**Penerapan Algoritma *Binary Search* UntukPencarian Judul Makanan Pada Aplikasi Penjualan Makanan *Digital* Berbasis Web (*E-Commerce*)**

SKRIPSI



**Oleh**

**HUDAI HANGUN**

**1112091000077**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH**

**JAKARTA**

**2017 / 1438 H**

**Penerapan Algoritma *Binary Search* Untuk Pencarian Judul MakananPada Aplikasi Penjualan Makanan *Digital* Berbasis Web (*E-Commerce*)**

Skripsi

Sebagai Salah Satu Syarat Untuk Memperoleh Gelar

Sarjana Komputer

Fakultas Sains Dan Teknologi

Universitas Islam Negeri Syarif Hidayatullah Jakarta



**Oleh**

**HUDAI HANGUN**

**1112091000077**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH**

**JAKARTA**

**2017 / 1438 H**

# LEMBAR PERSETUJUAN

**Penerapan Algoritma *Binary Search* Untuk Pencarian Judul MakananPada Aplikasi Penjualan Makanan *Digital***

**Berbasis Web (*E-Commerce*)**

Skripsi

Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Komputer

Pada Fakultas Sains Dan Teknologi

Universitas Islam Negeri Syarif Hidayatullah Jakarta

**Diajukan Oleh :**

**HUDAI HANGUN**

**1112091000077**

Menyetujui,

|  |  |  |
| --- | --- | --- |
| **Pembimbing I** |  | **Pembimbing II** |
| Anif Hanifa Setyaningrum, M.SI  **NIDN : 072802 034038 9 997** |  | Rizal Bahaweres, M. Kom  **NIP : 197108062014111001** |
|  |  |  |
| Mengetahui, | | |
| KETUA PROGRAM STUDI TEKNIK INFORMATIKA | | |
| Arini, M.T.  **NIP : 19760131200901 2 001** | | |

# LEMBAR PENGESAHAN

Skripsi yang berjudul **“Penerapan Algoritma Binary Search Untuk Pencarian Judul Makanan Pada Aplikasi Penjualan Makanan Digital Berbasis Web (E-Commerce)”** yang ditulis oleh Hudai Hangun, NIM. 1112091000077 telah diuji dan dinyatakan **LULUS** dalam sidang *Munaqosyah* Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta pada 06 Januari 2017. Skripsi ini telah diterima sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu (S1) Komputer (S.Kom) pada Program Studi Teknik Informatika.

Jakarta, 06 Januari 2017

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Menyetujui, | | | | |
| **Penguji I** |  | | | **Penguji II** |
|  |  | | |  |
| **Ria Hari Gusmita, M.Kom**  NIP : 198208172009122002 |  | | | **Nurul Faizah Rozy, MTI**  NIDN : 2009027202 |
|  | | | | |
| **Pembimbing I** | |  | **Pembimbing II** | |
|  | |  |  | |
| **Anif Hanifa Setyaningrum, M.SI**  NIDN. 0728020340389997 | |  | **Rizal Bahaweres, M.Kom**  NIP : 197108062014111001 | |
| Mengetahui, | | | | |
| **Dekan**  **Fakultas Sains dan Teknologi** |  | | | **Ketua**  **Program Studi Teknik Informatika** |
| **DR. Agus Salim, M.Si**  NIP. 197208161999031003 |  | | | **Arini, MT**  NIP. 197601312009012001 |

# PERNYATAAN ORISINALITAS

Dengan ini saya menyatakan bahwa:

1. Skripsi ini benar-benar Hasil Karya Saya sendiri diajukan untuk memenuhi salah satu persyaratan memperoleh Gelar Strata 1 di UIN Syarif Hidayatullah Jakarta.
2. Sumber yang di gunakan dalam penulisan ini telah di cantumkan sesuai dengan ketentuan yang berlaku di UIN Syarif Hidayatullah Jakarta.

Bogor, 24 Februari 2017

**Hudai Hangun**

**1112091000077**

# 

# KATA PENGANTAR

Bismillahirrahmanirrahim,

Puji dan syukur penulis panjatkan kehadirat Allah SWT atas segala rahmat dan karunia-Nya yang telah diberikan kepada penulis sehingga penulis dapat menyelesaikan penelitian dan penulisan skripsi ini. Shalawat dan Salam selalu tersampaikan kepada Nabi Besar Muhammad SAW, keluarga, sahabat dan para pengikutnya hingga akhir zaman. Skripsi penulis yang berjudul **“Penerapan Algoritma Binary Search Untuk Pencarian Judul Makanan Pada Aplikasi Penjualan Makanan Digital Berbasis Web (E-Commerce)”** disusun untuk memenuhi salah satu syarat dalam menyelesaikan Program Strata Satu (S1) pada Program Studi Teknik Informatika, Universitas Islam Negeri Syarif Hidayatullah Jakarta. Suatu kebanggaan tersendiri bagi penulis apabila skripsi ini dapat bermanfaat terutama bagi pihak-pihak yang membutuhkannya. Pada kesempatan ini, penulis ingin mengucapkan terima kasih kepaada pihak-pihak yang telah membantu penulis, baik secara moril maupun materil, sehingga penulisan ini terlaksana dengan baik. Secara khusus penulis mengucapkan terimakasih kepada:

1. Bapak Dr. Agus Salim, M.Si selaku Dekan Fakulas Sains dan Teknologi.
2. Ibu Arini, ST, MT selaku Ketua Program Studi Teknik Informatika dan Bapak Feri Fahrianto, M.Sc selaku Sekretaris Program Studi Teknik Informatika, Fakultas Sains dan Teknologi yang selalu memberikan motivasi dan arahan kepada penulis.
3. Ibu A. Hanifah Setyaningrum, M.Si. Selaku Dosen Pembimbing I dan Bapak Rizal Bahaweres, M. Kom selaku Dosen Pembimbing II, yang telah memberikan arahan atau bimbingan kepada penulis.
4. Seluruh Dosen Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, yang telah banyak memberikan bimbingan dan pengarahan kepada penulis selama menjalankan aktifitas perkuliahan.
5. Ayahanda Drs. Samsul Fadli yang telah memberikan moril maupun materil, Ibunda tercinta Dra. Heti Nurhayati yang telah setia mendukung serta mendoakan penulis untuk tetap bersungguh-sungguh dalam menyelesaikan skripsi ini.
6. Kepada Fauzan Burdi S.Kom, Muhammad Rifki S.Si, Chandra Muhammad Fikri, Julio Abdilla, Reza Hermawan, Teddy Perdana Azta, Santi Frannita SH, Firli Sucia Sari, Hidaya Sari, selaku rekan penulis yang telah banyak memberikan masukan, semangat, informasi dan arahan kepada penulis untuk menyelesaikan skripsi ini.
7. Terimakasih kepada rekan-rekan seperjuangan di UIN Syarif Hidayatullah Jakarta khususnya rekan-rekan CCIT FT UI Pola 2.2 dan 1.3 angkatan 2010, yang telah memberikan dukungan dan semangat serta saran-saran yang berguna hingga akhir penulisan skripsi ini.

Akhir kata, saya berharap semoga Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Ciputat, 06 Januari 2017

Penulis

|  |  |  |
| --- | --- | --- |
| **Nama** | **:** | **Hudai Hangun** |
| **Program Studi** | **:** | **1112091000077** |
| **Judul** | **:** | **Penerapan Algoritma Binary Search Untuk Pencarian Judul Makanan Pada Aplikasi Penjualan Makanan Digital Berbasis Web (E-Commerce)** |

# ABSTRAK

Saat ini telah banyak beredar aplikasi web *e-commerce* penjualan makanan di web, namun dari sekian banyak aplikasi yang telah beredar peneliti tidak melihat adanya aplikasi yang memberikan hasil pencarian data makanan yang sudah dijual seperti nama pembeli makanan dan harganya. Untuk mengatasi permasalahan tersebut dibuat sebuah aplikasi E-*Commerce* penjualan makanan dengan memanfaatkan algoritma *binary search* sebagai algoritma pencarian data nama makanan. *Database* yang digunakan untuk melakukan proses pencarian data makanan adalah berupa *file text* yang berjumlah 12 macam makan. Metode pengumpulan data yang digunakan dalam penelitian ini adalah observasi dan studi pustaka. Metode pengembangan sistem yang digunakan dalam penelitian ini adalah *Rapid Application Development* (RAD). Dengan adanya aplikasi web e-*commerce* pencarian data makanan ini diharapkan dapat memudahkan pengguna internet dalam mendapatkan nilai yang bagus.

|  |  |  |
| --- | --- | --- |
| Kata Kunci | : | *E-commerce*, Algoritma *Binary Search,* RAD |
| Jumlah Halaman | : | VI BAB + xxii Halaman + 101 Halaman + 65 Gambar + 47 Tabel + 4 Lampiran |
| Daftar Pustaka | : | 22 (tahun 2007-2016) |

|  |  |  |
| --- | --- | --- |
| **Name** | **:** | **Hudai Hangun** |
| **Study Program** | **:** | **1112091000077** |
| **Title** | **:** | **Penerapan Algoritma Binary Search Untuk Pencarian Judul Makanan Pada Aplikasi Penjualan Makanan Digital Berbasis Web (E-Commerce)** |

# *ABSTRACT*

*Currently have been circulating web applications e-commerce sales food on the internet , but how many application in circulation while we do not see any applications data search results feed with such as the buyer's name and the price of food. To solve the problem made an application e-commerce Makanan sales digital music using binary algorithm search as algorithms data search song .A database that are used to perform the process for finding data song was text of file totaling 12 food .Data collection method used in research is observation and the literature study .The development of system used in research is rapid application development ( rad ) .With the web applications e-commerce food is expected to facilitate internet users in obtaining food with accompanied complete.*

|  |  |  |
| --- | --- | --- |
| *Keywords* | *:* | *E-commerce, Binary Search Algorithm, RAD* |
| *Number of Pages* | *:* | *Chapter VI + xxii Page + 101 Page + 65 Image + 47 Tables + 4 Attachment* |
| *Bibliography* | *:* | *22 (years 2007-2016)* |

# DAFTAR ISI

[LEMBAR PERSETUJUAN iii](#_Toc484599535)

[LEMBAR PENGESAHAN iv](#_Toc484599536)

[PERNYATAAN ORISINALITAS v](#_Toc484599537)

[KATA PENGANTAR vi](#_Toc484599538)

[ABSTRAK viii](#_Toc484599539)

[*ABSTRACT* ix](#_Toc484599540)

[DAFTAR ISI x](#_Toc484599541)

[DAFTAR GAMBAR xiii](#_Toc484599542)

[DAFTAR TABEL xvi](#_Toc484599543)

[DAFTAR SIMBOL xviii](#_Toc484599544)

[BAB I PENDAHULUAN 2](#_Toc484599545)

[1.1. Latar Belakang 2](#_Toc484599546)

[1.2 Batasan dan Rumusan Masalah 3](#_Toc484599547)

[1.2.2. Batasan Masalah 3](#_Toc484599548)

[1.2.1. Rumusan masalah 4](#_Toc484599549)

[1.3. Tujuan Penelitian 4](#_Toc484599550)

[1.4. Manfaat Penelitian 4](#_Toc484599551)

[1.5 Metode Penelitian 5](#_Toc484599552)

[1.5.1 Metode Pengumpulan Data 5](#_Toc484599553)

[1.5.2 Metode Pengembangan Sistem 5](#_Toc484599554)

[1.6 Sistematika Penulisan 7](#_Toc484599555)

[BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI 9](#_Toc484599556)

[2.1 Tinjauan Pustaka 9](#_Toc484599557)

[2.2 Landasan Teori 10](#_Toc484599558)

[2.2.1 Binary Search 10](#_Toc484599559)

[2.2.2 Makanan 13](#_Toc484599560)

[2.2.3 Aplikasi Pengendalian *Internal* 14](#_Toc484599561)

[2.2.4 Penjualan 15](#_Toc484599562)

[2.2.5 Restoran 15](#_Toc484599563)

[2.2.6. Bisnis 17](#_Toc484599564)

[2.2.7 Definisi *E-Commerce* 16](#_Toc484599565)

[2.2.8. *Rapid Application Development* (RAD) 20](#_Toc484599566)

[2.2.9. *Unfield Model Language (UML)* 20](#_Toc484599567)

[BAB III METODOLOGI PENELITIAN 22](#_Toc484599568)

[3.1 Metode Pengumpulan Data 22](#_Toc484599569)

[3.1.1. Observasi 22](#_Toc484599570)

[3.1.2 Studi Pustaka 22](#_Toc484599571)

[3.2 Metode Pengembangan Sistem 22](#_Toc484599572)

[3.2.1. Fase – Fase RAD (Rapid Application Development) 23](#_Toc484599573)

[3.3 Kerangka Berpikir 24](#_Toc484599574)

[BAB IV IMPLEMENTASI 25](#_Toc484599575)

[4.1 Fase Perencanaan Syarat-Syarat 25](#_Toc484599576)

[4.1.1 Identifikasi Masalah 25](#_Toc484599577)

[4.1.2 Lingkup Sistem 25](#_Toc484599578)

[4.1.3 Tujuan Sistem 25](#_Toc484599579)

[4.1.4. Sistem Usulan 26](#_Toc484599580)

[4.2 Fase Workshop Desain 27](#_Toc484599581)

[4.2.1 Perancangan Algoritma *Binary Search* 27](#_Toc484599582)

[4.2.2. Pengkodean Algoritma *Binary Search* 32](#_Toc484599583)

[4.2.3 Perancangan Proses 32](#_Toc484599584)

[4.2.4 Perancangan Interface 61](#_Toc484599585)

[4.3. Fase Implementasi 63](#_Toc484599586)

[4.3.4 Pengujian (*Black Box Testing*) 63](#_Toc484599587)

[BAB V HASIL DAN PEMBAHASAN 87](#_Toc484599588)

[5.1 Pencarian Judul Makanan Dengan Algoritma *Binary Search* 87](#_Toc484599589)

[5.2 Proses Pemesanan Pencarian Makanan 91](#_Toc484599590)

[BAB VI PENUTUP 98](#_Toc484599591)

[1.1 Kesimpulan 98](#_Toc484599592)

[1.2 Saran 98](#_Toc484599593)

[DAFTAR PUSTAKA 99](#_Toc484599594)

[LAMPIRAN 102](#_Toc484599595)

# DAFTAR GAMBAR

[Gambar 3. 1 Kerangka Berpikir Penelitian 24](#_Toc484599370)

[Gambar 4. 1 Sistem Usulan Album Musik *Digital* 26](file:///E:\xampp\htdocs\Warung_Hudai\UML%20Warung%20Hudai\Skripsi%20hudai.docx#_Toc484606811)

[Gambar 4. 2 Gambar *Flowchart Algoritma Binary Search* 31](#_Toc484606812)

[Gambar 4. 3 Diagram *Use case* Monitoring Keuangan 35](#_Toc484606813)

[Gambar 4. 4 Gambar *Desain Activity diagram Login* 43](#_Toc484606814)

[Gambar 4. 5 *Gambar Desain Activity diagram* Memasukkan Menu 44](#_Toc484606815)

[Gambar 4. 6 Gambar *Desain Activity diagram* Menampilkan Menu 45](#_Toc484606816)

[Gambar 4. 7 Gambar *Desain Activity diagram* Mengubah Menu 46](#_Toc484606817)

[Gambar 4. 8 Gambar *Desain Activity diagram* Menghapus Menu 47](#_Toc484606818)

[Gambar 4. 9 Gambar Desain *Activity diagram* Memasukkan Pesanan 48](#_Toc484606819)

[Gambar 4. 10 Gambar Desain *Activity diagram* Menampilkan Pesanan 49](#_Toc484606820)

[Gambar 4. 11 Gambar Desain *Activity diagram* Mengubah Pesanan 50](#_Toc484606821)

[Gambar 4. 13 Gambar Desain *Activity diagram Logout Admin* 51](#_Toc484606822)

[Gambar 4. 14 *Activity diagram* menghapus pesanan 52](#_Toc484606823)

[Gambar 4. 15 Gambar Desain *Sequence Diagram Login* 53](#_Toc484606824)

[Gambar 4. 16 Gambar Desain *Sequence Diagram* Memasukkan Menu 54](#_Toc484606825)

[Gambar 4. 17 Gambar Desain *Sequence Diagram* Menampilkan Menu 55](#_Toc484606826)

[Gambar 4. 18 Gambar Desain *Sequence Diagram* Mengubah Menu 55](#_Toc484606827)

[Gambar 4. 19 Gambar Desain Sequence Diagram Menghapus Menu 56](#_Toc484606828)

[Gambar 4. 20 Gambar Desain *Sequence Diagram* Memasukkan Pesanan 57](#_Toc484606829)

[Gambar 4. 21 Gambar Desain *Sequence Diagram* Menampilkan Pesanan 58](#_Toc484606830)

[Gambar 4. 22 Gambar Desain *Sequence Diagram* Mengubah Pesanan 59](#_Toc484606831)

[Gambar 4. 23 Gambar Desain *Sequence Diagram* Menghapus Pesanan 60](#_Toc484606832)

[Gambar 4. 24 Gambar *User Interface Login Admin* 61](#_Toc484606833)

[Gambar 4. 25 Gambar *User Interface* Halaman Utama *Admin* 61](#_Toc484606834)

[Gambar 4. 26 Gambar *User Interface* Halaman Utama *Member* 62](#_Toc484606835)

[Gambar 4. 27 Gambar *User Interface form* Pencarian Menu (*Owner*) 62](#_Toc484606836)

[Gambar 4. 28 Gambar *User Interface* Laporan Keuangan 63](#_Toc484606837)

[Gambar 5. 1 Gambar Pencarian Judul Makanan 87](#_Toc484599398)

[Gambar 5 2 Hasil Pencarian Makanan "Seperti Hidup Kembali" 87](#_Toc484599399)

[Gambar 5. 3 Hasil Pencarian “Kuingin Kau Untukku” 88](#_Toc484599400)

[Gambar 5. 4 Hasil Pencarian Makanan "idul Fitri" 88](#_Toc484599401)

[Gambar 5 5 Hasil Pencarian Judul Makanan "Bukan Cinta Sempurna" 88](#_Toc484599402)

[Gambar 5 6 Makanan Bukan Cinta Sempurna 89](#_Toc484599403)

[Gambar 5 7 MP3 Bukan Cinta Sempurna 90](#_Toc484599404)

[Gambar 5. 8 Gambar Halaman Detail Makanan Dari Makanan 90](#_Toc484599405)

[Gambar 5 9 *Form Login Member* 91](#_Toc484599406)

[Gambar 5 10 Halaman Menu Pencarian Makanan (*Member*, Pengunjung) 91](#_Toc484599407)

[Gambar 5 11 Halaman Menu Keranjang Belanja (*Member*) 92](#_Toc484599408)

[Gambar 5 12. Halaman Menu *Member* 92](#_Toc484599409)

[Gambar 5 13. Halaman *Manage* Data Pesanan (*Member*) 93](#_Toc484599410)

[Gambar 5 14. Halaman Konfirmasi Pembayaran (*Member*) 93](#_Toc484599411)

[Gambar 5 15. Halaman *Manage* Data Pesanan (*Member*) Sudah Konfirmasi Pembayaran 94](#_Toc484599412)

[Gambar 5 16. Halaman *Login Admin* 94](#_Toc484599413)

[Gambar 5 17. Halaman Menu *Manage* Data Pemesanan (*Admin*) 95](#_Toc484599414)

[Gambar 5 18. Halaman Menu *Manage* Data Pemesanan (*Admin*) – Validasi 95](#_Toc484599415)

[Gambar 5 19. Halaman *Upload File* Makanan (*Admin*) 96](#_Toc484599416)

[Gambar 5 20. Halaman *Manage* Data Pemesanan (*Admin*) - *File* Makanan Sudah Terisi 96](#_Toc484599417)

[Gambar 5 21. Halaman *Manage* Data Pesanan (*Member*) - *Download* 97](#_Toc484599418)

[Gambar 5 22. Halaman *Pop Up* Hasil *Download* *File* Pencarian Makanan (*Member*) 97](#_Toc484599419)

[Lampiran 1 (Gambar) 1. Desain *Class Diagram* Aplikasi Monitoring Keuangan 102](#_Toc484599420)

[Lampiran 2 (Gambar) 1. Desain *Mapping Database* (Pencarian Makanan) 105](#_Toc484599421)

# DAFTAR TABEL

[Tabel 2 1Tabel Perbandingan Studi Literatur 9](#_Toc484599495)

[Tabel 2 2 Perbandingan Metode Pengembangan Sistem (Rizki, 2014) 20](#_Toc484599496)

[Tabel 4. 1 *Pseudocode* Algoritma *Binary Search* 27](#_Toc484599497)

[Tabel 4. 2 Tabel Data Makanan - Kasus 1 28](#_Toc484599498)

[Tabel 4. 3 Tabel Pembahasan Kasus 1 (Mid = Key) 28](#_Toc484599499)

[Tabel 4. 4 Tabel Data Makanan - Kasus 2 29](#_Toc484599500)

[Tabel 4. 5 Tabel Pembahasan Kasus 2 (Mid > Key) 29](#_Toc484599501)

[Tabel 4. 6 Tabel Data Makanan - Kasus 3 30](#_Toc484599502)

[Tabel 4. 7 Tabel Pembahasan Kasus 3 (Mid < Key) 30](#_Toc484599503)

[Tabel 4. 8. Pengkodean Algoritma Binary Search dengan Bahasa Pemograman PHP 32](file:///E:\xampp\htdocs\Warung_Hudai\UML%20Warung%20Hudai\Skripsi%20hudai.docx#_Toc484599504)

[Tabel 4. 9 Deskripsi Aktor *Use case* 33](#_Toc484599505)

[Tabel 4. 10 Identifikasi *Use case* 33](#_Toc484599506)

[Tabel 4. 11 Narasi *Use case Login* *Owner dan* *Admin* 36](#_Toc484599507)

[Tabel 4. 13 Narasi *Use case* Memasukkan Menu 36](#_Toc484599508)

[Tabel 4. 15 Narasi *Use case* Menampilkan Menu 37](#_Toc484599509)

[Tabel 4. 16 Narasi *Use case* Mengubah Menu 37](#_Toc484599510)

[Tabel 4. 17 Narasi *Use case* Menghapus Menu 38](#_Toc484599511)

[Tabel 4. 18 Narasi *Use case* Menampilkan Pesanan 40](#_Toc484599512)

[Tabel 4. 19 Narasi *Use case* Mengubah Pesanan 40](#_Toc484599513)

[Tabel 4. 20 Narasi *Use case* Menghapus Pesanan 41](#_Toc484599514)

[Tabel 4. 21 Narasi *Use case* Pencarian Menu Untuk Monitoring 42](#_Toc484599515)

[Tabel 4. 27 Tabel Pengujian *User* (*Admin*) 63](#_Toc484599516)

[Tabel 4. 28 Tabel Pengujian User (*Member*) 65](#_Toc484599517)

[Tabel 4. 29 Pengujian *User* (Pengunjung) 65](#_Toc484599518)

[Lampiran 2 (Tabel) 1. Tabel *User* 106](#_Toc484599519)

[Lampiran 2 (Tabel) 2. Tabel Testimoni 106](#_Toc484599520)

[Lampiran 2 (Tabel) 3. Tabel Profil Web 106](#_Toc484599521)

[Lampiran 2 (Tabel) 4. Tabel Chord Gitar 107](#_Toc484599522)

[Lampiran 2 (Tabel) 5. Tabel Chord *Keyboard* 107](#_Toc484599523)

[Lampiran 2 (Tabel) 6. Tabel Kategori 107](#_Toc484599524)

[Lampiran 2 (Tabel) 7. Tabel Genre 108](#_Toc484599525)

[Lampiran 2 (Tabel) 8. Tabel Makanan 108](#_Toc484599526)

[Lampiran 2 (Tabel) 9. Tabel Makanan Klik 109](#_Toc484599527)

[Lampiran 2 (Tabel) 10. Tabel Pemesanan 109](#_Toc484599528)

[Lampiran 2 (Tabel) 11. Tabel Keranjang 110](#_Toc484599529)

[Lampiran 2 (Tabel) 12. Tabel Makanan 110](#_Toc484599530)

[Lampiran 2 (Tabel) 13. Tabel Artis 111](#_Toc484599531)

[Lampiran 2 (Tabel) 14. Tabel Berita 111](#_Toc484599532)

[Lampiran 2 (Tabel) 15. Tabel Cara Pembayaran 111](#_Toc484599533)

[Lampiran 2 (Tabel) 16. Tabel Cara Pemesanan 112](#_Toc484599534)

# DAFTAR SIMBOL

Simbol *Use case* Diagram

(A.S & Shalahuddin, 2013)

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama** | **Gambar** | **Deskripsi** |
| 1. | *Use case* | Description: F:\Me\Pekerjaan\Perguruan Tinggi Negeri (KAMPUS)\seamobs\Rancangan BAB edisi 3 (revisi 23 Mei 2016)\Diagram1\daftar simbol\usecase\nama use case.jpg | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama *use case*. |
| 2. | Aktor/*aktor* | Description: F:\Me\Pekerjaan\Perguruan Tinggi Negeri (KAMPUS)\seamobs\Rancangan BAB edisi 3 (revisi 23 Mei 2016)\Diagram1\daftar simbol\usecase\nama aktor.jpg | Orang, proses, atau sistem lain yang berinteraksi de-ngan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |
| 3 | Asosiasi/ *Association* | Description: F:\Me\Pekerjaan\Perguruan Tinggi Negeri (KAMPUS)\seamobs\Rancangan BAB edisi 3 (revisi 23 Mei 2016)\Diagram1\daftar simbol\usecase\asosiasi.jpg | Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki interaksi dengan aktor. |
| 4. | Ekstensi/ *Extend* | Description: F:\Me\Pekerjaan\Perguruan Tinggi Negeri (KAMPUS)\seamobs\Rancangan BAB edisi 3 (revisi 23 Mei 2016)\Diagram1\daftar simbol\usecase\extends.jpg | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu; mirip dengan prinsip *inheritance* pada pem-rograman berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan. |
| 5. | Generalisasi/*Generali-zation* | Description: F:\Me\Pekerjaan\Perguruan Tinggi Negeri (KAMPUS)\seamobs\Rancangan BAB edisi 3 (revisi 23 Mei 2016)\Diagram1\daftar simbol\usecase\generalisasi.jpg | Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case*  dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |
| 6. | Mengguna-kan/*Include* | Description: F:\Me\Pekerjaan\Perguruan Tinggi Negeri (KAMPUS)\seamobs\Rancangan BAB edisi 3 (revisi 23 Mei 2016)\Diagram1\daftar simbol\usecase\include.jpg | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsi-nya atau sebagai syarat dijalankan *use case* ini. |

Simbol Class Diagram

(A.S & Shalahuddin, 2013)

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama** | **Gambar** | **Deskripsi** |
| 1. | Kelas/*Class* | nama\_kelas  +atribut  +operasi() | Kelas pada struktur sistem. |
| 2. | Antarmuka/ *Interface* | nama\_interface | Sama dengan konsep *interface* dalam pemrog-raman berorientasi objek yaitu kumpulan operasi tanpa implementasi dari suatu kelas. |
| 3 | Asosiasi/ *Association* |  | Relasi antar kelasdengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity.* |
| 4. | Asosiasi Berarah/ *Directed Association* |  | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity*. |
| 5. | Generalisasi/ Generali-zation |  | Relasi antar kelas dengan makna generalisasi-spesia-lisasi (umum-khusus). |
| 6. | Kebergantu-ngan/*Depe-dency* |  | Relasi antar kelas dengan makna kebergantungan antar kelas. |
| 7. | Agregasi/ *Aggregation* |  | Relasi antar kelas dengan makna semua bagian (*whole-part*). |

Simbol *Activity diagram*

(A.S & Shalahuddin, 2013)

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama** | **Gambar** | **Fungsi** |
| 1. | Status awal |  | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal |
| 2. | Aktivitas | Aktifitas | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| 3. | Percabang-an/*Decision* |  | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| 4. | Penggabu-ngan/*Join* |  | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan men-jadi satu. |
| 5. | Status akhir |  | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |
| 6. | *Swimlane* | swimlane | Memisahkan organisasi bis-nis yang bertanggung jawab terhadap aktivitas yang terjadi. |

Simbol Sequence Diagram

(A.S & Shalahuddin, 2013)

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama** | **Gambar** | **Fungsi** |
| 1. | Aktor | Description: F:\Me\Pekerjaan\Perguruan Tinggi Negeri (KAMPUS)\seamobs\Rancangan BAB edisi 3 (revisi 23 Mei 2016)\Diagram1\daftar simbol\usecase\aktor.jpg | Orang, proses, atau sistem lain yang berinteraksi de-ngan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar dari orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |
| 2. | Garis hidup/*life-line* |  | Menyatakan kehidupan suatu objek. |
| 3. | Objek | Nama objek : nama kelas | Menyatakan objek yang berinteraksi pesan. |
| 4. | Waktu aktif |  | Menyatakan objek dalam keadaan aktif dan berinter-aksi pesan. |
| 5. | Pesan tipe *create* | <<create>>  1 : CreateMessage() | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. |
| 6. | Pesan tipe *call* | 1 : namaModel() | Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.  Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagramkelas sesuai dengan kelas objek yang berinter-aksi. |
| 7. | Pesan tipe *send* | 1 : masukan() | Menyatakan bahwa suatu objek mengirimkan data/ masukan/informasi ke objek lainnya, arah panah meng-arah pada objek yang dikirimi. |
| 8. | Pesan tipe *return* | keluaran | Menyatakan bahwa suatu objek yang telah men-jalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kem-balian. |
| 9. | Pesan tipe *destroy* | Description: C:\Users\Reza\AppData\Local\Microsoft\Windows\INetCache\Content.Word\sd.jpg | Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada *create* maka ada *destroy*. |

**BAB I** **PENDAHULUAN**

## 1.1. Latar Belakang

Awal tahun 2000-an, Penjualan Makanan mulai mengalami penurunan disebakan penetrasi penjualan makanan dalam bentuk web yang begitu cepat. Ditunjang oleh semakin mudahnya orang pemesan Makanan di internet dan kemampuan *smartphone*, yang membuat penjualan Makanan dalam bentuk web menurun (kurang diminati). Makanan dalam format *digital* web dan penetrasi internet yang semakin meluas saat ini, membuat bermunculan berbagai layanan ditambah lagi *smartphone*.

Dalam melakukan pencarian data Makanan pada aplikasi web penjualan Pencarian Makanan, kecepatan waktu dalam melakukan proses pencarian data Makanan menjadi elemen penting. Terdapat banyak algoritma pencarian, diantaranya yaitu, algoritma *binary search* yang cocok untuk pencarian data dengan jumlah data yang banyak dan terurut menaik, elemen dalam data yang dapat ditemukan dalam waktu singkat. Lalu algoritma *sequential search* cocok untuk pencarian terhadap jumlah data yang tidak banyak dan data yang dicari tidak perlu terurut. **(Munir, 2011)**.

Sebagai rujukan adalah, penelitian mengenai pencarian data dengan algoritma *binary search* dilakukan oleh Toto Nafo Duha (2013) yang berjudul “**Perancangan Aplikasi Kidung Jemaat Berbasis Java Dengan Menggunakan Algoritma *Binary Search Ascending* Pada Telepon Seluler**” menerangkan pembuatan aplikasi pencarian kidung jemaat berbasis handphone dengan bahasa pemograman Java. Penelitian lain dilakukan oleh Qory Kurnia RC (2015) yang berjudul **“Aplikasi Kamus Indonesia – Korea Mengunakan Algoritma Binary Search Berbasis Android”,** menerangkan pembuatan aplikasi android kamus indonesia korea dengan bahasa pemograman Java. Dalam penelitian ini peneliti menggunakan algoritma *binary search* sebagai algoritma untuk proses pencarian data judul Makanan. Algoritma *binary search* dipilih karena aplikasi *e-commerce* Pencarian Makanan ini kedepannya akan memiliki jumlah data Makanan yang banyak di dalam *database* Makanan dan membutuhkan proses pencarian data yang cepat. Algoritma *binary search* cocok untuk melakukan pencarian data terurut menaik pada sekumpulan data Makanan di dalam *database* dan cocok untuk pencarian data dengan jumlah *input* Makanan dalam jumlah besar.

Berdasarkan hasil pengumpulan data yang diperoleh, maka penulis tertarik untuk membuat sebuah simulasi aplikasi web Pencarian Makanan (E-*Commerce*) sederhana yang bergerak dalam bidang penjualan Makanan secara *digita web* dengan proses pencarian judul Makanan menggunakan algoritma *binary search*. Hal ini yang mendasari penulis untuk membuat skripsi dengan judul **“Penerapan Algoritma Binary Search Untuk Pencarian Judul Makanan Pada Aplikasi Penjualan Makanan Digital Berbasis Restaurant”.**

## 1.2 Batasan dan Rumusan Masalah

### 1.2.2. Batasan Masalah

Batasan masalah pada penelitian ini adalah :

1. Metode pengumpulan data dalam penelitian ini adalah menggunakan observasi dan studi pustaka.
2. Penelitian ini menggunakan algoritma *binary search.*
3. Aplikasi dibangun dengan bahasa pemograman PHP Versi 5.0.
4. Media penyimpanan data menggunakan *database* MySQL 1.7.4.
5. Aplikasi dirancang dengan *tools* adobe dreamweaver 6.0.
6. Tidak membahas proteksi keamanan data Makanan.
7. Jumlah Makanan di *database* sebanyak 12 Makanan dan ribuan pembeli.

### 1.2.1. Rumusan masalah

Rumusan masalah dalam penelitian ini adalah :

Bagaimana penerapan algoritma *binary search* unntuk pencarian judul Makanan pada aplikasi penjualan makanan digital berbasis web (*E-Commerce*)?

## 1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah :

Mengimplementasikan algoritma *binary search* pada pencarian untuk judul makanan pada aplikasi penjualan makanan digital.

## 1.4. Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Dapat menerapkan algoritma *binary search* pada proses pencarian judul makanan.
2. Dapat merancang konsep aplikasi e-*commerce* penjualan makanan digital
3. Menambah referensi literatur kepustakaan Universitas Islam Syarif Hidayatullah Jakarta.
4. Bahan masukan bagi mahasiswa UIN Syarif Hidayatullah Jakarta dalam pengembangan penulisan atau penelitian terkait penulisan ini.
5. Pengguna bisa lebih mudah mendapatkan hasil penjualan makanan.

## 1.5 Metode Penelitian

Dalam penelitian skripsi ini menggunakan beberapa metode penelitian yaitu :

### 1.5.1 Metode Pengumpulan Data

Di dalam penyusunan skripsi penulis melakukan pengumpulan data guna untuk menunjang kesuksesan dalam penelitian, adapun metode yang digunakan dalam penyusunan skripsi adalah sebagai berikut :

1. Observasi

Pengumpulan data yang dilakukan oleh peneliti adalah, pengamatan kepada aplikasi web makanan serupa.

1. Studi Pustaka

Pengumpulan data studi pustaka yang dilakukan oleh peneliti adalah dengan membaca dan mempelajari buku-buku, *e-book*, serta artikel-artikel dari sumber *online* (*website*) yang berhubungan dengan penyusunan skripsi ini dan peneliti mencari jurnal penelitian sejenis dan membandingkan aplikasi sejenis yang telah lebih dahulu dibuat.

### 1.5.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam pengembangan sistem ini adalah, metode pengembangan sistem dengan pemodelan RAD (*Rapid Apllication Development*), dan menggunakan *tools* UML (*Unified Modelling Language*) sebagai desain perancangan sistemnya. Adapun tahap-tahap dalam pengembangan sistem dengan RAD (*Rapid Apllication Development*) ini adalah sebagai berikut :

1. Fase Perencanaan Syarat –Syarat (*Requiremet Planning*)

Pada fase ini dilakukaan analisa kebutuhan masalah yaitu menentukan identifikasi masalah, tujuan dan lingkup pengembangan dari aplikasi serta merancang sistem usulan yang diperlukan dalam pengembangan sistem aplikasi.

1. Fase Perancangan (*Workshop Design*)

Tahap ini berhubungan dengan bagaimana mendesain sistem agar sistem dapat berjalan dengan baik. Tahapan yang dilakukan pada fase perancangan adalah sebagai berikut :

1. Perancangan algoritma *binary search* sebagai algoritma pencarian judul makanan.
2. Pengkodean

Pada tahap ini dilakukannya pembuatan program dengan kode bahasa pemograman PHP, HTML, CSS dan MySql sebagai *database.*

1. Perancangan proses meliputi :

* Perancangan desain UML (*Unified Modelling Language*) sebagai perancangan desain alur sistem, yang terdiri dari *Usecase diagram, activity diagram*, *sequence diagram* dan *class diagram.*
* *Mapping* *database* sebagai perancangan basis data sistem.

1. Perancangan *Interface*

Pada tahap ini peneliti merancang desain *interface* yang bertujuan untuk pengguna agar dapat berkomunikasi dengan sistem.

1. Fase Implementasi (*Implementation*)

Pada tahap ini penulis melakukan pengujian sistem yang telah dibangun dan dijalankan secara *localhost.* Pengujian (*Black Box Testing*) bertujuan agar dapat mengetahui aksi yang akan dilihat dalam suatu reaksi dari sistem itu sendiri. Untuk mengetahui tentang alur sistem tersebut telah berjalan sesuai dengan yang diharapkan atau tidak.

## 1.6 Sistematika Penulisan

Sistematika penulisan pada skripsi ini disusun kedalam enam bab, secara singkat akan diuraikan berikut :

**BAB I PENDAHULUAN**

Pada bab ini dijelaskan mengenai latar belakang, batasan dan rumusan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

**BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI**

Pada bab ini dijelaskan mengenai literatur penelitian skripsi serupa dan berbagai teori yang mendasari penelitian dan permasalahan yang berhubungan dengan topik yang dibahas.

**BAB III METODOLOGI PENELITIAN**

Pada bab ini berisi uraian lebih rinci tentang metodologi penelitian yang meliputi metodologi pengumpulan data dan metodologi pengembangan sistem.

**BAB IV IMPLEMENTASI**

Pada bab ini berisi tentang analisa permasalahan, usulan perancangan sistem yang dibangun, alur kerja sistem, dan pengujian sistem.

**BAB V HASIL DAN PEMBAHASAN**

Pada bab ini berisi hasil dan pembahasan pada sistem yang telah dibangun.

**BAB VI PENUTUP**

Pada bab ini dijelaskan mengenai kesimpulan dari penelitian yang dilakukan dan saran untuk mengembangkan aplikasi kedepan agar lebih baik.

# BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI

## 2.1 Tinjauan Pustaka

Berikut adalah beberapa jurnal penelitian yang berkaitan dengan Algoritma *Binary Search*.

Tabel 2 1Tabel Perbandingan Studi Literatur

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Judul Penelitian** | **Nama Peneliti** | **Deskripsi Penelitian** | **Kekurangan** | **Kelebihan** |
| 1 | Perancangan Aplikasi Kidung Jemaat Berbasis Java Dengan Menggunakan Algoritma *Binary Search*  *Ascending* Pada Telepon Seluler  2013 | Toto Nafo Duha | Peneliti membuat aplikasi pencarian kidung jemaat berbasis handphone dengan bahasa pemograman Java | Pencarian dengan algoritma *binary search* hanya menampilkan informasi saja, data aplikasi bersiftat statis | Aplikasi mampu melakukan pencarian Makanan dengan inputan berdasarkan nomor |
| 2 | Aplikasi Kamus Indonesia – Korea Mengunakan Algoritma *Binary Search* Berbasis Android  2015 | Qory Kurnia RC | Peneliti membuat aplikasi kamus indonesia – korea berbasis android dengan | Memiliki fitur *audio* terjemahan Indonesia-korea | 1. Pengelolaan *database* yang belum terstrukur 2. Belum adanya fasilitas cari terjemahan bahasa korea dengan |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | bahasa pemograman java |  | memasukkan kata hanggul ke Indonesia   1. Aplkasi hanya untuk android (offline) |
| 3 | Pembuatan Sistem Penjualan *Online* Pada Toko PN Sukoharjo  2012 | Ida Astarina, Berliana Kusuma Riasti | Peneliti membuat website penjualan alat secara *online* | Tidak adanya penggunaan algoritma pencarian | Menggunakan CMS Prestashop 2.0 dalam perancangan desain *interface* |

Berdasarkan perbandingan studi literatur diatas, maka peneliti merancang sebuah aplikasi web e-*commerce* Pencarian Makanan dengan fitur :

1. Menggunakan algoritma *binary search* sebagai monitoring keuntungan dari penjualan makanan setiap hari.
2. Aplikasi mampu memberikan hasil pencarian informasi lengkap diantaranya, Nama Makanan disertai harganya, aplikasi mampu menampilkan informasi Detail pembeli beserta tanggal dll.
3. Aplikasi dibangun dengan tools adobe dreamweafer 6.0, serta media penyimpanan data yaitu dengan MySQL 1.7.4 serta pengelolaan *database* sistem yang dinamis, dikelola sepenuhnya oleh admin.

## 2.2 Landasan Teori

Berikut adalah landasan teori yang digunakan pada penelitian ini:

### 2.2.1 Binary Search

*Binary search* merupakan salah satu algoritma untuk melalukan pencarian data pada *array* yang sudah terurut. Jika kita tidak mengetahui informasi bagaimana *integer* dalam *array*, maka penggunaan *binary search* akan menjadi tidak efisien, kita harus melakukan *sorting* terlebih dahulu atau menggunakan metode lain yaitu *linear search*. Namun jika kita telah mengetahui *integer* dalam *array* terorganisasi baik secara menaik atau menurun, maka bisa dengan cepat menggunakan algoritma *binary search*. Pencarian dilakukan dengan cara menebak apakah data yang dicari berada ditengah-tengah data, kemudian membandingkan data yang dicari dengan data yang ada ditengah. Bila data yang ditengah sama dengan data yang dicari, berarti data ditemukan. Namun, bila data yang ditengah lebih besar dari data yang dicari, maka dapat dipastikan bahwa data yang dicari kemungkinan berada disebelah kiri dari data tengah dan data disebelah kanan data tengah dapat diabaikan. Sebaliknya bila data yang ditengah lebih kecil dari data yang dicari makan kemungkinan data berada disebelah kanan dari data tengah dan data kiri dapat diabaikan. **(Duha, 2013)**. Pencarian Biner (*Binary Search*) dilakukan untuk **(Duha, 2013)**:

1. Memperkecil jumlah operasi pembandingan yang harus dilakukan antara data yang dicari dengan data yang ada di dalam tabel, khususnya untuk jumlah data yang sangat besar ukurannya.
2. Prinsip dasarnya adalah melakukan proses pembagian ruang pencarian secara berulang-ulang sampai data ditemukan atau sampai ruang pencarian tidak dapat dibagi lagi (berarti ada kemungkinan data tidak ditemukan).
3. Syarat utama untuk pencarian biner adalah data di dalam tabel harus sudah terurut, misalkan terurut menaik (*ascending*).

Cara kerja algoritma *Binary Search* sebagai berikut (**McMilan, 2014**):

1. Menetapkan batas bawah ke posisi pertama dari *array* (0).
2. Menetapkan batas atas untuk elemen terakhir dari *array* (panjang *array* / Jumlah data).
3. Tentukan posisi tengah antara batas kiri dan batas kanan dengan menjumlahkan batas kiri dan batas kanan kemudian dibagi 2.
4. Jika data tengah = data yang dicari, maka data ditemukan dan pencarian selesai.
5. Jika tidak, bandingkan lagi, jika data tengah > data dicari, maka batas atas memiliki nilai data tengah -1.
6. Jika data tengah < dari data yang dicari, maka batas bawah memiliki nilai data tengah + 1

Berikut kelebihan dan kekurangan algoritma *binary search* dengan algoritma *sequential search*:

Tabel 2.1 Perbandingan Algoritma **(Munir, 2011)**

|  |  |  |
| --- | --- | --- |
| **Algoritma** | **Kelebihan** | **Kekurangan** |
| Algoritma *Binary Search* | 1. Cocok untuk mencari data yang jumlahnnya sangat banyak 2. Elemen dalam data dapat ditemukan dalam waktu yang singkat | 1. Data yang dicari harus dalam keadaan terurut, oleh karena itu harus dilakukan *sorting* (pengurutan) sebelum melakukan searching dalam metode ini 2. Algoritmanya lebih kompleks dari *sequential search* |
| Algoritma *Sequential Search* | 1. Jika elemen yang dicari berada di depan (indeks awal *array*) maka elemen tersebutakan cepat dapat ditemukan 2. Data yang dicari tidak harus dalam keadaan terurut 3. Algoritmanya cukup sederhana | 1. Jika elemen yang dicari berada di akhir (indeks awal *array*), maka pencarian akan sangat lama sehingga beban komputasi akan meningkat 2. Tidak cocok untuk mencari sebuah elemen pada kumpulan data yang sangat banyak |

### 2.2.2 Makanan

Makanan adalah bahan, biasanya berasal dari hewan atau tumbuhan, yang dimakan oleh makhluk hidup mendapatkan tenaga dan nutrisi. Cairan yang di pakai untuk maksud ini sering disebut minuman. Tetapi kata ‘makanan’ juga bisa dipakai. Istilah ini kadang-kadang dipakai dengan kiasan, seperti “makanan untuk pemikiran”. Kecukupan makanan dapat dinilai dengan status gizi secara antropometri.

Makanan yang dibutuhkan manusia biasanya diperoleh dari hasil bertani atau berkebun yang meliputi sumber hewan, dan tumbuhan. Beberapa orang menolak untuk memakan makanan dari hewan seperti daging, telur dan lain-lain. Mereka yang tidak suka memakan daging, dan sejenisnya disebut *vegetarian* yaitu orang yang hanya memakan sayuran sebagai makanan pokok mereka. Pada umumnya bahan makanan mengandung beberapa unsur atau senyawa seperti air, karbohidrat, protein, lemak, vitamin, enzim, pigmen dan lain-lain. **(id.wikipedia).**

### 2.2.3 Aplikasi Pengendalian *Internal*

Aplikasi pengendalian *Internal* merupakan suatu perencanaan yang meliputi struktur organisasi dan semua metode dan alat-alat yang di koordinasikan yang digunakan di dalam perusahaan dengan tujuan untuk menjaga keamanan harta milik perusahaan, memeriksa ketelitian dan kebenaran data akutansi, mendorong efisiensi, dan membantu mendorong dipatuhinya kebijakan manajemen yang telah ditetapkan.

**2.2.3.1 Peran Penting Aplikasi Pengendalian *Internal***

1. Membantu manajemen dalam mengendalikan dan memastikan keberhasilan kegiatan organisasi.

2. Menciptakan pengawasan melekat, menutupi kelemahan dan keterbatasan personil, serta mengurangi kemungkinan terjadinya kesalahan dan kecurangan.

3. Membantu auditor dalam menentukan ukuran sampel dan pendekatan audit yang akan di terapkan.

4. Membantu auditor dalam memastikan efektifitas.

5. Audit, dengan keterbatasan waktu dan biaya audit.

**2.2.3.2 Keterbatasan Aplikasi Pengendalian *Internal***

1. Kekeliruan pengoprasian sistem ( *Mistake in judgement* ) karena terbatasnya informasi dan waktu, karena tekanan lingkungam, atau karena terbatasnya kemampuan, meskipun sudah dilengkapi dengan pedoman penyelesaian masalah.

2. Pelanggaran sistem ( *breakdowns* ), baik disengaja atau tidak, misalnya karena kesalahan interpretasi, kecerobohan, gangguan lingkungan, perubahan personalia, atau perubahan sistem dan prosedur.

3. Kolusi, atau kerjasama negative sekelompok orang.

4. pelanggaran dengan sengaja oleh manajemen (*management override*).

5. Dilema biaya-manfaat ( *costs versus benefits* ).

### 2.2.4 Penjualan

Penjualan adalah ilmu dan seni yang mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang atau jasa yang ditawarkan. **(Swatha, 2010)**

### 2.2.5 Restoran

Menurut Keputusan Menteri Pariwisata, Pos dan Telekomunikasi Nomor KM 95/HK. 103/MPPT-87, Restoran adalah salah satu jenis usaha pangan yang bertempat di sebagian atau diseluruh bangunan yang permanen dilengkapi peralatan dan perlengkapan untuk proses pembuatan, penyimpanan, penyajian dan penjualan makanan dan minuman bagi umum di tempat usahanya dan memenuhi ketentuan persyaratan yang ditetapkan dalam keputusan ini. Restoran dalam operasinya terjadi semacam barter antara pembeli dengan penjual, dalam hal ini antara produk dengan uang. Hubungan tersebut tidak akan berjalan dengan mulus apabila petugas yang akan menangani pelayanan tidak terseleksi secara cermat, dididik dan dilatih dengan baik, diajar berkomunikasi serta dikoordinasikan dengan teliti serta dipersiapkan dengan kesungguhan hati.

**2.2.5.1 Jenis-jenis Restoran**

Restoran memiliki beberapa bentuk atau jenis. Terdapat sepuluh jenis restoran berdasarkan tingkat keorsinilannya, yaitu :

1. Family Continental, yaitu restoran yang didirikan berdasarkan pada tradisi keluarga dimana restoran ini mementingkan masakan enak, suasana yang nyaman serta harga yang bersahabat namun pelayanan dan dekorasi yang ditawarkan sederhana.
2. Fast Food, yaitu restoran yang mengutamakan kecepatan penyajian, terbatas dalam jenis, mahal dan mengutamakan banyak pelanggan. Dekorasi restoran jenis ini lebih mengutamakan warna-warna utama dan terang.
3. Kafetaria, biasanya terdapat di dalam gedung-gedung perkantoran atau pusat perbelanjaan, sekolah dan pabrik-pabrik. Menu yang ditawarkan agak terbatas dan biasanya berganti-ganti menurut hari, dan harga relatif ekonomis.
4. Gourmet, merupakan restoran berkelas memerlukan suasana restoran yang sangat nyaman dengan dekorasi artistik. Jenis ini ditujukan kepada mereka yang menuntut standar penyajian yang tinggi dan bergengsi.
5. Etnik, yaitu restoran yang menyajikan masakan dari daerah (suku atau negara) yang spesifik, misalnya masakan Padang, Jawa Timur, Cina, Jepang, dan lain-lain. Dekorasi biasanya disesuaikan menurut etnik yang bersangkutan bahkan termasuk seragam karyawannya.
6. Buffet, restoran dengan cirri utamanya adalah pemberian satu harga untuk makan sepuas-puasnya apa yang disajikan pada buffet. Peragaan dan display makanan sangat penting disini, sebab ia langsung menjual dirinya.
7. Coffee Shop, jenis restoran yang ditandai dengan pelayanan secara cepat dan cepat pergantian tempat duduk. Banyak seating menempati counter service untuk menekan suasana informal. Lokasi utama biasanya di gedung perkantoran dan pusat perbelanjaan.
8. Snack Bar, suatu ruangan yang biasanya lebih kecil namun cukup untuk melayani orang-orang yang ingin makanan kecil atau jajanan.
9. Drive in/thru or parking, dimana pada restoran ini pembelian dapat dilakukan tanpa perlu turun dari mobilnya. Pesanan diantar hingga ke mobil untuk ”eat in (parkir sementara) atau take away dengan jenis makanan dikemas secara praktis.
10. Speciality Restaurant, jenis restoran yang terletak jauh dari keramaian, tetap menyajikan makanan khas yang menarik dan bermutu. Ditujukan kepada turis dan keluarga dengan suasana yang khas dan unik. Harga yang ditetapkan restoran ini relatif mahal.

### 2.2.6. Bisnis

Dalam [ilmu ekonomi](https://id.wikipedia.org/wiki/Ilmu_ekonomi), **bisnis** adalah suatu organisasi yang menjual barang atau jasa kepada [konsumen](https://id.wikipedia.org/wiki/Konsumen) atau bisnis lainnya, untuk mendapatkan [laba](https://id.wikipedia.org/wiki/Laba). Secara historis kata bisnis dari [bahasa Inggris](https://id.wikipedia.org/wiki/Bahasa_Inggris) *business*, dari kata dasar *busy* yang berarti "sibuk" dalam konteks individu, komunitas, ataupun masyarakat. Dalam artian, sibuk mengerjakan aktivitas dan pekerjaan yang mendatangkan keuntungan.

Dalam [ekonomi](https://id.wikipedia.org/wiki/Ekonomi) kapitalis, di mana kebanyakan bisnis dimiliki oleh pihak swasta, bisnis dibentuk untuk mendapatkan profit dan meningkatkan kemakmuran para pemiliknya. Pemilik dan operator dari sebuah bisnis mendapatkan imbalan sesuai dengan [waktu](https://id.wikipedia.org/wiki/Waktu), usaha, atau [kapital](https://id.wikipedia.org/wiki/Kapital) yang mereka berikan. Namun tidak semua bisnis mengejar keuntungan seperti ini, misalnya bisnis koperatif yang bertujuan meningkatkan kesejahteraan semua anggotanya atau institusi pemerintah yang bertujuan meningkatkan kesejahteraan rakyat. Model bisnis seperti ini kontras dengan sistem sosialistik, di mana bisnis besar kebanyakan dimiliki oleh pemerintah, masyarakat umum, atau serikat pekerja.

Secara etimologi, bisnis berarti keadaan di mana seseorang atau sekelompok orang sibuk melakukan pekerjaan yang menghasilkan keuntungan. Kata "bisnis" sendiri memiliki tiga penggunaan, tergantung skupnya — penggunaan singular kata bisnis dapat merujuk pada [badan usaha](https://id.wikipedia.org/wiki/Badan_usaha), yaitu kesatuan yuridis (hukum), teknis, dan ekonomis yang bertujuan mencari laba atau keuntungan. Penggunaan yang lebih luas dapat merujuk pada sektor pasar tertentu, misalnya "bisnis pertelevisian." Penggunaan yang paling luas merujuk pada seluruh aktivitas yang dilakukan oleh komunitas penyedia barang dan jasa. Namun definisi "bisnis" yang tepat masih menjadi bahan perdebatan hingga saat ini.

### 2.2.7 Definisi *E-Commerce*

E-*Commerce* atau perdagangan elektronik, dalam arti luas, adalah penggunaan jaringan komputer untuk meningkatkan kinerja organisasi. Meningkatkan profitabilitas, memperoleh pangsa pasar, meningkatkan layanan pelanggan, dan memberikan produk lebih cepat. Perdagangan elektronik melibatkan penggunaan teknologi informasi untuk meningkatkan komunikasi dan transaksi dengan semua pemangku kepentingan organisasi. Pemangku kepentingan tersebut meliputi pelanggan, pemasok, regulator pemerintah, lembaga keuangan, karyawan, dan masyarakat luas **(Watson, 2008).**

#### 2.2.7.1 Karakteristik E-Commerce

Dari berbagai definisi yang ditawarkan dan dipergunakan oleh berbagai kalangan, terdapat kesamaan dari setiap definisi tersebut. Kesamaan ini menunjukkan bahwa e-*commerce* memiliki karakteristik **(Indriyani, 2009**):

1. Terjadinya transaksi antara dua belah pihak.
2. Adanya pertukaran barang, jasa atau informasi.
3. *Internet* merupakan medium utama dalam proses atau mekanisme perdangan tersebut.

#### 2.2.7.2 Jenis-Jenis E-Commerce

Dalam melakukan penelitian ini, penulis menggunakan jenis e-*commerce Business to Consumers* (B2C). **(Indriyani, 2009).** Perusahaan harus mengembangkan pasar elektronis dan menarik untuk menjual berbagai produk dan jasa ke para pelanggan. Sistem e-*commerce* yang berfokus pada pelanggan memiliki tujuan pentng yang sama, yaitu : menarik calon pembeli, melakukan transaksi atas barang dan jasa, serta membangun loyalitas pelanggan melalui pelayanan yang baik untuk setiap individu dan terlibat dengan berbagai fitur komunitas, *Busines to consumer* e-*commerce* memiliki karakteristik sebagai berikut.

1. Terbuka untuk umum, dimana informasi disebarkan ke umum.
2. Servis yang diberikan bersifat umum (*generic*) dengan mekanisme yang dapat digunakan oleh khalayak ramai. Sebagai contoh, karena sistem *web* sudah umum digunakan maka servis diberikan dengan menggunakan basis *web*.
3. Servis diberikan berdasarkan permohonan (*on demand*), yakni konsumen melakukan inisiatif dan produsen harus siap memberikan respon sesuai dengan permohonan.

#### 2.2.7.3 Keuntungan E-Commerce

Ada tiga keuntungan atau manfaat e-*commerce* dalam melakukan perdagangan elektronis, yaitu **(Nugroho, 2009)**.

1. Keuntungan bagi perusahaan
2. Memperpendek jarak

Perusahaan-perusahaan dapat lebih mendekatkan diri dengan konsumen.

1. Memperluas pasar

Jangkauan pemasaran menjadi semakin luas dan tidak terbatas oleh area geofrafis dimana perusahaan berada.

1. Perluasan jaringan mitra bisnis

Dengan adanya jaringan elektronis memudahkan mitra kerja dalam melakukan kerjasama.

1. Keutungan bagi konsumen
2. Efektif

Konsumen dapat memperoleh informasi tentang produk atau jasa yang dibutuhkannya dan berinteraksi dengan cara cepat dan mudah.

1. Aman secara fisik

Konsumen tidak perlu mendatangi toko secara langsung dan ini mungkin dapat bertransaksi dengan aman, sebab daerah-daerah tertentu mungkin sangat berbahaya jika berkendaraan dan membawa uang tunai dalam jumlah besar.

1. Fleksibel

Konsumen dapat melakukan transaksi dari berbagai lokasi, baik dari rumah, kantor, warnet atau tempat lainnya.

1. Keuntungan masyarakat umum
2. Membuka peluang kerja baru

Dalam hal ini akan membuka peluang-peluang kerja baru bagi mereka yang tidak buta teknologi.

1. Mengurangi polusi dari pencemaran lingkungan

Dengan adanya ini konsumen tidak perlu melakukan perjalanan ke toko-toko, dimana hal ini pada gilirannya akan mengurangi jumlah kendaraan yang berlalu-lalang di jalan.

1. Meningkatkan kualitas sumber daya manusia

Dalam hal ini hanya orang-orang yang tidak gagap teknologi, sehingga pada gilirannya akan merangsang orang-orang dalam mempelajari teknologi.

#### 2.2.7.4 Kerugian E-Commerce

Disamping keuntungan diatas, ada beberapa segi negatif dalam perdagangan elektronis, yaitu (**Nugroho, 2009**):

1. Meningkatkan individualisme

Dengan adanya ini seseorang dapat berinteraksi tanpa bantuan orang lain.

1. Terkadang menimbulkan kekecewaan

Apa yang dilihat dilayar monitor komputer kadang berbeda dengan apa yang dilihat secar kasat mata.

1. Tidak manusiawi

Dalam hal ini tidak dapat secara dekat bersosialisasi dengan orang lain, misalnya : tidak dapat berjabat tangan.

#### 2.2.7.5 Proses Pembayaran E-Commerce

Proses pembayaran e-*commerce* cukup rumit karena banyaknya jenis alternatif debit dan kredit serta lembaga keuangan dan perantara yang menjadi bagian dari proses tersebut diantara proses pembauaran tersebut antara lain **(Indriyani, 2009):**

1. Proses pembayaran *web* : Sistem pembayaran B2C
2. EFT (*Electronic Fund Transfer*) contoh : ATM, EPOS atau POS.
3. Proses pembayaran dalam sistem : Contoh : *Transfer*, COD (*Cash on Delivery*), dan kartu kredit.

### 2.2.8. *Rapid Application Development* (RAD)

*Rapid Application Development* (RAD) adalah suatu pendekatan berorientasi objek terhadap pengembangan sistem yang mencakup suatu metode pengembangan serta perangkat-perangkat lunak. RAD bertujuan mempersingkat waktu yang biasanya diperlukan dalam siklus hidup pengembangan sistem tradisional antara perancangan dan penerapan suatu sistem informasi. Pada akhirnya, RAD sama-sama berusaha memenuhi syarat-syarat bisnis yang berubah secara cepat. **(Kenneth & Julie, 2013).**

Tabel 2 2 Perbandingan Metode Pengembangan Sistem (Rizki, 2014)

|  |  |  |
| --- | --- | --- |
| **No** | **Metode** | **Keterangan** |
| 1 | *Rapid Application Development* (RAD) | Cocok dalam waktu pengerjaan yang singkat dan tidak memerlukan sumber daya yang cukup besar |
| 2 | *Prototype* | Cocok digunakan untuk menggali spesifkasi kebutuhan pelanggan secara lebih detail |
| 3 | *Waterfall* | Cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah |

### 2.2.9. *Unfield Model Language (UML)*

UML *(Unified Modelling Language)* merupakan bahasa visual untuk permodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks khusus **(A.S & Shalahuddin, 2013).** Pada penelitian ini penulis menggunakan 4 diagram, yaitu:

#### 2.2.9.1. Use Case Diagram

*Use case* atau diagram *use case* merupakan permodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih *aktor* dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu **(A.S & Shalahuddin, 2013).** Dibawah ini adalah deskripsi dari nama simbol *use case.*

#### 2.2.9.2. Activity Diagram

*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem, bukan apa yang dilakukan aktor, jadi aktivitas yang dilakukan oleh sistem **(A.S & Shalahuddin, 2013).**

#### 2.2.9.3. Sequence Diagram

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar *sequence diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu **(A.S & Shalahuddin, 2013).**

#### 2.2.9.4. Class Diagram

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi **(A.S & Shalahuddin, 2013).**

# BAB III METODOLOGI PENELITIAN

## Metode Pengumpulan Data

### 3.1.1. Observasi

Pengumpulan data yang dilakukan oleh peneliti adalah, pengamatan Warung Wara Wiri warungku sendiri.

### 3.1.2 Studi Pustaka

Pengumpulan data studi pustaka yang dilakukan oleh peneliti adalah dengan membaca dan mempelajari buku-buku, *e-book*, serta artikel-artikel dari sumber *online* (*website*) yang berhubungan dengan penyusunan skripsi ini dan peneliti mencari jurnal penelitian sejenis dan membandingkan aplikasi sejenis yang telah lebih dahulu dibuat.

## 3.2 Metode Pengembangan Sistem

Dalam pengembangan aplikasi web ini, metode pengembangan sistem yang digunakan adalah RAD (*Rapid Application Development*). Metode ini memiliki tiga fase yaitu fase perencanaan dan syarat-syarat, fase *workshop design*, dan fase implementasi. Pemilihan metode ini dikarenakan aplikasi web yang diharapkan mempunyai suatu desain yang dapat diterima oleh *user* dan dapat dikembangkan dengan mudah, karena perancangan aplikasi ini membutuhkan pengembangan lebih lanjut.

### 

### 3.2.1. Fase – Fase RAD (Rapid Application Development)

Tahapan-tahapan dalam pengembangan aplikasi dalam *Rapid Application Development* meliputi :

#### 3.2.1.1. Requirement Planning (Fase Perencanaan Syarat - Syarat)

Pada fase ini dilakukaan analisa kebutuhan masalah yaitu menentukan identfikasi permasalahan yang melandasi pembuatan sistem, lingkup sistem untuk menentukan batasan ruang lingkup yang dirancang, tujuan sistem untuk menentukan untuk apa dan untuk siapa pengguna sistem ini dirancang dan menentukan sistem usulan sebagai rancangan sebuah sistem aplikasi web dari hasil yang didapat pada identifikasi masalah berupa sistem aplikasi web penjualan Pencarian Makanan.

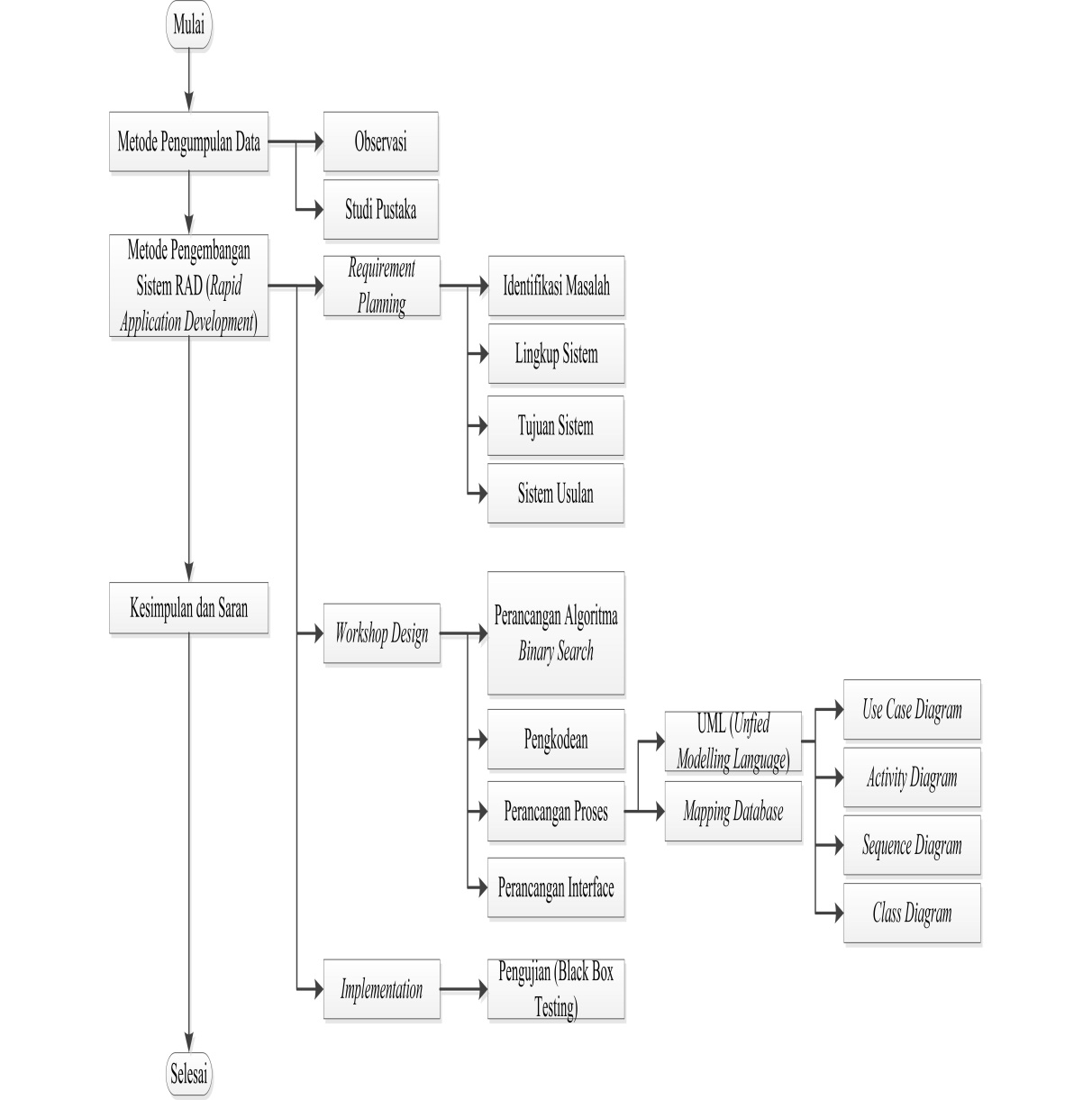
#### 3.2.1.2. Workshop Design

Setelah fase perencanaan syarat-syarat telah dilakukan, fase selanjutnya adalah fase *workshop design*. Tahap ini berhubungan dengan bagaimana mendesain sistem agar sistem dapat berjalan dengan baik dengan menggunakan. Pada fase ini dilakukannya perancangan algoritma *binary search,* pengkodean program dengan bahasa PHP sebagai inti proses dari sistem, bantuan *tools* adobe dreamweaver CS6 sebagai tool untuk mendesain *layout user client* dan *admin*, MySQL sebagai media penyimpanan data, perancangan proses dengan UML (*Unfield Modelling Language*) yang terdiri dari *usecase diagram, activity diagram*, *sequence diagram* dan *class diagram,* dan *mapping database* sebagai pembuatan *desain* *database* sistem, perancangan desain *interface* bertujuan untuk memudahkan pengguna dapat berkomunikasi dengan sistem.

#### 3.2.1.3. Implementation (Fase Implementasi)

Pada tahap ini penulis melakukan pengujian sistem yang telah dibangun dan dijalankan secara *localhost,* pada fase ini dilakukannya Pengujian (*Black Box Testing*) Pada tahap ini pengujian aplikasi web dilakukan untuk mengetahui *error* yang masih ada sebelum digunakan oleh pengguna (*user*).

## 3.3 Kerangka Berpikir

****

Gambar 3. 1 Kerangka Berpikir Penelitian

# BAB IV IMPLEMENTASI

## 4.1 Fase Perencanaan Syarat-Syarat

### 4.1.1 Identifikasi Masalah

Latar belakang masalah yang penulis identifikasi adalah, Hasil pencarian Makanan tidak menampilkan informasi lengkap, seperti tidak adanya informasi Makanan disertai dari Makanan yang dicari serta penjualan Pencarian Makanan dan proses pembelian Pencarian Makanan tidak mudah diperoleh melalui aplikasi web e-*commerce*.

### 4.1.2 Lingkup Sistem

Perancangan sistem ini berfokus pada permasalahan dan mencapai hasil yang optimal. Sistem digunakan oleh *admin*, pengunjung dan *member*, metode konfirmasi pembayaran yang digunakan dalam bertransaksi adalah transfer dengan cara meng-*upload* foto bukti pembayaran ke sistem. Lingkup utama sistem yang dibuat dalam penelitian ini adalah aplikasi web dijalankan secara localhost dan hanya mencakup cara pemesanan Pencarian Makanan dan pencarian judul Makanan dengan menggunakan algoritma *binary search* sebagai pemroses hasil pencarian. Adapun metode pengembangan sistem yang digunakan menggunakan RAD (*Rapid Application Development*) dari tahap fase perencanaan syarat-syarat, *workshop design*, dan fase implemenasi dengan pengujian *black box*. UML (*Unfied Modelling Language*) sebagai desain perancangan proses. Bahasa pemograman yang digunakan adalah PHP dan MySQL sebagai *database*.

### 4.1.3 Tujuan Sistem

Tujuan dari dibuatnya sistem web e-*commerce* aplikasi penjualan Pencarian Makanan adalah, untuk memudahkan

pengguna internet dalam mendapatkan informasi lengkap Makanan, merancang sistem yang dapat digunakan sebagai media promosi produk Makanan, merancang sistem yang dapat melakukan pemesanan Pencarian Makanan.

### 4.1.4. Sistem Usulan

Berdasarkan dari identifikasi masalah dan tujuan sistem, maka peneliti membuat sebuah sistem usulan dalam bentuk aplikasi web penjualan Makanan (Pencarian Makanan). Pada sistem ini terdapat tiga *user* (pengguna), yaitu *admin*, pengunjung dan *member*. masing-masing user memiliki hak akses yang berbeda, maka peneliti akan menjelaskan sistem usulan dengan gambar dibawah ini :



Gambar 4. 1 Sistem Usulan Album Musik *Digital*

Sistem aplikasi web penjualan Pencarian Makanan ini diharapkan dapat membantu pengguna internet dalam melakukan pencarian judul Makanan dan pemesanan Pencarian Makanan yang lengkap.

## 4.2 Fase Workshop Desain

### 4.2.1 Perancangan Algoritma *Binary Search*

Dalam perancangan algoritma *binary search*, peneliti menggunakan *pseudocode* dibawah ini :

Tabel 4. 1 *Pseudocode* Algoritma *Binary Search*

|  |
| --- |
| **Pseudocode**  **Deklarasi**  Jumlah = n  Labels = array [1..n] of integer  *Input* = key,  low = 0,  high = array [0..n],  middle of integer  **Algoritma**  BinarySearch (labels[], low, high, key)  **While** low > high **do**  Mid = (low + high) / 2 = key  **If** labels[mid] > key **then**  Mid = high – 1  **Else if** labels [mid] < key **then**  Mid = low + 1  **Else**  **Return** mid  **End if**  **End while**  **Return false**  **Print(BinarySearch)** |

Cara kerja dari algoritma pencarian judul Makanan pada aplikasi web Pencarian Makanan ini menggunakan algoritma *binary search* dan data judul Makanan harus sudah terurut secara *ascending* di dalam *database* fileMakanan.txt.

Algoritma *binary search* pada pencarian judul Makanan bekerja dengan cara memilih elemen judul Makanan pada *database* fileMakanan.txt dengan nilai *array* tengah dari tabel Makanan dan membandingkannya dengan data judul Makanan yang dicari. Jika elemen tersebut lebih rendah atau lebih tinggi maka tabel tersebut dibagi dua dan bagian tabel yang bersesuaian akan diproses kembali secara rekursif. Dibawah ini merupakan cara kerja dari algoritma *binary search* terhadap pencarian judul Makanan.

Tabel 4. 2 Tabel Data Makanan - Kasus 1

|  |  |
| --- | --- |
| **Array** | **judul\_Makanan** |
| [0]  **Low** | Ayam Bakar Khas Kuno |
| [1] | Nasi Ayam Jamur |
| [2] | Paket Ayam Goreng |
| [3] | Gulai Ayam |
| [4]  **Mid** | Kelinci Rica-Rica |
| **[5]** | **Paket Ikan Kembung** |
| [6] | Bubur Ayam Komplit |
| [7] | Nasi Goreng Ikan Asin |
| [8] | Cumi Asam Manis |
| [9]  **High** | Lontong Sayur Komplit |
| [10] | Sate ayam |

Tabel 4. 3 Tabel Pembahasan Kasus 1 (Mid = Key)

|  |  |  |
| --- | --- | --- |
|  |  | **Kasus 1 (Mid = Key)** |
| *Key* | = | **Ikan Kembung**, A[5] |
| *Low* | = | 0 |
| *High* | = | 10 |
| *Mid* | = | Low + High / 2 |
|  | = | 0 + 10 / 2 |
|  | = | 5 |
| **A[*Mid*]** | **=** | **A[5] = Ikan Kembung = Key = A[5] = Ikan Kembung** |
|  |  | Judul Makanan “**Paket Ikan Kembung**” ditemukan berada pada baris *array* ke 5 |

Tabel 4. 4 Tabel Data Makanan - Kasus 2

|  |  |
| --- | --- |
| **Array** | **judul\_Makanan** |
| [0]  **Low** | Ayam Bakar Khas Kuno |
| [1]  **Mid** | Nasi Ayam Jamur |
| **[2]** | **Paket Ayam Goreng** |
| [3] | Gulai Ayam |
| [4] | Kelinci Rica-Rica |
| [5] | Paket Ikan Kembung |
| [6] | Bubur Ayam Komplit |
| [7] | Nasi Goreng Ikan Asin |
| [8] | Cumi Asam Manis |
| [9]  **High** | Lontong Sayur Komplit |
| [10] | Sate ayam |

Tabel 4. 5 Tabel Pembahasan Kasus 2 (Mid > Key)

|  |  |  |
| --- | --- | --- |
|  |  | **Kasus 2 (Mid > Key)** |
| Key | = | **Ayam Goreng**, A[2] |
| Low | = | 0 |
| High | = | 10 |
| Mid | = | Low + High / 2 |
|  | = | 0 + 10 / 2 |
|  | = | 5 |
| A[Mid] | = | **A [5] = Paket Ikan Kembung** |
| A[Mid] | = | **A [5] = Ikan Kembung > Key = A[2] Paket Ikan Kembung** |
|  |  |  |
| **Loop 1** | = | High = Mid – 1 |
|  |  | High = 5 – 1 |
|  |  | High = 4 |
| Mid | = | 0 + 4 / 2 |
|  | = | 2 |
| A[Mid] | = | **A [2] = Ayam Goreng = A [2] = Ayam Goreng** |
|  |  | Judul Makanan “**Paket Ayam Goreng**” ditemukan berada pada baris *array* ke 2 dengan 1 kali perulangan |

Tabel 4. 6 Tabel Data Makanan - Kasus 3

|  |  |
| --- | --- |
| **Array** | **judul\_Makanan** |
| [0]  **Low** | Ayam Bakar Khas Kuno |
| [1] | Nasi Ayam Jamur |
| [2] | Paket Ayam Goreng |
| [3] | Gulai Ayam |
| [4] | Kelinci Rica-Rica |
| [5] | Paket Ikan Kembung |
| [6] | Bubur Ayam Komplit |
| [7]  **Mid** | Nasi Goreng Ikan Asin |
| **[8]** | **Cumi Asam Manis** |
| [9]  **High** | Lontong Sayur Komplit |
| [10] | Sate ayam |

Tabel 4. 7 Tabel Pembahasan Kasus 3 (Mid < Key)

|  |  |  |
| --- | --- | --- |
|  |  | **Kasus 3 (Mid < Key)** |
| Key | = | **Cumi**, A[8] |
| Low | = | 0 |
| High | = | 10 |
| Mid | = | Low + High / 2 |
|  | = | 0 + 10 / 2 |
|  | = | 5 |
| A[Mid] | = | **A [5] = Ikan Kembung** |
| A[Mid] | = | **A [5] = Ikan Kembung < Key = A[8] Ikan Kembung** |
|  |  |  |
| **Loop 1** | = | Low = Mid + 1 |
|  |  | Low = 5 + 1 |
|  |  | Low = 6 |
| Mid | = | 6 + 10 / 2 |
|  | = | 8 |
| A[Mid] | = | **A [8] = Cumi = Key = A[8] = Cumi** |
|  |  | Judul Makanan “**Cumi Asam Manis**” ditemukan berada pada baris *array* ke 8 dengan 1 kali perulangan |

#### 4.2.1.1 Flowchart Algoritma Binary Search



Gambar 4. 2 Gambar *Flowchart Algoritma Binary Search*

### 4.2.2. Pengkodean Algoritma *Binary Search*

Tabel 4. 8. Pengkodean Algoritma Binary Search dengan Bahasa Pemograman PHP

<?php

function BinarySearch( $array, $key, $low, $high )

{

if( $low > $high ) // termination case

{

return -1;

}

$middle = intval( ( $low+$high )/2 ); // gets the middle of the array

if ( $array[$middle] == $key ) // if the middle is our key

{

return $middle;

}

elseif ( $array[$middle] > $key) // our key might be in the left sub-array

{

return BinarySearch( $array, $key, $low, $middle-1 );

}

elseif ( $array[$middle] < $key)

{

return BinarySearch( $array, $key, $high, $middle+1 ); // our key might be in the right sub-array

}

}

?>

### 4.2.3 Perancangan Proses

#### 4.2.3.1 UML (Unfied Modelling Langguage)

##### Desain *Usecase Diagram*

Dibawah ini adalah Diagram *Use case* yang menggambarkan interaksi antara pengguna dan sistem. Berikut ini adalah langkah-langkah dalam membuat diagram *Use case,* yaitu:

1. Identifikasi Aktor

Dibawah ini adalah diagram *use case* yang menggambarkan interaksi anatara pengguna dan sistem. Berikut ini adalah langkah-langkah dalam membuat diagram *use case*, yaitu :

Tabel 4. 9 Deskripsi Aktor *Use case*

|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Deskripsi** |
| 1**.** | *Owner* | Aktor yang telah terdaftar dalam sistem yang hanya melihat dan mendowload laporan. |
| 2**.** | *Admin* | Aktor yang telah terdaftar dalam sistem memiliki hak akses untuk use case menu dan use case pesanan. |

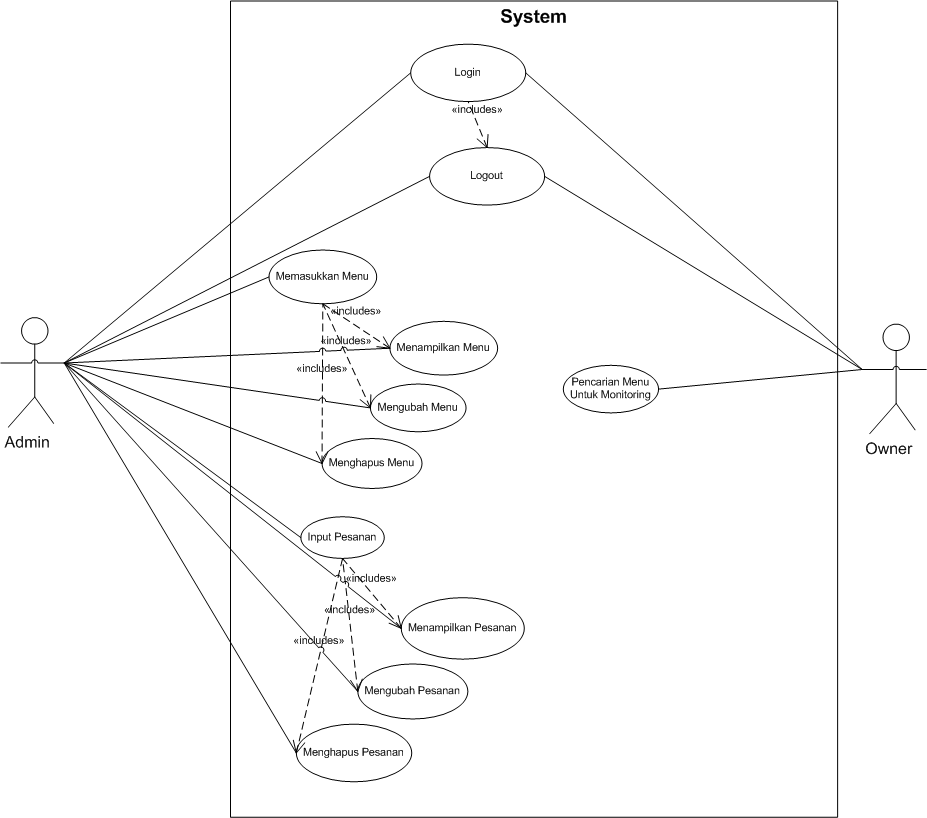
1. Identifikasi *Use case*

Dibawah ini adalah tabel identifikasi *use case.* Sebagai berikut :

Tabel 4. 10 Identifikasi *Use case*

| No | *Use case* | Aktor | Deskripsi |
| --- | --- | --- | --- |
| 1. | Login | *Owner dan* *Admin* | *Use Case* menggambarkan kegiatan *Login*, dengan memasukkan *username* dan *password.* |
| 2. | Memasukkan Menu | *Admin* | *Use Case* menggambarkan kegiatan memasukkan menu |
| 3. | Menampilkan Menu | *Admin* | *Use Case* menggambarkan kegiatan menampilkan menu |
| 4. | Mengubah Menu | *Admin* | *Use Case* menggambarkan kegiatan mengubah menu |
| 5. | Menghapus Menu | *Admin* | *Use Case* menggambarkan kegiatan menghapus menu |
| 6. | Memasukkan Pesanan | *Admin* | *Use Case* menggambarkan kegiatan memasukkan pesanan |
| 7. | Menampilkan Pesanan | *Admin* | *Use Case* menggambarkan kegiatan menampilkan pesanan |
| 8. | Mengubah Pesanan | *Admin* | *Use Case* menggambarkan kegiatan mengubah pesanan |
| 9. | Menghapus Pesanan | *Admin* | *Use Case* menggambarkan kegiatan menghapus pesanan |
| 10. | Pencarian Menu Untuk Monitoring Keuangan | *Owner* | *Use Case* menggambarkan kegiatan monitoring keungan berdasarkan menu yang dicari. |

1. *Diagram Use case*



Gambar 4. 3 Diagram *Use case* Monitoring Keuangan

1. Narasi *Use case*

Dibawah ini adalah narasi *use case* dari kegiatan bisnis dan *use case* diagram yang ada berdasarkan usulan sistem.

1. Narasi *Use case Login*

Tabel 4. 11 Narasi *Use case Login* *Owner dan* *Admin*

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Login | |
| ***Use case* Id** | 1 | |
| **Actor** | *Owner* *dan* *Admin* | |
| **Description** | *Use Case* menggambarkan kegiatan *Login*, dengan memasukkan *username* dan *password.* | |
| **Precondition** | Aktor memilih aplikasi | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih aplikasi  3.Mengisi *user*name dan password  4.Klik submit | Respon Sistem  2.Menampilkan halaman *login*  5.Menampilkan halaman index |
| **Alterate Course** | jika *username* dan *password* salah maka akan kembali ke halaman *login* dan aktor harus *inputusername* dan *password*kembali. | |
| **Conclusion** | Aktor berhasil *login* kedalam sistem | |
| **Post Condition** | Menampilkan halaman index dalam sistem sesuai dengan hak akses | |

1. Narasi *Use case* Memasukkan Menu

Tabel 4. 13 Narasi *Use case* Memasukkan Menu

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Memasukkan Menu | |
| ***Use case* Id** | 2 | |
| **Actor** | *Admin* | |
| **Description** | *Use Case* menggambarkan kegiatan memasukkan menu | |
| **Precondition** | Aktor telah *login* | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih menu menu  2.Pilih submenu masukkan menu  4.Mengisi form masukkan menu | Respon Sistem  3.Menampilkan form masukkan menu  5.Memasukkan data menu kedalam *database* |
| **Alterate Course** | 5.jika data yang dimasukkan terdapat kesalahan maka akan kembali ke halaman form masukkan menu | |
| **Conclusion** | Aktor berhasil memasukkan menu | |
| **Post Condition** | Menu tersimpan dalam *database* | |

1. Narasi *Use case* Menampilkan Menu

Tabel 4. 15 Narasi *Use case* Menampilkan Menu

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Menampilkan Menu | |
| ***Use case* Id** | 3 | |
| **Actor** | *Admin* | |
| **Description** | *Use Case* menggambarkan kegiatan menampilkan menu | |
| **Precondition** | Aktor telah *login* | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih menu menu  2.Pilih submenu tampilkan menu | Respon Sistem  3.Menampilkan menu yang ada dalam *database* |
| **Alterate Course** | 3.jika dalam *database* menu tidak ada maka akan menampilkan tabel kosong | |
| **Conclusion** | Aktor berhasil menampilkan menu | |
| **Post Condition** |  | |

1. Narasi *Use case* Mengubah Menu

Tabel 4. 16 Narasi *Use case* Mengubah Menu

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Mengubah Menu | |
| ***Use case* Id** | 4 | |
| **Actor** | *Admin* | |
| **Description** | *Use Case* menggambarkan kegiatan mengubah menu | |
| **Precondition** | Aktor telah *login* | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih menu menu  3.Pilih ubah menu dengan menu yang diinginkan  4.Ubah menu | Respon Sistem  2.Menampilkan menu yang ada dalam *database*  3.Tampilkan menu yang akan dirubah  5.Mengubah menu pada *database* |
| **Alterate Course** | 7.jika menu yang dimaksud tidak ada maka akan kembali ke halaman menu | |
| **Conclusion** | Aktor berhasil mengubah menu | |
| **Post Condition** | Menu yang dimaksud akan berubah | |

1. Narasi *Use case* Menghapus Menu

Tabel 4. 17 Narasi *Use case* Menghapus Menu

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Menghapus Menu | |
| ***Use case* Id** | 5 | |
| **Actor** | *Owner* | |
| **Description** | *Use Case* menggambarkan kegiatan menghapus*user* | |
| **Precondition** | Aktor telah *login* | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih menu menu  3.Pilih hapus dengan menu yang diinginkan  5.Pilih menu yang akan dihapus | Respon Sistem  2.Menampilkan menu yang ada dalam *database*  4.Tampilkan pesan peringatan karena menu akan dihapus  6.Menu dengan id yang dipilih terhapus dari tabel menu  7.Menampilkan menu dari *database* yang belum dihapus |
| **Alterate Course** | 6.jika id menu yang dimaksud tidak ada maka akan kembali ke halaman hapus menu | |
| **Conclusion** | Aktor berhasil menghapus menu | |
| **Post Condition** | Tabel menu berkurang satu | |

1. Narasi *Use case* Memasukkan Pesanan

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Memasukkan Pesanan | |
| ***Use case* Id** | 6 | |
| **Actor** | *Admin* | |
| **Description** | *Use Case* menggambarkan kegiatan memasukkan pesanan | |
| **Precondition** | Aktor telah *login* | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih menu pesanan  2.Pilih submenu masukkan pesanan  4.Mengisi form masukkan pesanan | Respon Sistem  3.Menampilkan form masukkan pesanan  5.Memasukkan pesanan kedalam *database* |
| **Alterate Course** | 5.jika data yang dimasukkan terdapat kesalahan maka akan kembali ke halaman form masukkan menu | |
| **Conclusion** | Aktor berhasil memasukkan pesanan | |
| **Post Condition** | Pesanan tersimpan dalam *database* | |

1. Narasi *Use case* Menampilkan Pesanan

Tabel 4. 18 Narasi *Use case* Menampilkan Pesanan

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Menampilkan Pesanan | |
| ***Use case* Id** | 7 | |
| **Actor** | *Admin* | |
| **Description** | *Use Case* menggambarkan kegiatan menampilkan menu | |
| **Precondition** | Aktor telah *login* | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih menu menu  2.Pilih submenu tampilkan menu | Respon Sistem  3.Menampilkan menu yang ada dalam *database* |
| **Alterate Course** | 3.jika dalam *database* menu tidak ada maka akan menampilkan tabel kosong | |
| **Conclusion** | Aktor berhasil menampilkan menu | |
| **Post Condition** |  | |

1. Narasi *Use case* Mengubah Pesanan

Tabel 4. 19 Narasi *Use case* Mengubah Pesanan

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Mengubah Pesanan | |
| ***Use case* Id** | 8 | |
| **Actor** | *Admin* | |
| **Description** | *Use Case* menggambarkan kegiatan mengubah pesanan | |
| **Precondition** | Aktor telah *login* | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih menu pesanan  3.Pilih ubah dengan pesanan yang diinginkan  5.Ubah pesanan | Respon Sistem  2.Menampilkan pesanan yang ada dalam *database*  4.Tampilkan pesanan yang akan dirubah  6.Mengubah pesanan pada *database* |
| **Alterate Course** | 4.jika pesanan yang dimaksud tidak ada maka akan kembali ke halaman menu | |
| **Conclusion** | Aktor berhasil mengubah menu | |
| **Post Condition** | Menu yang dimaksud akan berubah | |

1. Narasi *Use case* Menghapus Pesanan

Tabel 4. 20 Narasi *Use case* Menghapus Pesanan

|  |  |  |
| --- | --- | --- |
| ***Use case* Name** | Menghapus Pesanan | |
| ***Use case* Id** | 9 | |
| **Actor** | *Admin* | |
| **Description** | *Use Case* menggambarkan kegiatan menghapus pesanan | |
| **Precondition** | Aktor telah *login* | |
| **Typical Course of Events** | Aksi Aktor  1.Pilih menu pesanan  3.Pilih hapus dengan pesanan yang diinginkan  5.Pilih ya | Respon Sistem  2.Menampilkan pesanan yang ada dalam *database*  4.Tampilkan pesan peringatan karena pesanan akan dihapus  6.Pesanan dengan id yang dipilih terhapus dari tabel pesanan  7.Menampilkan pesanan dari *database* yang belum dihapus |
| **Alterate Course** | 4.jika pesanan yang dimaksud tidak ada maka akan kembali ke halaman pesanan | |
| **Conclusion** | Aktor berhasil menghapus pesanan | |
| **Post Condition** | Tabel pesanan berkurang satu | |

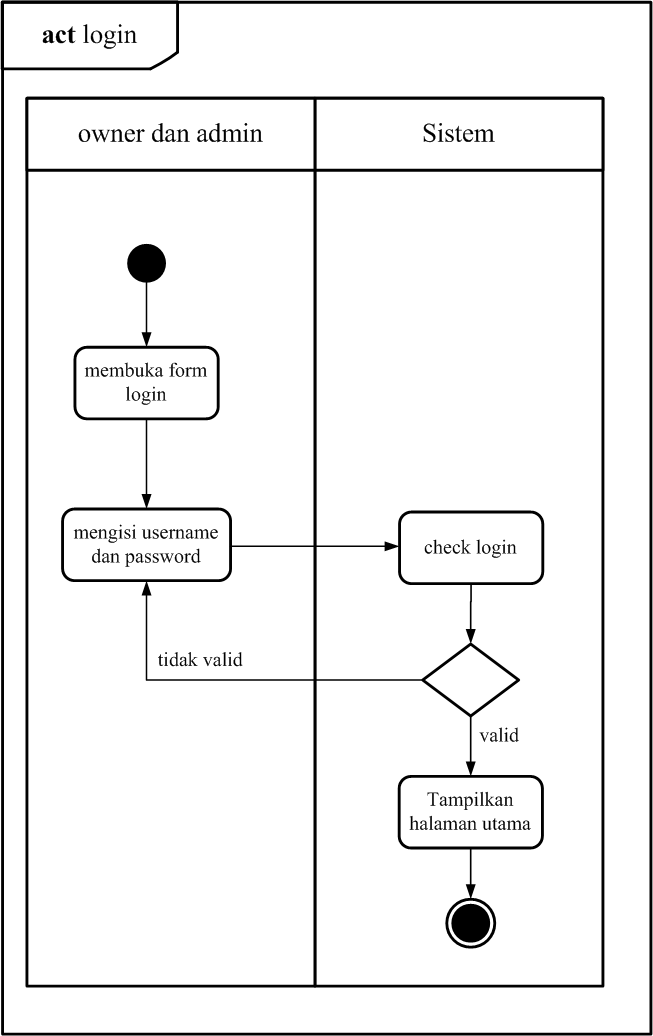
1. Narasi *Use case* Pencarian Menu Untuk Monitoring

Tabel 4. 21 Narasi *Use case* Pencarian Menu Untuk Monitoring

|  |  |
| --- | --- |
| ***Use case name*** | Pencarian Menu Untuk Monitoring |
| ***Use case id*** | 10 |
| ***Actor*** | *Owner* |
| ***Description*** | *Use case* menggambarkan kegiatan monitoring berdasarkan menu yang dicari |
| ***Precondition*** | Aktor memilih aplikasi |
| ***Typical Course of Events*** | |  |  | | --- | --- | | Aksi Aktor | Respon Sistem | | 1.Pilih menu pencarian  3.Memasukkan menu yang akan dicari laporan keuangannya  5.Mendownload hasil pencarian berupa pdf | 2.Menampilkan textbox pencarian  4.Mencari menu menggunakan binary search dan menampilkan berupa laporan  6.Merubah laporan menjadi pdf | |
| ***Alternate Course*** | - |
| ***Conclusion*** | Aktor berhasil mencari laporan sesuai dengan menu yang dimasukkan |
| ***Post Condition*** | Laporan sesuai dengan menu yang dicari berhasil ditampilkan |

##### Desain *Activity diagram*

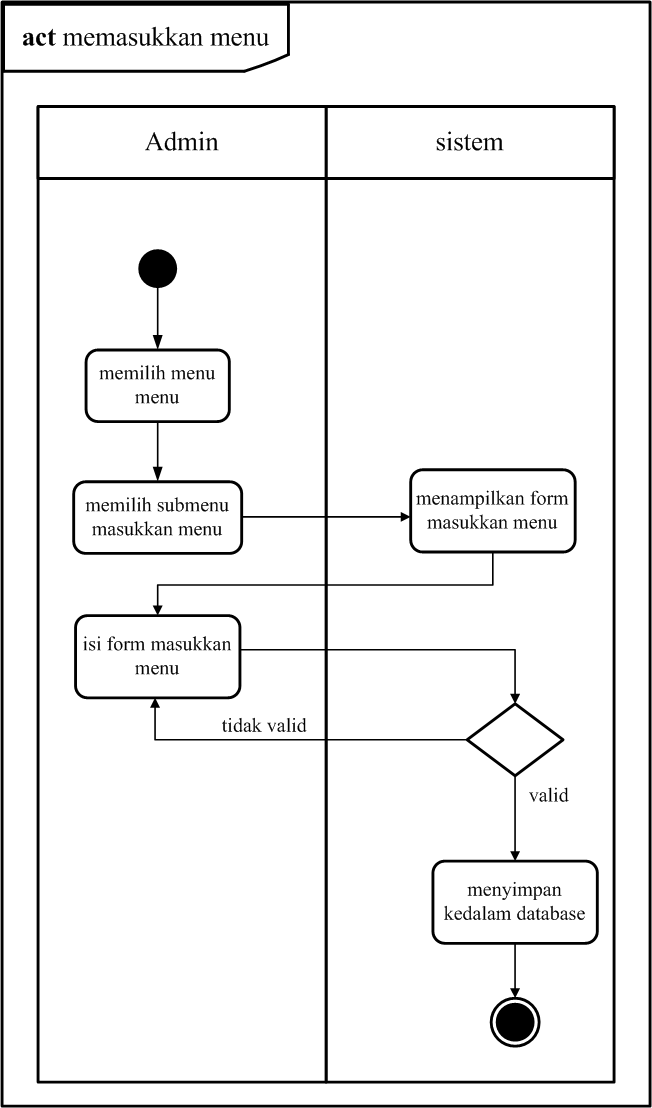
1. *Login*



Gambar 4. 4 Gambar *Desain Activity diagram Login*

Aktifitas yang terjadi pada *Activity diagram login,* adalah aktifitas aktor yang telah terdaftar dalam sistem, diawali dengan aktor mengakses *url login* dan akan menampilkan halaman *login.* Selanjutnya aktor menginpu t*username* dan *password* yang telah terdaftar dalam sistem, aktor dapat menekan tombol *submit* untuk masuk kedalam sistem. Sistem akan melakukan proses validasi, apabila *username* atau *password* tidakvalid, maka sistem akan menolak masuk dan akan menampilkan kembali halaman *login* untuk diisi kembali *username* dan *password*, dan jika *username* dan *password* valid, makasistem akan menampilkan halaman *home* dari aktor yang melakukan *login*.

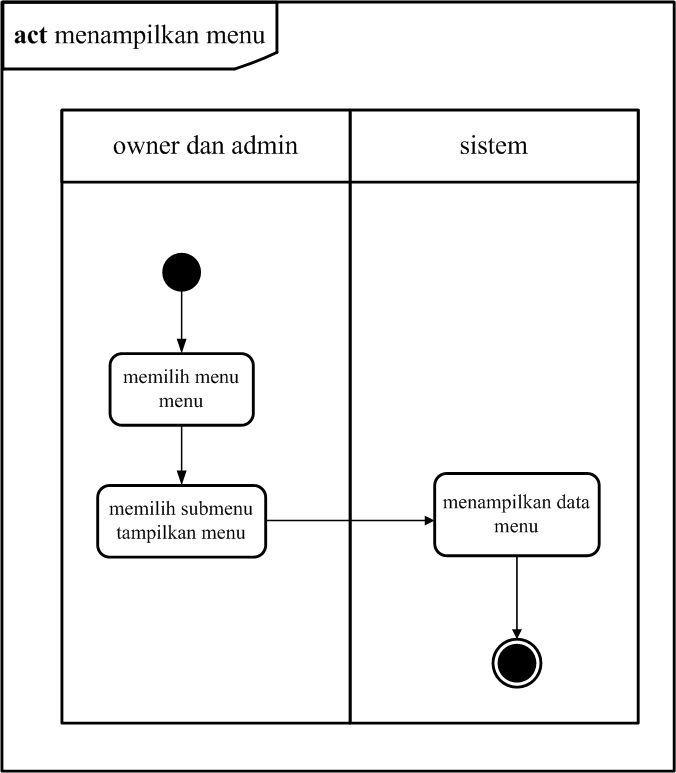
1. Memasukkan Menu



Gambar 4. 5 *Gambar Desain Activity diagram* Memasukkan Menu

Aktifitas yang terjadi pada Activity diagram Memasukkan Menu diawali dengan aktor memilih menu menu, lalu pilih submenu masukkan menu. Sistem akan menampilkan form untuk memasukkan menu, setelah diisi klik submit maka data yang dikirim akan di cek apakah tidak melanggar validasi atau tidak, jika ya maka akan kembali ke form memasukkan menu, jika tidak data menu akan tersimpan dalam database.

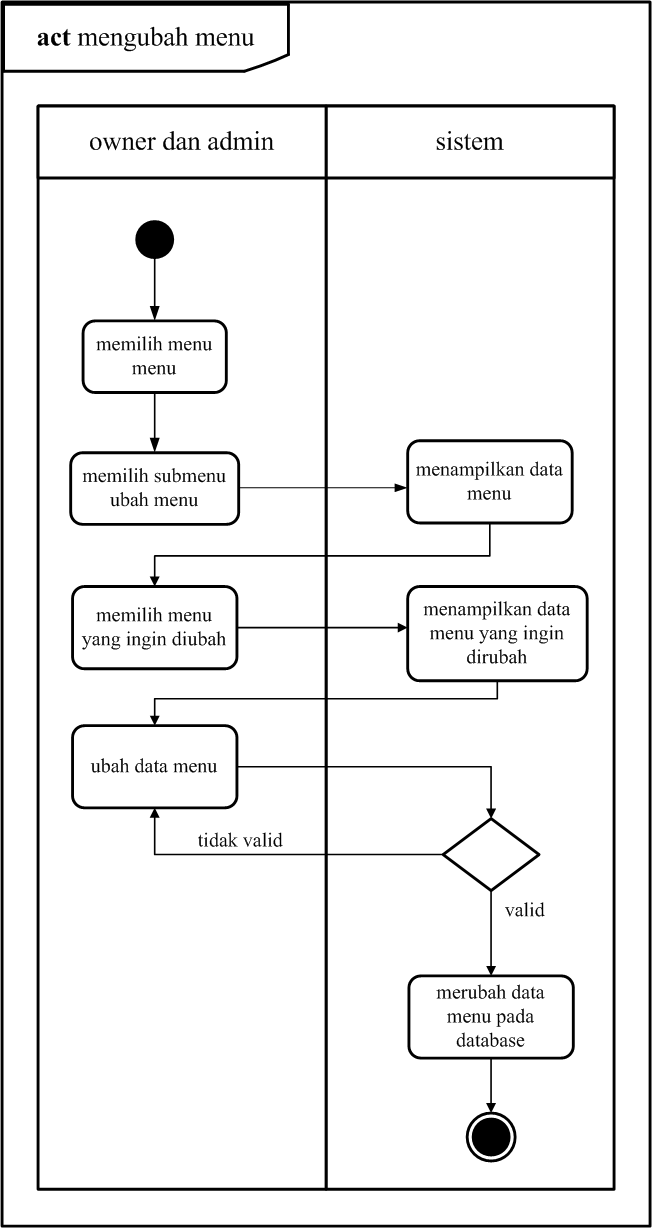
1. *Manage* Data Makanan (*Admin*)



Gambar 4. 6 Gambar *Desain Activity diagram* Menampilkan Menu

Aktifitas yang terjadi pada Activity diagram Menampilkan Menu diawali dengan aktor memilih menu menu, lalu pilih submenu tampilkan menu. Sistem akan menampilkan seluruh menu dari *database* pada halaman tampil menu.

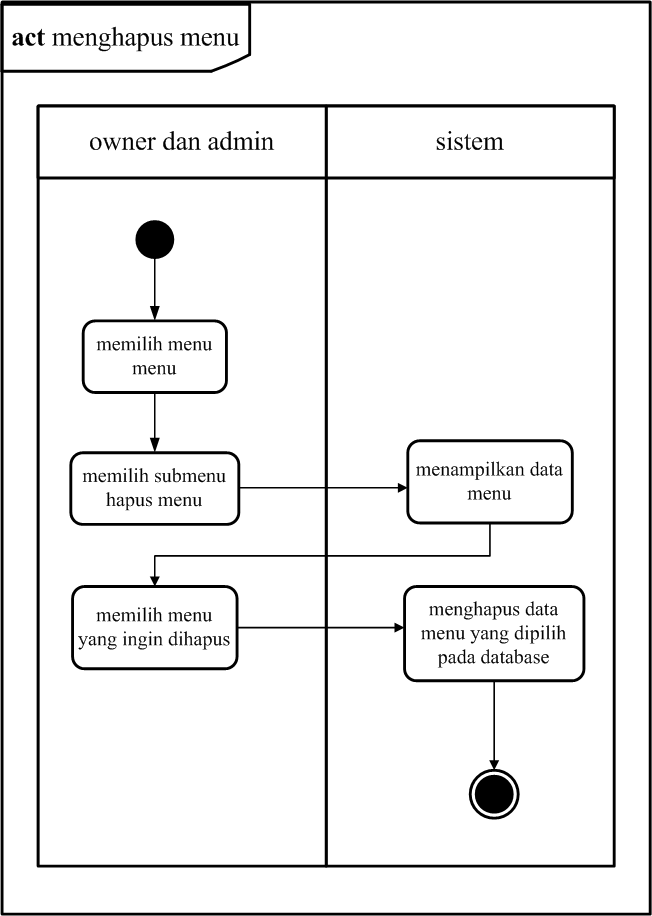
1. Mengubah Menu



Gambar 4. 7 Gambar *Desain Activity diagram* Mengubah Menu

Aktifitas yang terjadi pada Activity diagram Mengubah Menu diawali dengan aktor memilih menu menu, lalu pilih submenu ubah menu. Sistem akan menampilkan seluruh menu dari *database* pada halaman ubah menu. *Owner* akan memilih menu yang ingin dirubah setelah dipilih, sistem akan mencari menu pada *database* berdasarkan id\_menu yang telah dipilih. Setelah mendapatkan menu berdasarkan id\_menu yang telah dipilih sistem akan menampilkan kesuluruhan data pada form ubah data menu. Selanjutnya *admin* mengubah salah satu atau semua data menu, setelah selesai mengubah *admin* klik ubah. Setelah itu sistem akan melakukan validasi terhadap data menu yang telah dirubah jika ada yang salah maka akan kembali ke form ubah menu, jika tidak ada yang salah maka akan tersimpan dalam *database*.

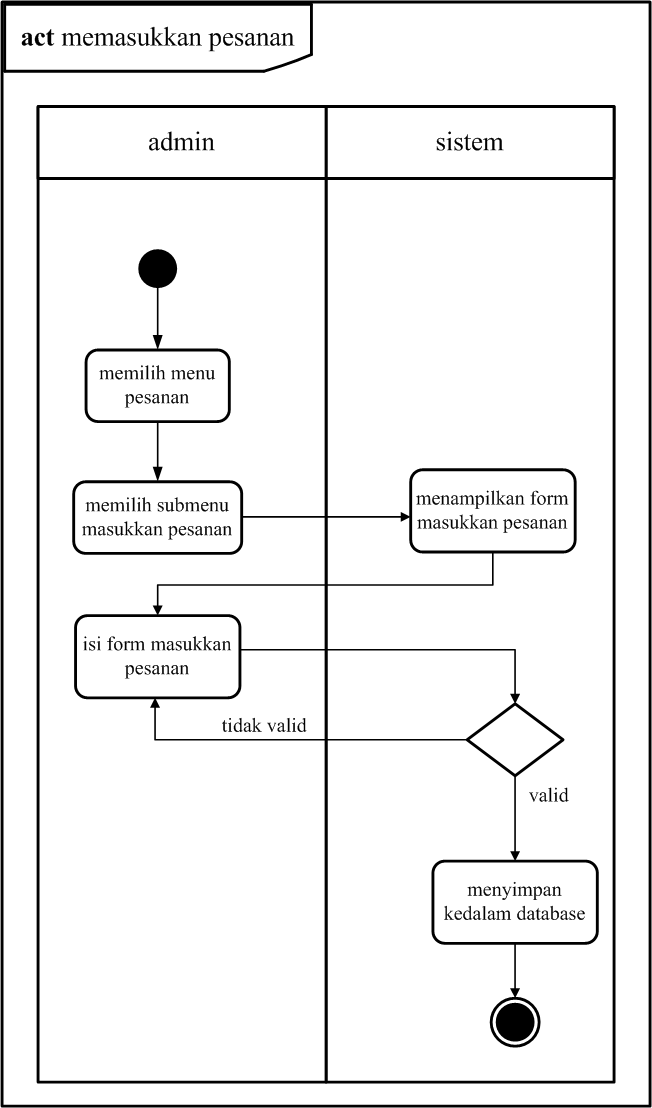
1. Menghapus Menu



Gambar 4. 8 Gambar *Desain Activity diagram* Menghapus Menu

Aktifitas yang terjadi pada *Activity diagram* Menghapus Menudiawali dengan aktor memilih menu menu, lalu pilih *submenu* hapus menu. Sistem akan menampilkan seluruh menu dari *database* pada halaman hapus menu. *Owner* akan memilih menu yang ingin dihapus setelah dipilih, sistem akan memberikan peringatan serta pertanyaan apakah menu dengan id\_menu yang dipilih ingin dihapus jika menjawab tidak maka akan kembali ke halaman hapus menu dan jika ya maka akan dihapus dari *database*.

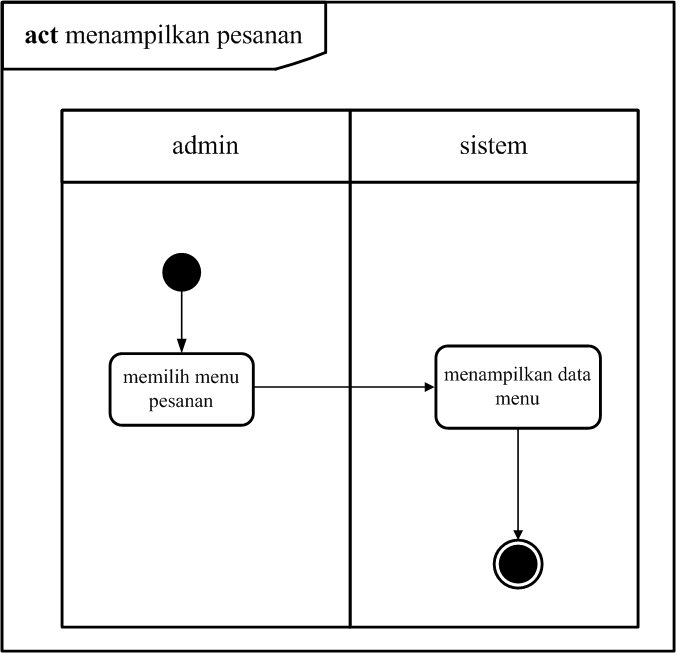
1. Memasukkan Pesanan



Gambar 4. 9 Gambar Desain *Activity diagram* Memasukkan Pesanan

Aktifitas yang terjadi pada *Activity diagram* Memasukkan Pesanandiawali dengan aktor memilih menu pesanan, lalu pilih *submenu* masukkan pesanan. Sistem akan menampilkan form untuk memasukkan pesanan, setelah diisi klik submit maka data yang dikirim akan di cek apakah tidak melanggar validasi atau tidak, jika ya maka akan kembali ke form memasukkan pesanan, jika tidak data pesanan akan tersimpan dalam *database*.

1. Menampilkan Pesanan



Gambar 4. 10 Gambar Desain *Activity diagram* Menampilkan Pesanan

Aktifitas yang terjadi pada Activity diagram Menampilkan Pesanan diawali dengan aktor memilih menu pesanan, lalu pilih submenu tampilkan pesanan. Sistem akan menampilkan seluruh pesanan dari *database* pada halaman tampil pesanan.

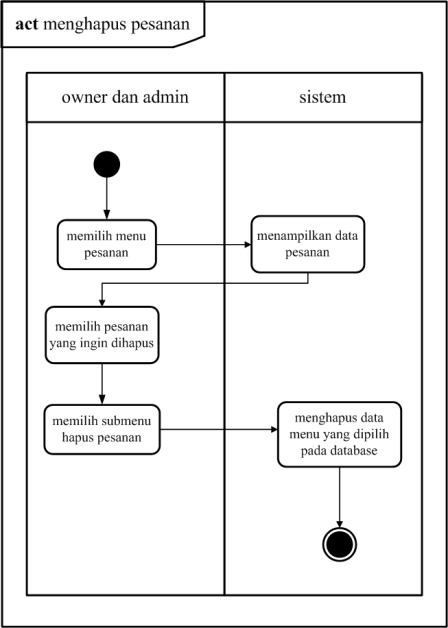
1. Mengubah Pesanan



Gambar 4. 11 Gambar Desain *Activity diagram* Mengubah Pesanan

Aktifitas yang terjadi pada Activity diagram Mengubah Pesanan diawali dengan aktor memilih menu pesanan, lalu pilih submenu ubah pesanan. Sistem akan menampilkan seluruh pesanan dari *database* pada halaman ubah pesanan. *Owner* akan memilih pesanan yang ingin dirubah setelah dipilih, sistem akan mencari pesanan pada *database* berdasarkan id\_pesanan yang telah dipilih. Setelah mendapatkan pesanan berdasarkan id\_pesanan yang telah dipilih sistem akan menampilkan kesuluruhan data pada form ubah data pesanan. Selanjutnya *admin* mengubah salah satu atau semua data pesanan, setelah selesai mengubah *admin* klik ubah. Setelah itu sistem akan melakukan validasi terhadap data pesanan yang telah dirubah jika ada yang salah maka akan kembali ke form ubah pesanan, jika tidak ada yang salah maka akan tersimpan dalam *database*.

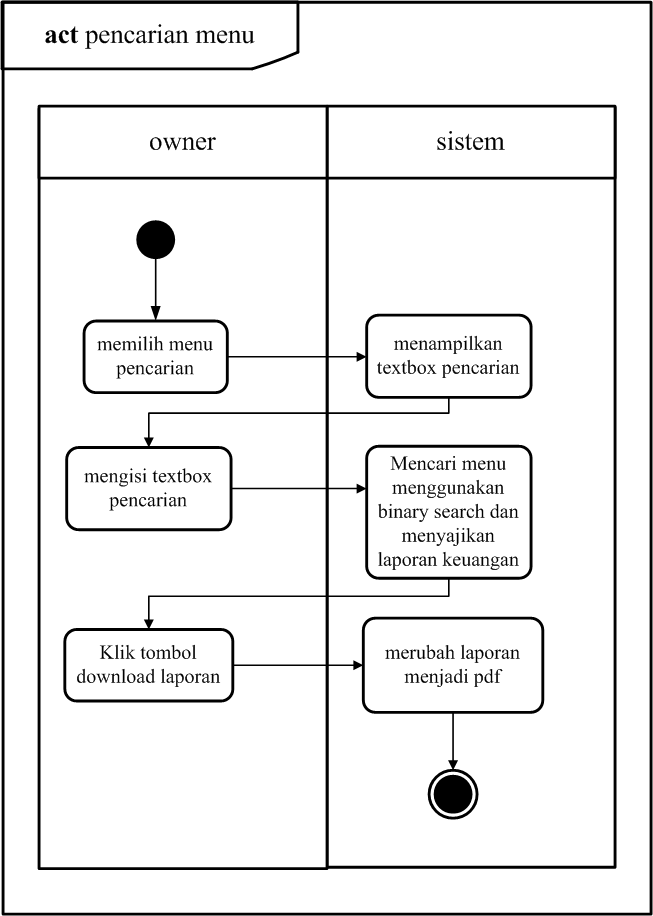
1. Menghapus Pesanan



Gambar 4. 12 Gambar Desain *Activity diagram Logout Admin*

Aktifitas yang terjadi pada *Activity diagram* Menghapus Pesanandiawali dengan aktor memilih menu pesanan, lalu pilih *submenu* hapus pesanan. Sistem akan menampilkan seluruh pesanan dari *database* pada halaman hapus pesanan. *Owner* akan memilih pesanan yang ingin dihapus setelah dipilih, sistem akan memberikan peringatan serta pertanyaan apakah pesanan dengan id\_pesanan yang dipilih ingin dihapus jika menjawab tidak maka akan kembali ke halaman hapus pesanan dan jika ya maka akan dihapus dari *database*.

1. Pencarian Menu Untuk Monitoring

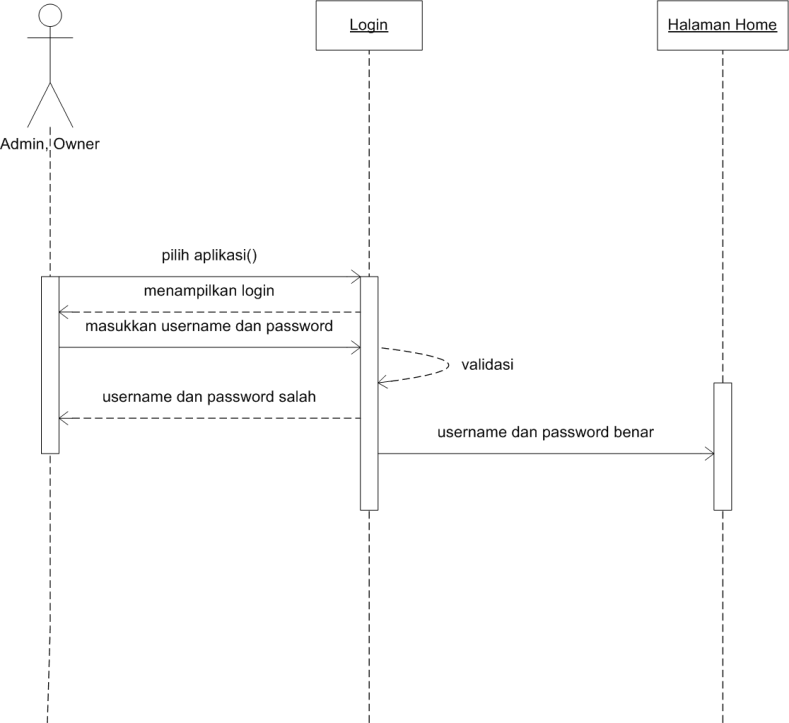


Gambar 4. 13 *Activity diagram* menghapus pesanan

Aktifitas yang terjadi pada *Activity diagram* Pencarian Menudiawali dengan aktor memilih menu pencarian. Sistem akan menampilkan kolom pencarian pada halaman pencarian. *Owner* akan memasukkan menu yang akan dicari dan sistem akan mencari menu menggunakan binary search dan menampilkan hasil pencarian berupa laporan keuangan sesuai dengan menu yang dicari.

##### Desain *Sequence Diagram*

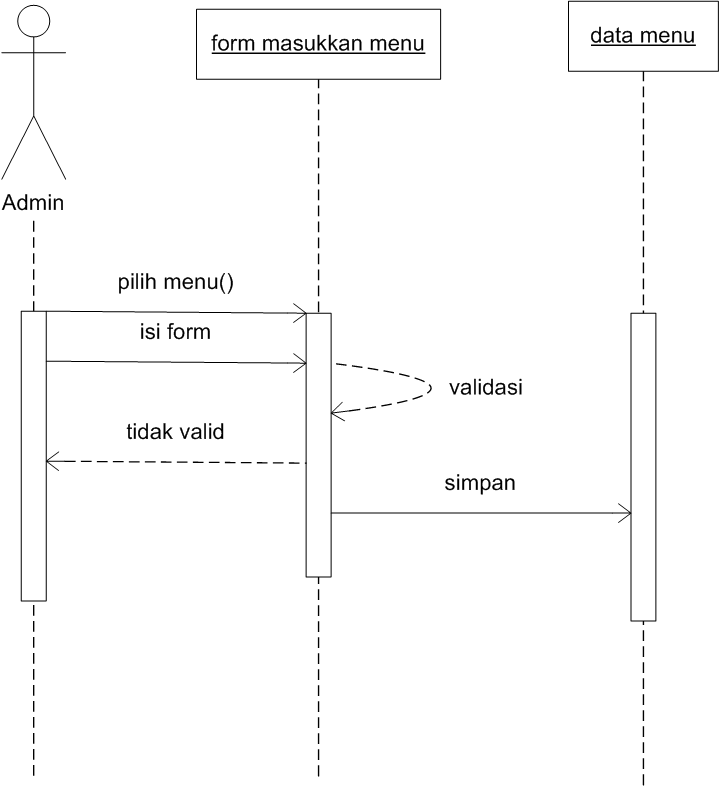
1. Login



Gambar 4. 14 Gambar Desain *Sequence Diagram Login*

*Sequence diagram* diatas menjelaskan aktor saat melakukan *login.* Terdapat dua aktor yang melakukan *login,* yaitu *Owner, Admin.* Pertama kali aktor mengetikkan *username* dan *password* pada halaman *login*, maka sistem akan memvalidasi, jika *username* dan *password* benar, maka akan menampilkan halaman *home,* jika *username* dan *password* salah, maka sistem akan menampilkan halaman *login* untuk kembali melakukan *login*.

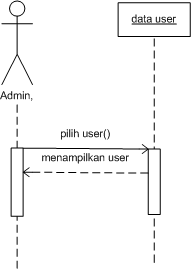
1. Memasukkan Menu



Gambar 4. 15 Gambar Desain *Sequence Diagram* Memasukkan Menu

*Sequence diagram* diatas menjelaskan aktor saat memasukkan menu*.* Aktor yang dapat mengelola menuadalah *Admin.* Pertama kali aktor memilih menu lalu tampil form masukkan menu, selanjutnya aktor memasukkan data menu, pilih tombol masukkan untuk post ke sistem. Setelah data menu dikirim ke sistem akan dilakukan validasi untuk mencegah adanya data yang tidak diinginkan, jika hasil dari validasi adalah tidak valid maka akan kembali ke halaman form masukkan menu, jika hasil dari validasi adalah valid data akan disimpan ke dalam *database*.

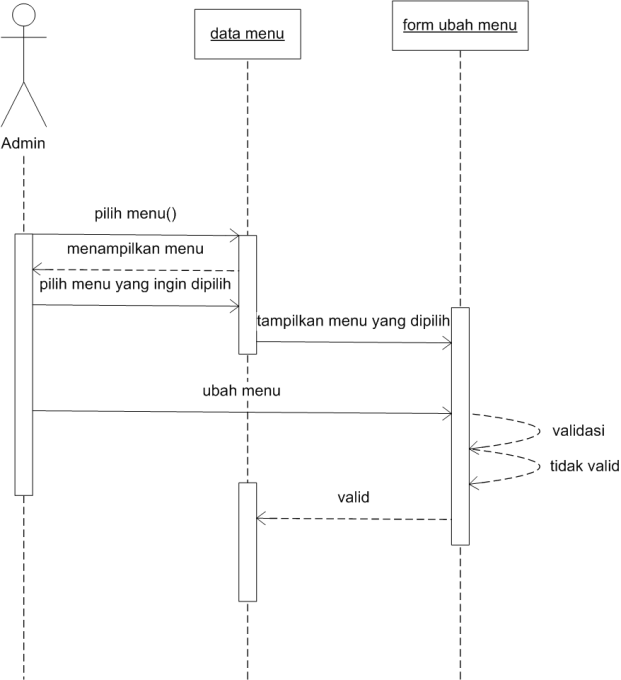
1. Menampilkan Menu



Gambar 4. 16 Gambar Desain *Sequence Diagram* Menampilkan Menu

*Sequence diagram* diatas menjelaskan aktor saat menampilkan menu*.* Aktor yang dapat menampilkan menuadalah *Admin.* Pertama kali aktor memilih menu lalu akan tampil seluruh menu dari *database*.

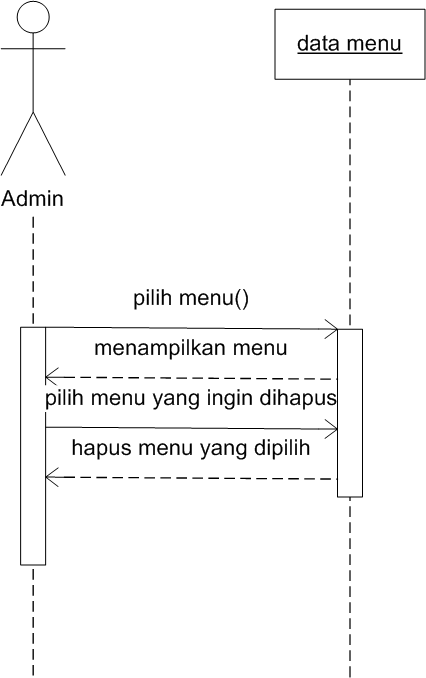
1. Mengubah Menu



Gambar 4. 17 Gambar Desain *Sequence Diagram* Mengubah Menu

*Sequence diagram* diatas menjelaskan aktor saat mengubah menu*.* Aktor yang dapat mengubah menuadalah *Admin.* Pertama kali aktor memilih menu lalu akan tampil seluruh menu dari *database*, setelah muncul pilih menu yang akan dirubah. Sistem akan mengambil id\_menu yang telah dipilih dan memunculkan seluruh data menu berdasarkan id\_menu yang telah dipilih ke dalam form ubah menu. Selanjutnya aktor akan mengubah data menu setelah selesai klik ubah. Ketika data dikirim sistem akan melakukan validasi terhadap data yang dikirim jika hasil validasi adalah salah maka akaan kembali ke form ubah menu.

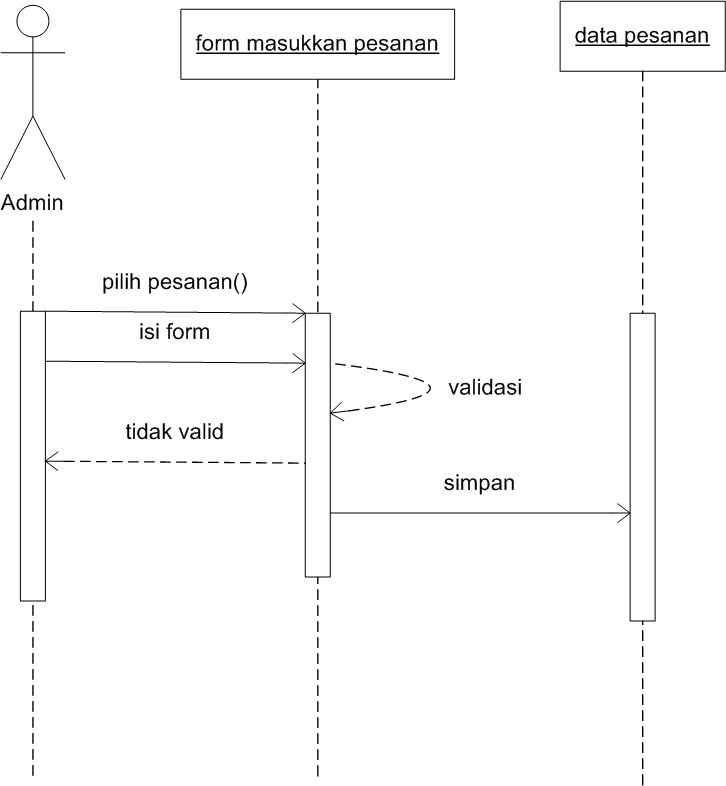
1. Menghapus Menu



Gambar 4. 18 Gambar Desain Sequence Diagram Menghapus Menu

*Sequence diagram* diatas menjelaskan aktor saat menghapus menu*.* Aktor yang dapat menghapus *user* adalah *Admin.* Pertama kali aktor memilih menu lalu akan tampil seluruh menu dari *database*, setelah muncul pilih menu yang akan dihapus. Sistem akan mengambil id\_menu yang telah dipilih dan memunculkan peringatan karena menu akan dihapus jika aktor menjawab tidak maka akan kembali ke halaman menu, jika ya maka menu berdasarkan id\_menu yang dipilih akan dihapus dari *database*.

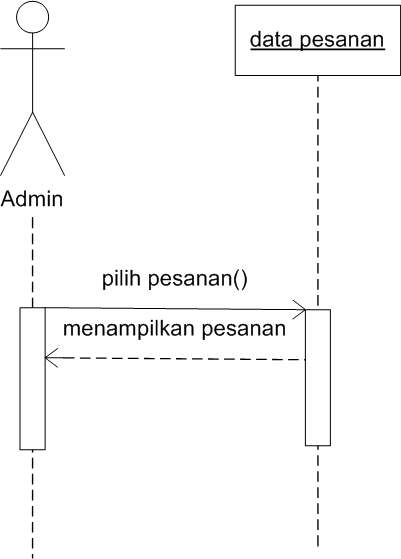
1. Memasukkan Pesanan



Gambar 4. 19 Gambar Desain *Sequence Diagram* Memasukkan Pesanan

*Sequence diagram* diatas menjelaskan aktor saat memasukkan pesanan*.* Aktor yang dapat mengelola pesanan adalah *Admin.* Pertama kali aktor memilih pesanan lalu tampil form masukkan pesanan, selanjutnya aktor memasukkan data pesanan, pilih tombol masukkan untuk post ke sistem. Setelah data pesanan dikirim ke sistem akan dilakukan validasi untuk mencegah adanya data yang tidak diinginkan, jika hasil dari validasi adalah tidak valid maka akan kembali ke halaman form masukkan pesanan, jika hasil dari validasi adalah valid data akan disimpan ke dalam *database*.

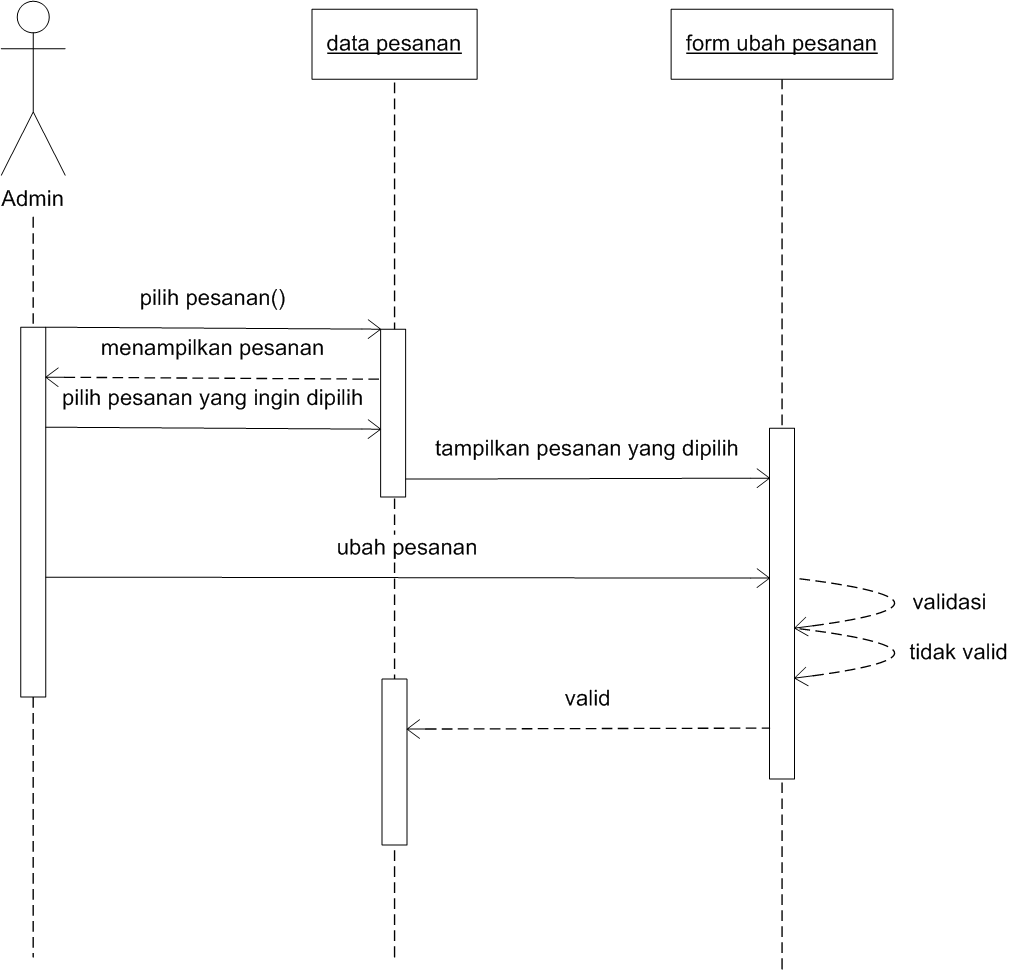
1. Menampilkan Pesanan



Gambar 4. 20 Gambar Desain *Sequence Diagram* Menampilkan Pesanan

*Sequence diagram* diatas menjelaskan aktor saat menampilkan pesanan*.* Aktor yang dapat menampilkan pesanan adalah *Admin.* Pertama kali aktor memilih pesanan lalu akan tampil seluruh pesanan dari *database*.

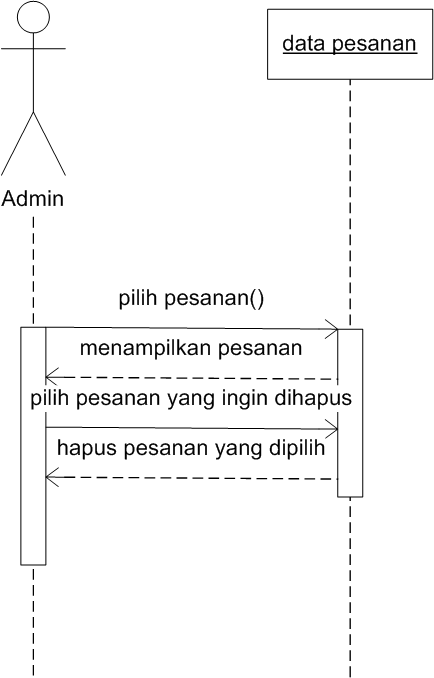
1. Mengubah Pesanan



Gambar 4. 21 Gambar Desain *Sequence Diagram* Mengubah Pesanan

*Sequence diagram* diatas menjelaskan aktor saat mengubah pesanan*.* Aktor yang dapat mengubah pesanan adalah *Admin.* Pertama kali aktor memilih pesanan lalu akan tampil seluruh pesanan dari *database*, setelah muncul pilih pesanan yang akan dirubah. Sistem akan mengambil id\_ pesanan yang telah dipilih dan memunculkan seluruh data pesanan berdasarkan id\_ pesanan yang telah dipilih ke dalam form ubah pesanan. Selanjutnya aktor akan mengubah data pesanan setelah selesai klik ubah. Ketika data dikirim sistem akan melakukan validasi terhadap data yang dikirim jika hasil validasi adalah salah maka akaan kembali ke form ubah pesanan.

1. Menghapus Pesanan



Gambar 4. 22 Gambar Desain *Sequence Diagram* Menghapus Pesanan

*Sequence diagram* diatas menjelaskan aktor saat menghapus pesanan*.* Aktor yang dapat menghapus pesanan adalah *Admin.* Pertama kali aktor memilih pesanan lalu akan tampil seluruh pesanan dari *database*, setelah muncul pilih pesanan yang akan dihapus. Sistem akan mengambil id\_pesanan yang telah dipilih dan memunculkan peringatan karena pesanan akan dihapus jika aktor menjawab tidak maka akan kembali ke halaman pesanan, jika ya maka pesanan berdasarkan id\_pesanan yang dipilih akan dihapus dari *database*.

##### Desain *Class Diagram*

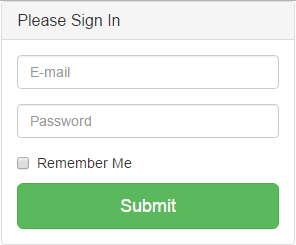
Gambar desain *class diagram* dan deskripsi singkat mengenai *desain class* diagram aplikasi monitoring keuangan berbasis web (*e-commerce*) berada pada halaman lampiran 1.

#### Mapping Database

Gambar desain *mapping database* dan spesifikasi *database* aplikasi monitoring keuangan berbasis web (*e-commerce*) berada pada halaman lampiran 2.

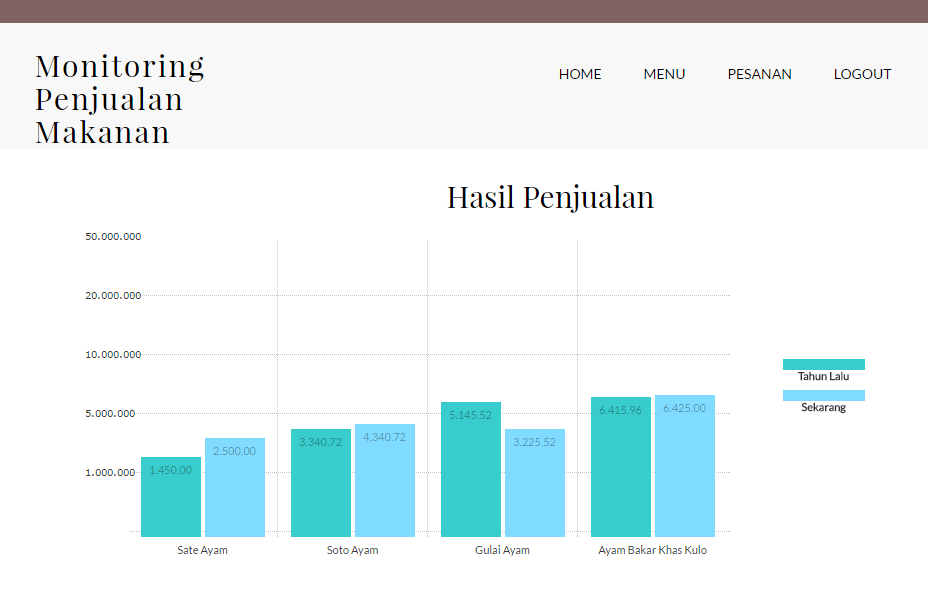
### 4.2.4 Perancangan Interface

1. *Login*



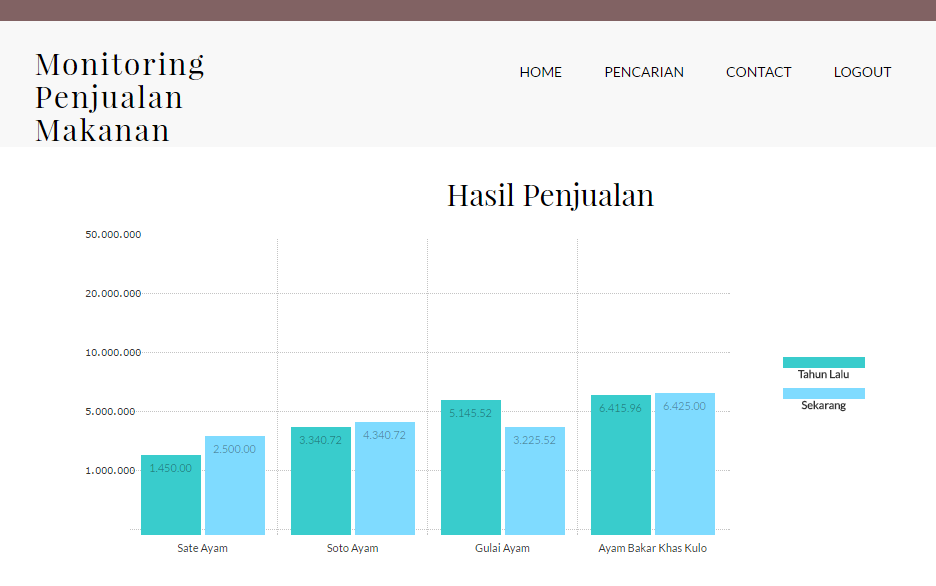
Gambar 4. 24 Gambar *User Interface Login Admin*

1. Home
2. *Admin*



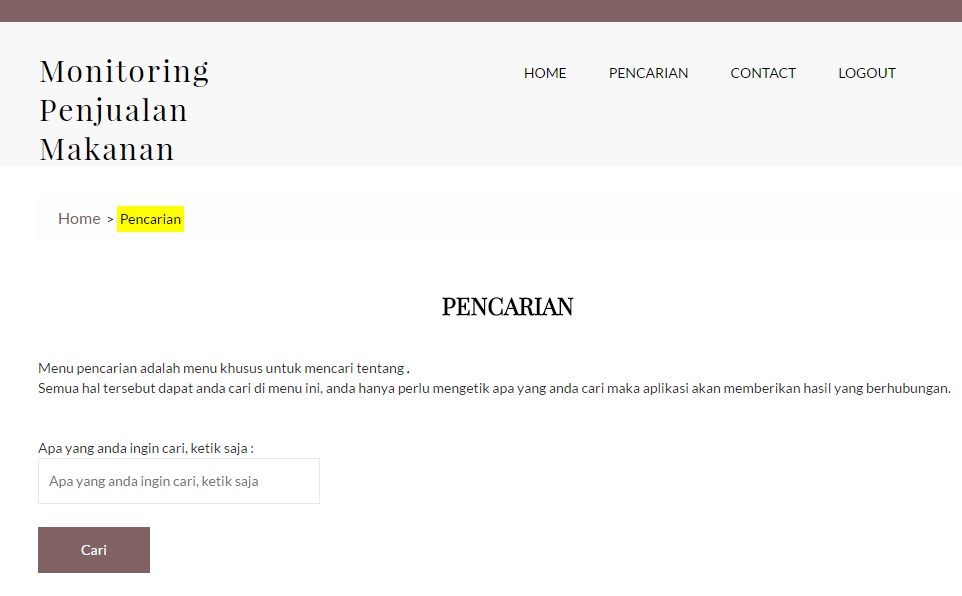
Gambar 4. 25 Gambar *User Interface* Halaman Utama *Admin*

1. *owner*



Gambar 4. 26 Gambar *User Interface* Halaman Utama *Member*

1. Pencarian Judul Makanan (*Owner*)



Gambar 4. 27 Gambar *User Interface form* Pencarian Menu (*Owner*)

1. Pencarian Menu untuk Monitoring



Gambar 4. 28 Gambar *User Interface* Laporan Keuangan

## 4.3. Fase Implementasi

### 4.3.4 Pengujian (*Black Box Testing*)

#### Pada tahap ini pengujian terhadap aplikasi web Pencarian Makanan () adalah menggunakan pengujian blackbox testing. Cara pengujian dilakukan dengan menjalankan aplikasi aplikasi web Pencarian Makanan () dengan melakukan input data dan melihat output apakah sesuai dengan yang diharapkan, berikut dibawah ini merupakan ujicoba pada sistem dan hasil yang di dapat pada tabel dibawah ini.

#### 4.3.4.1 Pengujian User (Admin)

Tabel 4. 27 Tabel Pengujian *User* (*Admin*)

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Rancangan Proses** | **Hal yang Diharapkan** | **Hasil** |
| 1 | Klik menu Masuk (*Login*) | Masuk ke *form login* | Sesuai |
| 2 | Pilih menu pesanan | Menampilkan halaman *manage* data pesanan | Sesuai |
| 3 | Klik validasi | Merubah status konfirmasi pembayaran menjadi YES | Sesuai |
| 4 | Klik *upload* *file* | Menampilkan form *upload* *file* Pencarian Makanan | Sesuai |
| 5 | Pilih menu Makanan | Menampilkan halaman *manage* data Pencarian Makanan | Sesuai |
| 6 | Klik tambah data Makanan | Menampilkan form tambah data Pencarian Makanan | Sesuai |
| 7 | Klik edit data Makanan | Menampilkan form edit data Pencarian Makanan | Sesuai |
| 8 | Klik hapus data Makanan | Hapus data Makanan | Sesuai |
| 9 | Cari data Makanan | Menampilkan data Makanan yang diinginkan | Sesuai |
| 10 | Pilih menu Makanan |  | Sesuai |
| 11 | Tambah data Makanan | Menampilkan halaman *manage* data Makanan | Sesuai |
| 12 | Edit data Makanan | Menampilkan form tambah data Makanan | Sesuai |
| 13 | Hapus data Makanan | Menampilkan form edit data Makanan dari *database* | Sesuai |
| 14 | Cari data Makanan | Hapus data Makanan | Sesuai |
| 15 | Klik menu *Logout* | Keluar dari sistem dan kembali ke halaman masuk (*login*) | Sesuai |

#### 4.4.4.2 Pengujian User (Member)

Tabel 4. 28 Tabel Pengujian User (*Member*)

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Rancangan Proses** | **Hal yang Diharapkan** | **Hasil** |
| 1 | Klik menu Masuk (*Login*) | Masuk ke *form* *login* | Sesuai |
| 2 | Klik sub menu pesanan | Masuk ke halaman *manage* data pesanan | Sesuai |
| 3 | Cari judul Makanan | Berhasil menampilkan data Makanan yang diinginkan | Sesuai |
| 4 | Konfirmasi pembayaran (Klik edit) | Berhasil menyimpan data pembayaran di *database* | Sesuai |
| 5 | Klik download | Menampilkan pop up download *file* Pencarian Makanan | Sesuai |
| 6 | Klik menu Makanan | Menampilkan halaman Pencarian Makanan | Sesuai |
| 7 | Pesan Makanan | berhasil menyimpan data pesanan di keranjang | Sesuai |
| 8 | Klik menu keranjang belanja | Menampilkan daftar pesanan Pencarian Makanan | Sesuai |
| 9 | Klik menu *logout* | Keluar dari sistem dan kembali ke halaman masuk (*login*) | Sesuai |

#### 4.3.4.3 Pengujian User (Pengunjung)

Tabel 4. 29 Pengujian *User* (Pengunjung)

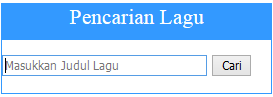
|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Rancangan Proses** | **Hal yang Diharapkan** | **Hasil** |
| 1 | Klik menu Makanan | Menampilkan halaman Pencarian Makanan | Sesuai |
| 2 | Cari judul Makanan | Menampilkan hasil pencarian judul Makanan yang diinginkan | Sesuai |

## BAB V HASIL DAN PEMBAHASAN

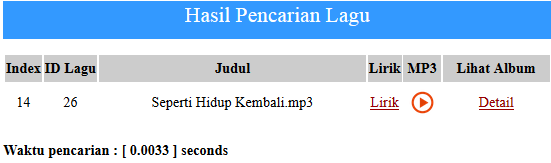
Pada bab ini menjelaskan hasil dan pembahasan mengenai pencarian judul Makanan dengan bantuan algoritma *binary search* dan proses pemesanan Pencarian Makanan.

## 5.1 Pencarian Judul Makanan Dengan Algoritma *Binary Search*

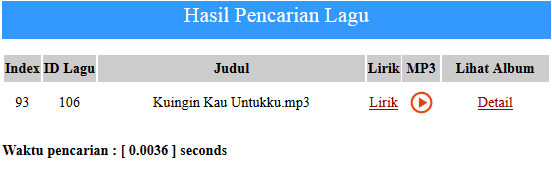
Implementasi algoritma *Binary Search* pada pencarian judul Makanan dilakukan dengan memasukkan judul Makanan pada kolom *textbox* pencaian judul Makanan. Judul Makanan yang dimasukkan disebut sebagai kata kunci (*key*), data Makanan pada *database* Makanan yaitu fileMakanan.txt sebanyak 212 Makanan. Berikut dibawah ini adalah hasil pencarian Makanan



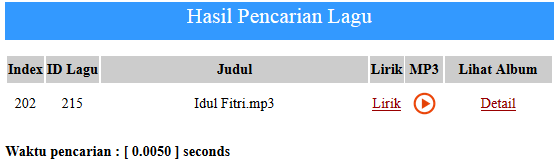
Gambar 5. 1 Gambar Pencarian Judul Makanan



Gambar 5 2 Hasil Pencarian Makanan "Seperti Hidup Kembali"

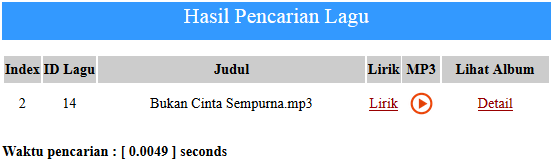


Gambar 5. 3 Hasil Pencarian “Kuingin Kau Untukku”



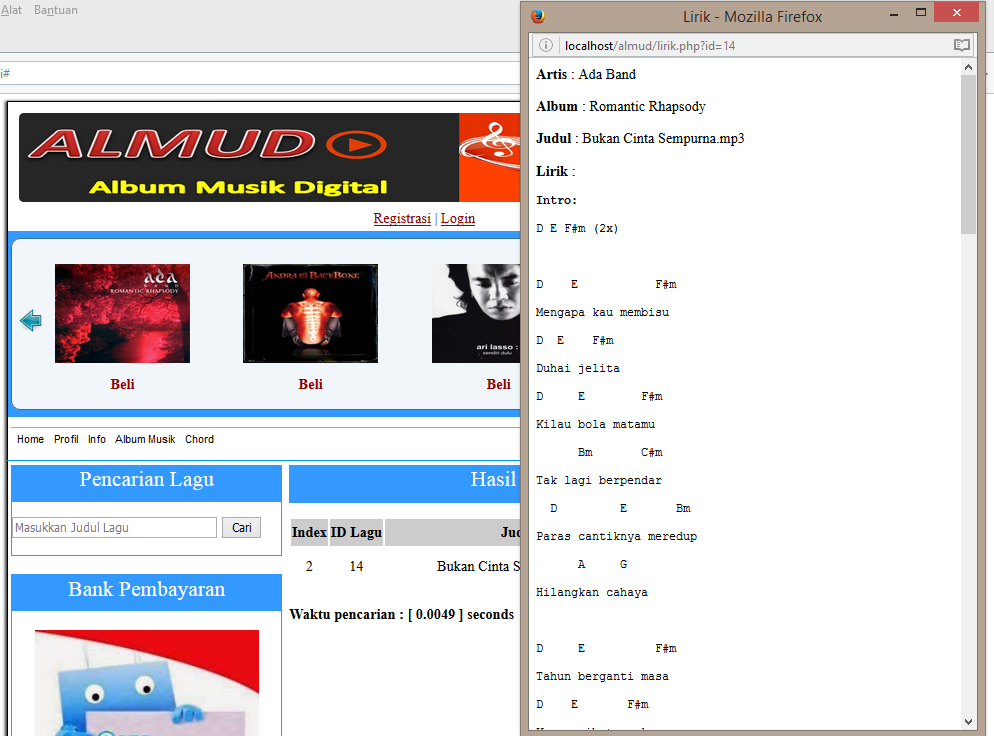
Gambar 5. 4 Hasil Pencarian Makanan "idul Fitri"

Pada gambar diatas didapat kesimpulan, bahwa algoritma *binary search* dapat berjalan sesuai rumus dan menampilkan hasil *output* sesuai dengan cara kerja algoritma *binary search* dimana algoritma *Binary search* adalah algoritma pencarian untuk data yang terurut menaik. Pencarian dilakukan dengan cara menebak apakah data yang dicari berada ditengah-tengah data (mid=low+high/2), kemudian membandingkan data yang dicari dengan data yang ada ditengah. Bila data yang ditengah sama dengan data yang dicari, berarti data ditemukan. Namun, bila data yang ditengah lebih besar dari data yang dicari, maka dapat dipastikan bahwa data yang dicari kemungkinan berada disebelah kiri dari data tengah dan data disebelah kanan data tengah dapat diabaikan (high=mid-1). Sebaliknya bila data yang ditengah lebih kecil dari data yang dicari maka kemungkinan data berada disebelah kanan dari data tengah dan data kiri dapat diabaikan (low=mid+1). **(Duha, 2013)**.

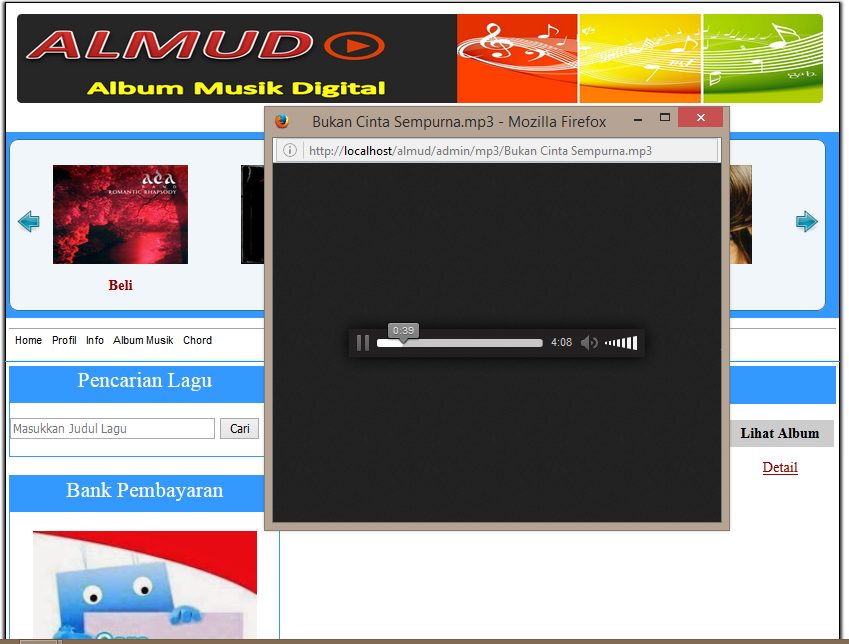


Gambar 5 5 Hasil Pencarian Judul Makanan "Bukan Cinta Sempurna"

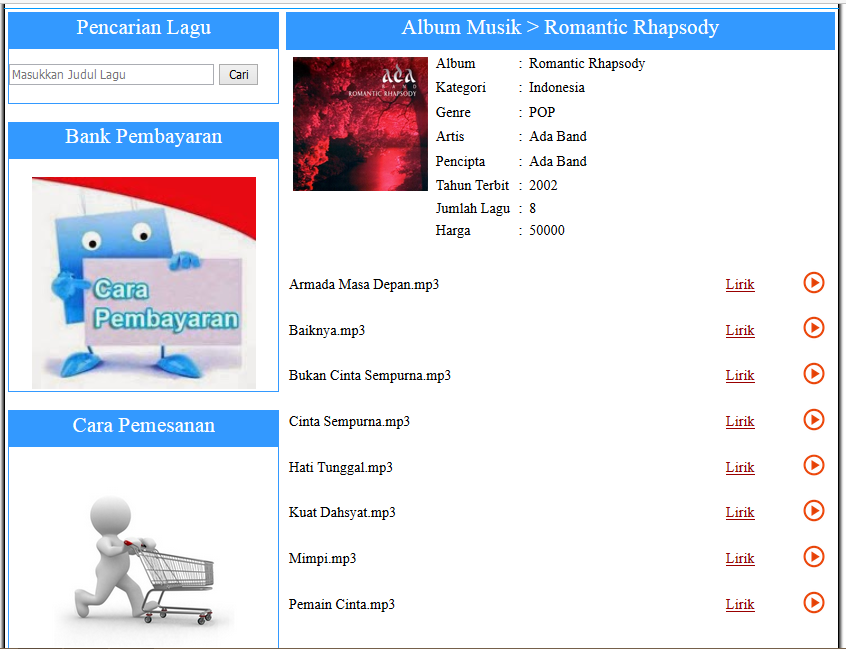
Gambar diatas menunjukan hasil pencarian judul Makanan “Bukan Cinta Sempurna” dan menjadi solusi dari identifikasi permasalahan pada bab 3 mengenai sulitnya mendapatkan Makanan / Pencarian Makanan disertai mp3, + chord . Hasil yang didapat melalui pencarian Makanan menampilkan isi Makanan + chord , dan MP3 yang bisa dimainkan serta rincian detail Makanan dari Makanan yang dicari.



Gambar 5 6 Makanan Bukan Cinta Sempurna



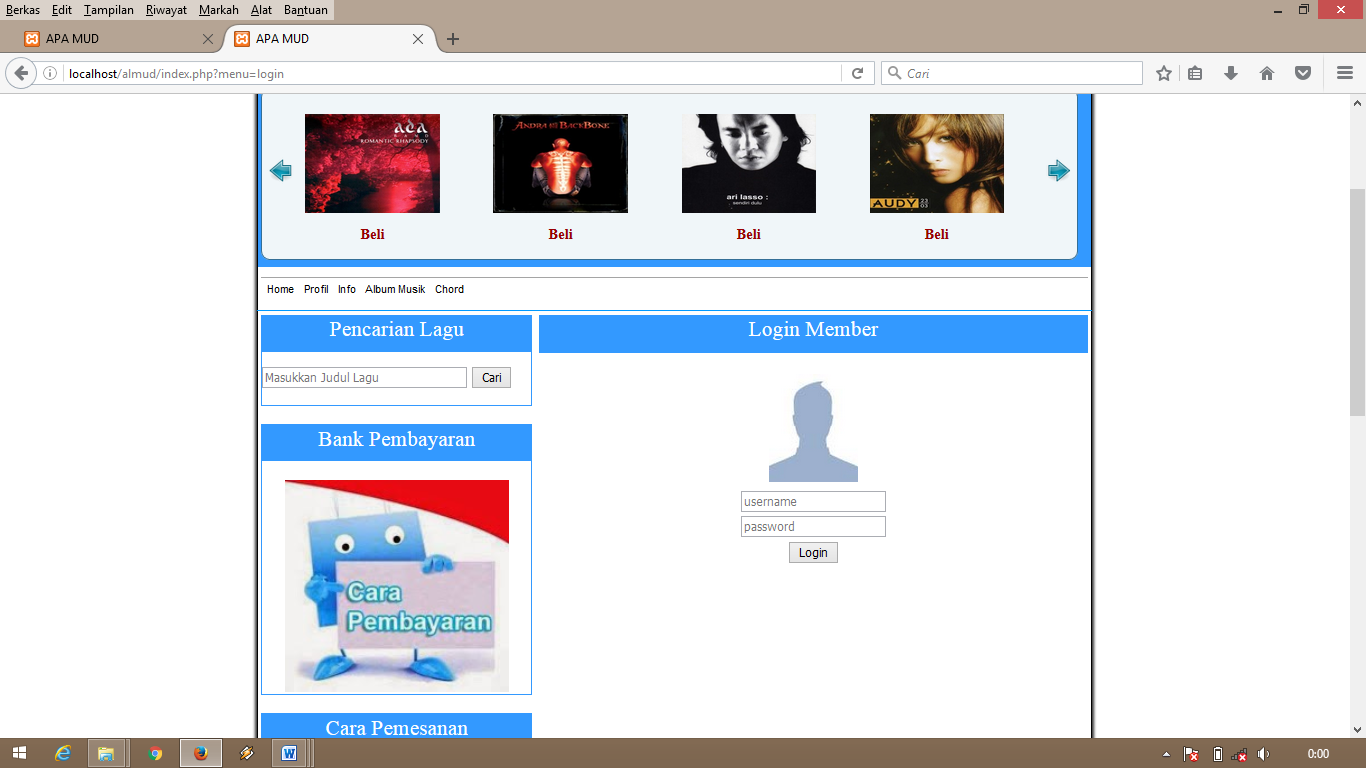
Gambar 5 7 MP3 Bukan Cinta Sempurna



Gambar 5. 8 Gambar Halaman Detail Makanan Dari Makanan

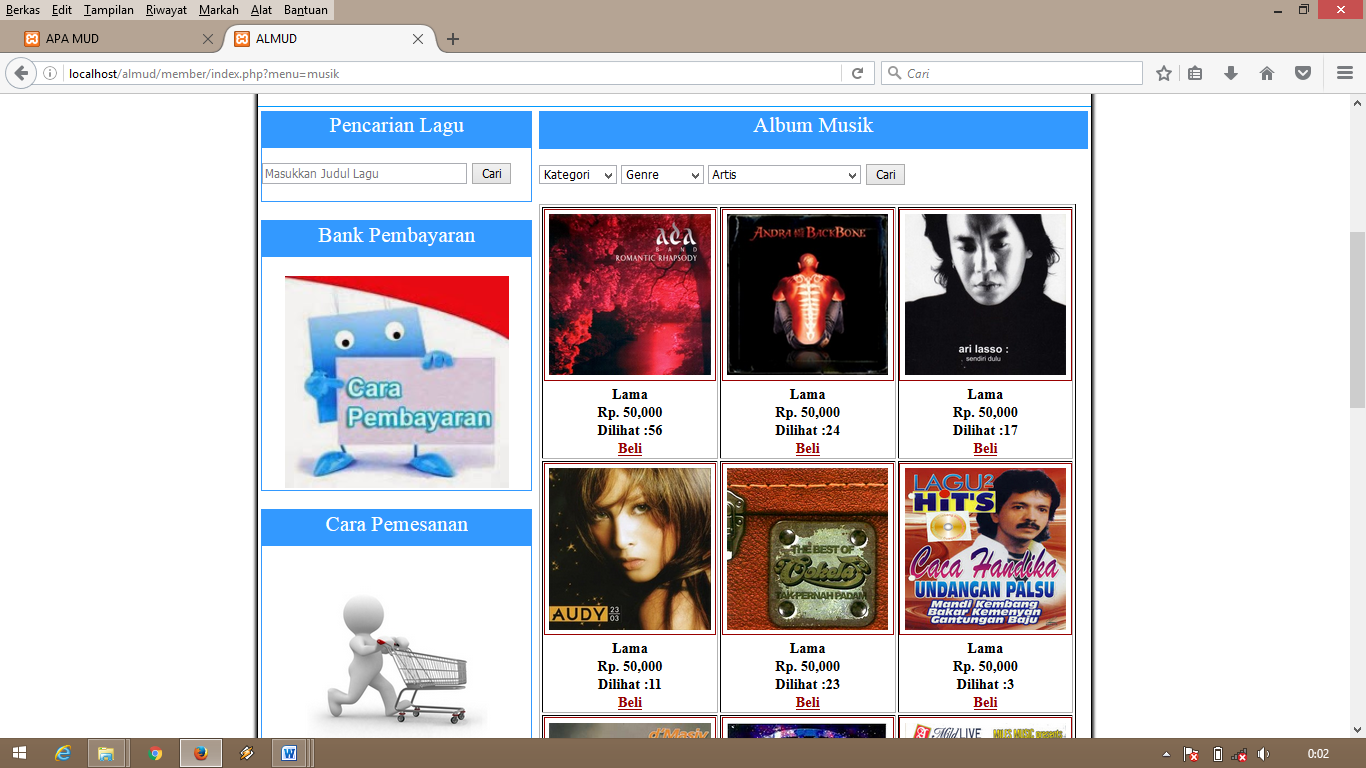
Bukan Cinta Sempurna

## Proses Pemesanan Pencarian Makanan



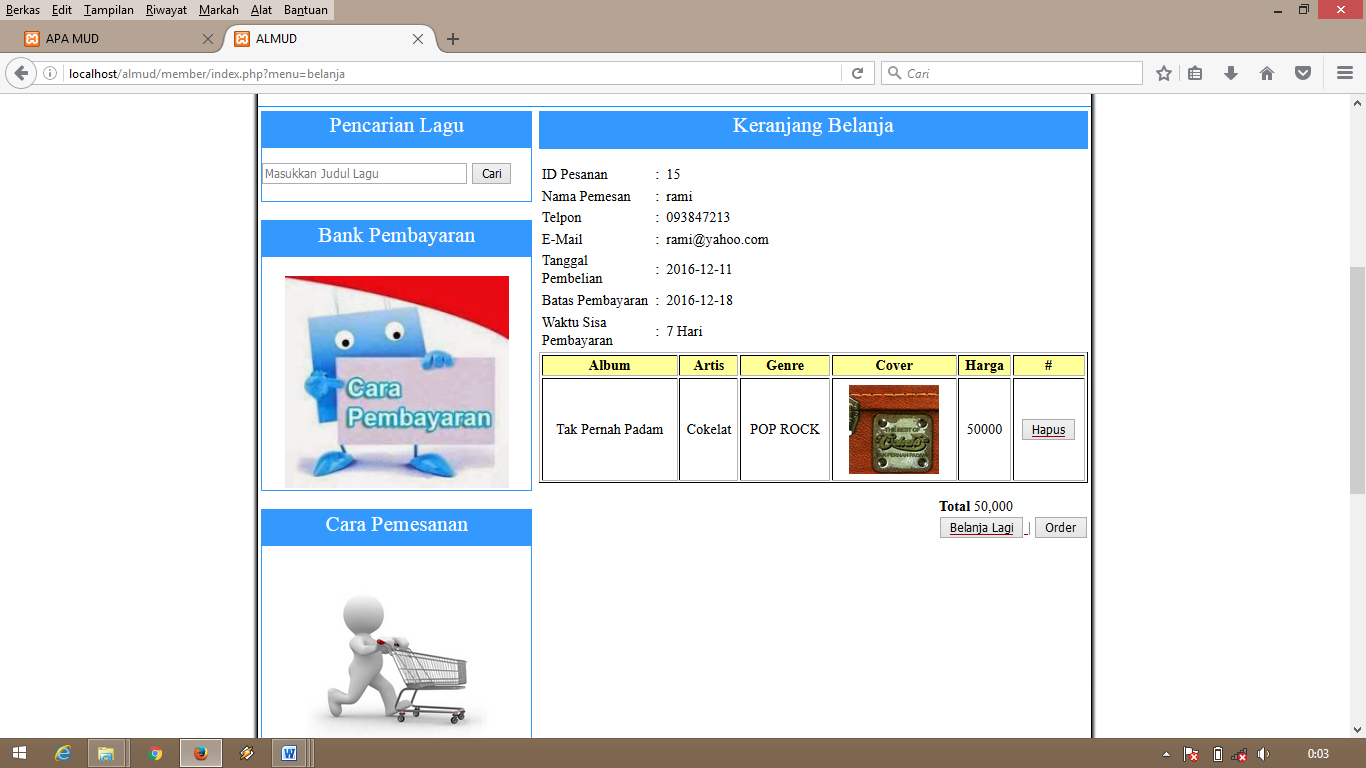
Gambar 5 9 *Form Login Member*

Pada halaman ini *member* memasukkan *username* dan *password*, lalu klik *button* “*Login*” untuk masuk ke sistem.



Gambar 5 10 Halaman Menu Pencarian Makanan (*Member*, Pengunjung)

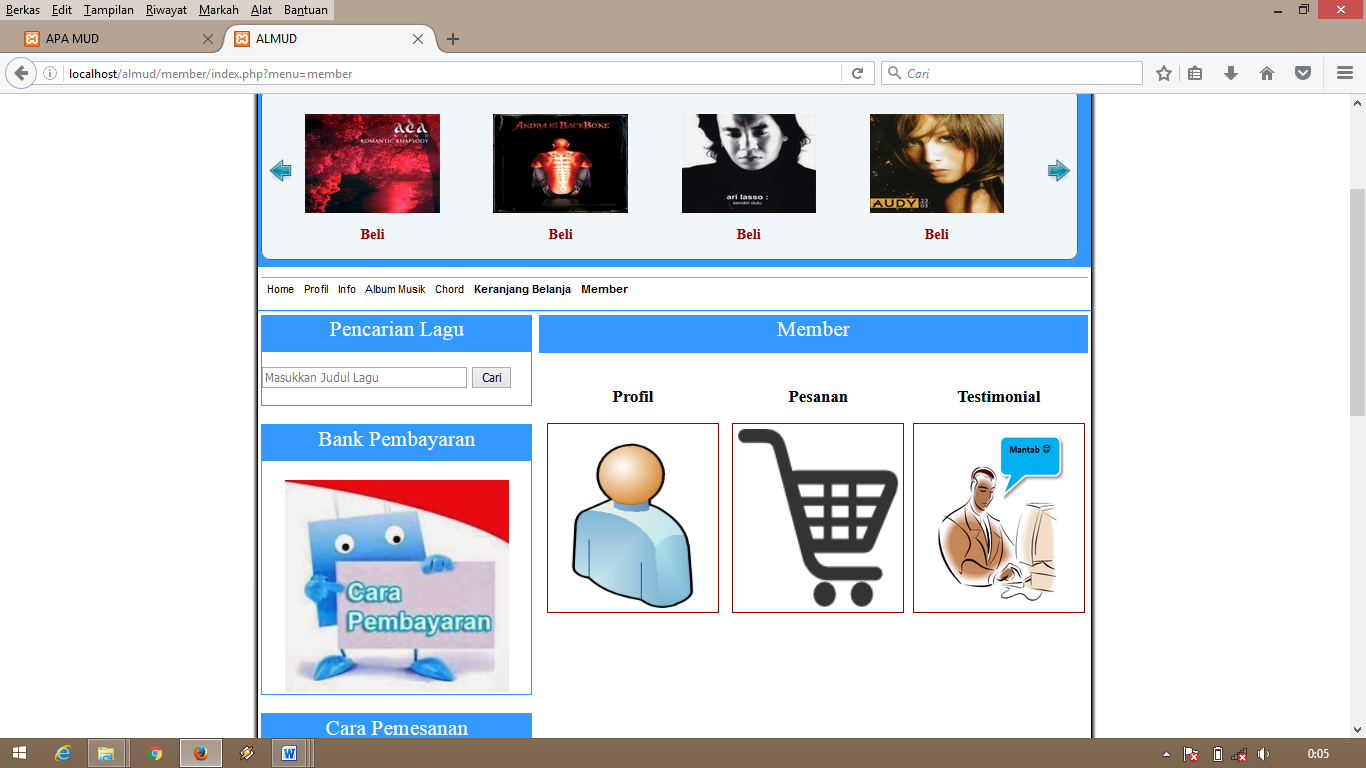
Gambar diatas merupakan halaman Makanan dengan hak akses *member* atau pengunjung, Makanan dapat dibeli oleh *member*, dengan cara memilih Makanan lalu klik “Beli”.



Gambar 5 11 Halaman Menu Keranjang Belanja (*Member*)

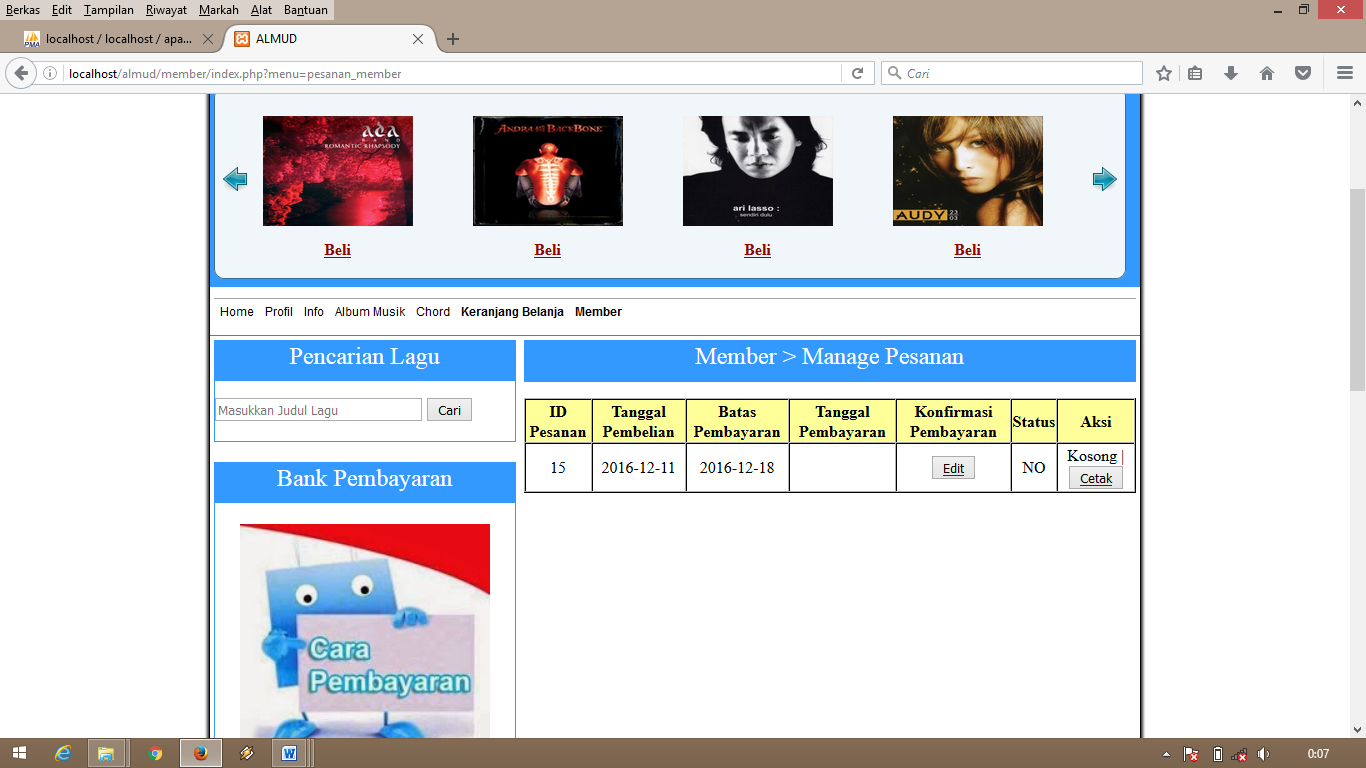
Gambar diatas merupakan halaman keranjang belanja setelah *member* melakukan pembelian, pada halaman ini berisikan informasi id pesanan, nama pemesan, telpon, email, tanggal pembelian, batas pembayaran, waktu sisa pembayaran, dan daftar pesanan Makanan yang dipesan.

Makanan yang dipesan sejumlah satu. Klik *button* “Belanja Lagi” jika ingin menambah Makanan , klik *button* “Hapus” jika ingin menghapus pesanan Makanan , klik *button* “*Order*” jika ingin lanjut memesan.



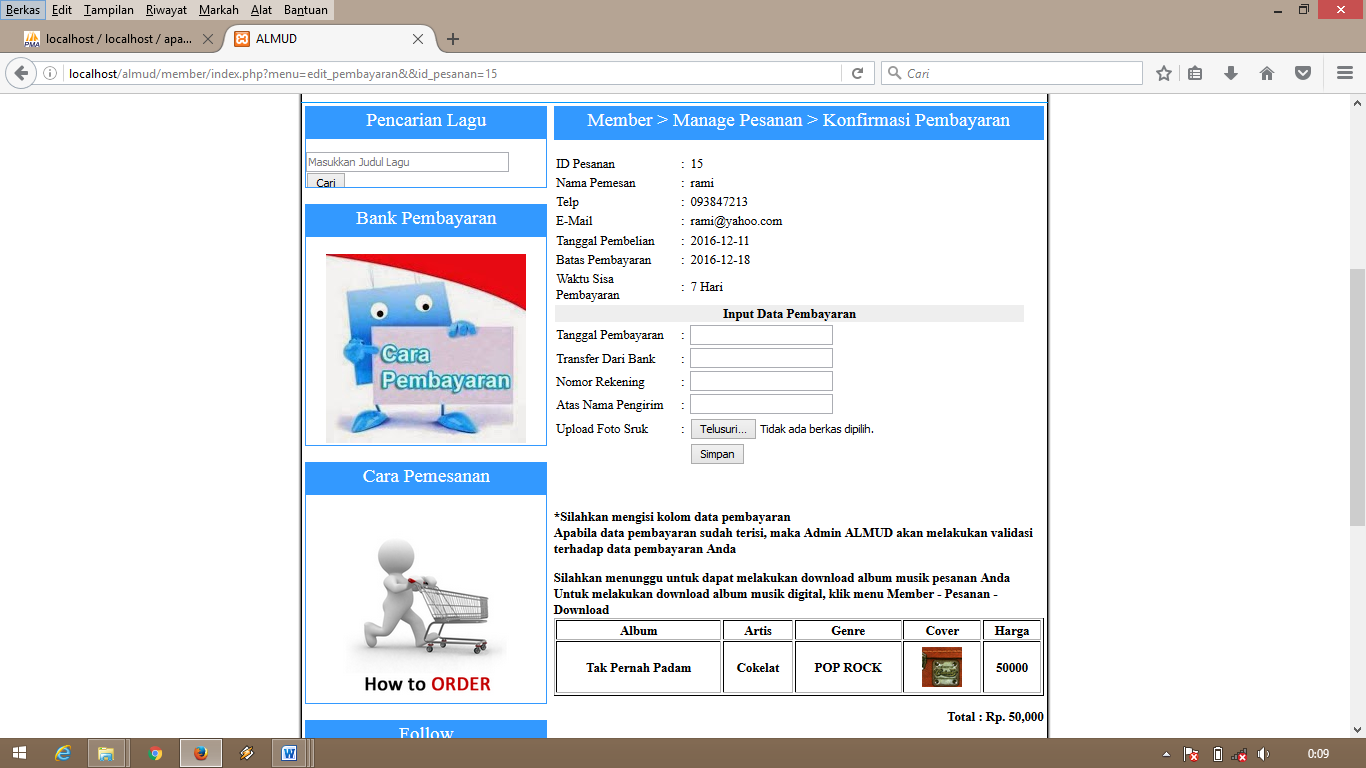
Gambar 5 12. Halaman Menu *Member*

Gambar diatas merupakan halaman *member* yang berisi profil untuk melihat profil atau mengubah data diri, pesanan untuk *manage* data pesanan, dan testimonial untuk mengisi testimoni.



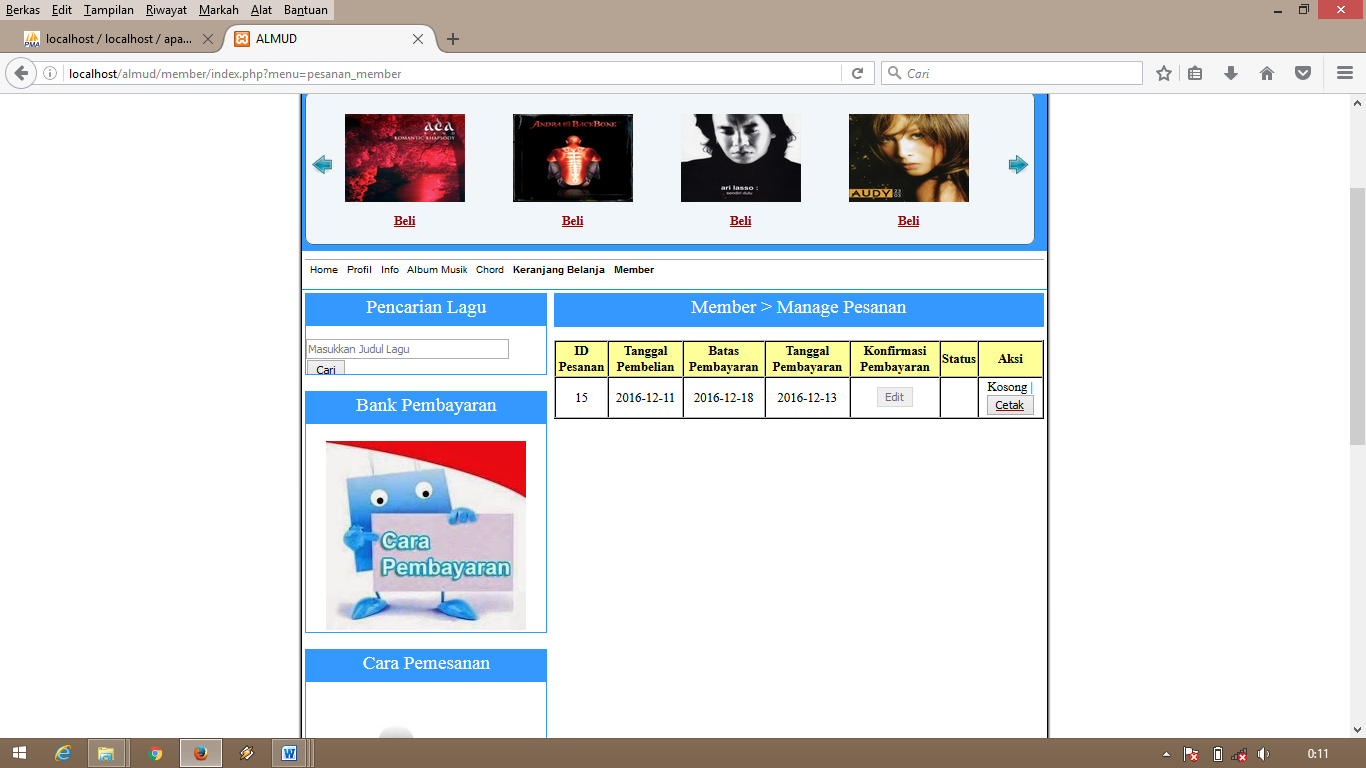
Gambar 5 13. Halaman *Manage* Data Pesanan (*Member*)

Gambar diatas merupakan halaman *manage* data pesanan (*Member*), pada halaman ini berisi *record* data pesanan pemesan (*member*). Pada halaman ini pemesan (*member*) harus melakukan konfirmasi pembayaran jika ingin mendapatkan *file* Pencarian Makanan yang dipesan ditandai dengan kolom tanggal pembayaran masih kosong.



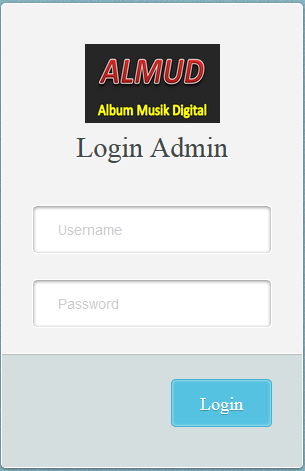
Gambar 5 14. Halaman Konfirmasi Pembayaran (*Member*)

Gambar diatas merupakan halaman *form* konfirmasi pembayaran, pada halaman ini pemesan (*member*) mengisikan data pembayaran dan *upload* foto struk pembayaran.



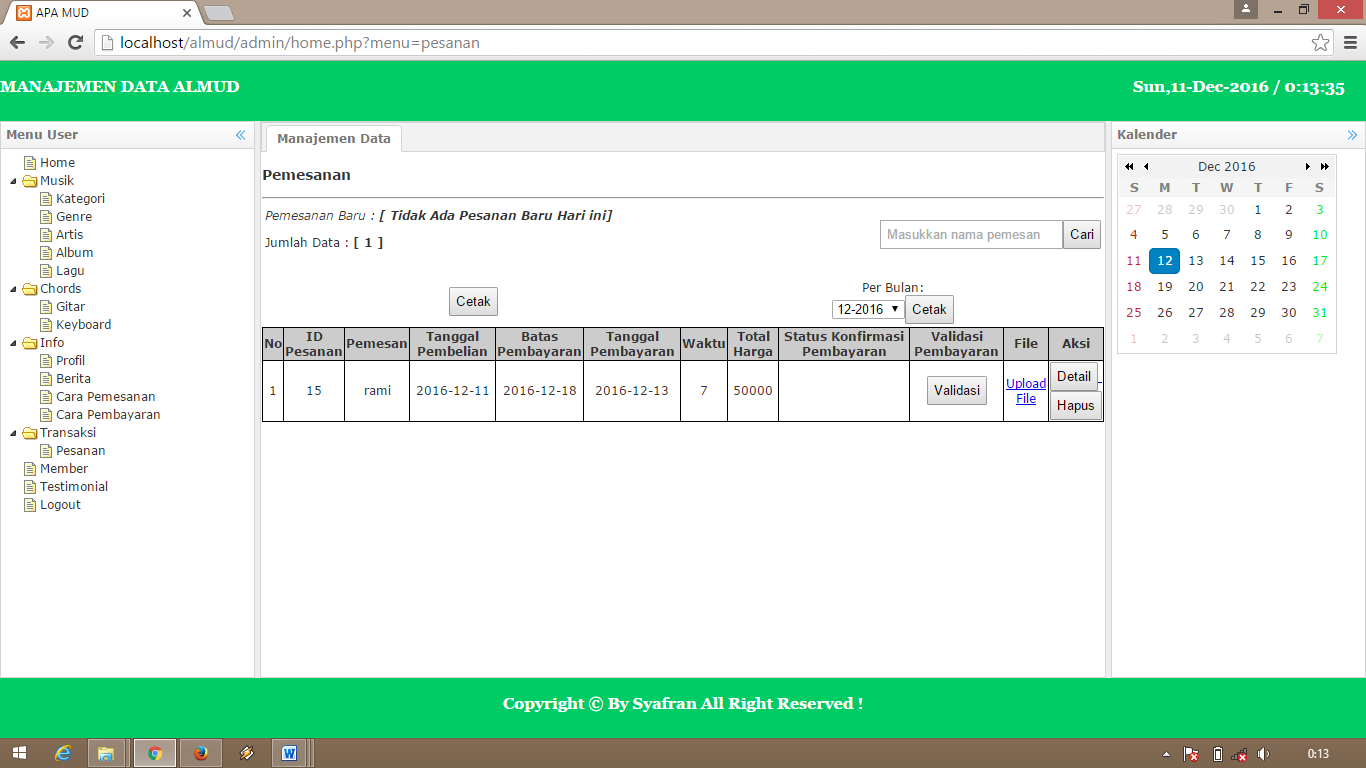
Gambar 5 15. Halaman *Manage* Data Pesanan (*Member*) Sudah Konfirmasi Pembayaran

Gambar diatas merupakan halaman *manage* data pesanan, pada halaman ini pemesan (*member*) telah mengisi konfirmasi pembayaran ditandai dengan terisinya data tanggal pembayaran pada kolom tanggal pembayaran.



Gambar 5 16. Halaman *Login Admin*

Gambar diatas merupakan halaman *login admin*, pada halaman ini *admin* memasukkan *username* dan *password* lalu klik *button* “*Login*” untuk masuk ke sistem.

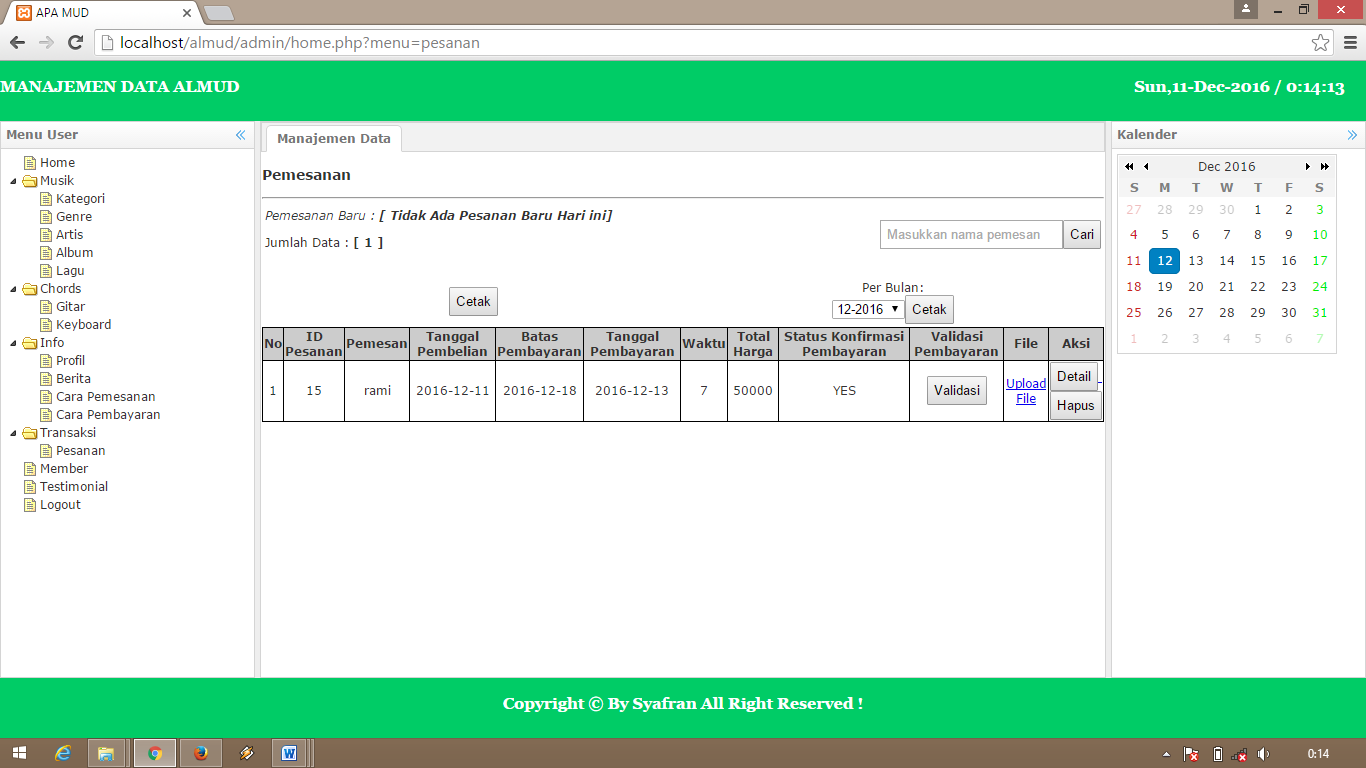


Gambar 5 17. Halaman Menu *Manage* Data Pemesanan (*Admin*)

Gambar diatas merupakan halaman *manage* data pesanan (*admin*), pada halaman ini berisikan informasi data pesanan pemesan (*member*), dan pada halaman ini dilakukannya validasi dan *upload* *file* Makanan yang telah dipesan.

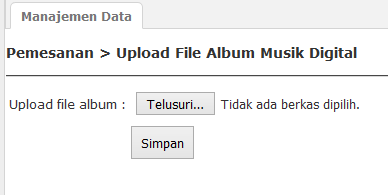
Gambar diatas menunjukkan status validasi pembayaran masih kosong, karena *admin* belum melakukan validasi, sebelum melakukan validasi *admin* harus melihat apakan tanggal pembayaran sudah terisi, jika sudah maka dapat dilakukan validasi dan jika belum tidak dapat dilakukan validasi.

Caranya klik *button* “Validasi” maka status konfimasi pembayaran akan otomatis berubah menjadi YES



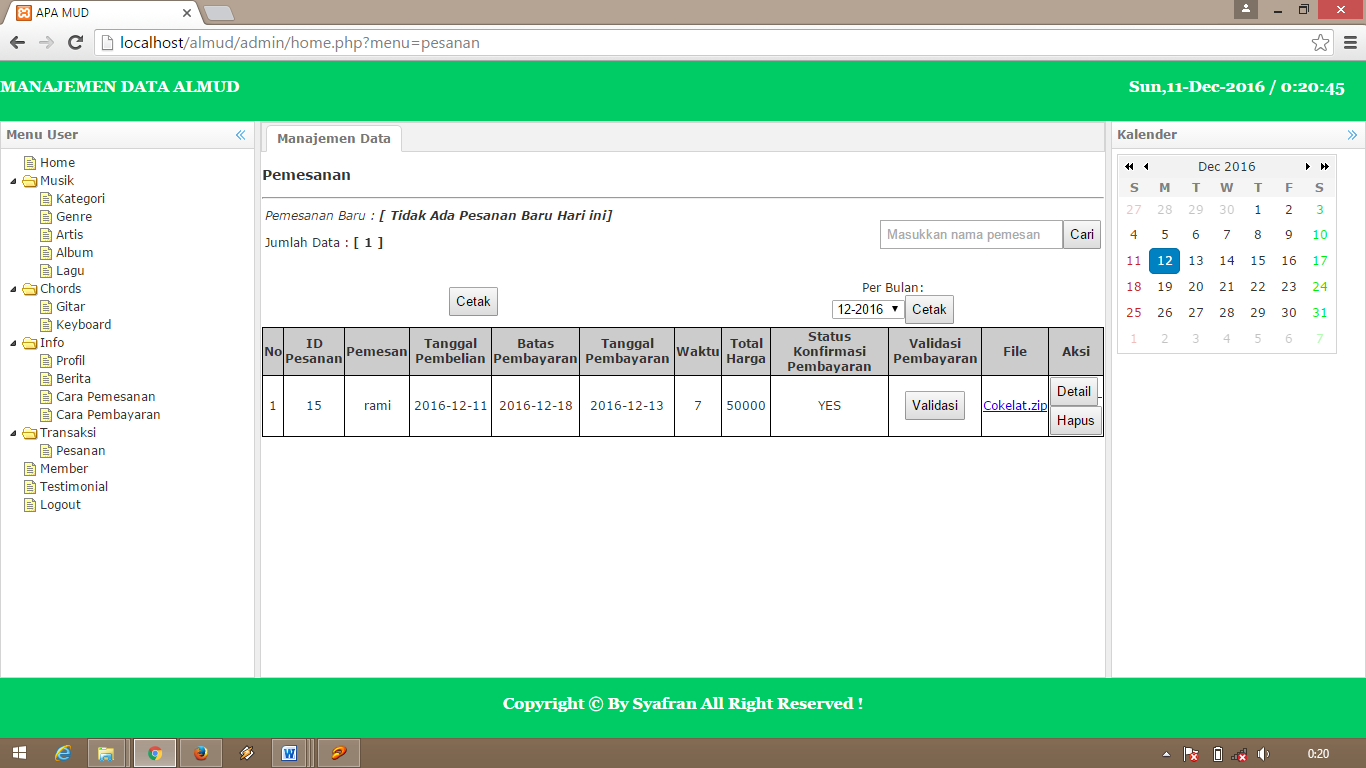
Gambar 5 18. Halaman Menu *Manage* Data Pemesanan (*Admin*) – Validasi

Gambar diatas merupakan halaman *manage* data pesanan (*admin*), setelah melakukan validasi selanjutnya *admin upload file* Makanan dengan cara klik “*Upload* *File*” pada kolom *File*.



Gambar 5 19. Halaman *Upload File* Makanan (*Admin*)

Gambar diatas merupakan halaman *upload* *file* Pencarian Makanan, pada halaman ini *admin* melakukan *upload* *file* Pencarian Makanan. Setelah *file* Pencarian Makanan berhasil di *upload* maka akan ditandai dengan adanya nama *file* Pencarian Makanan di kolom *File* seperti gambar dibawah ini.

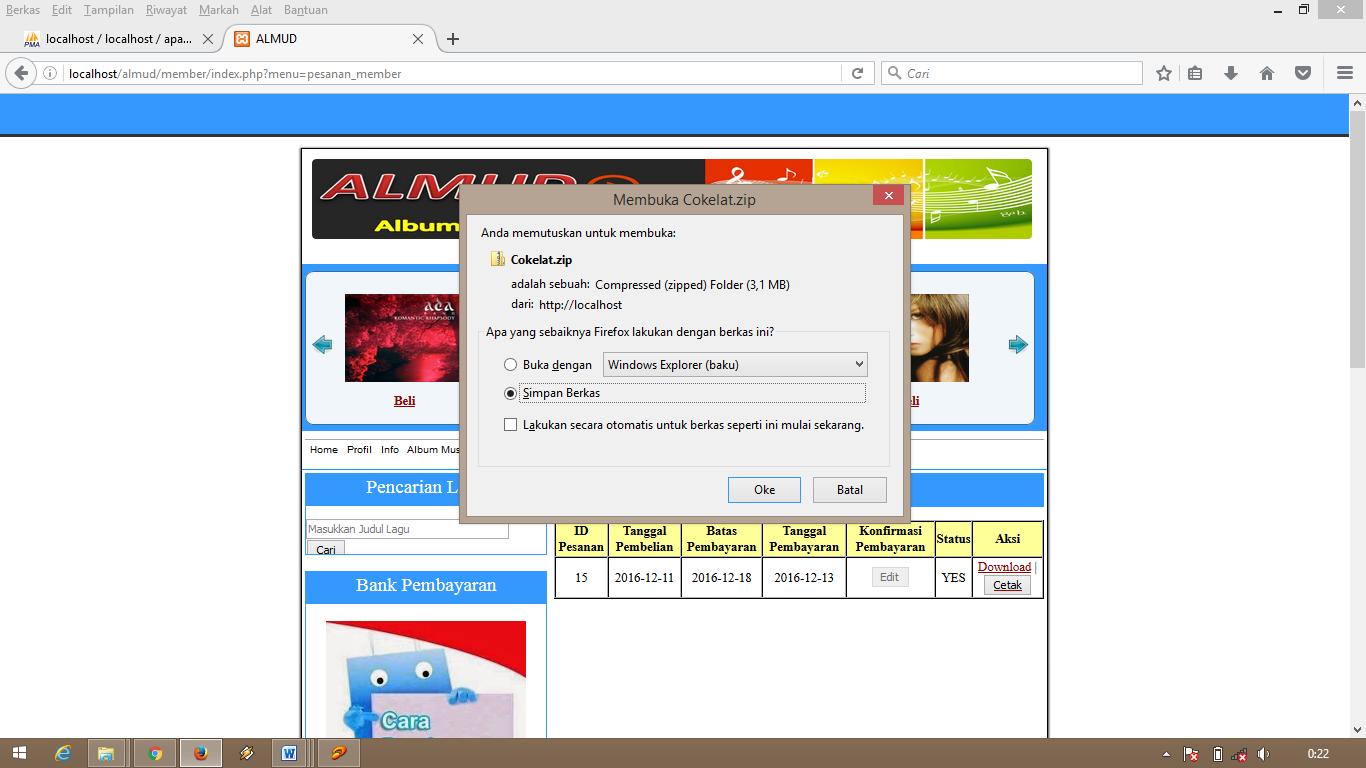


Gambar 5 20. Halaman *Manage* Data Pemesanan (*Admin*) - *File* Makanan Sudah Terisi

Setelah *admin* sudah melakukan validasi atas konfirmasi pemabyaran dan *upload* *file* Pencarian Makanan, maka pemesan (*member*) dapat *login* kembali dan langsung *download* *file* Pencarian Makanan dengan cara klik “*Download*” pada kolom aksi di gambar bawah ini.



Gambar 5 21. Halaman *Manage* Data Pesanan (*Member*) - *Download*



Gambar 5 22. Halaman *Pop Up* Hasil *Download* *File* Pencarian Makanan (*Member*)

# BAB VI PENUTUP

## Kesimpulan

Penggunaan algoritma *binary search* dalam penelitian ini adalah mencari *record* judul Makanan pada *database* Makanan Pencarian Makanan(). Algoritma ini bekerja dengan cara memilih elemen judul Makanan dengan *index* tengah dari tabel Makanan dan membandingkannya dengan data judul Makanan yang dicari, jika elemen dengan *index* tengah yang dicari sesuai dengan data elemen pada *index* tengah maka data ditemukan namun apabila elemen dengan *index* tengah pada *array* lebih besar dari *index* dengan elemen yang dicari maka nilai *index* batas atas sama dengan nilai *index* tengah – 1, apabila nilai elemen dengan *index* tengah pada *array* lebih kecil dari *index* dengan elemen yang dicari maka nilai *index* batas bawah sama dengan nilai *index* tengah + 1.

Aplikasi web penjualan Pencarian Makanan () mampu memberikan hasil pencarian Makanan dengan informasi lengkap diantaranya , mp3 dan chord dan mampu memberikan simulasi pemesanan Pencarian Makanan yang mudah.

## Saran

Aplikasi ini masih belum sempurna. Masih banyak hal yang dapat dikembangkan pada aplikasi ini agar menjadi lebih baik. Oleh karena itu penulis juga menyampaikan beberapa saran untuk pengembangan berikutnya, diantaranya :

1. Ditambahkan fitur pengamanan sistem dan proses bisnis transaksi pembayaran menggunakan *PayPal* atau *token pulsa*
2. Perlu dilakukannya kerjasama dengan para artis musisi untuk mendiskusikan hasil pembagian keuntungan terhadap penjualan Pencarian Makanan.

# DAFTAR PUSTAKA

A, S Rosa dan M. Shalahuddin. 2013. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung : Informatika.

Antaranews. 2016. Penjualan Digital Salip Fisik. [http://www.antaranews.com/berita/555046/penjualan--digital-salip--fisik](http://www.antaranews.com/berita/555046/penjualan-musik-digital-salip-musik-fisik). Diakses 12 Februari 2016.

Astarina, Ida., dan Riasti, Berliana Kusuma. 2012. Pembuatan Sistem Penjualan *Online* Pada Toko PN Sukoharjo. Surakarta : Universitas Surakarta.

Duha, Toto Nafo. 2013. Perancangan Aplikasi Kidung Jemaat Berbasis Java Dengan Menggunakan Algoritma *Binary Search Ascending* Pada Telepon Seluler. Medan : STMIK Budidarma Medan.

Dwipayana, Krisna. 2016. Review Tentang *Digital*. [http://dolanan.blogspot.com/2016/02/review-tentang--*digital*.html](http://dolananmusik.blogspot.com/2016/02/review-tentang-musik-digital.html). Diakses 12 Februari 2016.

Indriyani, Evi dan Humdiana. 2009. Sistem Informasi Manajemen : Mempersiapkan Pekerja Berbasasis Pengetahuan dalam Mengelola Sistem Informasi. Jakarta : Mitra Wacana Media.

Kadir A. 2014. Pengenalan Sistem Informasi (Edisi Revisi). Penerbit: Andi, Yogyakarta.

Kamil. 2012. Digital. <https://suhailykamil.wordpress.com/2012/01/05/music-digital/>. Diakses 12 Februari 2016.

Kartono, Arlo. 2007. Kreasi Seni Budaya. Jakarta : Ganeca Exact.

Kenneth, E. K., & Julie, E. K. 2013. *System Analysis and Design Ninth Edition.* New York: Prentice Hall.

Levitin, Anany. 2011. *Introduction to the Design and Analysis of Algorithms (3rd Edition).* USA : Addison Wesley.

McMilan, Michael. 2014. *Data Structures & Algorithms with JavaScript*. USA : O’Reilly Media.

Munir, Renaldi dan Leony Lidya. (2011). Algoritma dan Pemograman. Bandung : Informatika.

Netrilis. 2016. *Music Streaming* Mendominasi Penjualan Industri (Report Tahun 2015). <http://www.netrilis.com/2016/03/music-streaming-mendominasi-penjualan.html>. Diakses 12 Februari 2016.

Nugroho, Adi. 2009. (*E-Commerce*) Memahami Perdagangan Modern Di Dunia Maya. Bandung: Informatika.

Prakasa, Ahmad Reza Widia. 2014. Rancang Bangun Sistem Informasi *E-Commerce* Menggunakan *Payment Gateway* PayPal (Studi Kasus : Omekimai Gadget Store). Jakarta : Universitas Islam Negeri Syarif Hidayatullah Jakarta.

RC, Qory Kurnia. 2015. Aplikasi Kamus Indonesia-Korea Menggunakan Algoritma *Binary Search* Berbasis Android. Jakarta : Universitas Islam Negeri Syarif Hidayatullah Jakarta.

Rizqi, Dian Yutika. 2014. Implementasi Algoritma *Porter Stemmer* Dan Algoritma *Boyer Moore* Pada Pencarian Ensiklopedia Etika Islam Berdasarkan Al-Qur’an Dan Hadits. Jakarta : Universitas Islam Negeri Syarif Hidayatullah Jakarta.

Simarmata, Janner. 2010. Rekayasa Web. Yogyakarta, Andi Yogyakarta.

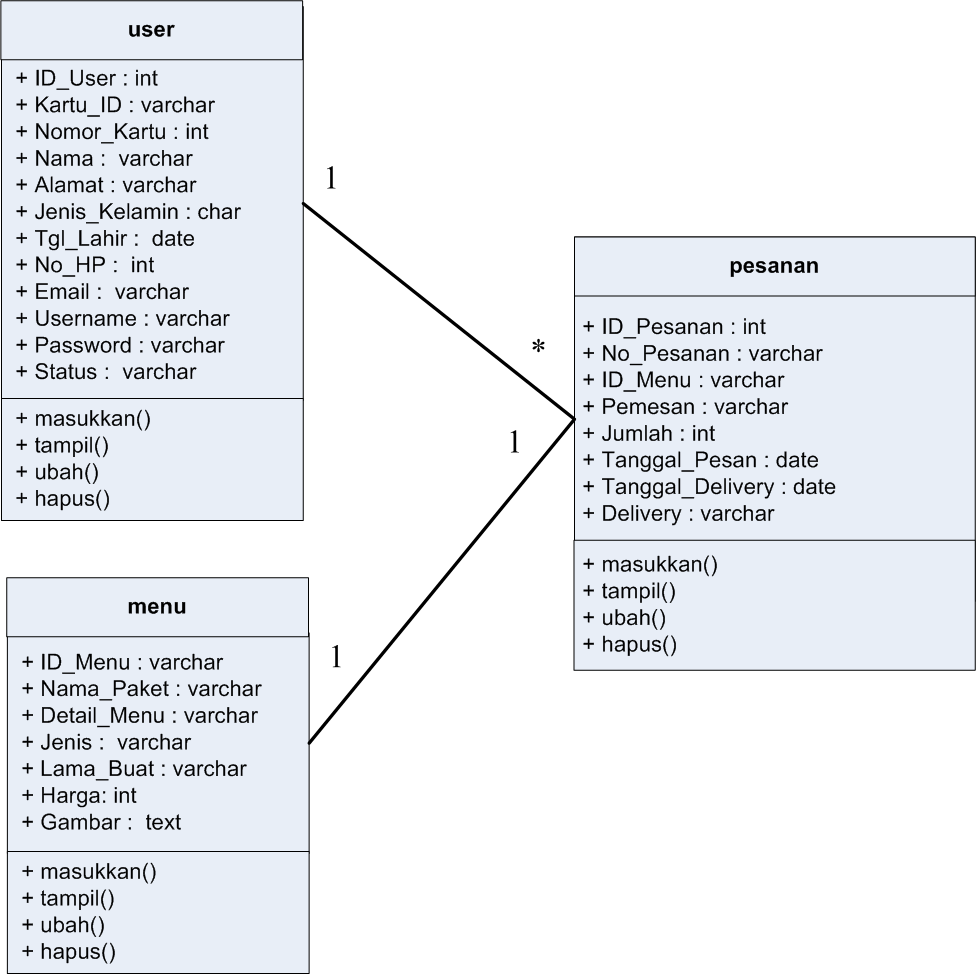
Swatha, Basu. 2010. Manajemen Penjualan Edisi 3. Yogyakarta : Liberty Yogyakarta.

Tyas, Hataris Andijaning. 2007. Seni . Jakarta : Erlangga.

Watson, Richard T, Pierre Berthon, Leyland F. Pitt, and George M. Zinkhan. 2008. *Electronic Commerce: The Strategic Perspective*. Switzerland : Jacobs Fondation.

# LAMPIRAN

**Lampiran 1 : *Class Diagram***



Lampiran 1 (Gambar) 1. Desain *Class Diagram* Aplikasi Monitoring Keuangan

Dibawah ini adalah deskripsi singkat mengenai desain class diagram aplikasi monitoring keuangan berbasis web (*e-commerce*).

1. *Association* (Asosiasi)
2. *Class user* memiliki hubungan asosiasi dengan beberapa *class,* yaituchord\_gitar, chord\_keyboard, kategori, genre, Makanan, pemesanan, Makanan, artis, berita, dan testimonial dan hubungannya 1 to 1..\*, karena 1 user dapat mengelola banyak data chord gitar, chord\_keyboard, kategori, genre, Makanan, pemesanan, Makanan, artis, berita, dan testimonial pada sistem.
3. *Class* *user* memiliki hubungan asosiasi dengan beberapa *clas,* yaituprofil\_web, cara\_pembayaran, cara\_pemesanandan hubungannya 1 to 1, karena 1 *user* hanyamelakukan pembaruan data profil web, cara\_pembayaran, cara\_pemesanan pada sistem.
4. *Agregation* (Agregasi)
5. *Class* Makananmemiliki hubungan agregasi dengan *class* keranjang karena *class* Makanan merupakan bagian dari *class* keranjang, namun *class* Makanan dapat berdiri sendiri dan hubungannya 1..\* to 1, karena 1 atau banyak Makanan dapat memiliki 1 keranjang.
6. *Class* pemesananmemiliki hubungan agregasi dengan *class* keranjang karena *class* pemesanan merupakan bagian dari *class* keranjang, namun *class* pemesanan dapat berdiri sendiri dan hubungannya 1 to 1, karena 1 pemesanan hanya dapat memiliki 1 keranjang.
7. *Composition* (Komposisi)
8. *Class* artis, genre, kategori memiliki hubungan komposisi dengan *class* Makanan karena jika *class* artis, genre, kategori tidak ada maka *class* Makanan juga tidak ada. Hubungannya 1 to 1..\*, karena 1 artis, genre, kategori memiliki 1 Makanan pada sistem.
9. *Class* artis memiliki hubungan komposisi dengan *class* Makanan karena jika *class* artis tidak ada maka *class* Makanan juga tidak ada. Hubungannya 1 to 1..\*, karena 1 artis memiliki banyak Makanan pada sistem.

**Lampiran 2 : *Mapping Database***



Lampiran 2 (Gambar) 1. Desain *Mapping Database* (Pencarian Makanan)

1. Spesifikasi *Database*
2. User

|  |  |  |
| --- | --- | --- |
| * *Primary Key* | : | ID\_User |
| * *Foreign Key* | : | - |

Lampiran 2 (Tabel) 1. Tabel *User*

|  |  |  |
| --- | --- | --- |
| ***Field*** | ***Type*** | ***Keterangan*** |
| ID\_User | int (11) | ***Unique id*** tabel user |
| Kartu\_ID | varchar (15) | Kartu ID di sistem |
| Nomor\_Kartu | int (20) | Nomor kartu di sistem |
| Nama | varchar (20) | Nama user di sistem |
| Alamat | varchar (100) | alamat user di sistem |
| Jenis\_Kelamin | varchar (15) | Jenis kelamin user di sistem |
| Tgl\_Lahir | date | Tanggal lahir user di sistem |
| No\_HP | int (20) | No hp user di sistem |
| Email | varchar (30) | Email user di sistem |
| Username | varchar (50) | Username di sistem |
| Password | varchar (50) | Password di sistem |
| Status | varchar (50) | Status di sistem |

1. Menu

|  |  |  |
| --- | --- | --- |
| * *Primary Key* | : | ID\_Menu |
| * *Foreign Key* | : | - |

Lampiran 2 (Tabel) 2. Tabel Menu

|  |  |  |
| --- | --- | --- |
| ***Field*** | ***Type*** | ***Keterangan*** |
| ID\_Menu | varchar (10) | ***Unique id*** tabel menu |
| Nama\_Paket | varchar (200) | Nama paket di sistem |
| Detail\_Menu | varchar (100) | Detail menu di sistem |
| Jenis | varchar (11) | Jenis menu di sistem |
| Lama\_Buat | varchar (15) | Lama buat menu di sistem |
| Harga | int (11) | Harga menu di sistem |
| Gambar | text | Gambar menu di sistem |

1. Pesanan

|  |  |  |
| --- | --- | --- |
| * Primary Key | : | ID\_Pesanan |
| * Foreign Key | : | ID\_User |
| * Foreign Key | : | ID\_Menu |

Lampiran 2 (Tabel) 3. Tabel pesanan

|  |  |  |
| --- | --- | --- |
| ***Field*** | ***Type*** | ***Keterangan*** |
| ID\_Pesanan | Int (11) | ***Unique id*** tabel pesanan |
| ID\_User | varchar (11) | **Foreign key** tabel pesanan |
| No\_Pesanan | varchar (50) | No pesanan di sistem |
| ID\_Menu | varchar (50) | **Foreign key** tabel menu |
| Pemesan | varchar (100) | Pemesan di sistem |
| Jumlah | int (11) | Jumlah pesanan di sistem |
| Tanggal\_Pesan | date | Tanggal pesanan di sistem |
| Tanggal\_Delivery | date | Tanggal delivery pesanan di sistem |
| Delivery | varchar (20) | Delivery pesanan di sistem |

**Lampiran 3: Surat Permohonan Bimbingan Skripsi**



**Lampiran 4: *Source Code***

1. Pencarian Judul Makanan
2. *Function Binary Search*

|  |
| --- |
| <?php  // BINARY SEARCH  function BinarySearch( $array, $key, $low, $high )  {  if( $low > $high ) // termination case  {  return -1;  }  $middle = intval( ( $low+$high )/2 ); // gets the middle of the array  if ( $array[$middle] == $key ) // if the middle is our key  {  return $middle;  }  elseif ( $key < $array[$middle] ) // our key might be in the left sub-array  {  return BinarySearch( $array, $key, $low, $middle-1 );  }  elseif ( $key > $array[$middle] )  {  return BinarySearch( $array, $key, $high, $middle+1 ); // our key might be in the right sub-array  }  }  ?> |

1. Form Pencarian Judul Makanan

|  |
| --- |
| <div class="panel\_cari">  <div class="head\_cari"> Pencarian Makanan </div>  <form method=POST action='?menu=hasil\_cari'>    <p><input name="cari" type="text" size="30" id="pencarian1" placeholder="Masukkan Judul Makanan">  <input type=submit value='Cari'></p>  </form>  </div> |

1. Proses Hasil Pencarian Judul Makanan

|  |
| --- |
| // Pencarian Makanan  if($\_GET['menu']=='hasil\_cari'){    // Untuk hasil waktu pencarian  $time\_pre = microtime(true);    // Mencari data Makanan dengan mengambil data melalui fileMakanan.txt  $myfile = fopen("class.html2text/fileMakanan.txt", "r") or die("Unable to open file!"); // Buka file fileMakanan.txt  $mf = fread($myfile, filesize("class.html2text/fileMakanan.txt")); // Baca file fileMakanan.txt    $explode = explode('.mp3', $mf);    $database = array();  foreach($explode as $ex) {  $explode2 = explode('=>', $ex);    // atribut id\_Makanan dari database fileMakanan.txt  // atribut id\_Makanan dari database fileMakanan.txt  // atribut judul\_Makanan dari database fileMakanan.txt  $database[] = array(  'id\_Makanan'=>$explode2[0],  'id\_Makanan'=>$explode2[1],  'judul\_Makanan'=>$explode2[2]  );  }    //////////////////////////////////////////////////////////////////////////////////////////////////////////  $cari = $\_POST['cari']; // Proses awal pencarian  foreach($database as $ex2) {  $data2[] = trim($ex2['judul\_Makanan']).".mp3"; // Hasil inputan keyword = judul Makanan pada form pencarian, disimpan  // di $data2[] = trim($ex2['judul\_Makanan']).".mp3";  }    foreach($data2 as $ex2) {  $strpos = strpos($ex2, $cari);    if($strpos !== false) {  $key = $ex2; // Keyword = Judul Makanan  break;  }  }    include "binary.php";  $array = $data2;  $key = $ex2; // Keyword judul Makanan  $low = '0'; // Batas bawah nilai array di database Makanan  $high = count($data2); // Batas atas nilai array di database Makanan  $data = BinarySearch( $array, $key, $low, $high ) ;      // Untuk hasil waktu pencarian  $time\_post = microtime(true);  $time = $time\_post - $time\_pre;    // Menampilkan output hasi pencarian  echo "      <div class=panel\_profil>  <div class=head\_profil > Hasil Pencarian Makanan</div>  </div>  </div>  <p>  <table width='100%' border='0'>  <tr bgcolor='#ccc' height=30>  <td align='left' width='65%'><b>Judul</b></td>  <td align='center' width='5%'><b></b></td>  <td align='center'width='5%'><b>MP3</b></td>  <td align='center'width='20%'><b>Lihat Makanan</b></td>    </tr>  <tr>  <td align='left'>$data2[$data]</td>  <td align=center><a href=\"#\" onclick=\"window.open('.php?id=".trim($database[$data]['id\_Makanan'])."','newWindow','height=800,width=500');\"></a></td>  <td align=center><a href=\"#\" onclick=\"window.open('admin/mp3/".trim($database[$data]['judul\_Makanan']).".mp3','newWindow','width=500,height=400,left=400')\";><img src='img/icon.png' width=30></td>  <td align=center><a href='?menu=detail\_Makanan&id=".$database[$data]['id\_Makanan']."'>Detail</a></td>  </tr>  </table>  <h4>Waktu pencarian : [ ".sprintf("%.4f",$time)." ] seconds </h4>  ";    } |