakukan pemesanan, melihat *QR Code member*, melihat sisa deposit, melihat status pesanan, dan melihat riwayat transaksi.

Halaman awal yang ditampilkan saat mengakses aplikasi *mobile* ini adalah halaman *login user*. Pada halaman ini *login user* terbagi dua, yaitu *login* sebagai pengunjung dan *login* sebagai *member* yang sudah terdaftar sebelumnya. Pada *login* sebagai pengunjung *user* akan melakukan scan kode QR yang ada di meja. Sedangkan untuk *login* sebagai *member*, *user* harus menginputkan *username* dan password untuk masuk ke dalam aplikasi. Pada halaman utama *user* akan memilih menu-menu yang tersedia. Tampilan halaman awal dapat dilihat pada Gambar 5.8 dan halaman *login* pada Gambar 5.9.

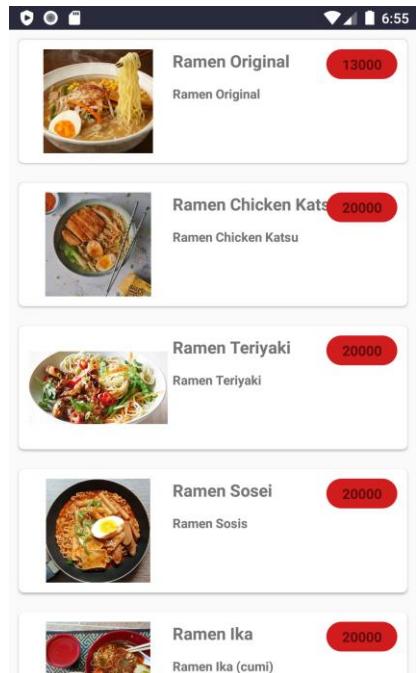


Gambar 5.8 Tampilan Halaman Awal

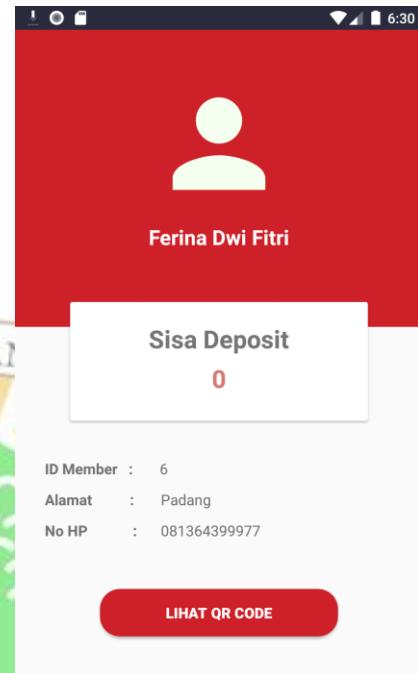
Gambar 5.9 Tampilan Halaman Utama

Berdasarkan Gambar 5.8 dan Gambar 5.9 *user* yang telah *login* dapat memilih menu pada halaman utama. Salah satu menu yang dapat dijalankan adalah melihat menu makanan yang tersedia. Pada menu ini terdapat informasi mengenai nama dan harga menu makanan tersedia yang dapat dipesan oleh *user*. Selain itu pada halaman utama juga terdapat juga menu profil. Menu profil dapat menampilkan informasi tentang *member*, termasuk sisa deposit yang tersisa dari *member*. Tampilan halaman menu makanan dapat dilihat pada Gambar 5.10 dan

halaman profil pada Gambar 5.11. Untuk tampilan *mobile* yang lainnya dapat dilihat pada lampiran E.



Gambar 5.10 Halaman Menu



Gambar 5.11 Halaman Profil

## 5.2. Pengujian Sistem

Pada tahap ini dilakukan pengujian terhadap aplikasi *e-menu* yang telah dibangun. Pengujian dilakukan dengan menggunakan *black box testing* untuk memeriksa apakah aplikasi yang dibangun sesuai dengan kebutuhan fungsional yang telah dirancang.

### 5.2.1. Fokus Pengujian

Fokus pengujian aplikasi ini menggunakan data uji berdasarkan data yang telah didapat melalui aplikasi *web* maupun *mobile*. Fokus pengujian dapat dilihat pada tabel 5.1

Tabel 5.1 Fokus Pengujian

No.	Item Uji	Jenis Aplikasi	Detail Pengujian
1	Mengelola data <i>user</i>	Web	Lihat, tambah, edit, hapus
2	Mengelola data master <i>cafe</i>	Web	Lihat, tambah, hapus, edit
3	Registrasi <i>member</i> baru	Web	Tambah
4	Mengelola data <i>member</i>	Web	Lihat, edit, hapus
5	Mengelola data meja	Web	Lihat, tambah, hapus, edit
6	Mengelola data menu	Web	Lihat, tambah, hapus, edit
7	Mengelola data kategori menu	Web	Lihat, tambah, hapus, edit
8	Melihat daftar pesanan	Web	Lihat
9	Melihat riwayat pesanan	Web	Lihat
10	Mengelola transaksi pembayaran	Web	Scanning, tambah
11	Melihat riwayat transaksi	Web, mobile	Lihat
12	Pengisian deposit <i>member</i>	Web	Scanning, tambah
13	Melihat dan mencetak laporan transaksi	Web	Lihat, cetak, unduh
14	Mengelola pesanan	Web	Lihat, edit

Tabel 5.1 Fokus Pengujian

15	Melihat menu	<i>Mobile</i>	Lihat
16	Melakukan pemesanan	<i>Mobile</i>	Tambah
17	Melihat status pesanan	<i>Mobile</i>	Lihat
18	Melihat profil	<i>Mobile</i>	Lihat
19	Melihat sisa saldo	<i>Mobile</i>	Lihat
20	Melihat <i>QR Code</i>	<i>Mobile</i>	Lihat

### 5.2.2. Kasus Hasil Pengujian

Pada bagian ini dijelaskan tiga kasus hasil pengujian, yaitu pengujian registrasi *member* baru, transaksi pesanan, transaksi pembayaran pesanan, dan lihat laporan transaksi.

#### 5.2.2.1. Pengujian Melakukan Pemesanan Melalui Aplikasi *Mobile*

Pada pengujian ini dilakukan *input* data pesanan oleh pelanggan pada aplikasi *mobile*. Hasil pengujian pemesanan melalui aplikasi *mobile* dapat dilihat pada Tabel 5.2.

Tabel 5.2 Pengujian Melakukan Pemesanan (Benar)

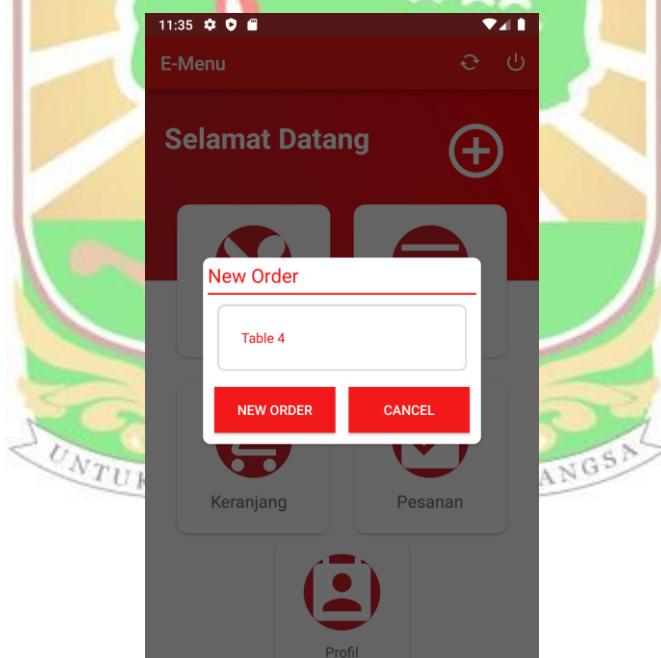
Kasus dan Hasil Uji (Benar)	
Data masukan	Membuka halaman pesan makanan dan menjalankan fungsi buat pesanan
Yang diharapkan	Sistem menampilkan notifikasi pesanan berhasil
Pengamatan	Notifikasi berhasil ditampilkan
Hasil	Sesuai

Pengujian dilakukan dengan menekan tombol tambah pesanan yang ada di halaman awal. Kemudian pelanggan akan diarahkan ke halaman pilih meja dan kemudian muncul halaman untuk memilih menu. Setelah memilih menu pelanggan menekan tombol pesan. Sistem memproses data dan menampilkan notifikasi

pesanan berhasil dibuat. Tampilan hasil pengujian melakukan pemesanan dapat dilihat pada Gambar 5.12, 5.13, dan 5.14.



Gambar 5.12 Halaman Awal Sebelum Tambah Pesan



Gambar 5.13 Halaman Pilih Meja



Gambar 5.14 Halaman Pilih Menu

SrNo	OrderId	UserName	TableName	Dish Name	Quantity	Status
1	114	ferina	Table 4	Takoyaki	1	New
2	114	ferina	Table 4	Ramen Original	2	New

Gambar 5.15 Halaman Pesanan Baru Telah Berhasil Ditambahkan

Berdasarkan gambar 5.13, 5.14, dan 5.15 dapat dilihat pesanan baru berhasil dilakukan. Data tersimpan ke database dan sistem menampilkan halaman pesanan baru di halaman web koki.

### 5.2.2.2. Pengujian Transaksi Pembayaran Pesanan

Pada pengujian ini dilakukan *scanning QR Code member* pada aplikasi *web*. Hasil pengujian transaksi pembayaran *cafe* dapat dilihat pada tabel 5.3 dan tabel 5.4.

Tabel 5.3 Pengujian Transaksi Pembayaran *Cafe* (Benar)

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	<i>QR Code member</i>
Yang diharapkan	Transaksi sukses dan sistem menampilkan notifikasi transaksi berhasil dan menampilkan halaman cetak nota transaksi <i>café</i>
Pengamatan	Transaksi pembayaran berhasil dan muncul notifikasi transaksi berhasil
Hasil	Sesuai

Pengujian dilakukan dengan membuka submenu “Order berjalan” pada halaman utama aplikasi, dan menekan tombol “Pembayaran” yang ada pada halaman “Daftar Transaksi Berlangsung”. Selanjutnya melakukan pemindaian terhadap *QR Code member*, dan sistem menampilkan notifikasi berhasil jika transaksi pembayaran sukses dilakukan, dan sebaliknya, sistem akan menampilkan notifikasi kesalahan jika transaksi gagal. Tampilan hasil pengujian transaksi pembayaran *cafe* (benar) dapat dilihat pada gambar 5.16 dan 5.17.



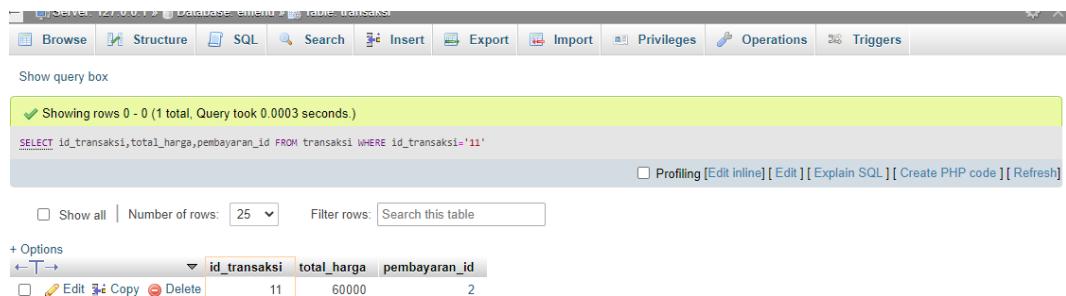
Gambar 5.16 *Input* Pengujian Transaksi Pembayaran (Benar)

Transaksi index								Home > Dashboard
Pembayaran Berhasil								
Kelola Transaksi								
Show <input type="button" value="10"/> entries								Search: <input type="text"/>
ID Transaksi	Meja	Tanggal	Total Harga	Status Bayar	ID Member	Pembayaran		
11	1	2022-01-02	60000	Selesai	Putri Maharani	Non Tunai		
Showing 1 to 1 of 1 entries								Previous <input type="button" value="1"/> Next

Gambar 5.17 *Output* Pengujian Transaksi Pembayaran (Benar)

Berdasarkan gambar di atas transaksi pembayaran berhasil dilakukan. Sistem menampilkan notifikasi pembayaran berhasil dan halaman daftar transaksi berlangsung.

Pembuktian pengujian dilakukan dengan mengakses langsung ke *database* dan melakukan *query select* dengan *field* pembayaran pada tabel transaksi. Pembuktian dengan *query database* dapat dilihat pada gambar 5.18.



The screenshot shows a MySQL Workbench interface with the following details:

- Toolbar: Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Triggers.
- Query Editor: Shows a green success message: "Showing rows 0 - 0 (1 total, Query took 0.0003 seconds.)" and the SQL query: "SELECT id\_transaksi, total\_harga, pembayaran\_id FROM transaksi WHERE id\_transaksi='11'".
- Result Set: A table with three columns: id\_transaksi, total\_harga, and pembayaran\_id. One row is selected and highlighted: id\_transaksi 11, total\_harga 60000, and pembayaran\_id 2.
- Buttons: Show all, Number of rows: 25, Filter rows: Search this table, Options, Edit, Copy, Delete.

Gambar 5.18 Pengecekan Pengujian Transaksi Pembayaran (Benar)

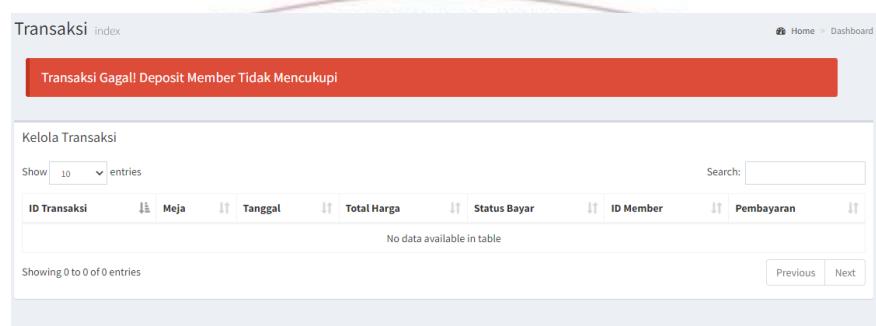
Tabel 5.4 Pengujian Transaksi Pembayaran (Alternatif)

Kasus dan Hasil Uji (Benar)	
Data masukan	Saldo tidak mencukupi dan data lengkap
Yang diharapkan	Transaksi gagal dan sistem menampilkan notifikasi deposit tidak mencukupi
Pengamatan	Transaksi pembayaran gagal dan sistem menampilkan notifikasi deposit tidak cukup
Hasil	Sesuai

Pengujian aplikasi dilakukan dengan melakukan *scanning QR Code member* saat deposit *member* tidak mencukupi untuk melakukan transaksi dengan total harga yang sudah ditentukan. Kemudian sistem menampilkan notifikasi kesalahan. Hasil pengujian transaksi pembayaran (alternatif) dapat dilihat pada gambar 5.19, dan 5.20.



Gambar 5.19 *Input* Pengujian Transaksi Pembayaran (Alternatif)



Gambar 5.20 *Output* Pengujian Transaksi Pembayaran (Alternatif)

Untuk melakukan pembuktian bahwa transaksi pembayaran gagal dilakukan, maka pengujian dilanjutkan dengan mengakses langsung ke *database* dan melakukan *query select* dengan *field* pembayaran pada tabel transaksi. Jika jumlah pembayaran =0 atau tidak sama dengan jumlah total pesanan, maka transaksi pembayaran gagal. Pembuktian dengan *query database* dapat dilihat pada gambar 5.21.



Gambar 5.21 Pengecekan Pengujian Transaksi Pembayaran (Alternatif)

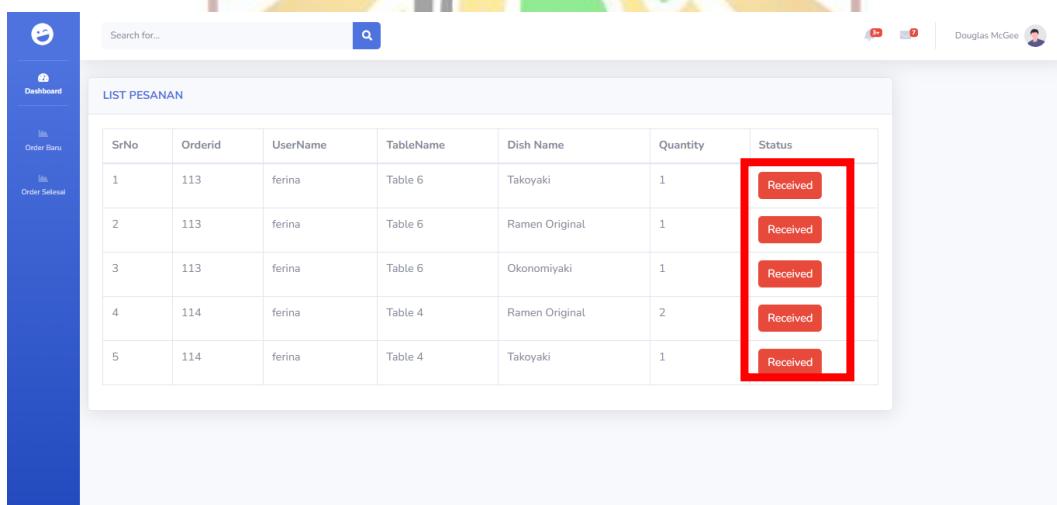
### 5.2.2.3. Pengujian Mengubah Status Pesanan

Pada pengujian ini dilakukan konfirmasi pesanan pada halaman koki. Hasil pengujian mengubah status pesanan dapat dilihat pada tabel 5.5.

Tabel 5.5 Pengujian Lihat Laporan Transaksi *Cafe* (Benar)

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Membuka halaman daftar pesanan
Yang diharapkan	Status pesanan berubah sesuai proses yang telah dikerjakan
Pengamatan	Status pesanan berubah
Hasil	Sesuai

Pengujian aplikasi dilakukan dengan menekan tombol “New” untuk pesanan yang baru masuk. Selanjutnya koki menekan tombol “Received” jika pesanan telah selesai dibuat dan disajikan ke pelanggan. Setelah selesai status akan menjadi “served” dan data pesanan berpindah ke halaman pesanan selesai. Hasil pengujian ubah status pesanan dapat dilihat pada Gambar 5.22 dan 5.23.



SrNo	Orderid	UserName	TableName	Dish Name	Quantity	Status
1	113	ferina	Table 6	Takoyaki	1	Received
2	113	ferina	Table 6	Ramen Original	1	Received
3	113	ferina	Table 6	Okonomiyaki	1	Received
4	114	ferina	Table 4	Ramen Original	2	Received
5	114	ferina	Table 4	Takoyaki	1	Received

Gambar 5.22 Ubah Status Pesanan Baru

SrNo	Orderid	UserName	TableName	Menu Name	Quantity	Status
1	103	Rahma	Table 2	Ramen Original	2	<span style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; display: inline-block;">Served</span>
2	113	ferina	Table 6	Takoyaki	1	<span style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; display: inline-block;">Served</span>
3	107	Rahma	Table 3	Ramen Original	1	<span style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; display: inline-block;">Served</span>
4	113	ferina	Table 6	Ramen Original	1	<span style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; display: inline-block;">Served</span>
5	107	Rahma	Table 3	Takoyaki	1	<span style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; display: inline-block;">Served</span>
6	113	ferina	Table 6	Okonomiyaki	1	<span style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; display: inline-block;">Served</span>
7	108	Rahma	Table 5	Takoyaki	2	<span style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; display: inline-block;">Served</span>
8	114	ferina	Table 4	Ramen Original	2	<span style="background-color: red; color: white; border-radius: 5px; padding: 2px 10px; display: inline-block;">Served</span>

Gambar 5.23 Status Pesanan Selesai

### 5.3. Kesimpulan Hasil Pengujian

Pada pengujian sistem yang dilakukan terhadap masing-masing kebutuhan fungsional tidak ditemukan kegagalan. Oleh karena itu, dapat disimpulkan bahwa pembangunan aplikasi pemesanan menu makanan secara *online* (*e-menu*) menggunakan teknologi *qr-code* berbasis *web* dan *mobile* pada *Cafe Deco Boco Yatai* telah berjalan sesuai dengan fungsional yang dirancang. Hasil pengujian secara lengkap dijelaskan pada lampiran I. Kesimpulan hasil pengujian sistem dapat dilihat pada Tabel 5.6.

Tabel 5.6 Tabel Hasil Pengujian

No.	Fungsional	Skenario Pengujian		Hasil yang Diharapkan		Hasil
		Benar	Alternatif	Benar	Alternatif	
1	Mengelola data <i>user</i>	Membuka menu manajemen <i>user</i> , menjalankan fungsi tambah, edit, hapus, dan cari.	Melakukan <i>input</i> data salah pada setiap fungsi.	Sistem menampilkan, menyimpan, dan menghapus data <i>user</i> serta menampilkan notifikasi kesalahan.	Sistem menampilkan, menyimpan, dan menghapus data <i>user</i> serta menampilkan notifikasi kesalahan.	Sesuai

Tabel 5.6 Tabel Hasil Pengujian

		cari, dan hapus data <i>user</i> , <i>input</i> data benar		notifikasi berhasil untuk setiap fungsi		
2	Mengelola data master <i>cafe</i>	Membuka menu data referensi, menjalankan fungsi tambah, edit, cari, dan hapus pada setiap sub menu data referensi, <i>input</i> data benar	Melakukan <i>input</i> data salah pada setiap fungsi	Sistem menampilkan, menyimpan, <i>update</i> dan menghapus data referensi serta menampilkan notifikasi berhasil untuk setiap fungsi.	Sistem menampilkan notifikasi kesalahan	Sesuai
3	Mengelola data menu	Membuka menu, menjalankan fungsi tambah, edit, cari, dan hapus data menu, <i>input</i> data benar	Melakukan <i>input</i> data salah pada setiap fungsi	Sistem menampilkan, menyimpan, <i>update</i> dan menghapus data menu serta menampilkan notifikasi	Sistem menampilkan notifikasi kesalahan	Sesuai

Tabel 5.6 Tabel Hasil Pengujian

				berhasil untuk setiap fungsi		
4	Melihat dan mencetak laporan transaksi	Membuka menu laporan dan memilih submenu transaksi, <i>input</i> periode laporan transaksi, jalankan Fungsi lihat laporan dan fungsi cetak laporan, <i>input</i> data benar	-	Sistem menampilkan informasi rekapan data transaksi sesuai dengan periode tanggal yang dipilih	-	Sesuai
5	Melihat dan mencetak laporan pengisian deposit	Membuka menu laporan dan memilih submenu pengisian deposit, <i>input</i> periode laporan pengisian deposit, jalankan fungsi lihat laporan dan	-	Sistem menampilkan informasi rekapan data pengisian deposit member sesuai dengan periode tanggal yang dipilih	-	Sesuai

Tabel 5.6 Tabel Hasil Pengujian

		fungsi cetak laporan, <i>input</i> data benar				
6	Registrasi <i>member</i> baru	Membuka menu <i>member</i> , memilih submenu daftar member dan menekan tombol registrasi <i>member</i> baru. Mengisi form pendaftaran dan menjalankan fungsi	Melakukan <i>input</i> data salah pada fungsi tambah	Sistem menampilkan dan menambah data <i>member</i> , serta menampilkan notifikasi berhasil	Sistem menampilkan notifikasi kesalahan	Sesuai
7	Mengelola data <i>member</i>	Membuka menu <i>member</i> , menjalankan fungsi edit, cari, dan hapus data <i>member</i> , <i>input</i> data benar	Melakukan <i>input</i> data salah pada setiap fungsi	Sistem menampilkan, menyimpan, mengupdate dan menghapus data <i>member</i> serta menampilkan notifikasi	Sistem menampilkan notifikasi kesalahan	Sesuai

Tabel 5.6 Tabel Hasil Pengujian

				berhasil untuk setiap fungsi		
8	Mengelola data transaksi	Membuka menu kelola transaksi, menjalankan fungsi tambah, <i>update</i> dan cari data transaksi, <i>input</i> data benar	Melakukan <i>input</i> data salah pada setiap fungsi	Sistem menampilkan, menyimpan dan mengupdate data transaksi serta menampilkan notifikasi kesalahan	Sistem menampilkan notifikasi kesalahan	Sesuai
9	Mengelola transaksi pembayaran cafe	Membuka menu “Kelola Transaksi”, memilih submenu “Order Berjalan”, menekan tombol “Pembayaran” dan melakukan <i>scanning QR Code member</i>	Melakukan <i>scanning</i> bukan data <i>QR Code</i> pelanggan atau melakukan pembayaran saat deposit tidak mencukupi untuk melakukan transaksi	Menampilkan halaman cetak nota transaksi dan menampilkan notifikasi berhasil	Menampilkan notifikasi kesalahan	Sesuai

Tabel 5.6 Tabel Hasil Pengujian

10	Pengisian deposit <i>member</i>	Membuka “Pengisian Deposit”, dan melakukan scanning <i>QR Code member</i> , kemudian mengisi nominal <i>top up</i> dan menekan tombol	Melakukan scanning bukan data <i>QR Code pelanggan</i> atau mengisi nominal <i>top up</i> dan menekan tombol	Menampilkan halaman daftar deposit <i>member</i> serta menampilkan notifikasi kesalahan	Menampilkan notifikasi kesalahan	Sesuai
11	Melihat riwayat transaksi	Membuka menu transaksi, menjalankan fungsi lihat riwayat transaksi	-	Sistem menampilkan data riwayat transaksi <i>member</i>	-	Sesuai
12	Melihat sisa saldo	Membuka menu profil, menjalankan fungsi lihat deposit	-	Sistem menampilkan sisa deposit <i>member</i>	-	Sesuai
13	Pendaftaran <i>member</i>	Memilih menu registrasi member baru	Menekan tombol kirim saat form	Sistem menampilkan notifikasi berhasil dan	Sistem menampilkan notifikasi berhasil	Sesuai

Tabel 5.6 Tabel Hasil Pengujian

		pada halaman login, input data dan tekan tombol kiri	masih kosong	menampilkan kembali halaman login	notifikasi kesalahan	
14	Mengelola data meja	Membuka daftar meja, menjalankan fungsi tambah, edit, cari, dan hapus data meja, input data benar	Melakukan input data salah pada setiap fungsi	Sistem menampilkan, menyimpan, mengupdate dan menghapus data meja serta menampilkan notifikasi berhasil untuk setiap fungsi	Sistem menampilkan notifikasi kesalahan	Sesuai
15	Mengelola data kategori menu	Membuka daftar kategori menu, menjalankan fungsi tambah, edit, cari, dan hapus data meja, input data benar	Melakukan input data salah pada setiap fungsi	Sistem menampilkan, menyimpan, mengupdate dan menghapus data kategori menu serta menampilkan notifikasi	Sistem menampilkan notifikasi kesalahan	Sesuai

Tabel 5.6 Tabel Hasil Pengujian

				berhasil untuk setiap fungsi		
16	Melihat daftar pesanan	Membuka menu daftar pesanan, menjalankan fungsi lihat daftar pesanan	-	Sistem menampilkan data daftar pesanan	-	Sesuai
17	Melihat riwayat pesanan	Membuka daftar pesanan, menjalankan fungsi lihat riwayat pesanan	-	Sistem menampilkan data riwayat pesanan	-	Sesuai
18	Mengelola pesanan	Membuka daftar pesanan, menjalankan fungsi lihat dan edit	-	Sistem menampilkan dan mengupdate data pesanan serta menampilkan notifikasi berhasil untuk setiap fungsi	-	Sesuai
19	Melakukan pemesanan	Membuka aplikasi	Melakukan <i>input</i> data	Sistem menampilkan	Sistem menampilkan	Sesuai

Tabel 5.6 Tabel Hasil Pengujian

		<p><i>mobile</i>, memilih meja, memilih menu, dan melakukan pemesanan</p>	<p>salah pada fungsi tambah</p>	<p>dan menambah data pesanan, serta menampilkan notifikasi berhasil</p>	<p>kan notifikasi kesalahan</p>	
17	Melihat profil	<p>Membuka halaman profil, menjalankan fungsi lihat profil</p>		<p>Sistem menampilkan data profil</p>	-	Sesuai



## BAB VI

### PENUTUP

Bab ini berisi tentang kesimpulan dan saran pada tugas akhir ini.

#### 6.1. Kesimpulan

Pemesanan Menu Makanan Secara *Online (E-Menu)* Menggunakan Teknologi *Qr-Code* Berbasis *Web* Dan *Mobile* Pada *Cafe Deco Boco Yatai*, telah berhasil dibangun. Dari setiap tahap yang dilakukan yaitu tahap analisis, *design*, *coding*, dan implementasi dapat diperoleh kesimpulan sebagai berikut.

1. Pembangunan sistem informasi pemesanan menu makanan secara *online (e-menu)* yang terdiri dari aplikasi *web* dan *mobile* telah selesai dibangun.
2. Aplikasi pemesanan menu makanan secara online (*e-menu*) membantu pengguna dalam proses pemesanan makanan, karena keseluruhan proses dari manual menjadi digital sehingga dapat dilakukan pada satu sistem.
3. Sistem informasi yang dibangun ini dapat membantu pihak Café Deco Boco Yatai dalam melakukan pengelolaan pesanan, menu dan pembayaran.

#### 6.2. Saran

Dalam pengembangan aplikasi diharapkan dapat dijalankan di berbagai platform (*multiplatform*) sehingga menjangkau semua sistem operasi yang digunakan pelanggan dan terus diperbarui dengan menambah fungsional yang disesuaikan dengan kebutuhan pelanggan. Dalam sistem pembayaran non tunai diharapkan dapat menggunakan verifikasi sebelum pembayaran dari pihak pelanggan supaya transaksi lebih aman dan terjaga.

## DAFTAR PUSTAKA

- Akbar, R., & Kamil, H. (2017). DEVELOPMENT OF SALES APPLICATION OF PREPAID ELECTRICITY VOUCHER BASED ON ANDROID PLATFORM USING QUICK RESPONSE CODE (QR CODE). *JURNAL INFORMATIKA*, Vol. 11, No. 2, 9.
- Android Studio. (2018). *Meet Android Studio / Android Developers*. Dipetik December 22, 2018, dari <https://developer.android.com/studio/intro/>
- Dharwiyanti, S., & Wahono, R. S. (2003). *Pengantar Unified Modeling Language (UML)*. IlmuKomputer.Com.
- Haryana, K. S. (2008). PENGEMBANGAN PERANGKAT LUNAK DENGAN MENGGUNAKAN PHP. *Jurnal Computech & Bisnis*, Vol. 2, No. 1, 14.
- Haryanti, S., & Irianto, T. (2011). Rancang Bangun Sistem Informasi E-Commerce Untuk Usaha Fashion Studi Kasus Omah Mode Kudus. *Journal Speed – Sentra Penelitian Engineering dan Edukasi*, Volume 3 No 1 - 2011, 10-11.
- Haviluddin. (2011). Memahami Penggunaan UML (Unified Modelling Language). *Jurnal Informatika Mulawarman*, Vol 6 No.1, 1-2.
- Hendini, A. (2016). PEMODELAN UML SISTEM INFORMASI MONITORING PENJUALAN DAN STOK BARANG (STUDI KASUS: DISTRO ZHEZHA PONTIANAK). *JURNAL KHATULISTIWA INFORMATIKA*, VOL. IV, NO. 2, 108-109.
- Hendri. (2016). PROTOTIPE APLIKASI PEMESANAN MAKANAN (ELECTRONIC MENU) PADA RESTORAN BERBASIS ANDROID DAN WEB. *Jurnal Ilmiah Media Processor*, Vol.11, No.1, 623.
- Listfield, R., & Montes-Negret, F. (1994). Modernizing Payment System in Emerging Economies. *World Bank Policy Research Working Paper*, 1336.
- Luthfi, F. (2017). Penggunaan Framework Laravel Dalam Rancang Bangun Modul Back-End Artikel Website Bisnisbisnis.ID. *JISKa*, Vol. 2, No. 1, 34.
- Mulyani, S. (2016). *Metode Analisis dan Perancangan Sistem*. Bandung: Abdi Sistematika.
- Nugraha, M. P., & Munir, R. (2011). Pengembangan Aplikasi QR Code Generator dan QR Code Reader dari Data Berbentuk Image. *Konferensi Nasional Informatika*, ISSN: 2087 - 3328, 148-149.
- Ollie. (2008). *Panduan Praktis Mendesain Template Joomla!* Jakarta Selatan: mediakita.
- Pramihapsari, M., & Kaldera, M. P. (2012). Perancangan Labelling Pada Dokumen Menggunakan QR Code. *Jurnal Teknik Komputer*.

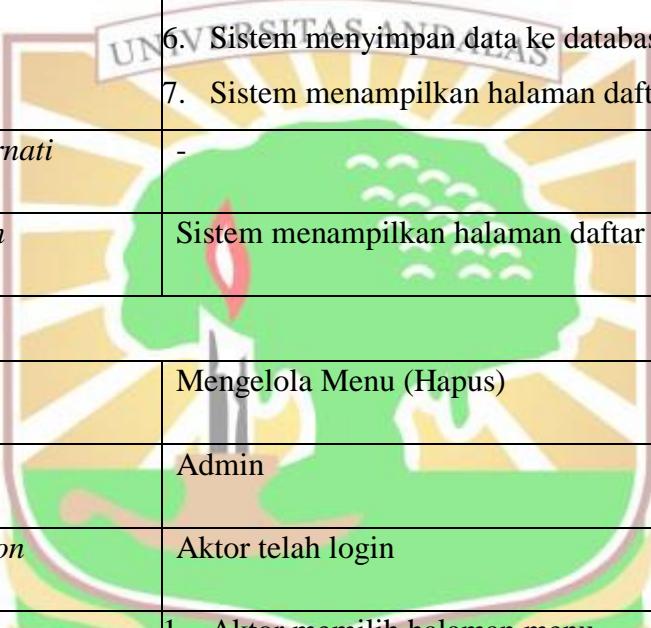
- Rahayu dkk, Y. D. (2006). *Pembuatan Aplikasi Pembacaan Quick Response Code Menggunakan Perangkat Mobile Berbasis J2ME Untuk Identifikasi Suatu Barang*. Surabaya: Politeknik Elektronika Negeri Surabaya Institut Teknologi Sepuluh Nopember.
- Rajesh, M., Satya, P. G., & Rao, P. P. (2015). E-Restaurant: Online Restaurant Management System for Android. *International Jurnal & Magazine of Engineering, Technology, Management and Research*, 574-579.
- Ropianto, M. (2016). Pemahaman Penggunaan Unified Modelling Language. *JT-IBSI, Volume 01, Nomor 01*, 47.
- Sari, S. K., & Asniar. (2015). Analisis Dan Pemodelan Proses Bisnis Prosedur Pelaksanaan Proyek Akhir Sebagai Alat Bantu Identifikasi Kebutuhan Sistem. *Jurnal Infotel*, Vol. 7 No. 2, 144-146.
- Sommerville, I. (2011). *SOFTWARE ENGINEERING* (Ninth Edition ed.). Boston: Pearson Education.
- Stone, A. G. (2006). The Case Study Approach to Scenario Planning. *Journal of Practical Consulting*, Vol. 1 Iss. 1, 2006.
- Susanto, R., & Andriana, A. D. (2016). PERBANDINGAN MODEL WATERFALL DAN PROTOTYPING UNTUK PENGEMBANGAN SISTEM INFORMASI. *Majalah Ilmiah UNIKOM*, Vol.14 No. 1, 43.
- Timotius Witono & Raphael Susanto. (2012). Aplikasi Pemesanan Tiket Bioskop Berbasis Mobile. *Jurnal Sistem Informasi*, Vol.7, No.2, 147.
- Tompoh, J. F., Sentinuwo, S. R., & Sinsuw, A. A. (2016). Rancang Bangun Aplikasi Pemesanan Menu Makanan Restoran Berbasis Android. *E-journal Teknik Informatika*, Vol 9, No 1, 1.
- Usman, R. (2017). Karakteristik Uang Elektronik Dalam Sistem Pembayaran. *Yuridika*, Volume 32 No. 1, 152.
- Wardani, K. T. (2012). Pembangunan Sistem Informasi Akademik Lembaga Bimbingan Belajar Spectrum. *Journal Speed – Sentra Penelitian Engineering dan Edukasi*, Volume 4 No 2, 70.
- Wibowo, R. T. (2014). *ANALISIS DAN PERANCANGAN SISTEM E-MENU PADA RUMAH MAKAN MUARA KAPUAS*. YOGYAKARTA: STMIK AMIKOM YOGYAKARTA.
- Widianti, U. D. (2012). Pembangunan Sistem Informasi Aset Di PT.Industri Telekomunikasi Indonesia (Persero) Berbasis Web. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, Volume. I Nomor. 2, 58.



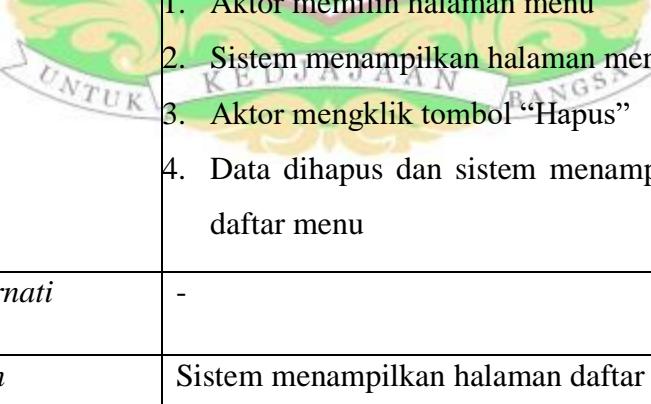
<i>Use case</i>	Login
<i>Actor</i>	Admin
<i>Entry Condition</i>	<i>Actor</i> belum login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor membuka aplikasi web e-menu</li> <li>2. Sistem manampilkan halaman login</li> <li>3. Aktor mengisi username dan password dan menekan tombol “Login”</li> <li>4. Sistem menampilkan halaman utama</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman utama

<i>Use case</i>	Mengelola Menu (Input)
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor telah login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman menu</li> <li>2. Sistem menampilkan halaman menu</li> <li>3. Aktor mengisi data menu dan menekan tombol “Tambah”</li> <li>4. Sistem menyimpan data ke database</li> <li>5. Sistem menampilkan halaman daftar menu</li> </ol>
<i>Scenario Alternatif</i>	<ol style="list-style-type: none"> <li>6. Aktor mengklik tombol “Tambah” saat form input data kosong</li> <li>7. Sistem menampilkan notifikasi error</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman daftar menu

<i>Use case</i>	Mengelola Menu (Edit)
-----------------	-----------------------



<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor telah login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman menu</li> <li>2. Sistem menampilkan halaman menu</li> <li>3. Aktor mengklik tombol “Edit”</li> <li>4. Sistem menampilkan halaman form edit menu</li> <li>5. Aktor mengubah data dalam form yang tersedia dan mengklik tombol “Simpan”</li> <li>6. Sistem menyimpan data ke database</li> <li>7. Sistem menampilkan halaman daftar menu</li> </ol>
<i>Scenario Alternati</i>	-
<i>Exit Condition</i>	Sistem menampilkan halaman daftar menu



<i>Use case</i>	Mengelola Menu (Hapus)
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor telah login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman menu</li> <li>2. Sistem menampilkan halaman menu</li> <li>3. Aktor mengklik tombol “Hapus”</li> <li>4. Data dihapus dan sistem menampilkan halaman daftar menu</li> </ol>
<i>Scenario Alternati</i>	-
<i>Exit Condition</i>	Sistem menampilkan halaman daftar menu

<i>Use case</i>	Mengelola Data Meja (Input)
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor telah login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman meja</li> <li>2. Sistem menampilkan halaman meja</li> <li>3. Aktor mengisi form data meja</li> <li>4. Sistem menampilkan daftar meja</li> </ol>
<i>Scenario Alternati</i>	-
<i>Exit Condition</i>	Sistem menampilkan halaman daftar meja

<i>Use case</i>	Mengelola Data Meja (Edit)
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor telah login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>5. Aktor memilih halaman meja</li> <li>6. Sistem menampilkan halaman meja</li> <li>7. Aktor mengklik tombol “Edit” dan mengisi form edit data meja</li> <li>8. Sistem menampilkan daftar meja</li> </ol>
<i>Scenario Alternati</i>	-
<i>Exit Condition</i>	Sistem menampilkan halaman daftar meja

<i>Use case</i>	Mengelola Data Meja (Hapus)
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor telah login

<i>Flow of Event</i>	9. Aktor memilih halaman meja 10. Sistem menampilkan halaman meja 11. Aktor mengklik tombol “Hapus” 12. Sistem menampilkan daftar meja
<i>Scenario Alternati</i>	-
<i>Exit Condition</i>	Sistem menampilkan halaman daftar meja

<i>Use Case</i>	Pengisian/isi ulang deposit <i>member</i>
<i>Actor</i>	Admin
<i>Entry Condition</i>	<i>Actor</i> telah <i>login</i>
<i>Flow of Event</i>	1. <i>Actor</i> membuka menu “Member” 2. Sistem menampilkan submenu <i>member</i> 3. <i>Actor</i> memilih submenu “Pengisian Deposit” 4. Sistem menampilkan halaman <i>scanning</i> “Pengisian Deposit Member” 5. <i>Actor</i> melakukan <i>scanning QR Code</i> <i>member</i> 6. Sistem membaca data <i>QR Code</i> <i>member</i> di <i>database</i> 7. Sistem menampilkan halaman <i>form top up</i> deposit <i>member</i> 8. <i>Actor</i> menginputkan nominal pengisian deposit dan menekan tombol “Tambah Deposit”
	1. Sistem menampilkan notifikasi top up berhasil dan menampilkan halaman “Riwayat Pengisian Deposit”
<i>Scenario Alternatif</i>	2. <i>Actor</i> melakukan <i>scanning QR Code</i> yang tidak terdaftar

	3. Sistem gagal membaca data <i>member</i> di <i>database</i> dan menampilkan halaman “Data Tidak ditemukan”
<i>Scenario Alternatif</i>	4. <i>Actor</i> menekan tombol “Top Up Deposit” saat inputan data salah atau kosong 5. Sistem menampilkan notifikasi kesalahan
<i>Exit Condition</i>	Sistem menampilkan notifikasi top up berhasil dan menampilkan halaman “Daftar Deposit Member”

<i>Use Case</i>	Cetak laporan transaksi
<i>Actor</i>	Admin
<i>Entry Condition</i>	<i>Actor</i> telah <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>Actor</i> membuka menu “Laporan” pada aplikasi</li> <li>2. Sistem menampilkan sub menu laporan</li> <li>3. <i>Actor</i> memilih sub menu transaksi</li> <li>4. Sistem menampilkan halaman “Laporan Transaksi”</li> <li>5. <i>Actor</i> memasukan tanggal periode laporan dan menekan tombol “Lihat Laporan”.</li> <li>6. Sistem menampilkan tabel data rekapan transaksi sesuai periode yang dipilih</li> <li>7. <i>Actor</i> menekan tombol cetak</li> <li>8. Sistem menampilkan dokumen <i>preview</i> percetakan laporan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan dokumen <i>preview</i> percetakan laporan



**LAMPIRAN B**

**STRUKTUR DATABASE**

## 1. Tabel Member

Nama atribut	Tipe data	Keterangan
ID_member	Integer (10)	PK
Nama	Varchar (30)	
No_telp	Varchar (13)	
Alamat	Longvarchar	
Saldo	Float	
reg_date	Date	
Email	Varchar (20)	
Password	Varchar (30)	

## 2. Tabel TopUp

Nama atribut	Tipe data	Keterangan
ID_topup	Integer (10)	PK
Nominal	Integer(10)	
Tanggal	Date	
Member_id	Integer(10)	FK
Password	Varchar (30)	

## 3. Tabel Status Bayar

Nama atribut	Tipe data	Keterangan
Id_status_bayar	Integer (10)	PK
Keterangan	Varchar (100)	

## 4. Tabel Status Order

Nama atribut	Tipe data	Keterangan
Id_status_order	Integer (10)	PK
Keterangan	Varchar (100)	

## 5. Tabel Transaksi

Nama atribut	Tipe data	Keterangan
Id_transaksi	Varchar (10)	PK
Meja_id	Integer (10)	FK
Tanggal	Date	
Member_id	Integer(10)	FK
Status_bayar_id	Integer (10)	FK
Total_harga	Integer (10)	

## 6. Tabel Menu

Nama atribut	Tipe data	Keterangan
ID_menu	Varchar (10)	PK
Nama	Varchar (255)	
Harga	Integer (10)	
Deskripsi	Varchar (255)	
Stok	Integer(10)	
Gambar	Varchar (255)	
Kategori_id	Integer(10)	FK

## 7. Tabel Meja

Nama atribut	Tipe data	Keterangan
ID_meja	Varchar (10)	PK
Nama_meja	Varchar (10)	

## 8. Tabel Kategori

Nama atribut	Tipe data	Keterangan
ID_kategori	Varchar (10)	PK
Nama_kategori	Varchar (10)	



UNIVERSITAS ANDALAS

LAMPIRAN C

## KODE PROGRAM APLIKASI

- Webservice (API)
  - Login



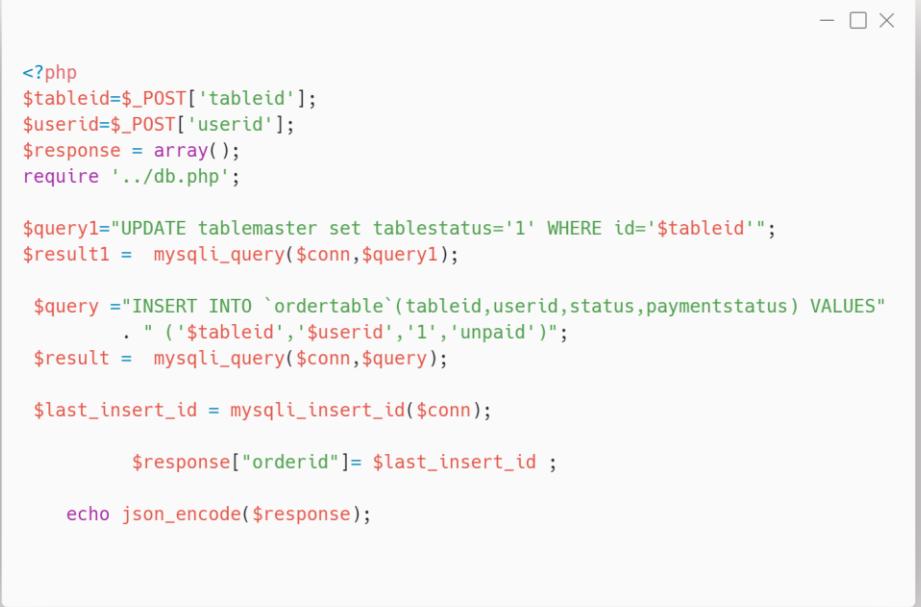
```

<?php
$username=$_POST["username"];
$password=$_POST["password"];
$response = array();
require '../db.php';
$result = mysqli_query($conn,"SELECT * FROM usermaster where username='".$username' and password='".$password"'") or die(mysql_error());

if (mysqli_num_rows($result) > 0) {
    while ($row = $result->fetch_assoc()) {
        $response["userid"]=$row['id'];
        $response["message"] ="Login Success";
        $response["success"] = 1;
    }
    echo json_encode($response);
} else {
    $response["success"] = 0;
    $response["message"] = "Invalid Username and Password";
    echo json_encode($response);
}

```

- Buat Pesanan



```

<?php
$tableid=$_POST['tableid'];
$userid=$_POST['userid'];
$response = array();
require '../db.php';

$query1="UPDATE tablemaster set tablestatus='1' WHERE id='".$tableid."'";
$result1 = mysqli_query($conn,$query1);

$query ="INSERT INTO `ordertable`(`tableid`, `userid`, `status`, `paymentstatus`) VALUES"
      . " (".$tableid.", '$userid', '1', 'unpaid')";
$result = mysqli_query($conn,$query);

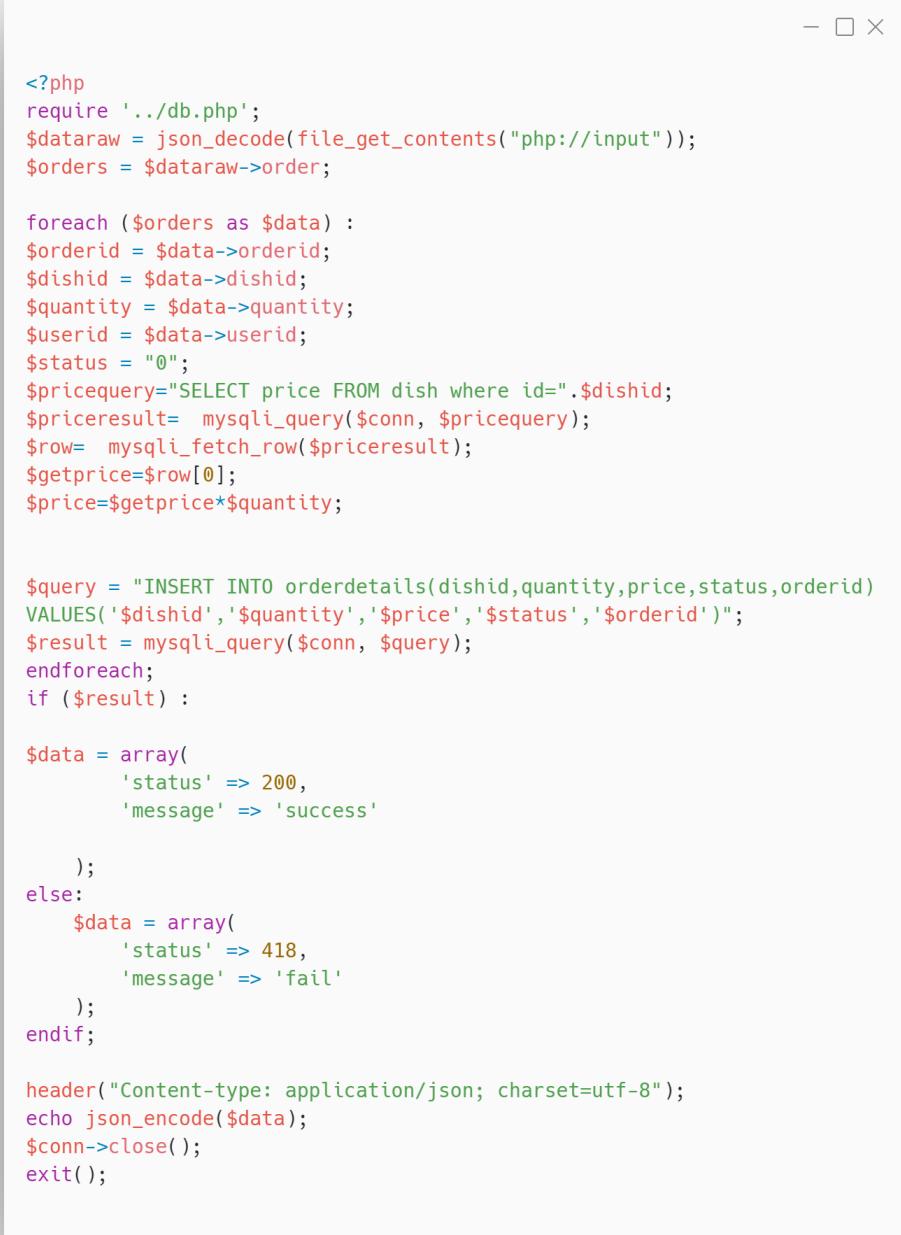
$last_insert_id = mysqli_insert_id($conn);

$response["orderid"]= $last_insert_id ;

echo json_encode($response);

```

- Update Pesanan



```
<?php
require '../db.php';
$dataraw = json_decode(file_get_contents("php://input"));
$orders = $dataraw->order;

foreach ($orders as $data) :
$orderid = $data->orderid;
$dishid = $data->dishid;
$quantity = $data->quantity;
$userid = $data->userid;
$status = "0";
$pricequery="SELECT price FROM dish where id=".$dishid;
$priceresult= mysqli_query($conn, $pricequery);
$row= mysqli_fetch_row($priceresult);
$getprice=$row[0];
$price=$getprice*$quantity;

$query = "INSERT INTO orderdetails(dishid,quantity,price,status,orderid)
VALUES('$dishid','$quantity','$price','$status','$orderid')";
$result = mysqli_query($conn, $query);
endforeach;
if ($result) :

$data = array(
    'status' => 200,
    'message' => 'success'
);

else:
$data = array(
    'status' => 418,
    'message' => 'fail'
);
endif;

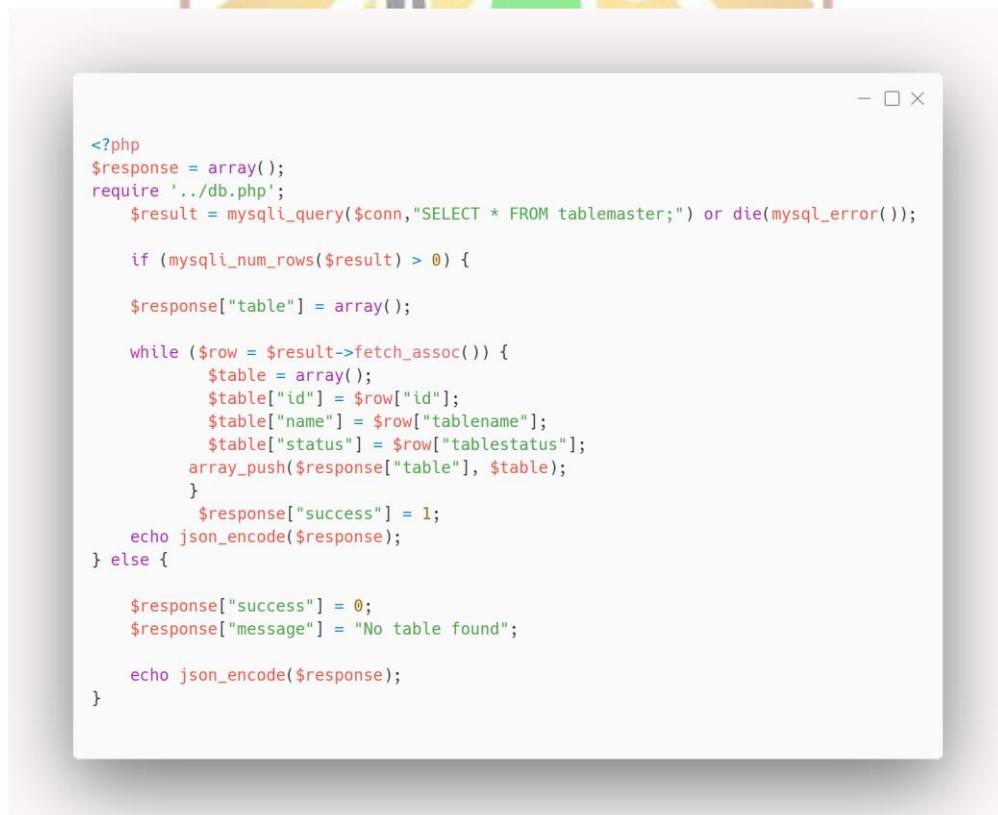
header("Content-type: application/json; charset=utf-8");
echo json_encode($data);
$conn->close();
exit();
```

- Daftar Menu Berdasarkan Kategori



```
<?php
$cusineid = $_POST['cusineid'];
$response = array();
require '../db.php';
$result = mysqli_query($conn, "SELECT * FROM dish WHERE cusineid='$cusineid';") or die(mysql_error());
if (mysqli_num_rows($result) > 0) {
    while ($row = mysqli_fetch_array($result)) {
        $cusinedish = array();
        $cusinedish["dishid"] = $row[0];
        $cusinedish["cusineid"] = $row[1];
        $cusinedish["dishname"] = $row[2];
        $cusinedish["dishimage"] = $row[3];
        $cusinedish["description"] = $row[4];
        $cusinedish["price"] = $row[5];
        array_push($response, $cusinedish);
    }
    echo json_encode($response);
} else {
    $response["message"] = "No dish found";
    echo json_encode($response);
}
```

- Daftar Meja



```
<?php
$response = array();
require '../db.php';
$result = mysqli_query($conn, "SELECT * FROM tablemaster;") or die(mysql_error());
if (mysqli_num_rows($result) > 0) {
    $response["table"] = array();
    while ($row = $result->fetch_assoc()) {
        $table = array();
        $table["id"] = $row["id"];
        $table["name"] = $row["tablename"];
        $table["status"] = $row["tablestatus"];
        array_push($response["table"], $table);
    }
    $response["success"] = 1;
    echo json_encode($response);
} else {
    $response["success"] = 0;
    $response["message"] = "No table found";
    echo json_encode($response);
}
```

- Daftar Semua Menu



```
<?php
$response = array();
require '../db.php';
$result = mysqli_query($conn, "SELECT * FROM dish;") or die(mysql_error());

if (mysqli_num_rows($result) > 0) {

    while ($row = $result->fetch_assoc()) {

        $dish = array();
        $dish["id"] = $row["id"];
        $dish["cusineid"] = $row["cusineid"];
        $dish["dishname"] = $row["dishname"];
        $dish["dishtype"] = $row["dishtype"];
        $dish["description"] = $row["description"];
        $dish["price"] = $row["price"];

        $dish["dishimage"] = $row["dishimage"];
        array_push($response, $dish);
    }

    echo json_encode($response);
} else {

    $response["message"] = "No dish found";
    echo json_encode($response);
}
```



- Checkout Pesanan

```

<?php
include '../db.php';
$orderId = $_POST['orderid'];
$response = array();
$respondedish = array();
$resultnew = mysqli_query($conn, "SELECT startdatetime,status,amount,grandtotal,paymentmode,discount,id
FROM ordertable where id='$orderId'");
if (mysqli_num_rows($resultnew) > 0) {
    $order = array();
    $orderdish=array();
    $rownew = mysqli_fetch_row($resultnew);
    $startdatetime = $rownew[0];
    $status = $rownew[1];
    $amount = $rownew[2];
    $grandtotal = $rownew[3];
    $model = $rownew[4];
    $disc = $rownew[5];
    $orderid = $rownew[6];
    $typem = mysqli_query($conn, "select type from paymentmode where id='$model'");
    $typerow = mysqli_fetch_row($typem);
    $mode = $typerow[0];
    if ($status == "0")
        $status1 = "Unpaid";
    else if ($status == "1")
        $status1 = "Process";
    else if ($status == "2")
        $status1 = "Paid";
    else if ($status == "3")
        $status1 = "Finish";
    else
        $status1 = "Cancel";
    $order["orderid"] = $orderid;
    //echo "orderid" . $orderid . "<br/>";
    $order["startdatetime"] = $startdatetime;
    //echo "time:" . $startdatetime . "<br/>";
    //echo "status:" . $status . "<br/>";
    $order["status"] = $status;

    $result = mysqli_query($conn, "SELECT od.price,od.quantity,od.dishid,d.dishname FROM ordertable
ot,orderdetails od,dish d where ot.id='$orderId' and od.orderid=ot.id and od.dishid=d.id") or
die(mysqli_error());
    $amount = 0;
    $grosstotal = 0;
    if (mysqli_num_rows($result) > 0) {
        while ($row = mysqli_fetch_array($result)) {
            $price = $row[0];
            $quantity = $row[1];
            $dish = $row[3];
            $amount = $amount + $price;
            //echo "dish:" . $dish . "<br/>";
            $orderdish["dishnm"] = $dish;
            //echo "quantity:" . $quantity . "<br/>";
            $orderdish["quantity"] = $quantity;
            //echo "price:" . $price . "<br/>";
            $orderdish["price"] = $price;
            array_push($respondedish, $orderdish);
        }
    }
    $order["amount"] = $amount;
    $order["grosstotal"] = $grosstotal;
    array_push($response, $order);
    echo json_encode($respondedish);
    echo json_encode($response);
}
else {
    $response["success"] = 0;
    $response["message"] = "No Order found";
    // echo no users JSON
    echo json_encode($response);
}
?>

```

- Tambah Kategori



```
<?php
ob_start();
include("db.php");
$cusinename='';
if (isset($_REQUEST['cusinename'])):
    if(!empty($_REQUEST['cusinename'])):
        $cusinename = safe($conn,htmlspecialchars($_POST['cusinename']));
    else:
        header("Location: addCuisine.php?add_cdata=empty");
        die();
    endif;
    else:
        header("Location: addCuisine.php?add_cdata=empty");
        die();
    endif;
$cusineimage = $_FILES['cusineimage']['name'];

$query ="INSERT INTO `cuisine`(`cusinename,cusineimage`) VALUES ('$cusinename','$cusineimage')";
$result = mysqli_query($conn,$query);
if($result==1){

    $file_path = 'upload/';

    $file_path = $file_path . basename( $_FILES['cusineimage']['name']);
    echo "image path:".$file_path;
    if(move_uploaded_file($_FILES['cusineimage']['tmp_name'], $file_path)) {

        header("Location: addCusinedb.php?add_cdata=success");
    } else{
        header("Location: addCusinedb.php?add_cdata=failed");
    }
} else {
    header("Location: addCusinedb.php?add_cdata=failed");
    die();
}

function safe($conn,$value){
    return mysqli_real_escape_string($conn,$value);
}
```

- Tambah Menu



```

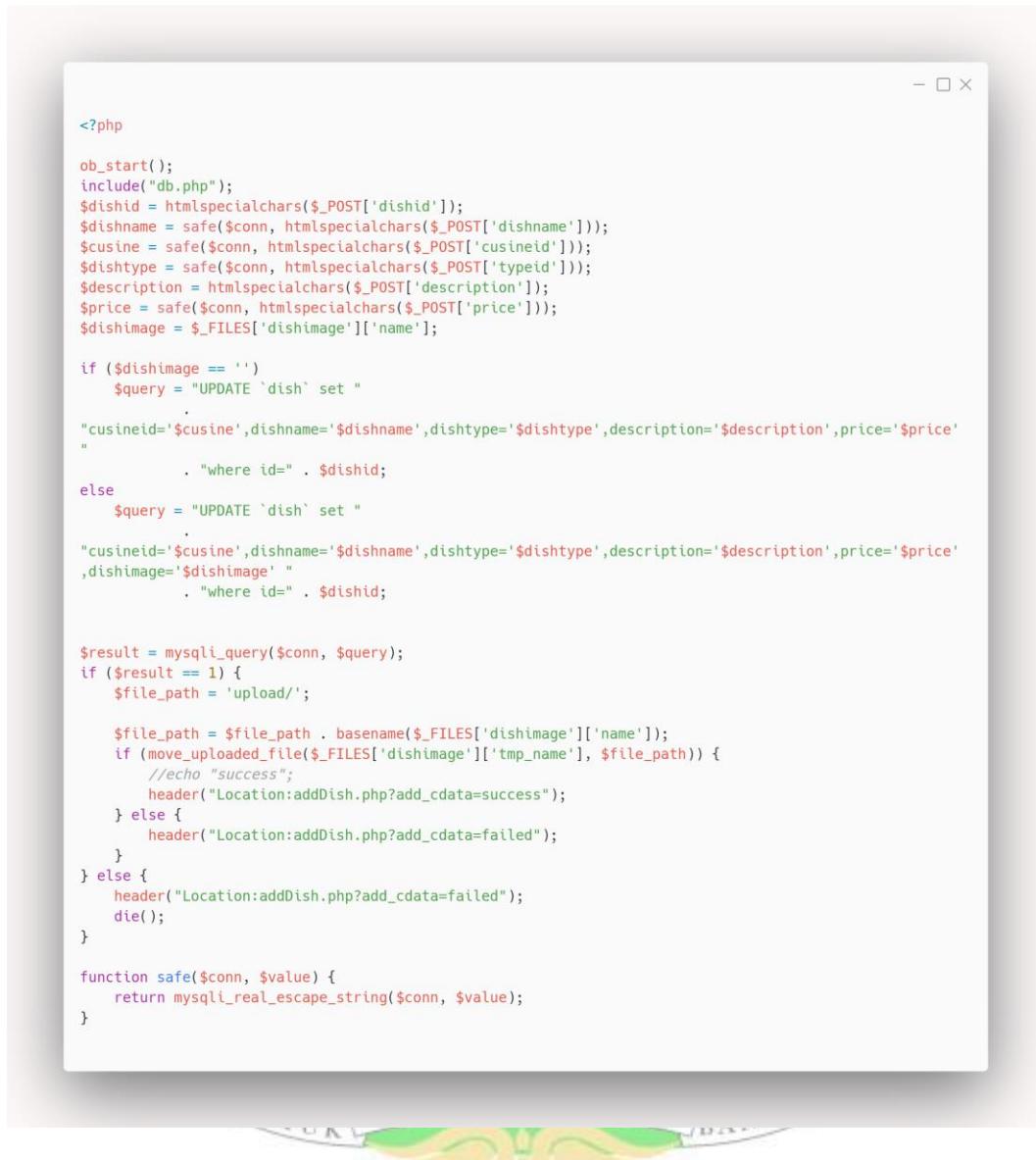
<?php
ob_start();
include("db.php");
$cuisine=htmlspecialchars($_POST['cuisineid']);
$dishname=htmlspecialchars($_POST['dishname']);
$desc=htmlspecialchars($_POST['description']);
$price=htmlspecialchars($_POST['price']);
// $type=htmlspecialchars($_POST['typeid']);
$dishimage = $_FILES['dishimage']['name'];
$dishflag=0;
$checkdish= mysqli_query($conn,"SELECT dishname FROM dish");
if(mysqli_num_rows($checkdish)>0)
{
    while ($row= mysqli_fetch_row($checkdish))
    {
        $dishnm=$row[0];
        if(strcasecmp($dishnm, $dishname)==0)
            $dishflag=1;
    }
}
else {
    $dishflag=0;
}
if($dishflag==0)
{
    $query ="INSERT INTO `dish`(`cuisineid`, `dishname`, `dishimage`, `description`, `price`) VALUES
    ('$cuisine', '$dishname', '$dishimage', '$desc', '$price')";
    $result = mysqli_query($conn,$query);
}
else {
    header("Location: addDish.php?add_rdata=checkdishname");
    die();
}

if($result==1){
    $file_path = 'upload/';
    $file_path = $file_path . basename( $_FILES['dishimage']['name']);
    if(move_uploaded_file($_FILES['dishimage']['tmp_name'], $file_path)) {
        header("Location: addDish.php?add_cdata=success");
    } else{
        header("Location: addDish.php?add_cdata=failed");
    }
}
else {
    header("Location: addDish.php?add_cdata=failed");
    die();
}

function safe($conn,$value){
    return mysqli_real_escape_string($conn,$value);
}

```

- Edit Menu



```
<?php

ob_start();
include("db.php");
$dishid = htmlspecialchars($_POST['dishid']);
$dishname = safe($conn, htmlspecialchars($_POST['dishname']));
$cusine = safe($conn, htmlspecialchars($_POST['cusineid']));
$dishtype = safe($conn, htmlspecialchars($_POST['typeid']));
$description = htmlspecialchars($_POST['description']);
$price = safe($conn, htmlspecialchars($_POST['price']));
$dishimage = $_FILES['dishimage']['name'];

if ($dishimage == '')
    $query = "UPDATE `dish` set "
    .
    "cusineid='$cusine',dishname='$dishname',dishtype='$dishtype',description='$description',price='$price'
    "
    .
    "where id=" . $dishid;
else
    $query = "UPDATE `dish` set "
    .
    "cusineid='$cusine',dishname='$dishname',dishtype='$dishtype',description='$description',price='$price'
    ,
    dishimage='$dishimage'
    "
    .
    "where id=" . $dishid;

$result = mysqli_query($conn, $query);
if ($result == 1) {
    $file_path = 'upload/';

    $file_path = $file_path . basename($_FILES['dishimage']['name']);
    if (move_uploaded_file($_FILES['dishimage']['tmp_name'], $file_path)) {
        //echo "success";
        header("Location:addDish.php?add_cdata=success");
    } else {
        header("Location:addDish.php?add_cdata=failed");
    }
} else {
    header("Location:addDish.php?add_cdata=failed");
    die();
}

function safe($conn, $value) {
    return mysqli_real_escape_string($conn, $value);
}
```

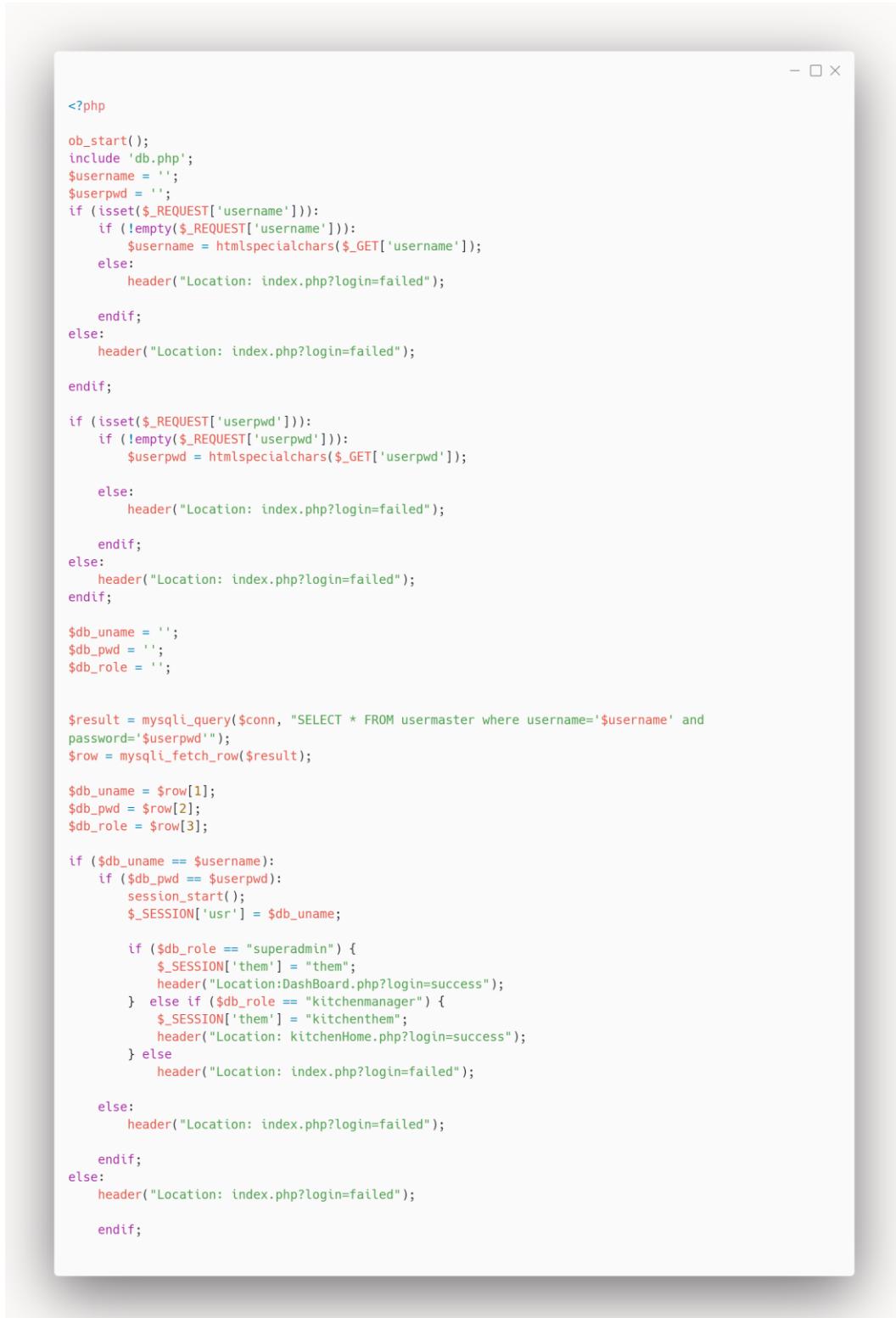
- Tambah Meja



```
<?php
ob_start();
include("db.php");
$tablename = safe($conn,htmlspecialchars($_POST['tablename']));
$tableflag=0;
$checktable= mysqli_query($conn,"SELECT tablename FROM tablemaster");
if(mysqli_num_rows($checktable)>0)
{
    while ($row= mysqli_fetch_row($checktable))
    {
        $tablenm=$row[0];
        if(strcasecmp($tablenm, $tablename)==0)
            $tableflag=1;
    }
}
if($tableflag==0)
{
    $query ="INSERT INTO `tablemaster`(`tablename`,tablestatus) VALUES"
        . " ('$tablename','0')";
    $result = mysqli_query($conn,$query);
}
else {
    header("Location: addTable.php?add_rdata=checktablename");
    die();
}
if($result==1){
    header("Location: addTable.php?add_rdata=success");
} else {
    header("Location: addTable.php?add_rdata=failed");
    die();
}

function safe($conn,$value){
    return mysqli_real_escape_string($conn,$value);
}
```

- Cek Login



```

<?php

ob_start();
include 'db.php';
$username = '';
$userpwd = '';
if (isset($_REQUEST['username'])):
    if (!empty($_REQUEST['username'])):
        $username = htmlspecialchars($_GET['username']);
    else:
        header("Location: index.php?login=failed");
    endif;
else:
    header("Location: index.php?login=failed");
endif;

if (isset($_REQUEST['userpwd'])):
    if (!empty($_REQUEST['userpwd'])):
        $userpwd = htmlspecialchars($_GET['userpwd']);
    else:
        header("Location: index.php?login=failed");
    endif;
else:
    header("Location: index.php?login=failed");
endif;

$db_uname = '';
$db_pwd = '';
$db_role = '';

$result = mysqli_query($conn, "SELECT * FROM usermaster where username='$username' and password='$userpwd'");
$row = mysqli_fetch_row($result);

$db_uname = $row[1];
$db_pwd = $row[2];
$db_role = $row[3];

if ($db_uname == $username):
    if ($db_pwd == $userpwd):
        session_start();
        $_SESSION['usr'] = $db_uname;

        if ($db_role == "superadmin") {
            $_SESSION['them'] = "them";
            header("Location:DashBoard.php?login=success");
        } else if ($db_role == "kitchenmanager") {
            $_SESSION['them'] = "kitchenthem";
            header("Location: kitchenHome.php?login=success");
        } else
            header("Location: index.php?login=failed");

    else:
        header("Location: index.php?login=failed");
    endif;
else:
    header("Location: index.php?login=failed");
endif;

```

- Detail Transaksi

```

<?php
session_start();
include 'db.php';
include 'index.php';
$tax = 0;
$vatTax = 0;
$additionalTax = 0;
$addDiscValue = 0;
$amount = 0;
$disc = 0;
$tot = 0;
$paymentMode = '';
$msg = '';
$grossTotal = 0;
if (!isset($_SESSION['usr'])) {
    header("Location: index.php");
}
$oid = '';
if (isset($_REQUEST['oid'])):
    if (!empty($_REQUEST['oid'])):
        $oid = htmlspecialchars($_GET['oid']);
    else:
        header("Location:viewOrderDetail.php?update=failed");
    endif;
else:
    header("Location:viewOrderDetail.php?update=failed");
endif;
//ReActive Table Button Click
if (isset($_POST['btntable'])) {
    $gettable = mysqli_query($conn, "SELECT tableId from ordertable where id='$oid'");
    $rowtable = mysqli_fetch_row($gettable);
    $tid = $rowtable[0];
    $chtbst = mysqli_query($conn, "select tablestatus from tablemaster where id='$tid'");
    $chrow = mysqli_fetch_row($chtbst);
    $stt = $chrow[0];
    if ($stt == '0') {
        mysqli_query($conn, "UPDATE tablemaster SET tablestatus='1' where id='$tid'");
        $updateordertable = "UPDATE ordertable SET status='1',paymentstatus='unpaid',paymentmode='0' where id='$oid'";
        $msg = 'Table ReActive';
        $result = mysqli_query($conn, $updateordertable);
        header("Location:Order.php");
    } else {
        $msg = ' Table is Inactive';
    }
}
//Order cancel Button Click
if (isset($_POST['btncancel'])) {
    mysqli_query($conn, "UPDATE ordertable SET status='4' where id='$oid'");
    mysqli_query($conn, "UPDATE orderdetails SET status='3' where orderId='$oid'");
    $getdata = mysqli_query($conn, "SELECT tableId from ordertable where id='$oid'");
    $getrow = mysqli_fetch_row($getdata);
    $tid = $getrow[0];
    mysqli_query($conn, "UPDATE tablemaster SET tablestatus='0' where id='$tid'");
}
//Paid Button Click
if (isset($_POST['btnsubmit'])) {
    $paymentMode = $_POST['paymentmode'];
    $amount = $_POST['amount'];
    $grandtotal = $_POST['grandtotal'];
    $discount = $_POST['disc'];
    if ($paymentMode == 0) {
        $msg = "Select Paymentmode";
    } else {
        $updateordertable = "UPDATE ordertable SET
amount='$amount',grandtotal='$grandtotal',status='3',paymentstatus='paid',paymentmode='$paymentmode',di
scount='$discount' where id='$oid'";
        $result = mysqli_query($conn, $updateordertable);

        $gettable = mysqli_query($conn, "SELECT tableId from ordertable where id='$oid'");
        $rowtable = mysqli_fetch_row($gettable);
        $tid = $rowtable[0];

        mysqli_query($conn, "UPDATE orderdetails SET status='2' where orderId='$oid'");
        mysqli_query($conn, "UPDATE tablemaster SET tablestatus='0' where id='$tid'");
        header("Location:paid.php?oid=$oid");
    }
}
?>

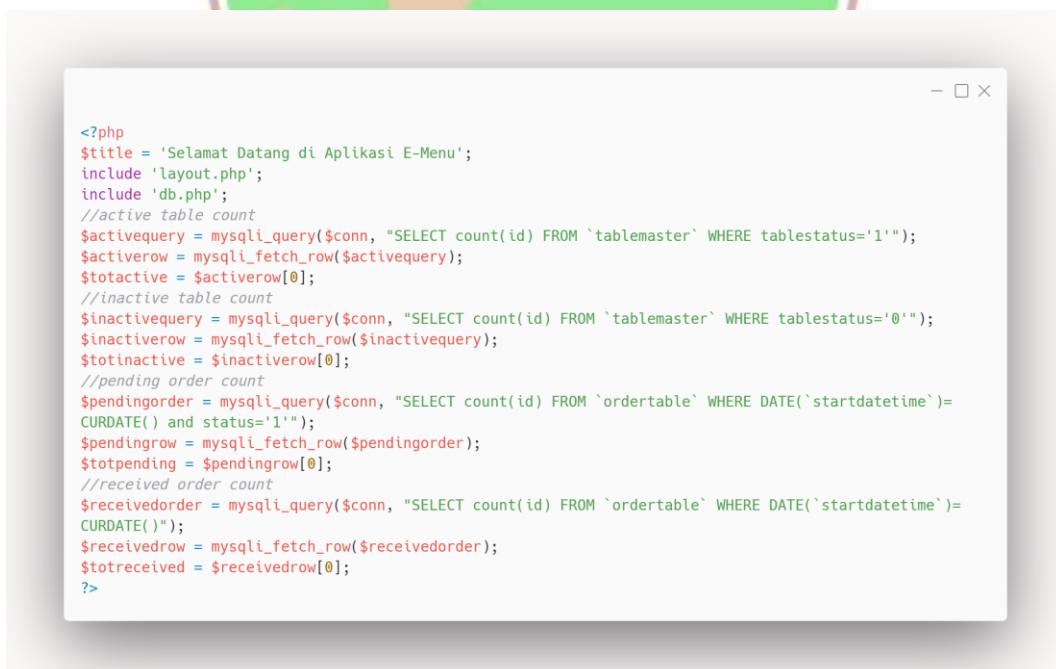
```

- Ubah Status Pesanan



```
<?php
session_start();
include 'db.php';
if (isset($_SESSION['them'])) {
    $them = $_SESSION['them'] . ".php";
    include $them;
}
$st = '';
if (isset($_REQUEST['kid'])) {
    $status = $_REQUEST['status'];
    $id = $_REQUEST['kid'];
    if ($status == "0") {
        $status = "1";
    } else if ($status == "1") {
        $status = "2";
    } else if ($status == "2") {
        $status = "2";
    }
    $updatestatus = "UPDATE orderdetails SET status='$status' WHERE id='$id'";
    mysqli_query($conn, $updatestatus);
}
?>
```

- Dashboard



```
<?php
$title = 'Selamat Datang di Aplikasi E-Menu';
include 'layout.php';
include 'db.php';
//active table count
$activequery = mysqli_query($conn, "SELECT count(id) FROM `tablemaster` WHERE tablestatus='1'");
$activrow = mysqli_fetch_row($activequery);
$totactive = $activrow[0];
//inactive table count
$inactivequery = mysqli_query($conn, "SELECT count(id) FROM `tablemaster` WHERE tablestatus='0'");
$inactivrow = mysqli_fetch_row($inactivequery);
$totinactive = $inactivrow[0];
//pending order count
$pendingorder = mysqli_query($conn, "SELECT count(id) FROM `ordertable` WHERE DATE(`startdatetime`)=
CURDATE() and status='1'");
$pendingrow = mysqli_fetch_row($pendingorder);
$totpending = $pendingrow[0];
//received order count
$receivedorder = mysqli_query($conn, "SELECT count(id) FROM `ordertable` WHERE DATE(`startdatetime`)=
CURDATE()");
$receivedrow = mysqli_fetch_row($receivedorder);
$totreceived = $receivedrow[0];
?>
```



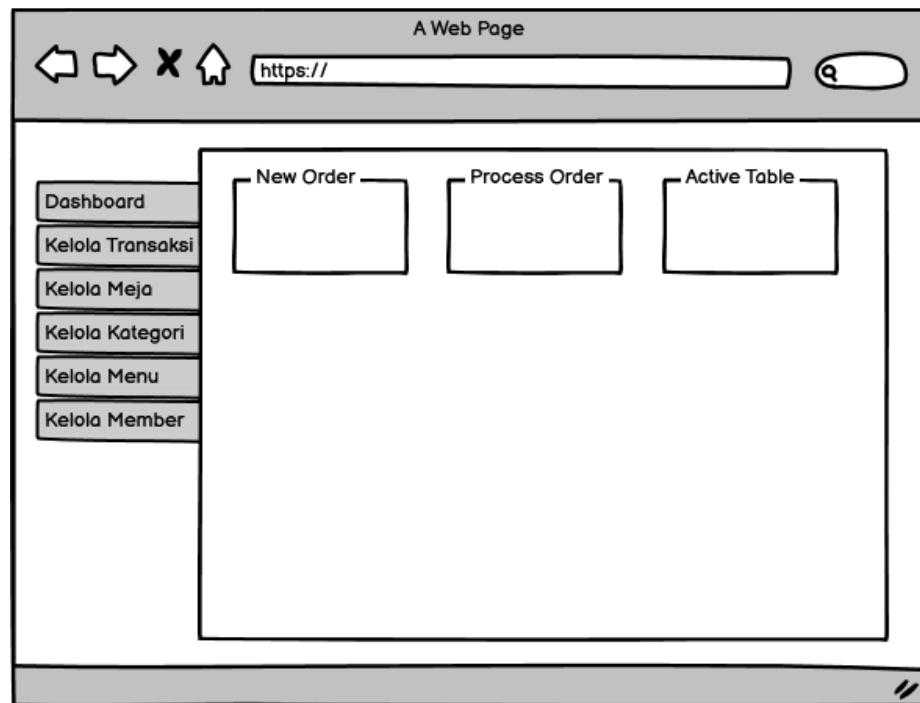
UNIVERSITAS ANDALAS

LAMPIRAN D

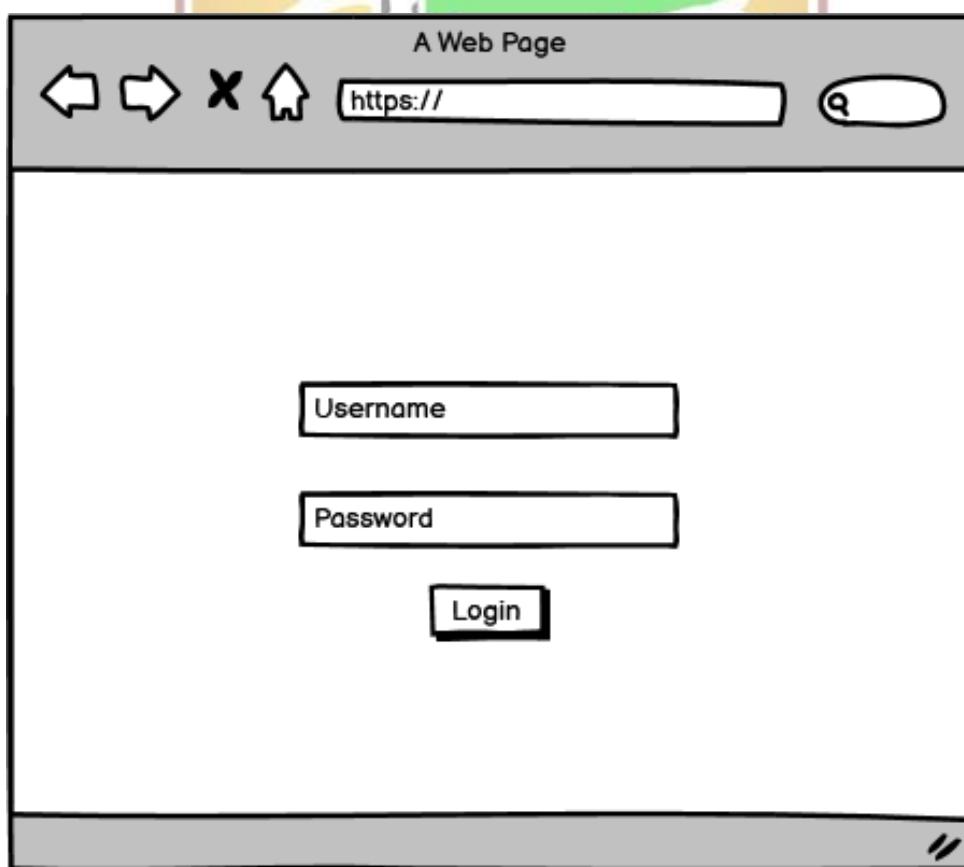
RANCANGAN ANTARMUKA

UNTUK KEDAJAAN BANGSA

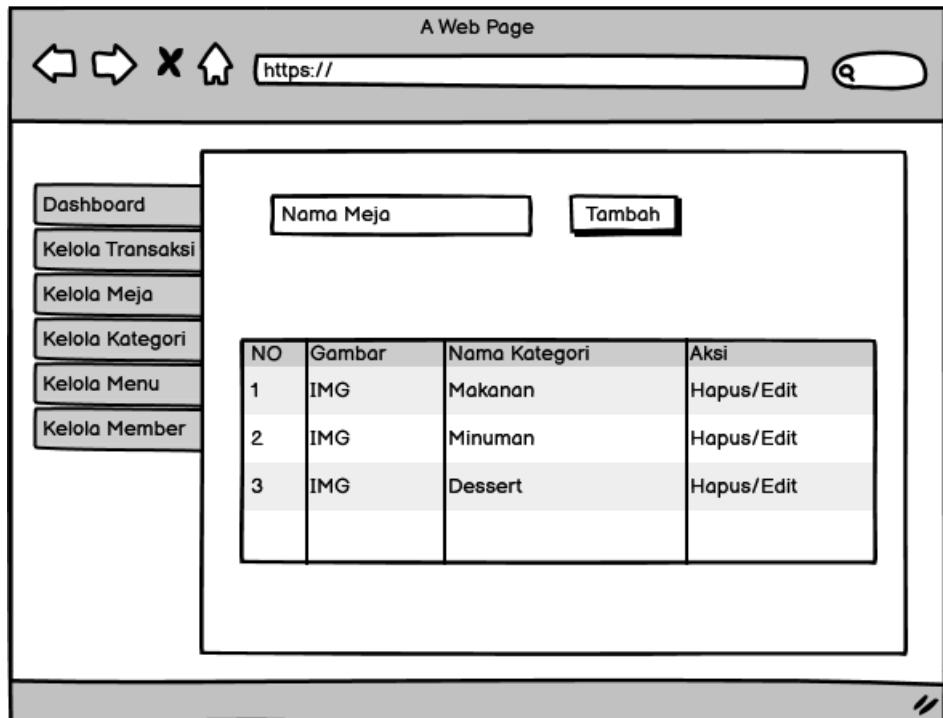
1. Dashboard Aplikasi Web



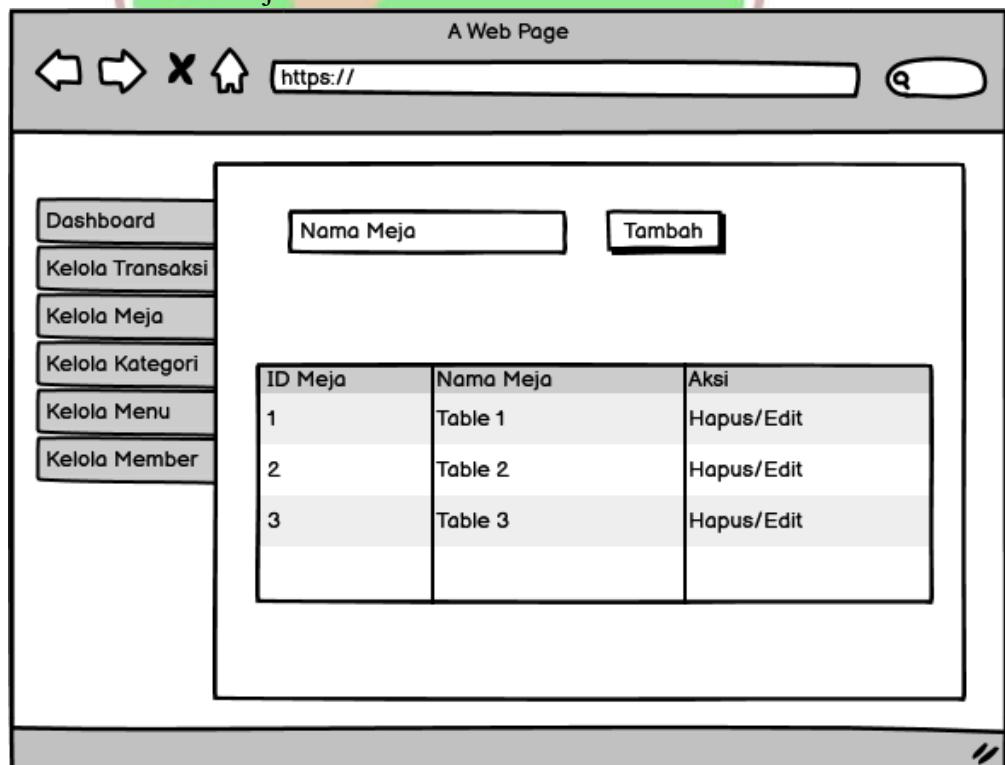
2. Halaman Login



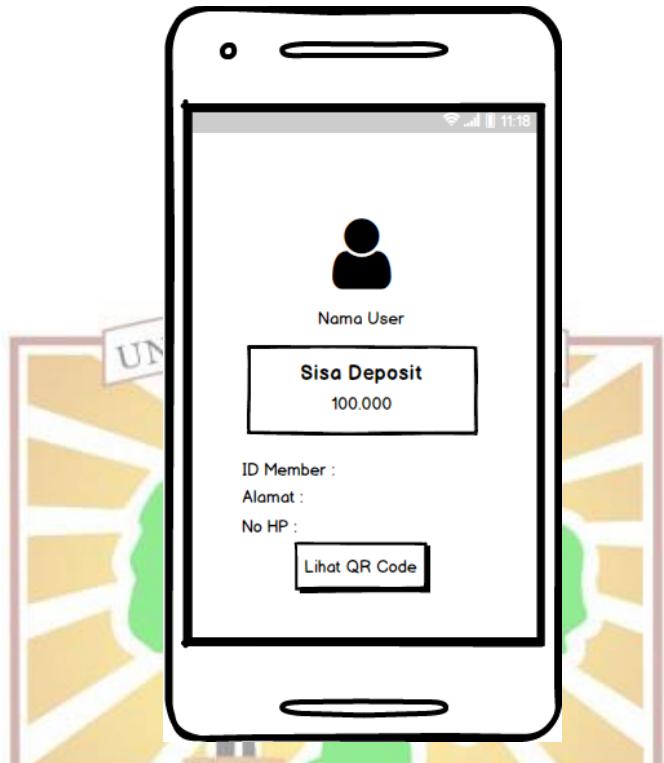
### 3. Halaman Daftar Menu



### 4. Halaman Daftar Meja

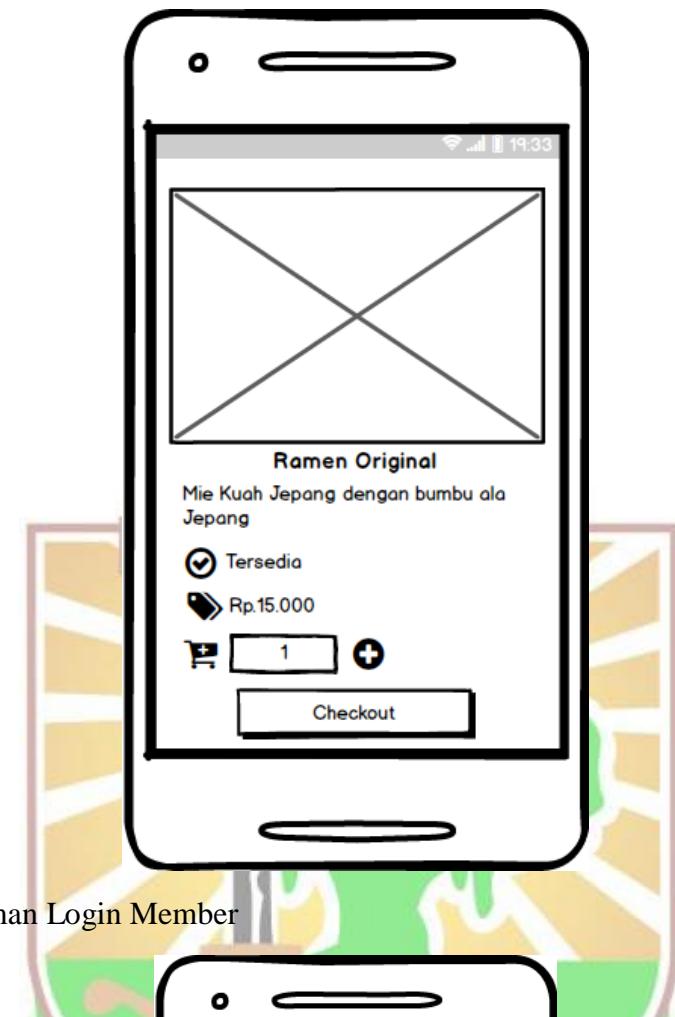


5. Halaman Profil Member

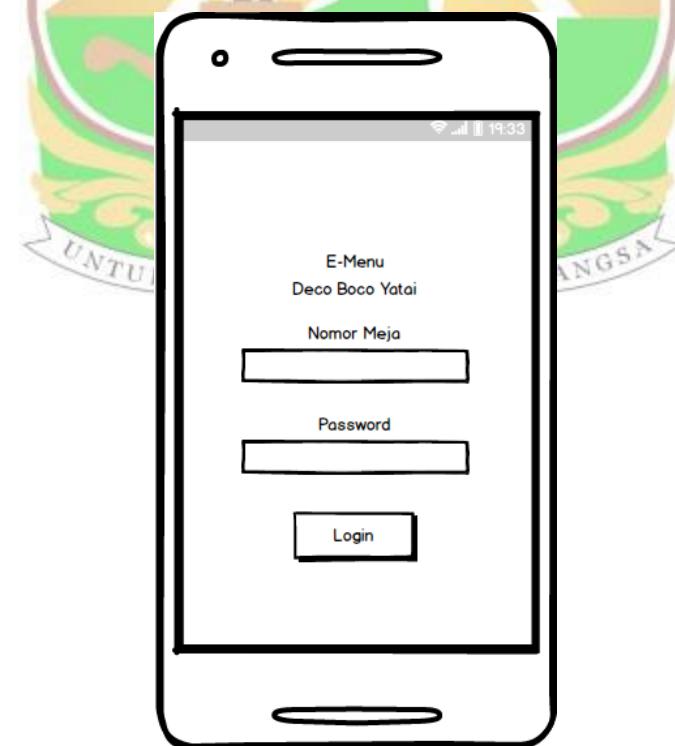


6. Halaman Detail Menu





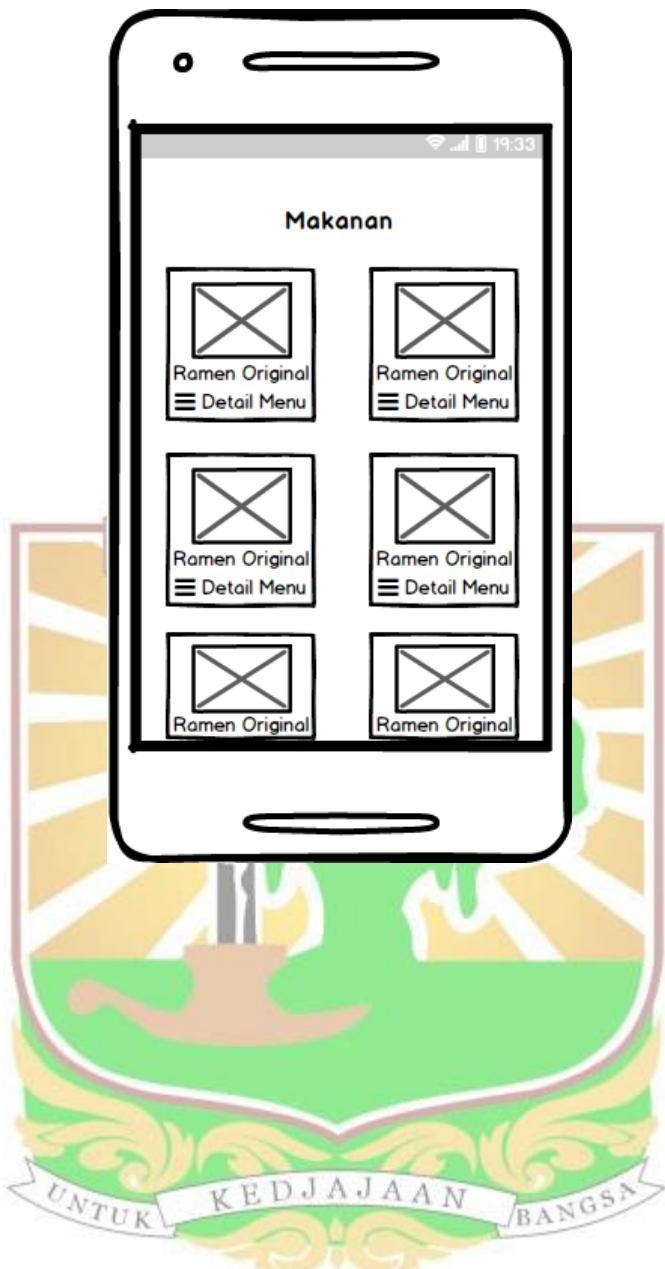
7. Halaman Login Member



8. Halaman Utama Aplikasi *Mobile*



9. Halaman Daftar Menu





UNIVERSITAS ANDALAS

LAMPIRAN E

IMPLEMENTASI ANTARMUKA

UNTUK KEDAJAAN BANGSA

## 1. Halaman Dashboard Aplikasi Web

The dashboard displays the following real-time data:

- MEJA TERISI**: 5 Meja
- DAFTAR MENU**: 14 Menu
- ORDER**: 5 Order

## 2. Halaman Daftar Meja

The page includes a table of existing tables:

SrNo	Table Name	Actions
1	Table 2	<a href="#">Cek</a> <a href="#">Hapus</a>
2	Table 3	<a href="#">Cek</a> <a href="#">Hapus</a>
3	Table 4	<a href="#">Cek</a> <a href="#">Hapus</a>
4	Table 5	<a href="#">Cek</a> <a href="#">Hapus</a>
5	Table 6	<a href="#">Cek</a> <a href="#">Hapus</a>
6	Table 7	<a href="#">Cek</a> <a href="#">Hapus</a>

## 3. Halaman Edit Data Meja

The page includes fields for updating a table:

Table Name:

status:

[Update Table](#)

#### 4. Halaman Daftar Kategori

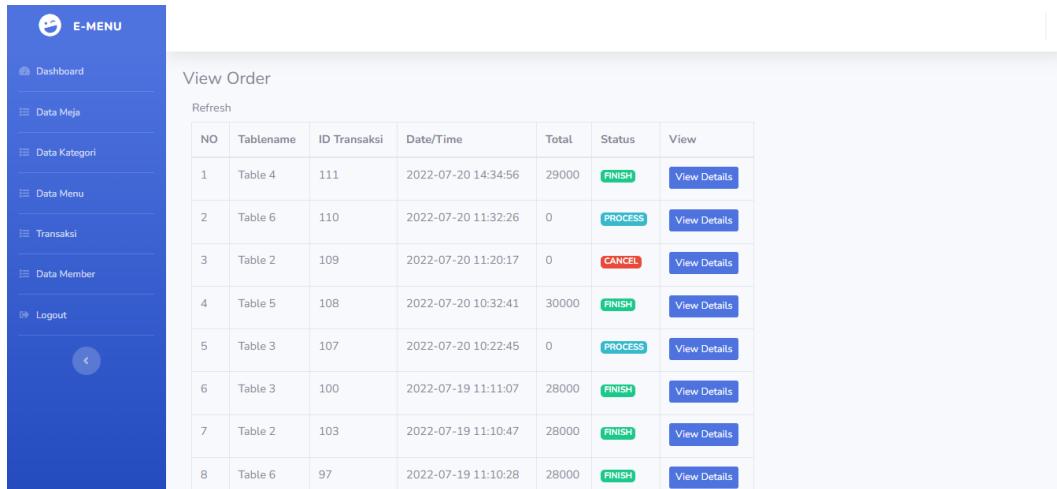
SrNo	Foto Kategori	Nama Kategori	Actions
1		Makanan	<button>Cek</button> <button>Hapus</button>
2		Minuman	<button>Cek</button> <button>Hapus</button>
3		Snack	<button>Cek</button> <button>Hapus</button>
4		Dessert	<button>Cek</button> <button>Hapus</button>

#### 5. Halaman Tambah Menu

#### 6. Halaman Daftar Menu

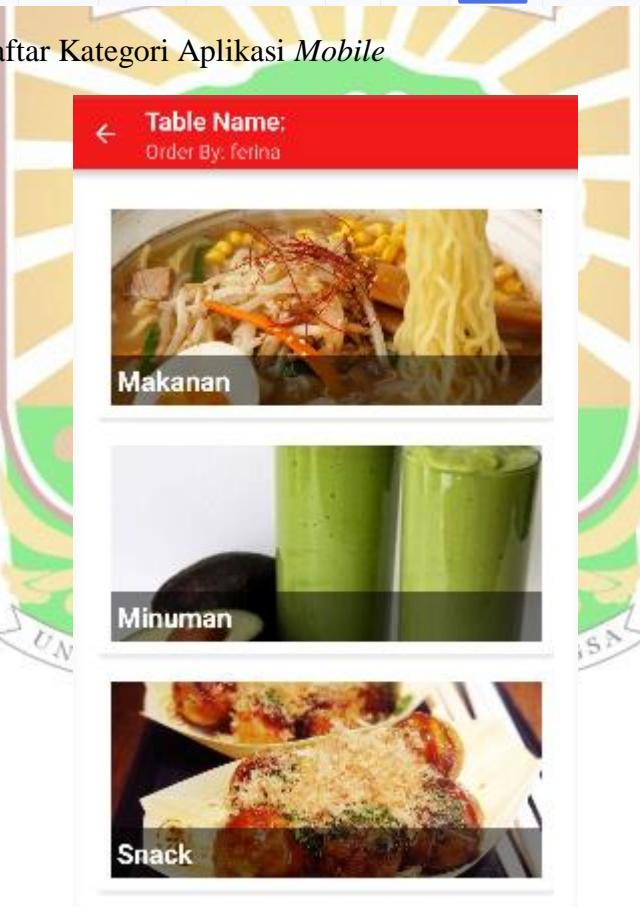
SrNo	Image	Dish Name	Cusine Name	Discription	Price (RS)	Actions
1		Ramen Original	Makanan	Ramen Original	14000	<button>Cek</button> <button>Hapus</button>
2		Takoyaki	Makanan	Takoyaki	15000	<button>Cek</button> <button>Hapus</button>
3		Okonomiyaki	Makanan	Okonomiyaki	18000	<button>Cek</button> <button>Hapus</button>
4		Ramen Chicken Katsu	Makanan	Ramen Chicken Katsu	20000	<button>Cek</button> <button>Hapus</button>
5		Ramen Teriyaki	Makanan	Ramen Teriyaki	20000	<button>Cek</button> <button>Hapus</button>
6		Ramen Sosei	Makanan	Ramen Sosei	20000	<button>Cek</button> <button>Hapus</button>
7		Chicken Katsu Nasi	Makanan	Chicken Katsu Nasi	20000	<button>Cek</button> <button>Hapus</button>
8		Yakisoba	Makanan	Yakisoba	16000	<button>Cek</button> <button>Hapus</button>

## 7. Halaman Transaksi



NO	Tablename	ID Transaksi	Date/Time	Total	Status	View
1	Table 4	111	2022-07-20 14:34:56	29000	FINISH	<a href="#">View Details</a>
2	Table 6	110	2022-07-20 11:32:26	0	PROCESS	<a href="#">View Details</a>
3	Table 2	109	2022-07-20 11:20:17	0	CANCEL	<a href="#">View Details</a>
4	Table 5	108	2022-07-20 10:32:41	30000	FINISH	<a href="#">View Details</a>
5	Table 3	107	2022-07-20 10:22:45	0	PROCESS	<a href="#">View Details</a>
6	Table 3	100	2022-07-19 11:11:07	28000	FINISH	<a href="#">View Details</a>
7	Table 2	103	2022-07-19 11:10:47	28000	FINISH	<a href="#">View Details</a>
8	Table 6	97	2022-07-19 11:10:28	28000	FINISH	<a href="#">View Details</a>

## 8. Halaman Daftar Kategori Aplikasi Mobile



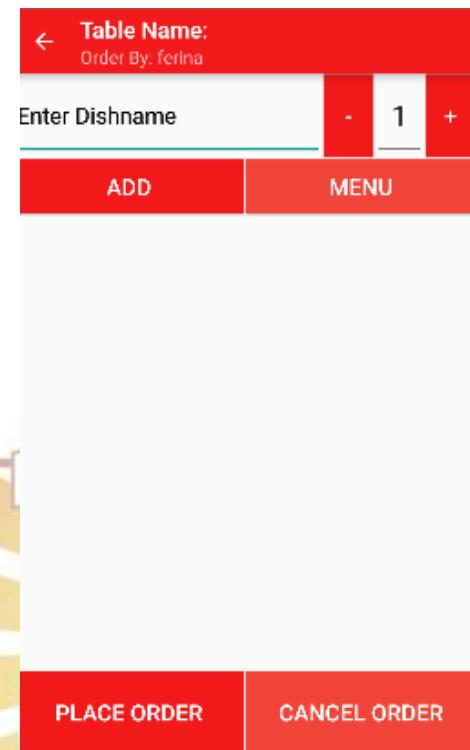
← Table Name: Order By: ferina

**Makanan**

**Minuman**

**Snack**

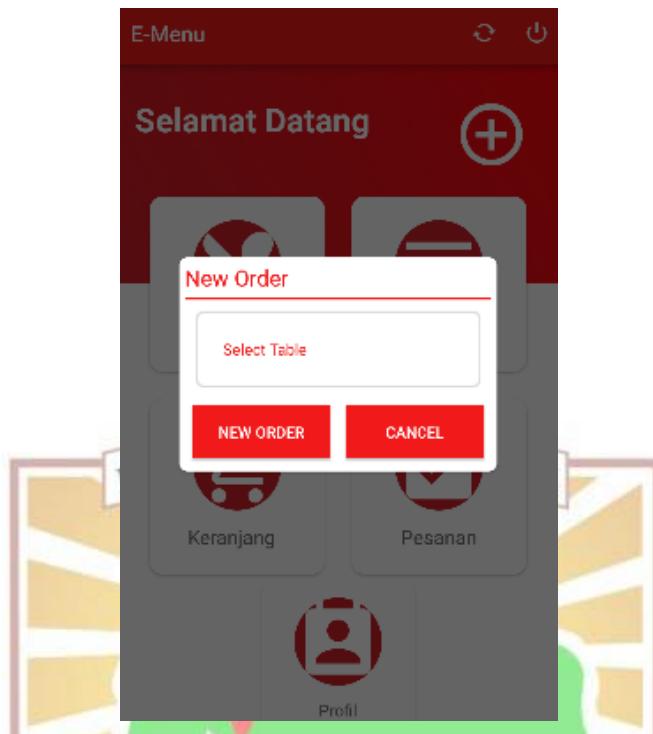
## 9. Halaman Pencarian Menu



## 10. Halaman Daftar Menu



11. Halaman Pilih Meja



12. Halaman Utama Aplikasi Mobile



PEMBANGUNAN APLIKASI PEMESANAN MENU MAKANAN  
SECARA ONLINE (E-MENU) DENGAN PEMBAYARAN NON-TUNAI  
MENGGUNAKAN TEKNOLOGI QR-CODE BERBASIS WEB DAN  
MOBILE PADA CAFE DECO BOCO YATAI

ORIGINALITY REPORT

<b>17</b> %	<b>17</b> %	<b>0</b> %	<b>0</b> %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	<a href="http://scholar.unand.ac.id">scholar.unand.ac.id</a> Internet Source	13%
2	<a href="http://www.scribd.com">www.scribd.com</a> Internet Source	4%

Exclude quotes  On  
Exclude bibliography  On

Exclude matches  < 3%