

UNIVERSITAS GUNADARMA
FAKULTAS TEKNOLOGI INDUSTRI



PENULISAN ILMIAH

**RANCANG BANGUN APLIKASI E-COMMERCE TOKO MAKANAN
“DAPURMBOKNDESO” BERBASIS WEB MENGGUNAKAN PHP
DENGAN FRAMEWORK CODEIGNITER**

| | |
|-------------------|---|
| Nama | : Willie Ryantama |
| NPM | : 56417206 |
| Jurusan | : Teknik Informatika |
| Pembimbing | : Dr. Cut Maisyarah Karyati, S.Kom., MM. |

**Diajukan Guna Melengkapi Sebagian Syarat Dalam Mencapai
Gelara Setara Sarjana Muda
Universitas Gunadarma
2021**

PERNYATAAN ORISINALITAS DAN PUBLIKASI

Saya yang bertanda tangan dibawah ini,

Nama : Willie Ryantama

NPM : 56417206

Judul PI : Rancang Bangun Aplikasi E-Commerce Toko Makanan
“DapurMbokNdeso” Berbasis Web Menggunakan PHP
Dengan Framework Codeigniter.

Tanggal Sidang :

Tanggal Lulus :

Menyatakan bahwa tulisan ini adalah merupakan hasil karya saya sendiri dan dapat dipublikasikan sepenuhnya oleh Universitas Gunadarma. Segala kutipan dalam bentuk apapun telah mengikuti kaidah, dan etika yang berlaku. Mengenai isi dan tulisan adalah merupakan tanggung jawab penulis, bukan Universitas Gunadarma.

Demikian pernyataan ini dibuat dengan sebenarnya dan dengan penuh kesadaran.

Jakarta, 29 Januari 2021

(Willie Ryantama)

LEMBAR PENGESAHAN

Judul PI : Rancang Bangun Aplikasi E-commerce Toko
Makanan “DapurMbokNdeso” Berbasis Web
Menggunakan PHP Dengan Framework CodeIgniter.
Nama : Willie Ryantama
NPM : 56417206
Tanggal Sidang :
Tanggal Lulus :

Menyetujui:

Pembimbing

Koordinator PI

(Dr.Cut Maisyarah Karyati, S.Kom., MM.) (Dr.Achmad Fahrurozi S.Si, M.Si)

Ketua Jurusan Teknik Informatika

(Dr. Lintang Yuniar Banowosari, S.Kom., M.Sc.)

ABSTRAK

Willie Ryantama, 56417206

Rancang Bangun Aplikasi E-Commerce Toko Makanan “DapurMbokNdeso” Berbasis Web Menggunakan PHP Dengan Framework CodeIgniter.

PI. Jurusan Teknik Informatika.

Fakultas Teknologi Industri. Universitas Gunadarma.

2021.

Kata Kunci: Toko Makanan, Website, Order, Pemesanan Online. Transaksi, Admin, User, CodeIgniter.

(xiii + 77 + Lampiran)

Perkembangan perekonomian masyarakat mempengaruhi budaya konsumsi di Indonesia, banyak masyarakat yang memanfaatkan berbagai cara untuk memperoleh kebutuhannya. Salah satunya adalah memanfaatkan perkembangan teknologi dalam memesan menu dalam sebuah tempat makan karena konsumen sering kali merasa malas untuk berpergian ke luar rumah hanya untuk membeli makanan. DapurMbokNdeso adalah sebuah toko makanan rumahan yang menjual makanan dan minuman khas desa hanya di daerah sekitaran saja. Dengan ini penulis membuat aplikasi pemesanan dan pembelian DapurMbokNdeso berbasis web menggunakan php dan mysql dengan framework codeigniter. Penelitian ini bertujuan untuk membantu konsumen untuk mendapatkan makanan dan minuman yang dijual di DapurMbokNdeso tanpa harus mendatangi tempatnya langsung. Kesimpulan yang didapat dari penelitian ini yaitu website pemesanan online ini menjadi daya tarik bagi konsumen, mudah diakses oleh konsumen, konsumen dapat melihat menu-menu yang tersedia dengan mudah, dan melakukan pemesanan dengan mudah dimana dan kapan saja. Serta memudahkan penjual dalam mengatur data produk dan data pesanan.

Daftar Pustaka (2011 – 2020)

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Tuhan YME atas berkat karunia-Nya sehingga penulis dapat menyelesaikan tugas penulisan ilmiah ini.

Penulisan ilmiah ini dimaksudkan untuk melengkapi salah satu syarat mencapai jenjang setara sarjana muda, Universitas Gunadarma. Penulis menyadari bahwa penulisan ilmiah ini masih jauh dari kesempurnaan untuk itu diharapkan dengan segala kerendahan hati kiranya dapat diberikan saran dan kritik yang bersifat membangun kepada penulis guna tercapainya karya yang lebih baik di masa depan nanti.

Di dalam menyusun laporan penulisan ilmiah ini, banyak sekali pihak-pihak yang membantu baik secara materil ataupun spiritual, serta saran-saran dari berbagai macam pihak, dan juga yang telah banyak memberikan pengarahan serta koreksi yang sangat bermanfaat dalam membimbing penulis sampai dengan terselesaikannya penulisan ilmiah ini.

Pada kesempatan ini juga, penulis tidak lupa untuk mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Ibu Prof. Dr. E. S. Margianti SE., MM., selaku rektor Universitas Gunadarma.
2. Bapak Prof. Dr. Ing. Adang Suhendra, Ssi. Skom, MSc., selaku Dekan Fakultas Teknologi Industri Universitas Gunadarma.
3. Ibu Dr. Lintang Yuniar Banowosari S.Kom., M.Sc., selaku Ketua Jurusan Teknik Informatika Universitas Gunadarma.
4. Bapak Dr. Achmad Fahrurrozi S.Si., M.Si., selaku koordinator Penulisan Ilmiah.
5. Ibu Dr. Cut Maisyarah Karyati S.Kom., MM., selaku dosen pembimbing yang telah meluangkan waktu dan pikirannya dalam bentuk bimbingan kepada penulis.

6. Bapak Suwandi Lie (Bapak), Ibu Supriyati (Ibu), serta saudara-saudara penulis yang telah memberikan dorongan, motivasi, doa dan menjadi tokoh inspirasi penulis sehingga dapat menyelesaikan penulisan ilmiah ini.
7. Semua teman-teman dan rekan-rekan seperjuangan Penulisan Ilmiah yang telah memberi dukungan dan semangat serta turut membantu dan memberikan inspirasi dan pendapat yang sangat bermanfaat bagi penulis.
8. Feraldi, Chintya, Joseph, Patrik dan Davi selaku sahabat dekat yang selalu memberikan motivasi, dukungan, dan arahan sehingga penulis dapat menyelesaikan penulisan ilmiah ini.

Semoga Tuhan YME melimpahkan berkat dan karunia-Nya kepada semua pihak atas segala bantuan dan bimbingannya.

Dengan segala kerendahan hati penulis sangat menyadari bahwa masih banyak kekurangan dalam penulisan ilmiah ini. Oleh karena itu penulis mohon maaf atas kekurangannya. Akhir kata, penulis mengharapkan semoga hasil dari penulisan ilmiah ini dapat bermanfaat bagi penulis khususnya dan pembaca pada umumnya.

Jakarta, 29 Januari 2021

(Willie Ryantama)

DAFTAR ISI

| | |
|---|------------|
| UNIVERSITAS GUNADARMA | i |
| PERNYATAAN ORISINALITAS DAN PUBLIKASI..... | ii |
| LEMBAR PENGESAHAN..... | iii |
| ABSTRAK..... | iv |
| KATA PENGANTAR..... | v |
| DAFTAR ISI | vii |
| DAFTAR GAMBAR..... | xi |
| DAFTAR TABEL | xii |
| 1 PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Rumusan Masalah | 6 |
| 1.3 Batasan Masalah | 6 |
| 1.4 Tujuan Penelitian | 7 |
| 1.5 Metode Penelitian..... | 7 |
| 1.6 Sistematika Penulisan..... | 8 |
| 2 LANDASAN TEORI..... | 9 |
| 2.1 Website | 9 |
| 2.1.1 Sejarah Website..... | 9 |
| 2.1.2 Unsur-Unsur Website | 9 |
| 2.1.3 Jenis-Jenis Website..... | 10 |
| 2.1.4 Manfaat Website | 11 |
| 2.1.5 Bahasa Pemrograman Web | 12 |
| 2.2 Pengertian <i>E-Commerce</i> | 13 |
| 2.2.1 Jenis <i>E-Commerce</i> | 13 |
| 2.2.2 Keuntungan <i>E-Commerce</i> | 15 |
| 2.2.3 Kerugian <i>E-Commerce</i> | 16 |

| | | |
|----------|---|-----------|
| 2.3 | Database..... | 16 |
| 2.3.1 | Jenjang Data..... | 17 |
| 2.4 | MySQL..... | 18 |
| 2.4.1 | Keuntungan MySQL | 18 |
| 2.4.2 | Tipe Data MySQL..... | 19 |
| 2.5 | PhpMyAdmin..... | 22 |
| 2.5.1 | PHP(<i>PersonalHomePage</i>) | 22 |
| 2.5.2 | Konsep Dasar PHP | 23 |
| 2.5.3 | Tipe Data PHP | 24 |
| 2.5.4 | Keunggulan PHP | 25 |
| 2.6 | Internet..... | 26 |
| 2.6.1 | Internet Commerce | 27 |
| 2.6.2 | Web Server..... | 27 |
| 2.7 | Hypertext Markup Language (HTML)..... | 28 |
| 2.8 | CSS(Cascading Style Sheet)..... | 29 |
| 2.9 | CodeIgniter | 29 |
| 2.9.1 | Fungsi CodeIgniter | 29 |
| 2.9.2 | Kelebihan CodeIgniter..... | 29 |
| 2.9.3 | Kekurangan CodeIgniter..... | 30 |
| 2.10 | Struktur Navigasi | 31 |
| 2.11 | MVC..... | 31 |
| 2.11.1 | Jenis-Jenis MVC | 31 |
| 2.12 | Visual Studio Code | 32 |
| 2.12.1 | Fitur-Fitur Visual Studio Code | 33 |
| 2.13 | Peneliti Terdahulu | 34 |
| 3 | PERANCANGAN DAN IMPLEMENTASI | 37 |
| 3.1 | Gambaran Umum..... | 37 |
| 3.2 | Identifikasi Kebutuhan | 38 |
| 3.2.1 | Identifikasi Kebutuhan Perangkat | 38 |

| | | |
|----------|---|----|
| 3.2.2 | Identifikasi Kebutuhan Aktor | 39 |
| 3.2.3 | Identifikasi Kebutuhan Fungsional | 39 |
| 3.2.3.1 | Kebutuhan Fungsional Admin | 40 |
| 3.2.3.2 | Kebutuhan Fungsional Pembeli..... | 40 |
| 3.3 | Perancangan | 41 |
| 3.3.1 | Perancangan Diagram Use Case | 41 |
| 3.3.1.1 | Diagram Use Case Admin..... | 41 |
| 3.3.1.2 | Diagram Use Case Pembeli..... | 42 |
| 3.3.2 | Perancangan Struktur Navigasi | 42 |
| 3.3.2.1 | Struktur Navigasi Admin | 42 |
| 3.3.2.2 | Struktur Navigasi Pembeli | 43 |
| 3.3.3 | Perancangan Interface..... | 44 |
| 3.3.3.1 | Perancangan Interface Halaman Utama Pembeli | 44 |
| 3.3.3.2 | Rancangan Interface Login | 45 |
| 3.3.3.3 | Rancangan Interface Registrasi..... | 46 |
| 3.3.3.4 | Rancangan Interface Daftar Menu..... | 46 |
| 3.3.3.5 | Rancangan Interface Keranjang Belanja..... | 47 |
| 3.3.3.6 | Rancangan Interface Detail Produk..... | 48 |
| 3.3.3.7 | Rancangan Interface Dashboard Admin | 49 |
| 3.3.3.8 | Rancangan Interface Data Barang Admin | 50 |
| 3.3.3.9 | Rancangan Interface Invoice Pemesanan..... | 50 |
| 3.3.3.10 | Rancangan Interface Detail Invoice | 51 |
| 3.3.4 | Perancangan Database | 52 |
| 3.4 | Implementasi | 54 |
| 3.4.1 | Tampilan Halaman Utama | 54 |
| 3.4.2 | Tampilan Login | 56 |
| 3.4.3 | Tampilan Registrasi..... | 56 |
| 3.4.4 | Tampilan Daftar Menu | 57 |
| 3.4.5 | Tampilan Detail Produk..... | 59 |

| | | |
|----------|----------------------------------|-----------|
| 3.4.6 | Tampilan Keranjang Belanja | 59 |
| 3.4.7 | Tampilan Pembayaran | 60 |
| 3.4.8 | Tampilan Proses Pembayaran | 60 |
| 3.4.9 | Tampilan Dashboard Admin | 61 |
| 3.4.10 | Tampilan Data Barang | 61 |
| 3.4.11 | Tampilan Invoice..... | 62 |
| 3.4.12 | Tampilan Detail Invoice | 62 |
| 3.4.13 | Implementasi Hosting..... | 63 |
| 3.5 | Uji Coba..... | 68 |
| 4 | PENUTUP | 73 |
| 4.1 | Kesimpulan | 73 |
| 4.2 | Saran..... | 73 |
| | DAFTAR PUSTAKA..... | 74 |
| | LAMPIRAN | 78 |

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 Jenjang Data | 17 |
| Gambar 3.1 Bagan Umum Metode Pembuatan Aplikasi | 37 |
| Gambar 3.2 Diagram Use Case Admin..... | 41 |
| Gambar 3.3 Diagram Use Case Pembeli..... | 42 |
| Gambar 3.4 Struktur Navigasi Admin | 43 |
| Gambar 3.5 Struktur Navigasi Pembeli | 44 |
| Gambar 3.6 Rancangan interface Halaman Utama Pembeli | 45 |
| Gambar 3.7 Rancangan Interface Login | 45 |
| Gambar 3.8 Rancangan Interface Halaman Registrasi | 46 |
| Gambar 3.9 Rancangan Interface Halaman Daftar Menu | 47 |
| Gambar 3.10 Rancangan Interface Halaman Keranjang Belanja | 48 |
| Gambar 3.11 Rancangan Interface Halaman Detail Produk | 49 |
| Gambar 3.12 Rancangan Interface Halaman Dashboard Admin | 49 |
| Gambar 3.13 Rancangan Interface Data Barang Admin..... | 50 |
| Gambar 3.14 Rancangan Interface Invoice Pemesanan..... | 51 |
| Gambar 3.15 Rancangan Interface Detail Invoice..... | 51 |
| Gambar 3.16 Tampilan Halaman Utama | 55 |
| Gambar 3.17 Tampilan Halaman Utama | 55 |
| Gambar 3.18 Tampilan Login | 56 |
| Gambar 3.19 Tampilan Registrasi Akun..... | 57 |
| Gambar 3.20 Tampilan Daftar Menu Makanan | 57 |
| Gambar 3.21 Tampilan Daftar Menu Lalapan | 58 |
| Gambar 3.22 Tampilan Daftar Menu Minuman | 58 |
| Gambar 3.23 Tampilan Detail Produk | 59 |
| Gambar 3.24 Tampilan Keranjang Belanja..... | 59 |
| Gambar 3.25 Tampilan Halaman Pembayaran..... | 60 |

| | |
|---|----|
| Gambar 3.26 Tampilan Halaman Proses Pembayaran..... | 60 |
| Gambar 3.27 Tampilan Halaman Dashboard Admin | 61 |
| Gambar 3.28 Tampilan Halaman Data Barang | 61 |
| Gambar 3.29 Tampilan Halaman Invoice | 62 |
| Gambar 3.30 Tampilan Halaman Detail Invoice..... | 62 |
| Gambar 3.31 Tampilan Utama 000webhost..... | 63 |
| Gambar 3.32 Tampilan Kelola Website..... | 64 |
| Gambar 3.33 Tombol Untuk Membuat Situs Baru..... | 64 |
| Gambar 3.34 Memasukkan Nama Website dan Password..... | 64 |
| Gambar 3.35 Website Yang Sudah Dibuat | 65 |
| Gambar 3.36 Pop-up Upload Files | 65 |
| Gambar 3.37 Tampilan File Yang Terupload | 66 |
| Gambar 3.38 Database Baru..... | 66 |
| Gambar 3.39 Konfigurasi Database..... | 67 |

DAFTAR TABEL

| | |
|--|----|
| Tabel 2.1 Tipe Data Numerik | 20 |
| Tabel 2.2 Tipe Data Waktu dan Tanggal | 21 |
| Tabel 2.3 Tipe Data String | 21 |
| Tabel 2.4 Tipe Data Lain | 22 |
| Tabel 2.5 Tabel Penelitian Terdahulu | 34 |
| Tabel 3.1 Kebutuhan Aktor Yang Terlibat..... | 39 |
| Tabel 3.2 Kebutuhan Fungsional Admin | 40 |
| Tabel 3.3 Kebutuhan Fungsional Pembeli | 40 |
| Tabel 3.4 Struktur Tabel Makanan | 52 |
| Tabel 3.5 Struktur Tabel tb_invoice | 53 |
| Tabel 3.6 Struktur Tabel tb_pesanan | 53 |
| Tabel 3.7 Struktur Tabel tb_user | 54 |
| Tabel 3.8 Tabel Pengujian Form User | 69 |
| Tabel 3.9 Tabel Pengujian Form Admin..... | 71 |

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi di Indonesia sangat cepat, internet menjadi salah satu teknologi yang berkembang pesat di Indonesia terutama dalam bidang teknologi informasi dan komunikasi, adanya internet memudahkan berbagai aktivitas sehari-hari manusia, hal ini disebabkan karena dengan menggunakan internet aktivitas manusia tidak lagi terhalang oleh batasan ruang dan waktu, salah satunya adalah kita dapat melakukan transaksi jual-beli tanpa harus bertatap muka.

Pemanfaatan teknologi informasi harus diimbangi dengan implementasi sistem informasi. Teknologi informasi dan sistem informasi merupakan dua bidang yang saling berdampingan dalam penyampaian informasi bagi suatu organisasi. Pemanfaatan teknologi informasi dan sistem informasi pada suatu organisasi dapat membantu organisasi untuk mencapai tujuan dengan maksimal. Hal ini terbukti dengan semakin berkembang dan mudahnya peran teknologi informasi dan sistem informasi di kalangan organisasi.

Hampir seluruh organisasi saat ini memakai teknologi informasi dan komunikasi terutama organisasi perdagangan, dengan menggunakan teknologi ini organisasi dagang tersebut dapat memaksimalkan perdagangannya dengan cara melakukan promosi dan transaksi produk secara digital.

Saat ini banyak masalah perdagangan dapat diselesaikan dengan menggunakan teknologi, salah satu teknologi dan sistem informasi yang banyak digunakan pada strategi bisnis saat ini adalah perdagangan elektronik (*electronic commerce* atau *e-commerce*), perdagangan elektronik atau *e-commerce* adalah penyebaran, pembelian, pemasaran barang, dan jasa melalui sistem elektronik

seperti internet atau televisi, *world wide web* (www), atau jaringan komputer lainnya.

E-commerce dapat melibatkan transfer data elektronik, pertukaran data elektronik, sistem manajemen inventory otomatis, dan sistem pengumpulan data otomatis, industri teknologi informasi melihat kegiatan *e-commerce* ini sebagai aplikasi dan penerapan dari bisnis elektronik (*e-business*) yang berkaitan dengan transaksi komersial, seperti transfer dana secara elektronik, atau pemasaran online (*online marketing*), pemrosesan transaksi online (*online transaction marketing*), pertukaran data elektronik (*electronic data interchange*), dan lain-lain.

E-commerce pertama kali diperkenalkan pada tahun 1994 pada saat pertama kali banner-elektronik dipakai untuk tujuan promosi dan periklanan di suatu halaman-*web* (*website*). Menurut Riset Forrester, perdagangan menghasilkan penjualan seharga AS\$12,2 miliar pada tahun 2003. Menurut laporan yang lain pada bulan oktober 2006 yang lalu, pendapat ritel online yang bersifat non-travel di Amerika Serikat diramalkan akan mencapai seperempat trilyun dolar US pada tahun 2011. *E-commerce* juga mempunyai bermacam bentuk atau model berikut ini adalah bentuk-bentuk atau model dari *e-commerce* yang terdapat di Indonesia.

Jenis yang pertama adalah *e-commerce* berbentuk iklan baris, merupakan salah satu bentuk *e-commerce* yang tergolong sederhana, bisa dianggap sebagai evolusi dari iklan baris yang biasanya ditemui di koran-koran ke dalam dunia online. Penjual yang menggunakan *social media* atau forum untuk beriklan, biasanya tidak bisa langsung menyelesaikan transaksi pada *website* yang bersangkutan. Namun penjual dan pembeli harus berkomunikasi secara langsung untuk bertransaksi. Contoh *e-commerce* berbentuk iklan baris di Indonesia: OLX.co.id, Berniaga.com, dan Forum Jual Beli (FJB)-Kaskus.

Jenis yang kedua *e-commerce* berbentuk retail, merupakan jenis *e-commerce* yang di mana semua proses jual-beli dilakukan melalui sistem yang

sudah diterapkan oleh situs retail yang bersangkutan. Oleh karena itu, kegiatan jual-beli di retail relatif aman, namun biasanya pilihan produk yang tersedia tidak terlalu banyak, atau hanya fokus ke satu-dua kategori produk saja. Contoh *e-commerce* berebentuk retail di Indonesia: BerryBenzka, Zalora, dan Lazada.

Setelah mengetahui apa itu perdagangan elektronik atau *e-commerce*, berikutnya adalah salah satu jenis *e-commerce* yang akan digunakan penulis yaitu *Business to Consumer* (B2C), jenis *e-commerce* b2c ini merupakan salah satu jenis yang populer pada saat ini, keunggulan utama yang didapatkan dengan menggunakan jenis *e-commerce* ini adalah penjual dapat langsung menawarkan, mempromosikan dan menjual produk-produknya langsung ke konsumen. Keunggulan itu berkemungkinan besar dapat meningkatkan penjualan produk, maka dari itu jenis *e-commerce* ini menjadi salah satu jenis yang populer dipakai untuk para pengusaha.

Berikut adalah salah satu contoh penerapan *e-commerce* menggunakan jenis b2c, berdasarkan [Adi Nurdian Candra, 2017] tentang Penerapan *E-commerce* Berbasis Business to Consumer Untuk Meningkatkan Penjualan Produk Makanan Ringan Khas Pringsewu. Dalam membangun sistem untuk aplikasi tersebut, penulis menggunakan metode *waterfall*, metode *waterfall* melakukan pendekatan secara sistematis dan bertahap mulai dari level kebutuhan sistem, kemudian menuju ketahap analisis, desain, *pengcoding-an*, eksperimen, *maintenance*. Lalu dilanjutkan dengan membuat dfd dan erd dan kemudian melakukan perancangan *web* dengan teknik pemrograman terstruktur menggunakan bahasa pemrograman PHP dan basis data MySQL, setelah selesai merancang dan melakukan implementasi, hasil yang didapat oleh penulis adalah dengan adanya *website* pada Toko Jajanan Pringsewu, dapat memperluas pemasaran produk-produknya, dan dapat diakses dengan cepat, dan juga dengan adanya *website* ini mempermudah konsumen dalam memesan, memilih produk, transaksi dan lainnya.

Pada [Mardhiah Fadhli, 2017] tentang Penerapan metode UCD (*User Centered Design*) pada *e-commerce* Putri Intan Shop Berbasis Web, metode UCD ini memanfaatkan pendapat pengguna, serta pola dan tingkah laku pengguna. Tujuan penggunaan metode User Centered Design ini adalah untuk mengatasi masalah ketidakmampuan pengguna dalam menggunakan sistem, dan diharapkan pengguna mampu mengetahui fungsi sistem hanya dalam sekali pakai. Metode User Centered Design melibatkan calon pengguna pada tahap awal pengembangan sehingga calon pengguna dapat memberikan masukan mengenai antarmuka sistem *e-commerce*. Pada [Amelia, 2019] tentang *Analysis and Design of E-commerce Information Systems Using the User Centered Method*, UCD memiliki tahapan yang harus dilakukan yaitu: merancang pertanyaan untuk mendapatkan masukan dan solusi dari pengguna, mengerti dan menentukan konteks pengguna, menentukan kebutuhan pengguna, mengimplementasikan solusi dari pengguna, evaluasi hasil dari desain berdasarkan kebutuhan pengguna.

Salah satu tujuan para pengusaha dengan membangun *web e-commerce* adalah untuk meningkatkan pemasaran, seperti pada [Adzinta Winerawan Tito, 2017] tentang pemanfaatan *web e-commerce* untuk meningkatkan strategi pemasaran, dalam penulisannya terdapat suatu masalah, masalah itu adalah kurangnya media pemasaran hasil produksi industri tersebut. Penulis kemudian merancang dan membangun sebuah *web e-commerce* dengan model pengembangan sistem *waterfall* dan aplikasi ini dirancang untuk sistem pemesanan dan pembelian secara online, dan konsumen akan diberikan pilihan sistem pembayaran secara *cash on delivery* (COD), dan sistem transaksi online menggunakan *virtual bank*.

Walaupun *e-commerce* merupakan suatu teknologi yang canggih dan mempermudah perbisnisan namun tentunya tidak luput dari masalah, menurut supplychainindonesia.com tentang tantangan *e-commerce* indonesia, tantangan dan hambatan yang dihadapi ketika menggunakan *web e-commerce* sebagai berikut, yang pertama adalah pembayaran bukan dari rekening bank, alat pembayaran di setiap transaksi *e-commerce* pada umumnya menggunakan metode pembayaran

secara transfer virtual dengan kartu debit atau kredit namun masalahnya di Indonesia sendiri masih banyak masyarakat yang belum memiliki rekening bank, yang kedua adalah faktor koneksi internet, tanpa koneksi internet tentu kita tidak dapat mengakses maupun melakukan pemesanan lewat *web e-commerce*, hal ini menjadi hambatan karena di Indonesia terutama di beberapa wilayah bagian pelosok koneksi internet masih menjadi masalah dan bahkan tidak memiliki akses internet sama sekali, yang ketiga adalah logistik pengiriman, banyak sekali konsumen yang merasa dirugikan karena masalah pengiriman barang pesanan, seperti keterlambatan pengiriman, barang yang dikirim mengalami kerusakan, hal tersebut terjadi dikarenakan standar logistik yang mencakup waktu pengiriman barang dan pengiriman barang belum memadai.

Aplikasi *web e-commerce* yang akan penulis buat adalah *website* untuk toko penjualan makanan berbasis online bernama “DapurMbokNdeso”, toko ini merupakan suatu badan usaha milik swasta atau perorangan. Toko ini terletak di Jalan Kelapa Dua Wetan Gang Hati Rongga, Ciracas, Jakarta Timur. DapurMbokNdeso sudah berdiri sejak tahun 2019. Badan usaha ini menjual berbagai macam makanan dan minuman, untuk makanan “DapurMbokNdeso” menjual makanan olahan rumah seperti ayam bakar, ayam goreng, garang asem, ikan pari sambal, dan olahan *frozen food* buatan rumah. Untuk minuman DapurMbokNdeso menjual jamu dengan berbagai varian rasa yaitu beras kencur, jahe, mpon, dan kunyit asem.

Untuk merancang web ini penulis menggunakan metode *waterfall*, menggunakan bahasa pemrograman PHP dengan framework Codeigniter dan untuk basis datanya menggunakan MySQL, dan penulis melakukan eksperimen dengan cara membuat pengguna (*user*) mencoba website tersebut, pengguna akan memberikan umpan balik tentang bagaimana pengalaman mereka menggunakan website tersebut, setelah itu akan dilakukan *maintenance* untuk memperbaiki apa yang dibutuhkan untuk website tersebut melalui umpan balik pengguna. Dengan adanya penulisan ilmiah ini diharapkan dapat meningkatkan pemasaran dan

penjualan toko makanan DapurMbokNdeso, dan memudahkan proses pemesanan dan penjualan barang baik dari sisi toko maupun dari sisi pembeli.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan sebelumnya, perumusan masalah pada penulisan ini adalah bagaimana membuat aplikasi pemesanan online berbasis web untuk mempercepat proses pelayanan dan meningkatkan penjualan atau pemasaran di toko makanan “DapurMbokNdeso”.

1.3 Batasan Masalah

Berdasarkan latar belakang di atas, maka diperlukan batasan masalah yang akan diteliti agar memperjelas lingkup penelitian ini. Berikut adalah batasan masalah tersebut :

1. Perancangan *web* berbasis *e-commerce* menggunakan PHP dengan framework Codeigniter versi 3.1.11 dan MySQL.
2. Menggunakan metode *waterfall*.
3. Aplikasi ini dibuat dengan menggunakan Visual Studio Code versi 1.43.1.
4. Target pengguna aplikasi ini adalah admin dan masyarakat umum.
5. Aplikasi ini melakukan pembayaran dengan cara transfer bank dan *Cash On Delivery* (COD).
6. Aplikasi ini menggunakan jenis *e-commerce business to consumer (b2c)*.
7. Aplikasi ini hanya untuk penjualan dan pemesanan barang.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah merancang dan membangun aplikasi e-commerce berbasis web pada toko makanan “DapurMbokNdeso” sehingga menghasilkan toko online yang dapat mempermudah proses pemesanan untuk pembeli, pendataan barang, dan mengelola data pesanan.

1.5 Metode Penelitian

Metode yang digunakan untuk perancangan dan pengembangan aplikasi pada penelitian ini adalah dengan model proses *waterfall*. Tahap penelitian yang dilakukan adalah:

1. Pengumpulan Data Kebutuhan

Pada tahap ini penulis mengumpulkan data yang terkait untuk proses perancangan dan pembangunan aplikasi pada penelitian ini diantaranya adalah data tentang produk-produk yang dijual pada toko makanan DapurMbokNdeso, perangkat yang digunakan, dan aktor yang terlibat.

2. Perancangan

Pada tahap ini dilakukan perancangan yang terdiri dari perancangan diagram *use case*, perancangan struktur navigasi, perancangan antar muka (*interface*) yang terdiri dari halaman utama dan halaman tampilan, dan perancangan antar muka pengguna.

3. Implementasi dan pengkodean

Pada tahap ini penulis melakukan translasi kedalam bahasa pemrograman menggunakan rancangan yang telah dibuat. Sehingga hasil dari tahap ini adalah program komputer yang sesuai dengan tujuan.

4. Pengujian

Pada tahap ini penulis melakukan pengujian aplikasi yang telah dibuat menggunakan *web browser* dan dilakukan juga uji coba aplikasi langsung kepada pengguna.

1.6 Sistematika Penulisan

Sistematika penulisan ini dibuat bertujuan untuk mempermudah dalam penyusunan penulisan ilmiah dan dapat memberikan gambaran secara jelas tentang permasalahan yang akan dibahas. Pada penulisan ini terdiri dari 4 (empat) bab yang menjelaskan berbagai persoalan mengenai alat yang dibuat. Untuk memahami isi dari masing-masing bab adalah sebagai berikut :

BAB 1 PENDAHULUAN

Bab ini berisikan mengenai latar belakang masalah, batasan masalah, tujuan penelitian, metode penelitian dan sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini berisikan uraian penjelasan tentang teori yang diperlukan dalam pembuatan penulisan ilmiah ini. Teori yang diperlukan tersebut meliputi definisi tentang *e-commerce*, metode pengembangan waterfall, penjelasan tentang tools yang digunakan dalam penulisan ini seperti: bahasa pemrograman yang digunakan yaitu PHP, basis data yang digunakan yaitu MySQL, framework yang digunakan yaitu Codeigniter, webserver yang digunakan yaitu xampp.

BAB 3 PERANCANGAN DAN IMPLEMENTASI

Bab ini menjelaskan tentang gambaran umum aplikasi, pengumpulan data, analisis sistem, perancangan aplikasi, pembuatan aplikasi, dan uji coba program pada komputer menggunakan web browser dan server lokal.

BAB 4 PENUTUP

Bab ini menjelaskan tentang pemberian kesimpulan dan saran untuk pengembangan dan penyempurnaan penulisan ilmiah ini selanjutnya.

BAB 2

LANDASAN TEORI

2.1 Website

Website adalah sebuah kumpulan halaman pada suatu domain di internet yang dibuat dengan tujuan tertentu dan berhubungan serta dapat diakses secara luas melalui halaman depan (*home page*) menggunakan sebuah browser menggunakan URL (*Uniform Resource Locator*) website [Waryanto, 2018].

2.1.1 Sejarah Website

Website pertama kali dibuat oleh Tim Berners-Lee pada akhir 1980an dan baru resmi online pada tahun 1991. Tujuan awal Tim Berners-Lee membuat sebuah website adalah supaya lebih memudahkan para peneliti di tempatnya bekerja ketika akan bertukar atau melakukan perubahan informasi, website dipublikasikan setelah adanya pengumuman dari CERN pada tanggal 30 april 1993. CERN menyatakan bahwa website dapat digunakan secara gratis oleh semua orang. Pada saat ini pengertian website sudah masuk ke dalam ranah publik karena sudah bisa digunakan oleh semua orang dimanapun dan kapanpun [Mustofa, 2018].

2.1.2 Unsur-Unsur Website

Terdapat 3 unsur yang sangat vital pada website, Tanpa adanya semua unsur ini, website anda tidak akan pernah ditemukan dan diakses oleh pengguna di internet. Ketiga unsur yang dimaksud adalah: [Waryanto, 2018]

1. **Domain.** Jika website diibaratkan sebagai produk, maka domain adalah merk. Penggunaan domain yang menarik akan membuat orang tertarik untuk memasuki suatu website. Dengan pemilihan nama domain yang unik juga membuat orang mudah mengingatnya untuk nantinya dikunjungi kembali.

2. **Hosting.** Tidak kalah pentingnya dengan domain, hosting memiliki peran untuk menyimpan semua database (script, gambar, video, teks dan lain sebagainya) yang diperlukan untuk membentuk suatu website. Banyak sekali penyedia jasa hosting.
3. **Konten.** Tanpa adanya konten pada website, maka website bisa dikatakan tidak memiliki tujuan yang jelas. Konten pada website dapat berupa teks, gambar atau video. Jika dilihat dari konten yang disuguhkan, terdapat beberapa macam website. Misalnya saja, sosial media, website berita, website jual beli atau website yang berisi konten yang berdasarkan minat, bakat serta hobi.

2.1.3 Jenis-Jenis Website

Website berdasarkan fungsinya secara spesifik maka bisa dibagi ke dalam beberapa jenis, yakni antara lain: [Anonim, 2018]

1. **Company Profile**, adalah website yang bersifat statis, dan biasanya dibuat oleh perusahaan atau organisasi untuk menampilkan informasi yang terkait dengan perusahaan, seperti nama perusahaan, produk perusahaan, visi misi, dan halaman kontak.
2. **Government (pemerintahan)**, website pemerintahan memang resmi milik pemerintah nasional, pemerintah lokal, dan departemen pemerintah. Di dalam website ini memberikan informasi mengenai peraturan-peraturan mengenai pemerintahan, serta informasi berita yang terkait.
3. **E-commerce**, *e-commerce* merupakan website yang menyajikan fitur untuk melakukan kegiatan jual beli suatu produk barang atau jasa melalui portal internet. Lazimnya website ini menawarkan fitur menarik dan bisa menggantikan fungsi pada toko offline, misalnya tampilan produk, pemesanan barang, pengecekan produk, dan melakukan transaksi online.
4. **Archive**, website *archive* digunakan untuk menyimpan konten-konten elektronik yang memungkinkan terancam hilang. Dengan begitu, akan

bermanfaat jika portal website disadap orang tak bertanggung jawab, karena bisa dikembalikan seperti semula.

5. **Portal berita dan informasi**, Website berita dan informasi termasuk situs yang membuat berita atau artikel untuk dibaca para pengunjung. Biasanya berisi opini, dan artikel seputar politik, teknologi, dan lain-lain.
6. **Blog**, kini banyak para blogger memanfaatkan media blog gratisan untuk menuliskan ide dan ekspresi dengan berbagai konten. Bahkan beberapa blogger profesional mendapatkan bayaran saat menulis topik tertentu.

2.1.4 Manfaat Website

Manfaat website tergantung dari jenis website yang dibuat, namun secara garis besar manfaat website pada kehidupan sehari-hari adalah: [Situseo, 2017]

1. **Memberikan informasi secara *Real-Time***. Informasi merupakan sebuah hal yang sangat penting demi pemahaman dan perkembangan pengetahuan umat manusia. Dahulu sebelum berkembangnya website seperti sekarang ini, penyampaian berbagai informasi yang dibatasi jarak dilakukan secara manual, seperti menggunakan tulisan dan mengirimkannya melalui kantor pos. Dengan adanya perkembangan teknologi jaringan internet saat ini, informasi apapun bisa didapatkan hanya hitungan detik saja.
2. **Memberikan pelayanan**. Manfaat website yang selanjutnya adalah memberikan pelayanan kepada publik atau konsumen. Penggunaan website jenis ini diperuntukkan kepada perusahaan yang menyediakan barang atau jasa. Sehingga dalam kesehariannya, website ini beroperasi guna menangani berbagai keluhan, pertanyaan, dan aduan terkait produk atau jasa yang dijual perusahaan.
3. **Informasi pekerjaan**. Terkait dengan dunia pekerjaan, saat ini: menurut survey Badan Pusat Statistik. Angka pengangguran di Indonesia sampai tahun 2019 mencapai 7 (tujuh) juta jiwa. Dengan adanya sebuah situs website yang bisa menampung informasi penyedia pekerjaan, sangatlah bermanfaat untuk para pencari kerja. Jenis website ini banyak digunakan

oleh para perusahaan yang mencari karyawan. Dengan cara memposting lowongan yang dibutuhkan untuk mengisi job tertentu pada perusahaannya.

4. **Sarana promosi dan pengenalan profil.** Fungsi website atau manfaat website yang selanjutnya adalah sebagai sarana media promosi, pihak yang banyak tertolong dari manfaat website yang satu ini adalah para perusahaan ataupun instansi tertentu. Memang sudah menjadi kewajiban setiap perusahaan untuk memasarkan atau mempromosikan produk dan jasanya.
5. **Forum jual-beli *online*.** Jual beli *online* adalah forum dimana aktifitas penjualan dan pembelian terjadi secara *online* dan *real-time*. Kelebihannya adalah tanpa adanya batasan tempat dan batasan daya tampung transaksi jual-beli yang terjadi di dalamnya. Para penjual menawarkan berbagai barang dan jasa, akan bertemu dengan para pembeli dalam satu media tempat yaitu situs website.
6. **Sarana hiburan dan *entertainment*.** Manfaat website selanjutnya bisa menjadi sarana hiburan kepada para pengguna internet. Konten dari jenis website ini adalah sesuatu yang bersifat menghibur kepada pengunjung, umumnya website sebagai sarana hiburan bisa dilihat dari adanya tontonan yang berbentuk *channel video*. Para pengunjung bisa menonton atau mengunduh berbagai hiburan yang disediakan oleh jenis website ini.

2.1.5 Bahasa Pemrograman Web

Bahasa pemrograman web terdiri dari beberapa unsur bahasa. Setidaknya terdapat 5 bahasa utama yang biasa digunakan dalam membuat website dinamis dimana masing-masing memiliki perannya sendiri-sendiri, yaitu sebagai berikut: [Abdullah, 2018]

- A. HTML, berperan sebagai pembentuk struktur halaman website yang menempatkan setiap elemen website sesuai layout yang diinginkan.
- B. CSS, berperan sebagai pembentuk desain website dengan mengatur setiap elemen HTML, agar tampil menarik pada browser.

- C. PHP, berperan sebagai pemroses data pada sisi server sesuai yang diminta oleh client menjadi informasi yang siap ditampilkan, juga sebagai penghubung aplikasi web dengan database.
- D. SQL, berperan sebagai bahasa yang mengatur transaksi data antara aplikasi dengan database sebagai tempat penyimpanan data. Database yang biasa digunakan diantaranya MySQL, Oracle, SQL Server.
- E. JavaScript, berperan sebagai bahasa yang memproses data pada sisi client serta dapat memanipulasi HTML dan CSS secara dinamis.

2.2 Pengertian *E-commerce*

Electronic Commerce (e-commerce) adalah proses pembelian, penjualan atau pertukaran produk, jasa dan informasi melalui jaringan komputer. *E-commerce* merupakan bagian dari *e-business*, di mana cakupan *e-business* lebih luas, tidak hanya perniagaan tetapi mencakup juga pengkolaborasian mitra bisnis, pelayanan nasabah, lowongan pekerjaan dll. Selain teknologi jaringan www, *e-commerce* juga memerlukan teknologi basis data atau pangkalan data (database), e-surat atau surat elektronik (e-mail), dan bentuk teknologi non komputer yang lain seperti halnya sistem pengiriman barang, dan alat pembayaran untuk *e-commerce* ini. [Sugeng S., 2018]

2.2.1 Jenis *E-commerce*

E-commerce dapat dibagi menjadi beberapa jenis yang memiliki karakteristik berbeda-beda yaitu: [FasaPay Administrator, 2019]

1. Business-to-Business (B2B)

B2B *e-commerce* meliputi semua transaksi elektronik barang atau jasa yang dilakukan antar perusahaan. Produsen dan pedagang jasa yang dilakukan antar perusahaan. Produsen dan pedagang tradisional biasanya menggunakan jenis *e-commerce* ini.

2. *Business-to-Consumer (B2C)*

B2C adalah jenis *e-commerce* antara perusahaan dan konsumen akhir. Hal ini sesuai dengan bagian ritel dari *e-commerce* yang biasa dioperasikan oleh perdagangan ritel tradisional. Jenis ini bisa lebih mudah dan dinamis, namun juga lebih menyebar secara tak merata atau bahkan bisa terhenti.

3. *Consumer-to-Consumer (C2C)*

C2C merupakan jenis *e-commerce* yang meliputi semua transaksi elektronik barang atau jasa antar konsumen. Umumnya transaksi ini dilakukan melalui pihak ketiga yang menyediakan platform online untuk melakukan transaksi tersebut.

4. *Consumer-to-Business (C2B)*

C2B adalah jenis *e-commerce* dengan pembalikan utuh dari transaksi pertukaran atau jual beli barang secara tradisional. Jenis *e-commerce* ini sangat umum dalam proyek dengan dasar multi sumber daya. Sekelompok besar individu menyediakan layanan jasa atau produk mereka bagi perusahaan yang mencari jasa atau produk tersebut.

5. *Business-to-Administration (B2A)*

B2A adalah jenis *e-commerce* yang mencakup semua transaksi yang dilakukan secara online antara perusahaan dan administrasi publik. Jenis *e-commerce* ini melibatkan banyak layanan, khususnya di bidang-bidang seperti fiskal, jaminan sosial, ketenagakerjaan, dokumen hukum dan register, dan lainnya. Jenis *e-commerce* ini telah meningkatkan dalam beberapa tahun terakhir dengan investasi yang dibuat melalui *e-government* atau pihak pemerintah.

6. *Online-to-Offline (O2O)*

O2O adalah jenis *e-commerce* yang menarik pelanggan dari saluran online untuk toko fisik. O2O mengidentifikasi pelanggan di bidang online seperti email dan iklan internet, kemudian menggunakan berbagai alat dan pendekatan untuk menarik pelanggan agar meninggalkan lingkup online. Contohnya, sebuah pusat kebugaran tidak akan bisa

didirikan di ruang tamu rumah Anda, namun dengan menggunakan layanan O2O yang disediakan perusahaan seperti Groupon Inc, pusat kebugaran tersebut bisa menyalurkan bisnis *offline*-nya menjadi *online*.

2.2.2 Keuntungan *E-commerce*

Ada beberapa keuntungan *e-commerce* yaitu : [Aco, Ambo dan Hutami, 2017]

- a. Bagi Perusahaan, memperpendek jarak, perluasan pasar, perluasan jaringan mitra bisnis dan *efisiensi*, dengan kata lain mempercepat pelayanan ke pelanggan, dan pelayanan lebih *responsif*, serta mengurangi biaya-biaya yang berhubungan dengan kertas, seperti biasa pos surat, pencetakan, *report*, dan sebagainya sehingga dapat meningkatkan pendapatan.
- b. Bagi Konsumen, efektif, aman secara fisik dan fleksibel.
- c. Bagi Masyarakat Umum, mengurangi polusi dan pencemaran lingkungan, membuka peluang kerja baru, menguntungkan dunia akademis, meningkatkan kualitas SDM.
- d. Mempersingkat rantai distribusi
- e. Pembayaran lebih mudah
- f. Brand lebih dekat dengan konsumen
- g. Belanja kapan saja

2.2.3 Kerugian E-commerce

Ada beberapa kerugian *e-commerce* yaitu : [Aco, Ambo dan Hutami, 2017]

- a. Meningkatkan Individualisme, pada perdagangan elektronik seseorang dapat bertransaksi dan mendapatkan barang/jasa yang diperlukan tanpa bertemu dengan siapapun.
- b. Terkadang Menimbulkan Kekecewaan, apa yang dilihat dilayar monitor komputer kadang berbeda dengan apa yang dilihat secara kasat mata.
- c. Ketergantungan yang sangat kuat pada teknologi informasi dan komunikasi.
- d. “Hilangnya” privasi, cakupan wilayah, serta identitas dan perekonomian Negara.
- e. Budaya pasar yang menolak perdagangan elektronik (pelanggan tidak bisa menyentuh atau mencoba produk)

2.3 Database

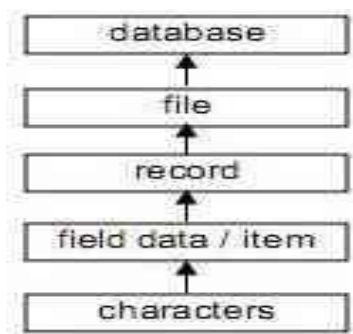
Database adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi

Database adalah sekumpulan tabel-tabel yang saling berelasi, relasi tersebut bisa ditunjukkan dengan kunci dari tiap tabel yang ada. Satu database menunjukkan satu ingkup perusahaan atau instansi.[Asrianda Urva, Gellysa, 2008]

Database juga merupakan kumpulan data yang umumnya menggambarkan aktifitas-aktifitas dan pelakunya dalam suatu organisasi. Sistem database merupakan sistem komputer yang digunakan untuk menyimpan dan mengelola data tersebut. Hampir semua aplikasi memakai database sebagai tulang punggungnya. Database diperlukan karena berbagai macam alasan, diantaranya:

1. Satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi.
2. Menentukan kualitas informasi akurat, tepat pada waktunya dan relevan. Informasi dapat dikatakan bernilai apabila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.
3. Mengurangi duplikasi data (*data redundancy*).
4. Hubungan data dapat ditingkatkan (*data reliability*)
5. Mengurangi pemborosan tempat penyimpanan eksternal.

2.3.1 Jenjang Data



Gambar 2.1 Jenjang Data

Adapun penjelasan dari tingkatan tersebut adalah sebagai berikut:
[Riyanto, 2005]

1. Database
Merupakan kumpulan dari file/table membentuk database.
2. File
Merupakan kumpulan dari record-record yang menggambarkan satu kesatuan data yang sejenis. Misalnya file mata pelajaran berisi data tentang semua mata pelajaran yang ada.
3. Record
Merupakan kumpulan dari field membentuk suatu record. Record menggambarkan satu unit data individu yang tertentu. Kumpulan

dari record membentuk suatu file. Misalnya file personalia, tiap-tiap record dapat mewakili data tiap-tiap karyawan.

4. Field

Mempresentasikan suatu atribut dari record yang menunjukkan suatu item dari data, seperti misalnya nama, alamat dan lain sebagainya.

5. Characters

Merupakan bagian data yang terkecil, dapat berupa karkater numerik, huruf ataupun karakter-karakter khusus (*special characters*) yang membentuk suatu item data/field.

2.4 MySQL

MySQL adalah salah satu database server yang cukup dikenal saat ini. MySQL keluaran T.c.X. data Consult AB, sebuah perusahaan IT swedia, yang menawarkan berbagai keunggulan dibandingkan database *server* lainnya, yaitu: [

1. Mampu menangani jutaan user dalam waktu yang bersamaan.
2. Mampu menampung lebih dari 50.000.000 record.
3. Sangat cepat dalam mengeksekusi perintah.

Selain itu MySQL juga menyediakan dukungan *open source*. Setiap pengguna MySQL diizinkan untuk mengubah source untuk keperluan pengembangan atau menyelaraskan spesifikasi database sesuai kebutuhan.

2.4.1 Keuntungan dari MySQL

MySQL adalah suatu database populer dengan pengembang web (*web devloper*). Kecepatan dan ukuran yang kecil membuatnya ideal untuk web site. Ditambah lagi dengan fakta bahwa MySQL adalah open source. Adapun keuntungan MySQL adalah: [Simamarta, J. 2006]

1. Cepat. Tujuan utama dari pengembangan MySQL adalah kecepatan, sebagai konsekuensi software yang dirancang dari awal untuk kecepatan.
2. Tidak mahal. MySQL adalah Cuma-Cuma dibawah lisensi GPL open source, sementara pembiayaan untuk lisensi komersialnya sangatlah pantas.
3. Mudah digunakan. Anda dapat membangun dan berinteraksi dengan database MySQL hanya dengan menggunakan sedikit pernyataan sederhana di dalam bahasa SQL, yang menjadi bahasa standar untuk komunikasi dengan RDBMS.
4. Dapat berjalan pada beberapa sistem operasi. MySQL dapat berjalan pada sistem operasi beragam, seperti Windows, Linux, MacOS.
5. Dukungan teknis secara luas tersedia.
6. Aman. MySQL adalah sistem otorisasi fleksibel yang mengijinkan beberapa atau semua privilege database untuk pengguna khusus atau kelompok pengguna.
7. Mendukung database yang besar. MySQL menangani database sampai 50 juta baris atau lebih.
8. Customizeable. Lisensi open source sehingga mengizinkan para pemrogram untuk memodifikasi software PHP.

2.4.2 Tipe Data MySQL

Dalam pembuatan tabel pada database hendaknya setiap kolom dibuat dengan tipe data yang tepat sesuai dengan data yang akan disimpan. Tipe data pada MySQL dibagi menjadi beberapa kelompok. [Abdullah, Rohi. 2018]

a. Tipe Data Numerik

Tipe data numerik digunakan untuk menyimpan data angka. Yang termasuk tipe data numerik sebagai berikut:

Tabel 2.1 Tipe Data Numerik

| Tipe Data | Keterangan |
|-----------|---|
| TinyInt | Menyimpan bilangan bulat dari -128 hingga 127. Ukuran 1 byte. |
| Smallint | Menyimpan bilangan bulat dari -8388608 hingga 8388608. Ukuran 3 byte. |
| Int | Menyimpan bilangan bulat dari -2147483648 hingga 2147483648. Ukuran 4 byte. |
| BigInt | Menyimpan bilangan bulat dari $-9,22 \times 10^{18}$ hingga $9,22 \times 10^{18}$. Ukuran 8 byte. |
| Float | Menyimpan bilangan pecahan dari $-3.402823466 \times 10^{38}$ hingga $1.175494351 \times 10^{38}$, 0 dan $1.17549351 \times 10^{-38}$ hingga $3.402823466 \times 10^{38}$. Ukuran 4 byte. |
| Double | Menyimpan bilangan pecahan dari -1.79×10^{308} hingga -2.22×10^{-308} hingga 1.79×10^{308} . Ukuran 8 byte. |
| Real | Alias dari double. |
| Decimal | Menyimpan bilangan desimal |
| Numeric | Alias dari decimal. |

b. Tipe Data Waktu dan Tanggal

Untuk penyimpanan data waktu atau tanggal memiliki pilihan tipe tersendiri. Tipe data yang termasuk dalam kelompok waktu dan tanggal dapat dilihat pada tabel berikut:

Tabel 2.2 Tipe Data Waktu dan Tanggal

| Tipe Data | Keterangan |
|-----------|---|
| Date | Menyimpan data tanggal dari 1000-01-01 hingga 9999-12-31. Ukuran 3 byte. |
| Time | Menyimpan data waktu dengan jangkauan – 838:59:59 hingga 838z59z59. Ukuran 3 byte. |
| Datetime | Menyimpan data tanggal dan waktu sekaligus dengan jangkauan 1000-01-01 00-00-00 s/d 9999-12-31 23:59:59. Ukuran 8 byte. |
| Year | Menyimpan data tahun dari sebuah tanggal dengan jangkauan 1900 hingga 2155. Ukuran 1 byte. |

c. Tipe Data String

Tipe data dalam kelompok ini digunakan untuk menyimpan data berupa teks. Adapun macam-macam tipe datanya sebagai berikut:

Tabel 2.3 Tipe Data String

| Tipe Data | Keterangan |
|-----------|---|
| Char | Menyimpan data string dengan ukuran tetap dengan jangkauan 0 sampai 255 karakter. |
| Varchar | Menyimpan data string dengan ukuran dinamis. Jangkaunnya 0 sampai 255. |
| Tinytext | Menyimpan data textdengan jangkauan 0 sampai 255 karakter. |
| Text | Menyimpan data text dengan jangkauan 0 sampai 65535 karakter. |

d. Tipe Data Lainnya

Ada 2 tipe data lagi yang tidak dapat dikelompokkan ke tipe data sebelumnya, yaitu Enum dan Set.

Tabel 2.4 Tipe Data Lain

| Tipe Data | Keterangan |
|-----------|--|
| Enum | Kumpulan data(enumerasi), dapat menampung hingga 65535 string. |
| Set | Himpunan data(kombinasi), dapat menampung hingga 355 string. |

2.5 PhpMyAdmin

Setiap RDBMS (*Relation Database Management System*) seperti Oracle, SQL Server, MySQL dan lain-lain, pasti memiliki tool yang dapat digunakan untuk mempermudah pengoperasian database. Oracle memiliki *TAOD*. SQL server memiliki *Enterprise Manager* dan *SQL Query Analyzer*. Sedangkan MySQL memiliki *tool* atau aplikasi yang disebut PhpMyAdmin.

PhpMyAdmin merupakan aplikasi berbasis web yang dikembangkan menggunakan bahasa pemrograman PHP. Melalui PhpMyAdmin, *user* dapat melakukan perintah *query*. Perintah tersebut misalnya administrasi *user* dan *privileges*, *export* dan *import* database, manajemen database, manajemen tabel dan struktur tabel, dan sebagainya. PhpMyAdmin sangat *user friendly*, sehingga mudah untuk digunakan walaupun pengguna baru (*newbie*).[Fachrul, S. 2018]

2.5.1 PHP(Personal Home Page)

PHP adalah bahasa (*Scripting Language*) yang dirancang khusus untuk penggunaan *Web*. PHP adalah *tool* anda untuk pembuatan halaman *web* dinamis. Kaya akan fitur yang membuat perancangan *web* dan pemrograman lebih mudah, PHP digunakan pada 13 juta *domain*.

Seperti bahasa pemrograman web lainnya. PHP memproses seluruh perintah yang berada dalam skrip PHP di dalam *web server* dan menampilkan outputnya ke dalam *web browser* klien. PHP adalah bahasa *scripting* yang menghasilkan output HTML ataupun output-output lainnya, sesuai keinginan pemrograman (misalnya: PDF, dan lain-lain) yang dijalankan pada *server side*. Artinya, semua sintaks yang kita berikan akan sepenuhnya dijalankan pada *server* sedangkan yang dikirimkan ke *browser* hanya hasilnya (*output*) saja.[Fachrul, S. 2018]

Tujuan dari bahasa ini adalah membantu para pengembangan web untuk membuat web dinamis dengan cepat.

Agar dapat menjalankan PHP harus menyediakan perangkat lunak berikut:

- Web Server (Apache, IIS, PersonalWebServer/PWS)
- PHP Server
- Database Server(MySQL, Interbase, MS SQL, dll)

2.5.2 Konsep Dasar PHP

Kode PHP diawali dengan tanda lebih kecil (<) dan diakhiri dengan tanda lebih besar (>). Ada empat cara untuk menuliskan skrip PHP, yaitu:[Fachrul, S. 2018]

- a. <? Hallo ?>
- b. <?php Hallo ?>
- c. <% Hallo %>
- d. <SCRIPT LANGUAGE="php">Hallo</SCRIPT>

Jika ingin menambahkan komentar, namun komentar tersebut tidak ikut dieksekusi maka dapat ditulis sebagai berikut:

`*Tulis Komentar*/atau//Tulis Komentar`

Skrip yang dibuat dengan PHP disimpan dengan nama file dan diikuti dengan ekstensi*.php, misalnya: contoh.php. Bila skrip PHP diakses melalui komputer lokal maka file PHP disimpan di folder htdocs di web server. Sama halnya

dengan penamaan dokumen HTML, pemberian nama dokumen yang sama tetapi dituliskan dengan case yang berbeda akan dianggap sebagai dokumen yang berbeda, misalnya contoh.php akan berbeda dengan CONTOH.php. Skrip PHP dapat disisipkan dibagian manapun dalam dokumen HTML, begitu pula sebaliknya skrip HTML dapat diletakkan diantara skrip PHP:

*Contoh.php (menyisipkan skrip PHP di dokumen HTML)

```
<html>

<head>

<title> Menyisipkan PHP di dokumen HTML </title>

</head>

<body>

    <?php echo ('Hello world') ?>

</body>
```

2.5.3 Tipe Data PHP

Variabel pada PHP dapat menyimpan berbagai macam tipe data. Beda tipe data dapat melakukan hal yang berbeda pula. Berikut tipe data yang didukung oleh PHP: [Abdullah, Rohi. 2018]

- String, yaitu tipe data berupa teks atau angka yang ditulis di antara petik ganda, seperti: “Selamat Pagi!” atau “12345”.
- Integer, yaitu tipe data numerik non desimal antara -2,147,483,648 hingga 2,147,483,647.
- Float, yaitu tipe data numerik berupa angka desimal. Tanda koma pada angka desimal menggunakan tanda titik, misal: 23.25.
- Boolean, yaitu tipe data yang hanya dapat bernilai true atau false.
- Array, yaitu tipe data yang menyimpan banyak nilai dalam satu variabel.

- Object, yaitu tipe data yang menyimpan data beserta informasi bagaimana data diproses.
- Null, yaitu tipe data yang hanya dapat bernilai NULL. Untuk menampilkan tipe data beserta nilainya pada PHP dapat menggunakan perintah `var_dump()`.

2.5.4 Keunggulan PHP

Adapun keunggulan PHP antara lain: [Peni Sri dan Eka Bambang. 2013]

1. Cepat, karena ditempelkan (*embedded*) di dalam kode HTML, sehingga waktu tanggap menjadi pendek.
2. Tidak mahal-gratis, pada kenyataannya PHP adalah gratis dan bisa diperoleh tanpa harus membayarnya.
3. Mudah untuk digunakan, PHP berisi beberapa fitur khusus dan fungsi yang dibutuhkan untuk membuat *web* dinamis.
4. Berjalan pada beberapa sistem Operasi, dapat berjalan pada sistem operasi yang beragam, *windows*, *linux*, *Mac OS*, dan kebanyakan variasi dari *Unix*.
5. Dukungan teknis tersedia secara luas karena PHP menyediakan dukungan gratis via daftar diskusi *e-mail*.
6. Aman, pengguna tidak melihat kode PHP, karena kode yang ditampilkan pada *browser* adalah kode HTML.
7. Dirancang untuk mendukung database. PHP meliputi ampunan yang dirancang untuk berinteraksi dengan database tertentu.
8. *Customizeable*, *Lisensi open source* sehingga mengizinkan para pemrogram untuk memodifikasi *software* PHP.

2.6 Internet

Internet (*Inter-Network*) adalah sebutan untuk sekumpulan jaringan komputer yang menghubungkan situs akademik, pemerintahan, komersial, organisasi, maupun perorangan. Internet menyediakan akses untuk layanan telekomunikasi dan sumber daya informasi untuk jutaan pemakainya yang tersebar di seluruh dunia. Layanan internet meliputi komunikasi langsung (email, chat), diskusi (Usenet News, email, milis), sumber daya informasi yang terdistribusi (World Wide Web, Gopher), remote login dan lalu lintas file (Telnet, FTP), dan aneka layanan lainnya.

Jaringan yang membentuk internet bekerja berdasarkan suatu set protokol standar yang digunakan untuk menghubungkan jaringan komputer dan mengamati lalu lintas dalam jaringan. Protokol ini mengatur format data yang diijinkan, penanganan kesalahan (*error handling*), lalu lintas pesan, dan standar komunikasi lainnya. Protokol standar pada internet dikenal sebagai TCP/IP (*Transmission Control Protocol/Internet Protocol*). Protokol ini memiliki kemampuan untuk bekerja diatas segala jenis komputer, tanpa terpengaruh oleh perbedaan perangkat keras maupun sistem operasi yang digunakan. [Abudllah, Rohi. 2018]

Penggunaan internet dipilih oleh kebanyakan orang sekarang ini karena kemudahan-kemudahan yang dimiliki oleh jaringan internet, yaitu:

1. Internet sebagai jaringan publik yang sangat besar (*huge/widespread network*), layaknya yang dimiliki suatu jaringan publik elektronik, yang murah, cepat, dan kemudahan akses.
2. Menggunakan electronic data sebagai media penyampaian pesan/data sehingga dapat dilakukan pengiriman dan penerimaan informasi secara mudah dan ringkas baik dalam bentuk data *elektronik analog* dan *digital*.

2.6.1 Internet Commerce

Internet Commerce adalah penggunaan internet yang berbasis teknologi informasi dan komunikasi untuk perdagangan. Kegiatan komersial ini seperti iklan dalam penjualan produk dan jasa. Transaksi yang dapat dilakukan di internet antara lain pemesanan dan pembelian barang dimana barang akan dikirim melalui pos atau sarana lain setelah pembeli mentranser uang ke rekening penjual. Harga lebih murah biayanya dibandingkan dengan membuka *outlet retail* di berbagai tempat. Internet media promosi perusahaan dan produk yang paling tepat dengan harga yang relatif lebih murah. Serta pembelian melalui internet akan diikuti dengan layanan pengantaran barang sampai ketempat pemesanan. [Sujana, Christian dan Darmansyah. 2018]

2.6.2 Web Server

Web server adalah software yang menjadi tulang belakang dari *world wide web* (www). Web server menunggu permintaan dari client yang menggunakan browser seperti Netscape Navigator, Internet Explorer, Mozilla, dan program browser lainnya. Jika ada permintaan dari browser, maka *web server* akan memproses permintaan itu kemudian memberikan hasil prosesnya berupa data yang diinginkan kembali ke *browser*. Data ini mempunyai format yang standar, disebut dengan format SGML (*standar general markup language*).

Data yang berupa format ini kemudian akan ditampilkan oleh browser sesuai dengan kemampuan browser tersebut. Contohnya, bila data yang dikirim berupa gambar, browser yang hanya mampu menampilkan teks (misalnya *lynx*) tidak akan mampu menampilkan gambar tersebut, dan jika ada akan menampilkan alternatifnya saja. Web server, untuk berkomunikasi dengan client-nya (*web browser*) mempunyai protokol sendiri, yaitu HTTP (*hypertext transfer protocol*). [Rante, Rintho. 2018]

2.7 Hypertext Markup Language (HTML)

HTML adalah bahasa pendeskripsi halaman yang menciptakan dokumen-dokumen *hypertext* atau *hypermedia*. HTML memasukkan kode-kode pengendali dalam sebuah dokumen pada berbagai poin yang dapat anda spesifikasikan, yang dapat menciptakan hubungan (hyperlink) dengan bagian lain dari dokumen tersebut atau dengan dokumen lain yang berbeda diWorld Wide Web. [Simarmata, J. 2006]

Di Bawah ini adlaah struktur dari dokumen HTML:

```
<html>

<head>

<title>Contoh HTML</title>

</head>

<body>

    Selamat Belajar HTML

</body>

</html>
```

Berikut ini adalah penjelasan kode pada HTML:

1. Pasangan tag <html> dan </html> menandakan bahwa kode yang terdapat di dalamnya adalah kode HTML sehingga browser akan menerjemahkan sebagai dokumen HTML.
2. Dua bagian yang terdapat dalam <html> dan </html> umumna terbagi atas:
 - Kepala dan Badan
3. Bagian kepala ditandai dengan pasangan tag <head> dan </head>, sedangkan bagian badan ditandai dengan tag <body> dan </body>
4. Pada bagian kepala, anda bisa menentukan judul dokumen HTML. Judul ini ditulis dalam pasangan tag <title> dan </title>

2.8 CSS(Cascading Style Sheet)

CSS digunakan dalam kode HTML untuk menciptakan suatu kumpulan *style* yang terkadang dapat digunakan untuk memperluas kemampuan HTML, sebagai contoh, kode HTML murni tidak memungkinkan untuk mengatur ukuran font yang diterapkan pada setiap sel dan bahkan tag `<basefont>` pun tak berpengaruh pada sel-sel tabel. [Edy, Ahmad. 2019]

2.9 Code Igniter

Code Igniter merupakan aplikasi sumber terbuka yang berupa kerangka kerja PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP. Code Igniter memudahkan pengembang web untuk membuat aplikasi web dengan cepat mudah dibandingkan dengan membuatnya dari awal. [Nurrahman, M dan Akhmad, L. 2018]

2.9.1 Fungsi Code Igniter

Fungsi Code Igniter yaitu: [Anonim. 2017]

1. Mempercepat dan mempermudah dalam pembuatan website.
2. Menghasilkan struktur pemrograman yang sangat rapih, baik dari segi kode maupun struktur file phpnya.
3. Memberikan standar koding sehingga memudahkan kita atau orang lain untuk mempelajari kembali sistem aplikasi yang dibangun.

2.9.2 Kelebihan Code Igniter

Kelebihan dari Code Igniter yaitu: [Anonim. 2017]

1. Berukuran sangat kecil. File downloadnya hanya sekitar 4MB, itupun sudah *include* dokumentasinya yang sangat lengkap.

2. Kompatibilitas dengan Hosting. CodeIgniter mampu berjalan dengan baik pada hampir semua platform hosting. CodeIgniter juga mendukung database-database paling umum, termasuk MySQL.
3. Tidak ada aturan coding yang ketat.
4. Kinerja yang baik. Codeigniter sangat cepat bahkan mungkin bisa dibilang merupakan framework yang paling cepat yang ada saat ini.
5. Sangat mudah diintegrasikan. CodeIgniter sangat mengerti tentang pengembangan berbagai library saat ini. Karenanya CodeIgniter memberikan kemudahan untuk diintegrasikan dengan library-library yang tersedia saat ini.
6. Sedikit konfigurasi. Konfigurasi CodeIgniter terletak di folder application/config. CodeIgniter tidak membutuhkan konfigurasi yang rumit, bahkan untuk mencoba menjalankannya, tanpa melakukan konfigurasi sedikitpun ia sudah bisa berjalan.
7. Mudah dipelajari. Disamping dokumentasi yang lengkap, ia juga memiliki berbagai forum diskusi.

2.9.3 Kekurangan Code Igniter

Kekurangan dari code igniter yaitu: [Anonim. 2017]

1. Codeigniter tidak ditujukan untuk pembuatan web dengan skala besar.
2. Library yang sangat terbatas. Hal ini dikarenakan sangat sulit mencari plugin tambahan yang terverifikasi secara resmi, karena pada situsnya CodeIgniter tidak menyediakan plugin-plugin tambahan untuk mendukung pengembangan aplikasi dengan CI.
3. Belum adanya editor khusus CodeIgniter, sehingga dalam melakukan create project dan modul-modulnya harus berpindah-pindah folder.

2.10 Struktur Navigasi

Struktur navigasi adalah alur yang digunakan dalam aplikasi yang dibuat. Sebelumnya menyusun aplikasi multimedia kedalam sebuah software, kita harus menentukan terlebih dahulu alur apa yang akan digunakan dalam aplikasi yang dibuat. Bentuk dasar dari struktur navigasi yang biasa digunakan dalam proses pembuatan aplikasi multimedia ada empat macam, yaitu struktur navigasi linier, hirarki, non-linier dan campuran. [Setiawati, Popong. 2018]

2.11 MVC

Model View Controller (MVC) sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (Model) dari tampilan (View) dan cara bagaimana memprosesnya (Controller). Dalam implementasinya kebanyakan framework dalam aplikasi website adalah berbasis arsitektur MVC.

MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna, dan bagian yang menjadi kontrol dalam sebuah aplikasi web. [Sugiarto, R. 2015.]

2.11.1 Jenis-jenis MVC

a. *Server Side MVC*

Server Side MVC biasa terjadi pada aplikasi web tradisional, yang tidak melibatkan client side seperti Javascript, Java applet, Flash, dan lain-lain. *Server Side MVC* menyerahkan keseluruhan proses bisnis pada server, aplikasi pada sisi pengguna hanya dapat menerima. MVC jenis ini kadang-kadang disebut juga dengan nama Thin Client.

b. *Mixed Client Side and Server Side MVC*

Pada *Mixed Client Side and Server Side MVC* 1 client tidak menggunakan

model sebagai jembatan untuk melakukan komunikasi pada server, dibandingkan dengan Server Side MVC, arsitektur ini memiliki tingkat kompleksitas yang lebih tinggi karena lebih banyak komponen yang terlibat. Untuk selanjutnya arsitektur ini disebut, dengan Mixed MVC 1. Pada Mixed Client Side and Server Side MVC 2, client menggunakan model sebagai jembatan untuk melakukan komunikasi pada server, dibandingkan dengan arsitektur MVC yang lain, arsitektur ini memiliki tingkat kompleksitas yang paling tinggi karena lebih banyak komponen yang terlibat, sehingga membutuhkan sumber daya yang lebih besar pula. Untuk selanjutnya arsitektur ini disebut dengan Mixed MVC 2

c. Rich Internet Application MVC

Application MVC Rich Internet Application (RIA) disebut juga dengan nama Fat Client, merupakan aplikasi web yang memiliki kemampuan dan fungsi hampir seperti aplikasi *desktop*. RIA pada sisi client, memiliki mesin untuk mengambil data yang berada pada server, sehingga pada client terdapat bagian MVC sendiri dan hanya membutuhkan bagian model pada sisi server.

2.12 Visual Studio Code

Visual Studio Code adalah *source code editor* yang dikembangkan oleh Microsoft yang bisa digunakan pada berbagai platform (Linux, Mac OS dan Microsoft). Visual Studio Code juga dapat digunakan sebagai editor source code berbagai bahasa pemrograman dari PHP, java, Python, Javascript, Node.js, C/C++, Go, C# sampai editor script web seperti HTML dan CSS.[Reyza, 2017]

2.12.1 Fitur-fitur Visual Studio Code

Berikut ini adalah beberapa fitur-fitur visual studio code: [Arumsari, 2018]

- a. **Cross Platform** – tersedia di macOS, Linux dan Windows artinya anda dapat bekerja pada sistem operasi manapun tanpa kahawatir belajar *coding tools* yang sama untuk sistem yang berbeda-beda.
- b. **Lightweight** – tak perlu menunggu lama untuk memulai. Anda mengontrol sepenuhnya bahasa, tema, debugger, commands dan lain-lainnya sesuai keinginan. Ini dapat dilakukan melalui *extensions* untuk bahasa populer seperti Python, node.js, Java dan lain-lainnya di Visual Studio Code Marketplace.
- c. **Powerful editor** – memfungsikan fitur untuk source code editing yang sangat produktif, seperti membuat code snippets, IntelliSense, auto correct, dan formatting.
- d. **Code Debugging** – salah satu fitur terkeren ditawarkan visual studio code adalah membantu anda melakukan *debug* pada kode dengan mengawasi kode, variabel, call stack dan expression yang mana saja.
- e. **Source Control** – Visual Studio Code memiliki integrated source control termasuk Git Support in-the-box dan penyedia source code control lainnya di pasaran. Ini meningkatkan siklus rilis proyek anda secara signifikan.
- f. **Integrated Terminal** – Tiada lagi *multiple windows* dan alt-tabs. Anda dapat melakukan command-line task sekejap dan membuat banyak terminal di dalam editor.

2.13 Peneliti Terdahulu

Dalam proses pengerjaan penulisan ini, penulis melakukan pencarian penelitian-penelitian yang sudah dilakukan untuk menjelaskan posisi penelitian ini. Penelitian-penelitian sebelumnya mengenai pembuatan sebuah website *e-commerce* untuk reservasi atau pemesanan makanan juga bisa dijadikan sebagai referensi yang dapat membantu proses pengerjaan penulisan penelitian ilmiah ini.

Tabel 2.5 Tabel Penelitian Terdahulu

| No | Nama, Judul dan Tahun | Tujuan Penelitian | Metode Penelitian | Hasil Penelitian | Kelebihan | Kekurangan |
|----|--|--|--|--|--|---|
| 1 | Saputri, Intan Sandra Yatana, Mardhiah Fadhlil, and Ibnu Surya. "Penerapan Metode UCD (User Centered Design) Pada E-Commerce Putri Intan Shop Berbasis Web". 2017. | Merancang sistem antarmuka web <i>e-commerce</i> yang <i>user-friendly</i> sesuai dengan pendapat pengguna. | Metode yang digunakan oleh peneliti adalah metode <i>User Centered Design</i> (UCD), dimana metode tersebut mengambil pendapat dari para pengguna dan nantinya akan diterapkan. | Hasil dari penelitian ini adalah sebuah website <i>e-commerce</i> yang <i>user-friendly</i> dan berfungsi sesuai dengan ekspektasi pembuat dan pengguna. | Kelebihan penelitian ini adalah melakukan pengujian terhadap seluruh fungsi website tersebut dan mempresentasikan-nya, sehingga didapat sebuah web yang berfungsi dengan baik. | Kekurangan dari penelitian ini adalah menunggu pendapat atau umpan balik dari pengguna. |
| 2 | Mumtahana, Hani Atun, Sekreningsih Nita, and Adzinta Winerawan Tito. "Pemanfaatan web <i>e-commerce</i> untuk meningkatkan strategi pemasaran.". 2017. | Tujuan dari penelitian ini adalah membangun sebuah aplikasi <i>e-commerce</i> dengan begitu akan memudahkan para pelaku bisnis untuk memberi dan menerima informasi. | Metode yang digunakan untuk penelitian ini adalah metode pengembangan perangkat lunak yaitu metode <i>waterfall</i> . Tahapan-tahapan dari metode pengembangan <i>waterfall</i> yaitu: <ul style="list-style-type: none"> • Analisa • Desain • Coding • Eksperimen • Pemeliharaan | Penelitian ini menghasilkan sebuah rancangan bangun aplikasi <i>e-commerce</i> untuk pemasaran produksi industri kulit di Kabupaten Magetan. | Kelebihan dari penelitian ini adalah merancang dan menganalisa seluruh sistemnya dalam sebuah diagram, sehingga bisa membuat website <i>e-commerce</i> yang sesuai. | Kekurangan dari penelitian ini adalah metode yang digunakan pada penelitian ini belum menghasilkan nilai kuantitatif pada hasil uji coba. |

| | | | | | | |
|---|---|--|---|---|---|---|
| 3 | Kasmi, Kasmi, And Adi Nurdian Candra. "Penerapan E-Commerce Berbasis Business To Consumers Untuk Meningkatkan Penjualan Produk Makanan Ringan Khas Pringsewu.". 2017. | Tujuan penelitian ini adalah membangun aplikasi <i>e-commerce</i> berjenis <i>business to consumer</i> berbasis web untuk meningkatkan penjualan produk makanan. | Peneliti menggunakan metode pengumpulan data yang meliputi: <ul style="list-style-type: none"> • Observasi • Angket • Interview • Pustaka Dan peneliti menggunakan metode <i>waterfall</i> dalam membangun aplikasinya. | Hasil dari penelitian ini adalah sebuah website untuk Toko Jajanan Pringsewu, dan di dapatkan hasilnya yaitu peningkatan penjualan yang signifikan pada Toko Jajanan Pringsewu. | Kelebihan dari penelitian ini adalah metode yang digunakan oleh peneliti, dimana peneliti mengumpulkan data langsung ke toko yang bersangkutan. | Kekurangan dari penelitian ini adalah untuk mendapatkan umpan balik untuk web tersebut, peneliti harus langsung bertanya kepada orang yang bersangkutan. |
| 4 | Andi, Amelia, and Yakub. "Analysis and Design of E-Commerce Information Systems Using the User Centered Design Method at PT. Pure Healthy Nature." 2019 | Tujuan penelitian ini adalah menganalisa desain dari web <i>e-commerce</i> PT. Pure Healthy Nature yang menggunakan metode <i>User Centered Design</i> . | Peneliti membuat sebuah tabel pertanyaan dan mengumpulkan responden, yang kemudian di analisa seperti apa web <i>e-commerce</i> yang diinginkan oleh banyak pengguna. | Hasil dari penelitian ini adalah penunjukkan bahwa metode UCD sangat efektif dalam membangun sebuah website <i>e-commerce</i> . Pengembangan web akan mengetahui apa pendapat para pengguna saat mereka mengakses web tersebut. | Kelebihan dari penelitian ini adalah penjelasannya yang lengkap dan rinci tentang metode UCD, dan juga disertai data-data yang aktual. | Kekurangan dari penelitian ini adalah kurangnya pembahasannya lebih untuk web tersebut. Di dalam jurnal hanya menunjukkan tampilan utama dan detail produknya saja. |
| 5 | Sari, Dwi Karunia. "Membangun Aplikasi E-Commerce Berbasis Website Sebagai Media Penjualan". 2018. | Tujuan dari penelitian ini adalah menghasilkan sistem informasi terkomputerisasi berbasis web untuk mempermudah pertukaran data antar toko dan mempermudah transaksi | Peneliti menggunakan metode Replikasi Synchronous dan Asynchronous untuk mengatasi permasalahan transaksi dan pertukaran data antara cabang satu dan cabang lainnya. Peneliti mengkoleksi | Hasil dari penelitian ini adalah sebuah aplikasi <i>e-commerce</i> yang dapat mempermudah dalam mempromosikan produk, memasarkan, dan mempermudah pelaporan penjualan. | Kelebihan dari penelitian ini adalah metode yang digunakan oleh peneliti dalam mengumpulkan data cukup efektif karena langsung kepada pemiliknya. Peneliti juga menjelaskan | Kekurangan dari penelitian ini terletak pada tahap eksperimen, karena tentu untuk memastikan bahwa web tersebut sudah baik atau belum pengembang harus menanyakan langsung kepada pemilik toko. |

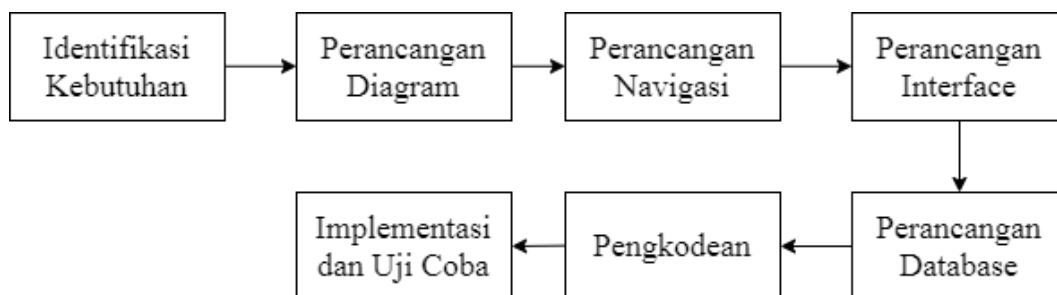
| | | | | | | |
|--|--|--|---|--|---|--|
| | | | data langsung kepada pemilik toko dengan cara datang langsung ke toko tersebut. | | secara rinci dan memberikan gambaran tentang web yang telah dibuat. | |
|--|--|--|---|--|---|--|

BAB 3

PERANCANGAN DAN IMPLEMENTASI

3.1 Gambaran Umum

Pada bab ini penulis akan membahas mengenai pembuatan aplikasi *e-commerce* berbasis web untuk toko makanan “DapurMbokNdeso”. Metode yang digunakan untuk merancang aplikasi ini adalah metode *waterfall*, berikut merupakan tahap-tahap yang dilakukan dalam pembuatan aplikasi ini adalah identifikasi kebutuhan, perancangan, pengkodean, pengisian konten, implementasi dan uji coba. Secara umum metode pada pembuatan aplikasi *e-commerce* toko makanan “DapurMbokNdeso” dapat dilihat pada gambar 3.1.



Gambar 3.1. Bagan umum metode pembuatan aplikasi toko makanan DapurMbokNdeso

Untuk merancang dan membangun aplikasi *web e-commerce* toko makanan “DapurMbokNdeso” ini, penulis menggunakan bahasa pemrograman PHP dengan framework Codeigniter dan basis data MySQL. Pada halaman utama terdapat bilah navigasi diatas halaman web yang mengandung menu *log-in* dan keranjang belanja. Kemudian ditengah halaman utama web terdapat informasi tentang produk-produk yang dijual.

Aplikasi *e-commerce* untuk toko makanan “DapurMbokNdeso” berbasis web dibuat untuk dapat memudahkan pengguna dalam memesan barang, pada aplikasi ini pengguna dapat melihat informasi barang yang tertera pada halaman

utama, yang digunakan oleh penjual untuk mempromosikan produk-produknya. Sebelum memesan barang, pengguna diharuskan untuk registrasi terlebih dahulu dengan cara memasukkan data diri, alamat *e-mail*, dan *password*. Jika pengguna sudah memiliki akun, pengguna dapat *log-in* dan dapat melakukan pemesanan. Pelanggan melakukan pembayaran melalui transfer bank menggunakan mesin ATM dan mengirim bukti pembayaran melalui *e-mail* atau nomor *Whatsapp* yang tertera pada daftar kontak. Setelah pembeli selesai melakukan pembayaran dan menunjukkan bukti pembayaran kepada penjual, penjual akan langsung memproses pesanan pembeli, pembeli juga dapat memilih untuk melakukan pembayaran dengan cara cash on delivery (COD).

Kemudian untuk halaman bagian admin. Pada halaman ini admin memiliki wewenang untuk melihat dan menerima pesanan barang oleh pembeli, sekaligus dapat mengelola data-data produk.

3.2 Identifikasi Kebutuhan

Pada tahap ini akan membahas tentang identifikasi kebutuhan apa saja yang diperlukan dalam pembuatan aplikasi *e-commerce* toko makanan “DapurMbokNdeso”. Tahap identifikasi kebutuhan ini dibagi menjadi beberapa bagian yaitu identifikasi kebutuhan perangkat yang digunakan (*software* dan *hardware*), identifikasi kebutuhan aktor yang terlibat, dan identifikasi kebutuhan fungsional pada aplikasi ini.

3.2.1 Identifikasi Kebutuhan Perangkat

Spesifikasi *hardware* dan *software* yang digunakan dalam pembuatan aplikasi *e-commerce* toko makanan “DapurMbokNdeso” berbasis web antara lain. Spesifikasi *hardware* yang digunakan antara lain:

1. Processor Intel Core i3-2120 CPU 3.30GHz.
2. Memori 8192MB RAM.
3. VGA Nvidia GeForce GT 720.
4. Harddisk 500 GB.
5. Keyboard.

6. Mouse.
7. Monitor.

Spesifikasi *software* yang digunakan antara lain:

1. Windows 7 Ultimate (64-bit).
2. XAMPP Web Server v3.2.4.
3. Visual Studio Code v1.43.1.
4. Google Chrome

3.2.2 Identifikasi Kebutuhan Aktor

Aktor adalah orang atau sistem lain yang berinteraksi dengan sistem yang akan dibuat. Berikut adalah tabel 3.1 yang akan menjelaskan aktor yang terlibat dalam penggunaan aplikasi ini.

Tabel 3.1 Kebutuhan Aktor yang terlibat

| No. | Aktor | Deskripsi |
|-----|---------|---|
| 1. | Admin | Admin dalam sistem adalah orang yang bertugas pada halaman sisi server, admin mempunyai wewenang untuk mengelola data dan akun pembeli, mengakses data produk dan data pesanan pembeli, menambah produk, mengubah data produk, dan menghapus data produk maupun data pesanan pada halaman sisi server untuk aplikasi pemesanan ini. |
| 2. | Pembeli | Pembeli merupakan pengguna atau <i>user</i> pada aplikasi pemesanan ini, pembeli hanya dapat melihat produk pada daftar menu, membuat akun, menambahkan item pada keranjang belanja, dan melakukan pembayaran. |

3.2.3 Identifikasi Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berisi tentang proses atau layanan apa saja yang dapat digunakan oleh aktor-aktor pada aplikasi ini.

Identifikasi kebutuhan fungsional ini akan dibuat dalam bentuk tabel berdasarkan identifikasi kebutuhan aktor pada tabel 3.1 diatas.

3.2.3.1 Kebutuhan Fungsional Admin

Berdasarkan tabel 3.1 kebutuhan aktor yang terlibat, maka kebutuhan fungsional admin pada aplikasi ini adalah sebagai berikut.

Tabel 3.2 Kebutuhan fungsional admin

| No | Fungsional |
|-----|--------------------------------|
| 1. | Login |
| 2. | Logout |
| 3. | Mengelola data pembeli |
| 4. | Mengelola akun pembeli |
| 5. | Mengakses data produk |
| 6. | Mengakses data pesanan pembeli |
| 7. | Menambah produk |
| 8. | Mengubah data produk |
| 9. | Menghapus data produk |
| 10. | Menghapus data pesanan pembeli |

3.2.3.2 Kebutuhan Fungsional Pembeli

Berdasarkan tabel 3.1 kebutuhan aktor yang terlibat, maka kebutuhan fungsional pembeli pada aplikasi ini adalah sebagai berikut.

Tabel 3.3 Kebutuhan fungsional pembeli

| No | Fungsional |
|----|-------------------------------|
| 1. | Login |
| 2. | Logout |
| 3. | Registrasi |
| 4. | Melihat produk pada dashboard |
| 5. | Memesan barang |
| 6. | Melakukan pembayaran |

| | |
|----|--------------------|
| 7. | Mengubah data akun |
| 8. | Melihat pembelian |

3.3 Perancangan

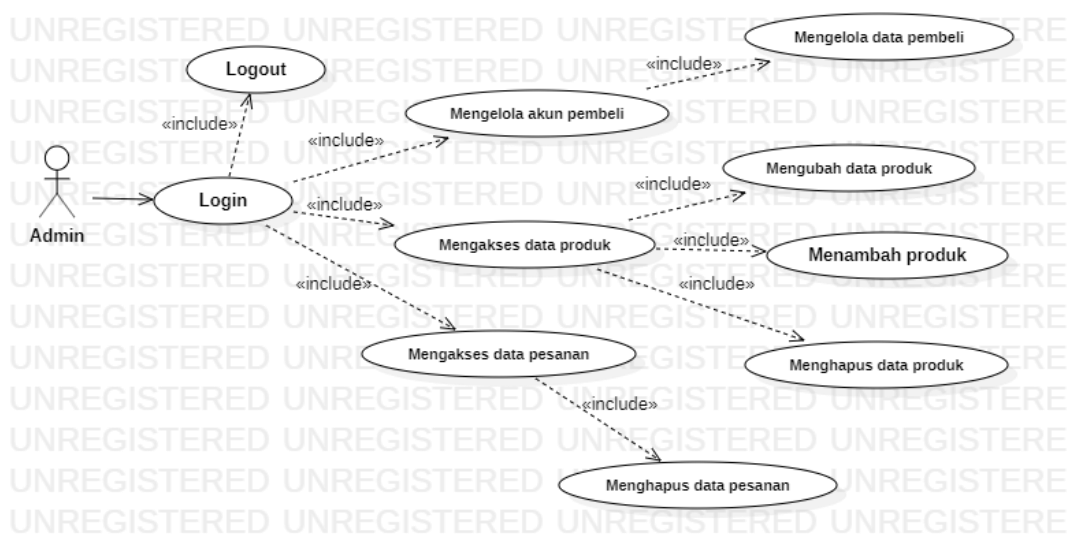
Perancangan yang dilakukan penulis dalam pembuatan aplikasi e-commerce toko makanan “DapurMbokNdeso” terbagi menjadi beberapa tahap yaitu perancangan diagram *use case*, perancangan struktur navigasi, perancangan interface, dan perancangan database.

3.3.1 Perancangan Diagram Use Case

Diagram *use case* merupakan gambaran grafis dari beberapa atau semua aktor, *use case*, dan interaksi diantaranya yang memperkenalkan suatu sistem. Pada tahap perancangan *use case diagram* ini dibagi menjadi 2 (dua) yaitu *use case* untuk admin dan *use case* untuk pembeli.

3.3.1.1 Diagram Use Case Admin

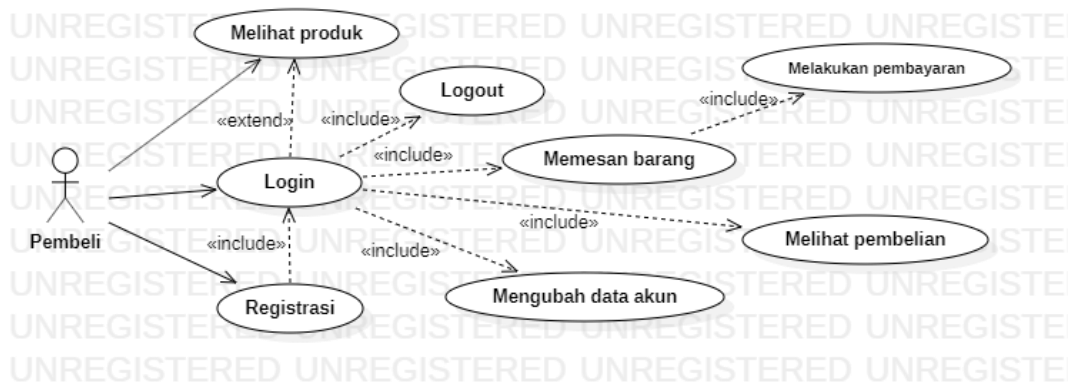
Berdasarkan identifikasi kebutuhan fungsional diatas, maka dapat dibuat gambar 3.2 *use case* untuk admin sebagai berikut.



Gambar 3.2 Diagram *use case* admin

3.3.1.2 Diagram Use Case Pembeli

Berdasarkan identifikasi kebutuhan fungsional diatas, maka dapat dibuat gambar 3.3 *use case* untuk pembeli sebagai berikut.



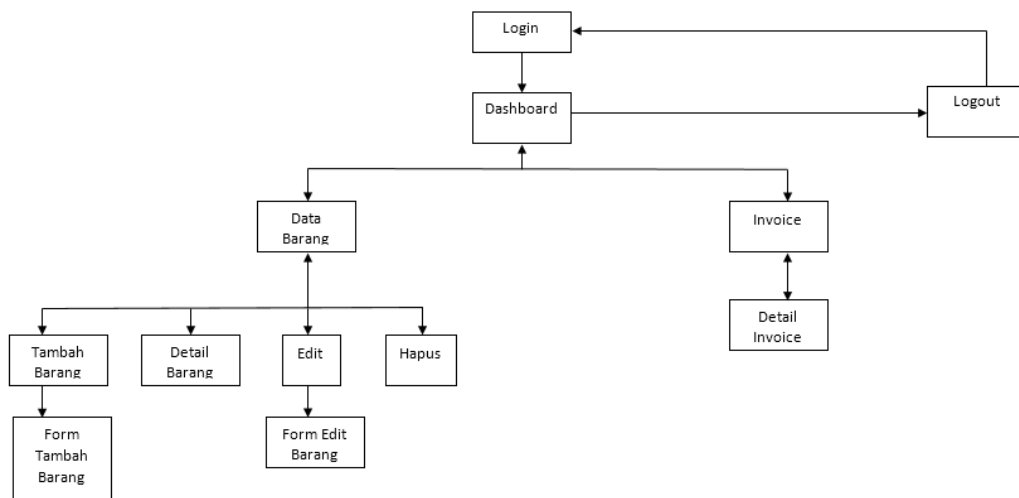
Gambar 3.3 Diagram *use case* pembeli

3.3.2 Perancangan Struktur Navigasi

Untuk mengetahui alur kerja dari aplikasi dibutuhkan struktur navigasi. Membuat struktur navigasi merupakan salah satu hal yang penting karena dapat melihat struktur atau alur kerja website tersebut. Dibawah ini merupakan struktur navigasi dari website e-commerce toko makanan “DapurMbokNdeso” yang dibuat berdasarkan perancangan diagram *use case* diatas.

3.3.2.1 Struktur Navigasi Admin

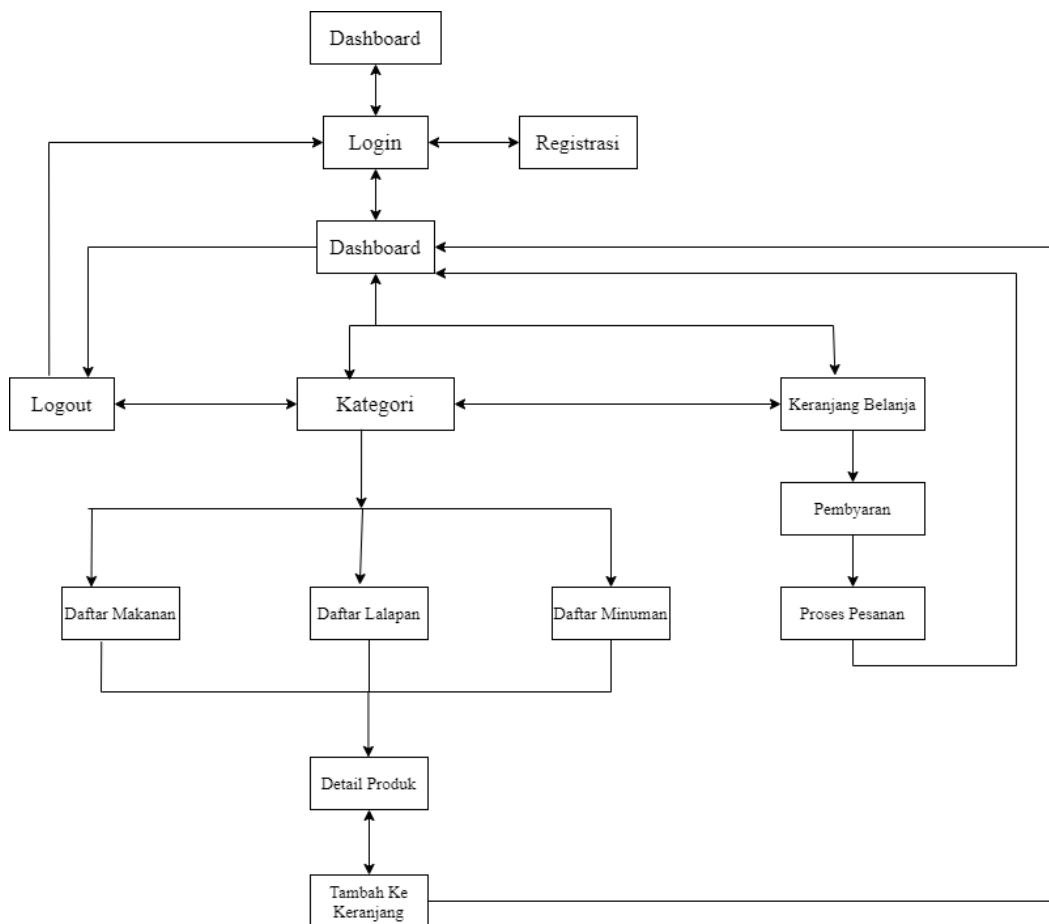
Dalam struktur navigasi admin digunakan struktur navigasi campuran yaitu struktur navigasi linier dan struktur navigasi hirarki. Struktur navigasi admin ini menggambarkan tentang bagaimana admin berinteraksi pada website ini, berikut adalah gambar 3.4 struktur navigasi admin.



Gambar 3.4 Struktur navigasi admin

3.3.2.2 Struktur Navigasi Pembeli

Dalam struktur navigasi pengguna digunakan struktur navigasi campuran yaitu struktur navigasi linier dan struktur navigasi hirarki. Struktur navigasi pengguna ini menggambarkan tentang bagaimana pengguna berinteraksi dengan website tersebut, berikut adalah gambar 3.5 struktur navigasi pembeli.



Gambar 3.5 Struktur navigasi pembeli

3.3.3 Perancangan Interface

Pada tahap ini, penulis akan menjelaskan tahapan perancangan interface untuk website ini yang terdiri dari tampilan halaman utama, tampilan halaman menu, tampilan halaman admin.

3.3.3.1 Perancangan Interface Halaman Utama Pembeli

Pada interface halaman utama ini mempunyai empat sub menu yaitu menu keranjang untuk mengumpulkan menu yang ingin dipesan, menu login untuk admin atau user, menu kategori untuk melihat daftar makanan, lalapan, dan minuman yang tersedia. Rancangan interface halaman utama pembeli terdapat pada gambar 3.6.

| | | |
|----------------|---|-------|
| DapurMbokNdeso | Keranjang Belanja | LOGIN |
| Dashboard | <p>Selamat Datang</p> <div>Carousel</div> <div>Info Website</div> | |
| Daftar Makanan | | |
| Daftar Lalapan | | |
| Daftar Minuman | | |
| Social Media | | |
| | | |

Gambar 3.6 Rancangan Interface Halaman Utama Pembeli.

3.3.3.2 Rancangan Interface Login

Halaman tampilan login ini digunakan untuk user atau admin masuk ke dalam web. Rancangan interface login user atau admin terdapat pada gambar 3.7.

LOGIN

Username

Password

Login

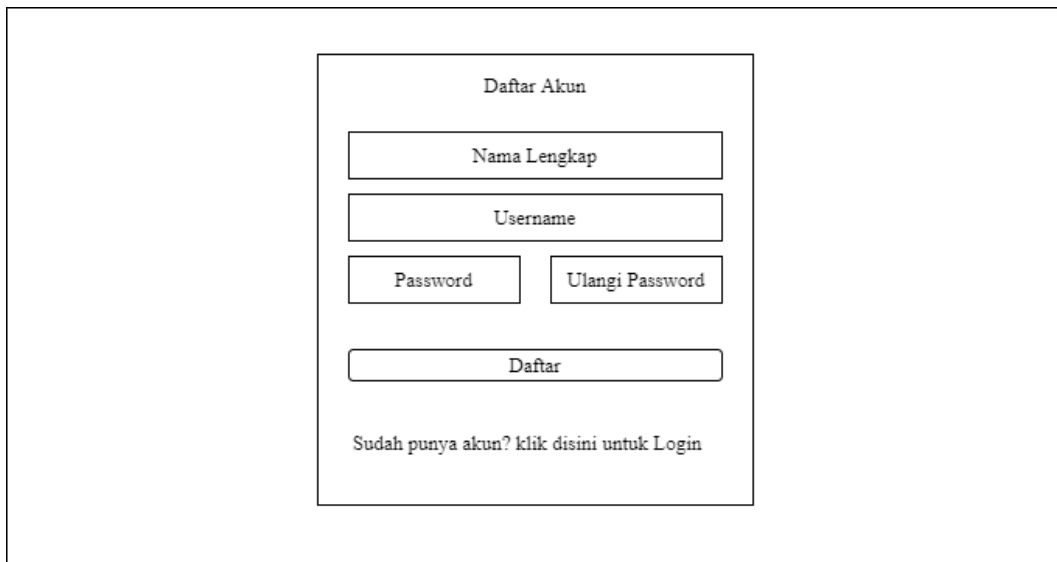
Kembali ke dashboard

Belum punya akun? daftar klik disini

Gambar 3.7 Rancangan Interface Login.

3.3.3.3 Rancangan Interface Registrasi

Halaman tampilan registrasi ini berguna untuk pengunjung website DapurMbokNdeso *non-user* menjadi *user member*. Rancangan interface halaman registrasi terdapat pada gambar 3.8.



The image shows a registration form titled "Daftar Akun". It contains the following elements:

- A title "Daftar Akun" at the top.
- A text input field labeled "Nama Lengkap".
- A text input field labeled "Username".
- Two text input fields side-by-side: "Password" and "Ulangi Password".
- A "Daftar" button.
- A link at the bottom: "Sudah punya akun? klik disini untuk Login".

Gambar 3.8 Rancangan Interface Halaman Registrasi.

3.3.3.4 Rancangan Interface Daftar Menu

Halaman daftar menu ini terdiri dari daftar makanan, daftar lalapan, dan daftar minuman. Halaman ini digunakan oleh pengguna untuk melihat menu-menu yang tersedia, terdiri dari gambar, harga, detail produk dan bisa dipilih untuk dimasukkan kedalam keranjang belanja. Rancangan tampilan daftar menu terdapat pada gambar 3.9.

| | | | |
|----------------|---|--|--|
| DapurMbokNdeso | Keranjang Belanja LOGIN | | |
| Dashboard | <div> <div>Gambar Produk</div> <div>Deskripsi Harga</div> <div>Tambah Detail</div> </div> <div> <div>Gambar Produk</div> <div>Deskripsi Harga</div> <div>Tambah Detail</div> </div> <div> <div>Gambar Produk</div> <div>Deskripsi Harga</div> <div>Tambah Detail</div> </div> | | |
| Daftar Makanan | | | |
| Daftar Lalapan | | | |
| Daftar Minuman | | | |
| Social Media | | | |
| | <div> <div>Gambar Produk</div> <div>Deskripsi Harga</div> <div>Tambah Detail</div> </div> <div> <div>Gambar Produk</div> <div>Deskripsi Harga</div> <div>Tambah Detail</div> </div> <div> <div>Gambar Produk</div> <div>Deskripsi Harga</div> <div>Tambah Detail</div> </div> | | |

Gambar 3.9 Rancangan Interface Halaman Daftar Menu

3.3.3.5 Rancangan Interface Keranjang Belanja

Halaman keranjang belanja ini digunakan untuk memudahkan pembeli dalam melihat pesanan, menentukan jumlah pesanan, menghapus pesanan dan melihat total belanja. Rancangan tampilan keranjang belanja terdapat pada gambar 3.10.

| | | | | | |
|----------------|--|--------------------|--------|-------|-----------|
| DapurMbokNdeso | Keranjang Belanja | Halo, Nama User | LOGOUT | | |
| Dashboard | Keranjang Belanja | | | | |
| Daftar Makanan | No | Nama Produk | Jumlah | Harga | Sub-total |
| Daftar Lalapan | | | | | |
| Daftar Minuman | | | | | |
| Social Media | | | | | |
| | <div>Hapus Keranjang</div> <div>Lanjut Belanja</div> <div>Pembayaran</div> | | | | |

Gambar 3.10 Rancangan Interface Halaman Keranjang Belanja

3.3.3.6 Rancangan Interface Detail Produk

Halaman detail produk ini digunakan untuk melihat informasi dari produk makanan atau minuman. Rancangan interface detail produk terdapat pada gambar 3.11.

| | | | |
|----------------|---|--------------------|--------|
| DapurMbokNdeso | Keranjang Belanja | Halo, Nama User | LOGOUT |
| Dashboard | <div> <div>Gambar Produk</div> <div>Detail Produk</div> <div>Tambah Ke Keranjang</div> </div> | | |
| Daftar Makanan | | | |
| Daftar Lalapan | | | |
| Daftar Minuman | | | |
| Social Media | | | |
| | | | |

Gambar 3.11 Rancangan Interface Halaman Detail Produk

3.3.3.7 Rancangan Interface Dashboard Admin

Halaman dashboard admin adalah halaman awal admin ketika sudah login. Rancangan interface dashboard admin terdapat pada gambar 3.12.

| | | |
|-------------|--------------------------------|---------|
| Admin | Halo, Admin Willie | LOGOUT |
| Dashboard | <div>Teks petunjuk admin</div> | |
| Data Barang | | |
| Invoice | | |
| | Data Barang | Invoice |

Gambar 3.12 Rancangan Interface Halaman Dashboard Admin

3.3.3.8 Rancangan Interface Data Barang Admin

Halaman data barang admin ini digunakan untuk melihat, mengedit atau menambahkan barang yang akan dijual. Di halaman ini terdapat tabel untuk menampilkan data produk, yang terdiri dari no, nama, keterangan, kategori, harga, stok produk. Rancangan data barang admin terdapat pada gambar 3.13.

| | | | | | | | |
|-------------|---------------------------------|-------------|------------|----------|-------|------|------|
| Admin | Halo, Admin Willie LOGOUT | | | | | | |
| Dashboard | Tambah Barang | | | | | | |
| Data Barang | No | Nama Barang | keterangan | Kategori | Harga | Stok | Aksi |
| Invoice | | | | | | | |
| | | | | | | | |

Gambar 3.13 Rancangan Interface Data Barang Admin

3.3.3.9 Rancangan Interface Invoice Pemesanan

Halaman invoice pemesanan ini digunakan admin untuk melihat data pembeli yang sudah memesan produk. Di halaman ini terdapat tabel yang berisikan data pemesanan, terdiri dari nama pemesan, alamat, tanggal pemesanan, dan batas pembayaran. Rancangan interface invoice pemesanan terdapat pada gambar 3.14.

| | | | | | | |
|-------------|---------------------------------|--------------|-------------------|-----------|-------------|------|
| Admin | Halo, Admin Willie LOGOUT | | | | | |
| Dashboard | Invoice Pemesanan | | | | | |
| Data Barang | Id Invoice | Nama Pemesan | Alamat Pengiriman | tgl.pesan | Batas Bayar | Aksi |
| Invoice | | | | | | |
| | | | | | | |

Gambar 3.14 Rancangan Interface Invoice Pemesanan

3.3.3.10 Rancangan Interface Detail Invoice

Halaman detail invoice ini digunakan admin untuk melihat detail pesanan pembeli, terdiri dari id makanan, nama makanan, jumlah pesanan, harga, dan jumlah harga. Rancangan interface detail invoice terdapat pada gambar 3.15.

| | | | | | |
|-------------|---|--------------|----------------|--------------|-----------|
| Admin | Halo, Admin Willie LOGOUT | | | | |
| Dashboard | Detail Invoice <input type="text" value="No. invoice"/> | | | | |
| Data Barang | Id Makanan | Nama Makanan | Jumlah Pesanan | Harga Satuan | Sub-Total |
| Invoice | | | | | |
| | | | | | |
| | Grand Total | | | | |
| | <input type="button" value="Kembali"/> | | | | |

Gambar 3.15 Rancangan Interface Detail Invoice

3.3.4 Perancangan Database

Untuk membuat struktur database harus mengetahui atribut-atribut apa saja yang harus diperlukan dalam pembuatan tabel di dalam database. Setelah mengetahui atribut-atribut apa saja yang diperlukan, juga menentukan jenis tipe data apa saja yang diperlukan untuk atribut-atribut tersebut.

a. Tabel makanan

Tabel makanan digunakan untuk menyimpan data makanan atau minuman yang tersedia di DapurMbokNdeso. Detail struktur tabel makanan terdapat pada Tabel 3.4.

Tabel 3.4 Struktur Tabel makanan

| No | Nama | Jenis | Atribut | Extra |
|----|------------|--------------|---------|----------------|
| 1 | id_mkn | int(11) | | AUTO_INCREMENT |
| 2 | nama_mkn | varchar(100) | | |
| 3 | Keterangan | varchar(225) | | |
| 4 | kategori | varchar(60) | | |
| 5 | harga | int(11) | | |
| 6 | stok | int(4) | | |
| 7 | gambar | text | | |

b. Tabel tb_invoice

Tabel tv_invoice digunakan untuk menyimpan data pesanan dan pemesan seperti nama, alamat, tanggal pesan dan batas bayar. Detail struktur tb_invoice terdapat pada tabel 3.5.

Tabel 3.5 Struktur Tabel tb_invoice.

| No | Nama | Jenis | Atribut | Extra |
|----|-------------|--------------|---------|----------------|
| 1 | id | int(11) | | AUTO_INCREMENT |
| 2 | nama | varchar(100) | | |
| 3 | alamat | varchar(225) | | |
| 4 | tgl_pesan | datetime | | |
| 5 | batas_bayar | datetime | | |

c. Tabel tb_pesanan

Tabel tb_pesanan digunakan untuk menyimpan data pesanan seperti nama, jumlah pesanan, harga, sampai pilihan pesanan. Detail struktur tb_pesanan terdapat pada Tabel 3.6.

Tabel 3.6 Struktur Tabel tb_pesanan.

| No | Nama | Jenis | Atribut | Extra |
|----|------------|-------------|---------|----------------|
| 1 | id | int(11) | | AUTO_INCREMENT |
| 2 | id_invoice | int(11) | | |
| 3 | id_mkn | int(11) | | |
| 4 | nama_mkn | varchar(50) | | |
| 5 | jumlah | int(3) | | |
| 6 | harga | int(10) | | |

d. Tabel tb_user

Tabel tb_user digunakan untuk menyimpan akun yang telah terdaftar dalam website DapurMbokNdeso, terdapat 2 jenis akun dalam tabel ini yaitu akun *user* dan akun admin. Detail struktur tb_user terdapat pada Tabel 3.7.

Tabel 3.7 Struktur Tabel tb_user

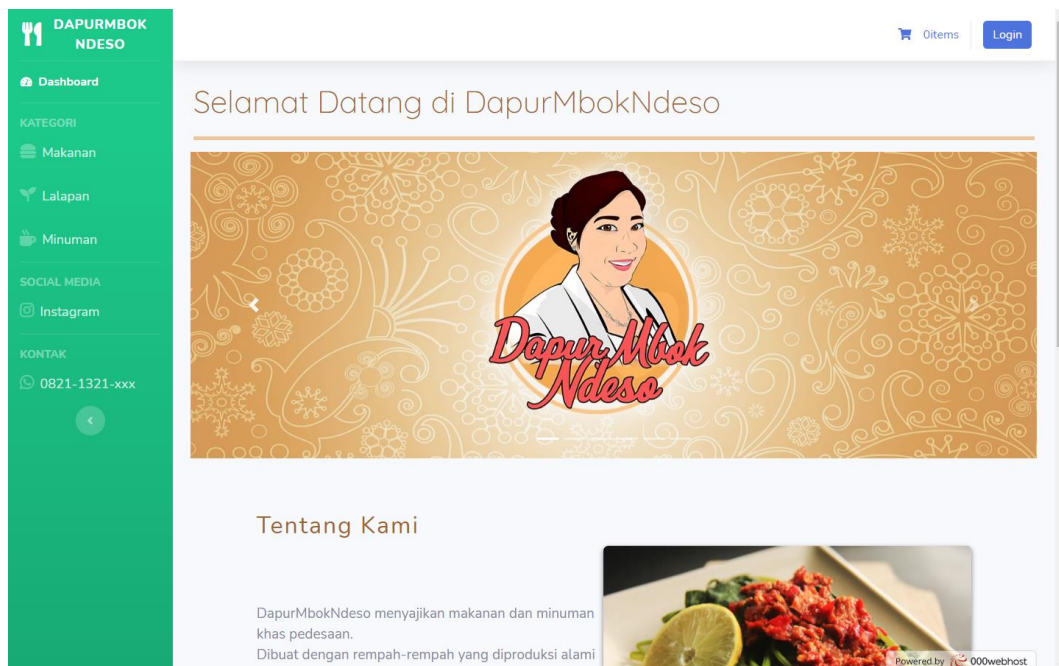
| No | Nama | Jenis | Atribut | Extra |
|----|----------|-------------|---------|----------------|
| 1 | id | int(11) | | AUTO_INCREMENT |
| 2 | nama | varchar(50) | | |
| 3 | username | varchar(50) | | |
| 4 | password | varchar(50) | | |
| 5 | role_id | tinyint(1) | | |

3.4 Implementasi

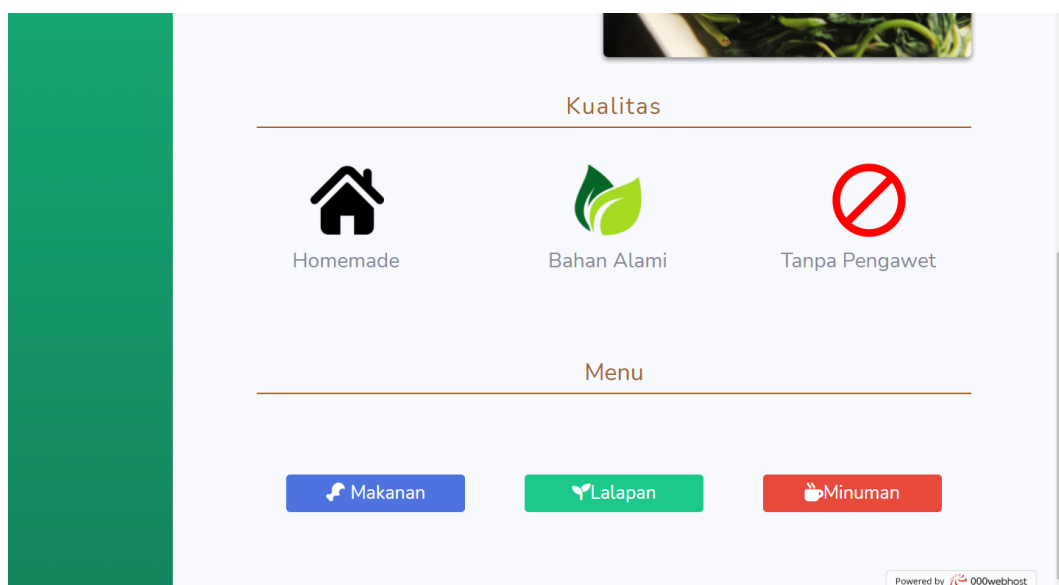
Pada bagian ini penulis akan menjelaskan implementasi tiap-tiap halaman yang ada di dalam website toko makanan DapurMbokNdeso, mulai dari implementasi tampilan halaman utama, halaman daftar menu(makanan, minuman, lalapan), keranjang belanja, pembayaran, *login/logout*, registrasi, halaman utama admin, data barang, dan invoice. Setelah selesai mengimplementasikan halaman-halaman tersebut penulis akan melakukan uji coba, apakah semua halaman dapat berfungsi sesuai dengan yang diharapkan.

3.4.1 Tampilan Halaman Utama

Halaman utama merupakan halaman yang akan dikunjungi oleh user ketika membuka web “DapurMbokNdeso”. Tampilan halaman utama terdapat pada gambar 3.16 dan gambar 3.17.



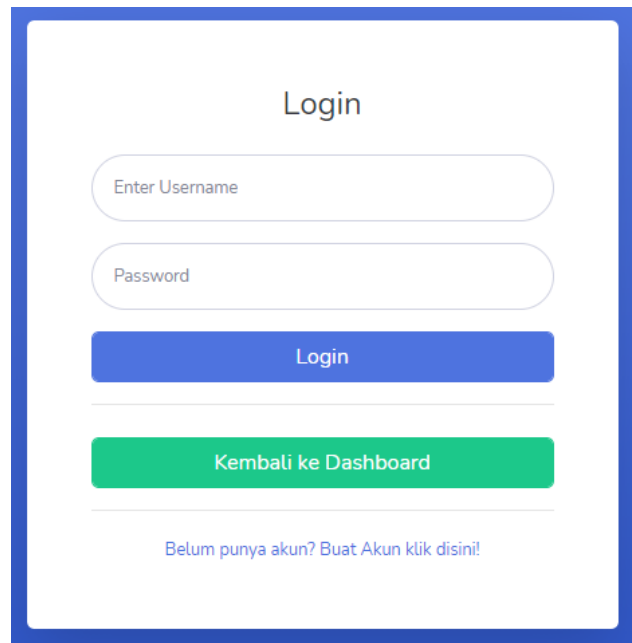
Gambar 3.16 Tampilan Halaman Utama



Gambar 3.17 Tampilan Halaman Utama

3.4.2 Tampilan Login

Halaman tampilan login berfungsi untuk *user* masuk sebagai anggota terdaftar dan untuk admin saat melakukan login akan otomatis ke halaman admin. Tampilan login terdapat pada gambar 3.18.

The image shows a login form titled "Login" centered at the top. Below the title are two input fields: "Enter Username" and "Password", both with rounded rectangular borders. Under the password field is a blue "Login" button. Below the button is a horizontal line, followed by a green button labeled "Kembali ke Dashboard". Another horizontal line is below this button. At the bottom, there is a link that says "Belum punya akun? Buat Akun klik disini!". The entire form is enclosed in a blue border.

Gambar 3.18 Tampilan Login

3.4.3 Tampilan Registrasi

Halaman registrasi ini berguna untuk *user* mendaftar akun baru pada web “DapurMbokNdeso”. Tampilan registrasi terdapat pada gambar 3.19.

Daftar Akun!

Nama

Username

Password

Ulangi Password

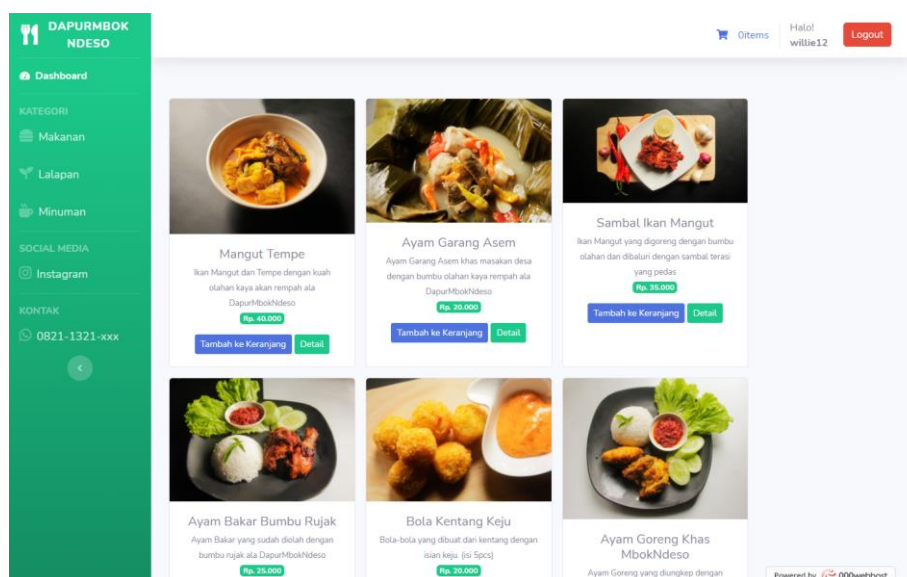
Daftar Akun

[Sudah Punya Akun? Klik disini untuk Login](#)

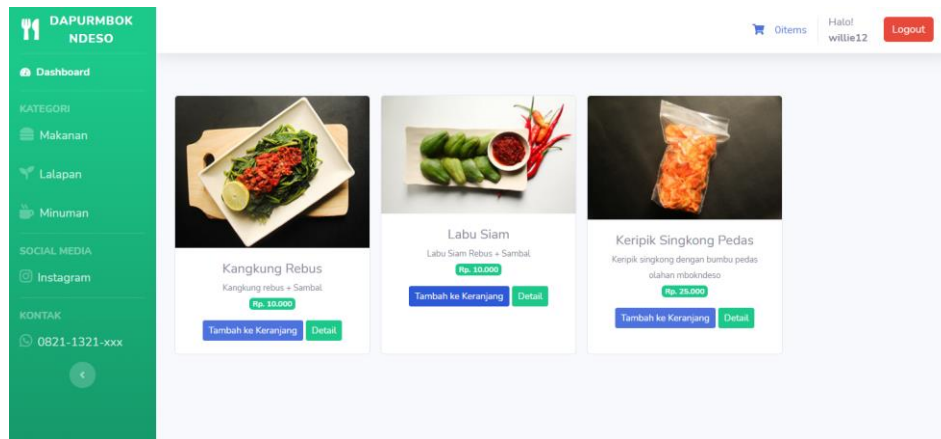
Gambar 3.19 Tampilan Registrasi Akun

3.4.4 Tampilan Daftar Menu

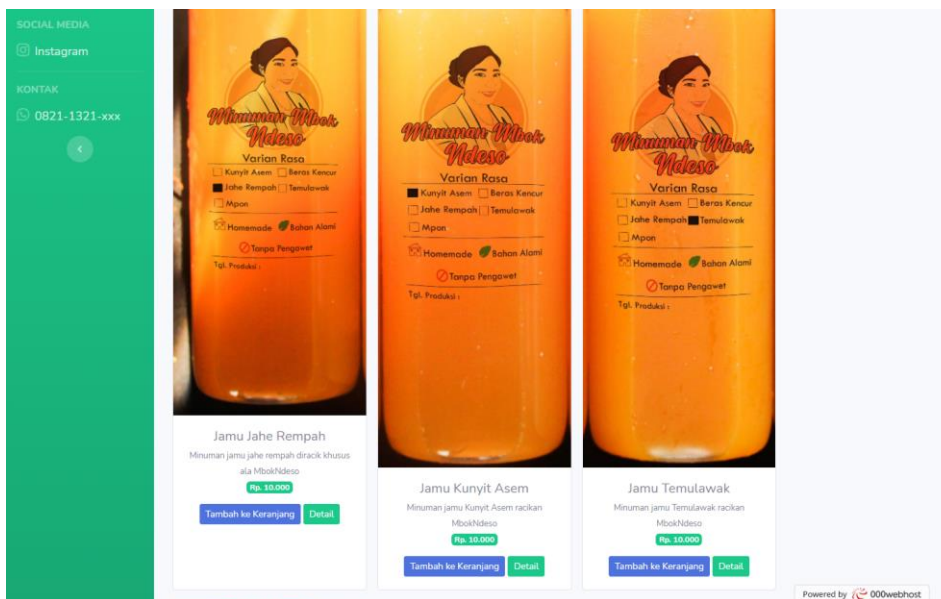
Halaman daftar menu ini berguna untuk *user* melihat list menu-menu yang tersedia. Daftar menu ini terdiri dari 3 bagian yaitu daftar menu makanan, lalapan dan minuman. Halaman ini terdiri dari gambar produk, keterangan, harga, detail produk dan tambah ke keranjang. Tampilan daftar menu terdapat pada gambar 3.20, gambar 3.21, dan gambar 3.21.



Gambar 3.20 Tampilan Daftar Menu Makanan.



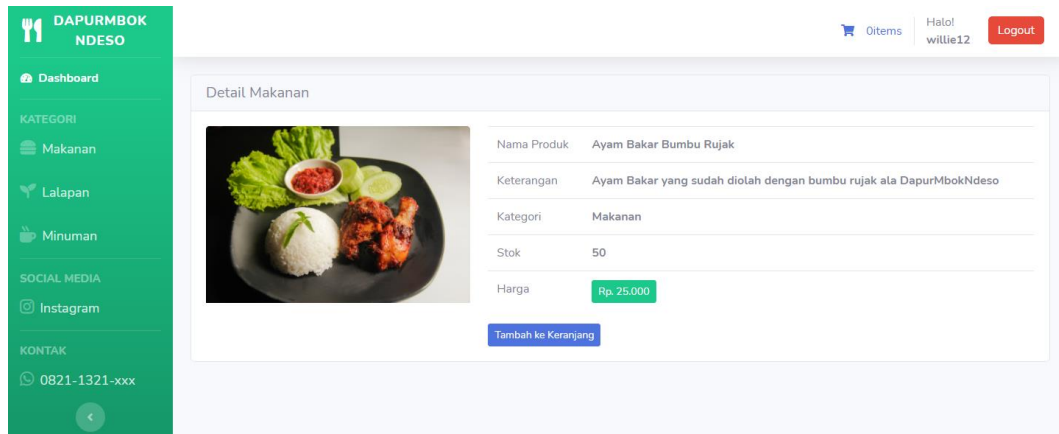
Gambar 3.21 Tampilan Daftar Menu Lalapan.



Gambar 3.22 Tampilan Daftar Menu Minuman.

3.4.5 Tampilan Detail Produk

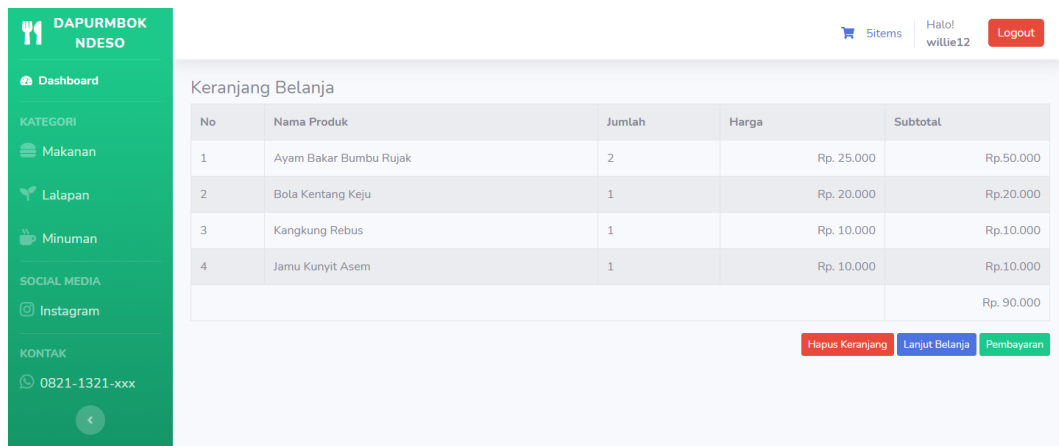
Halaman detail produk ini berguna untuk melihat lebih lengkap produk yang dijual. Rancangan tampilan detail produk terdapat pada gambar 3.23.



Gambar 3.23 Tampilan Detail Produk.

3.4.6 Tampilan Keranjang Belanja

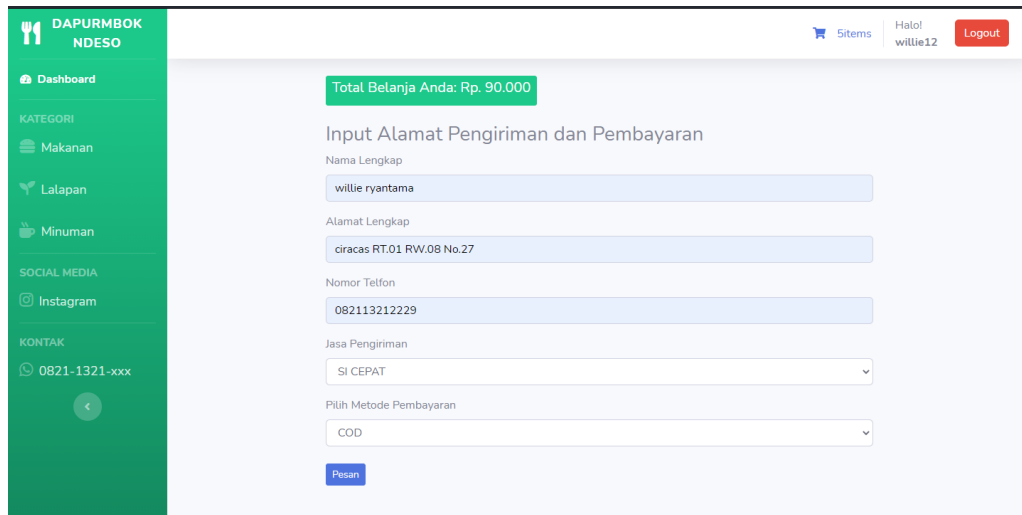
Halaman keranjang belanja ini digunakan untuk memudahkan pembeli dalam melihat pesanan, menentukan jumlah pesanan, menghapus pesanan dan melihat total belanja. Rancangan tampilan keranjang terdapat pada gambar 3.24.



Gambar 3.24 Tampilan Keranjang Belanja.

3.4.7 Tampilan Pembayaran

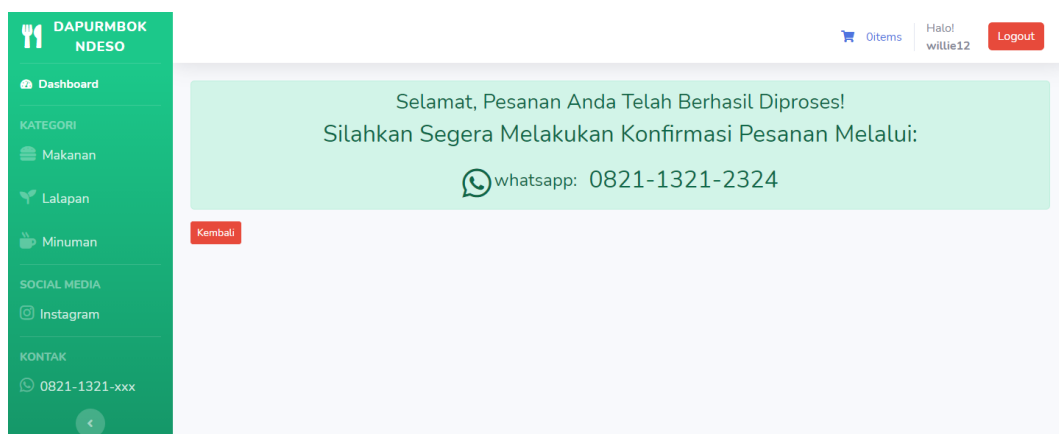
Halaman ini berguna untuk *user* melakukan pembayaran dengan mengisi data diri berupa nama lengkap, alamat, nomor telfon, kemudian memilih jasa pengiriman dan metode pembayaran. Rancangan tampilan pembayaran terdapat pada gambar 3.25.



Gambar 3.25 Tampilan Halaman Pembayaran.

3.4.8 Tampilan Proses Pembayaran

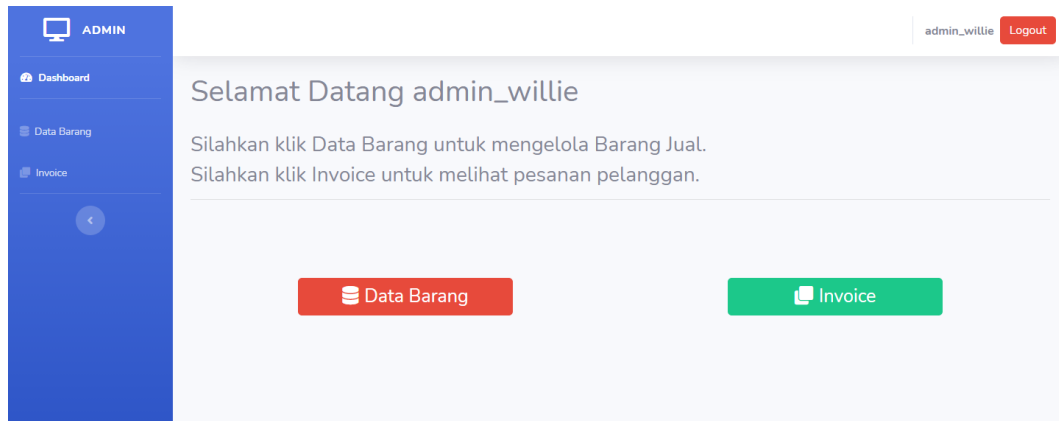
Halaman ini berguna untuk memberi tahu *user* bahwa pesanan sudah berhasil di proses dan akan memberikan kontak yang bisa langsung dihubungi untuk konfirmasi. Rancangan tampilan proses pembayaran terdapat pada gambar 3.26.



Gambar 3.26 Tampilan Halaman Proses Pembayaran.

3.4.9 Tampilan Dashboard Admin

Halaman ini merupakan tampilan utama untuk halaman admin ketika sudah login. Pada halaman ini terdapat fungsi yang hanya dapat diakses oleh admin yaitu data barang dan invoice. Rancangan tampilan dashboard admin terdapat pada gambar 3.27.



Gambar 3.27 Tampilan Halaman Dashboard Admin.

3.4.10 Tampilan Data Barang

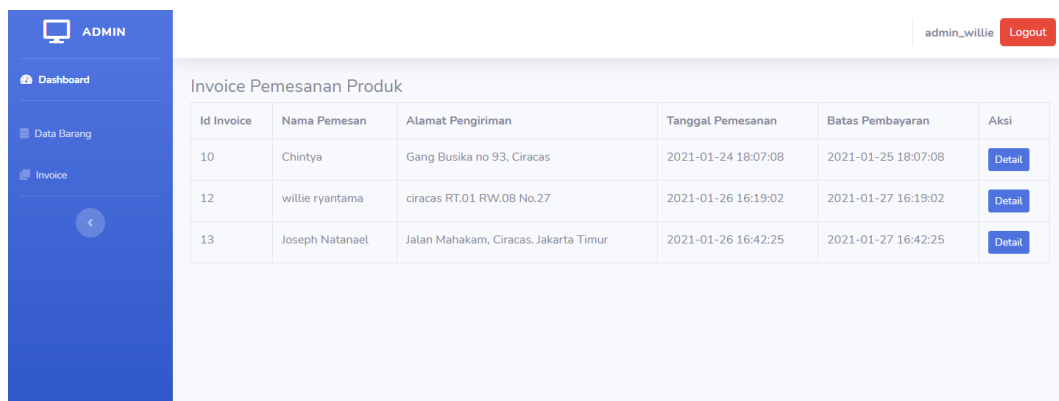
Halaman ini berguna untuk admin agar dapat melihat list barang, detail barang, menambah, mengedit dan menghapus data barang yang dijual di “DapurMbokNdeso”. Rancangan tampilan data barang terdapat pada gambar 3.28.



Gambar 3.28 Tampilan Halaman Data Barang.

3.4.11 Tampilan Invoice

Halaman ini berfungsi untuk admin agar dapat melihat data pemesan dan data pesanan. Untuk data pemesan, data yang ditunjukkan terdiri dari nama dan alamat, kemudian untuk data pesanan data yang ditunjukkan terdiri dari tanggal pemesanan dan batas pembayaran. Rancangan tampilan invoice terdapat pada gambar 3.29.

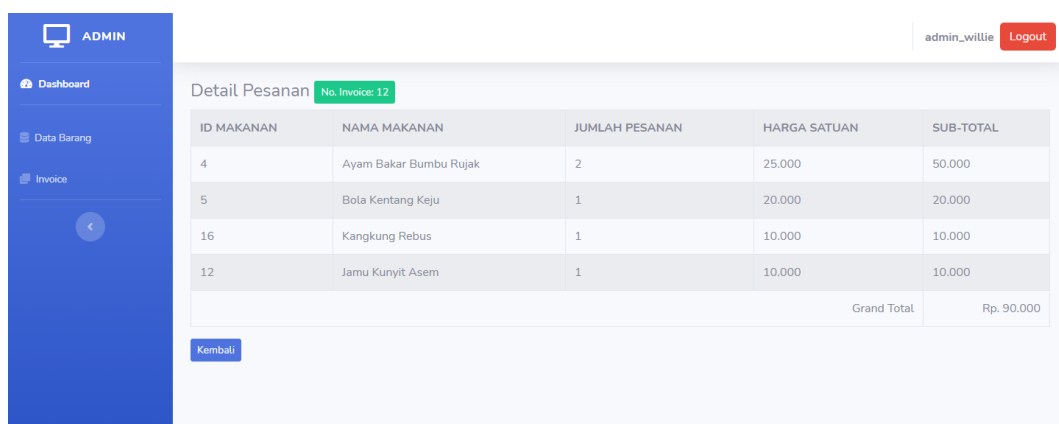


| Invoice Pemesanan Produk | | | | | |
|--------------------------|-----------------|---------------------------------------|---------------------|---------------------|------------------------|
| Id Invoice | Nama Pemesan | Alamat Pengiriman | Tanggal Pemesanan | Batas Pembayaran | Aksi |
| 10 | Chintya | Gang Busika no 93, Ciracas | 2021-01-24 18:07:08 | 2021-01-25 18:07:08 | Detail |
| 12 | willie ryantama | ciracas RT.01 RW.08 No.27 | 2021-01-26 16:19:02 | 2021-01-27 16:19:02 | Detail |
| 13 | Joseph Natanael | Jalan Mahakam, Ciracas, Jakarta Timur | 2021-01-26 16:42:25 | 2021-01-27 16:42:25 | Detail |

Gambar 3.29 Tampilan Halaman Invoice.

3.4.12 Tampilan Detail Invoice

Halaman ini berguna untuk admin agar dapat melihat lebih detail lagi pesanan para pemesan. Data yang ditunjukkan pada detail invoice ini terdiri dari, nomor invoice, id makanan, nama makanan, jumlah pesanan, harga satuan, sub-total dari jumlah pesanan dikalikan dengan harga satuan, dan grand total. Rancangan tampilan detail invoice terdapat pada gambar 3.30.



| Detail Pesanan No. Invoice: 12 | | | | |
|---|------------------------|----------------|--------------|------------|
| ID MAKANAN | NAMA MAKANAN | JUMLAH PESANAN | HARGA SATUAN | SUB-TOTAL |
| 4 | Ayam Bakar Bumbu Rujak | 2 | 25.000 | 50.000 |
| 5 | Bola Kentang Keju | 1 | 20.000 | 20.000 |
| 16 | Kangkung Rebus | 1 | 10.000 | 10.000 |
| 12 | Jamu Kunyit Asem | 1 | 10.000 | 10.000 |
| Grand Total | | | | Rp. 90.000 |

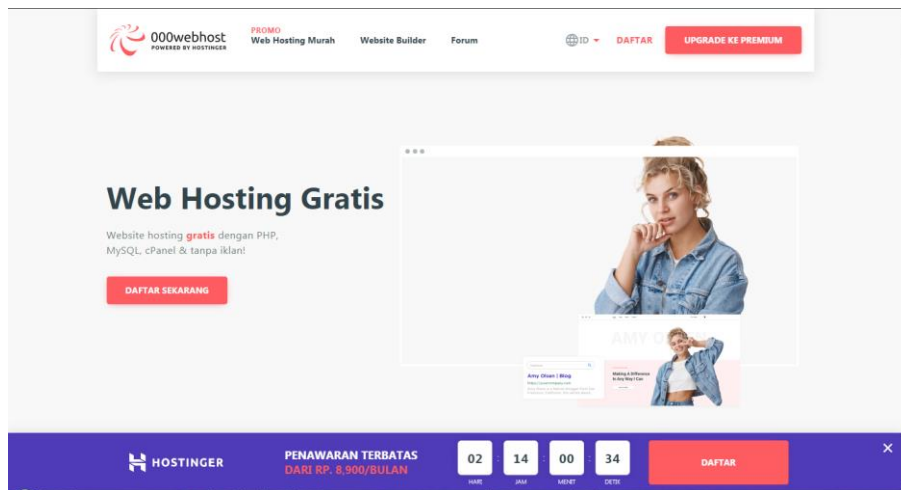
Gambar 3.30 Tampilan Halaman Detail Invoice.

3.4.13 Implementasi Hosting

Pada tahap ini penulis akan men-*deploy* web “DapurMbokNdeso” yang sudah selesai di buat ke salah satu penyedia web hosting gratis bernama 000webhost. Hosting atau web hosting adalah tempat dimana file dan data yang diperlukan website disimpan serta dapat diakses dan dikelola melalui internet. Berikut ini adalah langkah-langkahnya.

1) Daftar Akun

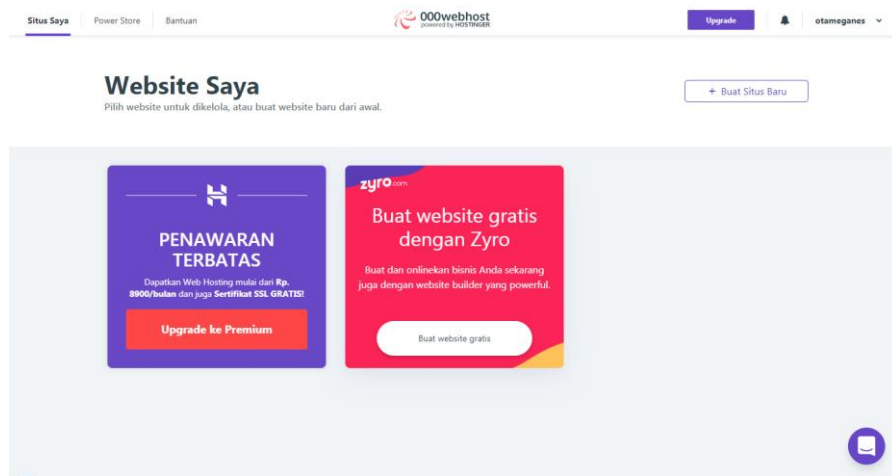
Pada tahap ini dilakukan pendaftaran akun dengan cara menekan tombol “daftar sekarang” dan masukkan e-mail serta password.



Gambar 3.31 Tampilan Utama 000webhost.

2) Buat Website Baru

Tahap selanjutnya yaitu membuat website baru, ketika sudah selesai mendaftar maka akan ditujukan pada halaman khusus yang hanya bisa diakses oleh akun yang bersangkutan untuk membuat website baru, berikut ini adalah tampilan kelola website yang terdapat pada gambar 3.32. Klik tombol “Buat Situs Baru” yang terdapat pada gambar 3.33.



Gambar 3.32 Tampilan Kelola Website.

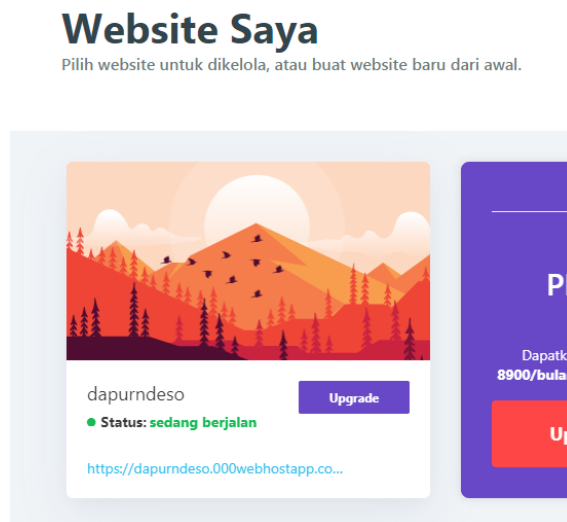


Gambar 3.33 Tombol Untuk Membuat Situs Baru.

Kemudian masukkan nama website dan passwordnya.

Gambar 3.34 Memasukkan Nama Website dan Password

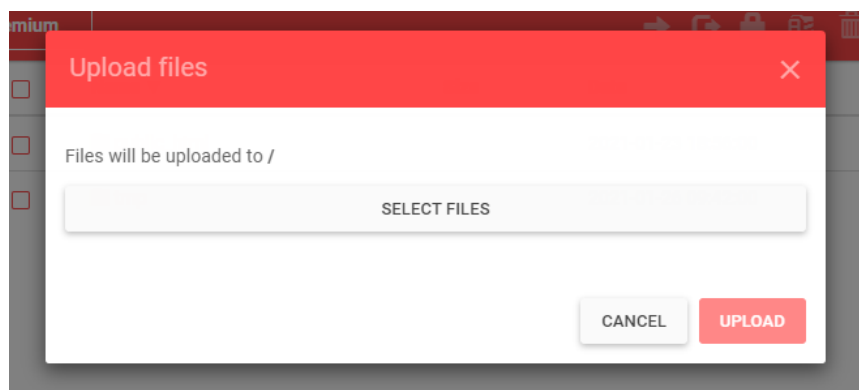
Setelah itu akan muncul website baru yang sudah dibuat.



Gambar 3.35 Website Yang Sudah Dibuat.

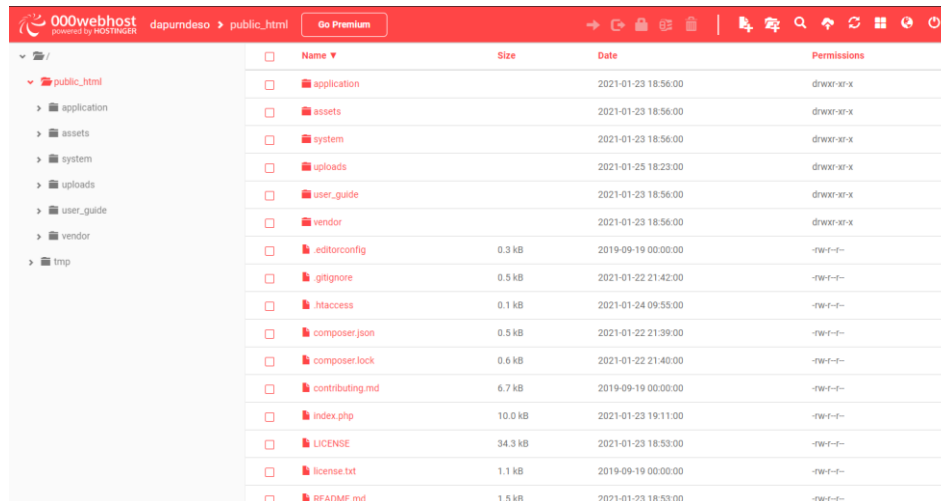
3) Mengupload File Website

Pada tahap ini penulis akan mengupload file-file website “DapurMbokNdeso” melalui cpanel. Masuk ke halaman <https://files.000webhost.com/> dan kemudian upload file.



Gambar 3.36 Pop-up Upload Files.

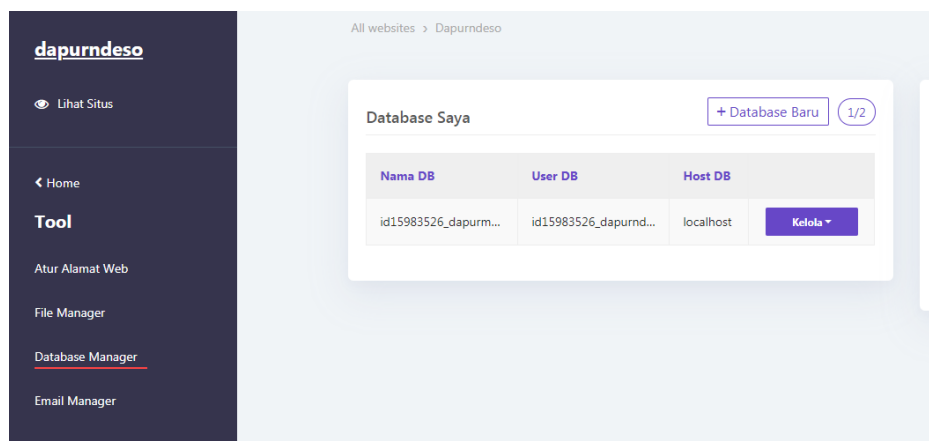
Setelah berhasil upload maka file-file website yang diperlukan sudah tersimpan kedalam web hosting.



Gambar 3.37 Tampilan File Yang Terupload.

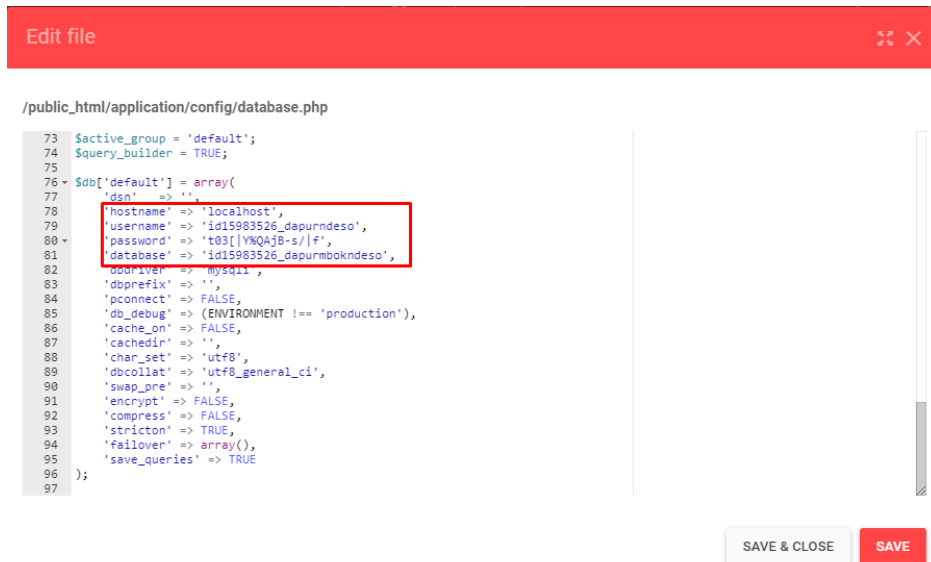
4) Mengelola database

Pada tahap ini penulis akan meng-*import* database, pada halaman kelola website pilih Tool > Database Manager. Buat sebuah database baru.



Gambar 3.38 Database Baru.

Kemudian konfigurasi file database.php yang terdapat pada folder config. Masukkan nama database, username database, dan password yang sudah dibuat.



Gambar 3.39 Konfigurasi Database.

3.5 Uji Coba

Tahap pengujian sistem merupakan tahapan yang dilakukan setelah tahap implementasi telah dilakukan. Pengujian yang pertama kali dilakukan pada web server lokal xampp sebagai sarana evaluasi sehingga dapat diketahui letak kekurangan dan kesalahan yang mungkin dapat terjadi didalamnya.

Hal pertama untuk menguji website ini adalah mengaktifkan server pada 000webhostapp. Kemudian ketikkan alamat file domain sesuai dengan yang ada pada website 000webhost yang penulis sudah buat pada 3.4.13, seperti <https://dapurndeso.000webhostapp.com/welcome> pada browser.

Pengujian sistem difokuskan pada fungsi dari sistem untuk melihat apakah sistem yang telah dirancang sebelumnya akan bekerja sebagaimana mestinya. Pengujian dinyatakan berhasil jika sistem mampu menjalankan fungsi dengan benar. Sebaliknya, pengujian dianggap gagal jika ada fungsi perangkat lunak yang tidak mampu untuk dijalankan. Tahap pengujian berdasarkan analisis dan desain yang telah dilakukan sebelumnya. Pada tahap pengujian ini, akan menguji fungsionalitas dari sistem yang dibangun.

Uji coba dilakukan dengan pengujian interface yang bertujuan untuk mengetahui apakah setiap tombol dan masing-masing menu berjalan dengan baik seperti yang diharapkan.

Tabel 3.8 Tabel Pengujian Form User

| No | Uji Kasus | Hasil yang diharapkan | Hasil Uji |
|----|----------------------------|---|-----------|
| 1 | Dashboard | ketika mengakses web, akan menampilkan halaman awal dashboard. | Berhasil |
| 2 | Halaman Daftar Makanan | Ketika user mengklik daftar makanan, maka akan menampilkan halaman yang berisi list makanan. | Berhasil |
| 3 | Halaman Daftar Lalapan | Ketika user mengklik daftar lalapan, maka akan menampilkan halaman yang berisi list lalapan. | Berhasil |
| 4 | Halaman Daftar Minuman | Ketika user mengklik daftar minuman, maka akan menampilkan halaman yang berisi list minuman. | Berhasil |
| 5 | Tombol Tambah Ke Keranjang | Ketika user mengklik tombol Tambah Ke Keranjang pada produk, maka produk tersebut akan langsung masuk ke dalam keranjang belanja. | Berhasil |
| 6 | Tombol Keranjang Belanja | Ketika user mengklik tombol keranjang belanja, maka user akan diarahkan ke halaman keranjang belanja, dan menampilkan pesanan yang sudah ditambahkan oleh user. | Berhasil |
| 7 | Tombol Lanjut Belanja | Ketika user mengklik tombol lanjut belanja, maka user akan | Berhasil |

| | | | |
|----|------------------------|--|----------|
| | | diarahkan ke halaman dashboard. | |
| 8 | Tombol Hapus Keranjang | Ketika user mengklik tombol hapus keranjang, maka keranjang belanja user akan dikosongkan sepenuhnya. | Berhasil |
| 9 | Tombol Pembayaran | Ketika user mengklik tombol pembayaran, maka user akan diarahkan ke halaman pembayaran untuk mengisi data. | Berhasil |
| 10 | Tombol Pesan | Ketika user mengklik tombol pesan, maka pesanan akan masuk ke dalam sistem. | Berhasil |
| 11 | Tombol Detail | Ketika user mengklik tombol detail pada produk, maka user akan diarahkan ke halaman detail produk. | Berhasil |
| 12 | Tombol Login | Ketika user mengklik tombol login, maka user akan diarahkan ke form login | Berhasil |
| 13 | Daftar akun | Ketika user mengklik daftar akun, maka user akan diarahkan ke form daftar akun | Berhasil |
| 14 | Tombol Logout | Ketika user mengklik tombol logout, maka user akan keluar dari akun tersebut dan diarahkan ke form login | Berhasil |

Tabel 3.9 Tabel Pengujian Form Admin

| No | Uji Kasus | Hasil yang diharapkan | Hasil Uji |
|----|--------------------------------------|--|-----------|
| 1 | Login Admin | Ketika login sebagai admin, maka akan menampilkan halaman awal dashboard admin | Berhasil |
| 2 | Form Data Barang | Ketika admin mengklik data barang, maka admin akan ditujukan ke halaman data barang disertai dengan button detail, edit dan hapus. | Berhasil |
| 3 | Form invoice | Ketika admin mengklik invoice, maka admin akan ditujukan ke halaman invoice disertai dengan tombol detail invoice | Berhasil |
| 4 | Tombol Tambah Barang | Ketika admin mengklik tombol tambah barang, maka akan memunculkan pop up yang menampilkan pengisian produk yaitu nama produk, keterangan, kategori, harga, stok, dan gambar produk | Berhasil |
| 5 | Tombol Batal pada Form Tambah Barang | Ketika admin mengklik button batal, maka pengisian tambah barang akan dibatalkan | Berhasil |
| 6 | Tombol Simpan | ketika admin mengklik tombol simpan setelah mengisikan data produk yang dibutuhkan, maka produk akan tersimpan kedalam database. | Berhasil |

| | | | |
|----|-----------------------|---|----------|
| 7 | Tombol Detail Produk | Ketika admin mengklik tombol detail, maka admin akan diarahkan ke halaman detail produk yang dipilih. | Berhasil |
| 8 | Tombol Detail Invoice | Ketika admin mengklik tombol detail invoice, maka admin akan melihat detail pesanan pemesan. | Berhasil |
| 9 | Tombol Edit Produk | Ketika admin mengklik tombol edit, maka admin akan diarahkan ke halaman edit barang yang dipilih. | Berhasil |
| 10 | Tombol Hapus Produk | Ketika admin mengklik tombol hapus produk, maka produk akan terhapus dari database. | Berhasil |

BAB 4

PENUTUP

4.1 Kesimpulan

Berdasarkan hasil penulisan ilmiah yang diperoleh, maka dapat ditarik kesimpulan bahwa aplikasi situs web toko makanan “DapurMbokNdeso” telah berhasil dirancang dan dibuat dengan menggunakan PHP, database MySQL serta framework CodeIgniter. Terdapat 2 hak akses, Admin dan User. Web ini telah berhasil di hosting dengan alamat <https://dapurndeso.000webhostapp.com>.

Berdasarkan hasil uji coba dapat disimpulkan bahwa semua tombol dan fungsi dapat berjalan dengan benar. Pengelolaan konten yang dilakukan oleh admin berjalan dengan baik dan transaksi pemesanan dan pembelian berhasil dilakukan.

Aplikasi ini juga membuat fitur pemesanan makanan dengan metode pembayaran COD(*Cash On Delivery*). Aplikasi ini dapat membantu pelanggan untuk memesan menu tanpa harus datang ke lokasi toko makanan. Aplikasi ini juga memudahkan penjual untuk memperlihatkan menu-menu yang dijual dan memberikan detailnya kepada pelanggan.

4.2 Saran

Saran yang diberikan untuk pengembangan selanjutnya diharapkan aplikasi ini juga bisa dikembangkan menjadi aplikasi mobile dan diharapkan kedepannya pembayaran bisa melalui *e-wallet* dan juga memberikan fitur kepada pelanggan agar dapat melakukan *tracking* pesanan.

DAFTAR PUSTAKA

- [1]. Waryanto, 2018 Apa itu website?
<https://www.niagahoster.co.id/blog/pengertian-website/> diakses pada tanggal 7 Mei 2020.
- [2]. Anonim, 2018 Jenis-jenis website <https://fastwork.id/blog/jenis-jenis-website/> diakses pada tanggal 7 Mei 2020.
- [3]. Abdullah, Rohi. 2018. 7 in 1 Pemograman Web Untuk Pemula. Jakarta.
- [4]. Situseo, 2017 Manfaat website <https://www.situseo.com/manfaat-website/> diakses pada tanggal 7 Mei 2020.
- [5]. Mustofa 2018 Sejarah dan Perkembangan Website
<https://dimbleweb.com/blog/read/sejarah-dan-perkembangan-website/124> diakses pada tanggal 7 Mei 2020.
- [6]. Sugeng,S. 2018. E-Commerce Sebagai Pendukung Pemasaran. Jakarta.
- [7]. FasaPay Administrator. Jenis-jenis E-Commerce. <https://blog.fasapay.id/jenis-jenis-e-commerce/>. Diakses pada 20 November 2020.
- [8]. Aco, Ambo dan Hutami, Andi. 2017. Analisis Bisnis E-Commerce pada Mahasiswa Universitas Islam Negeri Alauddin Makassar. Makassar.
- [9]. Riyanto. 2011. Microsoft sql server dengan PostgreSQL. Jakarta.
- [10]. Edy, Ahmad. Membuat file CSS didalam halaman WEB. <http://ahmad-edy.web.ugm.ac.id/materi.htm> . Diakses pada 15 Agustus 2020.
- [11]. Simarmata, J. 2014. Menggunakan PHP dan MySQL. Penerbit Andi: Yogyakarta.
- [12]. Abdullah, Rohi. 2018. 7 in 1 Pemograman Web Untuk Pemula. Jakarta.
- [13]. Fachrul, S. 2018. E-Commerce: Pengembangan Model Perniagaan Menggunakan Internet. Jakarta.

- [14]. Peni,Sri dan Eka, Bambang. 2013. Pembuatan Website E-Commerce Pada Distro Java Trend. Solo.
- [15]. Nurrahman, M dan Ahmad, L. 2018. Rancang Bangun Sistem Informasi Praktek Kerja Lapangan (PKL). Mataram.
- [16]. Anonim. 2017. Mengetahui Fungsi Kelebihan dan Kekurangan Code Igniter. <https://idcloudhost.com/panduan/mengetahui-fungsi-kelebihan-dan-kekurangan-codeigniter/> Diakses pada tanggal 17 November 2020.
- [17]. Setiawati, Popong. 2018. Analisa Dan Perancangan Sistem Informasi Penyedia Lowongan Pekerjaan Yang Direkomendasi Berdasarkan Standar Kompetensi Kerja Nasional Indonesia (SKKNI). JIK: Jurnal Ilmu Komputer Volume 3(2). Jakarta.
- [18]. Sugiarto, R. 2015. Implementasi MVC Pada Situs Portal Pencarian Universitas Di Daerah Istimewa Yogyakarta. Yogyakarta.
- [19]. Reyza, 2017. Visual Studio Code & MySQL <https://www.rezafaisal.net/visual-studio-code-&-MySql> diakses pada tanggal 2 Desember 2020.
- [20]. Mutiara Arumsari, 2018. Fitur-fitur microsoft visual studio code <https://www.dicoding.com/blog/microsoft-visual-studio-code/> diakses pada tanggal 2 Desember 2020.

LAMPIRAN

header.php

```
<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>DapurMbokNdeso</title>

    <!-- Custom fonts for this template-->
    <link href="<?php echo base_url() ?>assets/vendor/fontawesome-
free/css/all.min.css" rel="stylesheet" type="text/css">
    <link href="https://fonts.googleapis.com/css?family=Nunito:200,200i,30
0,300i,400,400i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">

    <!-- Custom styles for this template-->
    <link href="<?php echo base_url() ?>assets/css/sb-admin-
2.min.css" rel="stylesheet">

</head>
```

Sidebar.php

```
<body id="page-top">

    <!-- Page Wrapper -->
    <div id="wrapper">

        <!-- Sidebar -->
        <ul class="navbar-nav bg-gradient-success sidebar sidebar-
dark accordion" id="accordionSidebar">

            <!-- Sidebar - Brand -->
            <a class="sidebar-brand d-flex align-items-center justify-content-
center" href="<?php base_url('welcome') ?>">
                <div class="sidebar-brand-icon">
                    <i class="fas fa-utensils"></i>
                </div>
                <div class="sidebar-brand-text mx-3" style="font-
size:18px;">DapurMbok<br>Ndeso</div>
            </a>

            <!-- Divider -->
            <hr class="sidebar-divider my-0">

            <!-- Nav Item - Dashboard -->
            <li class="nav-item active">
                <a class="nav-link" href="<?php echo base_url('welcome') ?>">
                    <i class="fas fa-fw fa-tachometer-alt"></i>
                    <span style="font-size:16px;">Dashboard</span></a>
            </li>

            <!-- Divider -->
            <hr class="sidebar-divider">

            <!-- Heading -->
            <div class="sidebar-heading" style="font-size: 15px;">
                Kategori
            </div>
```

```

<!-- Nav Item - Tables -->
<li class="nav-item">
    <a class="nav-
link" href="<?php echo base_url('kategori/makanan') ?>">
        <i class="fas fa-hamburger" style="font-size: 22px;"></i>
        <span style="font-size:18px;">Makanan</span></a>
    </li>

<li class="nav-item">
    <a class="nav-
link" href="<?php echo base_url('kategori/lalapan') ?>">
        <i class="fas fa-seedling" style="font-size: 22px;"></i>
        <span style="font-size:18px;">Lalapan</span></a>
    </li>

<li class="nav-item">
    <a class="nav-
link" href="<?php echo base_url('kategori/minuman') ?>">
        <i class="fas fa-mug-hot" style="font-size: 22px;"></i>
        <span style="font-size:18px;">Minuman</span></a>
    </li>

<!-- Divider -->
<hr class="sidebar-divider d-none d-md-block">

<div class="sidebar-heading" style="font-size: 15px;">
    Social Media
</div>

<li class="nav-item" >
    <a class="nav-
link" href="https://www.instagram.com/dapurmbokndeso/" target="_blank">
        <i class="fab fa-instagram fa-10x" style="font-size: 22px;"></i>
        <span style="font-size: 18px;">Instagram</span></a>
    </li>

<!-- Sidebar Toggler (Sidebar) -->
<div class="text-center d-none d-md-inline">

```

```

        <button class="rounded-circle border-
0" id="sidebarToggle"></button>
    </div>

</ul>
<!-- End of Sidebar -->

<!-- Content Wrapper -->
<div id="content-wrapper" class="d-flex flex-column">

    <!-- Main Content -->
    <div id="content">

        <!-- Topbar -->
        <nav class="navbar navbar-expand navbar-light bg-
white topbar mb-4 static-top shadow">

            <!-- Sidebar Toggle (Topbar) -->
            <button id="sidebarToggleTop" class="btn btn-link d-md-
none rounded-circle mr-3">
                <i class="fa fa-bars"></i>
            </button>

            <!-- Topbar Navbar -->
            <ul class="navbar-nav ml-auto">

                <!-- Nav Item - Search Dropdown (Visible Only XS) -->
                <li class="nav-item dropdown no-arrow d-sm-none">
                    <a class="nav-link dropdown-
toggle" href="#" id="searchDropdown" role="button" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                        <i class="fas fa-search fa-fw"></i>
                    </a>
                    <!-- Dropdown - Messages -->
                    <div class="dropdown-menu dropdown-menu-right p-
3 shadow animated--grow-in" aria-labelledby="searchDropdown">
                        <form class="form-inline mr-auto w-100 navbar-search">
                            <div class="input-group">

```

```

        <input type="text" class="form-control bg-
light border-0 small" placeholder="Search for..." aria-
label="Search" aria-describedby="basic-addon2">
        <div class="input-group-append">
            <button class="btn btn-primary" type="button">
                <i class="fas fa-search fa-sm"></i>
            </button>
        </div>
    </div>
</form>
</div>
</li>

<div class="navbar">
    <ul class="nav navbar-nav navbar-right">
        <li>
            <?php
                $keranjang = '<i class="fa fa-shopping-cart mr-
3" aria-hidden="true"></i>'. $this->cart->total_items(). 'items'
                ?>

            <?php echo anchor('dashboard/detail_keranjang', $keran
jang) ?>
        </li>
    </ul>

    <div class="topbar-divider d-none d-sm-block"></div>

    <ul class="nav navbar-nav navbar-right">
        <?php if($this->session->userdata('username')) { ?>
            <li><div>Halo!<br><strong><?php echo $this->session-
>userdata('username') ?></strong></div></li>
            <li class="ml-4 mt-
1"><a href="<?php echo base_url('auth/logout')?>" class="btn btn-md btn-
danger">Logout</a></li>
        <?php } else { ?>
            <li><a href="<?php echo base_url('auth/login') ?>" c
lass="btn btn-md btn-primary">Login</a></li>
        <?php } ?>

```

```

        </ul>

    </div>

</nav>
<!-- End of Topbar -->

```

Footer.php

```

<!-- Bootstrap core JavaScript-->
<script src="<?php echo base_url() ?>assets/vendor/jquery/jquery.min.js"
></script>
<script src="<?php echo base_url() ?>assets/vendor/bootstrap/js/bootstrap
p.bundle.min.js"></script>

<!-- Core plugin JavaScript-->
<script src="<?php echo base_url() ?>assets/vendor/jquery-
easing/jquery.easing.min.js"></script>

<!-- Custom scripts for all pages-->
<script src="<?php echo base_url() ?>assets/js/sb-admin-
2.min.js"></script>

<!-- Page level plugins -->
<script src="<?php echo base_url() ?>assets/vendor/chart.js/Chart.min.js
"></script>

<!-- Page level custom scripts -->
<script src="<?php echo base_url() ?>assets/js/demo/chart-area-
demo.js"></script>
<script src="<?php echo base_url() ?>assets/js/demo/chart-pie-
demo.js"></script>

</body>

</html>

```

View/dashboard.php

```
<div class="container-fluid">
    <div class="row text-center ml-1 mr-1 mb-3" style="border-
bottom: 5px solid #f0c59c;">
        <p style="color: #a16428; font-family: 'Quicksand', sans-
serif; font-weight: 100; font-
size: 42px;">Selamat Datang di DapurMbokNdeso</p>
    </div>
    <div id="carouselExampleIndicators" class="carousel slide" data-
ride="carousel">
        <ol class="carousel-indicators">
            <li data-target="#carouselExampleIndicators" data-slide-
to="0" class="active"></li>
            <li data-target="#carouselExampleIndicators" data-slide-
to="1"></li>
            <li data-target="#carouselExampleIndicators" data-slide-
to="2"></li>
            <li data-target="#carouselExampleIndicators" data-slide-
to="3"></li>
            <li data-target="#carouselExampleIndicators" data-slide-
to="4"></li>
            <li data-target="#carouselExampleIndicators" data-slide-
to="5"></li>
        </ol>
        <div class="carousel-inner">
            <div class="carousel-item active">
                
            </div>
            <div class="carousel-item">
                
            </div>
            <div class="carousel-item">
                
            </div>
            <div class="carousel-item">
```

```

        
    </div>
    <div class="carousel-item">
        
    </div>
    <div class="carousel-item">
        
    </div>
</div>
<a class="carousel-control-
prev" href="#carouselExampleIndicators" role="button" data-slide="prev">
    <span class="carousel-control-prev-icon" aria-
hidden="true"></span>
    <span class="sr-only">Previous</span>
</a>
<a class="carousel-control-
next" href="#carouselExampleIndicators" role="button" data-slide="next">
    <span class="carousel-control-next-icon" aria-
hidden="true"></span>
    <span class="sr-only">Next</span>
</a>
</div>

<div class="row text-center mt-5 ml-1 mr-1">
    <div class="mt-4 text-left" style="width: 85%; margin:0 auto;">
        <h2 style="color:#a16428; letter-
spacing: 2px;">Tentang Kami</h2>
        <div style="display: flex; justify-content: space-
between;">
            <p style="font-size: 19px; margin-
top: 8%;">DapurMbokNdeso menyajikan makanan dan minuman khas pedesaan.<b
r>
                Dibuat dengan rempah-
rempah yang diproduksi alami di pedesaan.</p>

```



```

        
    </div>
    <h2 class="mt-5" style="text-
align: center; color:#a16428; letter-spacing: 2px;">Kualitas</h2>
    <div style="display: flex; border-
top: 2px solid #a16428; height: 35vh; flex-direction: column; align-
items: center;">
        <div class="mt-5" style="display: flex; align-
items: center; width: 85%; justify-content: space-between;">
            
            
            
        </div>
        <div class="mt-3" style="display: flex; align-
items: center; width: 90%; justify-content: space-between;">
            <h3>Homemade</h3>
            <h3 class="ml-5">Bahan Alami</h3>
            <h3>Tanpa Pengawet</h3>
        </div>
    </div>
    <h2 style="text-align: center; color:#a16428;">Menu</h2>
    <div style="display: flex; height: 30vh; align-
items: center; border-top: 2px solid #a16428; margin: 0 auto; justify-
content: space-around;">
        <a href="<?php echo base_url('kategori/makanan') ?>
" style="width: 25%; font-size: 25px; color: white;" class="btn btn-
md btn-primary"><i class="fas fa-drumstick-bite mr-2"></i>Makanan</a>
        <a href="<?php echo base_url('kategori/lalapan') ?>
" style="width: 25%; font-size: 25px; color: white;" class="btn btn-
md btn-success"><i class="fas fa-seedling"></i>Lalapan</a>
        <a href="<?php echo base_url('kategori/minuman') ?>
" style="width: 25%; font-size: 25px; color: white;" class="btn btn-
md btn-danger"><i class="fas fa-mug-hot"></i>Minuman</a>

```

```

        </div>
    </div>
</div>
</div>

```

View/detail_barang.php

```

<div class="container-fluid">
    <div class="card">
        <h5 class="card-header">Detail Makanan</h5>
        <div class="card-body">

            <?php foreach($barang as $brg):

                ?>
                <div class="row">
                    <div class="col-md-4">
                        
                    </div>
                    <div class="col-md-8">
                        <table class="table">
                            <tr>
                                <td>Nama Produk</td>
                                <td><strong><?php echo $brg-
>nama_mkn ?></strong></td>
                            </tr>

                            <tr>
                                <td>Keterangan</td>
                                <td><strong><?php echo $brg-
>keterangan ?></strong></td>
                            </tr>

                            <tr>
                                <td>Kategori</td>
                                <td><strong><?php echo $brg-
>kategori ?></strong></td>

```

```

        </tr>

        <tr>
            <td>Stok</td>
            <td><strong><?php echo $brg-
>stok ?></strong></td>
        </tr>

        <tr>
            <td>Harga</td>
            <td><strong><div class="btn btn-sm btn-
success">Rp. <?php echo number_format($brg-
>harga,0,',','.') ?></div></strong></td>
        </tr>
    </table>
    <?php echo anchor('dashboard/tambah_ke_keranjang/'.$brg-
>id_mkn, '<div class="btn btn-sm btn-
primary">Tambah ke Keranjang</div>') ?>

    </div>
</div>
<?php endforeach ?>
</div>
</div>
</div>

```

View/form_login.php

```

<!DOCTYPE html>
<html lang="en">

<body class="bg-gradient-primary">

    <div class="container">

        <!-- Outer Row -->
        <div class="row justify-content-center">

```

```

<div class="col-xl-5 col-lg-12 col-md-9">

    <div class="card o-hidden border-0 shadow-lg my-5">
        <div class="card-body p-0">
            <!-- Nested Row within Card Body -->
            <div class="row">

                <div class="col-lg-12">
                    <div class="p-5">
                        <div class="text-center">
                            <h1 class="h4 text-gray-900 mb-4">Login</h1>
                        </div>
                        <?php echo $this->session->flashdata('pesan') ?>
                        <form method="post" action="<?php echo base_url('auth/
login') ?>" class="user">
                            <div class="form-group">
                                <input type="text" class="form-control form-
control-user" id="exampleInputEmail" aria-
describedby="emailHelp" placeholder="Enter Username" name="username">
                                <?php echo form_error('username', '<div class="tex
t-danger small ml-2">', '</div>'); ?>
                            </div>
                            <div class="form-group">
                                <input type="password" class="form-control form-
control-
user" id="exampleInputPassword" placeholder="Password" name="password">
                                <?php echo form_error('password', '<div class="tex
t-danger small ml-2">', '</div>'); ?>
                            </div>

                                <button type="submit" class="btn btn-primary form-
control">Login</button>

                                <hr>
                                <a href="<?php echo base_url('welcome') ?>" class="b
tn btn-success form-control mt-2">Kembali ke Dashboard</a>

                            </form>
                        <hr>

```



```

        <td><?php echo $no++ ?></td>
        <td><?php echo $items['name'] ?></td>
        <td><?php echo $items['qty'] ?></td>
        <td align="right">Rp. <?php echo number_format($items['price
'], 0, ',', '.') ?></td>
        <td align="right">Rp.<?php echo number_format($items['subtot
al'] , 0, ',', '.')?></td>
    </tr>

    <?php endforeach ?>
    <tr>
        <td colspan="4"></td>
        <td align="right">Rp. <?php echo number_format($this->cart-
>total(), 0, ',', '.') ?></td>
    </tr>
</table>

<div align="right">
    <a href="<?php echo base_url('dashboard/hapus_keranjang') ?>
">

    <div class="btn btn-sm btn-danger">Hapus Keranjang</div></a>
    <a href="<?php echo base_url('dashboard/index') ?>">
    <div class="btn btn-sm btn-primary">Lanjut Belanja</div></a>
    <a href="<?php echo base_url('dashboard/pembayaran') ?>">
    <div class="btn btn-sm btn-success">Pembayaran</div></a>

</div>
</div>

```

View/lalapan.php

```
<div class="container-fluid">
  <div class="row text-center mt-5">

    <?php foreach ($lalapan as $brg) : ?>
      <div class="card ml-3 mt-3" style="width: 25%; ">
        
        <div class="card-body">
          <h5 class="card-title mb-1"><?php echo $brg->nama_mkn ?></h5>
          <small><?php echo $brg->keterangan ?></small>
          <br>
          <span class="badge badge-success mb-3">Rp. <?php echo number_format($brg->harga, 0, ',', '.') ?></span>
          <br>
          <?php echo anchor('dashboard/tambah_ke_keranjang/' . $brg->id_mkn, '
          <div class="btn btn-sm btn-primary">Tambah ke Keranjang</div>') ?>

          <?php echo anchor('dashboard/detail/' . $brg->id_mkn, '
          <div class="btn btn-sm btn-success">Detail</div>') ?>
        </div>
      </div>
    <?php endforeach; ?>
  </div>
</div>
```

View/makanan.php

```
<div class="container-fluid">
    <div class="row text-center mt-5">

        <?php foreach ($makanan as $brg) : ?>
            <div class="card ml-3 mt-3" style="width: 25%; ">
                
                <div class="card-body">
                    <h5 class="card-title mb-1"><?php echo $brg->nama_mkn ?></h5>
                    <small><?php echo $brg->keterangan ?></small>
                    <br>
                    <span class="badge badge-success mb-3">Rp. <?php echo number_format($brg->harga, 0, ',', '.') ?></span>
                    <br>
                    <?php echo anchor('dashboard/tambah_ke_keranjang/' . $brg->id_mkn, '
<div class="btn btn-sm btn-primary">Tambah ke Keranjang</div>') ?>

                    <?php echo anchor('dashboard/detail/' . $brg->id_mkn, '
<div class="btn btn-sm btn-success">Detail</div>') ?>
                </div>
            </div>
        <?php endforeach; ?>
    </div>
</div>
```


View/minuman.php

```
<div class="container-fluid">
  <div class="row text-center mt-5">

    <?php foreach ($minuman as $brg) : ?>
      <div class="card ml-3 mt-3" style="width: 25%; ">
        
        <div class="card-body">
          <h5 class="card-title mb-1"><?php echo $brg->nama_mkn ?></h5>
          <small><?php echo $brg->keterangan ?></small>
          <br>
          <span class="badge badge-success mb-3">Rp. <?php echo number_format($brg->harga, 0, ',', '.') ?></span>
          <br>
          <?php echo anchor('dashboard/tambah_ke_keranjang/' . $brg->id_mkn, '
          <div class="btn btn-sm btn-primary">Tambah ke Keranjang</div>') ?>

          <?php echo anchor('dashboard/detail/' . $brg->id_mkn, '
          <div class="btn btn-sm btn-success">Detail</div>') ?>
          </div>
        </div>
      <?php endforeach; ?>
    </div>
  </div>
```

View/pembayaran.php

```
<div class="container-fluid">
  <div class="row">
    <div class="col-md-2"></div>
    <div class="col-md-8">
      <div class="btn btn-sm btn-success">
        <?php
          $grandtotal = 0;
          if($keranjang = $this->cart->contents()){
            foreach($keranjang as $item){
              $grandtotal = $grandtotal + $item['subtotal'];
            }

            echo "<h5>Total Belanja Anda: Rp. ".number_format($g
randtotal,00,',','.'');

            ?>
          </div><br><br>
          <h3>Input Alamat Pengiriman dan Pembayaran</h3>
          <form method="post" action="<?php echo base_url() ?>dashboar
d/proses_pesanan">
            <div class="form-group">
              <label>Nama Lengkap</label>
              <input type="text" name="nama" placeholder="Nama Len
gkap Anda" class="form-control">
            </div>

            <div class="form-group">
              <label>Alamat Lengkap</label>
              <input type="text" name="alamat" placeholder="Alamat
Lengkap Anda" class="form-control">
            </div>

            <div class="form-group">
              <label>Nomor Telfon</label>
              <input type="text" name="no_telp" placeholder="Nomor
Telepon Anda" class="form-control">
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
```

```

<div class="form-group">
    <label>Jasa Pengiriman</label>
    <select class="form-control" name="pengiriman">
        <option value="">JNE</option>
        <option value="">TIKI</option>
        <option value="">POS INDONESIA</option>
        <option value="">GOJEK</option>
        <option value="">GRAB</option>
        <option value="">COD</option>
    </select>
</div>

<div class="form-group">
    <label>Pilih BANK</label>
    <select class="form-control" name="bank">
        <option value="">BCA - XXXXXX</option>
        <option value="">BNI - XXXXXX</option>
        <option value="">BRI - XXXXXX</option>
        <option value="">MANDIRI - XXXXXX</option>
    </select>

    <button type="submit" class="btn btn-sm btn-
primary mt-4">Pesan</button>

</div>
</form>

<?php
}else{
    echo"<h4>Keranjang Belanja Anda Masih Kosong</h4>";
}
?>
</div>

<div class="col-md-2"></div>
</div>
</div>

```

View/proses_pesanan.php

```
<div class="container-fluid">
    <div class="alert alert-success text-center">
        <h3 class="text-center align-
middle">Selamat, Pesanan Anda Telah Berhasil Diproses!</h3>
        <h2>Silahkan Segera Melakukan Konfirmasi Pesanan Melalui:</h2>
        <div style="display: flex; align-
items: center; width: 100%; justify-content:center;" class="mt-4">
            <i class="fab fa-whatsapp" style="font-size: 45px;"></i>
            <h4 class="mr-3 ml-1">whatsapp:</h4>
            <h2>0821-1321-2324</h2>
        </div>
    </div>
    <?php echo anchor('welcome/index/', '<div class="btn btn-sm btn-
danger">Kembali</div>') ?>
</div>
```

View/registrasi.php

```
<!DOCTYPE html>
<html lang="en">

<body class="bg-gradient-primary">

    <div class="container">

        <div class="card o-hidden border-0 shadow-lg col-lg-6 my-5 mx-auto">
            <div class="card-body p-0">
                <!-- Nested Row within Card Body -->
                <div class="row">
                    <div class="col-lg">
                        <div class="p-5">
                            <div class="text-center">
                                <h1 class="h4 text-gray-900 mb-4">Daftar Akun!</h1>
                            </div>
                            <form method="POST" action="<?php echo base_url('registras
i/index') ?>" class="user">
```

```

        <div class="form-group">
            <input type="text" class="form-control form-control-
user" id="exampleInputEmail" placeholder="Nama" name="nama">

            <?php echo form_error('nama','<div class="text-
danger small ml-2">','</div>') ?>
        </div>

        <div class="form-group">
            <input type="text" class="form-control form-control-
user" id="exampleInputEmail" placeholder="Username" name="username">

            <?php echo form_error('username','<div class="text-
danger small ml-2">','</div>') ?>
        </div>
        <div class="form-group row">
            <div class="col-sm-6 mb-3 mb-sm-0">
                <input type="password" class="form-control form-
control-
user" id="exampleInputPassword" placeholder="Password" name="password_1"
>

                <?php echo form_error('password_1','<div class="text
-danger small ml-2">','</div>') ?>
            </div>
            <div class="col-sm-6">
                <input type="password" class="form-control form-
control-
user" id="exampleRepeatPassword" placeholder="Ulangi Password" name="pas
sword_2">

            </div>
        </div>
        <button class="btn btn-primary btn-user btn-
block" type="submit">Daftar Akun</button>
    </form>
    <hr>
    <div class="text-center">

```

```

        <a class="small" href="<?php echo base_url('auth/login')
?>">Sudah Punya Akun? Klik disini untuk Login</a>
    </div>
</div>
</div>
</div>
</div>
</div>
</div>

</div>
</body>

</html>

```

View/admin/dashboard_admin.php

```

<div class="container-fluid">
    <!-- Content Row -->
    <div class="column">

        <div></div>
        <h1>Selamat Datang <?php echo $this->session-
>userdata('username') ?></h1>
        <br>
        <h3>Silahkan klik Data Barang untuk mengelola Barang Jual.</h3>
        <h3>Silahkan klik Invoice untuk melihat pesanan pelanggan.</h3>
        <hr>
        <div style="display: flex; height: 30vh; align-
items: center; margin: 0 auto; justify-content:space-around;">
            <a href="<?php echo base_url('admin/data_barang') ?>" style="
width: 25%; font-size: 25px; color: white;" class="btn btn-md btn-
danger"><i class="fa fa-database mr-2"></i>Data Barang</a>
            <a href="<?php echo base_url('admin/invoice') ?>" style="wi
dth: 25%; font-size: 25px; color: white;" class="btn btn-md btn-
success"><i class="fa fa-clone mr-2"></i>Invoice</a>
        </div>
    </div>

    <!-- Content Row -->

```

</div>

View/admin/data_barang.php

```
<div class="container-fluid">
    <button class="btn btn-sm btn-primary mb-3" data-
toggle="modal" data-target="#tambah_barang">
        <i class="fas fa-plus fa-sm"></i>
        Tambah Barang
    </button>
    <table class="table table-bordered">
        <tr>
            <th>NO</th>
            <th>NAMA BARANG</th>
            <th>KETERANGAN</th>
            <th>KATEGORI</th>
            <th>HARGA</th>
            <th>STOK</th>
            <th colspan="3">AKSI</th>
        </tr>

        <?php
        $no=1;
        foreach($barang as $brg) : ?>

            <tr>
                <td><?php echo $no++ ?></td>
                <td><?php echo $brg->nama_mkn ?></td>
                <td><?php echo $brg->keterangan ?></td>
                <td><?php echo $brg->kategori ?></td>
                <td><?php echo number_format($brg->harga, 0,',','.') ?></td>
                <td><?php echo $brg->stok ?></td>
                <td><?php echo anchor('admin/data_barang/detail/' . $brg-
>id_mkn, '<div class="btn btn-success btn-sm"><i class="fas fa-search-
plus"></i></div>') ?></td>
                <td><?php echo anchor('admin/data_barang/edit/' . $brg-
>id_mkn, '<div class="btn btn-primary btn-sm"><i class="fa fa-
edit"></i></div>') ?></td>
```

```

                <td><?php echo anchor('admin/data_barang/hapus/' . $brg-
>id_mkn, '<div class="btn btn-danger btn-sm"><i class="fa fa-
trash"></i></div>') ?></td>
            </tr>
            <?php endforeach; ?>
        </table>
    </div>

```

```

<!-- Modal -->
<div class="modal fade" id="tambah_barang" tabindex="-1" aria-
labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-
title" id="exampleModalLabel">Form Input Product</h5>
                <button type="button" class="close" data-dismiss="modal" aria-
label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                <form action="<?php echo base_url(). 'admin/data_barang/tambah_a
ksi' ?>" method="post" enctype="multipart/form-data">

```

```

                <div class="form-group">
                    <label for="">Nama Barang</label>
                    <input type="text" name="nama_mkn" class="form-control">
                </div>

```

```

                <div class="form-group">
                    <label for="">Keterangan</label>
                    <input type="text" name="keterangan" class="form-
control">
                </div>

```

```

                <div class="form-group">
                    <label for="">Kategori</label>
                    <select class="form-control" name="kategori">

```



```

        <option>makanan</option>
        <option>minuman</option>
        <option>lalapan</option>
    </select>
</div>

<div class="form-group">
    <label for="">Harga</label>
    <input type="text" name="harga" class="form-control">
</div>

<div class="form-group">
    <label for="">Stok</label>
    <input type="text" name="stok" class="form-control">
</div>

<div class="form-group">
    <label for="">Gambar Produk</label>
    <input type="file" name="gambar" class="form-control">
</div>

</div>
<div class="modal-footer">
    <button type="button" class="btn btn-danger" data-
dismiss="modal">Tutup</button>
    <button type="submit" class="btn btn-primary">Simpan</button>
</div>
</form>
</div>
</div>
</div>

```

View/admin/detail_invoice.php

```
<div class="container-fluid">
    <h4>Detail Pesanan <div class="btn btn-sm btn-
success">No. Invoice: <?php echo $invoice->id ?></div></h4>

    <table class="table table-bordered table-hover table-striped">
        <tr>
            <th>ID MAKANAN</th>
            <th>NAMA MAKANAN</th>
            <th>JUMLAH PESANAN</th>
            <th>PENGIRIMAN</th>
            <th>BANK</th>
            <th>HARGA SATUAN</th>
            <th>SUB-TOTAL</th>
        </tr>

        <?php
            $total = 0;
            foreach($pesanan as $psn) :
                $subtotal = $psn->jumlah * $psn->harga;
                $total += $subtotal;
            ?>
        <tr>
            <td><?php echo $psn->id_mkn ?></td>
            <td><?php echo $psn->nama_mkn ?></td>
            <td><?php echo $psn->jumlah ?></td>
            <td><?php echo $psn->pengiriman ?></td>
            <td><?php echo $psn->bank ?></td>
            <td><?php echo number_format($psn->harga,0, ',', '.') ?></td>
            <td><?php echo number_format($subtotal,0, ',', '.') ?></td>
        </tr>

        <?php endforeach ?>

        <tr>
            <td colspan="4" align="right">Grand Total</td>
            <td align="right">Rp. <?php echo number_format($total, 0, ',
', '.') ?></td>
```

```

        </tr>
    </table>

    <a href="<?php echo base_url('admin/invoice/index') ?>"><div class="
btn btn-sm btn-primary">Kembali</div></a>
</div>

```

View/admin/detail.php

```

<div class="container-fluid">
    <div class="card">
        <h5 class="card-header">Detail Makanan</h5>
        <div class="card-body">

            <?php foreach($barang as $brg):

                ?>
                <div class="row">
                    <div class="col-md-4">
                        
                    </div>
                    <div class="col-md-8">
                        <table class="table">
                            <tr>
                                <td>Nama Produk</td>
                                <td><strong><?php echo $brg-
>nama_mkn ?></strong></td>
                            </tr>

                            <tr>
                                <td>Keterangan</td>
                                <td><strong><?php echo $brg-
>keterangan ?></strong></td>
                            </tr>

                            <tr>
                                <td>Kategori</td>

```

```

        <td><strong><?php echo $brg-
>kategori ?></strong></td>
    </tr>

    <tr>
        <td>Stok</td>
        <td><strong><?php echo $brg-
>stok ?></strong></td>
    </tr>

    <tr>
        <td>Harga</td>
        <td><strong><div class="btn btn-sm btn-
success">Rp. <?php echo number_format($brg-
>harga,0,',','.') ?></div></strong></td>
    </tr>
</table>
    <?php echo anchor('admin/data_barang/', '<div class="btn
btn-sm btn-danger">Kembali</div>') ?>
    </div>
</div>
<?php endforeach ?>
</div>
</div>
</div>

```

View/admin/edit_barang.php

```

<div class="container-fluid">
    <h3><i class="fas fa-edit"></i>EDIT DATA BARANG</h3>

    <?php foreach($barang as $brg) : ?>

        <form action="<?php echo base_url(). 'admin/data_barang/update'
?>" method="post">

            <div class="form-group">
                <label>Nama Barang</label>

```

```

        <input type="text" name="nama_mkn" class="form-
control" value="<?php echo $brg->nama_mkn ?>">
    </div>

    <div class="for-group">
        <label>Keterangan</label>
        <input type="hidden" name="id_mkn" class="form-
control" value="<?php echo $brg->id_mkn ?>">
        <input type="text" name="keterangan" class="form-
control" value="<?php echo $brg->keterangan ?>">
    </div>

    <div class="for-group">
        <label>Kategori</label>
        <input type="text" name="kategori" class="form-
control" value="<?php echo $brg->kategori ?>">
    </div>

    <div class="for-group">
        <label>Harga</label>
        <input type="text" name="harga" class="form-
control" value="<?php echo $brg->harga ?>">
    </div>

    <div class="for-group">
        <label>Stok</label>
        <input type="text" name="stok" class="form-
control" value="<?php echo $brg->stok ?>">
    </div>

    <button type="submit" class="btn btn-primary btn-sm mt-
3">Simpan</button>
</form>

<?php endforeach; ?>
</div>

```

View/admin/invoice.php

```
<div class="container-fluid">
    <h4>Invoice Pemesanan Produk</h4>

    <table class="table table-bordered table-hover table-strip">
        <tr>
            <th>Id Invoice</th>
            <th>Nama Pemesan</th>
            <th>Alamat Pengiriman</th>
            <th>Tanggal Pemesanan</th>
            <th>Batas Pembayaran</th>
            <th>Aksi</th>
        </tr>

        <?php foreach ($invoice as $inv): ?>
            <tr>
                <td><?php echo$inv->id ?></td>
                <td><?php echo$inv->nama ?></td>
                <td><?php echo$inv->alamat ?></td>
                <td><?php echo$inv->tgl_pesan ?></td>
                <td><?php echo$inv->batas_bayar ?></td>
                <td><?php echo anchor('admin/invoice/detail/'.$inv-
                >id,'<div class="btn btn-sm btn-primary">Detail</div>') ?></td>
            </tr>

        <?php endforeach; ?>
    </table>
</div>
```

Models/model_auth.php

```
<?php
class model_auth extends CI_Model{
    public function cek_login(){
        $username = set_value('username');
        $password = set_value('password');

        $result = $this->db->where('username',$username)
                        ->where('password',$password)
                        ->limit(1)
                        ->get('tb_user');

        if($result->num_rows() > 0){
            return $result->row();
        }else{
            return array();
        }
    }
}
?>
```

Models/model_barang.php

```
<?php
class Model_barang extends CI_Model{
    public function tampil_data(){
        return $this->db->get('makanan');
    }

    public function tambah_barang($data,$table){
        $this->db->insert($table,$data);
    }

    public function edit_barang($where,$table){
        return $this->db->get_where($table,$where);
    }
}
```

```

public function update_data($where,$data,$table){
    $this->db->where($where);
    $this->db->update($table, $data);
}

public function hapus_data($where,$table){
    $this->db->where($where);
    $this->db->delete($table);
}

public function find($id){
    $result = $this->db->where('id_mkn', $id)
        ->limit(1)
        ->get('makanan');

    if($result->num_rows()>0){
        return $result->row();
    }else{
        return array();
    }
}

public function detail_brg($id_mkn){
    $result = $this->db->where('id_mkn',$id_mkn)->get('makanan');
    if($result->num_rows() > 0){
        return $result->result();
    }else{
        return false;
    }
}
}

?>

```


Models/model_invoice.php

```
<?php
class Model_invoice extends CI_Model{

    public function index(){
        date_default_timezone_set('Asia/Jakarta');
        $nama    = $this->input->post('nama');
        $alamat   = $this->input->post('alamat');
        $pengiriman = $this->input->post('pengiriman');
        $bank      = $this->input->post('bank');

        $invoice = array(
            'nama'      => $nama,
            'alamat'    => $alamat,
            'tgl_pesan' => date('Y-m-d H:i:s'),
            'batas_bayar'=> date('Y-m-
d H:i:s', mktime(date('H'), date('i'), date('s'), date('m'), date('d') +
1, date('Y'))),
            'pengiriman' => $pengiriman,
            'bank'       => $bank,
        );

        $this->db->insert('tb_invoice', $invoice);
        $id_invoice = $this->db->insert_id();

        foreach ($this->cart->contents() as $item){
            $data = array(
                'id_invoice'    => $id_invoice,
                'id_mkn'        => $item['id'],
                'nama_mkn'      => $item['name'],
                'jumlah'        => $item['qty'],
                'harga'         => $item['price'],
            );
            $this->db->insert('tb_pesanan', $data);
        }
        return true;
    }
}
```

```

    public function tampil_data(){
        $result = $this->db->get('tb_invoice');
        if($result->num_rows() > 0){
            return $result->result();
        }else{
            return false;
        }
    }

    public function ambil_id_invoice($id_invoice){
        $result = $this->db->where('id', $id_invoice)->limit(1)-
>get('tb_invoice');
        if($result->num_rows() > 0){
            return $result->row();
        }else{
            return false;
        }
    }

    public function ambil_id_pesanan($id_invoice){
        $result = $this->db->where('id_invoice', $id_invoice)-
>get('tb_pesanan');
        if($result->num_rows() > 0){
            return $result->result();
        }else{
            return false;
        }
    }
}

?>

```

Models/model_kategori.php

```
<?php
    class model_kategori extends CI_Model{
        public function data_makanan(){
            return $this->db-
>get_where('makanan',array('kategori' => 'makanan'));
        }

        public function data_minuman(){
            return $this->db-
>get_where('makanan',array('kategori' => 'minuman'));
        }

        public function data_lalapan(){
            return $this->db-
>get_where('makanan',array('kategori' => 'lalapan'));
        }
    }
?>
```

Controllers/auth.php

```
<?php
    class Auth extends CI_Controller{
        public function login(){

            $this->form_validation-
>set_rules('username', 'Username', 'required');
            $this->form_validation-
>set_rules('password', 'Password', 'required');
            if($this->form_validation->run() == FALSE){
                $this->load->view('templates/header');
                $this->load->view('form_login');
                $this->load->view('templates/footer');
            }else{
                $auth = $this->model_auth->cek_Login();
                if($auth == FALSE){
```

```

        $this->session->
>set_flashdata('pesan','<div class="alert alert-danger alert-
dismissible fade show" role="alert">
    Username atau Password yang anda masukkan salah!
</div>');
    redirect('auth/login');
}else{
    $this->session->set_userdata('username',$auth-
>username);
    $this->session->set_userdata('role_id',$auth-
>role_id);

    switch($auth->role_id){
        case 1 : redirect('admin/dashboard_admin');
        break;
        case 2 : redirect('welcome');
        break;
        default: break;
    }
    }
}

public function logout(){
    $this->session->sess_destroy();
    redirect('auth/login');
}
}
?>

```

Controllers/dashboard.php

```
<?php
class dashboard extends CI_Controller{
    public function __construct()
    {
        parent::__construct();
        if($this->session->userdata('role_id') != '2'){
            $this->session->
>set_flashdata('pesan','<div class="alert alert-danger alert-
dismissible fade show" role="alert">
            Anda Belum Login!
            </div>');
            redirect('auth/login');
        }
    }

    public function tambah_ke_keranjang($id){
        $barang = $this->model_barang->find($id);

        $data = array(
            'id'      => $barang->id_mkn,
            'qty'     => 1,
            'price'   => $barang->harga,
            'name'    => $barang->nama_mkn,

        );

        $this->cart->insert($data);
        redirect('welcome');
    }

    public function detail_keranjang(){
        $this->load->view('templates/header');
        $this->load->view('templates/sidebar');
        $this->load->view('keranjang');
        $this->load->view('templates/footer');
    }
}
```

```

public function hapus_keranjang(){
    $this->cart->destroy();
    redirect('welcome/index');
}

public function pembayaran(){
    $this->load->view('templates/header');
    $this->load->view('templates/sidebar');
    $this->load->view('pembayaran');
    $this->load->view('templates/footer');
}

public function proses_pesanan(){

    $is_processed = $this->model_invoice->index();
    if($is_processed){
        $this->cart->destroy();
        $this->load->view('templates/header');
        $this->load->view('templates/sidebar');
        $this->load->view('proses_pesanan');
        $this->load->view('templates/footer');
    }else{
        echo "Maaf, Pesanan Anda Gagal Diproses.";
    }

}

public function detail($id_mkn){
    $data['barang'] = $this->model_barang->detail_brg($id_mkn);
    $this->load->view('templates/header');
    $this->load->view('templates/sidebar');
    $this->load->view('detail_barang', $data);
    $this->load->view('templates/footer');
}
}
?>

```

Controllers/kategori.php

```
<?php
    class kategori extends CI_Controller{
        public function makanan(){
            $data['makanan'] = $this->model_kategori->data_makanan()-
>result();
            $this->Load->view('templates/header');
            $this->Load->view('templates/sidebar');
            $this->Load->view('makanan',$data);
            $this->Load->view('templates/footer');
        }

        public function minuman(){
            $data['minuman'] = $this->model_kategori->data_minuman()-
>result();
            $this->Load->view('templates/header');
            $this->Load->view('templates/sidebar');
            $this->Load->view('minuman',$data);
            $this->Load->view('templates/footer');
        }

        public function lalapan(){
            $data['lalapan'] = $this->model_kategori->data_lalapan()-
>result();
            $this->Load->view('templates/header');
            $this->Load->view('templates/sidebar');
            $this->Load->view('lalapan',$data);
            $this->Load->view('templates/footer');
        }
    }
?>
```

Controllers/registrasi.php

```
<?php
    class Registrasi extends CI_Controller{
        public function index(){

            $this->form_validation-
>set_rules('nama','Nama','required', ['required' => 'Nama Wajib Diisi!'
]);

            $this->form_validation-
>set_rules('username','username','required', ['required' => 'Username Wa
jib Diisi!' ]);

            $this->form_validation-
>set_rules('password_1','password','required|matches[password_2]', ['req
uired' => 'Password Wajib Diisi!', 'matches' => 'Password tidak cocok' ]
);

            $this->form_validation-
>set_rules('password_2','password','required|matches[password_1]');

            if($this->form_validation->run() == false){
                $this->load->view('templates/header');
                $this->load->view('registrasi');
                $this->load->view('templates/footer');
            }else{
                $data = array(
                    'id'           => '',
                    'nama'         => $this->input->post('nama'),
                    'username'     => $this->input->post('username'),
                    'password'     => $this->input->post('password_1'),
                    'role_id'      => 2,
                );

                $this->db->insert('tb_user',$data);
                redirect('auth/login');
            }
        }
    }
?>
```


Controllers/welcome.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Welcome extends CI_Controller {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     * - or -
     *      http://example.com/index.php/welcome/index
     * - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     * So any other public methods not prefixed with an underscore will
     * map to /index.php/welcome/<method_name>
     * @see https://codeigniter.com/user_guide/general/urls.html
     */
    public function index(){
        $data['barang'] = $this->model_barang->tampil_data()->result();
        $this->load->view('templates/header');
        $this->load->view('templates/sidebar');
        $this->load->view('dashboard', $data);
        $this->load->view('templates/footer');
    }
}
```

.htaccess

RewriteEngine On

RewriteCond %{REQUEST_FILENAME} !-f

RewriteCond %{REQUEST_FILENAME} !-d

RewriteRule ^(.*)\$ index.php/\$1 [L]