**RANCANG BANGUN SISTEM INFORMASI PENJUALAN**

**PADA RESTORAN MAMMA MASAK**



**PAPER UAS**

Diajukan Untuk Memenuhi Tugas Pemrograman Akuntansi II

**AMALIA SYIFA 11181272**

**ANGGI RIANTIKA DEWI 11180350**

**FENTI YULYA HALIM 11181055**

**NADA HORIYA 11180800**

**PASKARIA HARIANJA 11180305**

**Program Studi Sistem Informasi Akuntansi**

**Fakultas Teknologi Informasi**

**Universitas Bina Sarana Informatika**

**Jakarta**

**2020**

**KATA PENGANTAR**

Puji Syukur kehadirat Tuhan Yang Maha Kuasa atas segala limpahan Rahmat-Nya sehingga kami dapat menyelesaikan penyusunan paper UAS ini yang berjudul **“Rancang Bangun Sistem Informasi Penjualan Pada Restoran Mamma Masak”.** Semoga makalah ini dapat dipergunakan sebagai salah satu acuan, petunjuk maupun pedoman bagi pembaca dalam mempelajari matakuliah Pemrograman Akuntansi II. Harapan saya semoga makalah ini membantu menambah pengetahuan dan pengalaman bagi para pembaca.

Kami juga mengucapkan terima kasih kepada Dosen kami Bapak Deni Gunawan, M.kom yang telah membantu mengarahkan kami sehingga paper UAS ini dapat selesai tepat pada waktunya. Paper UAS ini kami akui masih banyak kekurangan karena pengalaman yang kami miliki sangat kurang. Oleh karena itu, kami harapkan kepada pembaca untuk memberikan masukan-masukan yang bersifat membangun untuk kesempurnaan paper UAS ini.

Jakarta, 08 Juli 2020

Penyusun

**DAFTAR ISI**

Lembar Judul

Kata Pengantar

Daftar Isi

Daftar simbol

Daftar Gambar

Daftar tabel

**BAB I PENDAHULUAN........................................................................................................... 1**

* 1. Latar Belakang Masalah......................................................................................................... 1
  2. Tujuan dan Manfaat................................................................................................................ 1
  3. Metode Penelitian.................................................................................................................... 2

1.3.1. Metode Pengumpulan Data........................................................................................... 2

1.3.2 Metode Pengembangan Perangkat Lunak...................................................................... 3

1.4 Ruang Lingkup........................................................................................................................ 4

**BAB II LANDASAN TEORI..................................................................................................... 5**

2.1 Konsep Dasar.......................................................................................................................... 5

2.1.1 Pengertian Sistem.......................................................................................................... 5

2.1.2 Karakteristik Sistem...................................................................................................... 5

2.1.3 Klasifikasi Sistem......................................................................................................... 7

2.1.4 Pengertian Internet........................................................................................................ 9

2.1.5 Pengertian Website....................................................................................................... 9

2.1.6 Bahasa Program (Script Program)............................................................................... 10

2.1.7 Sistem Informasi Akuntansi........................................................................................ 11

2.1.8 Penjualan Tunai dan Kredit......................................................................................... 11

2.2 Peralatan Pendukung............................................................................................................. 12

2.2.1 Unified Modeling Language (**UML**).......................................................................... 12

2.2.2 Entity relationship Diagram**(ERD**)............................................................................. 14

2.2.3 Logical Relational Structure (**LRS**)…........................................................................ 15

2.2.4 Pengujian Web............................................................................................................ 16

2.2.5 Blackbox-Testing........................................................................................................ 16

**BAB III PEMBAHASAN......................................................................................................... 18**

3.1 Tinjauan Perusahaan............................................................................................................... 18

3.1.1 Sejarah Perusahaan....................................................................................................... 18

3.1.2 Struktur Organisasi dan Fungsi.................................................................................... 18

3.2 Tinjauan Kasus....................................................................................................................... 20

3.2.1 Proses Bisnis Sistem Berjalan...................................................................................... 20

3.2.2Activity Diagram.......................................................................................................... 22

3.2.3 Dokumen Masukan...................................................................................................... 22

3.2.4 Dokumen Keluaran...................................................................................................... 23

3.2.5 Permasalahan Pokok.................................................................................................... 24

3.2.6 Pemecahan Masalah..................................................................................................... 25

3.3 Analisis Kebutuhan Software................................................................................................. 25

3.3.1 Analisis Kebutuhan........................................................................................................ 25

3.3.2 Use Case Diagram.......................................................................................................... 26

3.3.3 Activity Diagram........................................................................................................... 28

3.4 Desain..................................................................................................................................... 29

3.4.1 *Entity Relationship Diagram*(ERD)............................................................................... 29

3.4.2 *Logical Record Structure*(LRS)..................................................................................... 30

3.4.3 Spesifikasi File............................................................................................................... 31

3..4 Sequence Diagram........................................................................................................... 36

3.4.5 Deployment Diagram..................................................................................................... 37

3.4.6 User Interface................................................................................................................. 37

3.5 Implementasi........................................................................................................................... 41

3.5.1 Code Generation............................................................................................................ 41

3.5.2 Blackbox Testing........................................................................................................... 75

3.5.3 Spesifikasi Hardware dan Software............................................................................... 77

**BAB IV PENUTUP.................................................................................................................... 79**

4.1 Kesimpulan............................................................................................................................ 79

4.2 Saran....................................................................................................................................... 79

**DAFTAR PUSTAKA................................................................................................................. 80**

**DAFTAR SIMBOL**

**Daftar simbol Activity diagram**

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
| https://1.bp.blogspot.com/-8iXGpasv5sw/XoAiSJCySPI/AAAAAAAABAk/o5LrrfHszX08EvQpZo9leNGDbRfaNU3FwCLcBGAsYHQ/s1600/28.JPG | Status   awal | Sebuah diagram  aktivitas memiliki sebuah status  awal. |
| https://1.bp.blogspot.com/-U60e55A9_wE/XoAiWjWmqQI/AAAAAAAABAo/qr_m_Iqz22ADwVXJD9Lq9MnvGXQXPAASwCLcBGAsYHQ/s1600/29.JPG | Aktivitas | Aktivitas yang  dilakukan sistem,aktivitas  biasanya diawali   dengan kata kerja. |
| https://1.bp.blogspot.com/-ienDNaQT5sU/XoAibfP9KUI/AAAAAAAABAs/_Z4g9leMjxoaECeCyjo82rM4d0oHnJeXQCLcBGAsYHQ/s1600/30.JPG | Percabangan / Decision | Percabangan  dimana ada  pilihan aktivitas  yang lebih dari  satu. |
| https://1.bp.blogspot.com/-tiFv0q8EpH8/XoAiqyNM_BI/AAAAAAAABA0/T6HN29bvqSEdWS-41pw7Dic6V7lEEXlhQCLcBGAsYHQ/s1600/31.JPG | Penggabungan / Join | Penggabungan  dimana  yang  mana lebih dari  satu aktivitas lalu digabungkan  jadi satu. |
| https://1.bp.blogspot.com/-eL2fJ5XBsig/XoAiw8SN1aI/AAAAAAAABA8/qClHv83BpKAZPDNHT_FkS-OMzKSdLt4TQCLcBGAsYHQ/s1600/32.JPG | Status Akhir | Status akhir yang   dilakukan    sistem, sebuah  diagram aktivitas     memiliki sebuah  status akhir |
| https://1.bp.blogspot.com/-q8yoa2_yczE/XoAi4VDRYOI/AAAAAAAABBA/fNspEKEDYIIPBV7P-31vonrEquPsgmnGQCLcBGAsYHQ/s1600/33.JPG | Swimlane | Swimlane  memisahkan organisasi  bisnis  yang bertanggung jawab terhadap aktivitas yang  terjadi |

**Daftar Simbol use case diagram**

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
| Actor | Actor | Merupakan peran orang,  sistem yang lain, atau  alat ketika berhubungan  dengan use case. |
| Use Case | Use Case | Abstraksi dari penghubung  antara aktor dengan  use case. |
|  | Association | Abstraksi dari penghubung  antara aktor dengan  use case. |
| Generalisasi | Generalisasi | Menunjukkan spesialisasi  aktor untuk dapat  berpartisipasi dengan  use case. |
| Include | Include | Menunjukkan bahwa  suatu use case seluruhnya  merupakan fungsionalitas  dari use case lainnya. |
|  | Extend | Menunjukkan bahwa  suatu use case  merupakan tambahan  fungsinonal dari use case  lainnya jika suatu kondisi  terpenuhi. |

**Daftar simbol sequence**

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
| https://1.bp.blogspot.com/-uHGZk8RfJpc/Xn7dpsbkWRI/AAAAAAAAA-E/FDMxmDxG2eUfchpuIDjw7-HkFRnnpDWbgCPcBGAYYCw/s1600/16.JPG | Aktor | merepresentasikan  entitas yang berada  di luar sistem dan berinteraksi dengan  sistem |
| https://1.bp.blogspot.com/-wLnzy7ZYs7Y/XoP079udWNI/AAAAAAAABFA/Ge-NiAbA8ks9SApzoulfRfhnniscqxaVACLcBGAsYHQ/s1600/36.JPG | Lifeline | menghubungkan  objek selama  sequence (message  dikirim atau diterima dan aktifasinya). |
| https://1.bp.blogspot.com/-BkCxvmjoSdY/XoP1KRtiT4I/AAAAAAAABFE/kKVpODbY32ABUeNaINupbIIZ7MFkvNfJgCLcBGAsYHQ/s1600/37.JPG | General | Merepresentasikan  entitas tunggal dalam sequence diagram. |
| https://1.bp.blogspot.com/-xrrrHhB0lE8/XoP1Ty0K4jI/AAAAAAAABFM/R8YxJ3Qri_007WvPcQ30cJ1hSmLzKxeYgCLcBGAsYHQ/s1600/38.JPG | Boundary | berupa tepi dari  sistem, seperti user interface atau suatu  alat yang berinteraksi dengan sistem yang  lain. |
| https://1.bp.blogspot.com/-djfB_Btv1is/XoP1uJ6OL7I/AAAAAAAABFY/HlVcf1HmxfYEBKth8L-7gEEVj2aknNj8ACLcBGAsYHQ/s1600/39.JPG | Control | element mengatur aliran dari informasi  untuk sebuah skenario. Objek ini umumnya mengatur perilaku dan perilaku  bisnis. |
| https://1.bp.blogspot.com/-SygnXbSj64Q/XoP11GvMNgI/AAAAAAAABFc/QLHGc5cM_aQr3FelNYK8p4ouNAn5FjzDACLcBGAsYHQ/s1600/40.JPG | Entitas | elemen yang  bertanggung jawab menyimpan data atau informasi. Ini  dapat berupa beans  atau model object. |
| <https://1.bp.blogspot.com/-nUpod0h1t8E/XoP19QyMEZI/AAAAAAAABFg/11Ca6hozvQ46JVcRlnL7V4Al15Oe7a67wCLcBGAsYHQ/s1600/42.JPG> | Activation | suatu titik dimana  sebuah objek mulai berpartisipasi di  dalam sebuah  sequence yang menunjukkan kapan  sebuah objek  mengirim atau  menerima objek. |
|  | Message | berfungsi sebagai komunikasi antar  objek yang  menggambarkan  aksi yang akan dilakukan |
| <https://1.bp.blogspot.com/-5r_Ex4vd44Y/Xn7d3fB3HhI/AAAAAAAAA-Q/xHDeQm03vB0lH5Ge1nxR08_pzS4ggDO3gCPcBGAYYCw/s1600/18.JPG> | Message Entry | berfungsi untuk menggambarkan pesan/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi. |
| https://1.bp.blogspot.com/-Hu9bhfZB0_E/XoP3jlAMeAI/AAAAAAAABF0/qDY_EkPcqYg2ey-B7qZzD0nA2jI8OuZHgCLcBGAsYHQ/s1600/44.JPG | Message to Self | Simbol ini  menggambarkan pesan/hubungan  objek itu sendiri,  yang menunjukkan  urutan kejadian yang terjadi. |
| https://1.bp.blogspot.com/-2j92of0--oY/XoP3swXSQEI/AAAAAAAABF4/XPwpzOqPTJcRVswoCYdvamb-WDPzyG-LwCLcBGAsYHQ/s1600/43.JPG | Message Return | menggambarkan  hasil dari pengiriman message dan  digambarkan dengan  arah dari kanan ke kiri. |

**Daftar simbol deployment**

|  |  |  |
| --- | --- | --- |
| Simbol | Nama | Keterangan |
| Simbol Deployment Diagram | package | Package adalah sebuah bungkusan dari satu atau lebih node. |
| Simbol Deployment Diagram | **node** | Node biasanya mengacu pada hardware, perangkat lunak yang tidak dibuat sendiri (software). Jika didalam node disertai komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelunya pada diagram komponen. |
| Simbol Deployment Diagram | Dependency | Dependency adalah kebergantungan antar node, arah panah mengarah pada node yang dipakai. |
| Simbol Deployment Diagram | **link** | Link merupakan relasi antar node. |

**Daftar Simbol Entity Relationship Diagram (ERD)**

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | entitas | Entitas ialah suatu objek yang dapat dibedakan dengan objek lainnya. Entitas berfungsi untuk memberikan identitas pada entitas yang memiliki label dan nama. Entitas memiliki bentuk persegi panjang. |
|  | Relasi/Hubungan Antar Entitas (relationship) | Relasi ialah hubungan yang terjadi antara 1 entitas atau lebih yang tidak mempunyai fisik tetapi hanya sebagai konseptual. Dan berfungsi untuk mengetahui jenis hubungan yang ada antara 2 *file*. Relisi memiliki bentuk belah ketupat. |
|  | Atribut | Atribut ialah karakteristik dari entitas atau relasi yang menyediakan penjelasan detil tentang entitas atau relasi tersebut. Dan berfungsi untuk memperjelas atribut yang dimiliki oleh sebuah entitas. Atribut memiliki bentuk lingkarang lebih tepatnya elips. |
|  | Alur | Alur memiliki fungsi untuk menghubungkan atribut dengan entitas dan entitas dengan relasi. Dan berbentuk garis. |

**DAFTAR GAMBAR**

1. III.1 Struktur Organisasi Restoran Mamma Masak.....................................................18
2. III.2 Activity Diagram Sistem Berjalan.......................................................................22
3. III.3 Use Case Diagram Administrator........................................................................26
4. III.4 Activity Diagram Administrator Sistem Berjalan................................................28
5. III.5 Entity Realationship Diagram(ERD)...................................................................29
6. III.6 Logical Record Structure(LRS)...........................................................................30
7. III.7 Sequence Diagram...............................................................................................36
8. III.8 Deployment Diagram...........................................................................................37
9. III.9 User Interface Login............................................................................................37
10. III.10 User Interface Beranda......................................................................................38
11. III.11 User Interface Menu..........................................................................................38
12. III.12 User Interface Bahan..........................................................................................39
13. III.13 User Interface Supplier......................................................................................39
14. III.14 User Interface User............................................................................................40
15. III.15User Interface Pembelian....................................................................................40
16. III.16 User Interface Penjualan....................................................................................41

**DAFTAR TABEL**

Tabel III.1 Deskripsi Use Case Diagram Administrator............................................................... 26

Tabel III.2 Deskripsi file table Admin.......................................................................................... 31

Tabel III.3 Spesifikasi file table Pembelian.................................................................................. 32

Tabel III.4 Spesifikasi file table Supplier..................................................................................... 32

Tabel III.5 Spesifikasi file table Detail Pembelian....................................................................... 33

Tabel III.6 Spesifikasi file table Bahan Baku............................................................................... 34

Tabel III.7 Spesifikasi file table Menu......................................................................................... 34

Tabel III.8 Spesifikasi file table Kategori..................................................................................... 35

Tabel III.9 Spesifikasi file table Penjualan................................................................................... 36

Tabel III.10 Hasil pengujian blackbox testing form login admin................................................. 75

Tabel III.11 Hasil pengujian blackbox testing form daftar register Customer............................. 76

Tabel III.12 Hasil pengujian blackbox testing form Transaksi..................................................... 76

**BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang Masalah**

Restoran yang tersebar di beberapa tempat membuat restoran saling bersaing untuk mendapatkan pelanggan. Untuk menjaga loyalitas pelanggan dan memenangkan persaingan, banyak strategi yang ingin dikembangkan oleh restoran. Salah satu strategi yang ingin ditingkatkan adalah mengenai pelayanan *customer*.

Berdasarkan survey pada restoran Mamma Masak, terdapat permasalahan utama pada restoran mengenai proses pelayanan yang lamban. Yang pertama, pelayan mengalami kesulitan dalam proses persiapan makanan karena nota yang tidak teratur dan tidak mengetahui costumer mana yang lebih dulu datang.

Dan yang kedua proses penjumlahan pada sistem 1 nota masih menggunakan sistem manual yaitu nota, pulpen dan kalkulaktor.

Berdasarkan permasalahan diatas restoran membutuhkan aplikasi yang membuat semua data saling terintegrasi, diharapkan dengan adanya aplikasi ini bisa membuat pelayan dapat dengan mudah berkerja dan mengakses pesanan costumer, dan costumer bisa mendapatkan pelayanan yang cepat, tanggap dan menyenangakan.

* 1. **Tujuan dan Manfaat**

Tujuan dari penulisan tugas akhir ini yaitu sebagai berikut:

1. Proses nota yang taratur
2. Proses penjumlahan makanan
3. Stok bahan baku makanan
4. Lapoan Penjualan dan pembelian

Manfaat dari penulisan tugas akhir ini yaitu sebagai berikut:

1. Manfaat untuk penulis

Sebagai syarat memenuhi tugas matakuliah Pemprograman Akuntansi II Program Studi Sistem Informasi Akuntansi di Universitas Bina Sarana Informatika dan menerapkan ilmu yang sudah dipelajari selama masa perkuliahan.

1. Manfaat untuk objek penelitian

Mengenal sistem informasi guna membantu restaurant agar mendapatkan data hasil penjualan yang cepat dan akurat.

1. Manfaat untuk pembaca

Menambah wawasan ilmu tentang perancangan sistem informasi akuntansi dan cara penerapannya.

* 1. **Metode Penelitian**
     1. **Teknik Pengumpulan Data**

Metode pengumpulan data yang dilakukan oleh penulis dalam pembuatan Paper Uas ini adalah sebagai berikut:

1. Metode Observasi

Pengumpulan data dilakukan dengan pengamatan secara langsung terhadap objek  
penelitian yakni Mamma Masak, dengan mencatat hal-hal penting yang  
berhubungan dengan tema penulisan ini, sehingga diperoleh data yang  
lengkap dan akurat. Fokus pengamatan dilakukan pada sistem penjualan makanan.

1. Metode Wawancara

Pengumpulan data dilakukan dengan cara wawancara atau tanya jawab secara  
langsung terkait dengan sistem penjualan makanan.

1. Metode Studi Pustaka  
   Pengumpulan data dilakukan dengan cara melihat sumber-sumber tertulis serta jurnal ilmiah yang berhubungan dengan masalah yang sedang dibahas guna memperoleh gambaran secara teoritis.
   * 1. **Metode Pengembangan Perangkat Software**

Metode pengembangan *software* yang dilakukan dalam penelitian ini adalah  
menggunakan metode *waterfall* (Sukamto & Shalahuddin, 2016):

1. analisis kebutuhan *software*

Merupakan tahap analisa terhadap sistem yang berjalan dengan tujuan untuk merancang sistem usulan menggunakan *Neatbeans dan Xampp.*

1. *desain*  
    Desain adalah proses pembuatan rancangan perangkat lunak termasuk struktur data,arsitektur, perangkat lunak, representasi antar muka, *entity relationship diagram* dan *logical relational structure*. Adapun desain model menggunakan diagram *Unified Modelling Language*, desain *database* dilakukan dengan cara mengidentifikasi file-file yang diperlukan dalam sistem informasi usulan melihat desain model berbentuk diagram UML yang dikembangkan menggunakan teknik ERD dan LRS.
2. Pengkodingan (*Code generation*)  
    Pada tahap ini, penulis merancang suatu sistem usulan berdasarkan desain yang dibuat menggunakan diagram *Use Case*, *Activity*, ERD dan LRS ke dalam bentuk sebuah aplikasi *website* menggunakan *Neatbeans dan Xampp.*Aplikasi *website* ini berfungsi untuk memudahkan para calon pembeli dengan  
   pelayan penjualan dalam melakukan transaksi penjualan*.*
3. Pengujian (*Testing*)  
    Dalam penyusunan Tugas Akhir ini, penulis melakukan pengujian aplikasi yang telah dibuat menggunakan *blackbox testing,* yaitu metode yang memfokuskan kepada kebutuhan *software*, pengujian ini bertujuan untuk menemukan kesalahan.
4. Pendukung *(Support)* atau Pemeliharaan (*maintenance*)

Tidak tertutup kemungkinan aplikasi *website* yang sudah diimplementasikan mengalami perubahan dikarenakan adanya kesalahan yang tidak terdeteksi saat tahap pengujian. Sedangkan pemeliharaan dapat mengulangi proses pengembangan sistem dari analisa spesifikasi untuk perubahan perangkat lunak yang sudah ada namun tidak untuk membuat perangkat lunak baru.

* 1. **Ruang Lingkup**

Adapun ruang lingkup dalam penulisan Tugas ini membahas tentang  
sistem penjualan makanan dengan memberi batasan yang meliputi, *input*data *user, input* data customer, *input* data barang, *input* data supplier, *input* data bahan baku, *input* data pembelian *, input* transaksi penjualan serta  
pembuatan laporan penjualan makanan yang sudah terkomputerisasi menggunakan bahasa pemograman PHP dengan menggunakan *software Neatbeans* dan *database MySQL* dengan aplikasi *Xampp.* Aplikasi yang akan dirancang adalah sistem  
penjualan berbasis *web.*

**BAB II  
LANDASAN TEORI**

**2.1. Konsep Dasar**

Suatu sistem dapat terdiri dari sistem-sistem bagian (subsistem). Subsistem  
yang saling berinteraksi dan saling berhubungan membentuk suatu kesatuan sehingga  
tujuan atau sasaran sistem tersebut dapat tercapai. Interaksi dari subsistem-subsistem  
terjalin sedemikian rupa, sehingga dicapai suatu kesatuan yang terpadu atau  
terintegrasi

**2.1.1 Pengertian sistem** Menurut Atmosudirdjo dalam (Sutabri,2016) menyatakan bahwa:  
“suatu sistem terdiri atas objek-objek, atau unsur-unsur, atau komponen-komponen  
yang berkaitan atau berhubungan satu sama lainya sedemikian rupa sehingga unsur-unsur tersebut merupakan suatu kesatuan pemrosesan atau pengolahan yang  
tertentu”.  
 Menurut (Sutabri, 2016) menyatakan bahwa : Pendekatan yang menekankan pada prosedur mendefinisikan sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau komponen mendefinisikan system sebagai kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.

**2.1.2. Karakteristik Sistem** Menurut (Mustakini, 2017) model umum sebuah sistem terdiri dari *input*,  
proses dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana  
mengingat sebuah sistem dapat mempunyai beberapa masukan dan pengeluaran.  
Adapun karakteristik yang di maksud adalah sebagai berikut:

**1.** Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih basar, yang disebut dengan Supra sistem.

1. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara system dengan sistem lainya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan-pisahkan.

1. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, yang dengan demikian lingkungan luar tersebut harus selalu dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus  
dikendalikan, karena kalau tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

1. Penghubung Sistem (*Interface*)

Sebagai media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain. Keluaran suatu subistem akan menjadi masukan untuk subsistem yang lain dengan melewati penghubung. Dengan demikian terjadi suatu integrasi yang membentuk satu kesatuan.

1. Masukan Sistem (*Input*)

Energi yang di masukan ke dalam sistem disebut masukan sistem. Yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer,”program” adalah *maintenance input* yang digunakan untuk mengoprasikan komputer sementara “data” adalah *signal input* yang akan diolah menjadi informasi.

1. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang  
berguna, keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, yang mana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan input bagi subsistem lainya.

1. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh, sistem akutansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

1. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

**2.1.3. Klasifikasi Sistem**

Menurut (Sutabri, 2016) Sistem merupakan bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi di dalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang sebagai berikut:

1. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak nampak secara fisik, misalnya sistem theologi. Yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan, sedangkan system fisik merupakan sistem yang ada secara fisik, seperti sistem komputer, system produksi, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

1. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

1. Sistem deterministik dan sistem probabilistic

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang di jalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

1. Sistem terbuka dan sistem tertutup

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sedangkan sistem terbuka adalah system yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

**2.1.4. Pengertian Internet**

Menurut Rossa dan Shalahudin dalam (Supriyatna, 2015), *Internet* atau *internetwork* adalah sekumpulan jaringan berbeda yang saling berhubungan bersama sebagai salah satu kesatuan dengan menggunakan berbagai macam protocol, salah satunya adalah protocol TCP/IP (*Transmission Control Protocol*/*Internet Protocol*).

**2.1.5. Pengertian *Website***

Menurut raharjo dalam (Supriyanta & Nisa, 2015), "*World Wide Web* (WWW), sering disingkat dengan web, adalah suatu layanan di dalam jaringan yang berupa ruang informasi”. Sedangkan menurut Sibero dalam (Supriyanta & Nisa, 2015), “*World Wide Web* (W3) atau yang dikenal juga dengan istilah web adalah suatu sistem yang berkaitan dengan dokumen digunakan sebagai media untuk menampilkan teks, gambar, multimedia dan lainnya pada jaringan internet”.

Selanjutnya mari kita bahas pengertian dan perbedaan *web* statis dan *web*dinamis menurut (Ferdianto, 2013), yaitu sebagai berikut:

1. *Web* Statis

*Web* statis adalah *website* yang mana pengguna tidak bisa mengubah konten dari *web* tersebut secara langsung menggunakan *browser*. Interaksi yang terjadi antara pengguna dan *server* hanyalah seputar pemrosesan link saja. Halamanhalaman *web* tersebut tidak memliki *database*, data dan informasi yang ada pada *web* statis tidak berubah-ubah kecuali diubah sintaksnya. Dokumen *web* yang dikirim kepada *client* akan sama isinya dengan apa yang ada di *webserver*. Contoh dari *web* statis adalah *web* yang berisi profil perusahaan. Di sana hanya ada beberapa halaman saja dan kontennya hampir tidak pernah berubah karena konten langsung diletakan dalam file HTML saja.

1. *Web* Dinamis

Dalam *web dinamis*, interaksi yang terjadi antara pengguna dan *server* sangat kompleks. Seseorang bisa mengubah konten dari halaman tertentu dengan menggunakan *browser*. *Request* (permintaan) dari pengguna dapat diproses oleh *server* yang kemudian ditampilkan dalam isi yang berbeda-beda menurut alur programnya. Halaman-halaman *web* tersebut memiliki database. *Web* dinamis, memiliki data dan informasi yang berbeda-beda tergantung *input*apa yang disampaikan *client*. Dokumen yang sampai di *client* akan berbeda dengan  
dokumen yang ada di *webserver*. Contoh dari *web* dinamis adalah portal berita dan jejaring sosial. Lihat saja *web* tersebut, isinya sering diperbaharui (di*update*) oleh pemilik atau penggunanya. Bahkan untuk jejaring sosial sangat sering di-*update* setiap harinya.

**2.1.6. Bahasa Program *(script program*)**

Bahasa program merupakan bahasa yang digunakan untuk menterjemahkan setiap perintah pada saat *website* tersebut sedang dijalankan. Bahasa program yang penulis gunakan adalah *HTML dan PHP*.

1. HTML

Menurut Raharjo dalam (Supriyanta & Nisa, 2015), HTML adalah singkatan dari *HyperText Markup Language*, yaitu bahasa (aturan) standar yang digunakan untuk menampilkan teks, gambar, video atau audio ke dalam halaman web. Bahasa HTML merupakan bahasa yang digunakan untuk membuat halaman *website* dengan menggunakan tag-tag yang telah dideklarasikan pada halaman *notepad* dan dapat saling berhubungan dengan dokumen HTML lainnya atau yang sering disebut dengan istilah *link*.

1. PHP

Menurut (Winarno& Zaki, 2013) “PHP adalah sebuah bahasa pemrograman *scripting* untuk membuat halaman *web* yang dinamis. Maksud dari *scripting* adalah pembuatannya menggunakan editor teks biasa, seperti *Notepad* , *Notepad* ++ dan lainnya”.

1. CSS

Menurut (Hidayatullah& Kawistara, 2014) ”dengan CSS kita bisa menyimpan format dan menggunakannya kapanpun dan dimanapun kita inginkan.” Dengan CSS kita bisa membuat efek-efek tertentu untuk konten *web* yang kita buat. Contohnya kita ingin bagian *header* pada *web* selalu diberi *fontVerdana*, *size* 20 dan warna biru. Maka kita definisikan *style* tersebut untuk tag <TH>. CSS (*Cascading Style Sheet*) sudah didukung oleh hampir semua *web browser* membuat karena CSS telah distandarkan oleh *World Wide Web Consortium (W3C).* Jadi pilihan yang tepat untuk kita format halaman *web* kita agar terlihat lebih indah dan cantik dimanapun *user* membuka *web* kita.

**2.1.7. Sistem Informasi Akuntansi**

Menurut (Marshall & Paul, 2016) “Sistem Informasi Akuntansi (SIA) adalah sebuah sistem yang mengumpulkan, mencatat, menyimpan, dan mengolah data untuk menghasilkan informasi bagi para pembuat keputusan “

**2.1.8 Penjualan Tunai & Kredit**

Menurut (Sholikhah, Sairan, & Syamsiah, 2017) mengatakan bahwa: Penjualan secara umum ada 2, yaitu penjualan tunai dan penjualan kredit. Penjualan yang dilakukan dengan tidak ada sistem hutang disebut dengan penjualan tunai. Sedangankan penjualan yang dilakukan dengan system hutang disebut dengan penjualan kredit. Penjualan merupakan pendapatan yang diperoleh dari menjual barang yang mana jumlah yang dibebankan kepada pembeli untuk barang dagang yang diserahkan merupakan pendapatan perusahaan yang bersangkutan.

**2.2. Peralatan Pendukung**

**2.2.1. *Unified Modelling Language***

Menurut (Sukamto & Shalahuddin, 2016) “UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefisinikan requirement, membuat analisi dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek” UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari system perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dari teks-teks pendukung. Pada UML 2,3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori yaitu *structure diagrams* yaitu yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. Terdiri dari *class diagram*, *object diagram*, *component diagram*, *composite diagram*, *package diagram*, *deployment diagram*. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang digunakan untuk  
menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. Terdiri dari, *use case diagram*, *activity diagram*, *state machine diagram. Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sustem lain maupun interaksi antar subsistem pada suatu sistem. Terdiri dari *sequence diagram*, *communication diagram*, *timing diagram*, *interaction overview diagram* (Sukamto & Shalahuddin, 2016). Berikut penjelasan mengenai diagram-diagram UML menurut (Sukamto &  
Shalahuddin, 2016):

1. *Use Case Diagram*

*Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahi fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

1. *Activity Diagram*

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktiitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

1. *Deployment Diagram*

Diagram *deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat

digunakan untuk memodelkan hal-hal berikut, yaitu sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node*, dan *hardware*, sistem *client/server*, sistem terdistribusi murni dan rekayasa ulang aplikasi.

1. *Component Diagram*

Diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem.

**2.2.2. *Entity Relationship Diagram***

*Entity Relationship Diagram (ERD) digunakan dalam perancangan*

*database.* Menurut Rosa dan Shalahuddin dalam (Mulyadi & Sinnun, 2018) *Entity Relation Diagram* (ERD) adalah bentuk awal dalam melakukan perancangan basis data relasional. ERD merupakan suatu model yang menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. Pada dasarnya ada tiga simbol yang digunakan sebagai berikut:

1. Entitas

Digambarkan dengan kotak persegi panjang dan digunakan untuk menunjukkan sekumpulan orang,tempat, objek atau konsep dan sebagainya yang menunjukkan dimana data dicatat atau disimpan.

1. Hubungan atau Relasi

Digambarkan dengan kotak berbentuk *diamond* atau belah ketupat dengan garis yang menghubungkan ke entitas yang terkait. Maka *relationship* diberi nama dengan kata kerja. Hubungan atau relasi menunjukkan abstraksi dari sekumpulan hubungan yang mengaitkan antara entitas yang berbeda.

1. Atribut

Digambarkan dengan bentuk elips. Atribut menunjukkan karakteristik dari tiap entitas atau sesuatu yang menjelaskan entitas atau hubungan. Sehingga atribut dikatakan elemen dari entitas dan relasi. Dari setiap atribut entitas terdapat satu atribut yang dijadikan sebagai kunci (*key*). Beberapa jenis kunci tersebut antara lain :*Primary key, Candidate key, Composite key, Secondary key, Alternate key* dan *Foreign key*.

1. Tingkat Hubungan (*Cardinality*)

*Entity Relation Diagram* (ERD) juga menunjukkan tingkat hubungan yang

terjadi, dilihat dari segi kejadian atau banyak tidaknya hubungan antara entitas tersebut.

**2.2.3. *Logical Record Structure***

Menurut (Hasugian, 2012 ) menjelaskan bahwa :

LRS *(Logical Record Structure)* merupakan hasil proses transformasi dari ERD dimana *primary key* yang terdapat pada masing-masing relasi akan masuk pada entitas yang lebih kuat. Proses transformasi digambarkan dengan persegi empat dengan garis putus-putus. Pada LRS akan kelihatan setiap *primary key* yang akan menjadi *foreign key* pada entitas lain. LRS ini menjadi patokan untuk pembuatan *Conceptual Data Model.*

**2.2.4. Pengujian *Web***

Pengujian *web* dapat dilakukan dengan dua macam metode pengujian  
perangkat lunak yaitu *blackbox* atau *blackbox testing*. *Black-box testing* yaitu perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah (Sukamto dan Shalahuddin, 2016 ), misalkan untuk kasus proses *login* maka kasus uji coba yang dilakukan adalah:

1. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar.

*2.* Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalkan nama pemakai benar tapi kata sandi salah, atau sebaliknya atau keduanya salah.

**2.2.5. *Black-Box Testing***

Menurut (Sukamto dan Shalahuddin, 2016) mengatakan bahwa:

*“Black-Box testing* (pengujian kotak hitam) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi,masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan”. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus *login* maka kasus uji yang dibuat adalah :

1. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*)  
yang benar.

2. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*)  
yang salah, misalnya nama pemakai benar tapi kata sandi salah, atau  
sebaliknya, atau keduanya salah.

**BAB III**

**PEMBAHASAN**

**3.1 Tinjauan Perusahaan**

Dalam tinjauan perusahaan ini, penulis akan menjelaskan tentang sejarah  
perusahaan dan struktur organisasinya beserta tugas dan tanggung jawabnya.

**3.1.1 Sejarah Perusahaan**

Mamma Masak adalah perusahaan yang bergerak dibidang penjualan makanan dan minuman. Berdiri sejak tahun 2010 dan didirikan oleh Ibu Marwah Rianti dengan memiliki 6 karyawan dengan bagian masing masing. Mamma Masak menjalankan usahanya dengan membeli bahan baku mentah sayuran, buah-buahan dan bumbu dan dimasak menjadi berbagai macam makanan dan minuman. Namun setelah tahun 2019 Mamma Masak mengalami kendala mengenai pembelian produk dari pemasok yaitu nota pembelian masih menggunakan system manual dan dan nota-nota sering bercecran dan tidak teratur. Akibat dari kendala tersebut akhirnya Mamma Masak memutuskan untuk menggunakan aplikasi yang dapat dicacat oleh 1 orang.

**3.1.2 Struktur Organisasi dan Fungsi**

**OWNER**

**Manager Area**

**Assistant Manager**

**Supervisor**

**Leader**

**Pelayan**

**Kasir**

**Chef**

**Staff Pergudangan**

**Staff Pembukuan**

**Gambar III.1 Struktur Organisasi Mamma Masak**

Adapun fungsi dan tugas dari masing-masing bagian dalam struktur organisasi  
sebagai berikut :

1. Presiden Direktur

a. Sebagai pemimpin perusahaan

b. Merencanakan, mengawasi dan menilai hasil kerja direktur

c. Menetapkan kebijaksanaan untuk mencapai tujuan perusahaan

d. Menandatangani kontrak kerja atau kerja sama dengan pihak luar

e. Mengendalikan kekayaan perusahaan

2. Direktur

a. Bertindak sebagai presiden direktur jika presiden direktur berhalangan  
dan memberikan usulan kepada presiden direktur demi kemajuan perusahaan.

b. Memimpin, mengarahkan, menugaskan, mengawasi seluruh jajaran yang berada di bawah tanggung jawabnya.

c. Memastikan bahwa seluruh karyawan yang berada di bawah tanggung jawabnya menjalankan tugasnya dengan baik, efektif, dan efisien.

3. Sales

a. Mengolah kegiatan penjualan dan menjual barang dagang kepada *customer.*

b. Melaporkan hasil penjualan yang terjadi setiap hari.

c. Mempromosikan suatu inovasi produk terbaru kepada para *customer.*

4. Gudang

a. Mengantisipasi adanya keterlambatan dalam pengiriman barang yang  
dibutuhkan oleh perusahaan.

b. Mengantisipasi jika adanya pesanan barang yang salah sehingga barang  
tersebut harus di-retur kembali.

c. Mengantisipasi terjadinya inflasi atau kenaikan harga barang secara tiba-tiba.

d. Memperkirakan untung dari pembelian yang dilakukan berdasarkan *quantity discount* atau potongan kuantitas.

e. Memberikan pelayanan terhadap konsumen dengan adanya ketersediaan  
barang yang dibutuhkan oleh konsumen.

f. Penerimaan adalah proses untuk menerima barang pesanan perusahaan,  
dengan menjamin kuantitas barang yang dikirim oleh pihak *supplier.*

g. Persediaan adalah kegiatan untuk menjamin agar permintaan dapat dipenuhi sesuai dengan tujuan perusahaan yakni memenuhi kepuasan  
*customer.*

h. Penyimpanan adalah suatu bentuk fisik dari barang-barang yang disimpan sebelum ada permintaan

i. Pengambilan pesanan adalah suatu proses pengambilan barang dari gudang yang sesuai dengan permintaan

j. Pengepakan adalah langkah pilihan setelah proses pengambilan

k. Pengepakan dan pengiriman adalah suatu pemeriksaan barang dalam kontainer hingga terjadinya pengiriman

*5. Finance*

a. Mengatur keluar masuknya kas perusahaan

b. Membuat faktur dan surat jalan

c. Menyiapkan nota tagihan atau tanda terima yang akan di bawa oleh *sales*

d. Menagih *customer* yang telah jatuh tempo

e. Menerima pembayaran dari *customer* dan selanjutnya memproses nota yang telah dibayar

f. Menginput data pembelian barang dari *supplier*

**3.2 Tinjauan Kasus**

**3.2.1 Proses Bisnis Sistem Berjalan**

Sistem pemesanan makanan dan minuman yang saat ini sedang berjalan pada Rumah Makan Mamma Masak masih menggunakan cara manual dan tidak terkomputerisasi. Berikut adalah uraian proses dari kegiatan pemesanan makanan dan minuman yang sedang berjalan pada Rumah Makan Mamma Masak:

1. Calon pembeli datang dan memilih tempat duduk pada Rumah Makan Mamma Masak. Setelah itu, pelayan Rumah Makan Mamma Masak akan menghampiri dan memberikan menu makanan dan minuman.

2. Kemudian, calon pembeli menulis pesanan makanan dan minuman yang akan dipesan dan memberikan pesanan makanan dan minuman kepada pelayan Rumah Makan Mamma Masak.

3. Setelah pesanan makanan dan minuman telah diterima, kemudian pelayan memberikan pesanan ke bagian dapur untuk dilakukan proses pembuatan makanan dan minuman.

4. Makanan dan minuman yang telah selesai dibuat, siap diantarkan oleh pelayan Rumah makan Mamma Makan kepada pembeli dan siap untuk di nikmati pembeli.

5. Kemudian, pelayan memberikan struk total pembelian kepada pembeli. Pembeli memberikan uang kepada pelayan. Jika uang yang diberikan melebihi dari total harga pembelian, maka pelayan akan memberikan uang kembalian beserta bukti pembelian. Jika uang yang diberikan sesuai dengan total pembelian, maka pelayan hanya akan memberikan bukti pembelian saja.

**3.2.2 *Activity Diagram***

**

**Gambar III.2  
*Activity Diagram* Sistem Berjalan**

**3.2.3. Dokumen Masukan**

Dokumen masukan merupakan dokumen dasar yang akan mengalami proses dalam sistem berjalan. Pada sistem akuntansi berjalan ini terdapat beberapa bentuk dokumen masukan yang diperlukan sebagai berikut:

1. Nama Dokumen **:** Bahan

Fungsi : Sebagai Transaksi masuk keluar barang

Sumber : Costumer dan Supplier

Tujuan : Admin

Media : Kertas

Frekuensi : Setiap Transaksi masuk dan keluar barang

Bentuk : Lampiran A-1

1. Nama Dokumen : Menu

Fungsi : Sebagai Menampung data-data masukan menu

Sumber : Chef

Tujuan : Costumer

Media : Kertas

Frekuensi : Setiap ada menu baru

Bentuk : Lampiran A-2

1. Nama Dokumen : Pembelian

Fungsi : Sebagai Bukti pembelian pada costumer

Sumber : Supplier

Tujuan : Finance

Media : Kertas

Frekuensi : Setiap ada pembelian

Bentuk : Lampran A-3

* + 1. **Dokumen Keluaran**

Dokumen Keluaran adalah dokumen yang dihasilkan dari sistem yang berjalan yang sering juga disebut dokumen laporan. Dokumen yang dihasilkan sebagai berikut :

1. Nama Dokumen : Bahan

Fungsi : Sebagai Transaksi masuk keluar barang

Sumber : Costumer dan Supplier

Tujuan : Admin

Media : Kertas

Frekuensi : Setiap Transaksi masuk dan keluar barang

Bentuk : Lampiran B-1

1. Nama Dokumen : Penjualan

Fungsi : Sebagai Bukti Penjualan pada costumer

Sumber : Costumer

Tujuan : Finance

Media : Kertas

Frekuensi : Setiap ada transaksi penjualan

Bentuk : Lampiran B-2

* + 1. **Permasalahan Pokok**

Sistem penjualan yang telah berjalan di Mamma Masak umumnya telah berjalan dengan baik. Dalam proses penginputan bahan baku, proses transaksi pembelian, tetapi penulis melihat adanya kelemahan sistem tersebut.

Adapun kelemahannya adalah sebagai berikut:

1. Sering terjadi selisih dalam masalah pencatatan persediaan barang di gudang dikarenakan pencatatan barang masuk (BM) dan barang keluar (BK) kurang tertib.

2. Dalam kegiatan sehari-hari sistem yang digunakan sering mengalami masalah dan proses pengolahan data yang kurang cepat dan tepat, dikarenakan system yang berjalan masih manual.

* + 1. **Pemecahan Masalah**

Untuk mengatasi masalah tersebut disarankan :

1. Cara efektif yang dapat diambil untuk mengatasi semua permasalahan tersebut hanya dengan cara lebih teliti saat pencatatan data transaksi dan saat pencarian data penjualannya agar tidak terjadi kesalahan data stok barang.

2. Untuk karyawan/karyawati Mamma Masak perlunya komunikasi yang baik atau ditingkatkan kekompakannya agar tidak terjadi keteledoran dalam menginput atau menginput pencatatan bukti transaksi data penjualan, sehingga tidak terjadi selisih dalam pencatatan persediaan barang.

* 1. **Analisis Kebutuhan *Software***

Dalam tahapan perancangan perangkat lunak (*Software*), langkah awal yang perlu dilakukan adalah analisis kebutuhan terhadap para pengguna (*user*) perangkat lunak tersebut. Analis sistem akan menggali informasi sebanyak-banyaknya dari *user* sehingga dapat menjembatani antara *user* dengan *programmer*, sehingga dapat terciptanya sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh narasumber tersebut. Apabila terjadi kesalahan dalam tahapan analisis kebutuhan, maka perangkat lunak yang dibuat menjadi tidak berguna*.*

**3.3.1. Analisis Kebutuhan**

Berdasarkan *activity diagram* pada sistem yang berjalan saat ini di Restaurant Mamma Masak, kita dapat menganalisa kebutuhan-kebutuhan sistem yang akan dikembangkan, diantaranya : *administrator* dapat mengolah semua menu yang ada pada aplikasi, *administrator* dapat mencetak laporan transaksi seluruh penjualan pada Restaurant Mamma Masak.

**Halaman *Administrator***

A.1. *Administrator* melakukan *login* dengan akun yang telah dibuat

A.2. *Administrator* dapat mengolah data *admin*

A.3. *Administrator* dapat mengolah data Bahan baku

A.4. *Administrator* dapat mengolah data Kategori

A.5. *Administrator* dapat mengolah data menu

A.6. *Administrator* dapat mengolah data pembelian

A.7. *Administrator* dapat melihat data penjualan

A.8. *Administrator* dapat mengolah data laporan

A.9. *Administrator* dapat melakukan *logout* dari system

**3.3.2. *Use Case Diagram***

1. ***Use Case Diagram Administrator***



**GambarIII.3.  
*Use Case Diagram Administrator***

**Tabel III.1. Deskripsi *Use Case Diagram Administrator***

|  |  |
| --- | --- |
| Use Case Narative **Sistem Informasi Penjualan Makanan** | |
| Tujuan | **Melakukan Login dan Masuk kedalam Sistem Informasi Penjualan** |
| **Deskripsi** | **Sistem ini memungkinkan aktor untuk mengakses Sistem Informasi Penjualan** |
| Skenario Utama | |
| Aktor | *Administrator* |
| Kondisi Awal | Aktor membuka Sistem Informasi Penjualan |
| Aksi Aktor | Reaksi Sistem |
| 1. Aktor Melakukan Login 2. Aktor Melakukan Pembelian 3. Aktor Melakukan input bahan 4. Aktor Memlilih Kategori 5. Aktor Melakukan Olah Menu 6. Aktor Melakukan Penjualan 7. Aktor Menerima Pembayaran Costumer 8. Aktor Milih Tombol Keluar | **Sistem akan menampilkan form login untuk mengisi username dan password.**  **Aktor Melakukan pembelian bahan baku pada Supplier**  Setelah proses Pembelian Bahan Baku Aktor Melakukan Inputan Bahan Baku  Aktor akan memilih mengeluarkan bahan baku untuk untuk minuman atau makanan  Setelah melakukan pengeluaran bahan baku untuk Kitchen proses bahan baku akan menjadi hasil proses atau makanan  Setelah Melakukan Proses Produksi Makanan Aktor Akan melakukan penjualan pada costumer yang memesan  Aktor Akan menerima Pembayaran costumer jika costumer selesai memakan makanan/minuman  **Sistem akan membatalkan proses dan keluar dari aktivitas.** |
| **Kondisi Akhir** | **Jika perintah sesuai maka sistem akan masuk kedalam sistem informasi dan aktor dapat melakukan aktivitas didalam sistem.** |

**3.3.3. *Activity Diagram***

**2. *Activity Diagram***

**

**Gambar III.4.**

***Activity Diagram Administrator* Sistem Informasi Penjualan**

* 1. **Desain**

**3.4.1. *Entity Relationship Diagram***

** Gambar III.5. *Entity Relationship Diagram***

**3.4.2. *Logical Record Structure***

****

**Gambar III.6. *Logical Record Structure***

**3.4.3. Spesifikasi File**

Menjelaskan tentang file atau tabel yang terbentuk dari transformasi ERD (dan atau file-file penunjang untuk *web*). File-file ini tersimpan pada *database***restaurant.sql** dengan parameter-parameter sbb :

1. Spesifikasi File Tabel User

Nama File : User

Akronim : Tabel User

Fungsi : Untuk menyimpan data akun *administrator*

Tipe File : File *Master*

Organisasi File : *Indexed Sequential*

Akses File : *Random*

Media : *Harddisk*

Panjang *record* : 37 *Bytes*

Kunci *Field* : *id\_user*

*Software* : *Mysql*

**Tabel III.2.**

**Spesifikasi File Tabel User**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *No.* | *Elemen Data* | *Nama Field* | *Tipe* | *Size* | *Ket* |
| *1.* | *ID User* | *id\_user* | *Char* | *10* | *Primary Key* |
| *2.* | *Nama User* | *nm\_user* | *Char* | *20* |  |
| *3.* | *Hak Akses* | *Hak\_akses* | *Char* | *7* |  |

1. Spesifikasi File Tabel Pembelian

Nama File : Pembelian

Akronim : Tabel pembelian

Fungsi : Untuk melakukan transaksi pembelian pada customer

Tipe File : File *Master*

Organisasi File : *Indexed Sequential*

Akses File : *Random*

Media : *Harddisk*

Panjang *record* : 14 *Bytes*

Kunci *Field* : no\_beli

*Software* : *Mysql*

**Tabel III.3.**

**Spesifikasi File Pembelian**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *No.* | *Elemen Data* | *Nama Field* | *Tipe* | *Size* | *Ket* |
| *1.* | *No Beli* | *no\_beli* | *Char* | *8* | *Primary key* |
| *2.* | *Tanggal Beli* | *Tgl\_beli* | *Date* |  |  |
| *3.* | *Total Beli* | *Total\_beli* | *Int* | *11* |  |
| *4.* | *Kode Supplier* | *kd\_supp* | *Varchar* | *5* |  |

1. Spesifikasi File Tabel Supplier

Nama File : supplier

Akronim : Tabel supplier

Fungsi : Untuk menyimpan data akun *Supplier*

Tipe File : File *Master*

Organisasi File : *Indexed Sequential*

Akses File : *Random*

Media : *Harddisk*

Panjang *record* : 100 *Bytes*

Kunci *Field* : kd\_supp

*Software* : *Mysql*

**Tabel III.4.**

**Spesifikasi File Tabel Supplier**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *No.* | *Elemen Data* | *Nama Field* | *Tipe* | *Size* | *Ket* |
| *1.* | *Kode Supplier* | *Kd\_supp* | *Char* | *5* | *Primary key* |
| *2.* | *Nama Supplier* | *Nm\_supp* | *Char* | *25* |  |
| *3.* | *Alamat* | *Alamat* | *Char* | *50* |  |
| *4.* | *Telepon* | *Telpon* | *Char* | *15* |  |

1. Spesifikasi File Tabel Detail Pembelian

Nama File : detail pembelian

Akronim : Tabel detail pembelian

Fungsi : Untuk melakukan transaksi pembelian lebih detail

Tipe File : File *Master*

Organisasi File : *Indexed Sequential*

Akses File : *Random*

Media : *Harddisk*

Panjang *record* : 46 *Bytes*

Kunci *Field* : no\_beli

*Software* : *Mysql*

**Tabel III.5.**

**Spesifikasi File Tabel Detail Pembelian**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *No.* | *Elemen Data* | *Nama Field* | *Tipe* | *Size* | *Ket* |
| *1.* | *Nomor Beli* | *No\_beli* | *Char* | *8* | *Primary key* |
| *2.* | *Kode Menu* | *Kdmenu* | *Char* | *5* |  |
| *3.* | *Harga* | *Harga* | *Int* | *11* |  |
| *4.* | *Qty beli* | *Qty\_beli* | *Int* | *11* |  |
| *5.* | *Subtotal Beli* | *Sub\_beli* | *Int* | *11* |  |

1. Spesifikasi File Tabel Bahan Baku

Nama File : bahan

Akronim : Tabel bahan

Fungsi : Untuk menyimpan data pembelian bahan baku dari *Supplier*

Tipe File : File *Master*

Organisasi File : *Indexed Sequential*

Akses File : *Random*

Media : *Harddisk*

Panjang *record* : 110 *Bytes*

Kunci *Field* : kodebahan

*Software* : *Mysql*

**Tabel III.6.**

**Spesifikasi File Tabel Bahan Baku**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *No.* | *Elemen Data* | *Nama Field* | *Tipe* | *Size* | *Ket* |
| *1.* | *Kode Bahan* | *Kodebahan* | *Varchar* | *20* | *Primary key* |
| *2.* | *Nama Bahan* | *Namabahan* | *Varchan* | *30* |  |
| *3.* | *Jumlah* | *Jumlah* | *Int* | *20* |  |
| *4.* | *Kode Supplier* | *Kd\_supp* | *Varchar* | *20* |  |
| *5.* | *ID User* | *Id\_user* | *Varchar* | *10* |  |

1. Spesifikasi File Tabel Menu

Nama File : menu

Akronim : Tabel menu

Fungsi : Untuk menyimpan data Menu

Tipe File : File *Master*

Organisasi File : *Indexed Sequential*

Akses File : *Random*

Media : *Harddisk*

Panjang *record* : 90 *Bytes*

Kunci *Field* : kdmenu

*Software* : *Mysql*

**Tabel III.7.**

**Spesifikasi File Tabel Menu**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *No.* | *Elemen Data* | *Nama Field* | *Tipe* | *Size* | *Ket* |
| *1.* | *Kode Menu* | *Kdmenu* | *Char* | *5* | *Primary key* |
| *2.* | *Nama Menu* | *Nmmenu* | *Char* | *25* |  |
| *3.* | *Kode Kategori* | *Kodekategori* | *Varchar* | *20* |  |
| *4.* | *Harga* | *Harga* | *Int* | *20* |  |
| *5.* | *Stok* | *Stok* | *Varchar* | *20* |  |

1. Spesifikasi File Tabel Kategori

Nama File : kategori

Akronim : Tabel kategori

Fungsi : Untuk menyimpan menu kategori

Tipe File : File *Master*

Organisasi File : *Indexed Sequential*

Akses File : *Random*

Media : *Harddisk*

Panjang *record* : 50 *Bytes*

Kunci *Field* : kodekategori

*Software* : *Mysql*

**Tabel III.8.**

**Spesifikasi File Tabel Kategori**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *No.* | *Elemen Data* | *Nama Field* | *Tipe* | *Size* | *Ket* |
| *1.* | *Kode kategori* | *Kodekategori* | *Varchar* | *20* | *Primary key* |
| *2.* | *Nama Kategori* | *Namakategori* | *Varchar* | *30* |  |

1. Spesifikasi File Tabel Penjualan

Nama File : penjualan

Akronim : Tabel penjuakan

Fungsi : Untuk melakukan transaksi penjualan

Tipe File : File *Master*

Organisasi File : *Indexed Sequential*

Akses File : *Random*

Media : *Harddisk*

Panjang *record* : 46 *Bytes*

Kunci *Field* : no\_jual

*Software* : *Mysql*

**Tabel III.9.**

**Spesifikasi File Tabel Penjualan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *No.* | *Elemen Data* | *Nama Field* | *Tipe* | *Size* | *Ket* |
| *1.* | *Nomor Jual* | *No\_jual* | *Char* | *8* | *Primary key* |
| *2.* | *Tanggal Jual* | *Tgl\_jual* | *Date* |  |  |
| *3.* | *Kode Menu* | *Kdmenu* | *Char* | *5* |  |
| *4.* | *Qty Jual* | *Qty\_jual* | *Int* | *11* |  |
| *5.* | *Sub jual* | *Sub\_jual* | *Int* | *11* |  |
| *6.* | *Total* | *Total* | *Int* | *11* |  |

* + 1. ***Sequence Diagram***



**Gambar III.7 *Sequence Diagram***

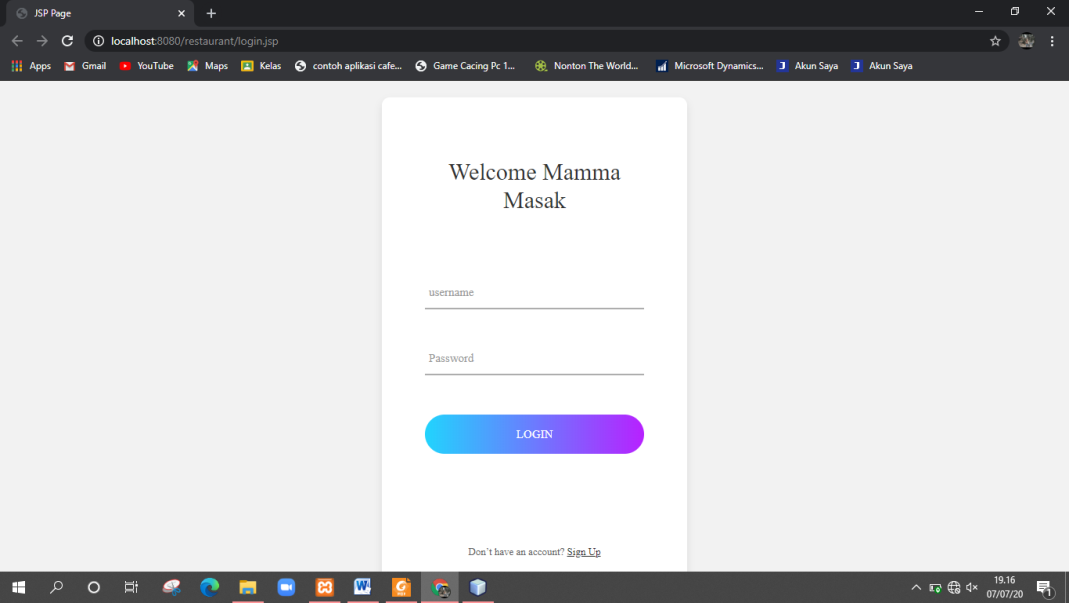
* + 1. ***Deployment Diagram***



**Gambar III.8 *Deployment Diagram***

* + 1. ***User Interface***

1. Berikut *User Interface* pada halaman kasir:
2. *User Interface* Halaman *Login*



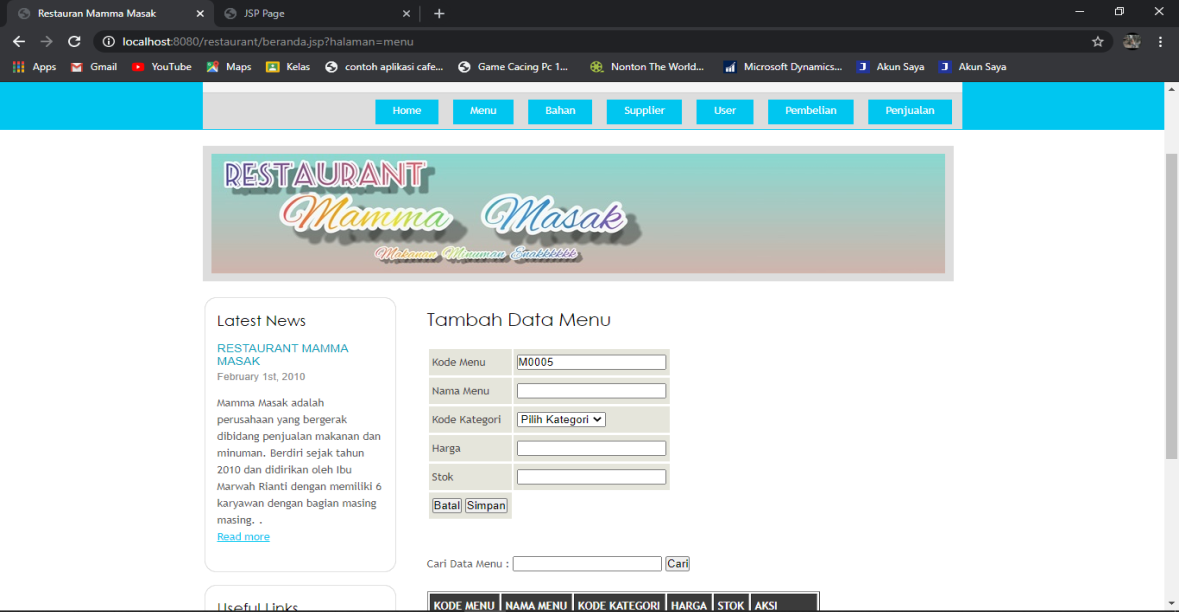
**Gambar III.9 *User Interface Login***

1. Admin Dapat Mengakses Beranda



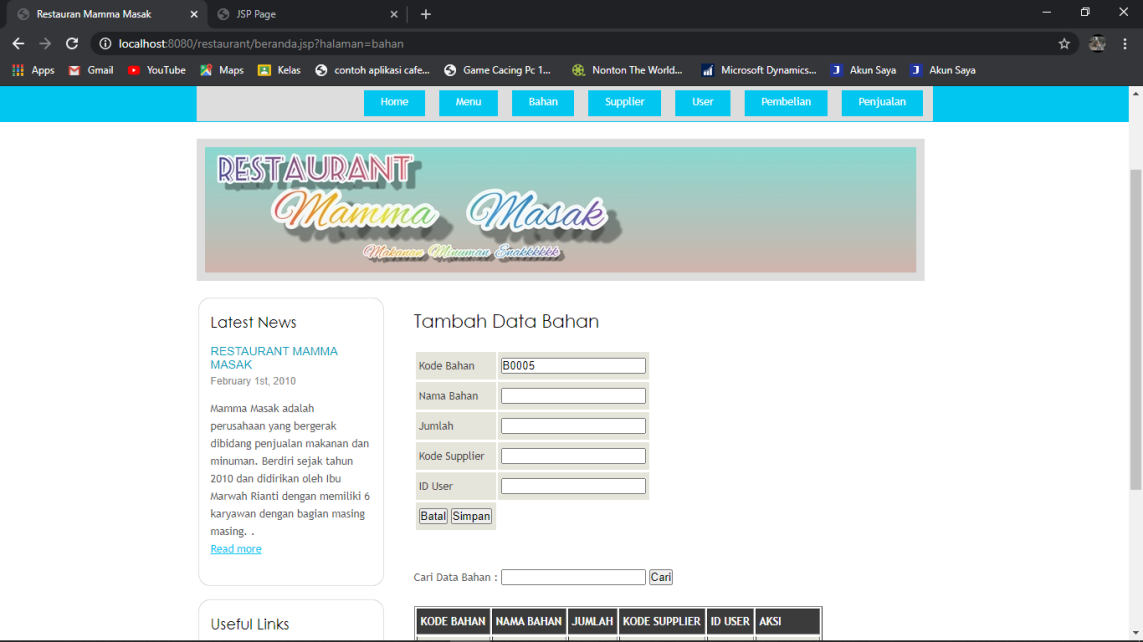
**Gambar III.10 *User Interface* Beranda**

1. Admin Dapat Mengakses Menu



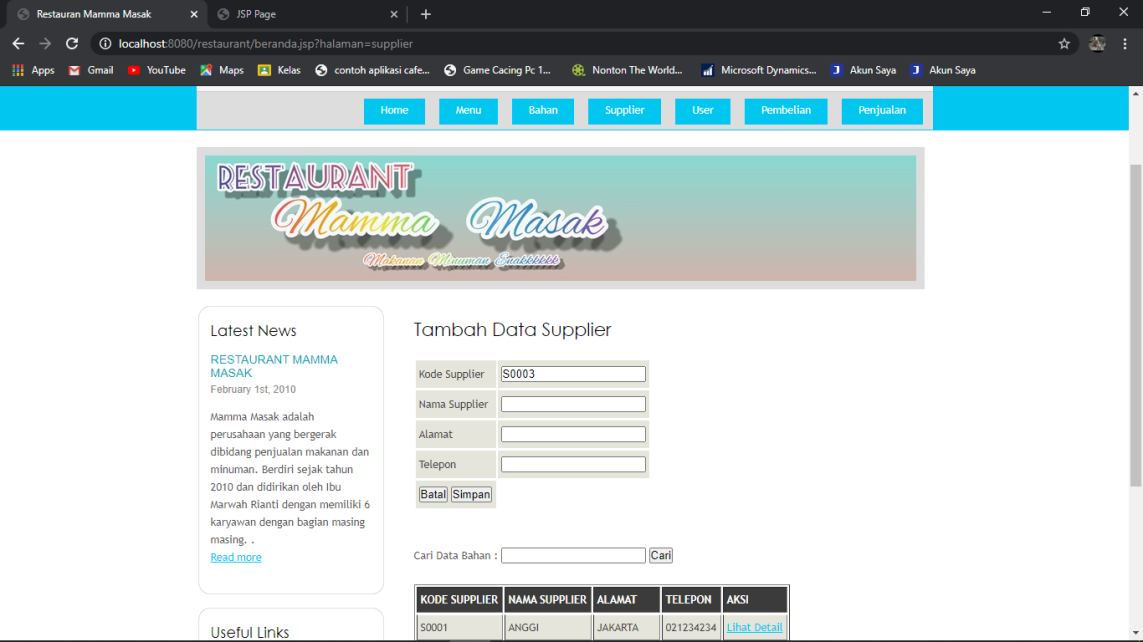
**Gambar III.11 *User Interface* Menu**

1. Admin Dapat Mengakses Bahan Baku



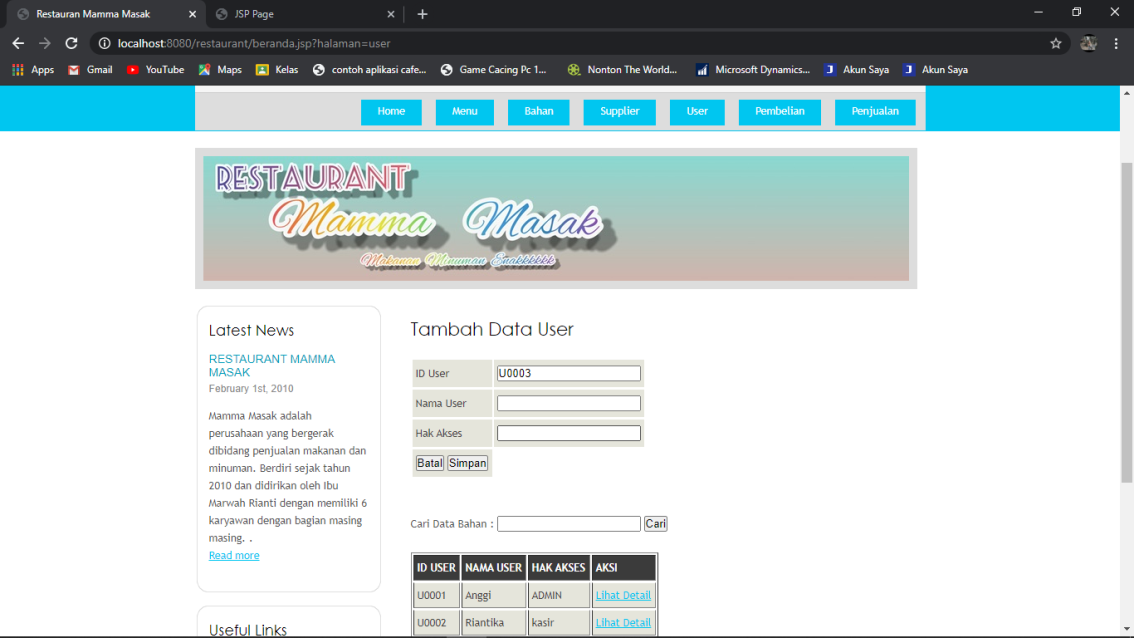
**Gambar III.12 *User Interface* Bahan**

1. Admin Dapat Mengakses Supplier



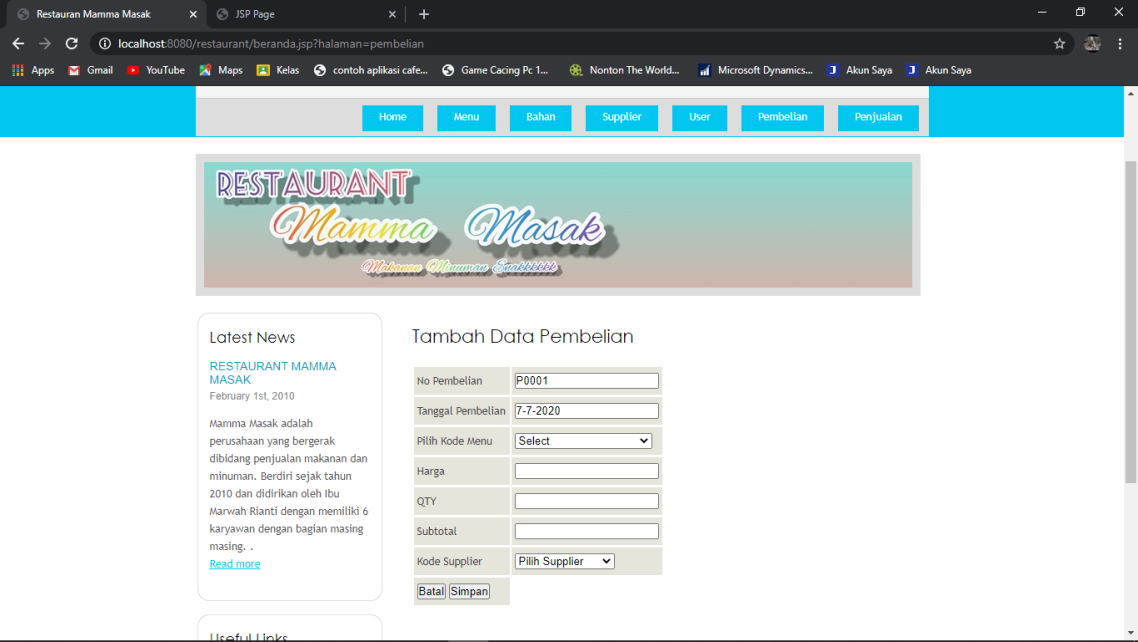
**Gambar III.13 *User Interface* Supplier**

1. Admin Dapat Mengakses User



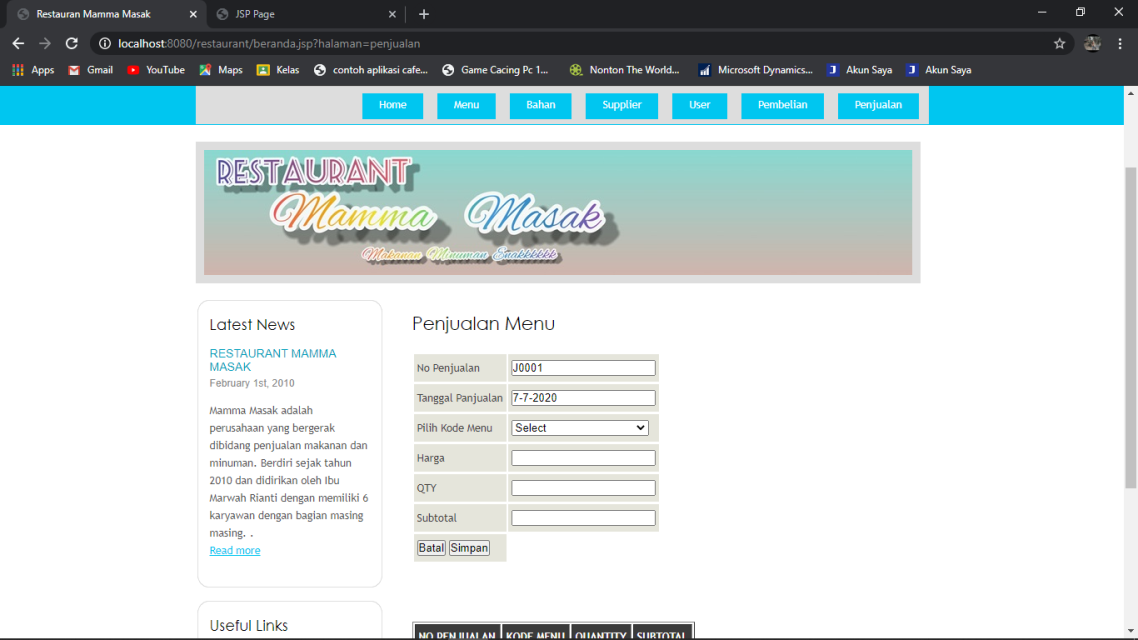
**Gambar III.14 *User Interface* User**

1. Admin Dapat Mengakses Pembelian



**Gambar III.15 *User Interface* Pembelian**

1. Admin Dapat Mengakses Penjualan



**Gambar III.16 *User Interface* Penjualan**

* 1. **Implementasi**

**3.5.1. Code Generation**

**1. Halaman** Home

**<%--**

**Document : home.jsp**

**Created on : May 22, 2020, 2:14:30 PM**

**Author : Anggi Riantika Dewi**

**--%>**

**<%@page contentType="text/html" pageEncoding="UTF-8"%>**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">**

**<title>Restauran Mamma Masak</title>**

**</head>**

**<body>**

**<h1>SELAMAT DATANG DI RESTAURAN MAMMA MASAK</h1>**

**<p> Disini anda bisa makan dan minum karena ini adalah restoran</p>**

**<p> disinipun kami menyediakan makanan ringan dan makanan berat, kami pula menyediakan minuman hangat dan segar</p>**

**<p> Tetapi kami tidak menyediakan jodoh untuk anda</p>**

**<h2>PILIHAN MAMA MASAK</h2>**

**<p> Silahkan dibaca </p>**

**<ul>**

**<li>Home Adalah tampilan utama</li>**

**<li> Menu Adalah tampilan menu makanan atau minuman yang akan dipilih</li>**

**<li> Bahan Adalah bahan baku makanan dan minuman yang akan dimasak</li>**

**<li> Supplier Adalah tampilan personal perusahaan untuk pembelian bahan baku</li>**

**<li> User Adalah kontak admin yang akan menginput </li>**

**</ul>**

**</body>**

**</html>**

1. Halaman Menu

*<%--*

*Document : menu*

*Created on : May 22, 2020, 2:41:02 PM*

*Author : Anggi Riantika Dewi*

*--%>*

*<%@page import="java.sql.SQLException"%>*

*<%@page import="control.koneksi"%>*

*<%@page import="java.sql.ResultSet"%>*

*<%@page contentType="text/html" pageEncoding="UTF-8"%>*

*<!DOCTYPE html>*

*<html>*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">*

*<title>Menu Mamma Masak</title>*

*</head>*

*<body>*

*<form method="POST" action="ServletMenu">*

*<h1>Tambah Data Menu</h1>*

*<table>*

*<tr><div>*

*<td><label for="inputEmail3">Kode Menu</label></td>*

*<div class="col-sm-10">*

*<td>*

*<%*

*try {*

*ResultSet rs = null;*

*koneksi kon = new koneksi();*

*rs = kon.stmt.executeQuery("select max(right(kdmenu,4)) as no from menu");*

*while (rs.next()) {*

*if (rs.first() == false) {*

*out.println("<input type='text' class='form-control' readonly value='MOOO1' name='kdmenu' >");*

*} else {*

*rs.last();*

*int autonokm = rs.getInt(1) + 1;*

*String nomorkm = String.valueOf(autonokm);*

*int noLong = nomorkm.length();*

*for (int a = 1; a < 5 - noLong; a++) {*

*nomorkm = "0" + nomorkm;*

*}*

*String nomerkm = "M" + nomorkm;*

*out.println("<input type='text' class='form\_control' readyonly value='" + nomerkm + "' name='kdmenu'>");*

*}*

*}*

*} catch (Exception e) {*

*out.println(e);*

*}*

*%>*

*</td>*

*</div>*

*</div>*

*</tr>*

*<tr>*

*<td>Nama Menu</td>*

*<td><input type="text" name="nmmenu"></td>*

*</tr>*

*<tr>*

*<td>Kode Kategori</td>*

*<td>*

*<select name="kodekategori">*

*<option value="">Pilih Kategori</option>*

*<option value="K0001">Makanan</option>*

*<option value="K0002">Minuman</option>*

*</select></td>*

*<tr>*

*<td>Harga</td>*

*<td><input type="text" name="harga"></td>*

*</tr>*

*<tr>*

*<td>Stok</td>*

*<td><input type="text" name="stok"></td>*

*</tr>*

*<tr>*

*<td>*

*<input type="reset" value="Batal">*

*<input type="submit" value="Simpan" name="aksi">*

*</td>*

*</tr>*

*</table>*

*</form>*

*<br>*

*<form action="" method="post">*

*Cari Data Menu : <input type="text" name="cari" value="${param.cari}">*

*<input type="submit" value="Cari" class="button"><br>*

*</form><br>*

*<table border="1">*

*<tr align="center">*

*<th>KODE MENU</th>*

*<th>NAMA MENU</th>*

*<th>KODE KATEGORI</th>*

*<th>HARGA</th>*

*<th>STOK</th>*

*<th>AKSI</th>*

*</tr>*

*<%*

*String kodemenu = null;*

*String namamenu = null;*

*String kodekategori = null;*

*String harga = null;*

*String stok = null;*

*ResultSet rs = null;*

*koneksi kon = new koneksi();*

*rs = kon.stmt.executeQuery("SELECT \* FROM menu ORDER BY kdmenu asc");*

*while (rs.next()) {*

*kodemenu = rs.getString("kdmenu");*

*namamenu = rs.getString(2);*

*kodekategori = rs.getString(3);*

*harga = rs.getString(4);*

*stok = rs.getString(5);*

*%>*

*<tr>*

*<td><% out.println(kodemenu);%></td>*

*<td><% out.println(namamenu);%></td>*

*<td><% out.println(kodekategori);%></td>*

*<td><% out.println(harga);%></td>*

*<td><% out.println(stok);%></td>*

*<td><a href="beranda.jsp?halaman=editmenu&kdmenu=<% out.println(kodemenu);%>">Lihat Detail</a></td>*

*</tr>*

*<% } %>*

*</table>*

*<% kon.close();%>*

*</body>*

*</html>*

1. Halaman Bahan

*<%--*

*Document : menu*

*Created on : May 22, 2020, 2:41:02 PM*

*Author : Anggi Riantika Dewi*

*--%>*

*<%@page import="java.sql.SQLException"%>*

*<%@page import="control.koneksi"%>*

*<%@page import="java.sql.ResultSet"%>*

*<%@page contentType="text/html" pageEncoding="UTF-8"%>*

*<!DOCTYPE html>*

*<html>*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">*

*<title>Menu Mamma Masak</title>*

*</head>*

*<body>*

*<form method="POST" action="ServletMenu">*

*<h1>Tambah Data Menu</h1>*

*<table>*

*<tr><div>*

*<td><label for="inputEmail3">Kode Menu</label></td>*

*<div class="col-sm-10">*

*<td>*

*<%*

*try {*

*ResultSet rs = null;*

*koneksi kon = new koneksi();*

*rs = kon.stmt.executeQuery("select max(right(kdmenu,4)) as no from menu");*

*while (rs.next()) {*

*if (rs.first() == false) {*

*out.println("<input type='text' class='form-control' readonly value='MOOO1' name='kdmenu' >");*

*} else {*

*rs.last();*

*int autonokm = rs.getInt(1) + 1;*

*String nomorkm = String.valueOf(autonokm);*

*int noLong = nomorkm.length();*

*for (int a = 1; a < 5 - noLong; a++) {*

*nomorkm = "0" + nomorkm;*

*}*

*String nomerkm = "M" + nomorkm;*

*out.println("<input type='text' class='form\_control' readyonly value='" + nomerkm + "' name='kdmenu'>");*

*}*

*}*

*} catch (Exception e) {*

*out.println(e);*

*}*

*%>*

*</td>*

*</div>*

*</div>*

*</tr>*

*<tr>*

*<td>Nama Menu</td>*

*<td><input type="text" name="nmmenu"></td>*

*</tr>*

*<tr>*

*<td>Kode Kategori</td>*

*<td>*

*<select name="kodekategori">*

*<option value="">Pilih Kategori</option>*

*<option value="K0001">Makanan</option>*

*<option value="K0002">Minuman</option>*

*</select></td>*

*<tr>*

*<td>Harga</td>*

*<td><input type="text" name="harga"></td>*

*</tr>*

*<tr>*

*<td>Stok</td>*

*<td><input type="text" name="stok"></td>*

*</tr>*

*<tr>*

*<td>*

*<input type="reset" value="Batal">*

*<input type="submit" value="Simpan" name="aksi">*

*</td>*

*</tr>*

*</table>*

*</form>*

*<br>*

*<form action="" method="post">*

*Cari Data Menu : <input type="text" name="cari" value="${param.cari}">*

*<input type="submit" value="Cari" class="button"><br>*

*</form><br>*

*<table border="1">*

*<tr align="center">*

*<th>KODE MENU</th>*

*<th>NAMA MENU</th>*

*<th>KODE KATEGORI</th>*

*<th>HARGA</th>*

*<th>STOK</th>*

*<th>AKSI</th>*

*</tr>*

*<%*

*String kodemenu = null;*

*String namamenu = null;*

*String kodekategori = null;*

*String harga = null;*

*String stok = null;*

*ResultSet rs = null;*

*koneksi kon = new koneksi();*

*rs = kon.stmt.executeQuery("SELECT \* FROM menu ORDER BY kdmenu asc");*

*while (rs.next()) {*

*kodemenu = rs.getString("kdmenu");*

*namamenu = rs.getString(2);*

*kodekategori = rs.getString(3);*

*harga = rs.getString(4);*

*stok = rs.getString(5);*

*%>*

*<tr>*

*<td><% out.println(kodemenu);%></td>*

*<td><% out.println(namamenu);%></td>*

*<td><% out.println(kodekategori);%></td>*

*<td><% out.println(harga);%></td>*

*<td><% out.println(stok);%></td>*

*<td><a href="beranda.jsp?halaman=editmenu&kdmenu=<% out.println(kodemenu);%>">Lihat Detail</a></td>*

*</tr>*

*<% } %>*

*</table>*

*<% kon.close();%>*

*</body>*

*</html>*

1. Halaman Supplier

*<%--*

*Document : supplier*

*Created on : May 30, 2020, 11:30:30 AM*

*Author : Anggi Riantika Dewi*

*--%>*

*<%@page import="control.koneksi"%>*

*<%@page import="java.sql.ResultSet"%>*

*<%@page contentType="text/html" pageEncoding="UTF-8"%>*

*<!DOCTYPE html>*

*<html>*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">*

*<title>Supplier Mamma Masak</title>*

*</head>*

*<body>*

*<form method="POST" action="SuppServlet">*

*<h1>Tambah Data Supplier</h1>*

*<table>*

*<tr><div>*

*</div>*

*<td><label for="inputEmail3">Kode Supplier</label></td>*

*<div class="col-sm-10">*

*<td>*

*<%*

*try {*

*ResultSet rs = null;*

*koneksi kon = new koneksi();*

*rs = kon.stmt.executeQuery("select max(right(kd\_supp,4)) as no from supplier");*

*while (rs.next()) {*

*if (rs.first() == false) {*

*out.println("<input type='text' class='form-control' readonly value='SOOO1' name='kodesupp' >");*

*} else {*

*rs.last();*

*int autonokm = rs.getInt(1) + 1;*

*String nomorkm = String.valueOf(autonokm);*

*int noLong = nomorkm.length();*

*for (int a = 1; a < 5 - noLong; a++) {*

*nomorkm = "0" + nomorkm;*

*}*

*String nomerkm = "S" + nomorkm;*

*out.println("<input type='text' class='form\_control' readyonly value='" + nomerkm + "' name='kodesupp'>");*

*}*

*}*

*} catch (Exception e) {*

*out.println(e);*

*}*

*%>*

*</td>*

*</div>*

*</div>*

*</tr>*

*<tr>*

*<td>Nama Supplier</td>*

*<td><input type="text" name="namasupp"></td>*

*</tr>*

*<tr>*

*<td>Alamat</td>*

*<td><input type="text" name="alamat"></td>*

*</tr>*

*<tr>*

*<td>Telepon</td>*

*<td><input type="text" name="telp"></td>*

*</tr>*

*<tr>*

*<td>*

*<input type="reset" value="Batal">*

*<input type="submit" value="Simpan" name="aksi">*

*</td>*

*</tr>*

*</table>*

*</form>*

*<br>*

*<form action="" method="post">*

*Cari Data Bahan : <input type="text" name="cari" value="${param.cari}">*

*<input type="submit" value="Cari" class="button"><br>*

*</form><br>*

*<table border="1">*

*<tr align="center">*

*<th>KODE SUPPLIER</th>*

*<th>NAMA SUPPLIER</th>*

*<th>ALAMAT</th>*

*<th>TELEPON</th>*

*<th>AKSI</th>*

*</tr>*

*<%*

*String kodesupp = null;*

*String namasupp = null;*

*String alamat = null;*

*String telp = null;*

*ResultSet rs = null;*

*koneksi kon = new koneksi();*

*rs = kon.stmt.executeQuery("SELECT \* FROM supplier ORDER BY kd\_supp asc");*

*while (rs.next()) {*

*kodesupp = rs.getString("kd\_supp");*

*namasupp = rs.getString(2);*

*alamat = rs.getString(3);*

*telp = rs.getString(4);*

*%>*

*<tr>*

*<td><% out.println(kodesupp);%></td>*

*<td><% out.println(namasupp);%></td>*

*<td><% out.println(alamat);%></td>*

*<td><% out.println(telp);%></td>*

*<td><a href="beranda.jsp?halaman=editsupp&kodesupp=<% out.println(kodesupp);%>">Lihat Detail</a></td>*

*</tr>*

*<% } %>*

*</table>*

*<% kon.close();%>*

*</body>*

*</html>*

1. Halaman Pembeian

*<%--*

*Document : menu*

*Created on : May 22, 2020, 2:41:02 PM*

*Author : Anggi Riantika Dewi*

*--%>*

*<%@page import="model.supplier"%>*

*<%@page import="model.pembelian"%>*

*<%@page import="control.koneksi"%>*

*<%@page import="java.sql.ResultSet"%>*

*<%@page import="model.menu" %>*

*<%@page contentType="text/html" pageEncoding="UTF-8"%>*

*<%*

*pembelian pemesanan=new pembelian();*

*menu menu=new menu();*

*supplier supplier=new supplier();*

*koneksi kon= new koneksi();*

*ResultSet rs=null;*

*ResultSet qrymenu=null;*

*ResultSet qrypembelian=null;*

*%>*

*<script type="text/javascript">*

*function showEmp(emp\_value)*

*{*

*if (document.getElementById("emp\_id").value != "-1")*

*{*

*xmlHttp = GetXmlHttpObject()*

*if (xmlHttp == null)*

*{*

*alert("Browser does not support HTTP Request")*

*return*

*}*

*var url = "getmenu.jsp"*

*url = url + "?emp\_id=" + emp\_value*

*xmlHttp.onreadystatechange = stateChanged*

*xmlHttp.open("GET", url, true)*

*xmlHttp.send(null)*

*} else*

*{*

*alert("Pilih Kode Menu");*

*}*

*}*

*function stateChanged()*

*{*

*document.getElementById("ename").value = "";*

*document.getElementById("emp\_id").value = "";*

*if (xmlHttp.readyState == 4 || xmlHttp.readyState == "complete")*

*{*

*var showdata = xmlHttp.responseText;*

*var strar = showdata.split(":");*

*if (strar.length == 1)*

*{*

*document.getElementById("emp\_id").focus();*

*alert("Pilih Kode Menu");*

*document.getElementById("ename").value = " ";*

*document.getElementById("emp\_id").value = " ";*

*} else if (strar.length > 1)*

*{*

*document.getElementById("ename").value = strar[1];*

*}*

*}*

*}*

*function GetXmlHttpObject()*

*{*

*var xmlHttp = null;*

*try*

*{*

*xmlHttp = new XMLHttpRequest();*

*} catch (e)*

*{*

*try*

*{*

*xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");*

*} catch (e)*

*{*

*xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");*

*}*

*}*

*return xmlHttp;*

*}*

*</script>*

*<!DOCTYPE html>*

*<html>*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">*

*<title>Pembelian Mamma Masak</title>*

*</head>*

*<body>*

*<form method="POST" action="ServletPembelian">*

*<h1>Tambah Data Pembelian</h1>*

*<table>*

*<tr><div>*

*<td><label for="inputEmail3">No Pembelian</label></td>*

*<div class="col-sm-10">*

*<td>*

*<%*

*try {*

*rs = kon.stmt.executeQuery("select max(right(no\_beli,4)) as no from pembelian");*

*while (rs.next()) {*

*if (rs.first() == false) {*

*out.println("<input type='text' class='form-control' readonly value='POOO1' name='no\_beli' >");*

*} else {*

*rs.last();*

*int autonokm = rs.getInt(1) + 1;*

*String nomorkm = String.valueOf(autonokm);*

*int noLong = nomorkm.length();*

*for (int a = 1; a < 5 - noLong; a++) {*

*nomorkm = "0" + nomorkm;*

*}*

*String nomerkm = "P" + nomorkm;*

*out.println("<input type='text' class='form\_control' readyonly value='" + nomerkm + "' name='no\_beli'>");*

*}*

*}*

*} catch (Exception e) {*

*out.println(e);*

*}*

*%>*

*</td>*

*</div>*

*</div>*

*</tr>*

*<%*

*java.util.Date waktu = new java.util.Date();*

*int tanggal = waktu.getDate();*

*int tahun = waktu.getYear() + 1900;*

*int bulan = waktu.getMonth() + 1;*

*String tgl = tanggal + "-" + bulan + "-" + tahun;*

*%>*

*<tr>*

*<td>Tanggal Pembelian</td>*

*<td><input type="text" name="tgl\_beli" value="<%=tgl%>"</td>*

*</tr>*

*<tr>*

*<td>Pilih Kode Menu</td>*

*<td><select name="semp\_id" onchange="showEmp(this.value);">*

*<option value="-1">Select</option>*

*<%*

*rs = kon.stmt.executeQuery("SELECT \* from menu");*

*while (rs.next()) {*

*menu.setKodemenu(rs.getString("kdmenu"));*

*menu.setNamamenu(rs.getString("nmmenu"));*

*%>*

*<option value="<%=menu.getKodemenu()%>"><%=menu.getKodemenu()%> || <%=menu.getNamamenu()%></option>*

*<% } %>*

*</select></td>*

*<input type="hidden" name="emp\_id" id="emp\_id" value="">*

*</tr>*

*<tr>*

*<td>Harga</td>*

*<td><input type="text" readonly name="emp\_name" id="ename" value="" onkeyup="sum();"</td>*

*</tr>*

*<tr>*

*<td>QTY</td>*

*<td><input type="text" name="qty\_beli" id="jml" onkeyup="sum();"></td>*

*</tr>*

*<tr>*

*<td>Subtotal</td>*

*<td><input type="text" name="sub\_beli" id="subtotal" readonly="readonly"></td>*

*</tr>*

*<tr>*

*<td>Kode Supplier</td>*

*<td><select name="kd\_supp">*

*<option value="-1">Pilih Supplier</option>*

*<%*

*rs=kon.stmt.executeQuery("select \* from supplier");*

*while(rs.next()){*

*supplier.setKodesupp(rs.getString("kd\_supp"));*

*supplier.setNamasupp(rs.getString("nm\_supp"));*

*%>*

*<option value="<%=supplier.getKodesupp()%>"><%=supplier.getKodesupp()%> || <%=supplier.getNamasupp()%></option>*

*<% } %>*

*</select></td>*

*</tr>*

*<script type="text/javascript">*

*function sum(){*

*var txtFirstNumberValue = document.getElementById('ename').value;*

*var txtSecondNumberValue = document.getElementById('jml').value;*

*var result = parseFloat(txtFirstNumberValue) \* parseFloat(txtSecondNumberValue);*

*if(!isNaN(result)){*

*document.getElementById('subtotal').value=result;*

*}*

*}*

*</script>*

*<tr>*

*<td>*

*<input type="reset" value="Batal">*

*<input type="submit" value="Simpan" name="aksi">*

*</td>*

*</tr>*

*</table>*

*</form>*

*<br>*

*<form method="POST" action="ServletPembelians">*

*<table border="1">*

*<tr align="center">*

*<th>NO PEMBELIAN</th>*

*<th>KODE MENU</th>*

*<th>QUANTITY</th>*

*<th>SUBTOTAL</th>*

*</tr>*

*<%*

*String nobeli = null;*

*String kodemenu = null;*

*String qty = null;*

*String subtotal = null;*

*rs = kon.stmt.executeQuery("SELECT \* FROM sementara ORDER BY no\_beli asc");*

*while (rs.next()) {*

*nobeli = rs.getString("no\_beli");*

*kodemenu = rs.getString(2);*

*qty = rs.getString(3);*

*subtotal = rs.getString(4);*

*%>*

*<tr>*

*<td><% out.println(nobeli);%></td>*

*<td><% out.println(kodemenu);%></td>*

*<td><% out.println(qty);%></td>*

*<td><% out.println(subtotal);%></td>*

*</tr>*

*<% } %>*

*<%!*

*int subqty = 0, sub = 0;*

*%>*

*<%*

*rs = kon.stmt.executeQuery("SELECT \* from sementara");*

*while (rs.next()) {*

*subqty = subqty + (Integer.parseInt(rs.getString(3)));*

*sub = sub + (Integer.parseInt(rs.getString(4)));*

*}*

*%>*

*<tr>*

*<td colspan="2"><b align="center">TOTAL</b></td>*

*<td><b><% out.println(subqty);%></b></td>*

*<td><b><% out.println(sub);%></b></td>*

*</tr>*

*</table>*

*<% kon.close();%>*

*<input type="submit" value="Proses" name="aksi">*

*</form>*

*</body>*

*</html>*

1. Halaman Penjualan

*<%--*

*Document : Pemesanan*

*Created on : Jun 27, 2020, 4:45:52 PM*

*Author : Anggi Riantika Dewi*

*--%>*

*<%@page import="model.supplier"%>*

*<%@page import="model.pembelian"%>*

*<%@page import="control.koneksi"%>*

*<%@page import="java.sql.ResultSet"%>*

*<%@page import="model.menu" %>*

*<%@page import="model.penjualan" %>*

*<%*

*penjualan pemesanan=new penjualan();*

*menu menu=new menu();*

*supplier supplier=new supplier();*

*koneksi kon= new koneksi();*

*ResultSet rs=null;*

*%>*

*<script type="text/javascript">*

*function showEmp(emp\_value)*

*{*

*if (document.getElementById("emp\_id").value != "-1")*

*{*

*xmlHttp = GetXmlHttpObject()*

*if (xmlHttp == null)*

*{*

*alert("Browser does not support HTTP Request")*

*return*

*}*

*var url = "getmenu.jsp"*

*url = url + "?emp\_id=" + emp\_value*

*xmlHttp.onreadystatechange = stateChanged*

*xmlHttp.open("GET", url, true)*

*xmlHttp.send(null)*

*} else*

*{*

*alert("Pilih Kode Menu");*

*}*

*}*

*function stateChanged()*

*{*

*document.getElementById("ename").value = "";*

*document.getElementById("emp\_id").value = "";*

*if (xmlHttp.readyState == 4 || xmlHttp.readyState == "complete")*

*{*

*var showdata = xmlHttp.responseText;*

*var strar = showdata.split(":");*

*if (strar.length == 1)*

*{*

*document.getElementById("emp\_id").focus();*

*alert("Pilih Kode Menu");*

*document.getElementById("ename").value = " ";*

*document.getElementById("emp\_id").value = " ";*

*} else if (strar.length > 1)*

*{*

*document.getElementById("ename").value = strar[1];*

*}*

*}*

*}*

*function GetXmlHttpObject()*

*{*

*var xmlHttp = null;*

*try*

*{*

*xmlHttp = new XMLHttpRequest();*

*} catch (e)*

*{*

*try*

*{*

*xmlHttp = new ActiveXObject("Msxml2.XMLHTTP");*

*} catch (e)*

*{*

*xmlHttp = new ActiveXObject("Microsoft.XMLHTTP");*

*}*

*}*

*return xmlHttp;*

*}*

*</script>*

*<%@page contentType="text/html" pageEncoding="UTF-8"%>*

*<!DOCTYPE html>*

*<html>*

*<head>*

*<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">*

*<title>Penjualan Mamma Masak</title>*

*</head>*

*<body>*

*<form method="POST" action="PenjualanServlet">*

*<h1>Penjualan Menu</h1>*

*<table>*

*<tr><div>*

*<td><label for="inputEmail3">No Penjualan</label></td>*

*<div class="col-sm-10">*

*<td>*

*<%*

*try {*

*rs = kon.stmt.executeQuery("select max(right(no\_jual,4)) as no from penjualan");*

*while (rs.next()) {*

*if (rs.first() == false) {*

*out.println("<input type='text' class='form-control' readonly value='JOOO1' name='no\_jual' >");*

*} else {*

*rs.last();*

*int autonokm = rs.getInt(1) + 1;*

*String nomorkm = String.valueOf(autonokm);*

*int noLong = nomorkm.length();*

*for (int a = 1; a < 5 - noLong; a++) {*

*nomorkm = "0" + nomorkm;*

*}*

*String nomerkm = "J" + nomorkm;*

*out.println("<input type='text' class='form\_control' readyonly value='" + nomerkm + "' name='no\_jual'>");*

*}*

*}*

*} catch (Exception e) {*

*out.println(e);*

*}*

*%>*

*</td>*

*</div>*

*</div>*

*</tr>*

*<%*

*java.util.Date waktu = new java.util.Date();*

*int tanggal = waktu.getDate();*

*int tahun = waktu.getYear() + 1900;*

*int bulan = waktu.getMonth() + 1;*

*String tgl = tanggal + "-" + bulan + "-" + tahun;*

*%>*

*<tr>*

*<td>Tanggal Panjualan</td>*

*<td><input type="text" name="tgl\_jual" value="<%=tgl%>"</td>*

*</tr>*

*<tr>*

*<td>Pilih Kode Menu</td>*

*<td><select name="semp\_id" onchange="showEmp(this.value);">*

*<option value="-1">Select</option>*

*<%*

*rs = kon.stmt.executeQuery("SELECT \* from menu");*

*while (rs.next()) {*

*menu.setKodemenu(rs.getString("kdmenu"));*

*menu.setNamamenu(rs.getString("nmmenu"));*

*%>*

*<option value="<%=menu.getKodemenu()%>"><%=menu.getKodemenu()%> || <%=menu.getNamamenu()%></option>*

*<% } %>*

*</select></td>*

*<input type="hidden" name="emp\_id" id="emp\_id" value="">*

*</tr>*

*<tr>*

*<td>Harga</td>*

*<td><input type="text" readonly name="emp\_name" id="ename" value="" onkeyup="sum();"</td>*

*</tr>*

*<tr>*

*<td>QTY</td>*

*<td><input type="text" name="qty\_jual" id="jml" onkeyup="sum();"></td>*

*</tr>*

*<tr>*

*<td>Subtotal</td>*

*<td><input type="text" name="sub\_jual" id="subtotal" readonly="readonly"></td>*

*</tr>*

*<script type="text/javascript">*

*function sum(){*

*var txtFirstNumberValue = document.getElementById('ename').value;*

*var txtSecondNumberValue = document.getElementById('jml').value;*

*var result = parseFloat(txtFirstNumberValue) \* parseFloat(txtSecondNumberValue);*

*if(!isNaN(result)){*

*document.getElementById('subtotal').value=result;*

*}*

*}*

*</script>*

*<tr>*

*<td>*

*<input type="reset" value="Batal">*

*<input type="submit" value="Simpan" name="aksi">*

*</td>*

*</tr>*

*</table>*

*</form>*

*<br>*

*<br>*

*<table border="1">*

*<tr align="center">*

*<th>NO PENJUALAN</th>*

*<th>KODE MENU</th>*

*<th>QUANTITY</th>*

*<th>SUBTOTAL</th>*

*</tr>*

*<%*

*String nojual = null;*

*String kodemenu = null;*

*String qty = null;*

*String subtotal = null;*

*rs = kon.stmt.executeQuery("SELECT \* FROM sementara\_jual ORDER BY no\_jual asc");*

*while (rs.next()) {*

*nojual = rs.getString("no\_jual");*

*kodemenu = rs.getString(2);*

*qty = rs.getString(3);*

*subtotal = rs.getString(4);*

*%>*

*<tr>*

*<td><% out.println(nojual);%></td>*

*<td><% out.println(kodemenu);%></td>*

*<td><% out.println(qty);%></td>*

*<td><% out.println(subtotal);%></td>*

*</tr>*

*<% } %>*

*<%*

*int subs = 0, sub = 0;*

*%>*

*<%*

*rs = kon.stmt.executeQuery("SELECT \* from sementara\_jual");*

*while (rs.next()) {*

*subs = subs + (Integer.parseInt(rs.getString(3)));*

*sub = sub + (Integer.parseInt(rs.getString(4)));*

*}*

*%>*

*<tr>*

*<td colspan="2"><b align="center">TOTAL</b></td>*

*<td><b><% out.println(subs);%></b></td>*

*<td><b><% out.println(sub);%></b></td>*

*</tr>*

*</table>*

*<% kon.close();%>*

*</body>*

*</html>*

**3.5.2. Blackbox Testing**

Pengujian dilakukan terhadap program yang dibuat menggunakan *blackboxtesting* yang fokus terhadap proses masukan dan keluaran program.

1. Pengujian Tahap *Form Login Admin*

**Tabel III.10**

**Hasil Pengujian *Black Box Testing Form Login Admin***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Skenario Penguji | Test Case | Hasil yang diharapkan | Hasil Pengujian | Kesimpulan |
| 1. | *Username* dan *Password* tidak diisi kemudian klik tombol *Login* | *Username :* (Kosong)  *Password :* (Kosong) | **Sistem akan menolak akses dan menampilkan "masukan username!"** | **Sesuai harapan** | Valid |
| 2. | **Username diisi dan passwordtidak diisi kemudian klik tombol login** | *Username :* Admin  *Password :* (Kosong) | **Sistem akan menolak akses dan menampilkan "masukan password!"** | **Sesuai harapan** | Valid |
| 3. | **Username tidak diisi dan password diisi kemudian klik tombol login** | **Username: (kosong) Password: admin** | **Username: (kosong) Password: admin** | **Sesuai harapan** | Valid |
| 4. | **Mengetikan** **salah satu** **kondisi salah** **pada** username**atau** password**kemudian klik** **tombol** login | **Mengetikan** **salah satu** **kondisi salah** **pada** username**atau** password**kemudian klik** **tombol** login | **Sistem akan** **menolak akses** user **dan** **menampilkan** **"Maaf,** username/ password**salah!"** | **Sesuai harapan** | Valid |
| 5. | **Mengetikan usernameatau password yang benar kemudian klik tombol login** | **Username: admin (benar) Password: admin** | **Sistem akan menerima akses login** | **Sesuai harapan** | Valid |

1. Pengujian Terhadap *Form* Daftar  *Customer*

**Tabel III.111**

**Hasil Pengujian *Black Box Testing Form* Daftar *Register Customer***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Skenario Penguji | Test Case | Hasil yang diharapkan | Hasil Pengujian | Kesimpulan |
| 1. | **Salah satu field kosong kemudian klik tombol simpan** | **Kode Supplier : (kosong) Nama Supplier : Dian Intan Alamat: Jl. Raya Jakarta Barat Telpon : 085716088528** | Sistem Akan menampilkan “Masukan Kode Supplier!” | **Sesuai harapan** | Valid |
| 2. | **Semua field diisi kemudian klik tombol register** | **Kode Supplier : Anggi Nama Supplier : Dian Intan Alamat: Jl. Raya Jakarta Barat Telpon : 085716088528** | Sistem Akan Menampilkan “ Data Berhasil Disimpan | **Sesuai harapan** | Valid |

1. Pengujian Terhadap *Form* Transaksi

**Tabel III.12**

**Hasil Pengujian *Black Box Testing Form* Transaksi**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Skenario Penguji | Test Case | Hasil yang diharapkan | Hasil Pengujian | Kesimpulan |
| 1. | Salah satu field kosong kemudian klik simpan | No Pembelian : P001  Tanggal Pembelian : 07/07/2020  Pilih Kode Menu: M0001 || Mie Ayam  Harga : 15.000  QTY : (KOSONG)  Subtotal : (Kosong)  Kode Supplier : S001 | Sistem Akan menampilkan “Harap Masukan QTY!” | **Sesuai harapan** | Valid |
| 2. | Salah satu field kosong kemudian klik simpan | No Pembelian : P001  Tanggal Pembelian : 07/07/2020  Pilih Kode Menu: (Kosong)  QTY : (KOSONG)  Subtotal : (Kosong)  Kode Supplier : S001 | Sistem Akan menampilkan “Harap Masukan Menu!” | **Sesuai harapan** | Valid |
| 3. | Field sudah diisi semua dengan benar | No Pembelian : P001  Tanggal Pembelian : 07/07/2020  Pilih Kode Menu: M0001 || Mie Ayam  Harga : 15.000  QTY : 1  Subtotal : 15.000  Kode Supplier : S001 | Sistem Akan menampilkan “Data Berhasil Disimpan!” | **Sesuai harapan** | Valid |

* + 1. **Spesifikasi Hardware & Software**

Berikut ini adalah spesifikasi sistem komputer yang terdiri dari perangkat  
keras dan perangkat lunak minimum yang dibutuhkan untuk mengimplementasikan  
sistem informasi penjualan Restaurant Mamma Masak

1. Spesifikasi Perangkat Keras
2. *Server*
3. CPU
4. *Processor* Pentium® Core 2 Duo.
5. RAM DDR2 4 GB.
6. *Hard Disk* 500 GB.
7. *Mouse*.
8. *Keyboard.*
9. *Monitor* dengan resolusi layar minimum 1024x768.
10. Koneksi *internet* dengan kecepatan 2 Mbps.
11. *Client*
12. CPU

a.) *Processor* Pentium® 4.

b.) RAM DDR2 1GB.

c.) *Hard Disk* 80 GB.

*2.) Mouse*

*3.) Keyboard*

4.) *Monitor* dengan resolusi layar minimum 1024x768

5.) Koneksi *internet* dengan kecepatan 56 kbps.

1. Spesifikasi Perangkat Lunak
2. *Server*
3. Sistem operasi yang umum digunakan seperti: *Microsoft  
   Windows* atau *Linux* (*Ubuntu*, *Fedora*, dan lain-lain ).
4. Aplikasi *bundle web server* seperti : *Xampp*, *WampServer*, *php2triad*yang terdiri dari beberapa komponen, diantaranya:
5. Aplikasi *Apache Server* v2
6. Aplikasi *PHP Server* v5
7. Aplikasi *MySQL Server* v5
8. Aplikasi *phpMyAdmin* v3

**BAB IV**

**PENUTUP**

**4.1 Kesimpulan**

Mamma Masak mengalami kendala mengenai pembelian produk dari pemasok yaitu nota pembelian masih menggunakan system manual dan dan nota-nota sering bercecran dan tidak teratur. Akibat dari kendala tersebut akhirnya Mamma Masak memutuskan untuk menggunakan aplikasi yang dapat dicacat oleh 1 orang.

Adapun kelemahannya adalah sebagai berikut:

1. Sering terjadi selisih dalam masalah pencatatan persediaan barang di gudang dikarenakan pencatatan barang masuk (BM) dan barang keluar (BK) kurang tertib.

2. Dalam kegiatan sehari-hari sistem yang digunakan sering mengalami masalah dan proses pengolahan data yang kurang cepat dan tepat, dikarenakan system yang berjalan masih manual.

**4.2 Saran**

Sebagus apapun susunan yang kami buat tanpa dukungan oleh pihak

manajemen dan operasional, maka akan menimbulkan masalah baru. Oleh karena itu diperlukan kesatuan dan kerjasama antar elemen kedisiplinan kerja.

Untuk mengatasi masalah tersebut disarankan :

1. Cara efektif yang dapat diambil untuk mengatasi semua permasalahan tersebut hanya dengan cara lebih teliti saat pencatatan data transaksi dan saat pencarian data penjualannya agar tidak terjadi kesalahan data stok barang.

2. Untuk karyawan/karyawati Mamma Masak perlunya komunikasi yang baik atau ditingkatkan kekompakannya agar tidak terjadi keteledoran dalam menginput atau menginput pencatatan bukti transaksi data penjualan, sehingga tidak terjadi selisih dalam pencatatan persediaan barang.

3. Diperlukan adanya backup data agar data dapat tersimpan dengan aman.