

APLIKASI PENJUALAN SPARE PART MOBIL MENGUNAKAN CODE IGNITER UNTUK KEAKURATAN PELAPORAN DATA

(Studi Kasus Pada Bengkel Moto Link Kecamatan Baleendah)

SKRIPSI

Karya Tulis sebagai syarat memperoleh
Gelar Sarjana Komputer dari Fakultas Teknologi Informasi
Universitas Bale Bandung

Disusun oleh :

CHANDRA RIZKI AZHARI

NPM. C1A160016



PROGRAM STRATA 1
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG
BANDUNG
2020

LEMBAR PERSETUJUAN PEMBIMBING

APLIKASI PENJUALAN SPARE PART MOBIL MENGGUNAKAN CODE IGNITER UNTUK KEAKURATAN PELAPORAN DATA

(Studi Kasus Pada Bengkel Moto Link Kecamatan Baleendah)

Disusun oleh :

CHANDRA RIZKI AZHARI

NPM. C1A160016

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2020

Disetujui oleh:

Pembimbing 1

Pembimbing 2

Zen Munawar, S.Kom., M.Kom.,
NIDN: 0422037002

Denny Rusdianto, S.T., M.Kom.,
NIDN. 04104808094

LEMBAR PERSETUJUAN PENGUJI

APLIKASI PENJUALAN SPARE PART MOBIL MENGGUNAKAN CODE IGNITER UNTUK KEAKURATAN PELAPORAN DATA

(Studi Kasus Bengkel Moto link Kecamatan Baleendah)

Disusun oleh :

CHANDRA RIZKI AZHARI

NPM. C1A160016

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

SARJANA KOMPUTER

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2020

Disetujui oleh:

Penguji 1

Penguji 2

Iim Abdurrohman, S.T., M.T.,
NIDN. 0413107002

Rosmalina, S.T., M.Kom.,
NIDN. 425038203

LEMBAR PERSETUJUAN PROGRAM STUDI

**APLIKASI PENJUALAN SPARE PART MOBIL MENGGUNAKAN
CODE IGNITER UNTUK KEAKURATAN PELAPORAN DATA**

(Studi Kasus Bengkel Moto link Kecamatan Baleendah)

Disusun oleh :

CHANDRA RIZKI AZHARI

NPM. C1A160016

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

SARJANA KOMPUTER

Pada

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS BALE BANDUNG

Baleendah, Juli 2020

Disetujui oleh:

Mengetahui,
Dekan,

Mengesahkan,
Ketua Program Studi

Yudi Herdiana, S.T., M.T.
NIK. 04104808008

Yaya Suharya, S.Kom., M.T.
NIK. 01043170007

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan dibawah ini:

Nama : CHANDRA RIZKI AZHARI

NIM : C1A160016

Judul Skripsi : **APLIKASI PENJUALAN SPARE PART MOBIL**

MENGGUNAKAN CODE IGNITER UNTUK

KEAKURATAN PELAPORAN DATA

Menyatakan dengan sebenarnya bahwa penulisan skripsi ini berdasarkan hasil penelitian, pemikiran dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan *programming* yang tercantum sebagai bagian dari skripsi ini. Jika terdapat karya orang lain, saya mencantumkan sumber yang jelas.

Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidak benaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sanksi lain sesuai dengan peraturan yang berlaku di FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS BALE BANDUNG.

Demikian surat pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

Baleendah, Juli 2020
Yang membuat pernyataan,

CHANDRA RIZKI AZHARI

NIM. C1A160016

ABSTRACT

Moto Link is a business that provides after-sales repair to consumers located on Jln. Raya Banjaran, No.183 Kec. Baleendah Kab. Bandung 40375. Moto Link Workshop was opened and inaugurated on June 20, 2016. Moto Link Workshop provides repair and sales of car spare parts. Manually quotation is very difficult for the workshop manager when they have to provide spare parts sales report to the workshop owner. This research aims to build a spare parts sales application for a web-based car repair shop. This application is built with the PHP programming language and MySQL database that has been supported by responsive web design. This application is an idea to improve the quality of the Moto Link workshop in terms of archiving documents. Through this application, the workshop can make better spare part sales management. The results of this study are the development of car spare parts sales applications for the needs of Moto Link Workshop that can assist in data reporting.

Keywords: Application, Service, Sparepart, Web.

ABSTRAK

Moto Link adalah salah satu usaha yang menyediakan perbaikan purna jual kepada konsumen yang terletak di Jln. Raya Banjaran, No.183 Kec. Baleendah Kab. Bandung 40375 . Bengkel Moto Link dibuka dan diresmikan pada tanggal 20 Juni 2016. Bengkel Moto Link menyediakan jasa perbaikan serta penjualan spare part mobil . Pencatatan secara manual sangat menyulitkan pihak karyawan bengkel ketika harus memberikan laporan penjualan sparepart kepada pemilik bengkel. Penelitian ini bertujuan membangun sebuah aplikasi penjualan sparepart untuk bengkel mobil berbasis web. Aplikasi ini dibangun dengan bahasa pemrograman PHP dan database MySQL yang telah didukung web responsive design. Aplikasi ini merupakan gagasan untuk meningkatkan mutu bengkel Moto Link dalam hal pengarsipan dokumen. Melalui aplikasi ini, pihak bengkel dapat melakukan manajemen penjualan sparepart menjadi lebih baik lagi. Hasil penelitian ini adalah terbangunnya aplikasi penjualan spare part mobil untuk kebutuhan Bengkel Moto Link yang dapat membantu dalam pelaporan data.

Kata Kunci: *Aplikasi, Service, Sparepart, Web.*

KATA PENGANTAR

Segala puji dan Syukur kami panjatkan ke Hadirat Allah S.W.T, atas segala Rahmat dan Karunia-Nya penulis dapat menyelesaikan Skripsi ini dengan baik dan tepat pada waktunya yang berjudul “APLIKASI PENJUALAN SPARE PART MOBIL MENGGUNAKAN CODEIGNITER UNTUK KEAKURATAN PELAPORAN DATA” dengan tepat pada waktunya.

Tujuan penulisan ini disusun guna untuk melengkapi salah satu syarat dalam menyelesaikan jenjang Strata satu jurusan Teknik Informatika. Dalam kesempatan ini juga penulis ingin menyampaikan ucapan terima kasih kepada :

1. Bapak Yudi Herdiana, S.T., M.T., selaku Dekan Fakultas Teknologi Informasi Universitas Bale Bandung.
2. Bapak Yaya Suharya, S.Kom., M.T., selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.
3. Bapak Zen Munawar, S.Kom., M.Kom., selaku pembimbing 1.
4. Bapak Denny Rusdianto, S.T., M.Kom., selaku pembimbing 2.
5. Semua pihak yang telah membantu dan memberikan dukungan kepada penulis untuk menyelesaikan Laporan Skripsi ini.

Penulis menyadari sepenuhnya bahwa laporan ini masih jauh dari kesempurnaan, oleh karena itu kritik dan saran dari semua pihak yang bersifat membangun selalu penulis harapkan demi kesempurnaan laporan ini.

Bandung, Juli 2020

Penulis

Chandra Rizki Azhari

DAFTAR ISI

ABSTRACT.....	i
ABSTRAK.....	ii
KATA PENGANTAR	iii
DAFTAR ISI.....	iv
DAFTAR TABEL.....	v
DAFTAR GAMBAR	vii
BAB I Pendahuluan	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	2
1.5 Metodologi Penelitian	2
1.6 Sistematika Penulisan.....	5
BAB II Tinjauan Pustaka	6
2.1 Landasan Teori	6
2.2 Dasar Teori	9
BAB III Metodologi Penelitian.....	25
3.1 Metode Penelitian.....	27
3.2 Penjelasan Skema Penelitian	28
BAB IV Analisis Dan Penelitian	34
4.1 Analisis	34
4.2 Perancangan.....	37
4.3 Basis Data.....	50
4.4 Desain	52
BAB V Implementasi Dan Pengujian	58
5.1 Implementasi	58
5.2 Pengujian	65
BAB VI Kesimpulan Dan Saran	82
6.1 Kesimpulan.....	82
6.2 Saran.....	82
DAFTAR PUSTAKA	83
LAMPIRAN.....	84

DAFTAR TABEL

Tabel 2.1 Simbol Use Case	13
Tabel 2.2 Simbol Activity Diagram	15
Tabel 2.3 Simbol Sequence Diagram	16
Tabel 4.1 Analisis Masalah Dengan Metode Pieces	34
Tabel 4.2 Definisi Actor dan Deskripsi Sistem Yang Diusulkan	38
Tabel 4.3 Definisi dan Deskripsi Sistem Yang Diusulkan	38
Tabel 4.4 Skenario Login	39
Tabel 4.5 Skenario Tampil Beranda Admin	39
Tabel 4.6 Skenario Pengelolaan Data Menu	40
Tabel 4.7 Skenario Pengelolaan Transaksi	41
Tabel 4.8 Skenario Pengelolaan Kasir	41
Tabel 4.9 Skenario Pengelolaan Laporan	42
Tabel 4.10 Field Data Menu	51
Tabel 4.11 File Data Kategori	51
Tabel 4.12 File Data Kategori	51
Tabel 4.13 File Data User	51
Tabel 4.14 File Data Transaksi	52
Tabel 5.1 Tabel Rencana Pengujian Admin	65
Tabel 5.2 Tabel Rencana Pengujian Kasir	67
Tabel 5.3 Tabel Hasil Pengujian Login Dengan Username dan Password yang Terdaftar	70
Tabel 5.4 Tabel Hasil Pengujian Login Dengan Username dan Password yang Tidak Terdaftar	70
Tabel 5.5 Tabel Hasil Pengujian Entry Data Sparepart Dengan Menginputkan Semua Data Sparepart	71
Tabel 5.6 Tabel Hasil Pengujian Entry Data Sparepart Dengan Tidak Menginputkan Salah Satu Data Sparepart	72
Tabel 5.7 Tabel Hasil Pengujian Entry Data Tambah Sparepart Dengan Menginputkan Semua Data Tambah Sparepart	73
Tabel 5.8 Tabel Hasil Pengujian Entry Data Tambah Sparepart Dengan Tidak	

Menginputkan Salah Satu Data Tambah Sparepart	73
Tabel 5.9 Tabel Hasil Pengujian Entry Data Kategori Sparepart Dengan	
Menginputkan Semua Data Kategori Sparepart	74
Tabel 5.10 Tabel Hasil Pengujian Entry Data Kategori Sparepart Dengan Tidak	
Menginputkan Salah Satu Data Kategori Sparepart	75
Tabel 5.11 Tabel Hasil Pengujian Entry Data Tambah Kasir Dengan	
Menginputkan Semua Data Tambah Kasir	76
Tabel 5.12 Tabel Hasil Pengujian Entry Data Tambah Kasir Dengan Tidak	
Menginputkan Salah Satu Data Tambah Kasir	77
Tabel 5.13 Tabel Hasil Pengujian Entry Data Work Order Dengan Menginputkan	
Semua Data Work Order	78
Tabel 5.14 Tabel Hasil Pengujian Entry Data Work Order Dengan Tidak	
Menginputkan Salah Satu Data Work Order	78
Tabel 5.15 Tabel Hasil Pengujian Entry Laporan Faktur Dengan Menginputkan	
Semua Data Laporan Faktur.....	79
Tabel 5.16 Tabel Hasil Pengujian Entry Laporan Faktur Dengan Tidak	
Menginputkan Menginputkan Salah Satu Data Laporan Faktur.....	80

DAFTAR GAMBAR

Gambar 1.1 Metode Model Driven Development	3
Gambar 2.1 Model-Driven Development Route.....	24
Gambar 3.1 Kerangka Pikir.....	27
Gambar 4.1 Usecase Diagram Sistem Yang Diusulkan.....	37
Gambar 4.2 Activity Diagram Sistem yang Diusulkan.....	43
Gambar 4.3 Activity Diagram Master Sparepart yang Diusulkan	44
Gambar 4.4 Activity Diagram Kategori Pelanggan	45
Gambar 4.5 Activity Diagram Kasir Yang Diusulkan	46
Gambar 4.6 Class Diagram	47
Gambar 4.7 Sequence Diagram Login	48
Gambar 4.8 Sequence Diagram Tambah Sparepart	49
Gambar 4.9 Sequence Diagram Transaksi.....	50
Gambar 4.10 Desain Input Admin Halaman Login	52
Gambar 4.11 Desain Input Kasir Halaman Login.....	53
Gambar 4.12 Desain Dashboard	54
Gambar 4.13 Desain Sparepart	54
Gambar 4.14 Desain Tambah Sparepart	55
Gambar 4.15 Desain Kategori Sparepart	55
Gambar 4.16 Desain Work Order	56
Gambar 4.17 Desain Transaksi Kasir	57
Gambar 4.18 Desain Laporan Faktur Oleh Kasir.....	57
Gambar 5.1 Tampilan Login Admin.....	59
Gambar 5.2 Tampilan Login Kasir	60
Gambar 5.3 Tampilan Dashboard	60
Gambar 5.4 Tampilan Sparepart	61
Gambar 5.5 Tampilan Tambah Sparepart	61
Gambar 5.6 Tampilan Tambah Kategori Sparepart	62
Gambar 5.7 Tampilan Work Order	62
Gambar 5.8 Tampilan Laporan Work Order.....	63
Gambar 5.9 Tampilan Cetak Laporan Work Order	64

GAMBAR 5.10 Tampilan Laporan Faktur	64
---	----

LAMPIRAN

BAB I

PENDAHULUAN

1.1 Latar Belakang

Moto Link Bengkel merupakan suatu usaha jenis usaha kecil dan menengah yang bergerak dalam bidang jasa pelayanan perbaikan mobil, lebih dari itu bengkel juga melakukan usaha penjualan *spare part* guna melengkapi kebutuhan penggantian *spare part* kendaraan yang rusak. Salah satu jenis bengkel yang bergerak dalam bidang jasa *service dan penjualan spare part* mobil adalah bengkel Moto Link yang bertempat di Jl. Raya Banjaran No. 183 Kabupaten Bandung, Jawa Barat.

Banyak perusahaan atau badan usaha yang menggunakan teknologi informasi untuk meningkatkan produktifitas dan efisiensi pekerja dalam bidang usaha, bengkel Moto Link adalah bengkel mobil yang menawarkan jasa *service* sekaligus menjual *spare part* mobil. Transaksi dan data yang semakin banyak menyebabkan pendataan *spare part* yang dinilai kurang efisien.

Bengkel Moto Link memiliki banyak pelanggan, namun terkadang seringkali terjadi kesalahan dalam penghitungan penjualan *spare part*, hal ini disebabkan karena sistem pendataan masih menggunakan cara manual dimana pencatatan ditulis pada secarik kertas atau buku catatan serta penggunaan kalkulator untuk penghitungan. Hal ini membuat pengolahan data tidak efisien dan sering timbul masalah seperti hilangnya faktur-faktur penjualan *spare part* yang mana merupakan bukti-bukti dari transaksi yang dilakukan.

Tujuan dibangunnya aplikasi berbasis web ini diharapkan mempermudah pengelolaan data transaksi dan laporan penjualan. Aplikasi berbasis web memiliki kelebihan performa yang baik karena berjalan diatas spesifikasi hardware yang relatif ringan karena hanya membutuhkan browser internet dan mudah diakses dimana saja.

Atas dasar inilah penulis mengambil judul “APLIKASI PENJUALAN SPARE PART MOBIL MENGGUNAKAN CODE IGNITER UNTUK KEAKURATAN PELAPORAN DATA PADA BENGKEL MOTO LINK”.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang ada maka rumusan permasalahan dalam penelitian ini adalah sebagai berikut :

1. Seringnya terjadi kehilangan faktur penjualan yang berakibat sulitnya pembuatan laporan pada pemilik bengkel.
2. Sering tidak sesuai hasil laporan dengan pendapatan yang diberikan.

1.3 Batasan Masalah

Adapun batasan-batasan masalah adalah sebagai berikut :

1. Aplikasi dibuat dapat faktur penjualan setelah dicetak.
2. Aplikasi hanya digunakan oleh admin dan kasir bengkel.

1.4 Tujuan Penelitian

Penelitian ini dibuat dengan tujuan dapat memberikan keamanan untuk faktur faktur agar tidak hilang serta memberikan hasil laporan yang valid dan dapat dipertanggung jawabkan

1.5 Metodologi Penelitian

Dalam upaya pengumpulan data dan penyelesaian masalah yang di teliti, penulis menggunakan metodologi sebagai berikut :

1. Metodologi pengumpulan data

Metode pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut :

1. Studi pustaka

Studi Pustaka merupakan pengumpulan data dengan menggunakan atau mengumpulkan sumber-sumber tertulis, dengan cara membaca, mempelajari dan mencatat hal-hal penting yang berhubungan dengan masalah yang sedang dibahas guna memperoleh gambaran secara teoritis.

2. Pengamatan (*Observasi*)

Pengumpulan data dengan melakukan pengamatan secara langsung terhadap objek penelitian. Dengan mencatat hal-hal penting yang berhubungan dengan judul laporan, sehingga diperoleh data yang lengkap dan akurat.

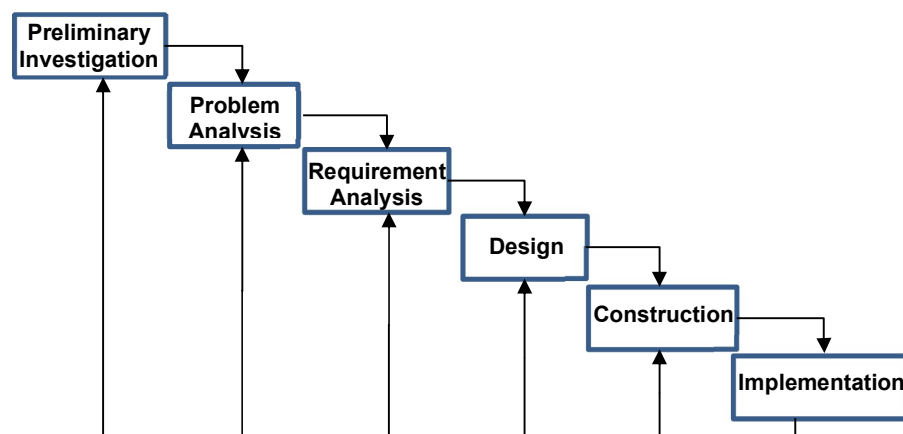
3. Wawancara (*Interview*)

Wawancara merupakan kegiatan tanya jawab yang terjadi secara langsung antara dua orang atau lebih. Tujuan wawancara adalah untuk mendapatkan informasi dari narasumber terpercaya.

2. Metodologi Pengembangan

Metodologi pengembangan sistem (*System Development Methodology*) adalah proses pengembangan sistem yang sangat formal dan akurat dalam mendefinisikan sekumpulan aktifitas, metode, praktek-praktek terbaik, penyampaian, dan alat terotomasi yang digunakan oleh pengembang sistem dan manajer proyek untuk mengembangkan dan memelihara sistem dan software informasi.

Metodologi pengembangan sistem yang akan digunakan menggunakan MDD (*Model Drive Development*). Model ini merupakan suatu teknik yang menekankan penggambaran model untuk memvisualisasikandan menganalisis masalah, mendefinisikan proses bisnis, dan merancang sistem informasi.



Gambar 1.1 MDD (Model Driven Development)

Pada tahap ini, penulis memulai pekerjaan dengan mendefinisikan dan mengumpulkan semua yang dibutuhkan dalam merancang suatu sistem seperti data yang dibutuhkan tempat penelitian.

1. *Preliminary Investigation*

Pada tahap ini, penulis memulai pekerjaan dengan mendefinisikan dan mengumpulkan semua yang dibutuhkan dalam merancang suatu sistem seperti data yang dibutuhkan tempat penelitian.

2. *Problem Analysis*

Pada tahap ini, penulis menganalisis masalah-masalah yang terdapat di lapangan dengan melakukan analisis terhadap sistem yang sedang berjalan.

3. *Requirements Analysis*

Pada tahap ini akan dilakukan pengurutan prioritas dari kebutuhan-kebutuhan bisnis yang ada. Tujuan dari tahapan ini adalah mengidentifikasi data, proses dan antarmuka yang diinginkan pengguna dari sistem yang baru.

4. *Design*

Pada tahap ini, dilakukan perancangan sistem yang bertujuan untuk memberikan gambaran apa yang seharusnya dikerjakan serta bagaimana tampilannya. Design tersebut meliputi rancangan output, rancangan input, rancangan lainnya. Tahap ini membantu dalam menspesifikasikan kebutuhan dan arsitektur sistem secara keseluruhan.

5. *Construction*

Setelah pembuatan *physical design*, maka akan dimulai untuk mengkontruksi dan melakukan tahap uji coba terhadap sistem yang memenuhi kebutuhan-kebutuhan bisnis dan spesifikasi desain. Basis data, program aplikasi, dan antarmuka akan mulai dibangun pada tahap ini. Setelah dilakukan uji coba terhadap keseluruhan sistem, maka sistem siap untuk diimplementasikan.

6. *Implementation*

Tahap ini menerapkan hasil rancangan yang telah disusun sedemikian rupa ke dalam sistem untuk mendapatkan kondisi yang sesuai dengan kebutuhan, input dari tahapan ini adalah sistem fungsional dari tahapan konstruksi.

1.6 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini menjelaskan secara singkat mengenai latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, metode penelitian dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi tinjauan penelitian yang menunjukkan bahwa skripsi yang ditulis terkait dengan penelitian dan dasar teori yang menjelaskan teori yang berkaitan dengan tema penelitian dari 3 jurnal yang sesuai.

BAB III METODOLOGI

Bab ini menjelaskan secara singkat metode yang digunakan untuk mendapatkan data yang diperlukan serta metode yang digunakan dalam penyelesaian masalah penelitian. Bab ini juga berisi gambar kerangka pikir dan deskripsi. Kerangka pikir berisi gambaran pola hubungan antara variabel atau kerangka konsep yang akan digunakan untuk menjelaskan masalah yang diteliti, disusun berdasarkan kajian teoritik. Deskripsi yaitu penjelasan langkah-langkah pada kerangka pikir.

BAB IV ANALISIS DAN PERANCANGAN

Pada bab ini menjelaskan analisis kebutuhan aplikasi yang akan dibangun, serta menjelaskan segala bentuk perancangan sistem yang akan dibuat seperti perancangan basis data, dan perancangan tampilan aplikasi yang diusulkan.

BAB V IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dijelaskan mengenai pembuatan aplikasi. Adapun tahapan-tahapan dalam pembuatan aplikasi meliputi pembuatan basis data, dan pembuatan aplikasi sistem informasi bimbingan skripsi online mahasiswa.

BAB VI KESIMPULAN DAN SARAN

Berisi kesimpulan dan saran. Kesimpulan berisi ringkasan hasil implementasi dan pengujian. Sedangkan saran berisi usulan-usulan lanjut dari permasalahan yang ditinjau.

BAB II

TINJAUAN PUSTAKA

2.1. Landasan Teori

Dalam melakukan penelitian, penulis juga mempelajari referensi dari penelitian yang terdahulu sebagai landasan teori yang melandasi penelitian yang tengah dilakukan. Beberapa landasan teori tersebut yaitu :

1. **Sistem Informasi Penjualan Suku Cadang Berbasis Web Pada Distributor dan Bengkel AHASS (ASTRA HONDA AUTHORIZED SERVICE STATION)**, Oleh : Metty Amelia , Sistem Informasi, Mercu Buana, Jl. Raya Menteng No. 29 Jakarta Pusat. AHASS 10147 Mustika Motor Indramayu merupakan sebuah bengkel yang bergerak dibidang pelayanan jasa perawatan / pemeliharaan (H2) dan penjualan spare part (H3). Persaingan bisnis yang sangat ketat dibidang sejenis mendorong AHASS 10147 Mustika Motor untuk meningkatkan kepercayaan pelanggan. Upaya yang dilakukan AHASS 10147 Mustika Motor dalam meningkatkan kepercayaan pelanggan adalah dengan menggunakan media penjualan yang berbasis web. Untuk melakukan strategi penjualan menggunakan metode analisis berupa PIECES dengan melalui beberapa tahapan seperti kinerja, informasi, ekonomi, pengendalian, efisiensi dan pelayanan. Adapun pengembangan sistemnya menggunakan metode Waterfall dengan melalui beberapa tahapan seperti analisis, perencanaan, perancangan, construction dan Deployment. Setelah pengembangan sistem dilakukan maka dapat disimpulkan bahwa sistem yang selama ini diterapkan pada AHASS 10147 Mustika Motor membutuhkan sistem penjualan yang baru dengan menggunakan web dan semua informasi yang ada di dalamnya (Amelia, 2018).

Kata kunci : Sistem, Informasi, Penjualan, PIECES, Waterfall.

2. Sistem Informasi Penjualan Sparepart Motor Secara Online Berbasis Web

Pada CV. APE Spare Part, Oleh : Doddy Hermawan Putranto, Sistem Informasi, Universitas Pembangunan Nasional Veteran Jakarta, Jl. RS. Fatmawati Raya, Pd. Labu, Kec Cilandak, Kota Depok. Perkembangan dunia teknologi komputer tidak akan pernah berhenti, bahkan kini telah meningkat ke dekade keempat sejak keberadaannya yang tidak dipisahkan dalam bidang bisnis online. Sekarang, hampir semua bidang usaha dalam menjalankan kegiatan perekonomian memerlukan perangkat keras komputer sebagai alat bantu untuk mempermudah pekerjaan. Dalam pengolahan data sistem informasi penjualan spare part MOTOR pada toko “CV. APE SPARE PART” ini masih belum terkomputerisasi. Oleh karena itu, penulis berusaha membuat aplikasi dengan menggunakan bahasa pemrograman Dreamweaver, dengan Apache sebagai web servernya dan MySQL sebagai databasenya untuk mengatasi kesulitan tersebut. Maka dengan adanya aplikasi penjualan online yang berbasis web akan mempermudah bagi para pelanggan dalam melakukan transaksi serta membantu dalam promosi barang kepada pelanggan. Dalam memperoleh data untuk menyusun penelitian ini, penulis menggunakan metode penelitian waterfall dikarenakan metode ini lebih sistematis dari setiap level kebutuhan sistem yang terjadi pada setiap penjualan. Kemudian luaran yang akan dihasilkan dalam skripsi ini adalah sebuah prototype dan aplikasi penjualan spare part sepeda motor secara online berbasis web, sehingga dapat membantu perusahaan dalam penyampaian informasi mengenai produk yang ditawarkan secara detail kepada konsumen dan memungkinkan pelanggan dalam melakukan transaksi pemesanan secara online. Berdasarkan hasil yang diperoleh diharapkan dapat membantu perusahaan memasarkan strategi dan jangkauan pemasaran secara luas (Putranto, 2015).

Kata Kunci : Penjualan, sparepart motor, online

3. Perancangan dan Implementasi Aplikasi Web Sistem Penjualan Suku

Cadangan Gondola Pada CV Yoga Perkasa Mandiri, Oleh: Nur Syamsiyah, Sistem Informasi, Universitas Darma Persada Jakarta Timur, Jl. Radin Inten II (Terusan Casablanca) Pondok Kelapa. CV. Yoga Perkasa Mandiri adalah

perusahaan yang bergerak pada bidang perawatan gedung yaitu penyedia sparepart gondola yang masih menggunakan dokumen-dokumen kertas dalam proses penjualan sparepart gondola atau belum terkomputerisasi. Supaya perusahaan dapat berkembang meningkat maka diperlukan sistem penyediaan layanan yang lebih efektif dan efisien dalam proses penjualannya. Tujuan penelitian dibangun sebuah aplikasi berbasis web yang bisa memproses penjualan sparepart gondola dengan cara lebih praktis, cepat, akurat dan handal. Perancangan aplikasi sistem penjualan berbasis web ini dibangun menggunakan konsep data model yaitu ERD (Entity Relationship Diagram), bahasa pemrograman PHP dan MySQL untuk database penyimpanan serta dijalankan pada sistem operasi Windows 7 atau versi Windows yang lebih tinggi. Hasil akhir dari perancangan yaitu aplikasi penjualan sparepart gondola dengan pemodelan terstruktur agar data masukan akan diolah dengan sistem manajemen database yang lebih baik sehingga memberikan kemudahan, ketepatan perhitungan, ketelitian, kecepatan proses pelaporan serta memperkecil kesalahan yang dapat terjadi karena sistem yang baru sudah terstruktur dan rapi (Syamsiyah, 2018).

Kata Kunci : Suku Cadang, Gondola, Aplikasi Web

4. **Perancangan E-Commerce Untuk Penjualan Spareparts Pada Bengkel ABM Motor**, Oleh : Wisnu Prabowo, Sistem Informasi, Universitas Budi Luhur, Jl. Raya Ciledug, Pertukangan Utara, Kebayoran Lama, Jakarta Selatan 122
ABM Motor merupakan sebuah Bengkel Motor. Bengkel ini didirikan oleh bapak Aris Paryanto pada tahun 2007. Bengkel ini membutuhkan sistem informasi yang mendukung. Sistem informasi sangat penting dalam mengelola dan menjalankan semua kegiatan yang ada didalam instansi. Diantaranya proses penjualan untuk memenuhi kebutuhan Bengkel, khusus diantaranya mengenai masalah yang timbul yaitu tidak mengetahui proses pencatatan pemesanan barang, tidak adanya jumlah perhitungan stok barang

5. karena tidak adanya perhitungan jumlah stok barang, tidak adanya laporan penjualan. Untuk mengetahui masalah yang sudah disebutkan diatas. ecommerce dibangun menggunakan bahasa pemogramanan PHP dan data base menggunakan MYSQL, Tujuan dari penelitian ini adalah perancangan sistem informasi penjualan barang berbasis web yang dapat digunakan untuk membantu dan mempermudah proses penjualan barang dan membantu Bengkel ABM Motor agar bisa bersaing dengan Bengkel lainnya di bidang penjualan. Kesimpulan penelitian ini adalah Dengan adanya sistem e-commerce penjualan pada Bengkel ABM Motor dapat meningkatkan penjualan karena customer tidak perlu lagi datang kebengkel untuk membeli spareparts (Prabowo, 2019).

Kata kunci: penjualan, e-commerce, bengkel, sistem informasi.

2.2. Dasar Teori

Dalam melakukan penelitian, penulis juga memiliki dasar untuk menunjang penelitian yang tengah dilakukan, dan dasar tersebut adalah :

2.2.1 Pengertian Aplikasi

Aplikasi merupakan bagian dari perangkat lunak (software) yang di rancang khusus untuk membantu mengatasi permasalahan tertentu sesuai dengan kebutuhan pemakai. Aplikasi digunakan dan dirancang sesuai dengan kebutuhan penggunanya. Mulai dari sebuah perusahaan besar, menengah bahkan sebuah komunitas yang menurut mereka aplikasi dapat dimanfaatkan dengan baik. Berikut beberapa pendapat para ahli mengenai pengertian dari aplikasi, yaitu:

Menurut Syahrial Chan (2017:4) mendefenisikan : “Aplikasi adalah koleksi window dan objek-objek yang menyediakan fungsi untuk aktivitas user, seperti pemasukan data, proses, dan pelaporan”.

2.2.2 Pengertian Penjualan

Berikut ini beberapa pengertian tentang penjualan menurut beberapa ahli yang dijabarkan dibawah ini:

Menurut William G. Nickles, penjualan adalah proses dimana penjual memuaskan segala kebutuhan dan keinginan pembeli agar dicapai manfaat baik bagi penjual maupun pembeli yang berkelanjutan dan menguntungkan kedua belah pihak

(Nickels, 2017).

Menurut Nitisemito, penjualan adalah sebuah kegiatan yang bertujuan untuk melancarkan arus barang dan jasa dari produsen ke konsumen secara paling efisien dengan maksud untuk menciptakan permintaan yang efektif (Nitisemito, 2017).

Menurut Moekijat, penjualan adalah suatu kegiatan yang ditujukan untuk mencari pembeli, mempengaruhi dan memberi petunjuk agar pembelian dapat menyesuaikan kebutuhannya dengan produksi yang ditawarkan serta mengadakan perjanjian mengenai harga yang menguntungkan kepada kedua belah pihak. Intinya pengertian ini penjualan yaitu perjanjian yang menguntungkan antara pembeli dan penjual (Moekijat, 2017).

Pengertian di atas maka dapat disimpulkan bahwa penjualan adalah bagaimana sebagai penjual membuat strategi penjualan yang menciptakan sesuatu yang menarik konsumen, dan berusaha untuk tidak mendapat keluhan atas barang ataupun jasa yang dipasarkan.

2.2.3 Pengertian Sparepart

Marsudi (2005:219), sparepart adalah suku cadang untuk menggantikan suku cadang yang mengalami kerusakan yang tidak dapat diperkirakan sebelumnya kapan akan terjadi.

Wikipedia (2015), suku cadang adalah komponen dari mesin yang dicadangkan untuk perbaikan atau penggantian bagian kendaraan yang mengalami kerusakan.

2.2.4 Pengertian Code Igniter

Code Igniter (CI) merupakan salah satu dari sekian banyak Framework php yang ada. Tujuan dari pembuatan Framework Code Igniter ini menurut user manual nya adalah untuk menghasilkan Framework yang akan dapat digunakan untuk pengembangan proyek pembuatan website secara lebih cepat dibanding dengan pembuatan website dengan cara coding secara manual, dengan menyediakan banyak sekali pustaka yang dibutuhkan dalam pembuatan website, dengan antarmuka yang sederhana dan struktur logika untuk mengakses pustaka yang dibutuhkan. Code Igniter sebuah Framework php yang bersifat open source dan

menggunakan metode MVC (Model, View, Controller). Code Igniter bersifat free alias tidak berbayar jika anda menggunakannya. Framework Code Igniter dibuat dengan tujuan sama seperti Framework lainnya yaitu untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal (Ngoding, 2018).

Code Igniter adalah sebuah web application Framework yang bersifat open source, digunakan untuk membangun aplikasi PHP dinamis. Tujuan utamanya adalah untuk membantu developer membuat aplikasi lebih cepat daripada menulis semua kode dari awal. Ada banyak library dan helper yang berguna di dalamnya dan tentunya mempermudah proses development. Code Igniter merupakan salah satu Framework tercepat dibandingkan dengan Framework lainnya.

2.2.5 OOAD (*Object Oriented Analysis Design*)

OOAD (*Object Oriented Analysis Design*) adalah metode analisis yang memeriksa *requirement* dari sudut pandang kelas-kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem. OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat Menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas. OOAD Mempelajari permasalahan dengan menspesifikasikan nya atau mengobservasi permasalahan tersebut dengan menggunakan metode berorientasi objek. Biasanya analisa sistem dimulai dengan adanya dokumen permintaan (*reuirement*) yang diperoleh dari semua pihak yang berkepentingan. Misalnya *klien*, *developer*, pakar, dan lain-lain.

Tahapan dalam proses OOAD adalah sebagai berikut:

a. Requirement

Pada tahap *requirement* menjelaskan tentang bagaimana membuat dan menjaga sebuah perjanjian kerjasama dengan *customer* dan *stakeholder* mengenai apa yang harus dilakukan pada pembuatan sistem misalnya kebutuhan-kebutuhan dari suatu sistem, seperti fungsi-fungsi apa saja yang akan digunakan oleh sistem, fitur-fitur apa saja yang akan dipakai

dalam pembuatan sistem, siapa saja yang terlibat atau yang memakai sistem tersebut dan lain sebagainya.

b. Analysis and Design

Tahapan ini menjelaskan bagaimana mengkonversi kebutuhan kebutuhan sistem yang telah dibuat sebelumnya menjadi sebuah bentuk rancangan sistem. Yang mana rancangan tersebut disajikan sebagai spesifikasi dari implementasi sistem dalam memilih lingkungan dalam penerapannya. Selain itu, tahapan analisis dan desain juga mencakup dalam mengembangkan sebuah arsitektur/bentuk rancangan yang kokoh.

c. Implementation

Setelah membuat suatu kebutuhan sistem, menganalisis dan merancang sistem yang akan dibuat. Kemudian pada tahap ini dilakukan implementasi unit sistem dan menggabungkan rancangan dengan sistem maksudnya adalah mengimplementasikan rancangan tersebut dalam sebuah sistem atau program dalam bentuk *coding* program. Selain itu juga menghasilkan sebuah sistem yang sudah dapat dijalankan.

d. Testing

Test/pengujian dilakukan untuk meyakinkan bahwa sistem telah sesuai dengan kebutuhan-kebutuhan yang telah dibuat sebelumnya (kebutuhan-kebutuhan tersebut adalah menerapkan yang sewajarnya). Memvalidasi fungsi sistem yang telah konkrit kemudian didemonstrasikan bahwa produk software tersebut sesuai dengan kebutuhan dan rancangan yang telah dibuat sebelumnya.

e. Deployment

Menyakinkan bahwa produk software tersebut (termasuk implementasi dan pengujian) telah tersedia untuk end user atau sudah dapat digunakan oleh pengguna akhir.

2.2.6. Unified Modeling Language (UML)



“Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan





requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek” (Rosa dan Shalahuddin, 2018:133).

a. Use Case Diagram

“Usecase diagram merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat secara kasar yang digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa siapa saja yang berhak menggunakan fungsi-fungsi itu” (Rosa dan Shalahuddin, 2018:155). Simbol Simbol yang digunakan pada use case diagram bisa dilihat pada tabel dibawah ini.

Tabel 2.1 Simbol-Simbol Use Case Diagram

NO	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor





3		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4		<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan
5		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
6		<i>Include</i>	Menspesifikasikan bahwa use case sumber secara eksplisit.


b. Activity Diagram

“Activity Diagram menggambarkan workflow atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor jadi aktivitas yang dapat dilakukan oleh sistem” (Rosa dan Shalahuddin, 2018:161).

Simbol Simbol yang digunakan pada Activty diagram bisa dilihat pada tabel dibawah ini.




Tabel 2-2 Simbol Activity Diagram

NO	Simbol	Nama	Keterangan
1		Status Awal	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
2		Aktivitas	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
3		Percabangan	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4		Penggabungan	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu

5		Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
---	---	--------------	---

c. Sequence Diagram

Tabel 2-3 Simbol-simbol Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

“Sequence Diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada use case” (Rosa dan Shalahuddin, 2018:165).

2.2.7. Balsamiq Mockup

Balsamiq Mockup adalah salah satu software yang digunakan dalam pembuatan desain atau prototipe dalam pembuatan tampilan user interface sebuah

aplikasi. Dengan menggunakan Balsamiq Mockup kita dimudahkan dalam pembuatan user interface karena Balsamiq Mockup sudah menyediakan tools yang dapat memudahkan dalam membuat desain prototyping aplikasi yang akan kita buat. Kelebihan Balsamiq Mockup dibanding software pembuat Mock Up lainnya adalah aplikasi ini berbasis cloud, disertai aplikasi desktop yang memungkinkan kita dengan cepat dan mudah membuat rancangan website. Di websitenya sendiri ada dua pilihan untuk para pengguna, ada versi trial for desktop dan ada juga yang bisa kita download untuk versi desktop. Namun ada juga yang disediakan dalam versi berbayar. Aplikasi ini bisa digunakan untuk sistem operasi Windows, Mac OS, dan Linux (Maimanah, 2015).

2.2.8. Hypertext Preprocessor

Menurut Firdaus (2017), PHP merupakan singkatan dari Hypertext Preprocessor, adalah sebuah bahasa scripting berbasis server Side scripting yang terpasang pada HTML dan berada di server dieksekusi di server dan digunakan untuk membuat halaman web yang dinamis. Sebagian besar sintaknya mirip dengan bahasa C atau Java, ditambah dengan beberapa fungsi PHP yang spesifik. Tujuan utama bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamis dengan cepat. Halaman Web biasanya disusun dari kode-kode HTML yang disimpan dalam sebuah file berekstensi titik .html. File html ini dikirimkan oleh server atau file ke browser, kemudian berputar menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program PHP, program ini harus diterjemahkan oleh webserver sehingga menghasilkan kode html yang dikirim ke browser agar dapat ditampilkan. Program ini dapat berdiri sendiri ataupun disisipkan diantara kode-kode HTML sehingga dapat langsung ditampilkan bersama dengan kode-kode HTML tersebut. Program PHP dapat ditambahkan dengan mengait program tersebut di antara tanda kurang <? Dan tanda >?. tanda-tanda tersebut biasanya digunakan untuk memisahkan kode PHP dari kode HTML. File html yang telah dibubuhi program PHP harus diganti ekstensinya menjadi di .php atau .php3 (Firdaus, 2017).

2.2.9. Database

Database merupakan aspek yang sangat penting dalam sistem informasi karena berfungsi sebagai gudang penyimpanan data yang lalu diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data menghindari, duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga update yang rumit.

Menurut Rosa dan Shalahuddin (2015:43) “basis data merupakan salah satu bagian dalam rekayasa perangkat lunak yang terkomputerisasi dan bertujuan utama memelihara data yang sudah diolah atau media penyimpanan informasi agar dapat diakses dengan mudah dan cepat”.

Sedangkan menurut Yakub dan Hisbanarto (2015:25) menjelaskan, “basis data (database) merupakan kumpulan data yang saling berhubungan atau punya relasi”. Dapat disimpulkan bahwa basis data bagian dari rekayasa perangkat lunak yang terkomputerisasi sebagai media penyimpanan informasi yang saling berhubungan atau punya relasi untuk penyimpanan data informasi agar dapat diakses dengan mudah dan cepat.

- Himpunan kelompok data yang saling terhubung dan diorganisasi sedemikian rupa supaya kelak dapat dimanfaatkan kembali secara cepat dan mudah.
- Kumpulan data dalam bentuk file/tabel/arsip yang saling berhubungan dan tersimpan dalam media penyimpanan elektronis, untuk kemudahan dalam pengaturan, pemilahan, pengelompokan dan pengorganisasian data sesuai tujuan.

Dengan basis data seseorang dapat menyimpan sebuah informasi seperti data mahasiswa, kepegawaian atau produk ke dalam media penyimpanan elektronis seperti cakram magnetis (*disk*) melalui perangkat komputer, untuk kemudian data tersebut dapat kita gunakan sesuai keperluan. Database mempunyai 8 operasi dasar diantaranya adalah Create database, Drop database, Create table, Drop table, Insert, Update dan delete (Salamadian, 2018).

2.2.10. Website

Website menurut Gregorius adalah kumpulan halaman web yang saling

terhubung dan file-filenya saling terkait. Web terdiri dari page atau halaman, dan kumpulan halaman yang dinamakan homepage. Homepage berada pada posisi teratas, dengan halaman terkait berada di bawahnya. Biasanya setiap halaman di bawah homepage disebut child page, yang berisi hyperlink ke halaman lain dalam web (Gregorius, 2019)

Menurut Hakim Lukmanul, Website merupakan salah satu dari sekian fasilitas internet yang menghubungkan dokumen dalam lingkup lokal dan juga jarak jauh. Adapaun dokumen pada website disebut dengan web page dan link dalam website memungkinkan pengguna bisa berpindah dari satu page ke page lain (hyper text), baik diantara page yang disimpan dalam server yang sama maupun server diseluruh dunia. Page diakses dan dibaca melalui browser seperti Netscape Navigator, Internet Explorer, Mozilla Firefox, Google Chrome dan aplikasi browser lainnya (Lukmanul, 2019)

Menurut Suwanto Raharjo, layanan web adalah salah satu Internet yang paling banyak dipergunakan dibandingkan dengan layanan lain seperti ftp, gopher, news atau bahkan email (Raharjo, 2019).

Menurut Maulana, website adalah salah satu aplikasi yang berisikan dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol HTTP (Hypertext Transfer Protocol) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut browser (Maulana, 2015). Menurut Puspitosari dalam Kesuma & Rahmawati (2017:3) menjelaskan bahwa “Website adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses diseluruh dunia, selama terkoneksi dengan jaringan internet” Berdasarkan penjelasan diatas penulis dapat menyimpulkan bahwa Website adalah suatu wadah yang dibuat menggunakan bahasa pemrograman yang dapat digunakan oleh user untuk membuat dan mengembangkan sesuatu, membuat halaman-halaman yang ditampilkan pada user *interface*.

a. Www (World Wide Web)

Menurut Fathansyah dalam Prayitno & Safitri (2015:2) mengatakan bahwa, “World Wide Web (WWW atau web) merupakan sistem informasi terdistribusi yang berbasis hypertext”. Menurut Kustiyahningsih dan Devie Rosa Anamisa dalam Fridayanthie & Mahdiati (2016:128) mengatakan

bahwa,” Word Wide Web (WWW). Informasi WWW ini disimpan pada web server untuk dapat diakses dari jaringan browser terlebih dahulu, seperti Internet Explorer atau Mozilla Firefox”. Berdasarkan penjelasan diatas penulis dapat menyimpulkan bahwa www adalah pada web server untuk dapat diakses dari jaringan browser.

b. Internet

Menurut Sibero Internet adalah jaringan komputer yang menghubungkan antar jaringan secara global”. Berdasarkan penjelasan dari para ahli, dapat disimpulkan bahwa Internet adalah jaringan komputer yang terkoneksi dengan jaringan lain yang mempunyai cakupan luas untuk mendapatkan informasi dari jaringan tersebut.(Arizona, 2017). Menurut Simarmata dalam Arizona (2017:107) menjelaskan bahwa “Internet adalah kelompok atau kumpulan dari jutaan komputer untuk mendapatkan informasi dari komputer yang ada didalam kelompok tersebut dengan asumsi bahwa pemilik komputer memberikan izin akses”. Berdasarkan dari kutipan diatas dapat disimpulkan bahwa Internet adalah jaringan komputer yang terkoneksi dengan jaringan lain yang mempunyai cakupan luas untuk mendapatkan informasi dari jaringan.

c. Web server

Web Server adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak. Secara bentuk fisik dan cara kerjanya, perangkat keras web server tidak berbeda dengan komputer rumah atau PC, yang membedakan adalah kapasitas dan kapabilitas Sihombing (2016). Sedangkan menurut Fathansyah dalam Prayitno & Safitri (2015:2) menerangkan bahwa pengertian web server adalah “Server Web (Web Server) merujuk pada perangkat keras (server) dan perangkat lunak yang menyediakan layanan akses kepada pengguna melalui protokol komunikasi HTTP ataupun variannya (seperti FTP dan HTTPS) atas berkas-berkas yang terdapat pada suatu URL ke pemakai”. Berdasarkan dari kutipan diatas dapat disimpulkan bahwa web server adalah web server yang merujuk pada perangkat keras dan perangkat lunak pada sebuah komputer.

d. Web browser

Menurut Winarno dan Utomo dalam Prayitno & Safitri (2015:2) menjelaskan bahwa, “web browser adalah alat yang digunakan untuk melihat halaman web“. Menurut Sadeli dalam Wasiyanti & Talaohu (2016:50) menjelaskan bahwa, “suatu perangkat lunak web editor keluaran Adobe System yang digunakan untuk membangun dan mendesign suatu website dengan fitur-fitur yang menarik dan kemudahan dalam penggunaannya“. Berdasarkan dari kutipan diatas dapat disimpulkan bahwa Web browser adalah alat yang digunakan untuk melihat halaman web dengan fitur- fitur yang menarik dan kemudahan dalam penggunaannya.

2.2.11. XAMPP

XAMPP adalah perangkat lunak bebas (free software), yang merupakan kompilasi dari beberapa program, XAMPP sudah menyertakan Apache web server, PHP dan basis data MySQL, sehingga kita tidak perlu lagi melakukan instalasi dan pengaturan karena XAMPP sudah menyediakan semuanya. XAMPP juga tersedia untuk sistem operasi Windows dan Linux.

XAMPP mampu menggabungkan tiga aplikasi ke dalam satu paket, yaitu Apache, MySQL, dan PHPMyAdmin, Dengan XAMPP pekerjaan sangat dimudahkan karena dapat menginstalasi dan mengkonfigurasi ketiga aplikasi tersebut dengan sekaligus dan otomatis. XAMPP adalah perangkat yang menggabungkan tiga aplikasi ke dalam satu paket, yaitu Apache, MySQL, dan PHPMyAdmin, dengan XAMPP pekerjaan Anda akan sangat dimudahkan karena dapat menginstalasi dan mengkonfigurasi ketiga aplikasi tersebut dengan sekaligus dan otomatis.

XAMPP telah mengalami perkembangan dari waktu ke waktu. Versi yang terbaru adalah revisi dari yang terdahulu, sehingga lebih baik dan lebih lengkap. Aplikasi utama dalam paket XAMPP yakni terdiri atas web server Apache, MySQL, PHP, dan PHPMyAdmin (Affandi, 2015).

1. Fitur-fitur XAMPP

Berikut ini terdapat beberapa fitur-fitur XAMPP, terdiri atas:

a. Apache

Apache adalah perangkat lunak sumber terbuka yang menjadi

alternatif dari server web netscape. Server http Apache atau server web/WWW Apache merupakan server web yang dapat dijalankan di banyak sistem operasi yang berguna untuk melayani dan memfungsikan situs web. Apache dikembangkan oleh komunitas terbuka yang terdiri dari pengembangan pengembangan di bawah naungan Apache Software Foundation.

b. MySQL

MySQL adalah singkatan “My Structured Query Language”. Program ini berjalan sebagai server menyediakan multi-user mengakses ke sejumlah database. MySQL umumnya digunakan oleh perangkat lunak bebas yang memerlukan fitur penuh sistem manajemen database, seperti WordPress, phpBB dan perangkat lunak lain yang dibangun pada perangkat lunak LAMP. Ia juga digunakan dalam skala sangat tinggi World Wide Web, termasuk produk-produk Google and Facebook.

c. PHP

PHP Adalah bahasa pemrograman script yang banyak dipakai untuk memprogramkan situs web dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. Contoh terkenal dari aplikasi PHP adalah phpBB dan MediaWiki (software dibelakang Wikipedia). Contoh terkenal dari aplikasi PHP adalah phpBB dan MediaWiki (software dibelakang Wikipedia).

Bagian-bagian XAMPP

Berikut ini terdapat beberapa bagian-bagian XAMPP, terdiri atas:

a) Htdocs

Htdocs adalah folder yang digunakan sebagai tempat penyimpanan berkas seperti PHP, HTML dan script lain yang digunakan dalam sebuah halaman website. Secara kapasitas penyimpanan, XAMPP tergantung dari seberapa besar kapasitas hardisk di laptop atau komputer anda. Sedangkan bila menggunakan hosting online, maka tergantung pilihan waktu membeli sebuah hosting.

b) PHPMyAdmin

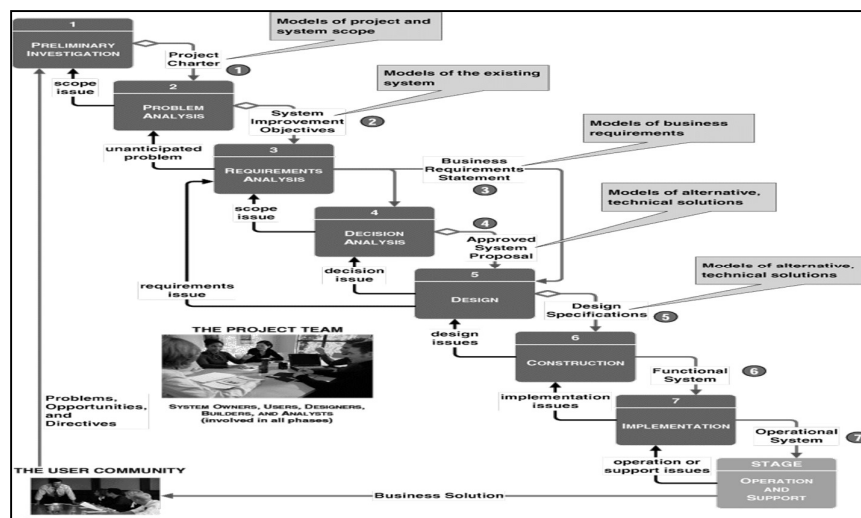
PHPMyAdmin adalah sebuah tempat yang digunakan untuk mengelola

database MySQL yang berada di komputer atau laptop. Untuk mengakses PHPMyAdmin yakni dengan membuka browser Internet (Mozilla atau Chrome) lalu ketikkan alamat [http://localhost/ PHPMyAdmin](http://localhost/PHPMyAdmin) maka akan muncul tampilannya.

2.2.12 Model-Driven Development (MDD)

Model-driven development (MDD) adalah sebuah paradigma untuk menulis dan mengimplementasikan program komputer secara cepat, efektif dan berbiaya minimum. Pendekatan MDD untuk pengembangan perangkat lunak memungkinkan orang untuk bekerja sama dalam sebuah proyek bahkan dengan tingkat pengalaman masing-masing yang sangat bervariasi. Hal ini memungkinkan perusahaan untuk memaksimalkan kerja yang efektif pada sebuah proyek dan meminimalkan overhead yang diperlukan untuk menghasilkan perangkat lunak yang dapat divalidasi oleh pengguna akhir dalam waktu sesingkat mungkin. MDD, merupakan metodologi tangkas (agile), terus berkembang untuk memenuhi kebutuhan bisnis. MDD adalah sebuah ide yang mampu mentransformasikan model ke dalam sistem komputer. Model dapat berbentuk berbagai macam seperti Parametrics for controllers, control diagrams, program, UML. Fokus penelitian ini adalah pembahasan tentang pengembangan model dan program menggunakan notasi UML. UML singkatan dari Unified Modelling Language, merupakan notasi dalam bentuk diagram untuk membangun model pada tahap analisis dan perancangan sistem. MDD merupakan pemikiran bagaimana membawa pengembangan perangkat lunak sebagai sebuah rangkaian dari transformasi model yang berawal dari kebutuhan prasyarat (requirements) menuju sebuah model yang independen dan spesifik, dan kemudian membangkitkan kode sumber (code) yang dapat dikompilasi dalam sistem komputer. Oleh karena itu pengembangannya berorientasi model-centric dan banyak aktifitas, meliputi analisis perancangan awal dan pembangkitan kasus uji (test case), yang semuanya akan didasarkan pada model menggunakan UML. Pengembangan sistem komputer menggunakan pendekatan MDD membutuhkan beberapa perkakas pengembangan sesuai tahapan. Pada tahapan analisis dan perancangan, dibutuhkan perkakas seperti Visual Paradigm yang mampu menggambar model dalam notasi UML. Pada tahap implementasi

(pemrograman) dibutuhkan perkakas pengembangan seperti Play Framework, Yii Framework. Untuk lebih menyempurnakan pengembangan perangkat lunak dibutuhkan perkakas Integrated Development Environment (IDE) seperti Eclipse. Pembuat model akan dengan cepat mengetahui apakah model yang dikembangkan layak diimplementasikan melalui konsep Forward Engineering, arah maju dari model menuju kode sumber, dan sebaliknya pekerjaan pembuat kode sumber yaitu Programmer akan dapat diperiksa dengan cepat oleh pembuat model apakah sesuai tidak dengan model yang dikembangkan melalui reverse engineering, arah balik dari kode sumber menuju model. Programmer pun bekerja dengan mudah karena sudah ada penuntun pengembangan program berupa nama fungsi/prosedur dalam kesatuan model yang memuat hubungan model beserta spesifikasi. Pekerjaan programmer adalah mendetilkkan dari spesifikasi menjadi kode sumber yang dapat dibaca oleh komputer. Pengembangan sistem komputer menggunakan MDD dan dibantu oleh perkakas memberikan kemudahan proses forward dan reverse engineering. Kebutuhan baru pengembangan sistem mudah untuk diterapkan, perubahan kode sumber dapat ditelusuri melalui model. Semua langkah dilangsakanakan terkomputerisasi dan tidak ada keadaan magic untuk menciptakan kode sumber, berbeda halnya dengan pengembangan pendekatan konvensional seperti pengembangan berorientasi fungsi.



Gambar 2. 1 Model-Driven Development Route

2.2.13 Sublime Text

Sublime Text adalah editor teks untuk berbagai bahasa pemrograman. Hal

yang menarik dari sublime text ialah tampilannya yang menyediakan minimap script yang ditampilkan. Aplikasi ini sangat ringan, tak terlalu banyak memakan resource. Bahkan lebih ringan dari gedit bawaan Ubuntu. Fitur yang tersedia: minimap, membuka script secara side by side, bracket highlight sehingga tidak bingung mencari pasangannya, kode snippets, drag and drop direktori ke sidebar terasa mirip dengan TextMate untuk Mac OS dan masih banyak lagi.

2.2.14. Blackbox

Blackbox testing adalah tahap yang digunakan untuk menguji kelancaran program yang telah dibuat. Pengujian ini penting dilakukan agar tidak terjadi kesalahan alur program yang telah dibuat. Menurut Rosa dan Salahuddin (2015:275) “Blackbox testing yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program”. Sebuah tes dapat dikatakan berhasil ketika sebuah sistem dapat memproses data dan hasil yang ada sesuai dengan apa yang diharapkan. Ketika menggunakan metode black box, tester tidak perlu mengetahui bagaimana struktur dan desain data yang ada di dalam sistem. Mereka hanya melihat apakah sistem terjadi bugs atau tidak (Farid, 2017).

2.2.15. Whitebox

Menurut Pressman (2010:588) “Pengujian white box (Pengujian Kotak Putih), terkadang disebut juga pengujian kotak kaca (glass box testing), merupakan sebuah filosofi perencanaan test case (uji kasus) yang menggunakan struktur control yang dijelaskan sebagai bagian dari perancangan perangkat komponen untuk menghasilkan test case”.

2.2.16. Printer

Menurut Lim Rusyamsi (2009:1) Printer merupakan sebuah perangkat keras yang dihubungkan pada komputer yang berfungsi untuk menghasilkan cetakan baik berupa tulisan ataupun gambar dari komputer pada media kertas atau yang sejenisnya. Jenis-Jenis Printer komputer dapat dibedakan menjadi beberapa jenis. Masing-masing jenis printer memiliki fungsi yang berbeda. Pengguna bisa menyesuaikannya tergantung pada kebutuhan masing-masing. Beberapa jenis printer

komputer adalah sebagai berikut :

a. Printer Dot Matrix Jenis Printer Dot

Matrix merupakan printer yang metode pencetakannya menggunakan pita. Cetakan yang dihasilkan terlihat seperti titik titik yang saling menghubungkan satu dengan yang lainnya, sehingga hasil cetakan kurang halus dan juga kurang bagus. Printer ini hanya menghasilkan warna tunggal sesuai dengan warna dari pita yang dipasangkan, biasanya warna merah atau hitam. Kecepatan jenis printer ini sekitar 500 cps (character per second).

b. Printer Ink Jet

Printer ini pertama dikembangkan secara ekstensif sejak 1950 dan printer inkjet yang dapat memproduksi citra dari komputer baru dikembangkan pada 1970 dan dikuasai oleh Epson, Hewlett-Packard, dan Canon.

c. Printer Laser Jet

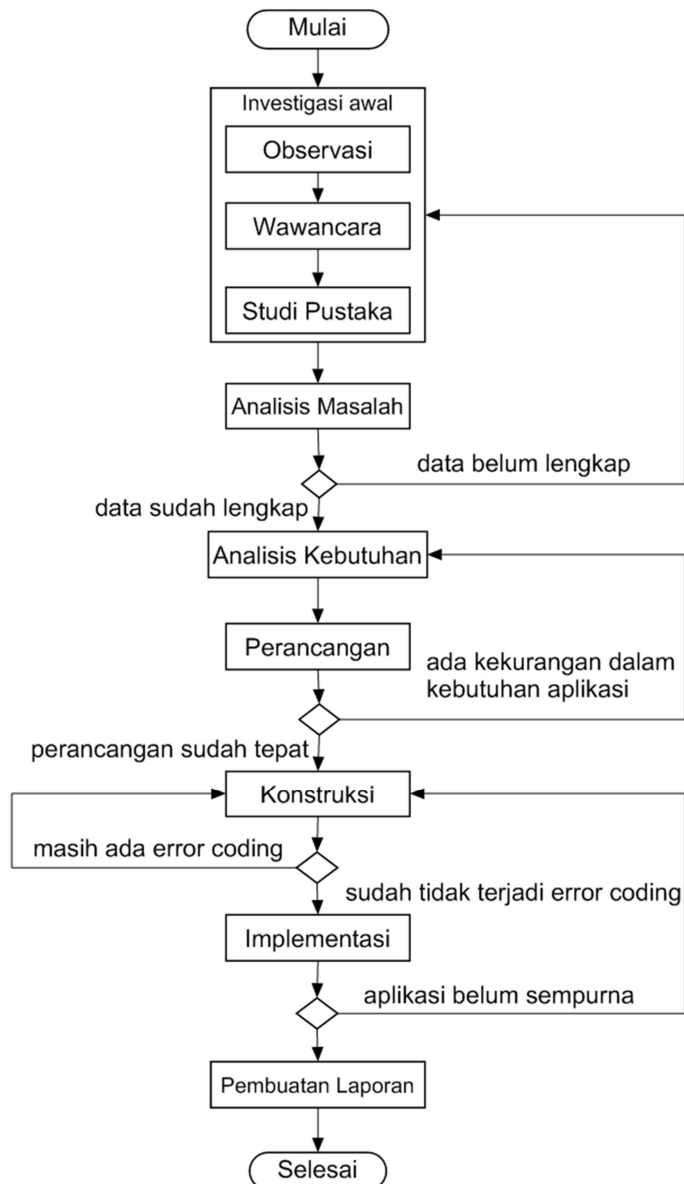
Sesuai dengan namanya laser, jenis printer ini sangat bagus kualitas cetakannya dibanding printer dot matrix dan inkjet. Sistem pencetakannya menggunakan infra merah melalui toner dengan menggunakan serbuk toner. Karena system cetak yang mirip dengan fotocopy, kualitas cetaknya berkecepatan tinggi.

BAB III

METODOLOGI PENELITIAN

3.1 Kerangka Pikir

Dalam penelitian ini penulis melakukan tahapan kegiatan sesuai dengan rencana kegiatan yang telah dibuat meliputi metedologi pengumpulan data dan tahapan pembagian aplikasi



Gambar 3.1 Kerangka Pikir

3.2 Penjelasan Skema Penelitian

3.2.1 Investigasi Awal

Investigasi dilakukan di Bengkel Moto Link, Jl. Raya Banjaran No.183, Kecamatan Baleendah Kabupaten Bandung. Ari Jacob merupakan pendiri Bengkel Moto Link. Investigasi awal dimulai dari melihat keadaan Bengkel Moto Link dan bagaimana rencana bisnis yang selama ini sudah dijalankan. Kemudian penulis melakukan wawancara, kendala yang ditemukan yakni karena bengkel ini tidak memiliki aplikasi yang dapat membantu dalam proses pelaporan data, dimana data tersebut selama ini masih dilakukan secara manual yaitu ditulis melalui faktur dan nota sebagai bukti transaksi dimana sering terjadinya kehilangan data tersebut.

3.2.2. Pengumpulan Data

Merupakan metode pengumpulan data/investigasi yang dilakukan dalam penelitian ini adalah :

1. Observasi

Observasi yang dilaksanakan di Bengkel Moto Link, Penulis melakukan Observasi pada :

Tanggal : Rabu, 02 April 2020

Tempat : Bengkel Moto Link

Alamat : Jl. Raya banjaran No. 183, Kec. Baleendah Kab. Bandung

Hasil dari Observasi yang sudah dilakukan adalah : SK Pembimbingan dan Kartu Bimbingan.

2. Wawancara

Wawancara yang dilaksanakan di Bengkel Moto Link, Penulis melakukan wawancara dengan Manager Bengkel Moto Link yang bernama Tri Hadi Kuswanto .dalam wawancara penulis mengajukan pertanyaan diantaranya sebagai berikut :

- a. Bengkel Moto Link belum memiliki aplikasi penjualan.
- b. Bengkel Moto Link membutuhkan aplikasi penjualan untuk keakuratan pelaporan data penjualan.
- c. Bengkel Moto Link berharap dengan adanya aplikasi ini maka pelaporan data dapat dilaporkan secara cepat dan tepat

3. Studi Pustaka

Mengumpulkan, membaca, mempelajari sumber-sumber tertulis, penelitian terdahulu yang pernah dilakukan sebelumnya, seperti skripsi, jurnal, dan modul-modul (tutorial), serta mencatat hal-hal penting yang berhubungan dengan pembahasan, yaitu pembahasan mengenai sistem informasi bimbingan skripsi online mahasiswa. Agar mendapat gambaran secara teoritis. Diantaranya adalah:

1. Skripsi : Sistem Informasi Penjualan Suku Cadang Berbasis Web Pada Distributor dan Bengkel AHASS (ASTRA HONDA AUTHORIZED SERVICE STATION), Oleh : Metty Amelia , Sistem Informasi, Mercu Buana, Jl. Raya Menteng No. 29 Jakarta Pusat.
2. Skripsi : Sistem Informasi Penjualan Sparepart Motor Secara Online Berbasis Web Pada CV. APE Spare Part, Oleh : Doddy Hermawan Putranto, Sistem Informasi, Universitas Pembangunan Nasional Veteran Jakarta, Jl. RS. Fatmawati Raya, Pd. Labu, Kec Cilandak, Kota Depok.
3. Perancangan dan Implementasi Aplikasi Web Sistem Penjualan Suku Cadangan Gondola Pada CV Yoga Perkasa Mandiri, Oleh: Nur Syamsiyah, Sistem Informasi, Universitas Darma Persada Jakarta Timur, Jl. Radin Inten II (Terusan Casablanca) Pondok Kelapa.
4. Perancangan E-Commerce Untuk Penjualan Spareparts Pada Bengkel ABM Motor, Oleh : Wisnu Prabowo, Sistem Informasi, Universitas Budi Luhur, Jl. Raya Ciledug, Pertukangan Utara, Kebayoran Lama, Jakarta Selatan 122

3.2.3. Analisis Masalah

Pada tahap ini, penulis melakukan analisis terhadap kegiatan yang sedang berjalan. Kegiatan tersebut selalu dijalankan oleh Bengkel Moto Link yaitu adanya kendala pada proses pelaporan data yang sering kali tidak sesuai dikarenakan sering hilangnya faktur atau nota hasil dari transaksi .

3.2.4. Analisis Kebutuhan Fungsional Sistem

Kebutuhan fungsional merupakan kebutuhan system dari segi fungsionalitas yang akan muncul pada system yang dirancang. Adapun

beberapa kebutuhan fungsional yaitu :

1. Mengolah data spare part
2. Mengolah konfirmasi transaksi
3. Mengolah konfirmasi pembayaran

3.2.5. Analisis Kebutuhan Non Fungsional Sistem

Analisis kebutuhan merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan aplikasi. Spesifikasi ini juga juga meliputi semua elemen dan komponen yang dibutuhkan untuk aplikasi yang akan dibuat, sampai dengan aplikasi tersebut diimplementasi. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang dibutuhkan oleh aplikasi, keluaran yang akan dihasilkan oleh aplikasi dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan. Kebutuhan aplikasiterbagi menjadi beberapa analisis, yaitu sebagai berikut :

1. Analisis Perangkat Keras (*Hardware*)

Spesifikasi perangkat keras yang digunakan untuk Aplikasi Penjualan SpareParttersebut adalah seperangkat komputer dengan spesifikasi sebagai berikut:

- 1) Processor Intel ® Core(TM) i7-7700HQ CPU
- 2) System Type 64-bit Operating System, x64-based processor
- 3) Memory berkapasitas 16 GB

2. Analisis Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan untuk membangun Penjualan Spare Part Motor tersebut adalah sebagai berikut :

- i. Sistem Operasi Window 10 Pro
- ii. Database Management System (DBMS) MySQL Xampp 7.
- iii. Web Browser Mozilla Firefox.
- iv. Framework Code Igniter.
- v. Argo UML
- vi. Balsamiq Mockups 3
- vii. Sublime Text

viii. Microsoft Visio 2013

3.2.6. Konseptual Design

Pada tahap ini, penulis membuat desain dari hasil analisis yang telah dilakukan. Ada beberapa proses desain yang akan dilakukan. Untuk lebih jelasnya dapat dilihat sebagai berikut :

1. Perancangan Modifikasi

Pada perancangan model ini dilakukan untuk mengetahui alur yang akan dibuat untuk membangun sistem informasi. Perancangan model akan di buat dengan menggunakan *UML (Unified Modeling Language)*. Perancangan menggunakan *UML* dilakukan supaya perancangan aplikasi yang dikerjakan akan lebih mudah, maka didapat perancangan design sebagai berikut :

- a) *Usecase Diagram* digunakan untuk keterhubungan *actor* dan *usecase* dalam aplikasi yang akan dibuat, *actor* dan *usecase* yang terlibat pada penelitian ini adalah :
 - i. *Actor* : Admin, Kasir
 - ii. *Usecase* :
 1. Admin : Menambahkan Spare Part, Kelola Kasir, Kelola Data, Harga Spare Part, Laporan.
 2. Kasir : Mengelola Transaksi
- b) *Class Diagram* digunakan untuk menggambarkan *attribute, operation, constraint* pada sistem yang akan dibuat. *Class Diagram* yang akan dibuat adalah sebagai berikut :
 - i. *Class Diagram* Admin
 - ii. *Class Diagram* Konsumen
 - iii. *Activity Diagram* digunakan untuk memodelkan alur kerja dari sistem dan aktivitas dari *actor* dalam aplikasi. Adapun diagram alur yang akan dibuat diantaranya :
 1. Admin : Menambahkan, transaksi, mengedit, menghapus data Spare Part, data kasir, laporan.

2. Kasir : Mengelola data penjualan dan pembayaran.

2. Database Design

Data adalah salah satu hal utama yang dibutuhkan dalam membangun suatu sistem. Pengelola data juga yang menjadi tujuan dingunkannya suatu agar pengelola perusahaan dapat lebih mudah mengetahui perkembangan perusahaan. Dalam perancangan database, data yang dibutuhkan sebelumnya telah didapatkan pada proses pengumpulan data dan sudah di analisis. Perancangan database menggunakan *Class Diagram*. Beberapa data yang digunakan adalah :

- a. Kelola Ubah Edit Tambah Data Dashboard
- b. Kelola Ubah Edit Tambah Data Sparepart
- c. Kelola Ubah Edit Tambah Data Kategori
- d. Kelola Ubah Edit Tambah Data User
- e. Kelola Ubah Edit Tambah Data Laporan

3. User Database Design

Perancangan antar muka, merupakan perancangan-perancangan terakhir dalam tahap design pengembangan sistem. Mendesain UI (User interface) atau tampilan program secara keseluruhan yang terdiri dari Menu, Sub Menu, Splash, dan lain-lain. Perancangan User Interface dengan menggunakan Balsamiq MockUps

3.2.7. Pembuatan Website

Dalam pembuatan aplikasi pada Bengkel Moto Link adalah sebagai berikut :

1. MySQL, digunakan untuk penyimpanan database dari setiap data yang dibutuhkan.
2. XAMPP, digunakan untuk sebagai Control Panel.
3. Notepad++, digunakan sebagai text editor.
4. Code Igniter, digunakan sebagai Framework pembuatan sistem informasi bimbingan skripsi online.

5. Mozilla Firefox, browser sebagai pembuka aplikasi web.
6. OS Window, digunakan penulis dalam pembuatan website adalah dari produk windows khususnya Microsoft Window 10 Pro 64-bit.

3.2.8. Testing

Sebelum implementasi dilakukan maka diawali melakukan penelitian terlebih dahulu terhadap aplikasi yang sudah dibuat. Pengujian terhadap aplikasi yang sudah dibuat, yaitu dilakukan pengujian dengan menggunakan metode *black box* yaitu untuk menguji fungsionalitas dari suatu aplikasi. Dalam pengujian melalui beberapa tahapan :

1. Tahap Pertama

Pengujian Pertama, pengujian pertama yaitu *graph-based* memahami objek yang terdapat dalam perangkat lunak dan menentukan hubungan yang dimiliki antara objek-objek tersebut.

2. Pengujian Kedua,

pengujian kedua yaitu *Equivalence Partioning* yang metode yang membagi domain input dari suatu program ke dalam kelas data

3. Pengujian Ketiga

pengujian ketiga *Graph Based Testing* yang melengkapi partisi *ekuivalensi* dengan focus pada domain *output*.

3.2.9 Pembuatan Laporan

Tahapan terakhir adalah pembuatan laporan sebagai salah satu persyaratan kelulusan. Laporan disusun sesuai dengan ketentuan yang tercantum dalam Pedoman Penulisan Skripsi Fakultas Teknologi Informasi Universitas Bale Bandung.

BAB IV

ANALISIS DAN PERANCANGAN

4.1 Analisis

4.1.1 Analisis Sistem

Analisis system adalah sistem informasi penjualan yang sedang berjalan di bengkel Moto Link. Analisis ini dapat disimpulkan sebagai penguraian dari suatu sistem yang dibuat untuk mempermudah mengatasi permasalahan yang terjadi, memenuhi kebutuhan yang diharapkan. Pada tahap analisis sistem dilakukan dengan tepat supaya diketahui bagaimana sistem yang sedang berjalan dan yang akan dibuat.

4.1.2 Analisis Masalah

Analisis permasalahan akan diuraikan dengan menggunakan kerangka PIECES :

Tabel 4.1 Analisis Masalah Dengan Metode PIECES

PIECES	Hasil Analis Terhadap Sistem Lama	Hasil Analisis Terhadap Sistem Baru
Performance (Kinerja)	Pencatatan dan transaksi pembayaran serta penjualan spare part masih dengan cara manual	Mempercepat dan mempermudah proses pembayaran transaksi
Information (Informasi)	Pencarian data sparepart dilakukan secara manual dalam pencatatan buku	Mempercepat dan mempermudah proses pencarian sparepart yang dibutuhkan
Economy (Ekonomi)	Harga sparepart terkadang tidak sesuai	Harga sparepart dapat diatur dengan baik
Control (Pengendalian)	Sering terjadinya kehilangan faktur	Data faktur faktur tersimpan dengan baik
Efisiensi (Efisiensi)	Memakan waktu untuk pembuatan laporan	Laporan mudah untuk dibuat
Service (Layanan)	Faktur lama dibuat	Faktur cepat dibuat

4.1.3 Analisis Kebutuhan

1. Analisis Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berisi layanan apa saja yang nantinya ditampilkan pada sistem.

- 1) Melakukan edit data sparepart, tambah data sparepart dengan mengisi nama sparepart, kategori, harga jual, dan hapus sparepart, cetak data sparepart
- 2) Mengelola data kategori sparepart
Mengisi kategori sparepart dengan menambah kategori sparepart, mengedit dan menghapus kategori sparepart
- 3) Mengelola data kasir
Menambah kasir dengan mengisi nama kasir dan kata sandi lalu menyimpan data, mengedit dan menghapus kasir
- 4) Mengelola transaksi
Menambahkan transaksi baru, menyimpan dan mencetak transaksi
- 5) Mengelola data laporan kas
Menampilkan laporan kas, mencetak laporan kas
- 6) Mengelola data laporan keuntungan
Menampilkan laporan keuntungan, mencetak laporan keuntungan

2. Analisis Kebutuhan Non-Fungsional

Kebutuhan Non Fungsional adalah suatu batasan layanan atau fungsi yang ditawarkan sistem, menggambarkan standarisasi pada sistem. Analisis kebutuhan merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi semua elemen dan komponen yang dibutuhkan untuk sistem yang akan dibangun, sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang dibutuhkan oleh sistem, keluaran yang akan dihasilkan oleh sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan. Kebutuhan sistem terbagi menjadi beberapa analisis, yaitu sebagai berikut :

1. Analisis Perangkat Keras (Hardware)

Spesifikasi perangkat keras yang digunakan untuk membangun Sistem Informasi Penjualan Sparepart tersebut adalah dengan spesifikasi sebagai berikut:

1. Processor Intel ® Core(TM) i7-7700HQ CPU
2. System Type 64-bit Operating System, x64-based processor
3. Memory berkapasitas 16 GB
4. Printer Epson L120

2. Analisis Perangkat Lunak (Software)

Perangkat lunak yang digunakan untuk membangun Aplikasi Penjualan Sparepart Motor tersebut adalah sebagai berikut :

5. Sistem Operasi Window 10 Pro
6. Database Management System (DBMS) MySQL Xampp 7.
7. Web browser Mozilla Firefox.
8. Framework CodeIgniter.
9. Argo UML
10. Balsamiq Mockups 3
11. Notepad ++
12. Microsoft Visio 2013

Uraian Tugas

Tugas dan tanggung jawab (jobdesk) tiap bagian struktur organisasi Bengkel Moto Link yang sedang berjalan saat ini sebagai berikut :

1. Pemilik (Manager)

- a. Mengontrol segala bentuk kegiatan usaha baik transaksi pembelian maupun penjualannya.
- b. Mengontrol persediaan sparepart apakah cukup stok untuk perhari atau perbulan.

2. Montir

- a. Melakukan perbaikan atau penggantian spare part

3. Kasir

- a. Menjalankan proses penjualan dan pembayaran

- b. Melakukan pencatatan atas semua transaksi
- c. Membantu pelanggan dalam memberikan informasi mengenai suatu sparepart

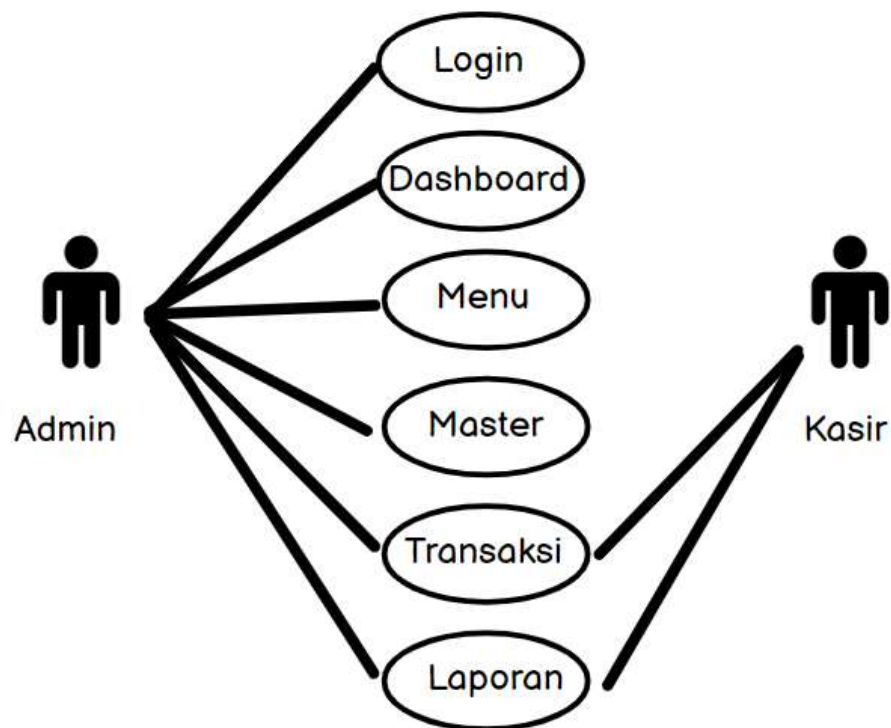
4.2 Perancangan

Perancangan sistem perangkat lunak dideskripsikan dengan model analisis menggunakan diagram Use Case. Analisis digunakan untuk pemetaan awal mengenai sistem yang akan dilakukan perancangan.

4.2.1 Pemodelan sistem

A. Usecase diagram

Usecase diagram yang diusulkan adalah sebagai berikut:



Gambar 4.1 Usecase diagram sistem yang diusulkan

Diagram Usecase diagram menunjukkan *actor* yang masing-masing mempunyai hak sebagai berikut:

Tabel 4.2 definisi *actor* dan deskripsi sistem yang diusulkan

No	Aktor	Deskripsi
1	Admin	Orang atau pengguna yang bisa mengelola data
2	Kasir	Orang atau pengguna yang bertugas untuk melakukan operasi transaksi penjualan

Tabel 4.3 definisi dan deskripsi sistem yang diusulkan

No	Aktor	Deskripsi
1	Validasi	Proses pengecekan hak akses Siapa yang diperbolehkan untuk melakukan operasi pengolahan data kasir dan <i>admin</i> , data laporan yang bisa diakses diantaranya admin dan kasir
2	Login	Proses untuk melakukan login agar mendapatkan hak akses tertentu dalam usecase ini.
3	Log out	Proses untuk melakukan <i>logout</i> agar <i>login</i> yang sebelumnya dilakukan akan dilepaskan.
4	Dashboard	Merupakan proses dimana tampilan awal <i>website</i> yang menampilkan <i>sidebar</i> yang tersedia
5	Kategori menu	Merupakan proses dimana admin dapat menambah, edit, maupun hapus data dan menyimpannya ke database.
6	Transaksi	Proses transaksi pemesanan menu yang kemudian diproses untuk menjumlahkan berapa total pembayaran
7	Laporan	Proses untuk melihat transaksi penjualan yang berjalan dengan rekap laporan perhari dan perbulan
8	Mencetak laporan	Merupakan proses Rekapitulasi data keseluruhan

Tabel 4.4 Skenario Login

Aksi Aktor	Reaksi Sistem
Skenario sukses	
1. Memasukkan username dan password	
	2. Mengecek valid tidaknya data masukan
	3. Masuk ke halaman beranda admin
Aksi Aktor	Reaksi Sistem
Skenario Gagal	
1. Memasukkan username dan password	
	2. Mengecek valid tidaknya data masukan
	3. Menampilkan pesan login tidak valid

Tabel 4.5 Skenario Tampil Beranda Admin

Aksi Aktor	Reaksi Sistem
1. Melihat halaman beranda admin	
	2. Menampilkan halaman beranda admin

Tabel 4.6 Skenario Pengelolaan Data Menu

Aksi Aktor	Reaksi Sistem
1. Memilih master	
	2. Menampilkan Tabel data master
3. Memilih tombol tambah sparepart	
	4. Menampilkan form data sparepart
5. Mengisi data sparepart	
6. Memilih tombol simpan	
Skenario Edit Data Menu	
Memilih master data sparepart	
	Menampilkan Tabel data sparepart
Memilih tombol edit	
	Menampilkan form edit data sparepart
Mengedit form data sparepart	
Memilih tombol simpan	
Skenario hapus data menu	
Memilih master data sparepart	
	Menampilkan Tabel data sparepart
Memilih tombol hapus	

Tabel 4.7 Skenario Pengelolaan Transaksi

Aksi Aktor	Reaksi Sistem
Memulai transaksi baru	
	Menampilkan tabel transaksi
Memilih tombol tambah transaksi	
	Menampilkan form transaksi
Mengisi jumlah beli	
	Menampilkan jumlah transaksi
Memilih tombol simpan	
Skenario Hapus Data Transaksi	
Memilih master data menu	
	Menampilkan Tabel data menu
Memilih tombol hapus	

Tabel 4.8 Skenario Pengelolaan Kasir

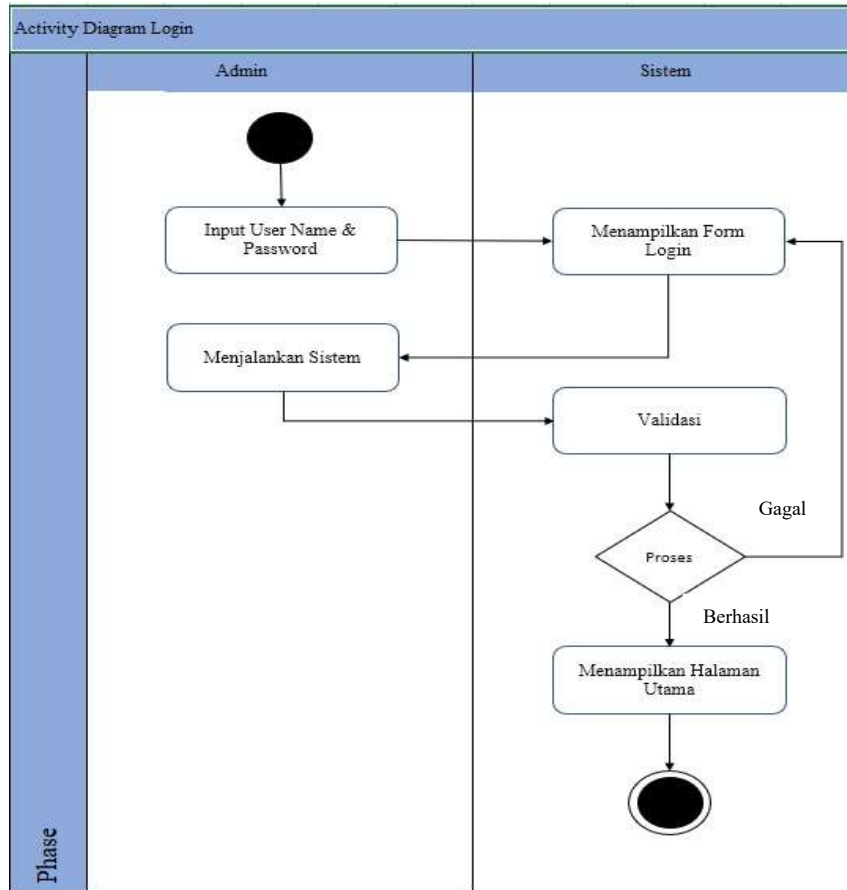
Aksi Aktor	Reaksi Sistem
Tambah kasir	
	Menampilkan table tambah kasih
Mengisi username dan password pilih simpan	
	Menampilkan Tabel data jumlah kasir
Skenario Edit Data Kasir	

Memilih data kasir	
	Menampilkan Tabel data kasir
Memilih tombol edit	
	Menampilkan form edit data kasir
Skenario Hapus Data Kasir	
Memilih data kasir	
	Menampilkan data kasir
Memilih tombol hapus	

Table 4.9 Skenario Pengelolaan Laporan

Aksi Aktor	Reaksi Sistem
Memilih tombol jenis cetak laporan	
	Menampilkan tabel cetak laporan
Memilih cetak laporan harian/bulanan	
	Menampilkan tabel cetak laporan harian/bulanan

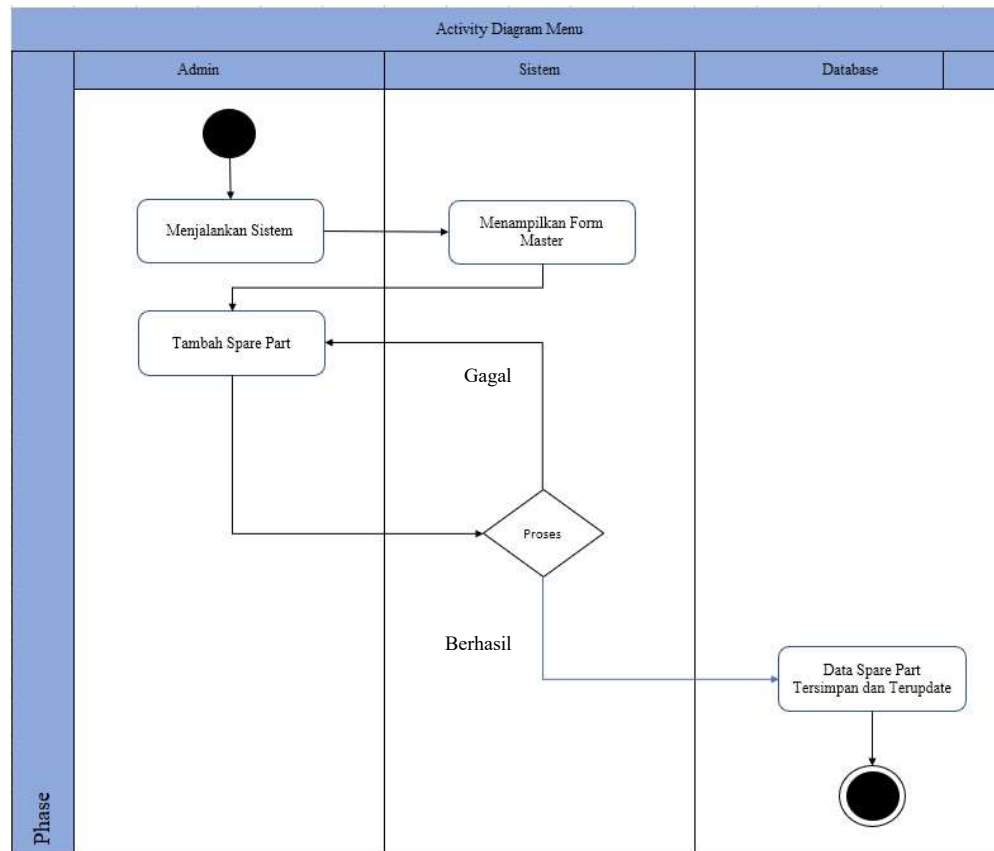
B. Activity diagram sistem yang diusulkan



Gambar 4.2 Activity diagram login yang diusulkan

Berikut ini adalah penjelasan dari gambar diatas yang mempunyai fungsi sebagai berikut :

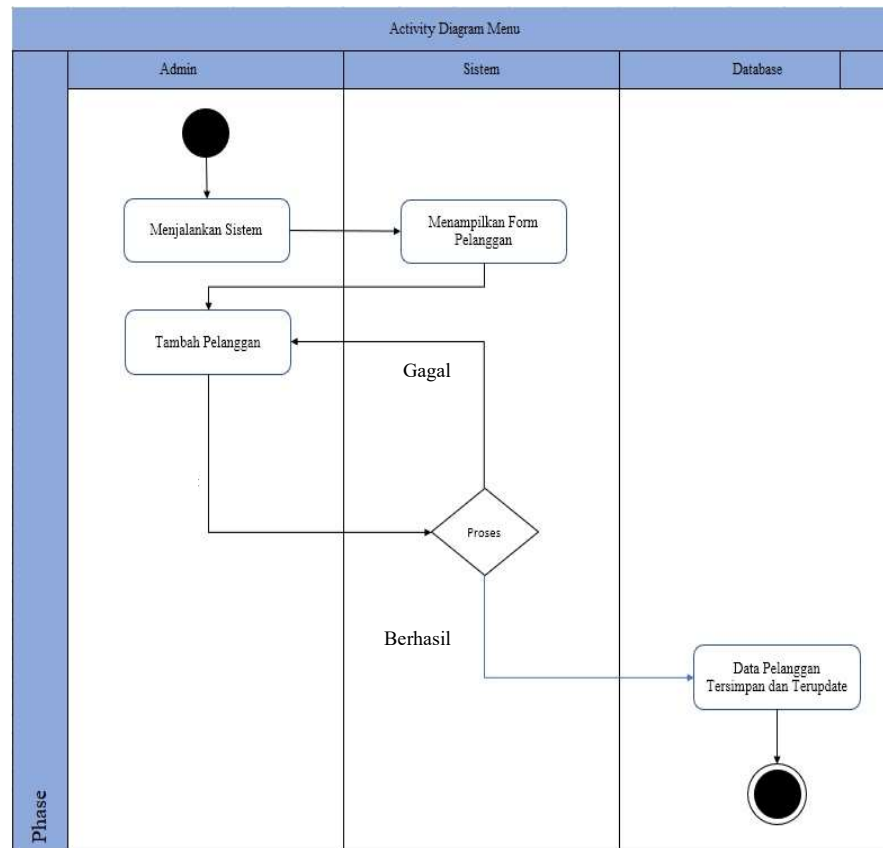
- 1) User melakukan proses login.
- 2) Lalu setelah melakukan login aplikasi menampilkan form login dan memilih login sebagai kasir atau admin.
- 3) jika berhasil login sesuai *Username* dan *Password* maka sistem akan menampilkan halaman utama atau *Dashboard*
- 4) jika dalam proses login tidak valid maka sistem tidak akan menampilkan halaman utama atau *Dashboard*.
- 5) selesai.



Gambar 4.3 activity diagram master sparepart yang diusulkan

Berikut ini adalah penjelasan dari gambar diatas yang mempunyai fungsi sebagai berikut:

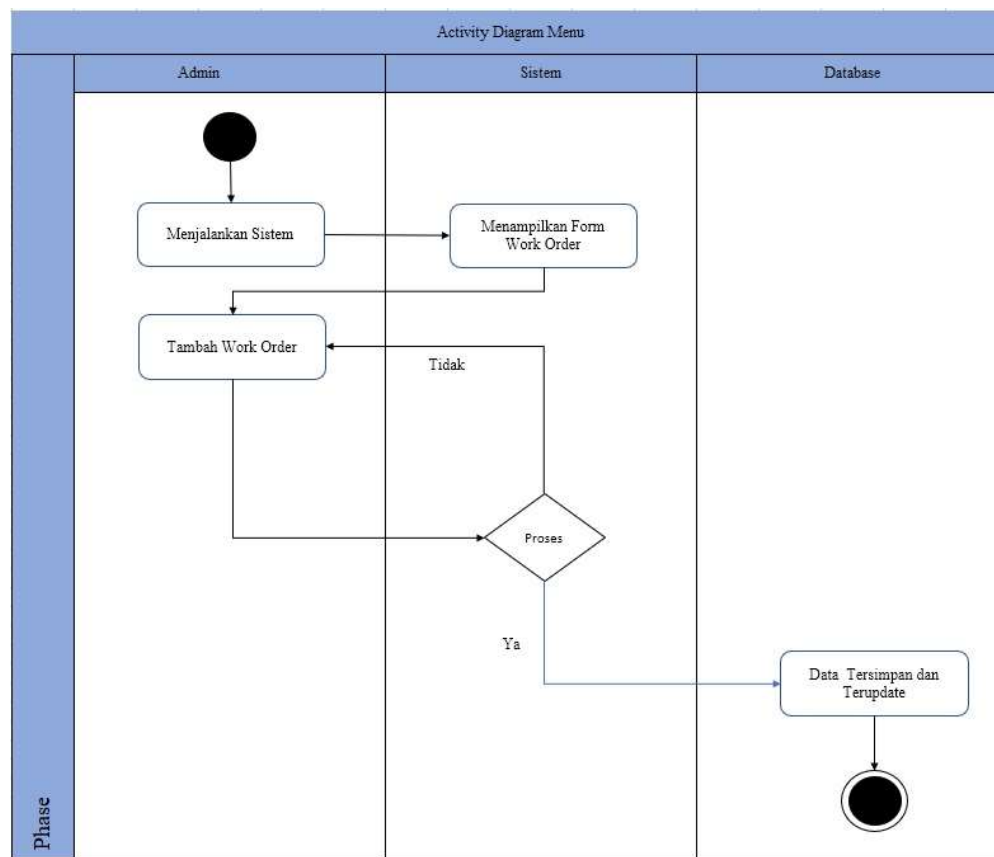
- 1) admin melakukan proses login.
- 2) Lalu setelah melakukan login aplikasi, admin memilih form master, kemudian sistem menampilkan form master, dan kemudian admin memilih tombol tambah
- 3) aplikasi menu menampilkan form tambahan sparepart yang digunakan untuk menambahkan sparepart yang akan ditambahkan.
- 4) setelah itu admin menginputkan sparepart yang ingin ditambahkan.
- 5) setelah itu sparepart tersimpan.
- 6) selesai



Gambar 4.4 Activity diagram Kategori Pelanggan

Berikut ini adalah penjelasan dari gambar diatas yang mempunyai fungsi sebagai berikut:

- 1) Admin melakukan proses login.
- 2) Lalu harus setelah melakukan login aplikasi, admin mini form tambahan pelanggan, kemudian sistem menampilkan tabel pelanggan, dan kemudian admin memilih tombol tambahkan pelanggan
- 3) Aplikasi menampilkan form data tambahan pelanggan yang akan ditambahkan
- 4) Setelah itu admin menginputkan pelanggan yang ingin ditambahkan.
- 5) Setelah itu data pelanggan terupdate dan tersimpan
- 6) Selesai



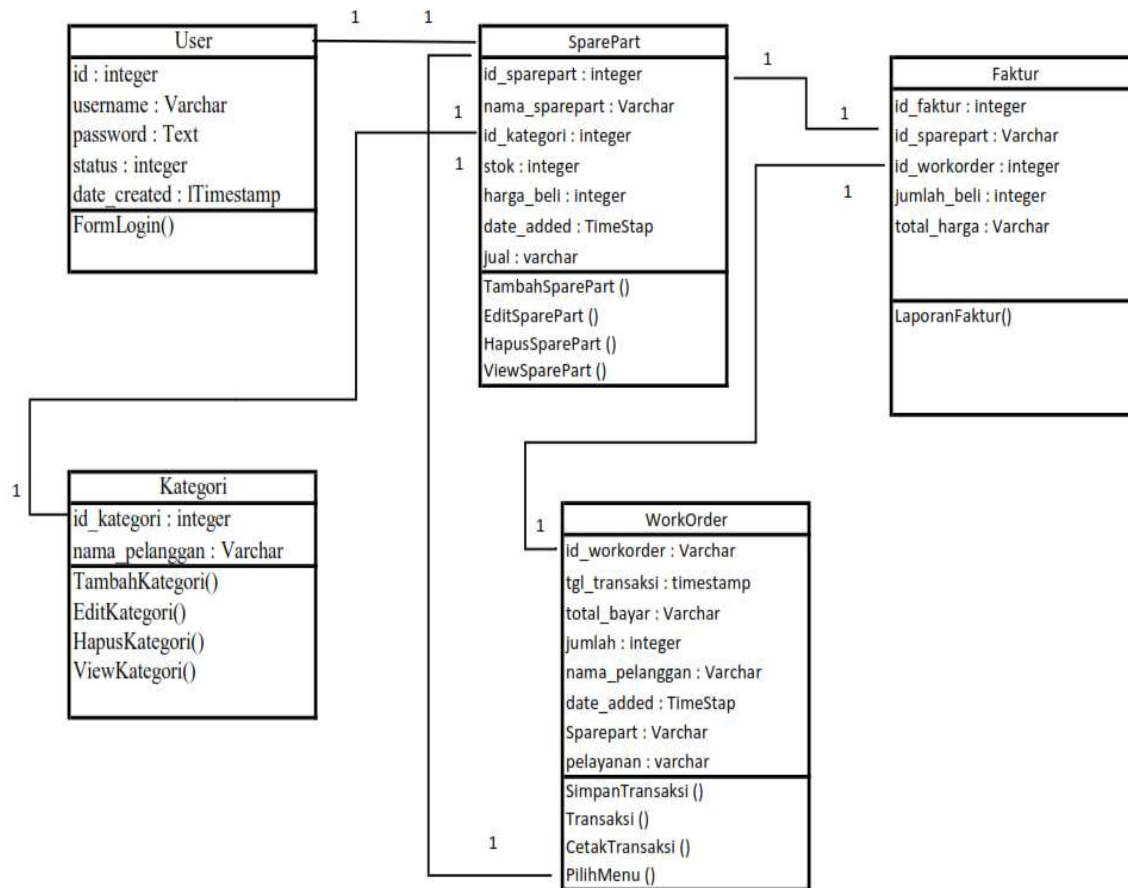
Gambar 4.5 activity diagram kasir yang diusulkan

Berikut ini adalah penjelasan dari gambar diatas yang mempunyai fungsi sebagai berikut:

- 1) Admin melakukan proses login
- 2) Lalu setelah melakukan login aplikasi, login menggunakan id dan password kasir
- 3) Aplikasi menampilkan form tambahkan work order yang digunakan untuk menambahkan transaksi yang akan ditambahkan
- 4) Setelah itu admin meng-input-kan kasir yang ingin ditambahkan
- 5) Setelah itu menu tersimpan
- 6) Selesai

C. Class Diagram

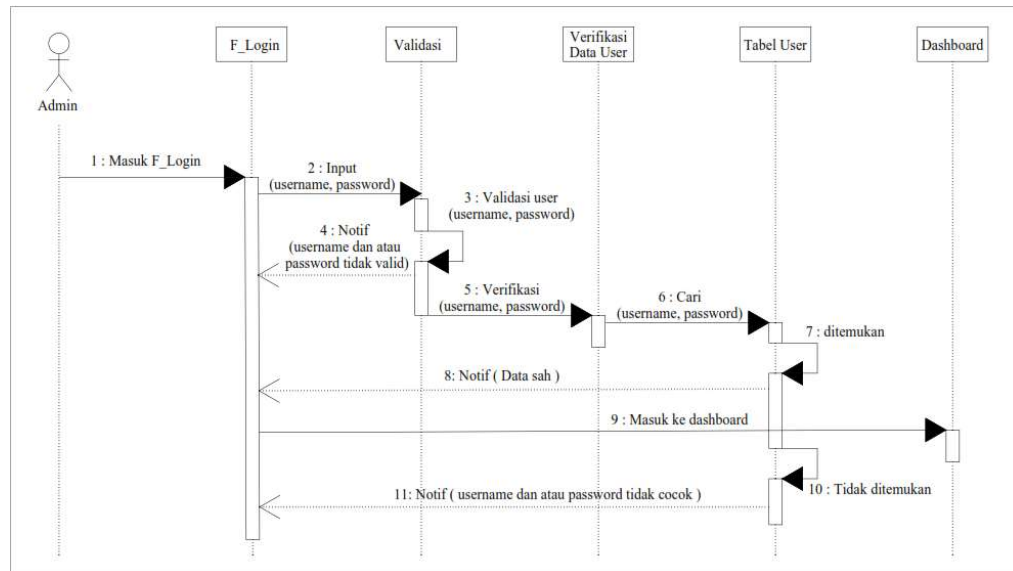
Class Diagram merupakan diagram untuk menampilkan beberapa kelas yang ada dalam aplikasi yang menunjukkan hubungan antara kelas dalam aplikasi yang akan dibangun. Berikut digambar *class diagram* dari sistem informasi penjualan.



Gambar 4.6 class diagram

F. Sequence Diagram

a. Sequence Diagram Login

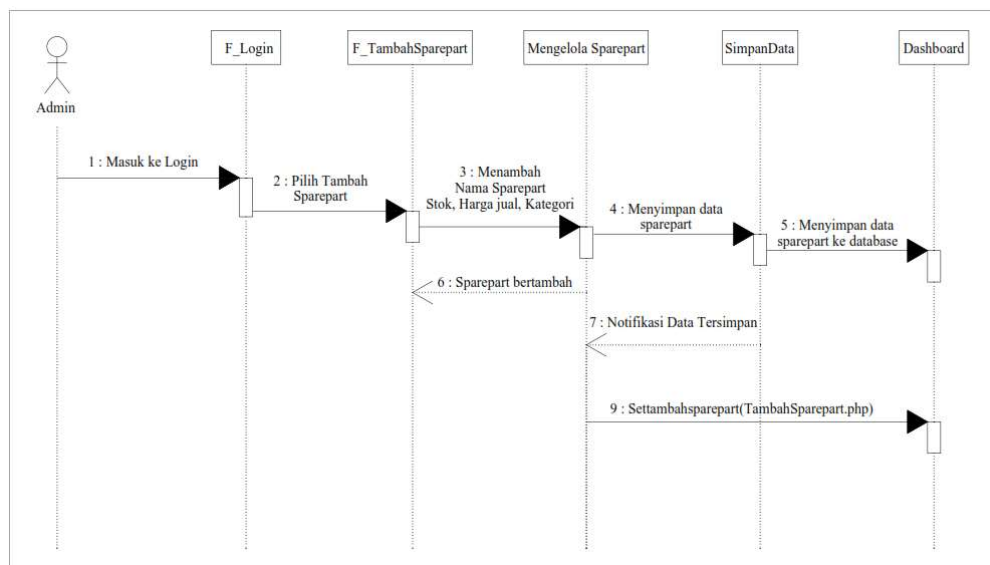


Gambar 4.7 *Sequence Diagram Login*

Berikut ini penjelasan dari *Sequence Diagram Login*:

- 1) User masuk ke form login
- 2) User memasukkan username dan password pada form login.
- 3) Aplikasi melakukan validasi username dan password.
- 4) Jika username atau password tidak valid maka akan menampilkan pesan username dan password tidak valid.
- 5) Jika username dan password maka aplikasi akan melakukan verifikasi data.
- 6) Username dan password dicari pada table user
- 7) Jika username dan password ditemukan.
- 8) Maka akan muncul notifikasi data sah.
- 9) Lalu masuk ke dashboard
- 10) Jika username password tidak ditemukan
- 11) Muncul username dan atau password tidak cocok.

b. Sequence Diagram Tambah Sparepart

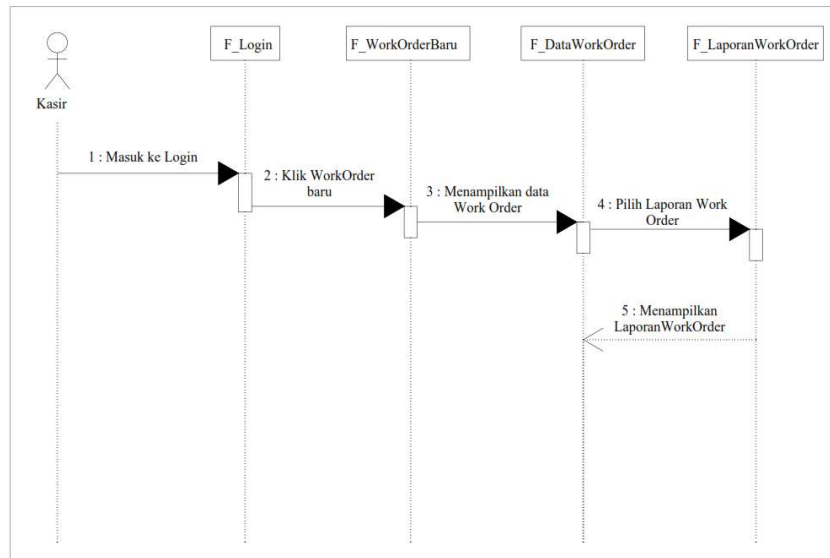


Gambar 4.8 *Sequence Diagram* Tambah Sparepart

Berikut ini adalah penjelasan dari *Sequence Diagram* Tambah Menu:

- 1) Username memilih tombol tambah menu baru pada form login.
- 2) Kemudian memasukkan data user pada form tambahan menu baru.
- 3) Kemudian pilih tambah.
- 4) Aplikasi melakukan validasi data menu
- 5) Jika data tidak valid maka akan menampilkan pesan data tidak valid.
- 6) Jika data user valid maka aplikasi akan melakukan verifikasi data menu.
- 7) Jika data menu sudah maka aplikasi akan menyimpan data menu dan menampilkan pesan menu berhasil ditambahkan.

c. Sequence Diagram Transaksi



Gambar 4.9 *Sequence Diagram* Transaksi

Berikut ini adalah penjelasan dari *Sequence Diagram* Transaksi:

- 1) Kasir memilih tombol tambah transaksi pada form transaksi
- 2) Kemudian memasukkan data transaksi pada form tambahan transaksi
- 3) Kemudian pilih nama menu, dan jumlah beli
- 4) Aplikasi melakukan validasi data
- 5) Jika data tidak valid maka akan menampilkan pesan data tidak valid, jika valid maka menampilkan data berhasil disimpan
- 6) Kemudian proses transaksi akan menampilkan laporan transaksi

4.3 Basis Data

Perancangan basis data merupakan perancangan yang digunakan untuk membuat dan menyimpan data ke dalam sistem yang terdiri dari beberapa file data. Pada perancangan ini basis data yang akan dibahas menggunakan struktur file.

1. Struktur File Data Menu

Tabel 4.10 File Data Menu

No	Nama Field	Type	Length	Keterangan
1	Id_sparepart	Integer	11	Primary key
2	Nama_sparepart	Varchar	100	
3	Id_kategori	Integer	11	
4	Stock	Integer	11	
5	Harga_jual	Varchar	100	
6	Date_added	Date		

2. Struktur File Data Kategori

Tabel 4.11 File Data Kategori

No	Nama Field	Type	Length	Keterangan
1	Id_kategori	Integer	11	Primary key
2	Nama_kategori	Varchar	100	

3. Struktur File Transaksi

Tabel 4.12 File Data Kategori

No	Nama Field	Type	Length	Keterangan
1	Id_transaksi	Integer	11	Primery key
2	Tgl_transaksi	Timestamp		
3	Kode_kasir	Integer	11	
4	Total_bayar	Varchar	20	
5	No_workorder	Varchar	20	
6	Nama_kategori	Varchar	20	

4. User

Tabel 4.13 File Data User

No	Nama Field	Type	Length	Keterangan
1	Id	Integer	11	Primery key
2	Username	Varchar	100	

3	Password	Text		
4	Status	Integer	22	
5	Date_created	Date		

5. Data Transaksi

Tabel 4.14 File Data Transaksi

No	Nama Field	Type	Length	Keterangan
1	Id_subtransaksi	Integer	11	Primery key
2	Id_menu	Integer	11	
3	Id_transaksi	Integer	11	
4	Jumlah_beli	Varchar	20	
5	No_invoice	Varchar	20	

4.4 Desain

1. Tampilan login admin

A Web Page

https://

sparepart management application

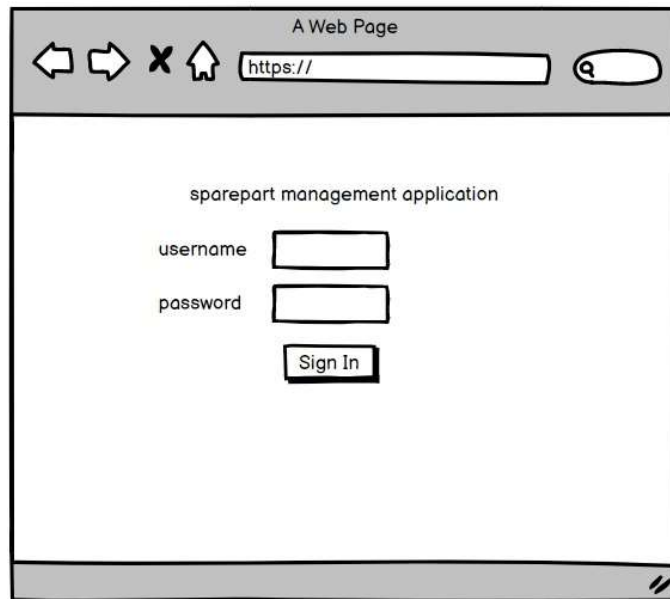
username

password

Gambar 4.10 Desain Input Admin Halaman Login

Tampilan gambar di atas merupakan tampilan login untuk login kedalam system, jika username dan password masih salah maka admin tidak diizinkan untuk masuk kedalam sistem dan akan terus ditampilkan form login tersebut jika username dan password benar maka dapat mengakses sistem tersebut.

2. Tampilan Login Kasir



A Web Page

https://

sparepart management application

username

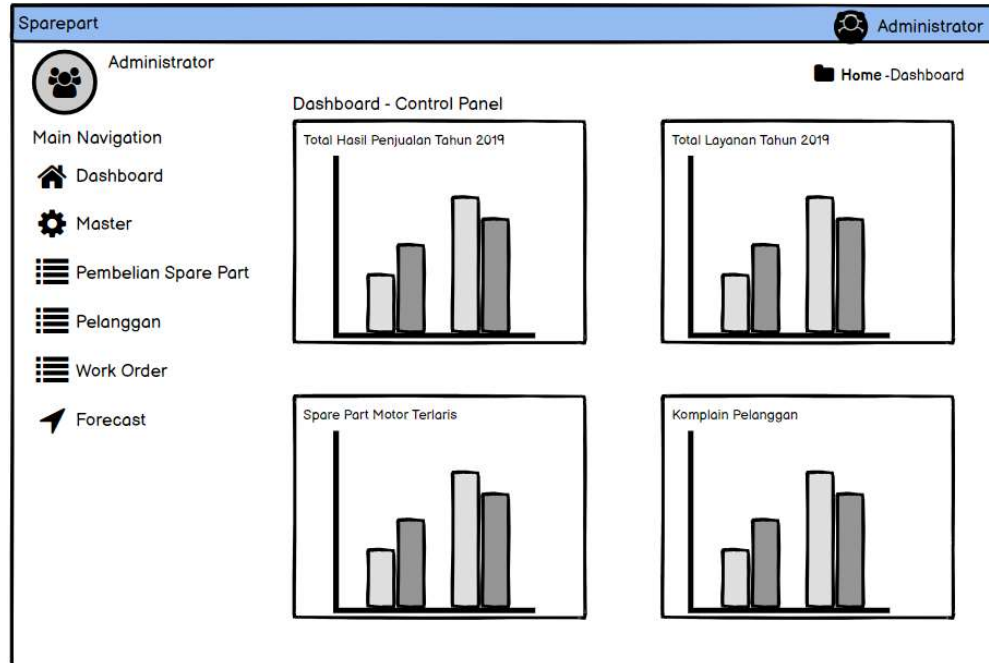
password

Gambar 4.11 Desain Input Kasir Halaman Login

Tampilan gambar di atas merupakan login untuk login ke dalam sistem jika username dan password masih salah maka admin tidak diizinkan untuk masuk kedalam sistem dan akan terus ditampilkan form login tersebut jika username dan password benar maka dapat mengakses sistem tersebut.

3. Tampilan Dashboard

Tampilan gambar dibawah merupakan dasboard untuk menampilkan beranda dan menu bar yang ada pada website tersebut. Menampilkan login sebagai admin, waktu, dan data menu titik data laporan yang ada dalam sistem tersebut.



Gambar 4.12 Desain Dashboard

4. Tampilan Sparepart

No	nama	kategori	jenis kendaraan	Harga Jual	stok	Satuan	Aksi
1	Repsol Diesel SHFD 15/40 Liter	oli	mini bus	65000	18	Liter	
2	Repsol Super SHFD 15/40 Liter	oli	Sport	85000	27	Liter	
3	Minyak Rem Bosch	minyak rem	van	25000	15	Buah	

Gambar 4.13 Desain Sparepart

Tampilan gambar di atas merupakan tampilan sparepart yang berfungsi menambah dan juga menampilkan sparepart yang akan ditampilkan. Proses ini dilakukan oleh admin yang dapat mengedit, menambahkan, dan menghapus menu.

5. Tampilan Tambahan Menu Sparepart

The screenshot shows a web application interface for managing spare parts. The top bar is labeled 'Sparepart' and 'Administrator'. The sidebar on the left lists navigation options: Dashboard, Master (expanded), Feedback, Jenis, Kategori, Merk, Satuan, Spare Part, Tipe, Pembelian Spare Part, Pelanggan, Work Order, and Forecast. The main content area is titled 'Tambah Data' and contains a form with the following fields:

- Nama: Text input field
- Kategori: Dropdown menu
- Jenis: Dropdown menu
- Satuan: Dropdown menu
- harga Jual: Text input field

At the bottom of the form are two buttons: 'Simpan' (Save) and 'Batal' (Cancel). To the right of the form, there is a table with columns 'Satuan' and 'Aksi' (Action). The table contains three rows of data:

Satuan	Aksi
Liter	[Edit] [Delete]
Liter	[Edit] [Delete]
Buah	[Edit] [Delete]

Gambar 4.14 Desain Tambah Sparepart

Tampilan gambar di atas merupakan tampilan tambah menu berfungsi menambah dan juga menampilkan menu yang akan ditampilkan. Proses ini dilakukan oleh admin yang dapat mengedit, menambahkan, dan menghapus menu.

6. Tampilan Kategori Sparepart

The screenshot shows the 'Kategori Sparepart' view in the application. The top bar is labeled 'Sparepart' and 'Administrator'. The sidebar on the left lists navigation options: Dashboard, Master (expanded), Feedback, Jenis, Kategori (highlighted), Merk, Satuan, Spare Part, Tipe, Pembelian Spare Part, Pelanggan, Work Order, and Forecast. The main content area displays a table with the following data:

No	nama	Aksi
1	oli	[Edit] [Delete]
2	minyak rem	[Edit] [Delete]
3	Filter	[Edit] [Delete]









Gambar 4.15 Desain Kategori Sparepart

Tampilan gambar di atas merupakan tampilan kategori sparepart yang berfungsi menambah dan juga menampilkan kategori menu yang akan ditampilkan. Proses ini dilakukan oleh admin yang dapat mengedit, menambahkan, dan menghapus. Menampilkan kategori dari menu yang tersedia.

No	nomor	tanggal masuk	tanggal keluar	keluhan	montir	keterangan	Aksi
1	WO0001	01-07-2020 09:01:51	-	servis	ali	Ganti Rem	
2	WO0002	01-07-2020 08:52:28	-	servis	ali	Ganti Oli	
3	WO0003	01-07-2020 07:52:28	-	servis	ali	ganti lampu	




Gambar 4.16 Desain Work Order

Tampilan gambar di atas merupakan tampilan untuk melalui work order dengan meng-klik pada tombol transaksi kemudian memasukkan data sparepart yang dibeli oleh pelanggan melalui proses transaksi ditampilkan pada *table* transaksi.

Sparepart							
Kasir							
Main Navigation		Sedang Berlangsung					
<ul style="list-style-type: none"> Work Order Faktur 		<div>Home - Dashboard</div> <div>+ Tambah</div>					
No	nomor	tanggal masuk	tanggal keluar	keluhan	montir	keterangan	
1	WO0003	03-07-2020 09:01:51	-	servis	ali	Ganti Rem	  
2	WO0004	03-07-2020 08:52:28	-	servis	ali	Ganti Oli	  
Transaksi Berhasil							
No	nomor	tanggal masuk	tanggal keluar	keluhan	montir	keterangan	
1	WO0001	01-07-2020 09:01:51	02-07-2020 09:01:51	servis	ali	Ganti Rem	
2	WO0002	01-07-2020 08:52:28	02-07-2020 08:52:28	servis	ali	Ganti Oli	

Gambar 4.17 Desain Transaksi Kasir

Tampilan gambar diatas merupakan proses work order untuk melakukan data transaksi dan mengetahui berapa total yang harus dibayar oleh pembeli. Data transaksi yang dikelola oleh kasir.

Sparepart							
Kasir							
Main Navigation		Faktur					
<ul style="list-style-type: none"> Work Order Faktur 		<div>Home - Dashboard</div> <div> Print</div>					
No	nomor	tanggal faktur	tanggal cetak	total	montir	keterangan	aksi
1	FA0001	01-07-2020	02-07-2020	65000	ali	Ganti Rem	
2	FA0002	01-07-2020	02-07-2020	65000	ali	Ganti Oli	

Gambar 4.18 Desain Laporan Faktur Oleh Kasir

BAB V

IMPLEMENTASI DAN PENGUJIAN

5.1 Implementasi

Setelah melakukan perancangan sistem informasi, maka tahapan selanjutnya adalah implementasi. Implementasi merupakan tahapan meletakkan aplikasi hingga siap untuk digunakan. Pada tahapan ini bertujuan untuk mengkonfirmasi modul-modul perancangan.

5.1.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan adalah perangkat keras yang dapat mendukung perangkat lunak yang memiliki kemampuan untuk mengolah data dan tampilan grafisnya cukup baik. Perangkat yang digunakan tersedia untuk pembuatan aplikasi adalah:

Laptop : Lenovo

Processor : Processor Intel ® Core(TM) i7-7700HQ CPU

RAM : Memory berkapasitas 16 GB

System Type : System Type 64-bit Operating System, x64-based processor

5.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang dibutuhkan dan dipergunakan dalam pembuatan aplikasi:

Sistem Operasi : Windows 10 Pro

Frame Work : CodeIgniter 3.0.0

Bahasa Pemrograman : -PHP (Hypertext Preprocessor)
-HTML (Hypertext Markup Language)

Database : MySQL

Server Offline : XAMPP 3.2.2

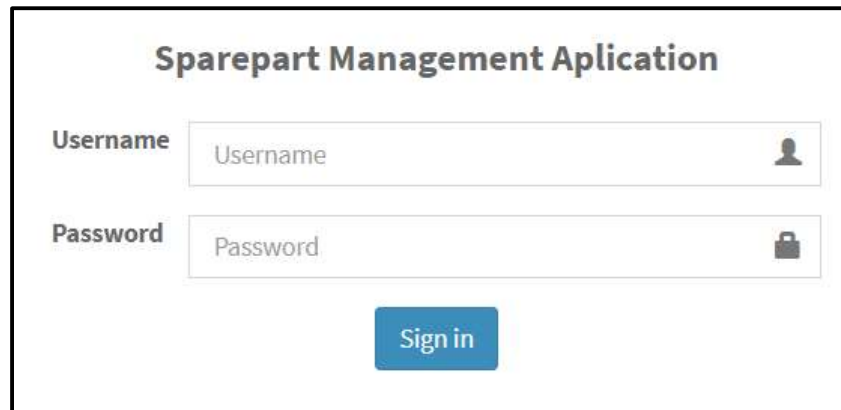
5.1.3 Implementasi Antar Muka

Berikut ini adalah contoh implementasi antar muka Sistem Informasi

Penjualan Sparepart Pada Bengkel Moto Link yang telah dibuat :

1. Tampilan Login Admin

Halaman login ketika mengakses website bengkel moto link. Halaman field terdiri dari 2 kotak isian, yaitu *username* dan *password*. Tampilan antar muka halaman *login* yang disajikan pada Gambar 5.1



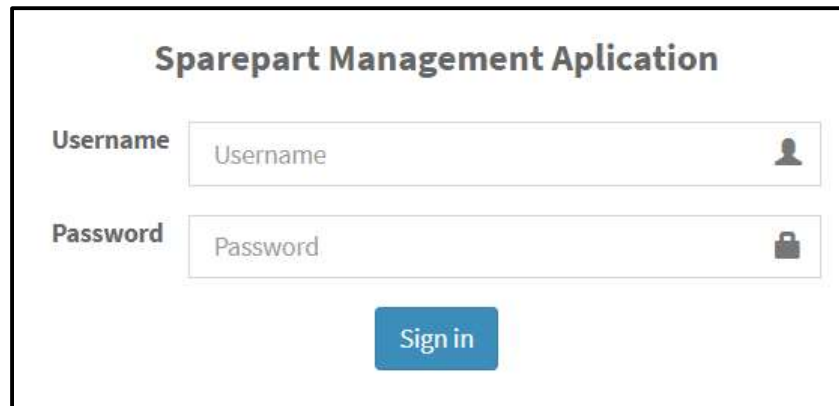
The image shows a web form titled "Sparepart Management Aplication". It contains two input fields: "Username" and "Password". The "Username" field has a placeholder text "Username" and a user icon. The "Password" field has a placeholder text "Password" and a lock icon. Below the input fields is a blue button labeled "Sign in".

Gambar 5.1 Tampilan Login Admin

Keterangan gambar diatas merupakan tampilan *login* admin dan kasir untuk masuk ke dalam sistem, jika *username* dan *password* masih salah maka admin tidak diizinkan untuk masuk kedalam sistem dan akan terus ditampilkan *form login* tersebut, jika *username* dan *password* benar maka dapat mengakses sistem tersebut.

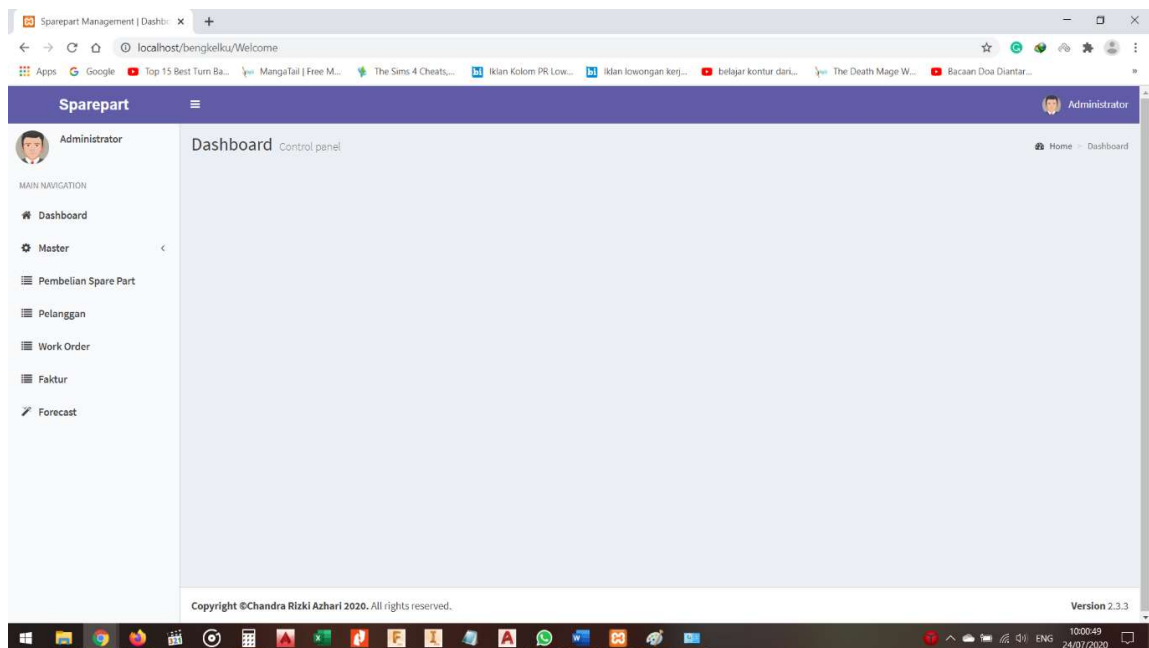
2. Tampilan Login Kasir

Tampilan gambar dibawah merupakan login untuk login kedalam sistem, jika *username* dan *password* masih salah maka kasir tidak diizinkan untuk masuk kedalam sistem dan akan terus ditampilkan form login tersebut, jika *username* dan *password* benar maka dapat mengakses sistem tersebut.



Gambar 5.2 Tampilan Login Kasir

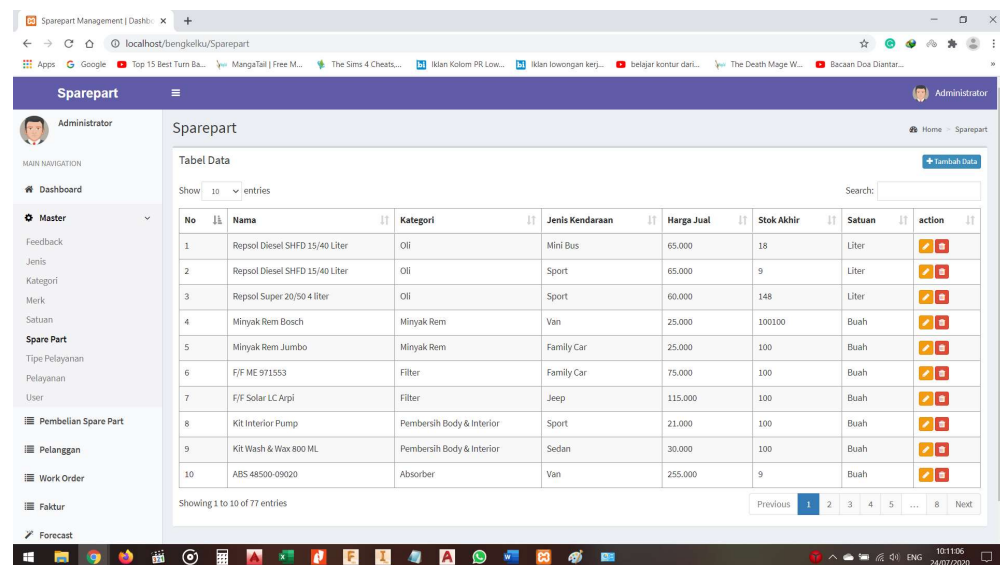
3. Tampilan Dashboard



Gambar 5.3 Tampilan Dashboard

Tampilan gambar diatas merupakan tampilan Dashboard untuk menampilkan beranda dan menu sidebar yang ada pada website tersebut. Menampilkan Login sebagai Admin.

4. Tampilan Sparepart



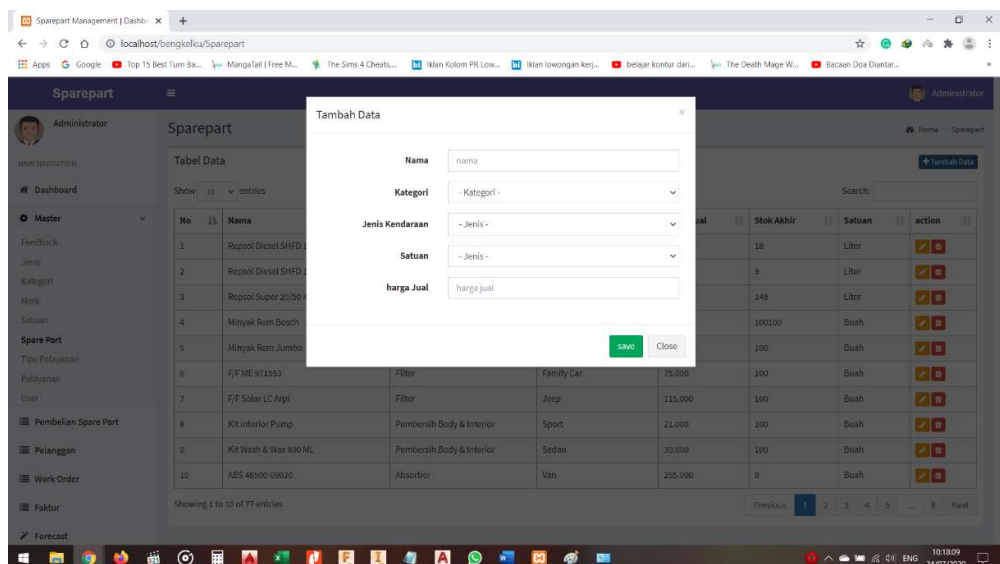
The screenshot shows the 'Sparepart' management interface. On the left is a sidebar with navigation options: Dashboard, Master (Feedback, Jenis, Kategori, Merk, Satuan, Spare Part, Tipe Pelayanan, Pelayanan, User), Pembelian Spare Part, Pelanggan, Work Order, and Faktur. The main area displays a 'Tabel Data' with 10 entries. Each entry includes a serial number, name, category, vehicle type, price, stock, unit, and action buttons (edit and delete). A search bar and a 'Tambah Data' button are at the top right. The table shows 10 entries, with the first 10 displayed.

No	Nama	Kategori	Jenis Kendaraan	Harga Jual	Stok Akhir	Satuan	action
1	Repsol Diesel SHFD 15/40 Liter	Oli	Mini Bus	65.000	18	Liter	[edit] [delete]
2	Repsol Diesel SHFD 15/40 Liter	Oli	Sport	65.000	9	Liter	[edit] [delete]
3	Repsol Super 20/50 4 liter	Oli	Sport	60.000	148	Liter	[edit] [delete]
4	Minyak Rem Bosch	Minyak Rem	Van	25.000	100100	Buah	[edit] [delete]
5	Minyak Rem Jumbo	Minyak Rem	Family Car	25.000	100	Buah	[edit] [delete]
6	F/F ME 9711553	Filter	Family Car	75.000	100	Buah	[edit] [delete]
7	F/F Solar LC Arpi	Filter	Jeep	115.000	100	Buah	[edit] [delete]
8	Kit Interior Pump	Pembersih Body & Interior	Sport	21.000	100	Buah	[edit] [delete]
9	Kit Wash & Wax 800 ML	Pembersih Body & Interior	Sedan	30.000	100	Buah	[edit] [delete]
10	ABS 48500-09920	Absorber	Van	255.000	9	Buah	[edit] [delete]

Gambar 5.4 Tampilan Sparepart

Tampilan sparepart berfungsi menampilkan seluruh sparepart yang telah ditambahkan. Proses ini dilakukan oleh admin yang dapat menambah, mengedit, dan menghapus sparepart.

5. Tampilan Tambah Sparepart



The screenshot shows the 'Sparepart' management interface with the 'Tambah Data' modal open. The modal contains fields for Nama, Kategori, Jenis Kendaraan, Satuan, and harga Jual. The background shows the same table as in Gambar 5.4, but it is dimmed. The 'Tambah Data' button is visible in the top right corner of the main area.

Gambar 5.5 Tampilan Tambah Sparepart

Tampilan tambah sparepart berfungsi menambahkan sparepart jika ada sparepart baru yang ingin ditambahkan. Proses ini dilakukan oleh admin yang dapat mengedit, menambah, dan menghapus sparepart.

6. Tampilan Kategori

Sparepart Management | Dashboard

Kategori

Tabel Data

Show 10 entries

Search:

No	Nama	action
1	Oli	[Edit] [Delete]
2	Minyak Rem	[Edit] [Delete]
3	Filter	[Edit] [Delete]
4	Pembersih Body & Interior	[Edit] [Delete]
5	Absorber	[Edit] [Delete]
6	Ball Joint	[Edit] [Delete]
7	Cross Joint	[Edit] [Delete]
8	Baud	[Edit] [Delete]
9	Bohlam	[Edit] [Delete]
10	Boot	[Edit] [Delete]

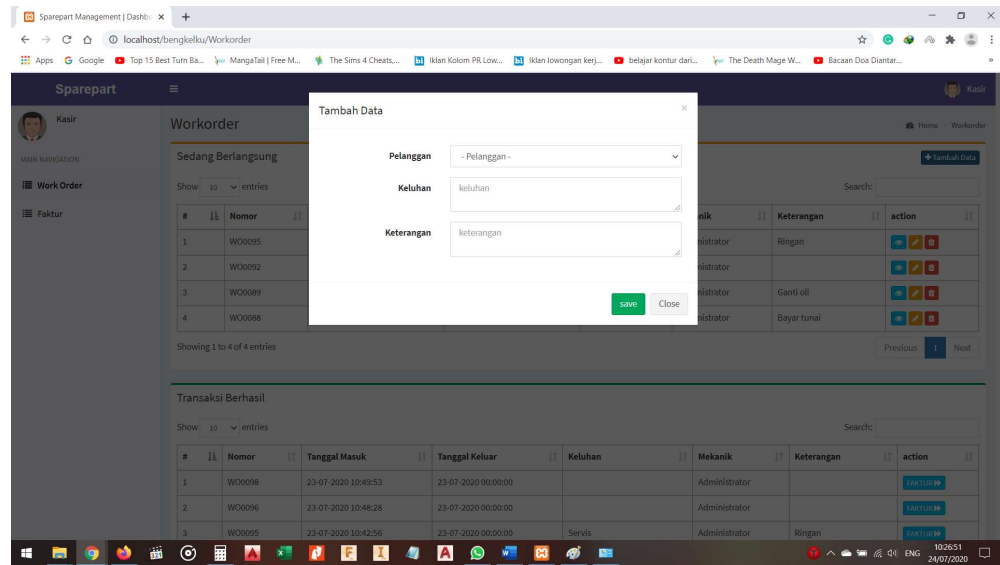
Showing 1 to 10 of 39 entries

Previous 1 2 3 4 Next

Gambar 5.6 Tampilan Tambah Kategori Sparepart

Keterangan diatas merupakan tampilan tambah kategori sparepart berfungsi menambah dan juga menampilkan kategori sparepart yang akan ditampilkan. Proses ini dilakukan oleh admin yang dapat mengedit, menambahkan, dan menghapus sparepart.

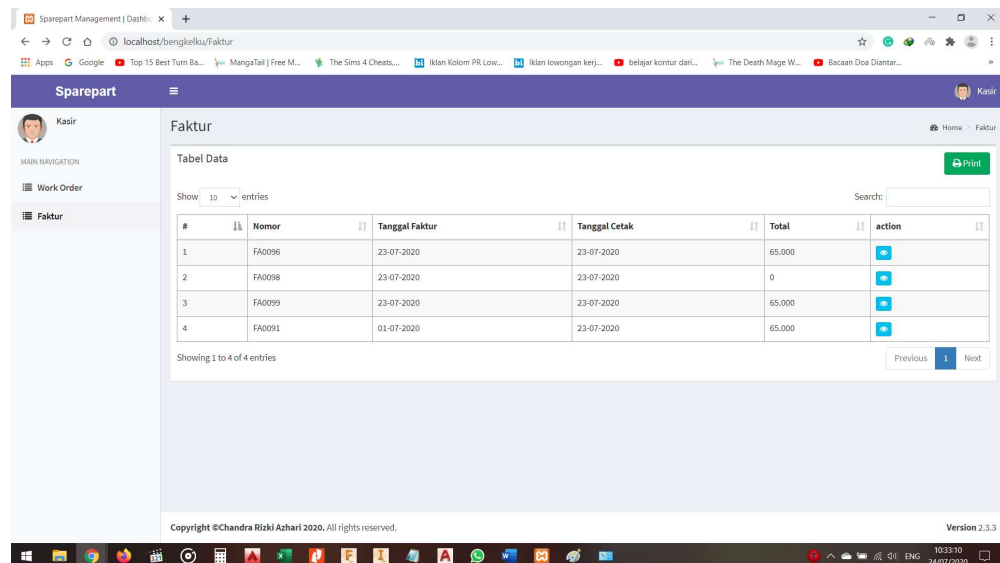
7. Tampilan Entry Work Order Oleh Kasir



Gambar 5.7 Tampilan Work Order

Tampilan gambar diatas berfungsi untuk memulai transaksi, klik pada tambah data kemudian memasukkan data sparepart yang dipesan oleh pelanggan lalu proses transaksi ditampilkan pada *table* transaksi.

8. Tampilan Laporan Work Order



Gambar 5.8 Tampilan Laporan Work Order

Tampilan gambar diatas berfungsi untuk menampilkan laporan work order oleh kasir, kemudian dapat dihitung berapa jumlah total transaksi penjualan melalui cetak laporan.

9. Tampilan Cetak Laporan Work Order

Workorder #WO0095				
Nama Pemilik : Adit		No Polisi : D 2049		
Alamat Rumah : TKI 1		Jenis : Sedan		
No Telepon : 08926472912		Merk : Honda		
No	Pelayanan	Sparepart	Jumlah	Sub Total
1	Ganti Oli	Repsol Diesel SHFD 15/40 Liter	1	65.000

Gambar 5.9 Tampilan Laporan Work Order

Tampilan gambar diatas merupakan tampilan cetak work order yang harus dibayar.

10. Tampilan Cetak Laporan Faktur

Laporan Faktur						
July						
No	Nomor Faktur	Nomor Workorder	Tanggal Masuk	Tanggal Faktur	Tanggal Cetak	Total
1	FA0091	WO0091	01-07-2020 09:18:43	01-07-2020 00:00:00	23-07-2020 00:00:00	65.000
2	FA0096	WO0095	23-07-2020 10:42:56	23-07-2020 00:00:00	23-07-2020 00:00:00	65.000
3	FA0098	WO0096	23-07-2020 10:48:28	23-07-2020 00:00:00	23-07-2020 00:00:00	0
4	FA0099	WO0098	23-07-2020 10:49:53	23-07-2020 00:00:00	23-07-2020 00:00:00	65.000
Total						195.000

Gambar 5.10 Tampilan Laporan Faktur

Tampilan gambar diatas untuk menampilkan laporan penjualan perbulan yang akan dicetak oleh admin, kemudian dapat dihitung berapa jumlah total transaksi penjualan.

5.2 Pengujian

5.2.1 Rencana Pengujian

Sebelum aplikasi diterapkan di lapangan, perlu adanya proses pengujian untuk menentukan kesalahan pada aplikasi. Pada tahap pengujian ini penulis menggunakan metode blackbox yaitu metode pengujian perangkat lunak dengan hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut.

Berikut adalah hasil pengujian yang telah dilakukan oleh penulis :

1. Rencana Pengujian

Tabel 5.1 Tabel Rencana Pengujian Admin

No.	Kode	Nama Proses	Input	Output
1	SKPL-01	Login Admin	Masukan username dan password ke halaman login aplikasi	Masuk Ke halaman Admin
2	SKPL-02	Login Kasir	Masukan username dan password ke halaman login aplikasi	Data tersimpan ke label galeri
3	SKPL-03	Entry Sparepart	Entry Data Sparepart	Data Tersimpan ke Tabel Tambah Sparepart

4	SKPL-04	Entry Tambah Sparepart	Entry Data Tambah Sparepart	Data Tersimpan ke Tabel Tambah Sparepart
5	SKPL-05	Entry Kategori Sparepart	Entry Data Kategori Sparepart	Data Tersimpan ke Tabel Kategori Sparepart
6	SKPL-06	Entry Tambah Kasir	Entry Tambah Kasir	Data Tersimpan ke Tabel Tambah Kasir
7	SKPL-07	Entry Work Order	Entry Data Work Order	Data Tersimpan ke Tabel Work Order
8	SKPL-08	Entry Laporan Work Order	Entry Data Laporan Work Order	Data Tersimpan ke Tabel Laporan Work Order
9	SKPL-09	Entry Laporan Faktur	Entry data Laporan Faktur	Data Tersimpan ke Tabel Laporan Faktur

Tabel 5.2 Tabel Rencana Pengujian Kasir

Kelas Uji	Butir Uji	Identifikasi		Jenis Pengujian	Teknik Pengujian
		SKPL	PDHUPL		
Login	Login dengan username dan password yang terdaftar	SKPL-01	PDHUPL-01	Pengujian Sistem	Blackbox
	Login dengan username dan password yang tidak terdaftar	SKPL-01	PDHUPL-02	Pengujian Sistem	Blackbox
Entry Sparepart	Entry Sparepart dengan menginputkan semua sparepart	SKPL-02	PDHUPL-03	Pengujian Sistem	Blackbox
	Entry Sparepart dengan tidak menginputkan salah satu inputan data sparepart	SKPL-02	PDHUPL-04	Pengujian Sistem	Blackbox

Entry Tambah Sparepart	Entry Tambah Sparepart dengan menginputkan semua inputan data tambah sparepart	SKPL-03	PDHUPL-05	Pengujian Sistem	Blackbox
	Entry Tambah Sparepart dengan tidak menginputkan salah satu inputan data tambah sparepart	SKPL-03	PDHUPL-06	Pengujian Sistem	Blackbox
Entry Kategori Sparepart	Entry Kategori sparepart dengan menginputkan semua inputan data kategori sparepart	SKPL-04	PDHUPL-07	Pengujian Sistem	Blackbox
	Entry Kategori sparepart dengan tidak menginputkan salah satu inputan data kategori sparepart	SKPL-04	PDHUPL-08	Pengujian Sistem	Blackbox

Entry Tambah Kasir	Entry tambah Kasir dengan menginputkan semua inputan data Entry Tambah Kasir	SKPL-05	PDHUPL-09	Pengujian Sistem	Blackbox
	Entry review dengan tidak menginputkan salah satu Entry Tambah Kasir	SKPL-05	PDHUPL-10	Pengujian Sistem	Blackbox
Entry Work Order	Entry Work Order dengan menginputkan semua inputan data Entry Work Order	SKPL-06	PDHUPL-11	Pengujian Sistem	Blackbox
	Entry Work Order dengan tidak menginputkan salah satu Entry Work Order	SKPL-06	PDHUPL-12	Pengujian Sistem	Blackbox
Entry Laporan Faktur	Entry Laporan Faktur dengan menginputkan semua inputan data Entry Laporan Faktur	SKPL-07	PDHUPL-13	Pengujian Sistem	Blackbox
	Entry Laporan Faktur dengan tidak menginputkan salah satu inputan data Entry Laporan Faktur	SKPL-07	PDHUPL-14	Pengujian Sistem	Blackbox

5.2.2 Deskripsi dan Hasil Pengujian

Tabel 5.3 hasil pengujian login dengan username dan password yang terdaftar

Identifikasi	PDHUPL-01		
Nama Butir Uji	Login dengan username dan password yang terdaftar		
Tujuan	Memeriksa username dan password yang diinputkan dan terdaftar masuk ke halaman admin		
Kondisi Awal	User membuka halaman login		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input username dan password			
2. Klik tombol login			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Username=admin Password=admin	Masuk ke halaman admin	Akses website sesuai login	OK

Tabel 5.4 hasil pengujian login dengan username dan password yang tidak terdaftar

Identifikasi	PDHUPL-02		
Nama Butir Uji	Login dengan username dan password yang tidak terdaftar		

Tujuan	Memeriksa username dan password yang diinputkan dan tidak terdaftar tidak dapat masuk ke halaman admin		
Kondisi Awal	User membuka halaman login		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input username dan password			
2. Klik tombol login			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Username=administrator Password=administrator	Tidak Masuk ke halaman admin dan kembali ke halaman login	Login ditolak tidak dapat masuk ke halaman login	OK

Tabel 5.5 hasil pengujian entry data sparepart dengan menginputkan semua data sparepart

Identifikasi	PDHUPL-03
Nama Butir Uji	Entry data sparepart dengan menginputkan semua inputan data sparepart
Tujuan	Memeriksa apakah inputan data sparepart tersimpan ke database
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data sparepart
Tanggal Pengujian	20/07/2019
Penguji	Chandra Rizki Azhari
Skenario	

1. Input data sparepart			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id_saprepart=sp01 Nama_sparepart=RepsolDiesel	Data sparepart tersimpan ke database	Data sparepart tersimpan ke database	OK

Tabel 5.6 hasil pengujian entry data sparepart dengan tidak menginputkan salah satu data sparepart

Identifikasi	PDHUPL-04		
Nama Butir Uji	Entry data sparepart dengan tidak menginputkan salah satu inputan data sparepart		
Tujuan	Memeriksa apakah inputan data sparepart tersimpan ke database		
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data sparepart		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data sparepart			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id_saprepart=sp01 Nama_sparepart=RepsolDiesel	Data sparepart tidak tersimpan ke database	Data sparepart tidak tersimpan ke database	OK

Tabel 5.7 hasil pengujian entry data tambah sparepart dengan menginputkan semua data tambah sparepart

Identifikasi	PDHUPL-05		
Nama Butir Uji	Entry data sparepart dengan menginputkan semua data tambah sparepart		
Tujuan	Memeriksa apakah inputan data tambah sparepart tersimpan ke database		
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data sparepart		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data sparepart			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id_saprepart=sp01 Nama_sparepart=RepsolDiesel	Data sparepart tersimpan ke database	Data sparepart tersimpan ke database	OK

Tabel 5.8 hasil pengujian entry data tambah sparepart dengan tidak menginputkan salah satu data tambah sparepart

Identifikasi	PDHUPL-06
Nama Butir Uji	Entry data sparepart dengan tidak menginputkan salah satu data tambah sparepart
Tujuan	Memeriksa apakah inputan data tambah sparepart tersimpan ke database
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data tambah sparepart

Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data tambah sparepart			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id_List=Fst01 Nama_sparepart=RepsolDiesel	Data tambah sparepart tidak tersimpan ke database	Data tambah sparepart tidak tersimpan ke database	OK

Tabel 5.9 hasil pengujian entry data kategori sparepart dengan menginputkan semua data kategori sparepart

Identifikasi	PDHUPL-07
Nama Butir Uji	Entry data sparepart dengan menginputkan semua inputan kategori sparepart
Tujuan	Memeriksa apakah inputan data kategori sparepart tersimpan ke database
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data kategori sparepart
Tanggal Pengujian	20/07/2019
Penguji	Chandra Rizki Azhari
Skenario	
1. Input data kategori sparepart	
2. Klik tombol simpan	
Hasil	

Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id_List=Fst01 Nama_sparepart = oli	Data kategori sparepart tersimpan ke database	Data kategori sparepart tersimpan ke database	OK

Tabel 5.10 hasil pengujian entry data kategori sparepart dengan tidak menginputkan salah satu data kategori sparepart

Identifikasi	PDHUPL-08		
Nama Butir Uji	Entry data sparepart dengan tidak menginputkan salah satu inputan data kategori sparepart		
Tujuan	Memeriksa apakah inputan data kategori sparepart tersimpan ke database		
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data kategori sparepart		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data kategori sparepart			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan

Id_List=Fst01	Data kategori sparepart tidak tersimpan ke database	Data kategori sparepart tidak tersimpan ke database	OK
Nama_sparepart = oli			

Tabel 5.11 hasil pengujian entry data tambah kasir dengan menginputkan semua data tambah kasir

Identifikasi	PDHUPL-09		
Nama Butir Uji	Entry data sparepart dengan menginputkan semua inputan data ke tambah kasir		
Tujuan	Memeriksa apakah inputan data tambah kasir tersimpan ke database		
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data tambah kasir		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data tambah kasir			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id tambah kasir =Ftk01 tambah Kasir = Sonia	Data tambah kasir tersimpan ke database	Data tambah kasir tersimpan ke database	OK

Tabel 5.12 hasil pengujian entry data tambah kasir dengan tidak menginputkan salah satu data tambah kasir

Identifikasi	PDHUPL-10		
Nama Butir Uji	Entry data sparepart dengan tidak menginputkan salah satu inputan data ke tambah kasir		
Tujuan	Memeriksa apakah inputan data tambah kasir tersimpan ke database		
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data tambah kasir		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data tambah kasir			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id tambah kasir =Ftk01 tambah Kasir = Sonia	Data tambah kasir tidak tersimpan ke database	Data tambah kasir tidak tersimpan ke database	OK

Tabel 5.13 hasil pengujian entry data work order dengan menginputkan semua data work order

Identifikasi	PDHUPL-11		
Nama Butir Uji	Entry data work order dengan menginputkan semua inputan data ke work order		
Tujuan	Memeriksa apakah inputan data work order tersimpan ke database		
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data work order		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data work order			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id work order = Wo01 transaksi = kasir	Data work order tersimpan ke database	Data work order tersimpan ke database	OK

Tabel 5.14 hasil pengujian entry data work order dengan tidak menginputkan salah satu data work order

Identifikasi	PDHUPL-12
Nama Butir Uji	Entry data work order dengan tidak menginputkan salah satu inputan data ke work order

Tujuan	Memeriksa apakah inputan data work order tersimpan ke database		
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data work order		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data work order			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id work order = Wo01 transaksi = kasir	Data work order tidak tersimpan ke database	Data work order tidak tersimpan ke database	OK

Tabel 5.15 hasil pengujian entry laporan faktur dengan menginputkan semua data laporan faktur

Identifikasi	PDHUPL-13
Nama Butir Uji	Entry data laporan faktur dengan semua inputan data ke laporan faktur
Tujuan	Memeriksa apakah inputan laporan faktur tersimpan ke database
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data faktur
Tanggal Pengujian	20/07/2019
Penguji	Chandra Rizki Azhari
Skenario	

1. Input data faktur			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan
Id laporan faktur = FA01 transaksi = kasir	Data laporan faktur tersimpan ke database	Data laporan faktur tersimpan ke database	OK

Tabel 5.16 hasil pengujian entry laporan faktur dengan tidak menginputkan menginputkan salah satu data laporan faktur

Identifikasi	PDHUPL-14		
Nama Butir Uji	Entry data laporan faktur dengan tidak menginputkan salah satu inputan data ke laporan faktur		
Tujuan	Memeriksa apakah inputan laporan faktur tersimpan ke database		
Kondisi Awal	Admin sudah memilih (membuka) halaman web untuk entry data faktur		
Tanggal Pengujian	20/07/2019		
Penguji	Chandra Rizki Azhari		
Skenario			
1. Input data faktur			
2. Klik tombol simpan			
Hasil			
Data yang diberikan	Hasil yang diharapkan	Pengamatan	Kesimpulan

Id laporan faktur = FA01 transaksi = kasir	Data laporan faktur tidak tersimpan ke database	Data laporan faktur tidak tersimpan ke database	OK
---	---	---	----

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Aplikasi penjualan sparepart di bengkel Moto Link dibangun untuk menjadi alat bantu bagi admin dan kasir dalam proses pengolahan data dan transaksi. Sistem informasi penjualan produk yang diusulkan telah memenuhi fungsi-fungsi yang diharapkan, seperti pengolahan data menu, mengelola konfirmasi transaksi, dan laporan penjualan.

Dengan dibangunnya aplikasi ini, membantu untuk pengontrolan faktur sehingga dapat mengantisipasi kehilangan faktur.

6.2 Saran

Pada penulisan skripsi ini penulisan meyakini tentu masih banyak terdapat kesalahan dan kekurangan, baik dalam sisi penulisan dokumen serta web yang dibuat ini masih perlu banyak pengembangan. Maka saran yang diberikan penulis untuk membantu pengembangan aplikasi ini kedepannya antara lain:

- 1 Memberikan tampilan user interface yang lebih menarik.
- 2 Aplikasi penjualan sparepart di bengkel Moto Link ini dapat dikembangkan, akan tetapi aplikasi ini dibuat hanya dikhususkan untuk penjualan sparepart bengkel moto link yang saat ini sedang berjalan.

DAFTAR PUSTAKA

- Arief, M, Rudiyanto. (2011). *Pemrograman Web Dinamis Menggunakan Php dan Mysql*. Yogyakarta: CV.
- Arizona, N. D. (2017). Aplikasi Pengolahan Data Anggaran Pendapatan dan Belanja Desa (APBDES) Pada Kantor Desa Bakau Kecamatan Jawai Berbasis Web, 01(02), 105–119. Retrieved from <http://openjournal.unmuhpnk.ac.id/index.php/CN/article/view/745>
- A. S., Rosa dan Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Chan, Syahrial. (2017). *Membuat Aplikasi Database dengan PowerBuilder 12.6 dan MySQL*. Jakarta: PT Elek Media Komputindo.
- Edi. (2019). *Pengertian Website. Jenis, Manfaat, dan Unsur-Unsur Website*, p. Maxmanroe.com.
- Farid. (2017, Juli). *Pengertian Blackbox. Kumpulan Pengertian Menurut Para Ahli*, Pp. <http://www.infodanpengertian.com/2018/11/pengertian-black-box-testing-menurut.html>.
- Firdaus. (2017). *7 Jam Belajar Interaktif PHP & MySQL dengan Dreamweaver* Palembang: Maxikom.
- Marsudi, D. (2005), “Pembangkitan Energi Listrik”. Jakarta: Erlangga
- Maimanah. (2015). *Prakerin uBig.co.id Apa itu Balsamiq Mockup?* <https://prakerinmalang.wordpress.com/perihal/>.
- Moekijat. (2020, Juli 7). *Pengertian Penjualan Menurut Para Ahli. Pp.* <https://www.pelajaran.co.id/2017/20/pengertian-penjualan-menurut-para-ahli-tujuan-dan-jenis-penjualan.html>.

- Ngoding, M. (2020, Mei 14). Code Igniter Part 1. Pengertian dan Cara Menggunakan CodeIgniter, pp.<https://www.malasngoding.com/pengertian-dan-cara-menggunakan-codeigniter/>.
- Nickels, W.G. (2020, Juli 7). *Pengertian Penjualan Menurut Para Ahli*. Pp. <https://www.pelajaran.co.id/2017/20/pengertian-penjualan-menurut-para-ahli-tujuan-dan-jenis-penjualan.html>.
- NitiSemitto. (2020, Juli 7). *Pengertian Penjualan Menurut Para Ahli*. Pp. <https://www.pelajaran.co.id/2017/20/pengertian-penjualan-menurut-para-ahli-tujuan-dan-jenis-penjualan.html>.
- Peter, (2020, Juli 7) Pengetian Object Oriented Analysis Design, Pp. <https://peterdraw.wordpress.com/2011/10/30/konsep-ooad-object-oriented-analysis-design/>
- Prayitno, Agus, dan Yulia Safitri. (2015). *Pemanfaatan Sistem Informasi Perpustakaan Digital Berbasis Website Untuk Para Penulis Vol 1*. Pp. <http://lppm3.bsi.ac.id/jurnal/index.php/ijse/article/viewFile/214/149>.
- Riyanto. (2015). XAMPP.
- Wasiyanti, S., Talaohu, R., Studi, P., Akuntansi, K., Bandung, A., Studi, P., & Akuntansi, K. (2016). PARADIGMA Vol. XVIII. No.2 September 2016 SISTEM INFORMASI PENJUALAN OBAT BERBASIS WEB PADA APOTEK KONDANG WARAS DEPOK, *XVIII*(2), 49–62.
- Wikipedia. (2015). *Pengertian Spare Part*. <https://id.wikipedia.org/wiki/Onderdil>
- Yakub, dan Vico Hisbanarto. (2015). *Sistem Informasi Manajemen Pendidikan*. Yogyakarta: Graha Ilmu.

Lampiran *Source Code* Aplikasi

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

/**
 *
 * Controller Welcome
 *
 * This controller for ...
 *
 * @package CodeIgniter
 * @category Controller CI
 * @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
 * @param ...
 * @return ...
 *
 */

class Welcome extends CI_Controller
{

    function __construct(){
        parent::__construct();
        // if (!$this->session->userdata("id")){
        //     redirect("Login");
        // }

        $id = $this->session->userdata("id");
        $this->user = $this->M_user->getDetail($id);
    }
}
```

```

public function index()
{
    if (!$this->session->userdata("id")){
        redirect("Login");
    }
    $data["konten"] = "dashboard";
    $this->load->view('template',$data);
}

public function dashboard()
{
    $data["konten"] = "dashboard";
    $this->load->view('template',$data);
}

public function profil(){
    if($this->input->post("btnsubmit")){
        $data["nama"] = $this->input->post("nama");
        if($this->input->post("password")){
            $data["password"] = $this->encrypt-
>encode($this->input->post("password"));
        }
        $this->M_user->update($this->user->userid,$data);
        redirect("Welcome/profil");
    }

    $data['user'] = $this->user;
    $data['konten'] = "profil";
    $this->load->view('template',$data);
}

public function export(){

```

```

        $data['filename'] = "Laporan";
        $data["rowData"] = $this->M_user->getAll();
        $data['konten'] = "page/export/user";
        $this->load->view("page/export/templatePdf",$data);
    }
}

```

/* End of file Welcome.php */

/* Location: ./modules/controllers/Welcome.php */

```

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

/**
 *
 * Controller Workorder
 *
 * This controller for ...
 *
 * @package CodeIgniter
 * @category Controller CI
 * @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
 * @param ...
 * @return ...
 */

class Workorder extends CI_Controller
{
    var
    $kelas = "Workorder";

    function
    __construct(){
        parent::__construct();
    }
}

```



```

if
(!$this->session->userdata("id")){

    redirect(
    }

    $id
    = $this->session->userdata("id");

    $this->
    >user = $this->M_user->getDetail($id);

}

public
function index(){

    $data["r
owData"] = $this->M_workorder->getAll();

    $data['k
onten'] = "workorder/index";

    $this->
    >load->view('template',$data);

}

public
function detail($id){

    $data["
data"] = $wo = $this->M_workorder->getDetail($id);

    $data["r
owData"] = $this->M_workorder_detail->getAllBy("workorderid = ".$id);

    $data["r
owCust"] = $this->M_pelanggan->getDetail($wo->pelangganid);

    $data['k
onten'] = "workorder/detail";

    $this->
    >load->view('template',$data);

}

```

function detailJson(\$id){	public
header('Content-Type: application/json');	
a = \$this->M_workorder->getDetail(\$id);	\$rowDat
json_encode(\$rowData);	echo
	}
function add(){	public
= \$this->input->post("workorderid");	\$id
userid"] = \$this->user->userid;	\$data["
nomor"] = \$this->nomor("WO");	\$data["
pelangganid"] = \$this->input->post("pelangganid");	\$data["
anggal_masuk"] = date("Y-m-d H:i:s");	\$data["t
keluhan"] = \$this->input->post("keluhan");	\$data["
keterangan"] = \$this->input->post("keterangan");	\$data["
	if(\$id)
>M_workorder->update(\$id,\$data);	\$this-
	else
>M_workorder->add(\$data);	\$this-
\$this->kelas;	redirect(
	}

```

public
function addDetail(){
    $data["
workorderid"] = $workorderid = $this->input->post("workorderid");
    $data["
pelayananid"] = $pelayananid = $this->input->post("pelayananid");
    $data["s
parepartid"] = $sparepartid = $this->input->post("sparepartid");
    $sparep
art = $this->M_sparepart->getDetail($sparepartid);
    $data["
qty"] = $qty = $this->input->post("qty");
    if($spar
epart->stokakhir < $qty){
        $this-
>session->set_flashdata("warning","Stok <strong>$sparepart-
>nama</strong> tidak mencukupi");
    }else{
        $data["t
otal"] = $sparepart->hargajual*$qty;
        $this-
>M_workorder_detail->add($data);
    }
    redirect(
$this->kelas."/detail/".$workorderid);
}
public
function save($id){
    //
    $id = $this->input->post("workorderid");
    $rowDet
ailWo = $this->M_workorder_detail->getAllBy("workorderid = ".$id);

```

```

0;
                                $total =

($rowDetailWo as $detailWo) {
                                foreach

                                //update
stok SPAREPART

                                $total+=
$detailWo->total;

                                $sparep
art = $this->M_sparepart->getDetail($detailWo->sparepartid);

                                $dataSp
arepart["stokakhir"] = $sparepart->stokakhir - $detailWo->qty;

                                $this-
>M_sparepart->update($detailWo->sparepartid,$dataSparepart);
                                }

                                //add
FAKTUR

                                $dataPe
njualan["woid"] = $id;

                                $dataPe
njualan["total"] = $total;

                                $dataPe
njualan["userid"] = $this->user->userid;

                                $dataPe
njualan["nomorfaktur"] = $this->nomor("FA");
                                //
TODO DEV
                                //
$dataPenjualan["tanggal_faktur"] = date("Y-m-d H:i:s");

                                $dataPe
njualan["tanggal_faktur"] = $this->input->post("tanggal");

```

>M_penjualan->add(\$dataPenjualan);	\$this-
status WO	//update
o["status"] = "F";	\$dataW
TODO DEV	//
\$dataWo["tanggal_keluar"] = date("Y-m-d H:i:s");	//
o["tanggal_keluar"] = \$this->input->post("tanggal");	\$dataW
>M_workorder->update(\$id,\$dataWo);	\$this-
\$this->kelas;	redirect(
	}
function update(\$id){	public
>input->post("btnsubmit")){	if(\$this-
drefatribut"] = \$this->input->post("idrefatribut");	\$data["i
>M_workorder->update(\$id,\$data);	\$this-
\$this->kelas;	redirect(
	}
data"] = \$this->M_workorder->getDetail(\$id);	\$data["
onten'] = "workorder/index";	\$data['k

>load->view('template',\$data);	\$this-
	}
function batal(\$id){	public
	\$data["s
tatus"] = "NF";	\$this-
>M_workorder->update(\$id,\$data);	redirect(
\$this->kelas);	}
function delete(\$id){	public
	\$rowWo
rkorder = \$this->M_workorder_detail->getAllBy("workorderid = \$id");	foreach
(\$rowWorkorder as \$row) {	\$this-
>deleteDetail(\$row->detailwoid, 1);	}
	\$this-
>M_workorder->delete(\$id);	redirect(
\$this->kelas);	}
function deleteDetail(\$id, \$loop = 0){	public
	\$workor
derid = \$this->M_workorder_detail->getDetail(\$id)->workorderid;	\$this-
>M_workorder_detail->delete(\$id);	

```

!= 1) redirect($this->kelas."/detail/".$workorderid);

function nomor($param){

    $this->M_workorder->getMax()->workorderid;

    = sprintf("%04d",$max);

    $param.$nomor;

}

/* End of file Workorder.php */
/* Location: ./modules/controllers/Workorder.php */

```

```

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

/**
 *
 * Controller Sparepart
 *
 * This controller for ...
 *
 * @package CodeIgniter
 * @category Controller CI
 * @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
 * @param ...
 * @return ...
 */

class Sparepart extends CI_Controller
{

    var

    $kelas = "Sparepart";

```

__construct(){	function
__construct();	parent::
(!\$this->session->userdata("id")){	if
"Login");	redirect(
	}
	}
function index(){	public
rowData"] = \$this->M_sparepart->getAll();	\$data["r
onten'] = "sparepart/index";	\$data['k
>load->view('template',\$data);	\$this-
	}
function detail(\$id){	public
data"] = \$this->M_sparepart->getDetail(\$id);	\$data["
onten'] = "sparepart/detail";	\$data['k
>load->view('template',\$data);	\$this-
	}
function detailJson(\$id){	public
header('Content-Type: application/json');	
a = \$this->M_sparepart->getDetail(\$id);	\$rowDat

json_encode(\$rowData);	echo
	}
function add(){	public
= \$this->input->post("sparepartid");	\$id
nama"] = \$this->input->post("nama");	\$data["
kategoriid"] = \$this->input->post("kategoriid");	\$data["
enisid"] = \$this->input->post("jenisid");	\$data["j
atuanid"] = \$this->input->post("satuanid");	\$data["s
hargajual"] = \$this->input->post("hargajual");	\$data["
	if(\$id)
>M_sparepart->update(\$id,\$data);	\$this-
	else
>M_sparepart->add(\$data);	\$this-
\$this->kelas);	redirect(
	}
function delete(\$id){	public
>M_sparepart->delete(\$id);	\$this-
\$this->kelas);	redirect(
	}

```
}
```

```
/* End of file Sparepart.php */
```

```
/* Location: ./modules/controllers/Sparepart.php */
```

```
<?php
```

```
defined('BASEPATH') or exit('No direct script access allowed');
```

```
/**
```

```
*
```

```
* Controller Login
```

```
*
```

```
* This controller for ...
```

```
*
```

```
* @package CodeIgniter
```

```
* @category Controller CI
```

```
* @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
```

```
* @param ...
```

```
* @return ...
```

```
*
```

```
*/
```

```
class Login extends CI_Controller
```

```
{
```

var

```
$kelas = "Login";
```

function

```
__construct(){
```

parent::

```
__construct();
```

```
date_default_timezone_set("Asia/Jakarta");
```

}

public

```
function index(){
```

if

```
($this->input->post('btnsubmit')) {
```

\$userna

```
me = $this->input->post('username');
```

rd = \$this->input->post('password');	\$passwo
\$this->M_user->login(\$username,\$password);	\$cek =
	if(\$cek){
>session->set_userdata("id",\$cek->userid);	\$this-
>session->set_userdata("error",FALSE);	\$this-
>session->set_userdata("welcome",TRUE);	\$this-
>jabatan == "Administrator")	if(\$cek-
"Welcome");	redirect(
if(\$cek->jabatan == "Kasir")	else
"Welcome");	redirect(
if(\$cek->jabatan == "Manager")	else
"Welcome");	redirect(
if(\$cek->jabatan == "Kepala Gudang")	else

```

"Welcome");                                redirect(

                                            else

if($cek->jabatan == "Service Advisor")

                                            redirect(

"Welcome");                                }

                                            $this->

>session->set_userdata("error",TRUE);      }

                                            $this->

>load->view("page/login");                 }

                                            public

function logout(){

                                            $this->

>session->unset_userdata("id");             $this->

                                            $this->

>session->unset_userdata("error");          $this->

                                            redirect(

>session->unset_userdata("welcome");        }

}

/* End of file Login.php */
/* Location: ./modules/controllers/Login.php */

```

```

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

/**

```

```

*
* Controller Kategori
*
* This controller for ...
*
* @package CodeIgniter
* @category Controller CI
* @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
* @param ...
* @return ...
*
*/

class Kategori extends CI_Controller
{

    var
    $kelas = "Kategori";

    function
    __construct(){
        parent::__construct();
        if
        (!$this->session->userdata("id")){
            redirect(
            "Login");
        }
    }

    public
    function index(){
        $data["rowData"] = $this->M_mst_kategori->getAll();
        $data["konten"] = "kategori/index";
        $this->load->view('template',$data);
    }

    public
    function detail($id){

```

```

header('Content-Type: application/json');

a = $this->M_mst_kategori->getDetail($id);
json_encode( $rowData );

function add(){
    $id = $this->input->post("kategoriid");
    $data["nama"] = $this->input->post("nama");

    if($id)
        $this->M_mst_kategori->update($id,$data);
    else
        $this->M_mst_kategori->add($data);

    redirect(
        $this->kelas);
}

function delete($id){
    $this->M_mst_kategori->delete($id);

    redirect(
        $this->kelas);
}

}

/* End of file Kategori.php */
/* Location: ./modules/controllers/Kategori.php */

```

```

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

/**
 *
 * Controller Faktur
 *
 * This controller for ...
 *
 * @package CodeIgniter
 * @category Controller CI
 * @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
 * @param ...
 * @return ...
 */

class Faktur extends CI_Controller
{
    var $kelas = "Faktur";

    function __construct(){
        parent::__construct();

        if (!$this->session->userdata("id")){
            redirect("Login");
        }

        public function index(){
            $data["rowData"] = $this->M_penjualan->getAll();

            $data['konten'] = "faktur/index";

            $this->load->view('template',$data);
        }
    }
}

```

```

public
function detail($id){

    $data["data"] = $faktur = $this->M_penjualan->getDetailByFk($id);

    $data["rowDetailWo"] = $this->M_workorder_detail->getAllBy("workorderid = ".$id);

    $data["rowWo"] = $wo = $this->M_workorder->getDetail($id);

    $data["rowFeedback"] = $this->M_feedback->getAllBy("penjualanid = ".$faktur->penjualanid);

    $data["rowCust"] = $this->M_pelanggan->getDetail($wo->pelangganid);

    $data['konten'] = "faktur/detail";

    $this->load->view('template',$data);
}

public
function add(){

    if($this->input->post("btnsubmit")){

        $data["idrefatribut"] = $this->input->post("idrefatribut");

        $this->M_penjualan->add($data);

        redirect($this->kelas);
    }

    $data['konten'] = "faktur/index";

```


>load->view('template',\$data);	\$this-
	}
	public
function update(\$id){	
	if(\$this-
>input->post("btnsubmit")){	
	\$data["i
drefatribut"] = \$this->input->post("idrefatribut");	
	\$this-
>M_penjualan->update(\$id,\$data);	
	redirect(
\$this->kelas);	}
	\$data["
data"] = \$this->M_penjualan->getDetail(\$id);	
	\$data['k
onten'] = "faktur/index";	
	\$this-
>load->view('template',\$data);	}
	public
function cetak(\$id){	
	\$data["t
anggal_cetak"] = date("Y-m-d");	
	//add
FEEDBACK	
	\$rowFee
dback = \$this->M_mst_feedback->getAll();	
	foreach
(\$rowFeedback as \$row) {	

edback["penjualanid"] = \$id;	\$dataFe
edback["nilai"] = \$this->input->post(\$row->mstfeedbackid);	\$dataFe
edback["mstfeedbackid"] = \$row->mstfeedbackid;	\$dataFe
>M_feedback->add(\$dataFeedback);	\$this-
	}
>M_penjualan->update(\$id,\$data);	\$this-
"Cetak/faktur/\$id");	redirect(
	}
function delete(\$id){	public
>M_penjualan->delete(\$id);	\$this-
\$this->kelas);	redirect(
	}
}	
/* End of file Faktur.php */	
/* Location: ./modules/controllers/Faktur.php */	

```

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

/**
 *
 * Controller Cetak
 *
 * This controller for ...
 */

```

```

* @package CodeIgniter
* @category Controller CI
* @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
* @param ...
* @return ...
*
*/

class Cetak extends CI_Controller
{
    function __construct(){
        parent::__construct();
        // if
        (!$this->session->userdata("id")){
            //
            redirect(
                "Login");
            // }
            $id
            = $this->session->userdata("id");
            $this->
            >user = $this->M_user->getDetail($id);
        }
        public
        function faktur($id){
            $data['filename'] = "Faktur";

            $data["
            data"] = $faktur = $this->M_penjualan->getDetailByFk($id);

            $data["r
            owDetailWo"] = $this->M_workorder_detail->getAllBy("workorderid =
            ".$id);

            $data["r
            owWo"] = $wo = $this->M_workorder->getDetail($id);

            $data["r
            owFeedback"] = $this->M_feedback->getAllBy("penjualanid = ".$faktur-
            >penjualanid);

```

```

        $data["rowCust"] = $this->M_pelanggan->getDetail($wo->pelangganid);

        $data["konten"] = "page/export/faktur";

        //
        echo json_encode($data);
        $this->load->view("page/export/templatePdf",$data);
    }

    public
    function laporanfaktur(){
        $data['filename'] = "Laporan Faktur";

        $data["rowData"] = $this->M_penjualan->getAll();

        $data["konten"] = "page/export/laporanfaktur";
        $this->load->view("page/export/templatePdf",$data);
    }

    public
    function pelanggan($id){
        $data['filename'] = "Pelanggan";

        $data["rowData"] = $this->M_pelanggan->getDetail($id);

        $data["rowData"] = $this->M_workorder->getAllBy("pelangganid = $id");

        $data["konten"] = "page/export/pelanggan";
        $this->load->view("page/export/templatePdf",$data);
    }

    public
    function workorder($id){
        $data['filename'] = "Workorder";
    }

```

```

data"] = $wo = $this->M_workorder->getDetail($id);
                                $data["r
owData"] = $this->M_workorder_detail->getAllBy("workorderid = ".$id);
                                $data["r
owCust"] = $this->M_pelanggan->getDetail($wo->pelangganid);

                                $data['k
onten'] = "page/export/workorder";
                                $this->load->view("page/export/templatePdf",$data);
                                }

                                public
function pembelian($id){
    $data['filename'] = "Pembelian";

                                $data["
data"] = $this->M_pembelian->getDetail($id);
                                $data["r
owData"] = $this->M_pembelian_detail->getAllBy("pembelianid = ".$id);
                                $data['k
onten'] = "page/export/pembelian";
                                $this->load->view("page/export/templatePdf",$data);
                                }

}

/* End of file Cetak.php */
/* Location: ./modules/controllers/Cetak.php */

```

```

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

/**
 *
 * Controller Jenis
 *
 * This controller for ...

```

```

*
* @package CodeIgniter
* @category Controller CI
* @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
* @param ...
* @return ...
*
*/

class Jenis extends CI_Controller
{
    var $kelas = "Jenis";

    function __construct()
    {
        parent::__construct();

        if (!$this->session->userdata("id")){
            redirect("Login");
        }

        public function index()
        {
            $data["rowData"] = $this->M_mst_jenis->getAll();
            $data['konten'] = "jenis/index";
            $this->load->view('template',$data);
        }

        public function detail($id){
            header('Content-Type: application/json');

            $rowDat
            a = $this->M_mst_jenis->getDetail($id);

```

json_encode(\$rowData);	echo
	}
function add(){	public
= \$this->input->post("jenisid");	\$id
nama"] = \$this->input->post("nama");	\$data["
	if(\$id)
>M_mst_jenis->update(\$id,\$data);	\$this-
	else
>M_mst_jenis->add(\$data);	\$this-
\$this->kelas);	redirect(
	}
function delete(\$id){	public
>M_mst_jenis->delete(\$id);	\$this-
\$this->kelas);	redirect(
	}
}	
/* End of file Jenis.php */	
/* Location: ./modules/controllers/Jenis.php */	

```

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

/**

```

```

*
* Controller Jenis
*
* This controller for ...
*
* @package CodeIgniter
* @category Controller CI
* @author Chandra Rizki Azhari <chandra.r.azhari@gmail.com>
* @param ...
* @return ...
*
*/

class Jenis extends CI_Controller
{
    var $kelas = "Jenis";

    function __construct(){
        parent::__construct();
        if (!$this->session->userdata("id")){
            redirect("Login");
        }
    }

    public function index(){
        $data["rowData"] = $this->M_mst_jenis->getAll();
        $data["konten"] = "jenis/index";
        $this->load->view('template',$data);
    }

    public function detail($id){

```



```

header('Content-Type: application/json');

a = $this->M_mst_jenis->getDetail($id);
json_encode( $rowData );

function add(){
    = $this->input->post("jenisid");
    nama"] = $this->input->post("nama");

    >M_mst_jenis->update($id,$data);

    >M_mst_jenis->add($data);

    $this->kelas);

function delete($id){
    >M_mst_jenis->delete($id);

    $this->kelas);

}

/* End of file Jenis.php */
/* Location: ./modules/controllers/Jenis.php */

```