

**PANDUAN LENGKAP SI SISTEM INFORMASI
PENGELOLAAN PAKET DI KAMPUS POLITEKNIK POS
INDONESIA MENGGUNAKAN EMAIL**

**PANDUAN LENGKAP SI SISTEM INFORMASI
PENGELOLAAN PAKET DI KAMPUS POLITEKNIK POS
INDONESIA MENGGUNAKAN EMAIL**

Lidwina Triniska Gulo, Student

Penulis

ISBN

Editor

Penyunting Dan

lain-lain

Quotes

“ Jika kamu menghakimi orang, Kamu tidak akan punya waktu untuk mencintai mereka.”

(Mother Teresa)

DAFTAR ISI

Quotes	i
DAFTAR ISI	v
DAFTAR GAMBAR	v
DAFTAR TABEL.....	vi
DAFTAR SYMBOL.....	vii
KATA PENGANTAR	viii
BAB I PENDAHULUAN	I
1.1 Pengenalan PHP.....	I
1.2 Pengertian PHP	I
1.3 Fungsi PHP	II
1.4 Keunggulan & Kekurangan.....	IV
BAB II LANDASAN TEORI.....	5
Pengenalan Email.....	5
2.1 Email.....	5
2.1.1 Cara Kerja Email.....	5
2.1.2 Fungsi dan Manfaat Email.....	6
2.1.3 Jenis-Jenis Email	10
2.2 Pengertian CodeIgniter (CI)	11
2.2.1 View	12
2.2.2 Model.....	12
2.2.3 Controller.....	12
2.2.4 Kerangka kerja	13
2.2.5 Fitur-fitur Codeigniter	13
2.2.6 Mengenal Konsep MVC	14

2.2.7	Sistem Kerja Codeigniter.....	15
2.2.8	Struktur Folder Codeigniter	17
2.2.9	Library	20
2.2.10	Helper	22
2.2.11	Keuntungan Codeigniter	25
2.2.12	Kekurangan Codeigniter	27
2.3	Whatsapp API WhatsApp	27
2.4	UML (Unified Modeling Language).....	29
2.5	Pengenalan XAMPP	32
2.5.1	Pengertian XAMPP	32
2.5.2	Penjelasan singkatan XAMPP	32
2.5.3	Fitur-fitur XAMPP.....	33
2.5.4	Kelebihan dan kekurangan XAMPP	36
2.6	Web.....	36
2.6.1	Beberapa pengertian web:.....	37
2.6.2	Server Online	37
2.6.3	Diakses Browser.....	38
2.6.4	Jenis Website.....	39
2.6.5	Fungsi Website	39
BAB III INSTALASI APLIKASI	41
INSTALASI APLIKASI YANG AKAN DIGUNAKAN	41
3.1	Sublime Text	41
3.2	Xampp	50
3.3	Codeigniter	57
3.4	Konfigurasi dasar pada codeigniter	64
3.5	Hello World Codeigniter	69
BAB IV METODELOGI PENELITIAN	73
METODELOGI PENELITIAN	73
4.1	Diagram Alur Metodelogi Penelitian.....	73

4.2	Tahapan-Tahapan Diagram Alur Metodelogi Penelitian	74
4.2.1	Perancangan Sistem.....	74
4.2.2	Diagram Alur Metodelogi Penelitian.....	74
4.2.3	Analisis Sistem	74
4.2.4	Perancangan Sistem	75
4.2.5	Implementasi System	75
4.2.6	Pengujian System.....	75
4.2.7	Metodelogi Waterfall	75
	BAB V ANALISIS DAN PERANCANGAN	76
	ANALISIS DAN PERANCANGAN.....	76
5.1	Analisis Dan Perancangan Sistem	76
5.1.1	Analisis Sistem yang Sedang Berjalan	76
5.1.2	Analisis Sistem yang akan Dibangun	80
5.2	Perancangan.....	83
5.2.1	Use Case Diagram	83
5.2.2	<i>Class</i> Diagram	97
5.2.3	<i>Sequence</i> Diagram	98
5.2.4	Collaboration Diagram.....	109
5.2.5	Activity Diagram	116
5.2.5.10	Activity Diagram Logout.....	126
5.2.6	Statechart Diagram	127
5.2.7	Component Diagram.....	134
5.2.8	Deployment Diagram.....	134
5.2.8.1	Deployment Diagram Software.....	135
5.2.8.2	Deployment Diagram Hardware	135
5.2.9	Struktur Menu.....	136
5.2.9.1	Struktur Menu pada Admin	137
5.2.9.2	Struktur Menu pada Petugas.....	137

BAB VI IMPLEMENTASI.....	143
IMPLEMENTASI.....	143
6.1 Implementasi	143
6.1.1 Lingkungan Implementasi.....	143
6.2 Pembahasan Hasil Implementasi.....	144
6.2.1 Halaman <i>Login</i> Admin.....	145
6.2.2 Halaman Dasboard pada Halaman Admin	145
6.2.3 Halaman Data Barang	146
6.2.4 Halaman From Data Barang.....	147
6.2.5 Halaman Data Penerima.....	148
6.2.6 Halaman From Data Penerima	148
6.2.7 Halaman Data Kategori Penerima	149
6.2.8 Halaman Data Kategori Paket	150
6.2.9 Halaman User Management	151
6.2.10 Halaman From User Management	152
6.3 Pengujian dan Hasil Pengujian.....	153
6.3.1 Identifikasi dan Rencana Pengujian.....	153
6.3.2Hasil Pengujian.....	155

DAFTAR GAMBAR

Gambar 1.1 Cara kerja Email	5
Gambar 2.2 Kegunaan Email.....	6
Gambar 3. 2 Dokumen atau File	7
Gambar 4.4 Daftar Situs Online	8
Gambar 5.5 Email Marketing	10
Gambar 6.6 Cara kerja MVC.....	15
Gambar 7.7 Alur Kerja Codeigniter.....	16
Gambar 8.8 Struktur Directory Codeigniter	19
Figure 9.9 Htdocs.....	34
Gambar 10.10 PhpMyadmin	35
Gambar 11.11 Control Panel	35
Gambar 12.12 Kumpulan File dan Folder.....	37
Gambar 13.13 Server	38
Gambar 14.14 Browser	38
Gambar 15.2 File pada Sublim Text	42
Gambar 16.3 Kode Sublime Text	43
Gambar 17.4 Preference	44
Gambar 18.5 Packages	44
Figure 19.6 File Dowload.....	45
Gambar 20.7 Setup Sublime	45
Gambar 21.8 Ekxplore	46
Gambar 22.9 Istall.....	47
Gambar 23..10 Proses Install	47
Gambar 24.11 Menu Windows.....	49
Gambar 25.12 Halaman Utama Sublime Text.....	49
Gambar 26.13 File Instalerr Xampp	51
Gambar 27.14 Setup Xampp	51
Gambar 28.15 Select Componen	52
Gambar 29.16 Installator Folder.....	53
Gambar 30.17 Bitami for Xampp	53
Gambar 31.18 Proses Install	54
Gambar 32.19 Bitnami Finish	54
Gambar 33.19 Halaman Bahasa yang digunakan	55
Gambar 34.20 Control Panel Xampp	57

Gambar 35.21 Website	59
Gambar 36.22 Ekstrak Fie	59
Gambar 37.23 Kopy File	61
Gambar 38.24 Control Panel Xampp	61
Gambar 39.25 Halaman Utama	62
Gambar 40.26 Folder pada CodeIgneter	62
Gambar 41.27 Folder Config	64
Gambar 42.28 Scirpt Autoload	66
Gambar 43.29 Script kode	66
Gambar 44.30 Kode Database	66
Gambar 45.31 Kode Url	66
Gambar 46.31 Script Konfigurasi	67
Gambar 47.32 Script Config	67
Gambar 48.33 Nama Kode File	67
Gambar 49.4 Text Editor	68
Gambar 50.35 File Database	68
Gambar 51.36 File Database	69
Gambar 52.37 Config Hello Word	70
Gambar 53.38 Script Hello Word	72
Gambar 54.39 Halaman Web Hello Word	72
Gambar 55.1 Diagram Alur Metodelogi Penelitian	73
Gambar 56.1 Flochart Buku Tamu yang sedang berjalan	77
Gambar 57.2 Flochart Buku Paket yang sedang berjalan	78
Gambar 58.3 Flochart Buku Surat yang sedang berjalan	79
gambar 59.4 Flochart Buku Tamu yang sedang dibangung	80
Gambar 60.5 Flochart Buku Paket yang sedang dibangung	81
Gambar 61.6 Flochart Buku Surat yang sedang dibangun	82
Gambar 62.7 Use Case Diagram	84
Gambar 63.8 Class Diagram	97
Gambar 64.9 Sequence Diagram Login	99
Gambar 65.10 Sequence Diagram Paket	100
Gambar 66.11 Sequence Diagram Surat	101
Gambar 67.12 Sequence Diagram Penerima	102
Gambar 68.13 Sequence Diagram Update Paket	104
Gambar 69. 14 <i>Sequence Diagram Update Surat</i>	105
Gambar 70.15 Sequence Diagram Update Penerima	106
Gambar 71.16 Sequence Diagram Tambah Petugas	107

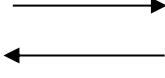
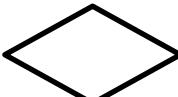
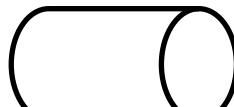
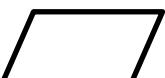
Gambar 72.17 Sequence Diagram Update Petugas	108
Gambar 73.18 Collaboration Diagram Login.....	111
Gambar 74.19 Collaboration Diagram Paket	112
Gambar 75.20 Collaboration Diagram Surat.....	113
Gambar 76..23 Collaboration Diagram update Penerima	114
Gambar 77.24 Collaboration Diagram update Petugas.....	115
Gambar 78.25 Collaboration Diagram tambah Petugas.....	116
Gambar 79.26 Activity Diagram Login	117
Gambar 80.27 Activity Diagram tambah Paket.....	118
Gambar 81.28 Activity Diagram tambah Surat	119
Gambar 82.29 Activity Diagram tambah Penerima.....	120
Gambar 83.30 Activity Diagram Update Paket.....	121
Gambar 84.31 Activity Diagram Update Surat	122
Gambar 85.32 Activity Diagram Update Penerima	123
Gambar 86.33 Activity Diagram tambah Petugas	124
Gambar 87.34 Activity Diagram Update Petugas	125
Gambar 88.35 Activity Diagram Logout	126
Gambar 89.36 Statechart Diagram Login.....	128
Gambar 90.37 Statechart Diagram tambah Paket.....	129
Gambar 91.38 Statechart Diagram tambah Surat	130
Gambar 92.39 Statechart Diagram tambah Penerima	131
Gambar 93.40 Statechart Diagram tambah Petugas.....	132
Gambar 94.41 Statechart Diagram Update Data	133
Gambar 95.42 Component Diagram	134
Gambar 96.43 Deployment Diagram Software	135
Gambar 97.44 Deployment Diagram Hardware	135
Gambar 98.1 Halaman Login Admin.....	145
Gambar 99.2 Halaman Dasboard.....	145
Gambar 100.3 Halaman Data Barang	146
Gambar 101.4 Halaman From Data Barang	147
Gambar 102.5 Halaman Data Penerima	148
Gambar 103.6 Halaman Data Penerima	148
Gambar 104.7 Halaman Data Kategori Penerima	149
Gambar 105.8 Halaman Data Kategori Paket	150
Gambar 106.9 Halaman User Management	151
Gambar 107.10 Halaman From User Management	152

DAFTAR TABEL

Table 1.1 Definisi Use Case	84
Table 2.2 Definisi Aktor.....	85
Table 3.3 Skenario Use Case Login.....	86
Table 4.4 Skenario Use Case Validasi	87
Table 5.5 Skenario Use Case Tambah Barang	88
Table 6.6 Skenario Use Case Penerima	90
Table 7.7 Skenario Use Case Notifikasi	91
Table 8.8 Skenario Use Case Update Anggota.....	92
Table 9.9 Skenario Use Case Mengkonfirmasi Penerima	94
Table 10.10 Skenario Use Case Melakukan Logout.....	95
Table 11.11 Skenario Use Case Tambah Penerima	95
Table 12.12 Skenario Use Case Update Petugas	96
Table 13.1 Spesifikasi Perangkat Lunak	143
Table 14.2 Spesifikasi Perangkat Keras	144
Table 15.3 Identifikasi dan Rencana Pengujian	153
Table 16.4 Butir Uji Buka Website.....	155
Table 17.5 Butir Uji Data Penerima	155
Table 18.6 Butir Uji Input Data Master Devisi	156
Table 19.7 Butir Uji Input Data Rekap Paket.....	157
Table 20.8 Butir Uji Input Data Rekap Per Devisi	159
Table 21.9 Butir Uji Data Laporan	159
Table 22.9 Butir Uji Data Historis	161

DAFTAR SYMBOL

Daftar Simbol pada Flowmap

Gambar	Pengertian	Keterangan
	Dokumen (<i>Document</i>)	Menunjukkan semua jenis dokumen yang digunakan untuk merekam <i>entry data</i> .
	Operasional Manual	Menunjukkan proses yang dikerjakan secara <i>manual</i> pada.
	Garis Aliran (<i>flow line</i>)	Menunjukkan arus data antar simbol.
	<i>Decision</i>	Menunjukkan keputusan dalam proses pengelolaan data.
	<i>Direct Data</i>	Digunakan untuk penyimpanan <i>database</i> .
	<i>Proses</i>	Digunakan untuk kegiatan proses dari sistem.
	<i>Input/output</i>	Digunakan untuk keterangan input/output data

Daftar Simbol pada UML

No.	Simbol	Keterangan
1.		Actor adalah elemen diluar sistem yang berkepentingan secara langsung dengan sistem
2.		Use Case adalah urutan interaksi antar aktor dengan sistem.
3.		Include adalah relasi dari <i>base use case</i> kepada <i>inclusion use case</i> , menggambarkan bagaimana perilaku dari <i>inclusion use case</i> disisipkan ke dalam <i>base use case</i>
4.		Extend adalah relasi dari <i>base use case</i> kepada <i>extension use case</i> , menggambarkan bagaimana perilaku dari <i>extension use case</i> disispkan ke dalam <i>base use case</i>
4.		Association adalah partisipasi aktor dalam suatu use case yang saling berkomunikasi satu dangan yang lainnya.
5.		System boundary , untuk merepresentasikan batasan fisik sistem dengan aktor yang berinteraksi dengan system
6.		Class adalah deskripsi dari kumpulan objek yang memiliki atribut, operation, method, relationship dan semantics yang sama
7.		Generalization adalah hubungan taksonomik antara use case yang lebih umum dan use case yang lebih spesifik
8.		Start State adalah memperlihatkan dimana aliran kerja berawal

9.		<i>End State</i> adalah memperlihatkan dimana aliran kerja berakhir
----	---	--

KATA PENGANTAR

Buku ini merupakan panduan lengkap mengenai system informasi pengelolaan paket di kampus politeknik pos Indonesia menggunakan email

BAB I

1.1 Pengenalan PHP

PHP pertama kali dibuat oleh *Rasmus Lerdorf*, seorang programmer C. Pada waktu itu PHP masih bernama *Form Interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk menghitung data *form* dari *web*. Jadi, awal mula PHP digunakannya untuk menghitung jumlah pengunjung didalam *web*-nya.

Dengan alasan untuk meningkatkan performa, Rasmus Lerdorf membuat ulang kode program tersebut dalam bahasa C. Lerdorf menyebut kode program ini sebagai *Personal Home Page*. Versi ini pertama kali keluar pada tahun 1995. Isinya adalah sekumpulan skrip PERL yang dibuatnya untuk membuat halaman *web*-nya menjadi dinamis.

Awalnya, PHP bekerja di lingkungan Linux, karena merupakan ‘turunan’ bahasa PEARL yang merupakan bahasa pemrograman native di lingkungan Linux dan Unix. Namun seiring perkembangan dan kebutuhan maka PHP dibuat secara *cross platform*. PHP dapat bekerja di lingkungan Windows maupun Linux. PHP mengeksekusi *server script* lebih cepat dari *script* manapun. PHP menempati rangking pertama dalam hal kecepatan eksesksi, diikuti oleh ASP dan JSP.

PHP difokuskan pada scripting *server-side*, jadi anda dapat melakuka apa yang bisa dilakukan CGI dengan menggunakan PHP seperti mengambil data inputan form, mengenerate konten halaman dinamis, mengirim dan menerima *cookies* dan masih banyak lagi. Kemampuan dan *support*-nya untuk *database* juga sangat dapat diandalkan.

1.2 Pengertian PHP

PHP adalah bahasa pemrogmana *script* sisi server yang didesain untuk pengembangan *web*. Dimana PHP ini merupakan singkatan dari *Hypertext Preprocessor* yang digunakan sebagai bahasa pemrograman umum. Selain itu PHP

juga digunakan bersamaan dengan bahasa pemrograman lainnya seperti bahasa pemrograman HTML, dan Javascript.

PHP disebut sebagai bahasa pemrograman *server-side* karena, hal ini berbeda dengan bahasa pemrograman *client-side* seperti Javascript yang diproses pada *web browser (client)*. PHP juga menjadi dasar dari aplikasi *Content Management System* (CMS) yang populer seperti Joomla, Drupal, dan Wordpress.

1.3 Fungsi PHP

Salah satu fungsi dari PHP ini dapat disisipkan pada dokumen HTML. Karena kemampuan inilah PHP juga sering disebut sebagai bahasa pemrograman script atau scripting language. Berikut ini merupakan sintaksis dasar pada PHP.

a. Pembatas

PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP apapun diluar pembatas tidak diproses oleh PHP. Pembatas paling umum adalah “<?php” untuk membuka dan “?>” untuk menutup kode PHP.

✓ Variabel

Variabel dalam PHP diawali dengan simbol *dollar* “\$”, kemudian diikuti dengan huruf sebagai karakter pertama. Setelah itu, dapat dilanjutkan dengan kombinasi huruf dan angka. Variabel tidak boleh mengandung spasi maupun tanda baca di dalamnya, kecuali *underscore* “_”. Variabel pada PHP bersifat “case sensitif”, yang berarti anda harus memperhatikan penulisan huruf besar dan urutan

kecil. Variabel \$jumlah berbeda dengan \$Jumlah.

Contoh penamaan variabel yang benar :

\$nama_pemakai

\$kota_3

\$user1

Contoh penamaan variabel yang salah :

\$nama pemakai

\$3kota

\$us\er1

✓ Komentar

Ada kalanya kita sebagai pemrogram, karena banyaknya kode program atau variabel dalam program , perlu menandai atau memberi komentar pada program. Komentar pada program merupakan tulisan pada program yang tidak dieksekusi. Pada PHP, ada 3 macam cara penulisannya :

- /* kometar */

Tulisan apapun berada di antara ‘/*’ akan dianggap sebagai komentar. Cara seperti ini sangat berguna dan efisien untuk pemberian komentar yang memakan banyak baris.

- // komentar

Tulisan di baris yang sama setelah ‘//’ akan dianggap sebagai komentar. Cara ini berguna untuk pemberian komentar singkat yang tak lebih dari 1 baris saja.

- # komentar

Sama seperti ‘//’, tulisan di baris yang sama setelah '#' akan dianggap sebagai komentar.

Cara ini berguna untuk pemberian komentar singkat yang tak lebih dari 1 baris saja.

1.4 Keunggulan & Kekurangan

Keunggulan bahasa pemrograman PHP, antara lain :

- a. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaanya.
- b. Web server yang mendukung PHP dapat ditemukan di mana-mana dari mulai *apache*, *IIS*, *Lightpad*, hingga *xitami* dengan konfigurasi relatif mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyak *developer* yang siap membantu dalam pengembangan.
- d. PHP adalah bahasa *open source* yang digunakan berbagai mesin seperti *Linux*, *Unix*, *Macintosh*, *Windows* dan dapat dijalankan secara runtime melalui *console* serta dapat menjalankan perintah-perintah sistem.
- e. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.

Kekurangan bahasa pemrograman PHP, antara lain :

- a. Tidak memiliki sistem pemrograman berorientasi objek yang sesungguhnya.
- b. PHP mempunyai kelemahan security tertentu yang mana jika programmer kurang teliti dalam melakukan pemrograman dan mengabaikan isu dan konfigurasi PHP.
- c. Tidak ideal jika untuk pengembangan skala besar.
- d. Tidak dapat memisahkan antara tampilan dengan logik dengan baik.

BAB II

Pengenalan Email

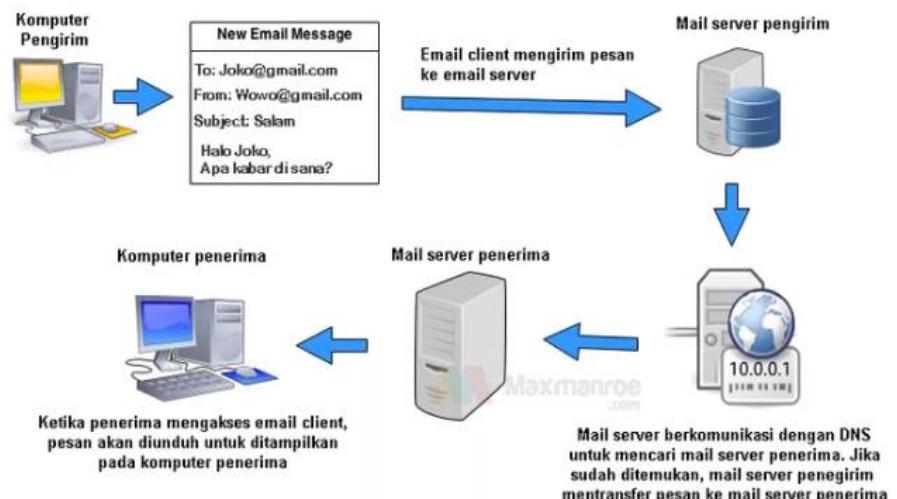
2.1 Email

Email atau surat elektronik adalah suatu sarana untuk mengirim dan menerima surat atau pesan dengan format digital melalui jalur jaringan komputer dan internet. Pesan digital atau surat elektronik tersebut dapat dibuat dan dikirim ke alamat email lainnya dengan menggunakan computer atau laptop dan gadget lainnya, seperti smartphone dan tablet yang terhubung dengan internet.

2.1.1 Cara Kerja Email

Cara kerja email sangat sederhana. Sama seperti ketika saat menulis surat kepada orang lain. Bedanya, dengan email kita bisa mengirim pesan lebih cepat dan dapat disertai dengan file tertentu yang ingin kita kirim juga.

Berikut ini adalah proses cara kerja email secara singkat:



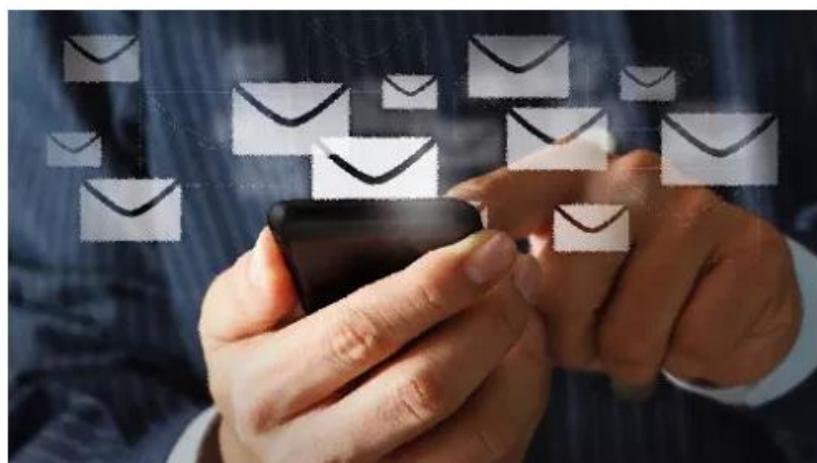
Ilustrasi cara kerja email

Gambar 1.1 Cara kerja Email

2.1.2 Fungsi dan Manfaat Email

Mengacu pada pengertian email di atas, maka fungsi dan manfaat email secara umum adalah untuk mengirimkan pesan atau surat kepada orang lain. Lebih lanjut, berikut ini adalah beberapa fungsi email selengkapnya:

1.4.1.1 Mengirim Pesan



Gambar 2.2 Kegunaan Email

Kegunaan email yang paling umum adalah untuk mengirim dan menerima email dari orang lain di manapun mereka berada. Pesan tersebut dapat terkirim ke tujuan dalam waktu yang sangat cepat, bahkan dalam hitungan detik saja. Namun, tentu saja kecepatan tersebut dipengaruhi oleh kecepatan internet, baik si pengirim atau si penerima email.

1.4.1.2 Mengirim dan menerima File



Gambar 3. 2 Dokumen atau File

Email juga dapat mengirimkan berbagai jenis file dan dokumen digital dengan ukuran tertentu, baik itu foto, video, teks, dan lain-lain. Biasanya file tersebut ditambahkan di dalam lampiran (attachment) pada email.

1.4.1.3 Sebagai Identitas Diri

Email juga berfungsi sebagai identitas diri di internet dimana orang lain dapat menghubungi pengguna email melalui alamat email miliknya.

Dalam berbagai aspek, penggunaan email bagi seseorang sudah menjadi kewajiban. Email seringkali menjadi syarat mutlak untuk proses aktivasi berbagai layanan digital, misalnya untuk aktivasi smartphone, untuk aktivasi layanan berlangganan surat, dan lain-lain.

1.4.1.4 Email Untuk Mendaftar di Berbagai Situs



Gambar 4.4 Daftar Situs Online

Sebagian besar layanan website saat ini membutuhkan email penggunanya agar bisa melakukan aktivitas di website tersebut. Mulai dari situs media sosia, situs e-commerce, game online, situs forum, dan berbagai layanan lainnya, mengharuskan pengguna mendaftarkan email.

Berbagai layanan website tersebut membutuhkan verifikasi dari penggunanya, dan ini hanya bisa dilakukan bila penggunanya memiliki alamat email yang valid.

1.4. 1.5 Email untuk Media Pemasaran



Gambar 5.5 Email Marketing

Pemasaran online saat ini banyak dilakukan dengan memanfaatkan email. Dan sebagian besar pebisnis online melakukan pemasaran melalui email untuk meningkatkan penjualan.

2.1.3 Jenis-Jenis Email

a. POP Mail

Pengertian POP mail (*Post Office Protocol*) adalah jenis email yang didapatkan oleh pengguna dari ISP yang menjadi langganan atau alamat email pekerja yang diberikan perusahaan kepada pegawainya.

b. Webmail

Pengertian webmail adalah email yang berbasis website. Dengan kata lain, pengguna email hanya bisa membuka email miliknya maka ia harus membuka halaman situs penyedia layanan email tersebut.

c. Forward Mail

Pengertian forward mail adalah jenis layanan email yang memforward atau meneruskan email yang diterima pengguna ke alamat email yang lain. Biasanya forward mail digunakan oleh mereka yang sering berganti alamat email.

2.2 Pengertian CodeIgniter (CI)

Codeigniter merupakan sebuah *web framework* yang dikembangkan oleh Rick Ellis dari Ellis Lab. Codeigniter dirancang untuk menjadi sebuah web framework yang ringan dan mudah untuk digunakan. Bahkan pengakuan dari Rasmus Lerdorf, sang pencipta bahasa pemrograman PHP mengatakan bahwa Codeigniter merupakan web framework yang mudah, cepat dan handal.

Codeigniter dirilis pertama kali pada tanggal 28 Februari 2006, namun pada bulan Juli 2013 Ellis Lab mengumumkan bahwa mereka mencari pemilik baru untuk Codeigniter karena pada lingkup internal sendiri tidak memiliki cukup fokus untuk mengembangkan Codeigniter, dan pada akhirnya pada bulan Oktober 2014, kepemilikan Codeigniter berpindah ke tangan British Columbia Institute of Technology, yakni salah satu sekolah tinggi teknologi di Kanada.

CodeIgniter adalah aplikasi open source yang merupakan framework dengan konsep model MVC (Model, View, Controller) untuk membangun sebuah website dinamis dengan menggunakan PHP. CodeIgniter memudahkan developer dalam membuat sebuah aplikasi web dengan cepat dan mudah dibandingkan membuat dari awal.

CodeIgniter juga merupakan sebuah Framework untuk membuat sebuah aplikasi dinamis menggunakan PHP.

Model View Controller merupakan suatu konsep dalam pembangunan sebuah aplikasi web, berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, user interface, dan bagian yang

menjadi kontrol aplikasi.

Ada 3 jenis komponen yang membangun MVC dalam membuat aplikasi yaitu:

2.2.1 View

Merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data kepada user. Bagian ini tidak memiliki akses langsung terhadap bagian model.

2.2.2 Model

Biasanya berhubungan langsung dengan database untuk memanipulasi data (insert, update, delete, search), menangani validasi dari bagian controller, tetapi tidak dapat berhubungan langsung dengan bagian view.

2.2.3 Controller

Merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, controller berfungsi menerima request dan data user kemudian menentukan apa yang akan diproses oleh aplikasi.

Setelah hampir lima bulan lamanya sejak peralihan kepemilikan, BCIT akhirnya merilis Codeigniter 3.0. Dibanding versi sebelumnya tentunya Codeigniter 3 mempunya fitur yang lebih kaya seperti pengembangan *Database Driver*, terdapat pustaka yang baru dan juga PDO Codeigniter kini telah berfungsi secara penuh dengan subdriver.

Sebelum mencoba Codeigniter, perlu kita ketahui istilah *web framework* itu sendiri. Menurut *Microsoft Computer Dictionary*, *web* adalah sekumpulan

dokumen yang saling terhubung dalam sistem *hypertext* yang penggunanya akan menjelajahi web melalui halaman beranda. Sedangkan *framework* adalah desain struktur dasar yang dapat digunakan kembali (*reusable*) yang terdiri dari *abstract class* dan *concrete class* di pemrograman yang berorientasi objek.

Dirilis pada dokumentasi Codeigniter, Codeigniter merupakan toolkit bagi orang yang ingin membangun aplikasi *web* menggunakan PHP. Tujuannya adalah membuat pengembangan proyek menjadi lebih cepat dibandingkan dengan menulis kode dari awal (*stracth*). Codeigniter menyediakan kumpulan *library* untuk tugas-tugas yang sering dilakukan dan sangat mudah untuk mengakses *library* yang tersedia di Codeigniter. Dengan Codeigniter, kita cukup fokus pada pengembangan *project* dan meminimalisir kode yang akan dituliskan.

2.2.4 Kerangka kerja

Ada beberapa alasan mengapa menggunakan Kerangka kerja:

1. Mempercepat dan mempermudah pembangunan sebuah aplikasi web
2. Relatively mempermudah dalam proses maintenance karena sudah ada pola tertentu dalam sebuah kerangka.
3. Umumnya kerangka kerja menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal.
4. Lebih bebas dalam pengembangan jika dibandingkan CMS

2.2.5 Fitur-fitur Codeigniter

Menilai baik dan buruknya sebuah aplikasi dari fiturnya, bukanlah sebuah cara yang tepat. Karena ini tidak menggambarkan bagaimana sebuah aplikasi dirancang. Fitur-fitur tersebut tidak menggambarkan kualitas kode, performance dan keamanan (*security*). Satu-satunya cara untuk mengetahui yaitu dengan melihat kodennya.

Berikut ini adalah fitur-fitur yang didukung oleh Codeigniter :

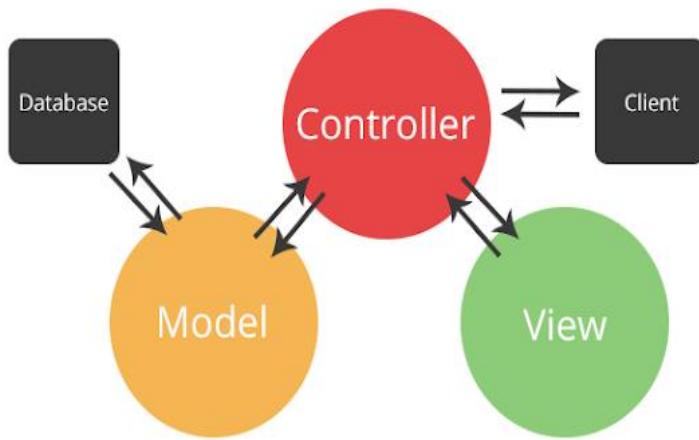
1. Model-View-Controller Based
2. PHP 7 Compatible
3. Extremely Light Weight
4. Full Featured database classes with support for several platform
5. Active Record Database Support
6. Forms and Data Validation
7. Security and XSS Filtering
8. Session Management
9. Email Sending Class, Support Attachment, HTML/Text email, multiple protocols (sendmail, SMTP, and Mail)
10. Image Manipulation Library (cropping, resizing, rotation) Support GD, Image Magic, and NetPBM
11. File Uploading Class
12. FTP Class
13. Localization
14. PAGINATION
15. Data Encryption
16. Dan lainnya.

2.2.6 Mengenal Konsep MVC

Codeigniter menggunakan pendekatan Model-View-Controller, yang bertujuan untuk memisahkan logika dan presentasi. Konsep ini mempunyai keunggulan dimana desainer dapat bekerja pada template file, sehingga redundansi kode presentasi dapat diperkecil. Berikut adalah konsep Model-View-Controller yang diterapkan pada Codeigniter.

1. Model, menggambarkan struktur data. Biasanya kelas model akan berisi fungsi yang digunakan untuk mengambil, menambah dan memperbaharui informasi yang ada didatabase.

2. View, informasi yang diperlihatkan kepada user. View adalah halaman web yang terdiri dari HTML, CSS dan Javascript, akan tetapi pada codeigniter, view dapat juga sebagai potongan halaman seperti header atau footer. Bahkan dapat juga halaman RSS atau tipe halaman lainnya.
3. Controller, perantara Model, View, dan resource lainnya yang dibutuhkan untuk menangani HTTP request dan menghasilkan halaman web.



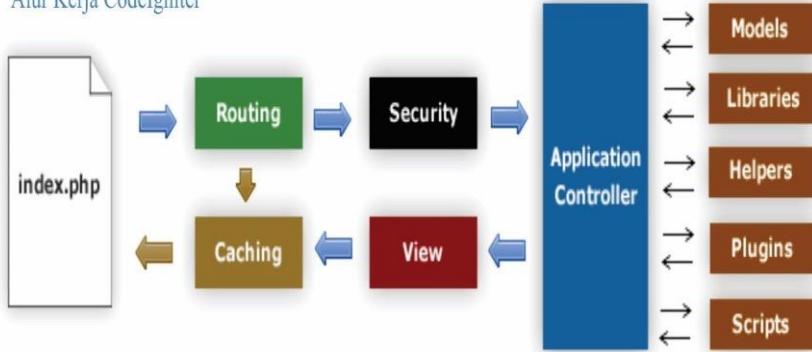
Gambar 6.6 Cara kerja MVC

Pada Codeigniter, programmer juga dapat mengabaikan pemakaian Model dan cukup menggunakan Controller dan View. Cara tersebut sah-sah saja dipakai, tetapi akan lebih baik jika kita memakai Model juga agar hierarki MVC tercapai dan disamping itu beban di Controller akan berkurang

2.2.7 Sistem Kerja Codeigniter

Untuk melengkapi pemahaman mengenai Codeigniter, berikut adalah gambaran sederhana yang menjelaskan bagaimana Codeigniter bekerja :

Alur Kerja CodeIgniter



Gambar 7.7 Alur Kerja Codeigniter

- a. Index.php: berfungsi sebagai file pertama dalam program yang akan dibaca oleh program.
- b. Router : router akan memeriksa HTTP request untuk menentukan hal apa yang harus dilakukan oleh program.
- c. Cache File : apabila dalam program sudah terdapat “cache file” maka file tersebut akan langsung dikirim ke browser. File cache inilah yang dapat membuat sebuah website dapat dibuka dengan lebih cepat. Cache file dapat melewati proses yang sebenarnya harus dilakukan oleh program codeigniter.
- d. Security : Sebelum controller di load secara keseluruhan , maka data yang disubmit oleh user dalam bentuk request HTTP akan di periksa terlebih dahulu melalui security yang dimiliki oleh Codeigniter.
- e. Controller : controller akan membuka file bagian model, core libraries, helper dan semua resources yang dibutuhkan dalam program tersebut.
- f. View: bagia akhir akan dilakukan pengecekan semua program yang ada dalam view kemudian file akan mengirimkannya ke browser supaya dapat dilihat. Apabila file view sudah ada yang di “cache” maka file view baru yang belum ter-cache akan update file view yang sudah ada.

2.2.8 Struktur Folder Codeigniter

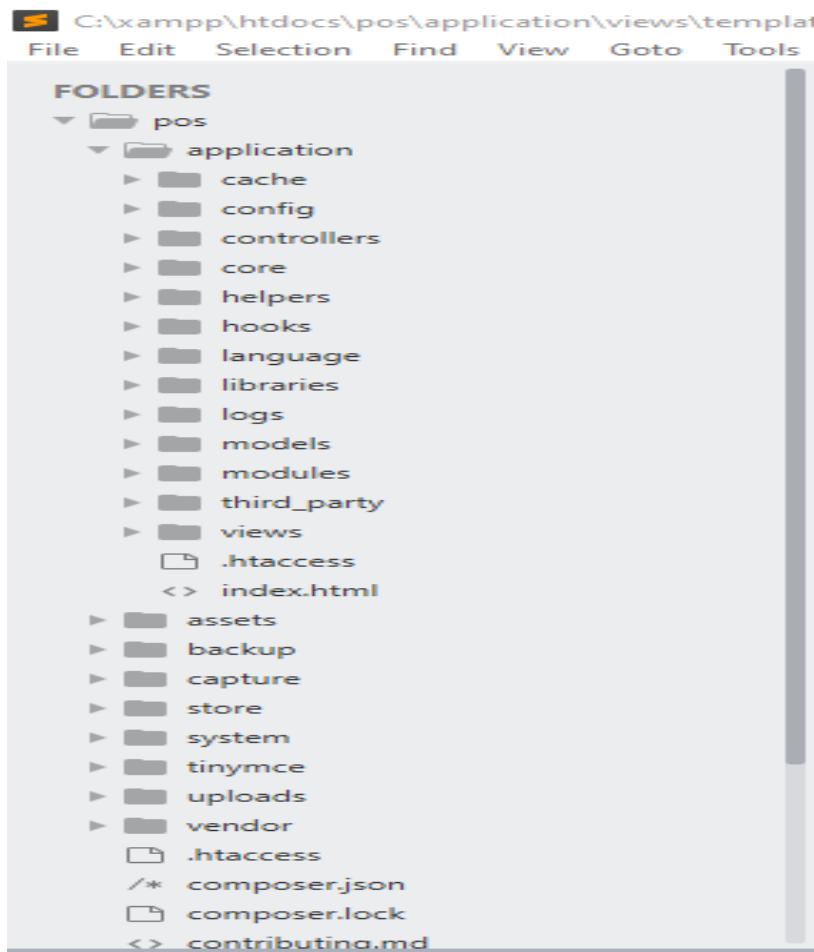
Pada Codeigniter terdapat hirarki yang dikepalai oleh tiga folder utama, yaitu application, system, dan user_guide. Agar aplikasi semakin ringkas maka folder user_guide lebih baik dihapus saja, sehingga kita akan memakai 2 folder di dalam folder Codeigniter, yaitu application dan system. Di dalam folder application terdapat banyak sub-folder, folder-folder tersebut berguna untuk menyimpan berbagai jenis file sesuai dengan nama foldernya, misalnya file-file model akan disimpan di dalam folder models, file-file view akan disimpan di folder views, dan seterusnya.

Adapun susunan folder Codeigniter secara default adalah sebagai berikut :

1. Folder application, disini aplikasi yang akan kita bangun diletakkan.
 - a. Folder cache, tempat menyimpan semua cache yang dibuat caching library.
 - b. Folder config, tempat menyimpan semua file konfigurasi yang ada di dalam aplikasi, mulai dari database, router dan autoload aplikasi.
 - c. Folder controllers, tempat menyimpan semua file controller.
 - d. Folder core, class-class utama yang sangat vital bagi Codeigniter. Jika ada salah satu atau beberapa class yang termasuk core library tidak ada, maka Codeigniter tidak dapat berfungsi dengan baik.
 - e. Folder helpers, tempat menyimpan helper-helper yang bukan berasal dari Codeigniter. Gunanya untuk melakukan tugas tertentu yang sederhana, misalnya memformat tanggal, membuat link, membuat input form dan sebagainya.
 - f. Folder hooks, tempat menyimpan hook yang digunakan mengubah alur fungsi dari core Codeigniter.
 - g. Folder language, tempat menyimpan bahasa-bahasa yang akan digunakan.
 - h. Folder libraries, tempat menyimpan semua library buatan kita sendiri.

- i. Folder logs, tempat menyimpan semua logs generated oleh CI.
 - j. Folder models, tempat menyimpan semua model.
 - k. Folder third_party, file ini adalah library pihak ketiga. Maksudnya library yang berasal dari sumber lain (bukan berasal dari Codeigniter). Misalnya untuk membuat laporan dengan format PDF, maka kita perlu memerlukan library yang dapat merubah string HTML menjadi PDF.
 - l. Folder views, tempat menyimpan semua file view aplikasi.
2. Folder system, menyimpan semua file system pada core *framework*.
- a. Folder core, tempat menyimpan semua class library bawaan Codeigniter.
 - b. Folder database, tempat menyimpan semua driver database drivers dan class yang akan digunakan.
 - c. Folder fonts, tempat menyimpan semua font yang digunakan *image manipulation* library.
 - d. Folder helpers, tempat menyimpan semua helper core Codeigniter.
 - e. Folder language, tempat menyimpan file bahasa.
 - f. Folder libraries, tempat menyimpan semua library.

Untuk lebih jelasnya mengenai susunan folder di Codeigniter, anda dapat melihat pada gambar xx.



Gambar 8.8 Struktur Directory Codeigniter

Bila anda perhatikan pada Gambar xx di dalam folder application ada nama folder yang sama yaitu core, helpers, libraries.

1. Folder core, helper dan libraries yang ada di folder system tempat semua class library dan helper asli yang terbawa oleh Codeigniter.
2. Folder core, helpers dan libraries yang ada di folder application adalah tempat untuk semua custom class library dan custom helper yang akan mungkin akan dibuat oleh programmer. Dengan adanya custom library dan custom helper, maka kita dapat membuat helper atau library sendiri.

3. Hal ini terjadi karena terkadang library atau helper bawaan Codeigniter tidak menyediakan fungsionalitas yang dibutuhkan.

Folder system berisi file-file inti Codeigniter, sebaiknya anda tidak merubah ubah isi di dalam folder ini. Pada prakteknya, jika kita membuat aplikasi memakai Codeigniter maka sebagian besar waktu akan dihabiskan untuk bekerja dengan folder application dan sub folder yang ada di dalamnya.

2.2.9 Library

Library Codeigniter membantu anda dalam mengerjakan operasi tertentu dengan lebih cepat dan efisien. Tiap library adalah sebuah class PHP dengan berbagai method yang dapat digunakan begitu class tersebut di load oleh Controller. Beberapa class sangat berguna, dan beberapa diantaranya mungkin akan selalu anda butuhkan di berbagai Controller

Jika demikian, class yang termasuk dalam kategori ini sebaiknya di set autoload (dijalankan secara otomatis) misalnya library database ataupun library session.

Library-library yang terdapat dalam Codeigniter di antaranya adalah :

1. Benchmarking, library benchmarking selalu aktif. Dapat digunakan untuk menghitung penggunaan memory.
2. Calendaring, library ini digunakan untuk membuat kalender bulan dan tahun tertentu.
3. Config, library config diinisialisasi secara otomatis oleh system. Digunakan untuk menerima informasi mengenali konfigurasi system.
4. Database, library database sangat berguna sekali jika aplikasi terhubung dengan database, dilengkapi dengan berbagai method untuk memanipulasi database.
5. Email, digunakan untuk mengirim email.

6. Encryption, berguna untuk operasi enkripsi.
7. File uploading. Library digunakan untuk melakukan upload file. Menyediakan fitur untuk validasi, agar file sesuai dengan *mime type*, ukuran (size dalam kilobyte) atau dimensi untuk gambar.
8. FTP, library FTP digunakan untuk mentransfer file ke semua server.
9. HTML Table, jika anda membutuhkan cara yang cepat untuk menggenerate table html, maka gunakan library ini. Library html table akan menggenerate table dari array atau hasil dari query database.
10. Image Manipulation, digunakan untuk meresize, membuat thumbnail, crop atau rotate gambar, serta watermark. Beberapa fungsi membutuhkan support PHP yang lebih baik, misalnya GD/GD2.
11. Input and Security, digunakan untuk pre-process input data dari form dan URL serta untuk menghandle beberapa fungsi security (misalnya untuk menghadapi serangan XSS attacks).
12. Language, library ini digunakan untuk meload beberapa bahasa yang berbeda.
13. Loader, library ini secara otomatis diload oleh system. Anda akan sering menggunakan library ini untuk meload view di dalam controller, selain itu juga untuk meload libraries.
14. Output, otomatis diload oleh system. Library ini mempunyai sebuah fungsi utama yaitu mengirimkan halaman web ke browser yang merequest. Juga digunakan untuk proses caching.
15. Pagination, digunakan untuk membuat pagination dari hasil query database. Dapat diatur beberapa record yang ingin ditampilkan dalam satu halaman. Anda juga dapat mengatur tampilan pagination.
16. Session, digunakan untuk *maintenance state* seorang user. Library ini tidak menggunakan session milik PHP, tetapi menghasilkan data untuk session sendiri.

17. Template Parser, digunakan untuk membuat template yang berisi pseudocode. Jika anda pernah menggunakan Smarty, maka Template Parser Codeigniter tidak selengkap dan sepowerfull Smarty tetapi cukup berguna.
18. Trackback, digunakan untuk mengirim dan menerima data Trackback.
19. Unit Testing, library ini digunakan untuk unit testing.
20. URI Class, library ini diload secara otomatis oleh system. Digunakan untuk memparser URL, membaginya ke dalam segment-segment yang dapat dilewatkan melalui Controller atau disimpan sebagai variabel .
21. User Agent, digunakan untuk mengidentifikasi browser, mobile device atau robot (misalnya milik search engine) yang berkunjung ke situs anda.
22. Form Validation, sebuah class yang sangat berguna. Digunakan untuk memvalidasi input form.
23. XML-RPC, digunakan untuk menseting XML-RPX client dan server.
24. Zip Encoding, digunakan untuk membuat arsip Zip untuk text dan data binary.

Untuk lebih jelasnya mengenai library pada Codeigniter, anda dapat melihat user_guide nya

2.2.10 Helper

Helper adalah file-file program pada Codeigniter yang berisi fungsi -fungsi bantuan yang dapat dipakai untuk mengerjakan tugas tertentu. Isi dari helper adalah fungsi-fungsi biasa, bukan class. File-file native helper (helper asli bawaan Codeigniter) tersimpan di system/helpers.

Codeigniter memiliki beberapa helper, namun di dalam buku ini kita tidak membahas secara detail penggunaan dari masing-masing helper tersebut. Kita hanya akan membahas manfaat helper tersebut dan cara memanggilnya. Untuk mengetahui fungsi-fungsi yang ada pada suatu helper secara detail, anda dapat menemukannya pada user_guide di bagian helper yang dimaksud.

Untuk memanggil suatu helper, anda dapat menggunakan perintah :

```
$this→load→helper('nama_helper');
```

Jika ingin memanggil beberapa helper sekaligus, maka nama helper diberikan dalam suatu array :

```
$helpers = ['helper1', 'helper', 'helper3'];
```

```
$this→load→helper($helpers);
```

Atau dibisa dipanggil secara langsung :

```
$this→load→helper(['helper1', 'helper2', 'helper3']);
```

Berikut ini adalah beberapa helper yang ada di dalam Codeigniter :

1. Array Helper, berisi fungsi-fungsi yang berkaitan dengan array :

```
$this→load→helper('array');
```

2. Captcha Helper, berisi fungsi-fungsi untuk pembuatan captcha. Captcha adalah cara untuk membedakan apakah user adalah seseorang atau bot/robot. Biasanya dalam bentuk karakter gabungan angka ataupun huruf ataupun teka-teki soal untuk diisikan ke dalam suatu input *text*.

3. Captcha yang dihasilkan helper ini merupakan captcha dasar berupa gambar berisi campuran antara huruf dan angka :

```
$this→load→helper('captcha');
```

4. Cookie Helper, berisi fungsi-fungsi yang berkaitan dengan pembuatan cookie:

```
$this→load→helper('cookie');
```

5. Date Helper, berisi fungsi-fungsi yang berkaitan dengan waktu dan tanggal :

```
$this→load→helper('date');
```

6. Directory Helper, berisi fungsi-fungsi berkaitan dengan folder :

```
$this→load→helper('directory');
```

7. Download Helper, manfaat utama helper ini adalah untuk membuat force download, yaitu suatu fungsi yang akan memaksa user untuk mendownload suatu file ketika link menuju suatu file di-klik oleh user.

```
$this→load→helper('download');
```

8. Email Helper, berisi fungsi-fungsi yang berkaitan dengan pengiriman email :

```
$this→load→helper('email');
```

9. File Helper, berisi fungsi-fungsi yang berkaitan dengan file, misalnya membaca isi file text, menyimpan file text dan sebagainya :

```
$this→load→helper('file');
```

10. HTML Helper, berisi fungsi-fungsi yang berkaitan dengan pembuatan element-element HTML :

```
$this→load→helper('html');
```

11. Inflector Helper, berisi fungsi-fungsi yang berkaitan dengan penformatan untuk string , misalnya membuat bentuk plural dari string “book” menjadi “books”, atau sebaliknya. Bisa juga untuk mengubah string ke dalam bentuk camelCase :

```
$this→load→helper('inflector');
```

12. Language Helper, berisi fungsi-fungsi yang berkaitan dengan bahasa :

```
$this→load→helper('language');
```

13. Number Helper, berisi fungsi-fungsi yang berkaitan dengan angka, misalnya untuk memformat satuan sata dalam bytes :

```
$this→load→helper('number');
```

14. Path Helper, berisi fungsi yang berkaitan dengan file path server :

```
$this→load→helper('path');
```

15. Security Helper, berisi fungsi yang berkaitan dengan keamanan, misalnya terkait dengan XSS Filtering :

```
$this→load→helper('security');
```

16. Smiley Helper, berisi fungsi-fungsi yang membantu memanage emoticons :

```
$this→load→helper('smiley');
```

17. String Helper, berisi fungsi yang berkaitan dengan string, misalnya untuk menghapus double slash pada alamat URL, atau untuk menghasilkan random string dan sebagainya

```
$this→load→helper('string');
```

18. Text Helper, berisi fungsi-fungsi yang berkaitan dengan teks. Misalnya untuk membatasi jumlah kata yang akan ditampilkan pada suatu artikel, membatasi jumlah karakter, menjalankan fungsi sensor yang mengantikan daftar kata-kata yang terlarang dengan kata-kata pengganti lainnya, dan sebagainya :

```
$this→load→helper('text');
```

19. Typography Helper, berisi fungsi-fungsi untuk memformat teks agar tepat secara semantic, misalnya mengubah / ganti baris menjadi tag
, mengubah suatu karakter ke dalam entitas HTML dan sebagainya :

```
$this→load→helper('typography');
```

20. URL Helper, berisi fungsi-fungsi yang berkaitan dengan URL. Misalnya untuk membuat link, mendapatkan nilai base url, mendapatkan nilai URL, melakukan redirect, dan sebagainya :

```
$this→load→helper('url');
```

21. XML Helper, berisi fungsi untuk bekerja dengan XML :

```
$this→load→helper('xml');
```

2.2.11 Keuntungan Codeigniter

Sebagai *web framework* populer yang menggunakan bahasa pemrograman PHP, Codeigniter mempunyai beberapa keunggulan seperti yang telah disebutkan di dokumentasinya, keunggulan-keunggulan tersebut meliputi :

1. *Free*, karena berada di bawah lisensi open source, kita dapat melakukan apapun dengan Codeigniter. Lisensi lengkapnya dapat dokumentasi.

2. *Light weight*, sistem inti Codeigniter memerlukan library yang sedikit. Sangat berbeda dengan *framework* lainnya yang membutuhkan banyak sumber daya tambahan. *Library* tambahan akan digunakan ketika *request* secara dinamis, membuat *system* yang dibangun menjadi efisien dan cukup cepat.
3. *Fast*, menurut dokumentasi, performa yang dimiliki Codeigniter terbukti cepat setelah dibandingkan dengan framework lainnya.
4. Menggunakan kaidah MVC, dengan menggunakan Model-View-Controller, kita dapat memisahkan bagian logic dan presentasian dari aplikasi yang kita bangun. Hal ini tentu sangat cocok dan bagus untuk proyek yang memfokuskan. Desainer fokus pada template file dan programmer fokus pada pembangunan logic dari aplikasi yang dibangun.
5. Menghasilkan URL yang bersih, URL yang dihasilkan oleh Codeigniter bersih dan ramah terhadap *search engine*. Codeigniter menggunakan pendekatan *segment-based* dibandingkan dengan query string yang biasa digunakan oleh programmer yang tidak menggunakan *web framework*.
6. *Packs a Punch*, Codeigniter hadir dengan berbagai *library* yang akan membantu tugas-tugas di pengembangan web yang sudah umum dan sering dilakukan seperti mengakses database, mengirim email, validasi data dari form, mengelola *session*, memanipulasi gambar, dan masih banyak lagi.
7. *Extensible*, kita dapat menambahkan library atau helper yang kita ciptakan sendiri ke dalam Codeigniter. Selain itu kita dapat juga menambahkan fitur lewat class *extensions* atau system hooks.
8. *Thoroughly Documented*, hampir semua fitur, *library* dan helper yang ada di Codeigniter telah terdokumentasi dengan lengkap dan tersusun dengan baik. Ketika mendapatkan unduhan Codeigniter, dokumentasinya sudah tersedia dan siap digunakan.
9. Mempunyai komunitas yang ramah, bergabung di komunitas Codeigniter tentunya akan membantu sekali para pengguna Codeigniter yang masih

10. pemula atau yang sudah mahir untuk saling berbagai ilmu pengetahuan. Komunitas ini dapat ditemui di codeigniter.com/forums.

2.2.12 Kekurangan Codeigniter

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

2.3 Whatsapp API WhatsApp

Messengeratau WhatsApp adalah aplikasi pesan untuk smartphone dengan basic mirip BlackBerry Messenger. WhatsApp Messengermerupakan aplikasi pesan lintas platform yang memungkinkan kita bertukar pesan tanpa biaya sms karena WhatsApp Messenger menggunakan paket data internet yang sama untuk email, browsing web, dan lain-lain. Aplikasi WhatsApp Messenger menggunakan koneksi internet 3G, 4G atau WiFi untuk komunikasi data. Dengan menggunakan WhatsApp, kita dapat melakukan obrolan online, berbagi file, bertukar foto dan lain-lain.

API adalah aplikasi pemrograman yang secara khusus dikembangkan untuk digunakan sebagai perantara komunikasi antara komponen-komponen perangkat lunak. Biasanya, hasiloutput dari API dapat berupa data XML ataupun JSON, tergantung dari situs mana yang menyediakan APItersebut (Hasan,La Ode Malik ,dkk.2016) [2].

2.4 UML (Unified Modeling Language)

UML (*Unified Modeling Language*) adalah sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem [5]. Bagian-bagian UML:

a) *View*

View digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. *View* bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram.

Beberapa jenis *view* dalam UML antara lain: *use case view*, *logical view*, *component view*, *concurrency view* dan *deployment view*.

b) *Use case view*

Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan *external actors*. *Actor* yang berinteraksi dengan sistem dapat berupa *user* atau sistem lainnya. *View* ini digambarkan dalam *use case diagrams* dan kadang-kadang dengan *activity diagrams*. *View* ini digunakan terutama untuk pelanggan, perancang (*designer*), pengembang (*developer*), dan penguji sistem (*tester*).

c) *Logical view*

Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (*class*, *object*, dan *relationship*) dan kolaborasi dinamis yang terjadi ketika *object* mengirim pesan ke *object* lain dalam suatu fungsi tertentu. *View* ini digambarkan dalam *class diagrams* untuk struktur statis dan dalam *state*, *sequence*, *collaboration*, dan *activity* diagram untuk model dinamisnya. *View* ini digunakan untuk perancang (*designer*) dan pengembang (*developer*).

d) *Component view*

Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari *code module* diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi

administrative lainnya. *View* ini digambarkan dalam *component view* dan digunakan untuk pengembang (*developer*).

e) *Concurrency view*

Membagi sistem ke dalam proses dan prosesor. *View* ini digambarkan dalam diagram dinamis (*state, sequence, collaboration* dan *activity diagrams*) dan diagram implementasi (*component* dan *deployment* diagrams) serta digunakan untuk pengembang (*developer*), pengintegrasи (*integrator*), dan penguji (*tester*).

f) *Deployment view*

Mendeskripsikan fisik dari sistem seperti komputer dan perangkat (*nodes*) dan bagaimana hubungannya dengan lainnya. *View* ini digambarkan dalam *deployment diagram* dan digunakan untuk pengembang (*developer*), pengintegrasи (*integrator*), dan penguji (*tester*).

g) Diagram

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Adapun jenis diagram antara lain:

1) *Use Case Diagram*

Use case adalah abstraksi dari interaksi antara *system* dan *actor*. *Use case* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah *system* dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah *system* dipakai. *Use case* merupakan konstruksi untuk mendeskripsikan bagaimana *system* akan terlihat di mata *user*. Sedangkan *use case diagram* memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan *client*.

2) *Class Diagram*

Class adalah dekripsi kelompok *obyek-obyek* dengan *property*, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya *class diagram* dapat memberikan pandangan global atas sebuah *system*. Hal tersebut tercermin dari *class-class* yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem

biasanya mempunyai beberapa *class* diagram. *Class* diagram sangat membantu dalam visualisasi struktur kelas dari suatu *system*

3) *Component* Diagram

Component software merupakan bagian fisik dari sebuah *system*, karena menetap di komputer tidak berada di benak para analis. Komponen merupakan implementasi *software* dari sebuah atau lebih *class*. Komponen dapat berupa *source code*, komponen biner atau *executable component*. Sebuah komponen berisi informasi tentang *logic class* atau *class* yang diimplementasikan sehingga membuat pemetaan dari *logical view* ke *component view*. Sehingga *component diagram* merepresentasikan dunia riil yaitu *component software* yang mengandung *component*, *interface* dan *relationship*.

4) *Deployment* Diagram

Menggambarkan tata letak sebuah *system* secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware*, menunjukkan hubungan komputer dengan perangkat (*nodes*) satu sama lain dan jenis hubungannya. Di dalam *nodes*, *executeable component* dan *object* yang dilokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh *node* tertentu dan ketergantungan komponen.

5) *State* Diagram

Menggambarkan semua *state* (kondisi) yang dimiliki oleh suatu *object* dari suatu *class* dan keadaan yang menyebabkan *state* berubah. Kejadian dapat berupa *object* lain yang mengirim pesan. *State class* tidak digambarkan untuk semua *class*, hanya yang mempunyai sejumlah *state* yang terdefinisi dengan baik dan kondisi *class* berubah oleh *state* yang berbeda.

6) *Sequence* Diagram

Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object*, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

7) *Collaboration Diagram*

Menggambarkan kolaborasi dinamis seperti *sequence* diagrams. Dalam menunjukkan pertukaran pesan, *collaboration* diagrams menggambarkan *object* dan hubungannya (mengacu ke konteks). Jika penekannya pada waktu atau urutan gunakan *sequence* diagrams, tapi jika penekannya pada konteks gunakan *collaboration* diagram.

8) *Activity Diagram*

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.

2.5 Pengenalan XAMPP

XAMPP adalah pengembangan dari LAMP (Linux Apache, MySQL, PHP and PERL). XAMPP didirikan oleh Kai ‘Oswald’ seidler dan Kay Vogelgesang pada tahun 2002. Xampp tersebut merupakan salah satu project non-profit, project ini bertujuan untuk mempromosikan Apache web server kepada publik.

2.5.1 Pengertian XAMPP

XAMPP yaitu perangkat lunak yang mendukung banyak system operasi, merupakan kompilasi dari beberapa program. XAMPP merupakan tools yang menyediakan paket perangkat lunak. Dengan menginstall XAMPP maka tidak perlu melakukan kembali instalasi, konfigurasi web server Apache seperti PHP dan MySQL secara manual. XAMPP akan menginstallasi dan mengkonfigurasiknya secara otomatis. XAMPP merupakan salah satu paket installasi Apache, PHP dan MySQL yang dapat digunakan untuk membantu proses installasi ketiga aplikasi tersebut. Selain paket installasi instant XAMPP versi 1.6.4 juga memberikan fasiltias pilihan pengunaan PHP v4.0 atau PHP v5.0.

2.5.2 Penjelasan singkatan XAMPP

- X : Program ini bisa dijalankan di banyak sistem operasi seperti Windows, Linux, Mac OS, dan Solaris.

- A : Apache, server aplikasi Web.
 - M : MySQL, merupakan aplikasi database yang berfungsi untuk mengolah/menyimpan data.
 - P : PHP, merupakan bahasa pemrograman web.
 - P : Perl, merupakan bahasa pemrograman untuk semua tujuan.

2.5.3 Fitur-fitur XAMPP

Berikut ini merupakan beberapa fitur-fitur dari xampp :

1. Apache

Apache dapat diartikan sebagai perangkat lunak sumber terbuka yang menjadi alternatif dari server web Netscape.

2. MySQL

MySQL atau sering disebut “*My Structured Query Language*”. MySQL ini berjalan sebagai server yang menyediakan multi-user untuk mengakses ke sejumlah database.

3. PHP

Bahasa pemrograman *script* sisi *server* yang didesain untuk pengembangan *web*.

2.8.4 Bagian-bagian XAMPP

Berikut ini adalah merupakan bagian-bagian dari XAMPP:

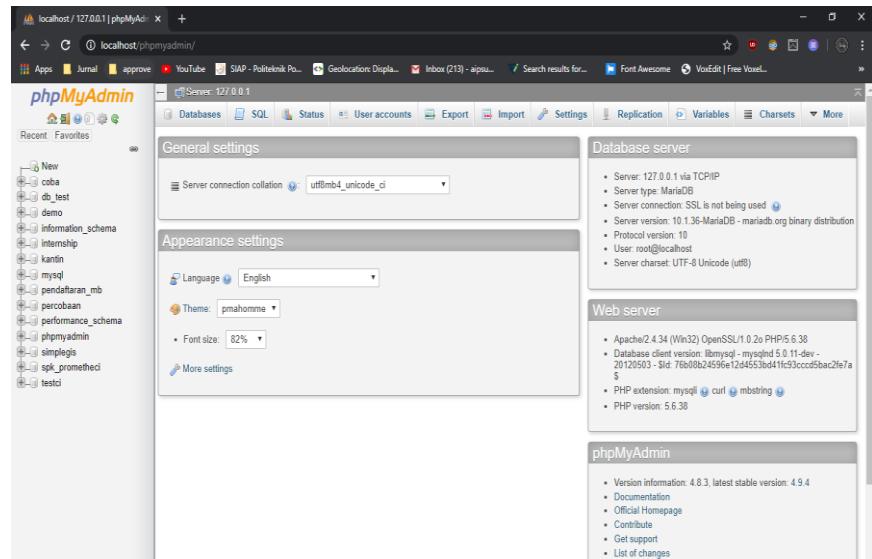
1. Htdocs

Name	Date modified	Type	Size
anonymous	19/10/2019 14.55	File folder	
apache	19/10/2019 14.56	File folder	
cgi-bin	19/10/2019 15.09	File folder	
contrib	19/10/2019 14.56	File folder	
FileZillaFTP	19/10/2019 15.09	File folder	
htdocs	07/01/2020 07.21	File folder	

Figure 9.9 Htdocs

Htdocs adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.

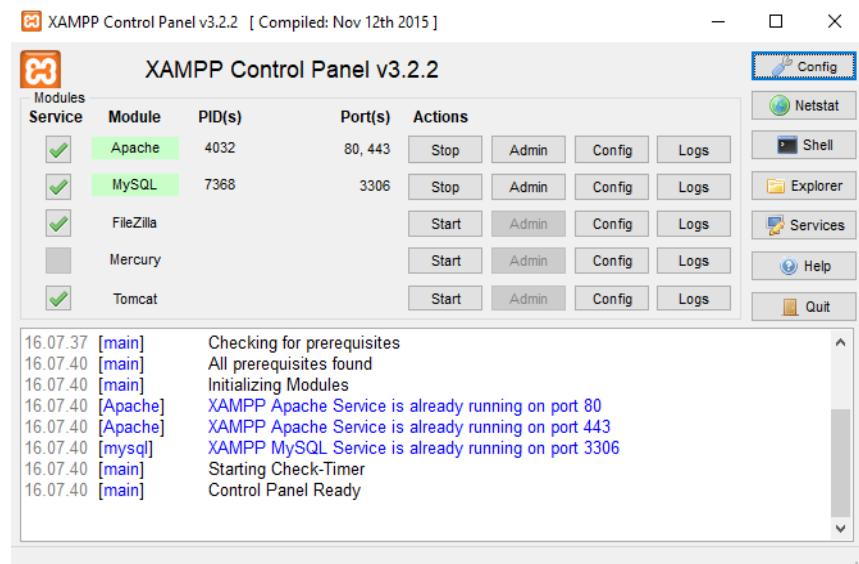
2. PhpMyadmin



Gambar 10.10 PhpMyadmin

PhpMyadmin merupakan bagian untuk mengelola basis data MySQL yang ada dikomputer. Untuk membukanya, buka browser internet lalu ketikkan alamat <http://localhost/phpMyadmin/>, maka akan muncul halaman phpMyAdmin.

3. Control Panel



Gambar 11.11 Control Panel

Control panel yang berfungsi untuk mengelola layanan (service) XAMPP. Seperti menghentikan (stop) layanan, ataupun memulai (start).

2.5.4 Kelebihan dan kekurangan XAMPP

Berikut ini terdapat beberapa kelebihan dan kekurangan XAMPP, terdiri atas:

a. Kelebihan :

1. *Database Storage Engine* ini banyak digunakan oleh programmer apalagi oleh *web developer* karena sifatnya yang *free*.
2. Keamanan datanya cukup aman walaupun tidak sehebat Postgre apalagi Oracle.
3. *Engine* ini *multiplatform* sehingga mampu diaplikasikan diberbagai sistem operasi.
4. Kelebihan paling utama *engine* ini adalah kecepatan.
5. Kemampuannya sudah bisa diandalkan, mempunyai kapasitas yang cukup mumpuni sekitar 60.000 tabel dengan jumlah *record* mencapai 5.000.000.000 bahkan untuk yang tterbaru sudah lebih.

b. Kekurangan :

1. Tidak cocok untuk menangani data dengan jumlah yang besar, baik untuk menyimpan data maupun untuk memproses data.
2. Memiliki keterbatasan kemampuan kinerja pada server ketika data yang disimpan telah melebihi batas maksimal kemampuan daya tampung server karena tidak menerapkan konsep *Technology Cluste*.

2.6 Web

Menurut Rahmat Hidayat [3] *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan

yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Kumpulan folder dan file yang dimaksud diatas, diupload ke server hosting online. Setelah semua settingan kumpulan script dan folder pada server hosting selesai dan benar, serta telah diletakkan pada domainnya, setelah alamat website tersebut bisa diakses melalui browser, tentunya membutuhkan koneksi internet untuk itu.

2.6.1 Beberapa pengertian web:

Kumpulan folder dan file yang mengandung banyak perintah dan fungsi fungsi tertentu, seperti fungsi tampilan, fungsi menangani penyimpanan data.

This PC > Local Disk (C:) > xampp > mysql				
	Name	Date modified	Type	Size
ss	backup	6/27/2019 12:02 AM	File folder	
ds	bin	6/27/2019 12:06 AM	File folder	
its	data	1/24/2020 10:52 AM	File folder	
	scripts	6/27/2019 12:00 AM	File folder	
	share	6/27/2019 12:02 AM	File folder	
ship	COPYING	2/5/2019 12:02 AM	File	18 KB
buku HKI	COPYING.thirdparty	2/5/2019 12:02 AM	THIRDPARTY File	85 KB
	CREDITS	2/5/2019 12:02 AM	File	3 KB
R TUJUH	EXCEPTIONS-CLIENT	2/5/2019 12:02 AM	File	9 KB
	my-huge	2/5/2019 2:13 AM	Configuration sett...	5 KB
	my-innodb-heavy-4G	2/5/2019 2:13 AM	Configuration sett...	21 KB
	my-large	2/5/2019 2:13 AM	Configuration sett...	5 KB
ts	my-medium	2/5/2019 2:13 AM	Configuration sett...	5 KB
	my-small	2/5/2019 2:13 AM	Configuration sett...	3 KB
its	mysql_installservice	3/30/2013 7:29 PM	Windows Batch File	1 KB
its	mysql_uninstallservice	3/30/2013 7:29 PM	Windows Batch File	1 KB
	README	2/5/2019 12:02 AM	File	3 KB
	resetroot	3/30/2013 7:29 PM	Windows Batch File	2 KB

Gambar 12.12 Kumpulan File dan Folder

2.6.2 Server Online

Kumpulan folder dan file yang dimaksud diatas, diupload ke server hosting online. Banyak sekali website penyedia jasa sewa hosting dan menjual domain.



Gambar 13.13 Server

2.6.3 Diakses Browser

Kumpulan script dan folder pada server hosting selesai dan benar, serta telah diletakkan pada domainnya, alamat website bisa diakses melalui browser, tentunya membutuhkan koneksi internet.



Gambar 14.14 Browser

2.6.4 Jenis Website

Website bisa digolongkan ke dalam beberapa jenis, seperti website pribadi, website komersial, website pemerintahan, dan website lembaga nirlaba. Website ditinjau dari segi dinamisnya dapat dibagi menjadi dua, yaitu:

1. website statis

Halaman web statis adalah halaman web di mana file halaman web yang disimpan di server sama seperti yang terlihat di browser. Konsekuensi jika isi file di server tidak diubah, maka user tidak dapat melihat perubahan pada halaman web yang dibukanya. Sebuah halaman web statis dibuat menggunakan bahasa HTML dan XHTML, CSS dan interaktivitas dengan JavaScript.

2. Dinamis.

Website dinamis adalah halaman web di mana tampilan yang dilihat oleh user tidak sama dengan file web di server. Aplikasi web dinamis juga disebut sebagai aplikasi web. Untuk menghasilkan tampilan yang dinamis, website menggunakan bahasa pemrograman web, seperti Java Server Pages (JSP), PHP, ASP, dan Perl programming languages, selain itu ada juga YUMA dan ColdFusion (CFM).

2.6.5 Fungsi Website

- a. Media Promosi

Sebagai media promosi dapat dibedakan menjadi media promosi utama, website yang berfungsi sebagai search engine atau toko Online, atau sebagai penunjang promosi utama, namun website dapat berisi informasi yang lebih lengkap daripada media promosi offline seperti koran atau majalah.

- b. Media Pemasaran

Website merupakan media pemasaran yang cukup baik, dibandingkan dengan toko sebagaimana di dunia nyata, untuk membangun toko online diperlukan modal yang relatif lebih kecil, dan dapat beroperasi 24 jam walaupun pemilik website tersebut sedang istirahat atau sedang tidak ditempat, serta dapat diakses darimana saja.

c. Media Informasi

Website portal dan radio atau tv online informasi bersifat global karena dapat diakses dari mana saja selama dapat terhubung ke internet, sehingga dapat menjangkau lebih luas daripada media informasi konvensional seperti koran, majalah, radio atau televisi yang bersifat lokal.

d. Media Pendidikan

Komunitas yang membangun website khusus berisi informasi atau artikel yang sarat dengan informasi ilmiah misalnya wikipedia.

e. Media Komunikasi

Sekarang banyak terdapat website yang dibangun khusus untuk berkomunikasi seperti forum yang dapat memberikan fasilitas fasilitas bagi para anggotanya untuk saling berbagi informasi atau membantu pemecahan masalah tertentu.

BAB III

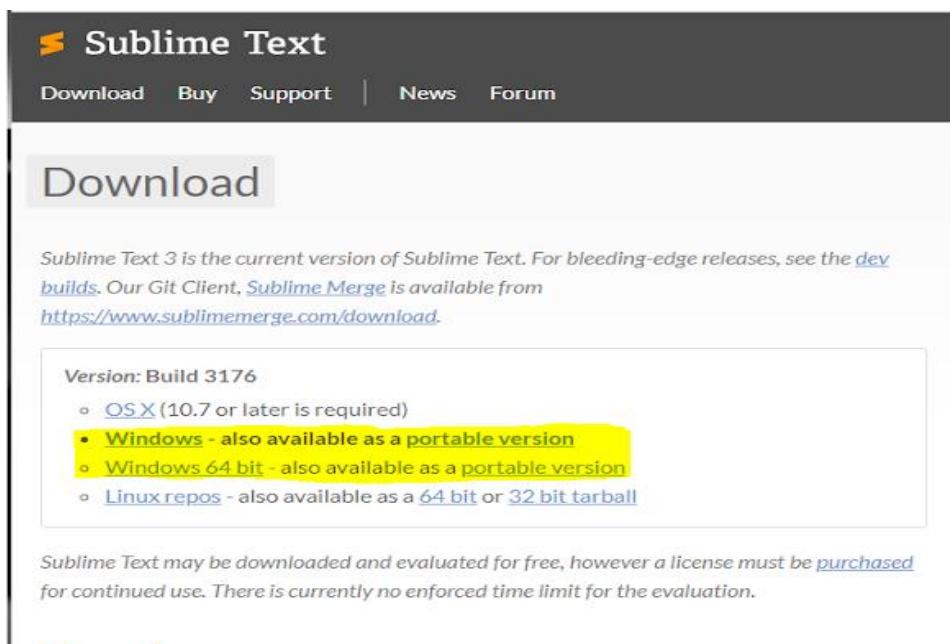
INSTALASI APLIKASI YANG AKAN DIGUNAKAN

3.1 Sublime Text

Sublime Text adalah editor yang canggih untuk belajar coding, markup dan prosa setelah Notepad++. Sublime Text mempunyai anatarmuka kinerja yang polos, dengan fitur-fitur yang canggih. Fitur tersebut terletak pada plugin-pluginya yang bisa kita instal dengan mudah.

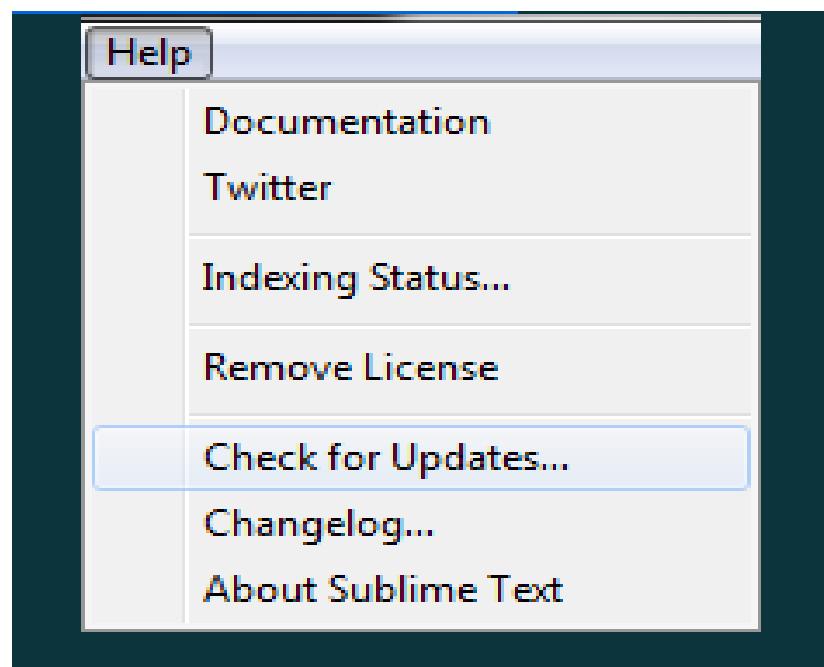
Berikut ini merupakan langkah-langkah instalasi visual studio code pada sistem operasi windows:

- a. Pertama, *download* terlebih dahulu file installer sublime text melalui situs resmi pada link berikut ini <https://www.sublimetext.com/> pada bagian atas pilih Download. Pilih sesuai versi Sistem Operasinya seperti contoh dibawah menggunakan Windows 32bit atau 64bit pilih salah satunya dengan portable version.



Gambar 3.1 Sublime Text

- b. Setelah selesai di Download, ekstrak semua file. Buka file Sublime Text 3 nya di dalam folder yang telah di ekstrak tadi. Pastikan Sublime Text 3 sudah versi terbaru cara mengecek nya dengan cara Help>>Check for Updates



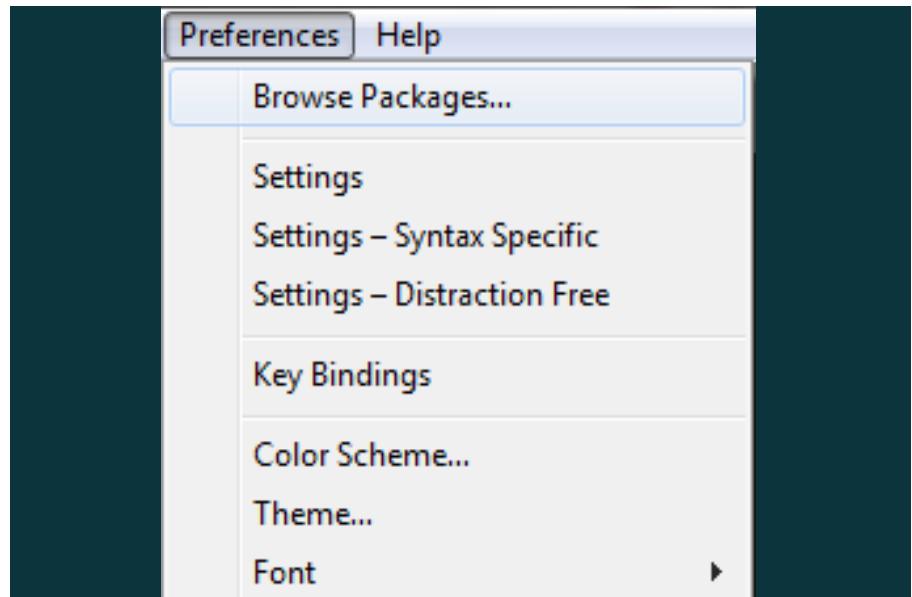
Gambar 15.2 File pada Sublim Text

- c. Langkah selanjutnya, buat supaya Sublime Text teregistrasi dengan cara isi pada Help>>Enter License. Isi dengan kode berikut, disini saya menggunakan Sublime Text 3 build 3176.

```
----- BEGIN LICENSE -----  
sgbteam  
Single User License  
EA7E-1153259  
8891CBB9 F1513E4F 1A3405C1 A865D53F  
115F202E 7B91AB2D 0D2A40ED 352B269B  
76E84F0B CD69BFC7 59F2DFEF E267328F  
215652A3 E88F9D8F 4C38E3BA 5B2DAAE4  
969624E7 DC9CD4D5 717FB40C 1B9738CF  
20B3C4F1 E917B5B3 87C38D9C ACCE7DD8  
5F7EF854 86B9743C FADC04AA FB0DA5C0  
F913BE58 42FEA319 F954EFDD AE881E0B  
----- END LICENSE -----
```

Gambar 16.3 Kode Sublime Text

- d. Setelah teregistrasi, barulah kita memulai untuk menambah plugin-plugin yang canggih yang berguna di Sublime Text 3 ini.
- e. Plugin yang wajib dan harus diinstal terlebih dahulu yaitu **Package Control** karena plugin inilah yang berfungsi untuk menambah/menginstal plugin, mengunistal plugin, disable, enable plugin dan fungsi lainnya. Caranya yaitu :
 - a. Download terlebih dahulu **Package Control** nya disini.
 - b. Copy file yang telah didownload tersebut.
 - c. Buka Sublime Text 3 dan pilih menu **Preference>>Browse Packages...**



Gambar 17.4 Preference

- f. Maka akan muncul menu Explorer, cari folder Installed Packages pastekan file yang telah dicopy kedalam folder tersebut

A screenshot of a Windows File Explorer window titled "Sublime Text 3". The address bar shows the path: "This PC > Local Disk (C:) > Program Files > Sublime Text 3". The left sidebar shows "Quick access" and a list of folders like Desktop, Downloads, Documents, Pictures, and SEMESTER TUJUH. The main pane displays a list of files and folders under "Sublime Text 3". One folder, "Installed Packages", is highlighted with a blue circle. Inside the "Installed Packages" folder, two files are visible: "sublime_plugin.py" and "sublime_text.exe".

Name	Date modified	Type	Size
Packages	8/13/2019 11:03 AM	File folder	
changelog	4/6/2019 11:46 AM	Text Document	55 KB
crash_reporter	4/6/2019 11:59 AM	Application	262 KB
msvcr100.dll	10/11/2018 7:11 PM	Application exten...	810 KB
plugin_host	4/6/2019 11:59 AM	Application	7,831 KB
python3.3	10/11/2018 7:11 PM	WinRAR ZIP archive	2,567 KB
python33.dll	10/11/2018 7:11 PM	Application exten...	6,923 KB
subl	4/6/2019 11:59 AM	Application	485 KB
sublime.py	2/14/2019 5:09 PM	PY File	38 KB
sublime_plugin.py	2/1/2019 2:43 PM	PY File	37 KB
sublime_text.exe	4/6/2019 11:58 AM	Application	8,153 KB
unins000.dat	8/13/2019 11:03 AM	DAT File	12 KB
unins000	8/13/2019 11:03 AM	Application	1,160 KB
unins000.msg	8/13/2019 11:03 AM	MSG File	23 KB
update_installer	4/6/2019 11:59 AM	Application	188 KB

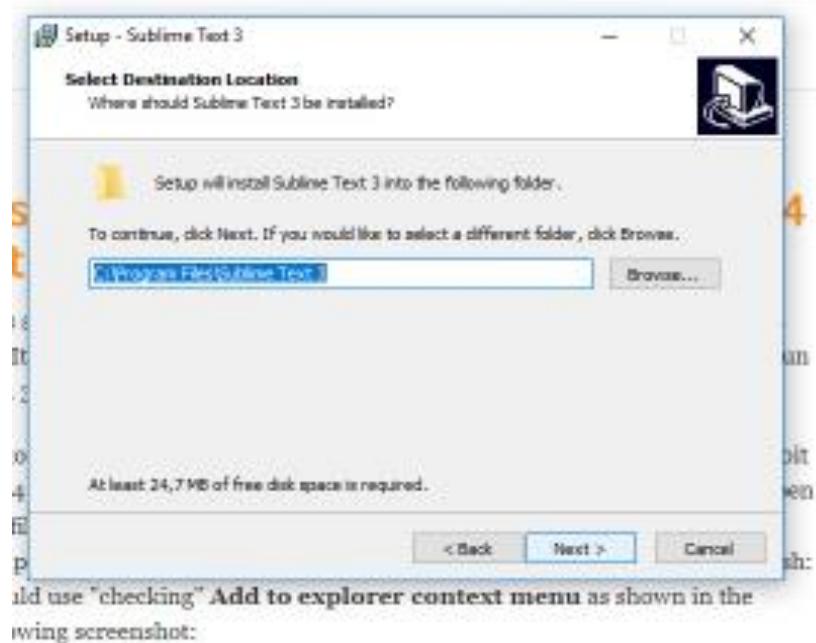
Gambar 18.5 Packages

- g. Klik dua kali pada file exe pada hasil download.



Figure 19.6 File Dowload

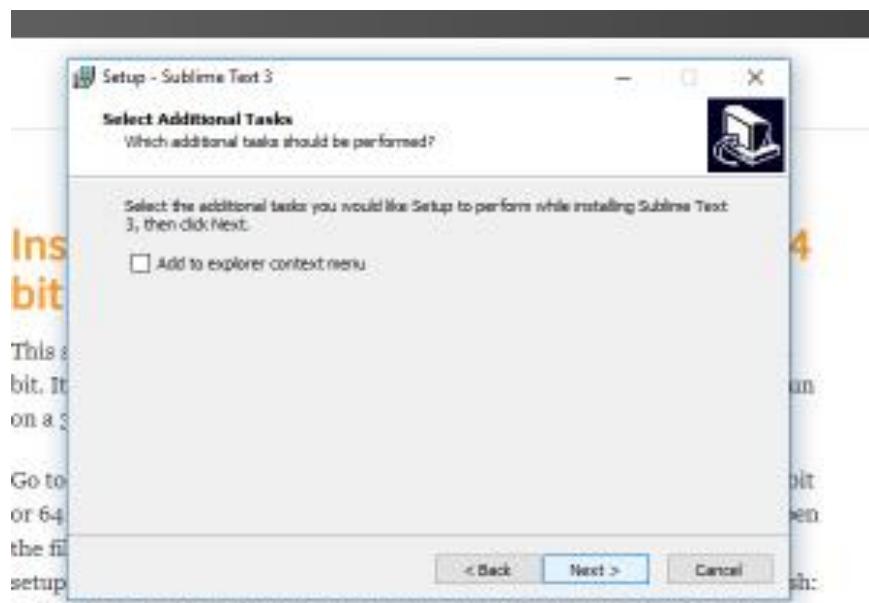
h. Klik next



old use "checking" Add to explorer context menu as shown in the wing screenshot:

Gambar 20.7 Setup Sublime

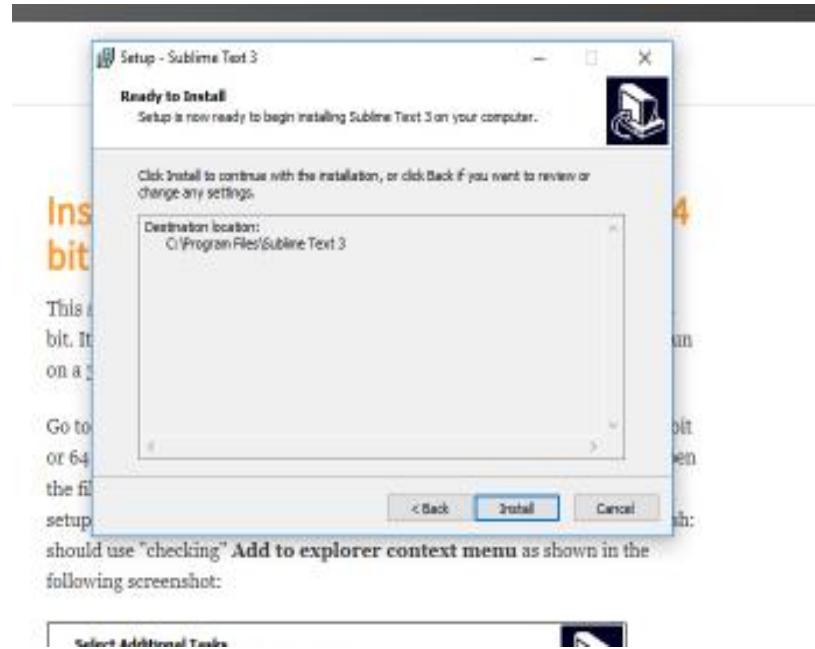
- i. Klik Add to explorel kemudian klik next lagi



should use "checking" Add to explorer context menu as shown in the following screenshot:

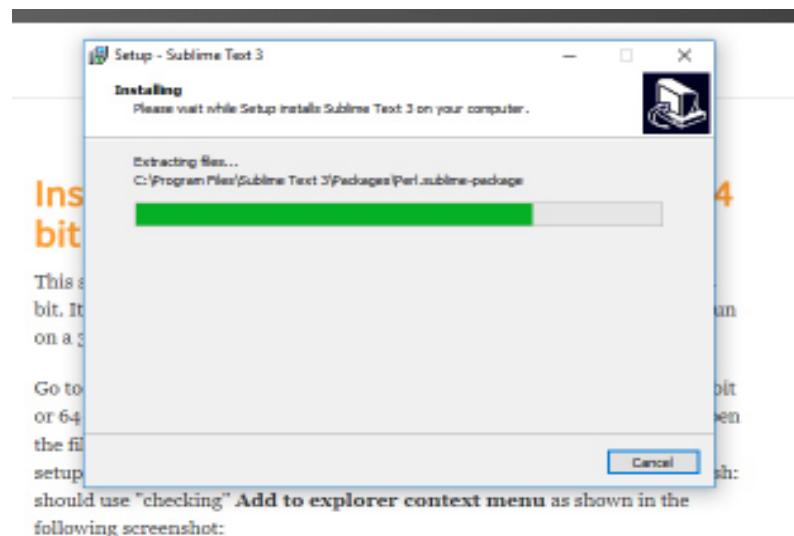
Gambar 21.8 Ekxplorel

- j. Klik install



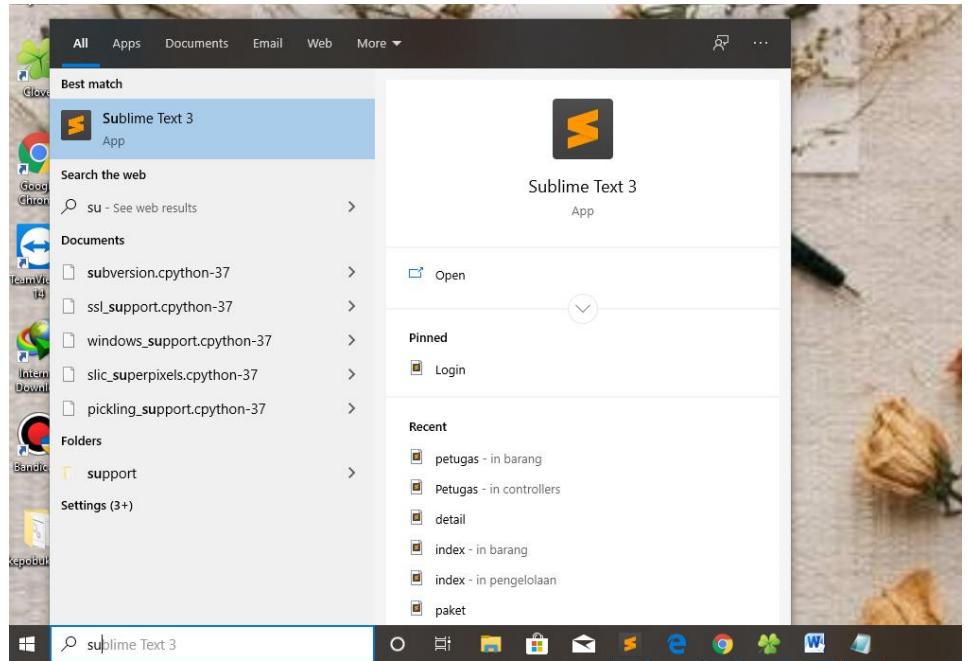
Gambar 22.9 Istall

- k. Kemudian tunggu sampe proses install selesai



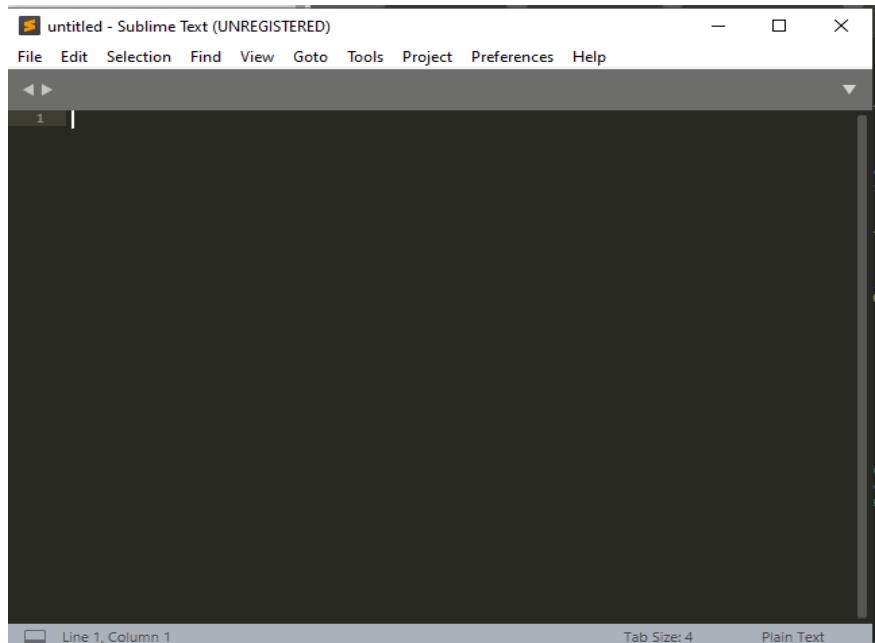
Gambar 23..10 Proses Install

- l. Setalah proses instal selesai kemudian bukan pada di menu windows 10 seperti gambar dibawah ini



Gambar 24.11 Menu Windows

m. Sublime siap untuk membuat aplikasi



Gambar 25.12 Halaman Utama Sublime Text

3.2 Xampp

Xampp adalah sebuah paket perangkat lunak (*software*) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL / MariaDB, PHP, dan PERL. Sementara huruf “X” yang terdapat pada awal kata berasal dari istilah *cross platform* sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi yang berbeda.

Jika dijabarkan secara gamblang, masing-masing huruf yang ada di dalam nama XAMPP memiliki arti sebagai berikut.

X = **Cross Platform**, merupakan kode penanda untuk *software cross platform* atau yang bisa berjalan di banyak sistem operasi.

A = **Apache**, apache adalah aplikasi *web server* yang bersifat gratis dan bisa dikembangkan oleh banyak orang (*open source*).

M = **MySQL / MariaDB**, MySQL atau MariaDB merupakan aplikasi *database server* yang dikembangkan oleh orang yang sama. MySQL berperan dalam mengolah, mengedit, dan menghapus daftar melalui *database*.

P = **PHP**, huruf “P” yang pertama dari akronim kata XAMPP adalah inisial untuk menunjukkan eksistensi bahasa pemrograman PHP. Bahasa pemrograman ini biasanya digunakan untuk membuat *website dinamis*.

P = **Perl**, untuk huruf “P” selanjutnya merupakan singkatan dari bahasa pemrograman Perl yang kerap digunakan untuk memenuhi berbagai macam kebutuhan. Perl ini bisa berjalan di dalam banyak sistem operasi, sehingga sangat fleksibel dan banyak digunakan.

Berikut merupakan langkah-langkah untuk melakukan instalasi xampp pada sistem operasi windows.

- a. Pertama, *download* terlebih dahulu file installer xampp pada link berikut ini. <https://www.apachefriends.org/>



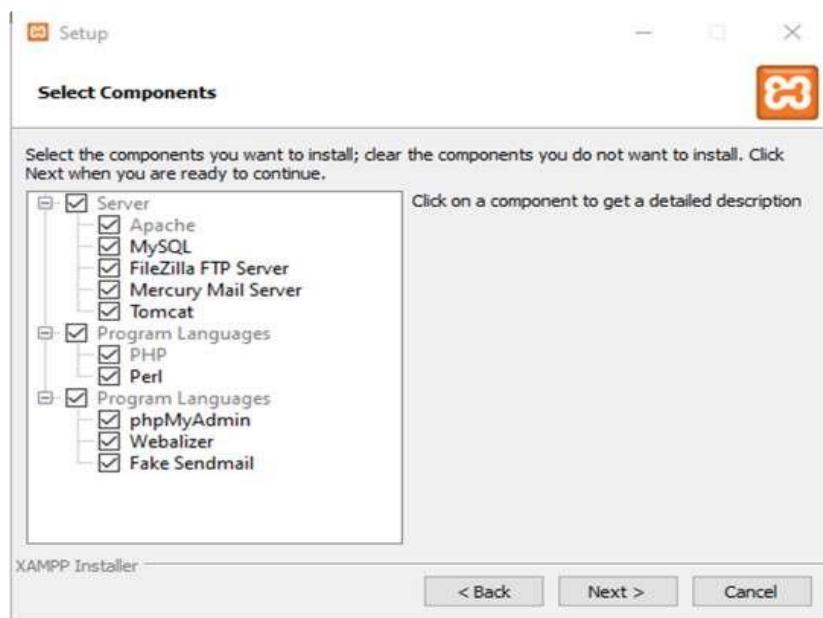
Gambar 26.13 File Instalerr Xampp

- b. Pilih file installer sesuai dengan spesifikasi laptop yang digunakan.
- c. Klik file xampp yang sudah berhasil di *download*, maka akan muncul tampilan seperti ini.



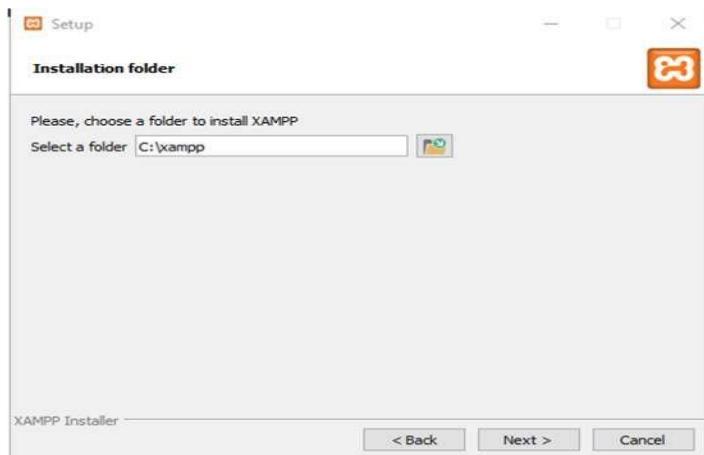
Gambar 27.14 Setup Xampp

- d. Kemudian, pilih next untuk melanjutkan pada proses instalasi.



Gambar 28.15 Select Componen

- e. Pada tampilan *select component*, centang semua pilihannya agar dapat menjalankan semua program yang dipilih. Kemudian pilih *next*.



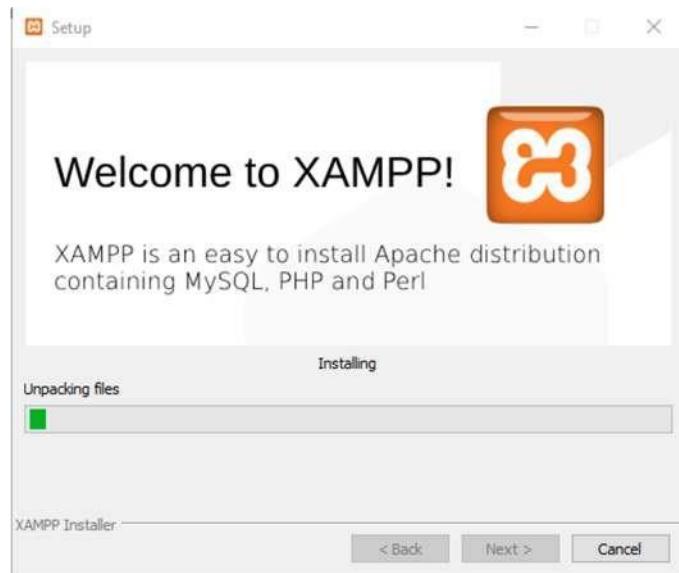
Gambar 29.16 Installator Folder

- f. Pada, *installation folder* secara *default* sistem akan menyimpannya pada direktori C. Jika sudah klik *next* untuk melanjutkan proses instalasi.



Gambar 30.17 Bitami for Xampp

- g. Pada tampilan ini langsung saja klik *next* untuk melanjutkan.



Gambar 31.18 Proses Install

- h. Kemudian tunggu hingga proses instalasi selesai.



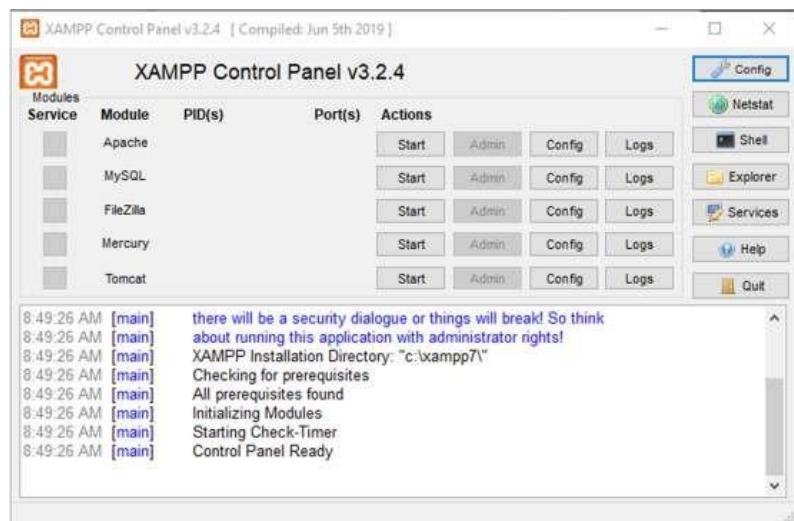
Gambar 32.19 Bitnami Finish

- i. Jika telah selesai, selanjutnya klik *Finish* untuk membuka *control panel*.
- j. Pilih bahasa yang akan digunakan pada xampp. Lalu klik *save*



Gambar 33.19 Halaman Bahasa yang digunakan

- k. Jika berhasil, maka tampilan utama *control panel* dari xampp seperti berikut.



Gambar 34.20 Control Panel Xampp

3.3 Codeigniter

Codeigniter (CI) merupakan sebuah *web application framework* yang bersifat *open source* dimana digunakan untuk membangun aplikasi php dinamis. *Framework* itu sendiri merupakan abstraksi di dalam sebuah perangkat lunak yang menyediakan fungsi yang *generic* sehingga dapat dirubah oleh kode yang dibuat *user*, sehingga dapat menyediakan perangkat lunak untuk aplikasi tertentu.

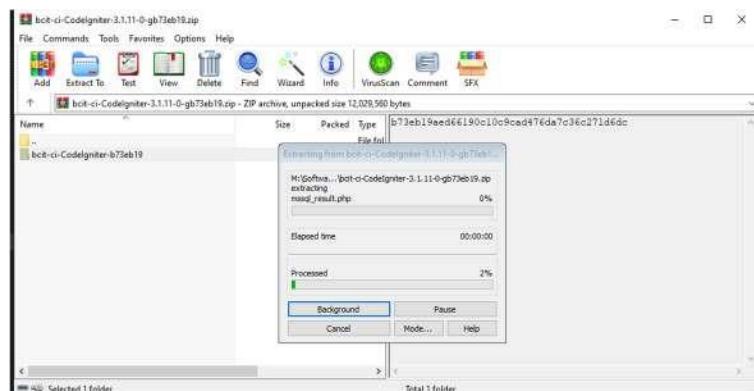
Berikut ini merupakan tahapan-tahapan untuk instalasi codeigniter pada sistem operasi windows.

- a. Pertama, *download* terlebih dahulu codeigniter pada *website* resminya pada link <https://codeigniter.com/download>.



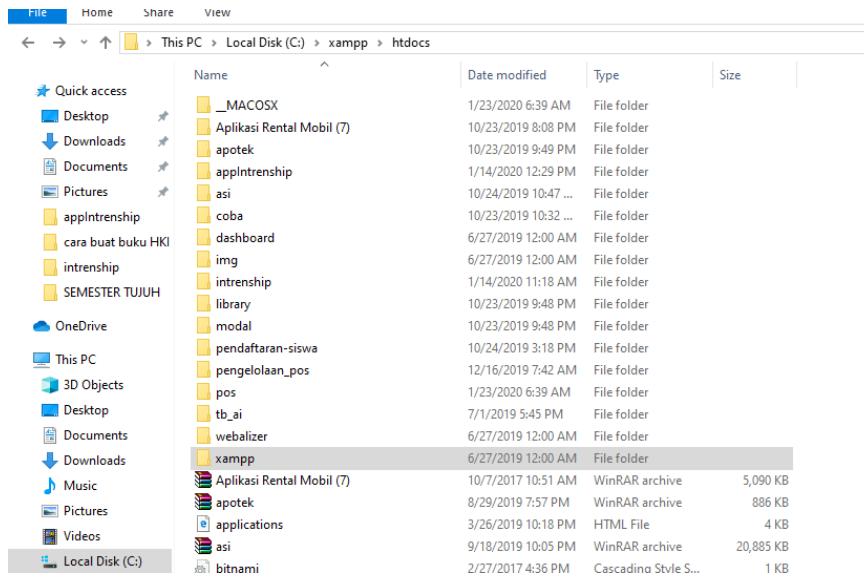
Gambar 35.21 Website

- b. Setelah berhasil di *download*, selanjutnya ekstrak file tersebut.



Gambar 36.22 Ekstrak File

- c. Kemudian, *copy* file yang telah di ekstrak ke dalam direktori htdocs, seperti gambar berikut.



Gambar 37.23 Kopy File

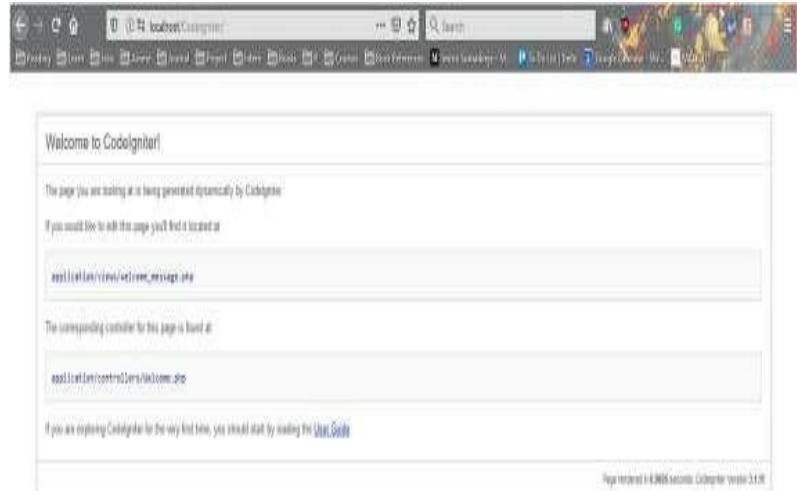
Untuk menjalankannya, kita menggunakan xampp, untuk itu silakan jalankan terlebih dahulu aplikasi xampp-nya seperti berikut.

- Buka aplikasi xamppnya, kemudian klik tombol start pada kolom actions baris pertama untuk menjalankan apache, dan klik tombol start pada kolom actions baris kedua untuk menjalankan MySQL.



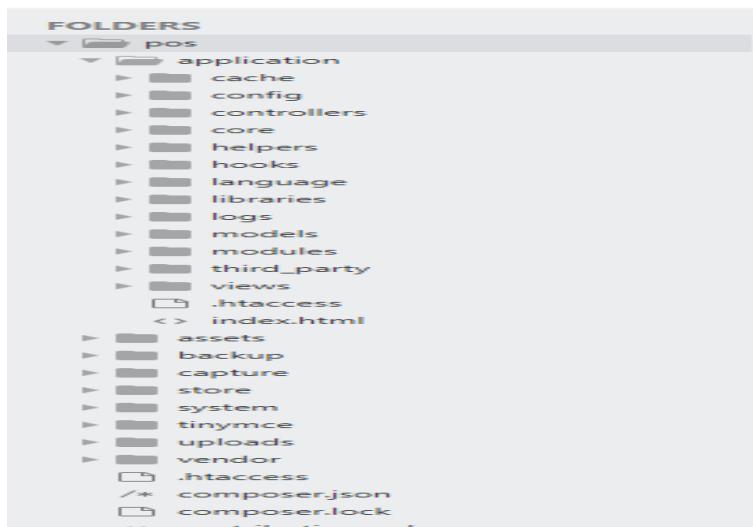
Gambar 38.24 Control Panel Xampp

- e. Setelah xamppnya berhasil dijalankan, proses selanjutnya tinggal memanggil folder yang telah disimpan pada direktori htdocs di browser. Maka tampilan utamanya seperti gambar berikut.



Gambar 39.25 Halaman Utama

- f. Berikut ini merupakan struktur folder yang terdapat pada codeigniter, diantaranya sebagai berikut.

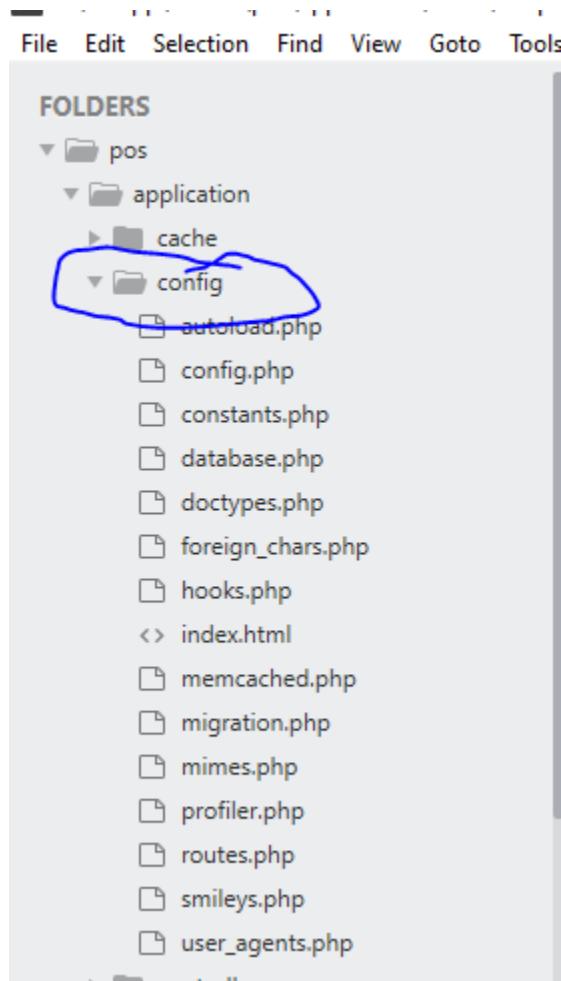


Gambar 40.26 Folder pada CodeIgneter

- ✓ *Application*, merupakan folder yang pada dasarnya menyimpan aplikasi yang sedang kita buat.
- ✓ *Cache*, merupakan folder yang menyimpan semua cache yang dibuat oleh cachelibrary
- ✓ *Config*, merupakan folder yang menyimpan informasi mengenai konfigurasi aplikasi seperti autoload, database, routes, dan lainnya.
- ✓ *Controller*, merupakan folder yang menyimpan controller-controller aplikasi yang dapat digunakan untuk menyusun aktivitas program.
- ✓ *Core*, merupakan folder untuk memperluas *class* inti codeigniter.
- ✓ *Helpers*, merupakan folder untuk menyimpan helpers.
- ✓ *Hooks*, merupakan folder untuk menyimpan hooks untuk mengubah alur fungsi dari core codeigniter.
- ✓ *Libraries*, merupakan folder untuk menyimpan library.
- ✓ *Logs*, merupakan folder untuk menyimpan semua error log apabila error log diaktifkan.
- ✓ *Models*, merupakan folder untuk menyimpan models yang akan didefinisikan tabel dari database yang dapat kita gunakan oleh controller yang kita buat untuk mengakses database.
- ✓ *Third_party*, merupakan folder untuk menyimpan fungsi-fungsi tambahan dalam cara kerja codeigniter.
- ✓ *Views*, merupakan folder untuk menyimpan tampilan dari aplikasi yang kita buat.
- ✓ *System*, merupakan folder untuk menyimpan sistem inti dari codeigniter.
- ✓ *Language*, merupakan folder untuk menyimpan bahasa-bahasa yang akan digunakan.

3.4 Konfigurasi dasar pada codeigniter

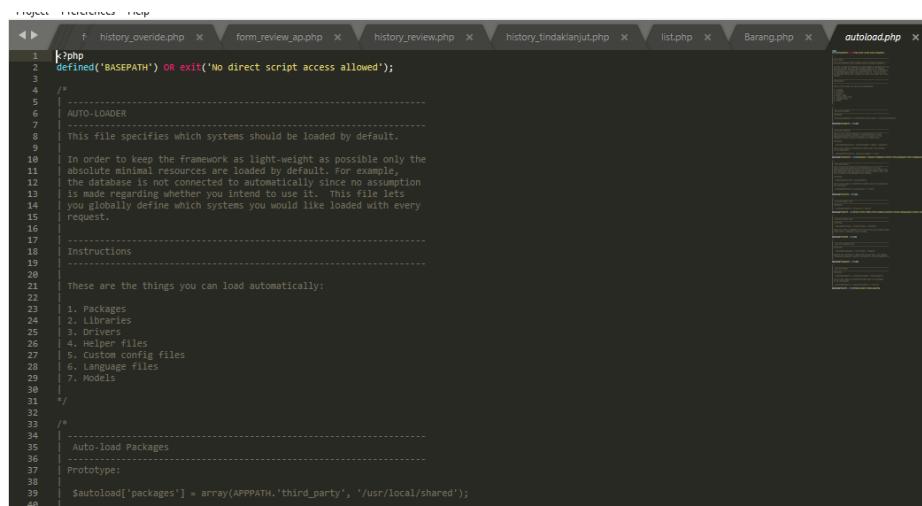
Dalam memulai codeigniter, ada beberapa konfigurasi dasar yang perlu kita ketahui, diantaranya autoload.php, config.php dan database.php. Semua konfigurasi pada codeigniter, terletak pada satu tempat yakni di dalam folder *application/config*.



Gambar 41.27 Folder Config

- a. Autoload.php, file ini digunakan untuk mengatur fungsi-fungsi yang akan dimuat otomatis di awal ketika program dijalankan. Untuk melakukan

- b. konfigurasi pada file autoload.php, silakan buka file- nya seperti gambar berikut.



```

1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3  /*
4  |-----+
5  | AUTO-LOADER
6  |-----+
7  | This file specifies which systems should be loaded by default.
8  |
9  | In order to keep the framework as light-weight as possible only the
10 | absolute minimal resources are loaded by default. For example,
11 | the database is not connected to automatically since no assumption
12 | is made regarding whether you intend to use it. This file lets
13 | you globally define which systems you would like loaded with every
14 | request.
15 |
16 |-----+
17 | Instructions
18 |-----+
19 |
20 |
21 | These are the things you can load automatically:
22 |
23 | 1. Packages
24 | 2. Libraries
25 | 3. Drivers
26 | 4. Helper files
27 | 5. Custom config files
28 | 6. Language files
29 | 7. Models
30 |
31 | */
32 |
33 |/*
34 | | Auto-load Packages
35 | | Prototype:
36 | |
37 | | $autoload['packages'] = array(APPPATH.'third_party', '/usr/local/shared');
38 |
39 |
40 |*/
41 /**
42 | $autoload['packages'] = array();
43 |
44 |/*

```

Gambar 42.28 Scirpt Autoload

- c. Kemudian temukan kode berikut

```

41 /**
42 | $autoload['packages'] = array();
43 |
44 |/*

```

Gambar 43.29 Script kode

- d. Ubah kode tersebut menjadi seperti kode berikut

```

50 /**
51 | $autoload['libraries'] = array('database', 'session', 'template', 'xmlrpc', 'form_validation', 'auth', 'mypagination', 'email');
52 |

```

Gambar 44.30 Kode Database

Kode tersebut dapat diartikan kita dapat meload library ‘database’ secara otomatis.

- e. Selanjutnya, temukan kode berikut, dan tambahkan ‘url’ didalamnya.

```

91 /**
92 | $autoload['helper'] = array('url', 'form', 'html', 'file', 'cookie', 'utility', 'string', 'menunavigasi', 'asset', 'options', 'notification');
93 |

```

Gambar 45.31 Kode Url

Kode tersebut dapat diartikan kita dapat meload helper “url” secara otomatis. Config.php, pada file ini terdapat beberapa konfigurasi yang secara standar sudah terkonfigurasi, namun terdapat beberapa konfigurasi yang perlu diperhatikan, untuk konfigurasi dasar, cukup mengetahui konfigurasi base_url.



```
1  k?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  /*
5  -----
6  Base Site URL
7  -----
8
9  URL to your CodeIgniter root. Typically this will be your base URL,
10 WITH a trailing slash:
11
12     http://example.com/
13
14 WARNING: You MUST set this value!
15
16 if it is not set, then CodeIgniter will try guess the protocol and path
17 your installation, but due to security concerns the hostname will be set
18 to $_SERVER['SERVER_NAME'] if available, or localhost otherwise.
19 The auto-detection mechanism exists only for convenience during
20 development and MUST NOT be used in production!
21
22 if you need to allow multiple domains, remember that this file is still
23 a PHP script and you can easily do that on your own.
24
25 */
26 error_reporting(0);
```

Gambar 46.31 Script Konfigurasi

- f. Kemudian kode berikut

```
/*
$config['url_suffix'] = '';
/*
/*
```

Gambar 47.32 Script Config

- g. Kemudian tambahkan kode tersebut menjadi nama folder yang disimpan di hatedocks

```
165  /*
166  $config['permitted_uri_chars'] = 'a-z 0-9~%.:_\-\';
167  */
```

Gambar 48.33 Nama Kode File

- h. Database.php, file ini digunakan untuk melakukan konfigurasi yang berkaitan dengan konfigurasi database dari website yang akan dibuat. Adapun konfigurasi yang perlu diperhatikan, diantaranya: hostname, username, password dan database. Buka file database.php pada teks editor seperti gambar berikut.

```

1 // ...
2 if(!defined('BASEPATH')) OR exit('No direct script access allowed');
3 
4 /**
5  * DATABASE CONNECTIVITY SETTINGS
6  * -----
7  * This file will contain the settings needed to access your database.
8  *
9  * For complete instructions please consult the 'Database Connection'
10 * page of the User Guide.
11 *
12 * -----
13 * EXPLANATION OF VARIABLES
14 *
15 *
16 * ['dsn'] The full DSN string describe a connection to the database.
17 * ['hostname'] The hostname of your database server.
18 * ['username'] The username used to connect to the database
19 * ['password'] The password used to connect to the database
20 * ['database'] The name of the database you want to connect to
21 * ['dbdriver'] The database driver. e.g.: mysqli.
22 *             Currently supported:
23 *             cubrid, mysqli, mysql, oracle, odbc,
24 *             odbc, pgsql, sqlite, sqlite3, sqlsrv
25 *             ['dbprefix'] You can add an optional prefix, which will be added
26 *             to the table name when using the Query Builder class
27 *             ['pconnect'] TRUE/FALSE - Whether to use a persistent connection
28 *             ['db_debug'] TRUE/FALSE - Whether database errors should be displayed.
29 *             ['cache_on'] TRUE/FALSE - Enables/disables query caching
30 *             ['cachedir'] The path to the folder where cache files should be stored
31 *             ['char_set'] The character set used in communicating with the database
32 *             ['dbcollat'] The character collation used when communicating with the database
33 *             NOTE: For MySQL and MySQLi databases, this setting is only used
34 *             as a backup if your server is running PHP < 5.2.3 or MySQL < 5.0.7
35 *             (and in table creation queries made with DB Forge).
36 *             There is an incompatibility in PHP with mysql_real_escape_string() which
37 *             can make your site vulnerable to SQL injection if you are using a
38 *             multi-byte character set and are running versions lower than these.
39 *             Sites using Latin-1 or UTF-8 database character set and collation are unaffected.
40 *             ['swap_pre'] A default table prefix that should be swapped with the dbprefix
41 *             ['encrypt'] Whether or not to use an encrypted connection.
42 *             ['compress'] (deprecated) 'useCompression' and 'ndo/useCompress' drivers accept TRUE/FALSE
43 *
44 *             'mysql' (deprecated) 'useMyISAM' and 'ndo/useMyISAM' drivers accept TRUE/FALSE
45 */

```

Gambar 49.4 Text Editor

- i. Kemudian cari kode seperti file berikut pada file database.php

```

72 /**
73  * $active_group = 'default';
74  * $query_builder = TRUE;
75  *
76  * $db['default'] = array(
77  *     'dsn' => '',
78  *     'hostname' => 'localhost',
79  *     'username' => 'root',
80  *     'password' => '',
81  *     'database' => 'db_pos',
82  *     'dbdriver' => 'mysqli',
83  *     'dbprefix' => 'profile_',
84  *     'pconnect' => FALSE,
85  *     'db_debug' => (ENVIRONMENT !== 'production'),
86  *     'cache_on' => FALSE,
87  *     'cachedir' => '',
88  *     'char_set' => 'utf8',
89  *     'dbcollat' => 'utf8_general_ci',
90  *     'swap_pre' => '',
91  *     'encrypt' => FALSE,
92  *     'compress' => FALSE,
93  *     'stricton' => FALSE,
94  *     'failover' => array(),
95  *     'save_queries' => TRUE
96  * );

```

Gambar 50.35 File Database

- j. Kemudian tambahkan kode seperti gambar berikut

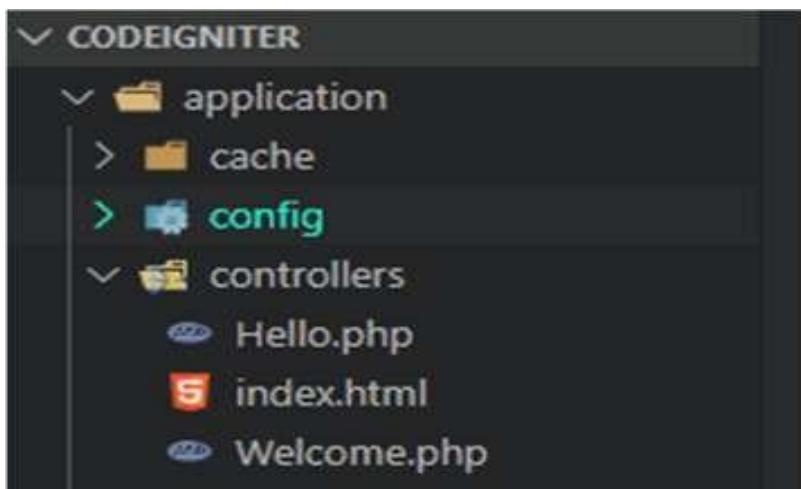
```
72  */
73  $active_group = 'default';
74  $query_builder = TRUE;
75
76  $db['default'] = array(
77      'dsn' => '',
78      'hostname' => 'localhost',
79      'username' => 'root',
80      'password' => '',
81      'database' => 'db_pos',
82      'dbdriver' => 'mysqli',
83      'dbprefix' => 'profile_',
84      'pconnect' => FALSE,
85      'db_debug' => (ENVIRONMENT !== 'production'),
86      'cache_on' => FALSE,
87      'cachedir' => '',
88      'char_set' => 'utf8',
89      'dbcollat' => 'utf8_general_ci',
90      'swap_pre' => '',
91      'encrypt' => FALSE,
92      'compress' => FALSE,
93      'stricton' => FALSE,
94      'failover' => array(),
95      'save_queries' => TRUE
96  );
97
```

Gambar 51.36 File Database

3.5 Hello World Codeigniter

Untuk menguji pemahaman pada codeigniter, kita akan mencoba dengan menampilkan text “Hello World” pada browser menggunakan controller.

Buat sebuah controller dengan nama Hello.php seperti gambar berikut



Gambar 52.37 Config Hello Word

Setelah itu, ketikan kode seperti gambar berikut.



```
application > controllers > Hello.php
1 <?php
2 class Hello extends CI_Controller{
3     function index(){
4         echo "Hello World!";
5     }
6 }
7 ?>
```

Gambar 53.38 Script Hello Word

Jika berhasil maka hasilnya akan seperti gambar berikut.



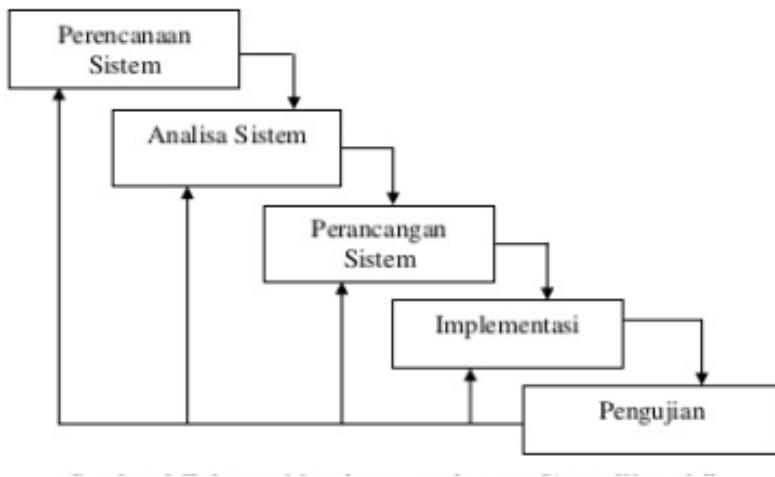
Gambar 54.39 Halaman Web Hello Word

BAB IV

METODELOGI PENELITIAN

4.1 Diagram Alur Metodelogi Penelitian

Metode penelitian yang di gunakan dalam penelitian ini di bagi kedalam beberapa tahap agar peroses yang dilakukan lebih terarah. Secara umum langkah-langkah penelitian yang di lakukan untuk system informasi pengelolaan paket di kampus Politeknik Pos Indonesia adalah sebagai berikut :



Gambar 55.1 Diagram Alur Metodelogi Penelitian

4.2 Tahapan-Tahapan Diagram Alur Metedelogi Penelitian

Tahapan -tahapan diagram alur metedelogi penelitian yang digunakan peneliti adalah sebagai berikut :

4.2.1 Perancangan Sistem

Pada langkah ini penulis akan melakukan pendefinisian seluruh pendefinisian seluruh kebutuhan perangkat lunak yang nantinya akan dijadikan sebagai SRS (*Software Requirements Specification*). RSR adalah dokumen lengkap mengenai apa kemampuan perangkat lunak (*what*) tanpa menjelaskan bagaimana (*how*) perangkat lunak akan melaksanakan kemampuan perangkat tersebut.

4.2.2 4.1 Diagram Alur Metodelogi Penelitian

Dari hasil peninjauan masalah penulis menemukan bahwa sistem kegiatan system informasi pengelolaan pos satpam di Politeknik Pos Indonesia ditemukan permasalahan yang terjadi yaitu :

1. Bagaimana memodelkan aplikasi system informasi pengelolaan pos satpam di kampus Politeknik Pos Indonesia.
2. Bagaimana membuat aplikasi system informasi pengelolaan pos satpam di kampus Politeknik Pos Indonesia berdasarkan proses bisnis yang ada.
3. Bagaimana menggunakan metode harvesine untuk menghitung jarak kedua titik koordinat.

4.2.3 Analisis Sistem

Kebutuhan data harus dianalisis diawal secara lengkap dan menyeluruh Kebutuhan data dapat ditambah ataupun dikurangi sesuai dengan kebutuhan user, ketika dilakukan testing. Perubahan data ataupun fungsional akan merubah keseluruhan proses pada tahapan berikutnya. Perubahan dapat dilakukan selama sistem atau perangkat lunak masih dalam bentuk prototype System informasi pengelolaan pos satpam politekik pos Indonesia.

4.2.4 Perancangan Sistem

Testing dilakukan ketika semua tahapan pada model sudah selesai. Tidak dapat memberikan gambaran yang jelas mengenai sistem yang dibangun, karena sistem bisa dilihat jika semua tahapan telah dilakukan. Testing dapat dilakukan ketika prototype telah dibangun, sehingga hasil testing dapat merubah rancangan sistem. Memberikan prototype sebagai gambaran sistem yang akan dibangun, sehingga user dapat melihat dan berinteraksi langsung dengan gambaran sistem.

4.2.5 Implementasi System

Menerapkan proses perancangan yang baik evaluasi dilakukan ketika sistem telah dibangun mengedepankan kebutuhan fungsional sistem. Tidak menerapkan proses perancangan yang baik Evaluasi dilakukan ketika prototype telah dibangun mengedepankan aspek kenyamanan user.

4.2.6 Pengujian System

Dari data yang telah dikumpulkan selanjutnya dilakukan analisis sistem yang akan dibangun berdasarkan sistem yang sedang berjalan saat ini. Kemudian selanjutnya mulai dilakukan perancangan sistem berdasarkan hasil analisis kebutuhan sistem.

4.2.7 Metodelogi Waterfall

Model waterfall adalah model pengembangan perangkat lunak yang paling sering digunakan. Model pengembangan ini bersifat linear dari tahap awal pengembangan system yaitu tahap perencanaan sampai tahap akhir pengembangan system yaitu tahap pemeliharaan. Tahapan berikutnya tidak akan dilaksanakan sebelum tahapan sebelumnya selesai dilaksanakan dan tidak bisa kembali atau mengulang ke tahap sebelumnya.

BAB V

ANALISIS DAN PERANCANGAN

5.1 Analisis Dan Perancangan Sistem

Dalam rancang bangun sistem informasi *pengelolaan paket di kampus politeknik pos indonesia menggunakan email* diperlukan sebuah analisis. Analisis sangat diperlukan untuk merancang sebuah aplikasi yang akan dibangun. Dengan adanya analisis ini dapat membantu dalam menyelesaikan permasalahan yang terjadi agar dapat dilakukan perbaikan karena dalam merancang dan membangun sistem informasi *pengelolaan paket di kampus politeknik pos indonesia menggunakan email* tidak akan berjalan dengan baik tanpa adanya analisis.

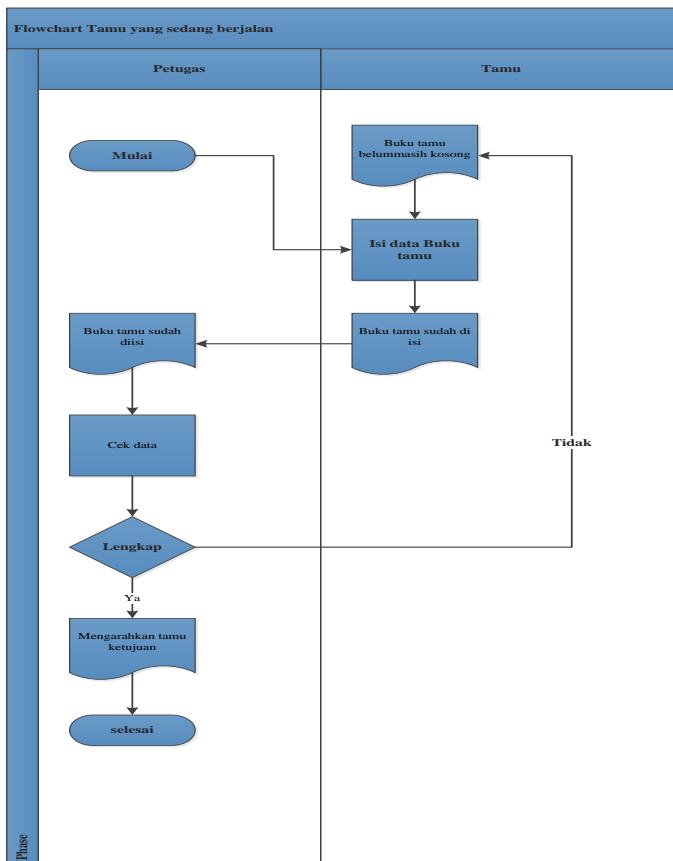
Perencanaan sistem menyangkut estimasi dari kebutuhan-kebutuhan fisik, tenaga kerja dan dana yang dibutuhkan untuk mendukung pengembangan sistem ini serta untuk mendukung operasinya setalah diterapkan sedangkan analisis merupakan proses untuk menentukan bentuk dari kebutuhan sistem yang menunjang kebutuhan pada saat membangun dan implementasi. Secara garis besar disebut juga sebagai proses mempelajari aktifitas system untuk memahami gambaran menyeluruh tentang sehingga perancang telah mengetahui apa saja kebutuhan dari sistem tersebut.

Oleh karena itu, dalam bab ini akan diuraikan tentang analisis yang dilakukan untuk membantu pembuatan program.

5.1.1 Analisis Sistem yang Sedang Berjalan

Analisis ini merupakan penggambaran proses yang sedang berjalan didalam pengelolaan. Proses yang ada hanya pendataan barang, penerima tamu dan pemberitahuan secara manual.

5.1.1.1 Analisis Flowchart Tamu yang sedang berjalan

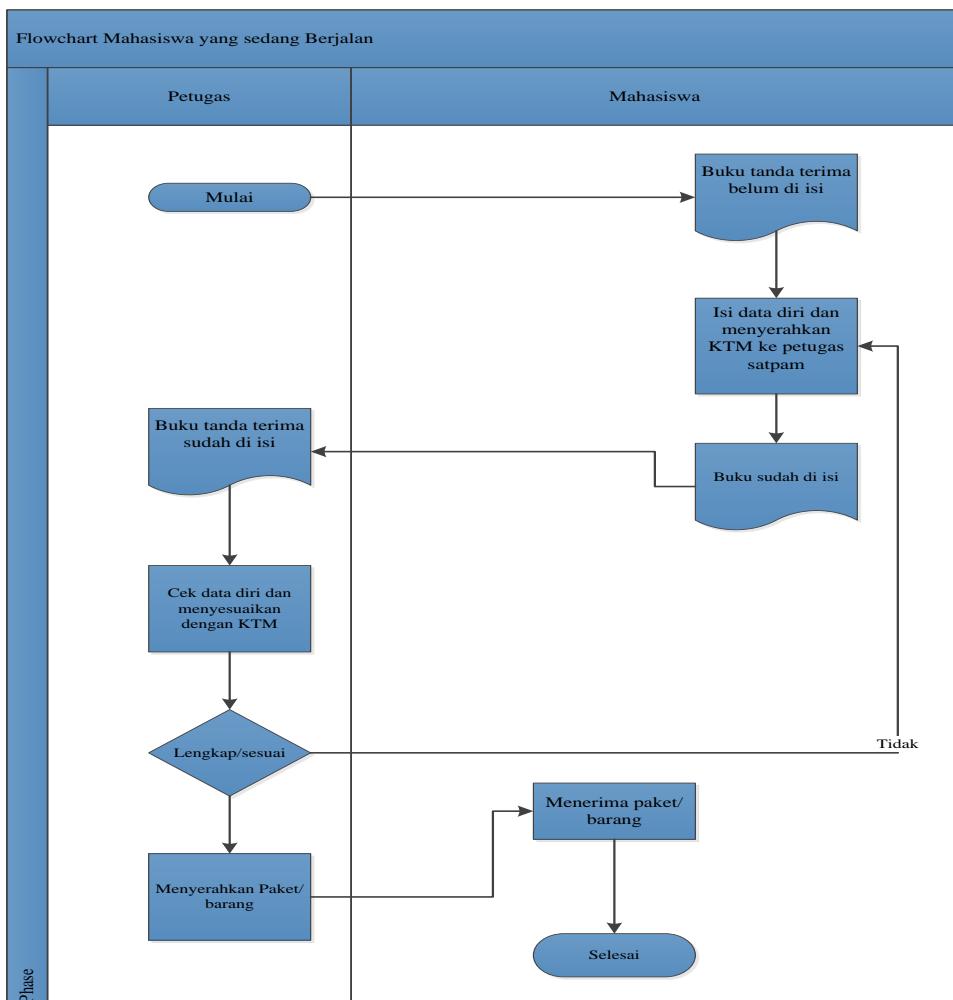


Gambar 56.1 Flochart Buku Tamu yang sedang berjalan

Keterangan :

1. Satpam memberikan buku tamu untuk tamu yang berkunjung
2. Tamu mengisi data diri di buku tamu sesuai dengan isi buku
3. Buku yang sudah diisi diserahkan kembali ke petugas satpam
4. Petugas mengecek buku tamu tersebut
5. Jika kurang lengkap maka tamu kembali mengisi data diri
6. Jika lengkap maka petugas akan mengarahkan tamu ke tujuan

5.1.1.2 Analisis Flowchart Buku Paket yang sedang berjalan

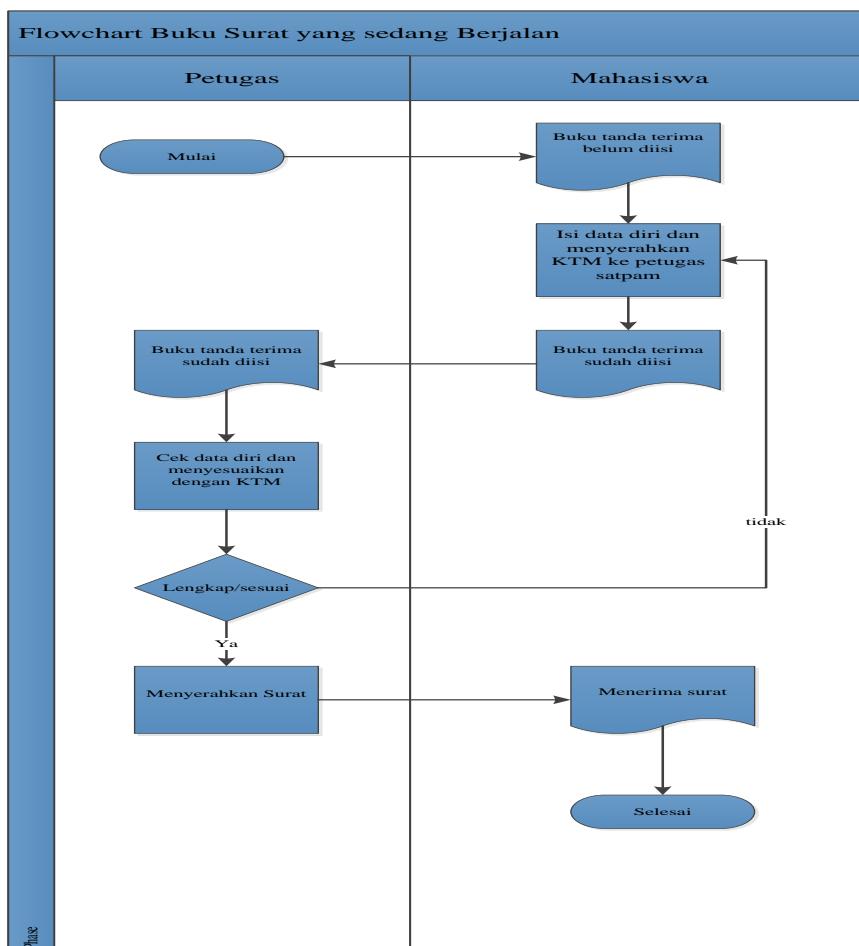


Gambar 57.2 Flochart Buku Paket yang sedang berjalan

Keterangan :

1. Satpam memberikan buku paket untuk mahasiswa yang mengambil paket
2. Siswa mengisi data diri di buku paket sesuai dengan data buku
3. Buku yang sudah diisi diserahkan kembali ke petugas satpam
4. Petugas mengecek buku paket tersebut dan menyesuaikannya dengan KTM
5. Jika kurang lengkap maka mahasiswa kembali mengisi data diri
6. Jika lengkap maka petugas akan menyerahkan paket
7. Mahasiswa menerima paket

5.1.1.3 Analisis Flowchart Buku Surat yang sedang berjalan



Gambar 58.3 Flochart Buku Surat yang sedang berjalan

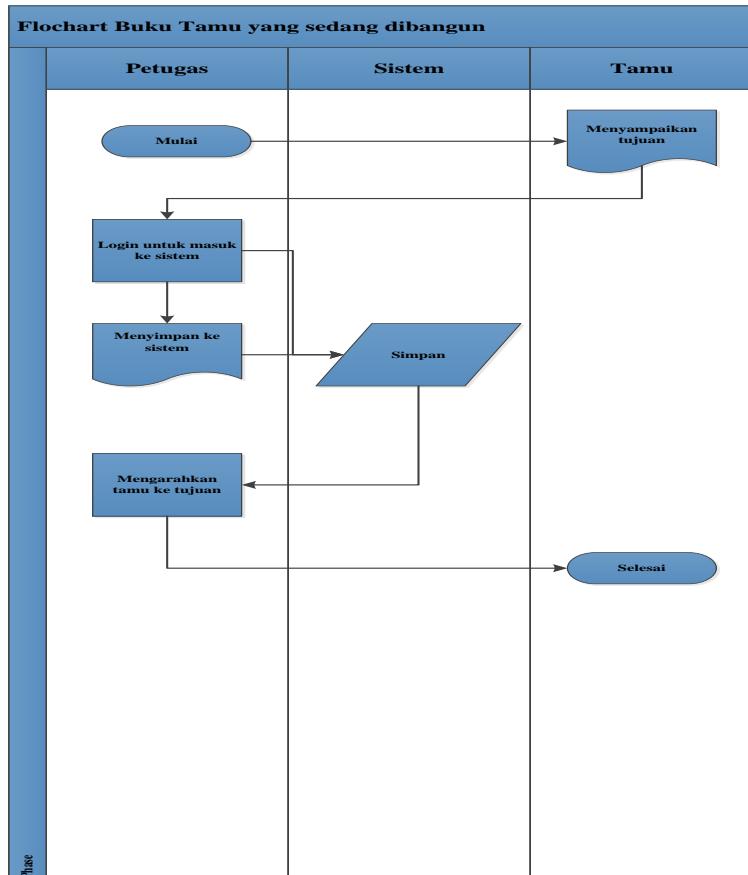
Keterangan :

1. Satpam memberikan buku surat untuk mahasiswa yang mengambil surat
2. Siswa mengisi data diri di buku surat sesuai dengan data buku
3. Buku yang sudah diisi diserahkan kembali ke petugas satpam
4. Petugas mengecek buku surat tersebut dan menyesuiakannya dengan KTM
5. Jika kurang lengkap maka mahasiswa kembali mengisi data diri
6. Jika lengkap maka petugas akan menyerahkan surat
7. Mahasiswa menerima surat

5.1.2 Analisis Sistem yang akan Dibangun

Analisis ini merupakan penggambaran proses yang akan di dalam perpustakaan. Proses yang ada dilakukan melibatkan sistem.

5.1.2.4 Analisis Flowchart Buku Tamu yang sedang di Bangun

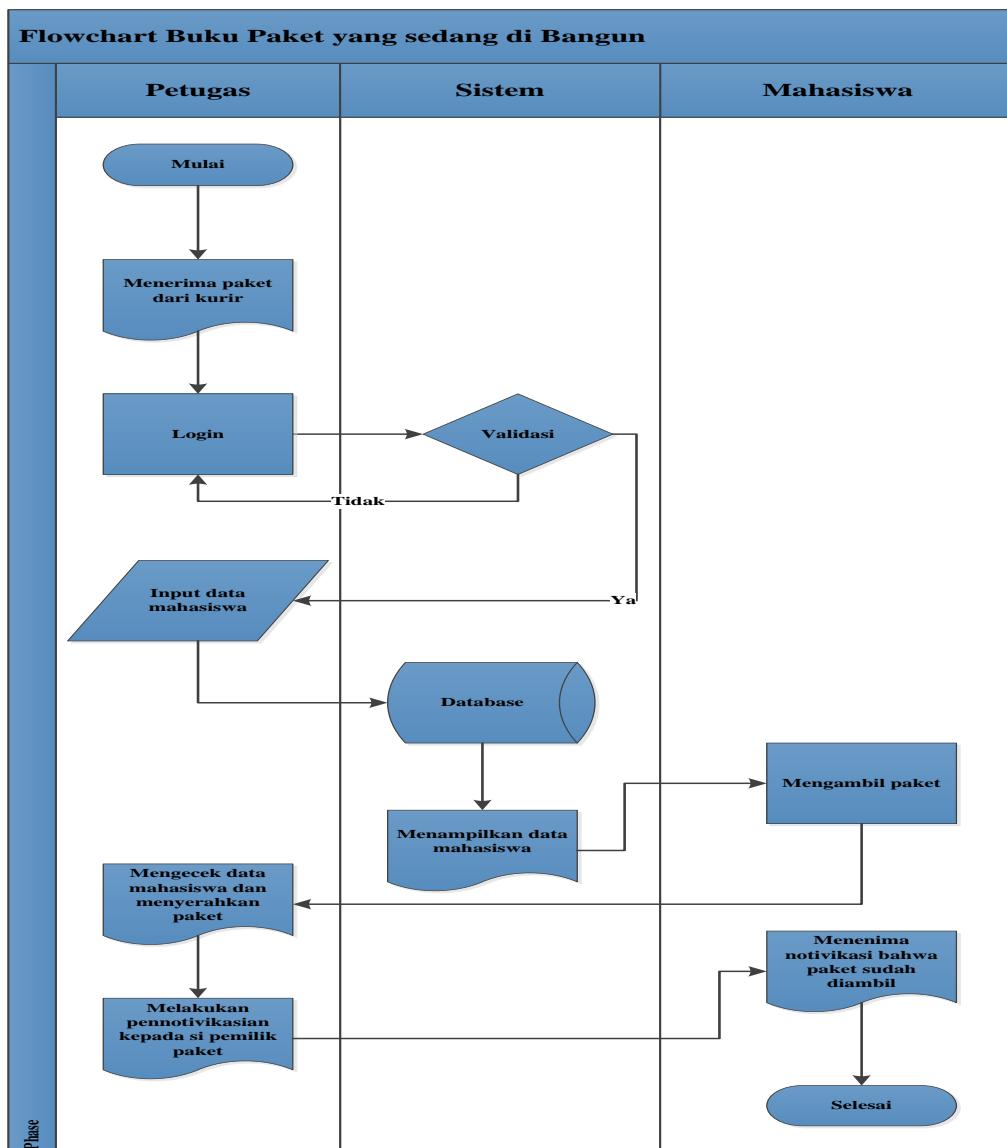


gambar 59.4 Flochart Buku Tamu yang sedang dibangung

Keterangan :

1. Tamu datang menyampaikan tujuan bertamu
2. Petugas masuk login untuk masuk ke system
3. Petugas menyimpan data tamu yang berkunjung
4. Data tamu tersimpan di system
5. Petugas mengarahkan tamu ke tujuannya.

5.1.2.5 Analisis Flowchart Buku Paket yang sedang di Bangun



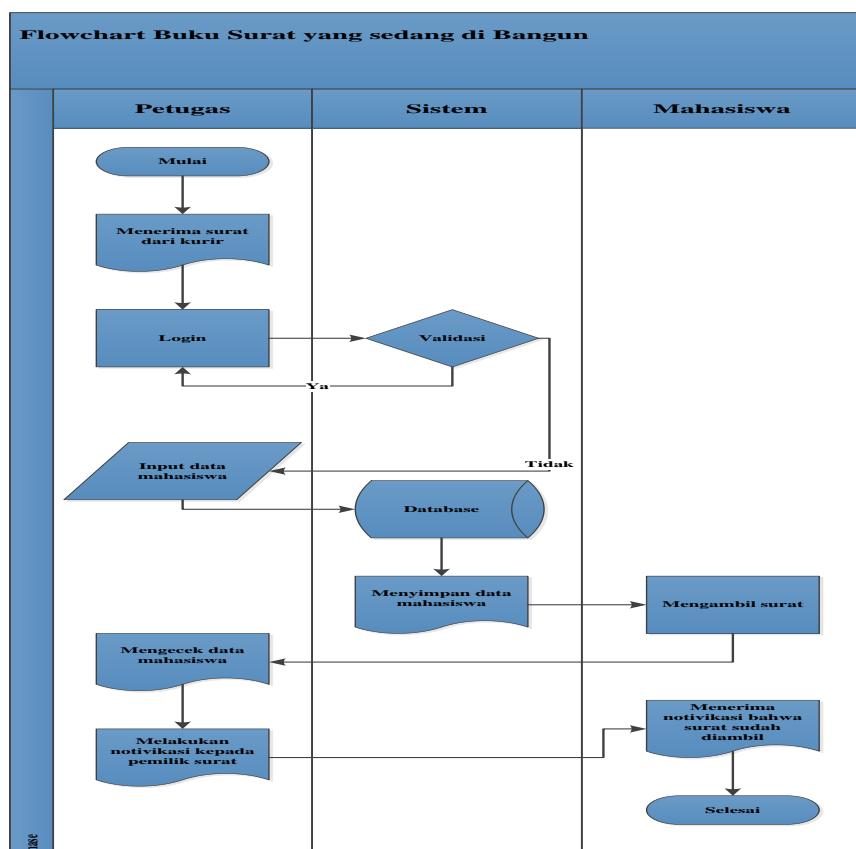
Gambar 60.5 Flochart Buku Paket yang sedang dibangung

Keterangan :

1. Petugas menerima paket dari kurir
2. Petugas melakukan login jika user name dan password salah maka petugas harus melakukan login lagi terlebih dahulu. Jika benar maka petugas menginputkan data mahasiswa yang ada di keterangan paket

3. Data mahasiswa sudah disimpan di database
4. Data mahasiswa tampil pada halaman mahasiswa
5. Mahasiswa mengambil paket
6. Petugas melakukan pengecekan pada data mahasiswa
7. Petugas melakukan notifikasi kepada pemilik paket bahwa paket telah dimbil
8. Menerima bukti notifikasi pengambilan paket sudah diambil

5.1.2.6 Analisis Flowchart Buku Surat yang sedang di Bangun



Gambar 61.6 Flochart Buku Surat yang sedang dibangun

Keterangan:

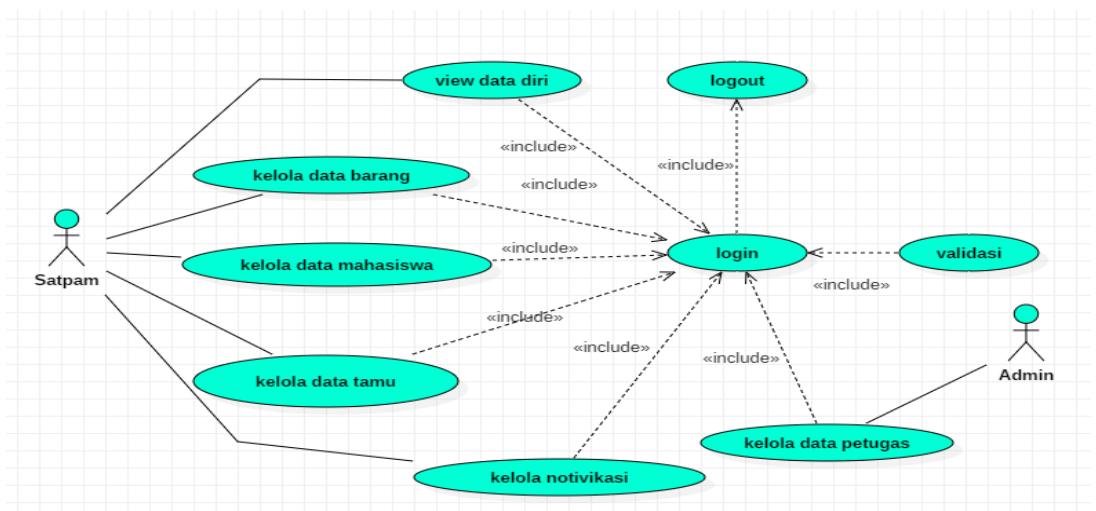
1. Petugas menerima surat dari kurir
2. Petugas melakukan login jika user name dan password salah maka petugas harus melakukan login lagi terlebih dahulu. Jika benar maka petugas menginputkan data mahasiswa yang ada di keterangan surat
3. Data mahasiswa sudah disimpan di database
4. Data mahasiswa tampil pada halaman mahasiswa
5. Mahasiswa mengambil surat
6. Petugas melakukan pengecekan pada data mahasiswa
7. Petugas melakukan notifikasi kepada pemilik surat, bahwa surat telah dimiliki
8. Menerima bukti notifikasi pengambilan surat sudah diambil

5.2 Perancangan

Perancangan merupakan tahap analisis untuk mendefinisikan kebutuhan-kebutuhan fungsional yang menggambarkan bagaimana suatu aplikasi dibentuk. Perancangan dapat berupa penggambaran, perencanaan dan pembuatan sketsa. Perancangan ini menitik beratkan kepada perancangan aplikasi, tahap perancangan data pada perangkat lunak tersebut dipakai kedalam pemodelan umum yang digunakan yaitu menggunakan *UML (Unified Modeling Language)* yang memberikan gambaran tentang proses yang terjadi dalam aplikasi, yang dapat berupa suatu proses untuk mendapatkan suatu informasi yang dicari.

5.2.1 Use Case Diagram

Use case diagram merupakan pemodelan untuk tingkah laku sistem informasi yang dibuat. *Use case* diagram mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi itu. Syarat penamaan *use case* nama didefinisikan sesederhana mungkin dan dapat dipahami.



Gambar 62.7 Use Case Diagram

5.2.1.1 Definisi Use Case

1. Definsi Aktor dan Use Case

Table 1.1 Definisi Use Case

No.	No Definisi	Nama Use Case	Keterangan
1.	UC-01	Login	Satpam melakukan <i>login</i> pada aplikasi Sistem Informasi Pengelolaan Paket di Kampus Politeknik Pos Indonesia
2.	UC-02	Validasi	Merupakan proses pengecekan hak akses apakah petugas masukkan benar atau tidak
3.	UC-03	Kelola data barang	Satpam dapat mengelola data barang yang datang

4.	UC-04	Kelola penerima data	Satpam dapat menambah data mahasiswa pada aplikasi Sistem Informasi Pengeloaan Paket di Kampus Politeknik Pos Indonesia
5.	UC-05	Kelola status diri	Satpam mengelola status data diri
6.	UC-06	Notifikasi	Satpam memberikan notifikasi ke pada mahasiswa
7.	UC-07	<i>Logout</i>	Satpam melakukan <i>logout</i> pada aplikasi Sistem Informasi Pengelolaan Paket di Kampus Politeknik Pos Indonesia
8.	UC-08	Kelola Petugas	Admin dapat menambahkan data petugas untuk loginnya petugas

Table 2.2 Definisi Aktor

No.	Aktor	Keterangan
1.	Satpam	Satpam yang dapat mengelola system
2.	Admin	Admin yang dapat mengelola petugas

5.2.1.2 Skenario Use Case

Skenario *Use case* mendeskripsikan urutan langkah-langkah dalam proses bisnis, baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Berikut ini penjelasan dari masing-masing skenario tersebut:

1. Skenario *Use Case* pada *Login*

Table 3.3 Skenario Use Case Login

Identifikasi	
Nomor	UC-01

Nama	<i>Login</i>
Tujuan	Memberikan Layanan untuk melakukan <i>Login</i> dan menuju halaman utama web
Deskripsi	
Aktor	Satpam
Skenario Utama	
Kondisi Awal	Halaman <i>Login web</i>
Aksi Aktor	Reaksi Sistem
<ol style="list-style-type: none"> 1. Memasukkan <i>username</i> dan <i>password</i>. 2. Mencocokan data <i>login</i> dengan data <i>user</i> pada <i>database</i>. 3. Bila valid, maka akan memunculkan halaman selanjutnya 	
Kondisi Akhir	Menampilkan halaman utama

2. Skenario Use Case pada Validasi

Table 4.4 Skenario Use Case Validasi

Identifikasi	
Nomor	UC-02
Nama	Validasi
Tujuan	Memastikan data yang dimasukkan telah benar atau belum.
Deskripsi	

Aktor	Satpam
Skenario Utama	
Kondisi Awal	Halaman <i>Login web</i>
Aksi Aktor	Reaksi Sistem
<ol style="list-style-type: none"> 1. User masukkan <i>Username</i> dan <i>Password</i> 3. Tekan tombol <i>login now</i> 	<ol style="list-style-type: none"> 2. Sistem akan menerima masukkan data. 4. Sistem akan memvalidasi data yang baru dimasukkan. Apabila data benar maka sistem akan memunculkan halaman selanjutnya. Jika salah atau kurang lengkap, maka sistem akan meminta mengisi kembali <i>username</i> dan <i>password</i>.
Kondisi Akhir	Memasuki halaman utama

3. Skenario Use Case pada Data Barang

Table 5.5 Skenario Use Case Tambah Barang

Identifikasi	
Nomor	UC-03
Nama	Tambah barang
Tujuan	Memberikan Layanan untuk menambahkan data barang
Deskripsi	

Aktor	Satpam
Skenario Utama	
Kondisi Awal	Halaman Data barang <i>web</i>
Aksi Aktor	Reaksi Sistem
Petugas melakukan <i>input</i> data	Memasukan data ke <i>database</i> .
barang.	
Kondisi Akhir	Menampilkan data barang yang telah dimasukkan oleh petugas pada Halaman Data Barang

4. Skenario Use Case pada Penerima

Table 6.6 Skenario Use Case Penerima

Identifikasi	
Nomor	UC-04
Nama	Penerima
Tujuan	Memeriksa dan penyimpanan data penerima
Deskripsi	
Aktor	Satpam
Skenario Utama	
Kondisi Awal	Halaman Data mahasiswa <i>web</i>
Aksi Aktor	Reaksi Sistem
Satpam melakukan input data penerima	Memasukan data ke <i>database</i> .
Kondisi Akhir	Menampilkan data penerima yang telah dimasukkan oleh satpam pada Halaman penerima.

1. Skenario Use Case pada Notifikasi

Table 7.7 Skenario Use Case Notifikasi

Identifikasi	
Nomor	UC-03
Nama	Notifikasi
Tujuan	Memberikan Layanan pemberitahuan
Deskripsi	
Aktor	Satpam
Skenario Utama	
Kondisi Awal	Halaman Data no tlpn <i>web</i>

Aksi Aktor	Reaksi Sistem
Satpam melakukan pendataan no tlfn	Masuk ke halaman data penerima dan simpan dalam <i>database</i> .
Kondisi Akhir	Menampilkan data penerima yang telah diubah oleh admin pada halaman Data Penerima

2. Skenario Use Case pada Update Anggota

Table 8.8 Skenario Use Case Update Anggota

Identifikasi	
Nomor	UC-04
Nama	<i>Update Anggota</i>
Tujuan	Memberikan Layanan untuk mengubah data anggota
Deskripsi	
Aktor	Satpam
Skenario Utama	
Kondisi Awal	Halaman Data Anggota <i>web</i>
Aksi Aktor	
Petugas melakukan ubah data yang sudah ada di data anggota.	Mengubah data yang sudah ada dan di simpan dalam <i>database</i> .
Kondisi Akhir	Menampilkan kembali data anggota yang telah diubah oleh petugas pada Halaman

Data Anggota.

3. Skenario Use Case pada Mengkonfirmasi Penerima

Table 9.9 Skenario Use Case Mengkonfirmasi Penerima

Identifikasi	
Nomor	UC-06
Nama	Mengkonfirmasi Penerima
Tujuan	Memberikan Layanan untuk mengkonfirmasi Penerima
Deskripsi	
Aktor	Satpam
Skenario Utama	
Kondisi Awal	Halaman Data Mahasiswa <i>web</i>
Aksi Aktor	Reaksi Sistem
Satpam melakukan konfirmasi untuk pemberitahuan barang	Mengubah status barang.
Kondisi Akhir	Menampilkan data barang yang telah dikonfirmasi oleh satpam pada Halaman Data Penerima.

4. Skenario Use Case pada Melakukan Logout

Table 10.10 Skenario Use Case Melakukan Logout

Identifikasi	
Nomor	UC-07
Nama	Melakukan <i>Logout</i>
Tujuan	Memberikan Layanan untuk melakukan <i>Logout</i> dan menuju halaman <i>Login</i>
Deskripsi	
Aktor	Satpam
Skenario Utama	
Kondisi Awal	Halaman Utama <i>web</i>
Aksi Aktor	Reaksi Sistem
Petugas melakukan proses <i>logout</i> .	Keluar dari halaman utama
Kondisi Akhir	Kembali pada Halaman <i>Login</i>

5. Skenario Use Case pada Tambah Penerima

Table 11.11 Skenario Use Case Tambah Penerima

Identifikasi	
Nomor	UC-08
Nama	Tambah petugas
Tujuan	Memberikan Layanan untuk menambahkan data petugas

Deskripsi

Aktor Admin

Skenario Utama

Kondisi Awal Halaman Data petugas *web*

Aksi Aktor **Reaksi Sistem**

Admin melakukan *input* data Memasukan data ke *database*.
petugas.

Kondisi Akhir Menampilkan data Penerima yang telah
dimasukkan oleh satpam pada Halaman
Data penerima.

6. Skenario Use Case pada Update Petugas

Table 12.12 Skenario Use Case Update Petugas

Identifikasi

Nomor UC-08

Nama *Update Petugas*

Tujuan Memberikan Layanan untuk mengubah data Petugas

Deskripsi

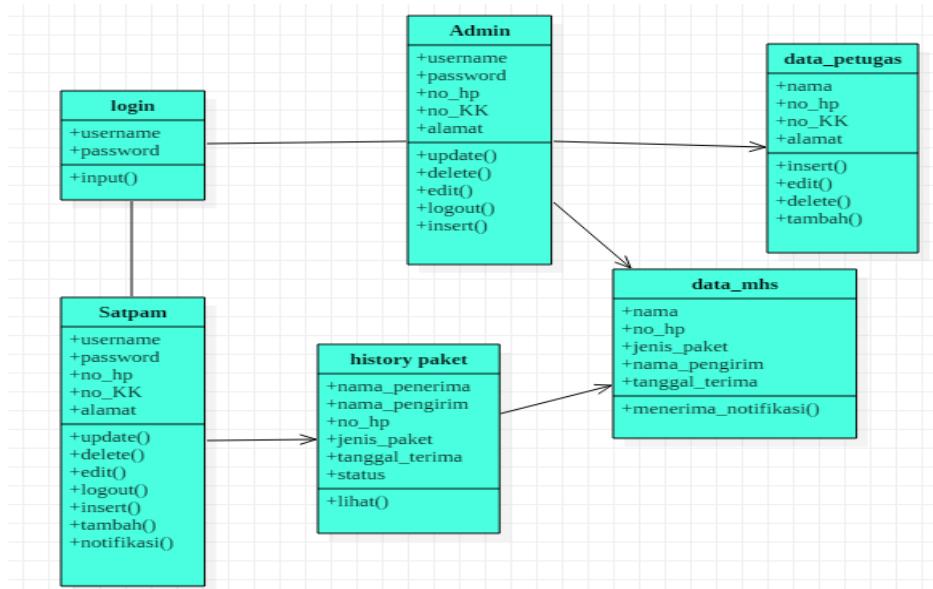
Aktor Petugas

Skenario Utama

Kondisi Awal	Halaman Data Petugas <i>web</i>
Aksi Aktor	Reaksi Sistem
Admin melakukan ubah data yang sudah ada di data petugas.	Mengubah data yang sudah ada dan disimpan dalam <i>database</i> .

5.2.2 Class Diagram

Class diagram adalah sebuah *class* yang menggambarkan struktur dan penjelasan *class*, paket, dan objek serta hubungan satu sama lain. *Class* diagram juga menjelaskan antara *class* dalam sebuah sistem yang sedang berjalan dan bagaimana cara sebuah *class* tersebut saling terhubung ke sebuah sistem untuk mencapai sebuah tujuan.

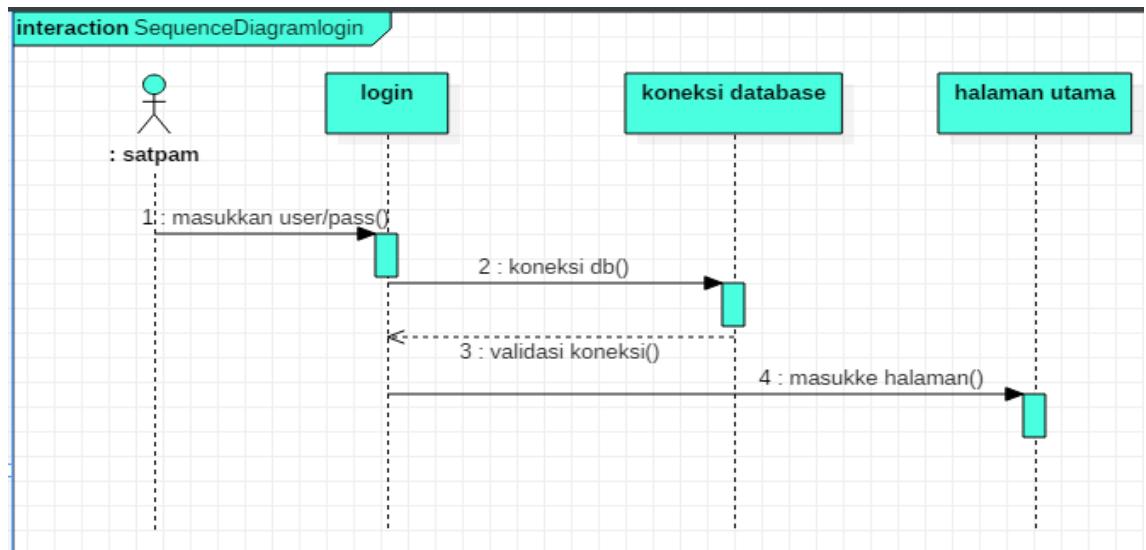


Gambar 63.8 Class Diagram

5.2.3 Sequence Diagram

Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

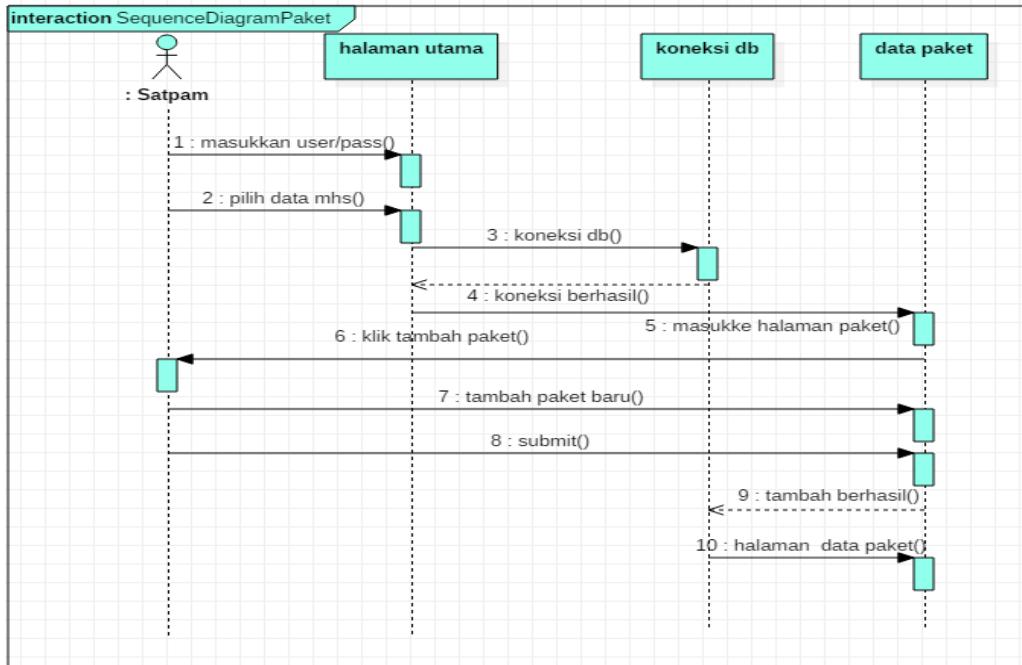
5.2.3.1 Sequence Diagram Login



Gambar 64.9 Sequence Diagram Login

Sequence diagram ini menjelaskan proses *login*. Petugas mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian petugas memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama.

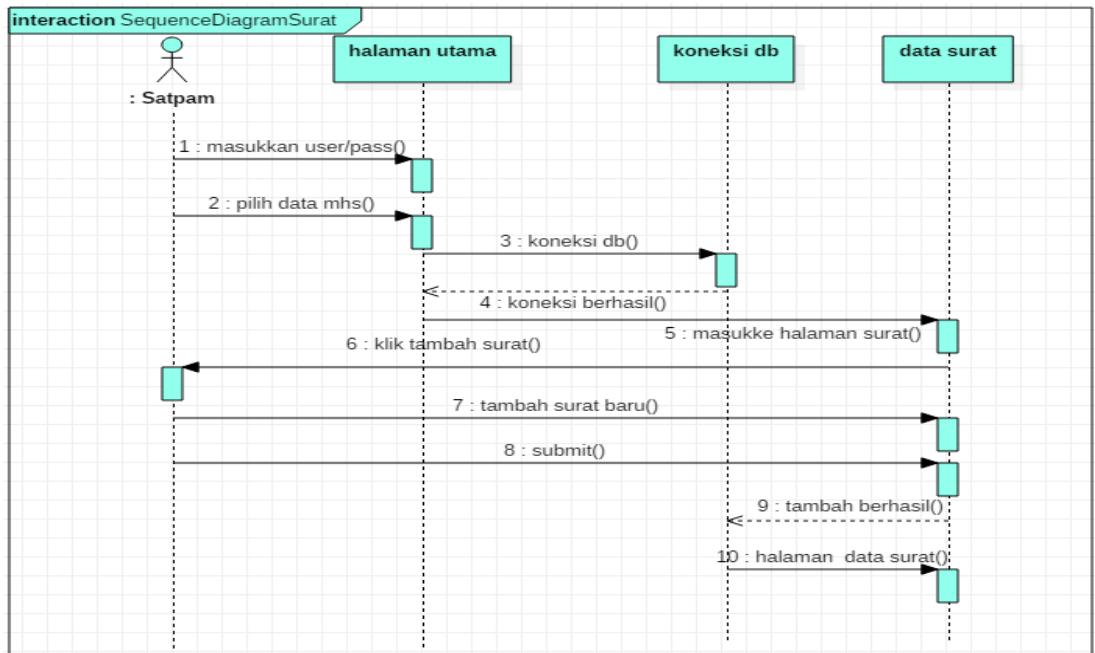
5.2.3.2 Sequence Diagram Paket



Gambar 65.10 Sequence Diagram Paket

Sequence diagram ini menjelaskan proses Tambah Paket. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data paket dan klik tambah paket. Nanti akan ditampilkan *form* data paket di mana satpam dapat menambahkan paket baru.

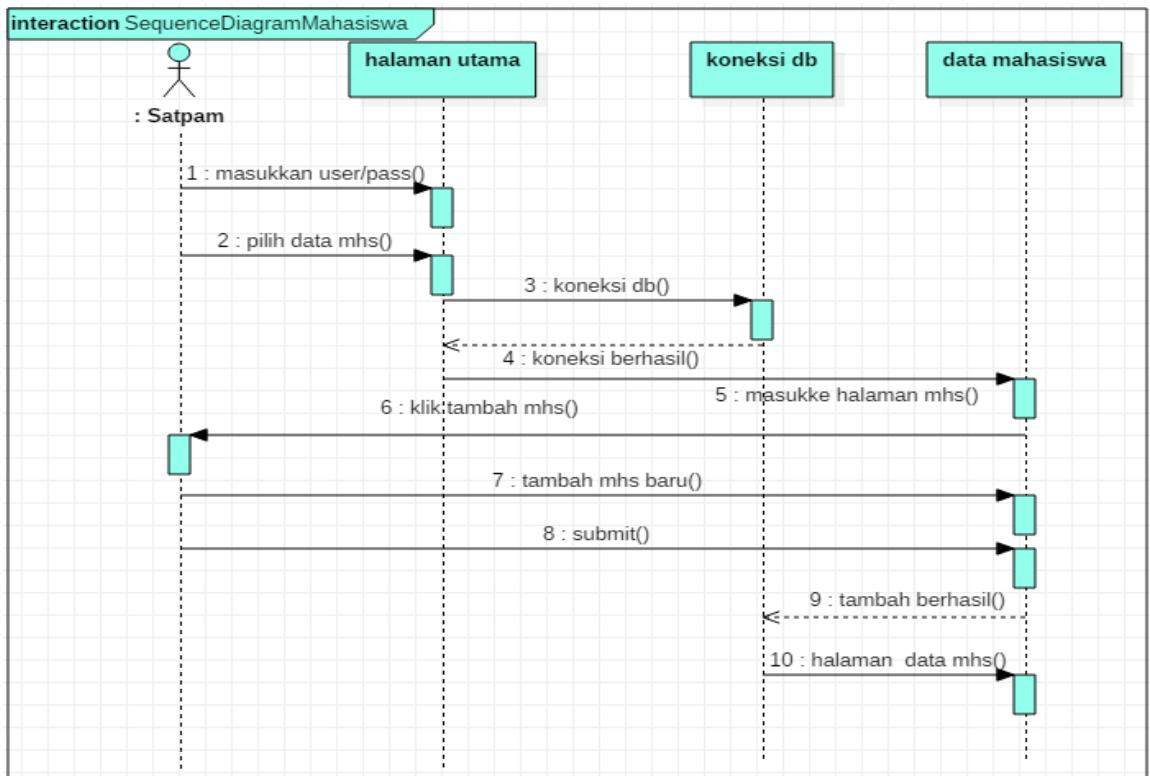
5.2.3.3 Sequence Diagram Surat



Gambar 66.11 Sequence Diagram Surat

Sequence diagram ini menjelaskan proses Tambah Surat. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data surat dan klik tambah surat. Nanti akan ditampilkan *form* data surat di mana satpam dapat menambahkan surat baru.

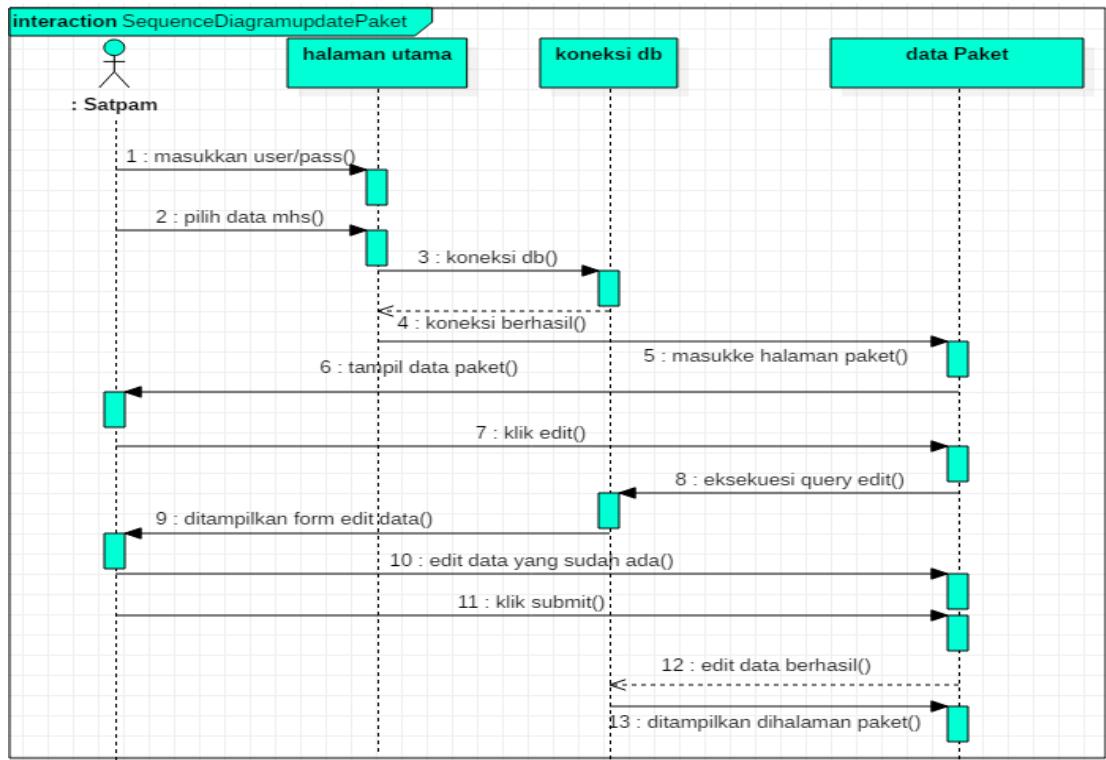
5.2.3.4 Sequence Diagram Penerima



Gambar 67.12 Sequence Diagram Penerima

Sequence diagram ini menjelaskan proses Tambah Penerima. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data mahasiswa dan klik tambah surat. Nanti akan ditampilkan *form* data surat di mana satpam dapat menambahkan mahasiswa baru.

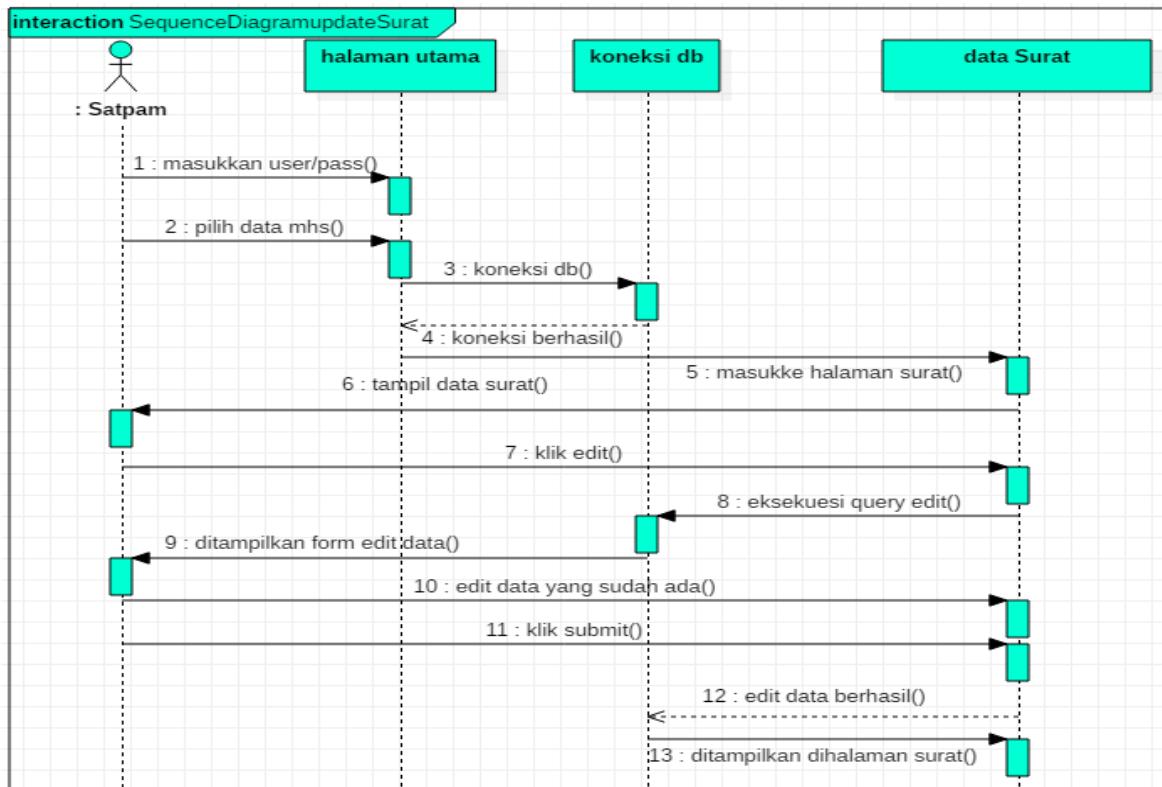
5.2.3.5 Sequence Diagram Update Paket



Gambar 68.13 Sequence Diagram Update Paket

Sequence diagram ini menjelaskan proses *Update Paket*. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data paket dan akan ditampilkan data paket yang sudah ada yang dapat diubah.

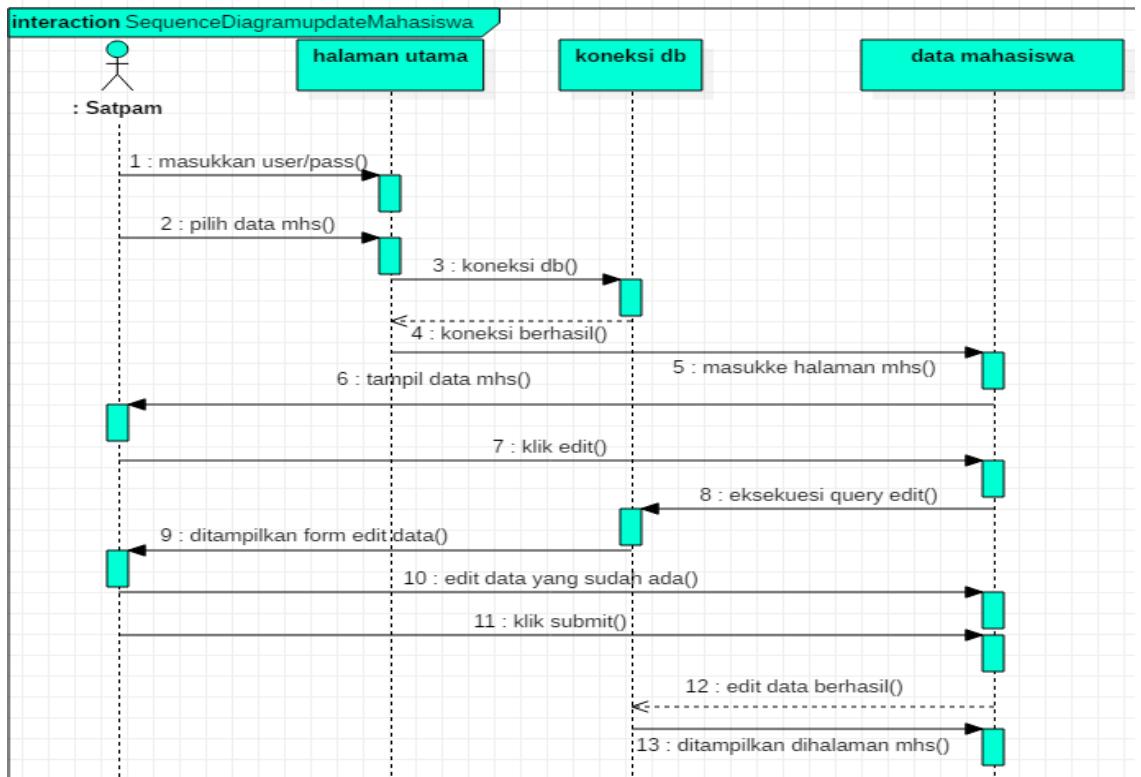
5.2.3.6 Sequence Diagram Update Surat



Gambar 69. 14 Sequence Diagram Update Surat

Sequence diagram ini menjelaskan proses *Update Surat*. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data surat dan akan ditampilkan data surat yang sudah ada yang dapat diubah.

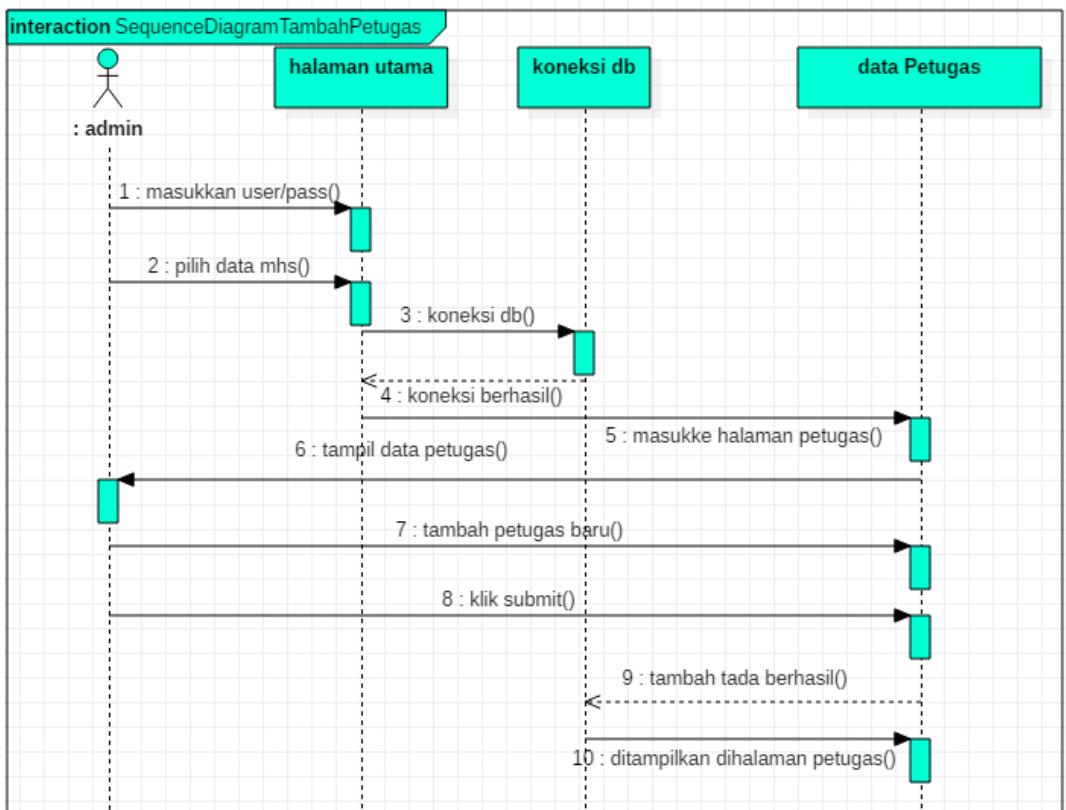
5.2.3.7 Sequence Diagram Update Penerima



Gambar 70.15 Sequence Diagram Update Penerima

Sequence diagram ini menjelaskan proses *Update* Penerima. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data penerima dan akan ditampilkan data penerima yang sudah ada yang dapat diubah.

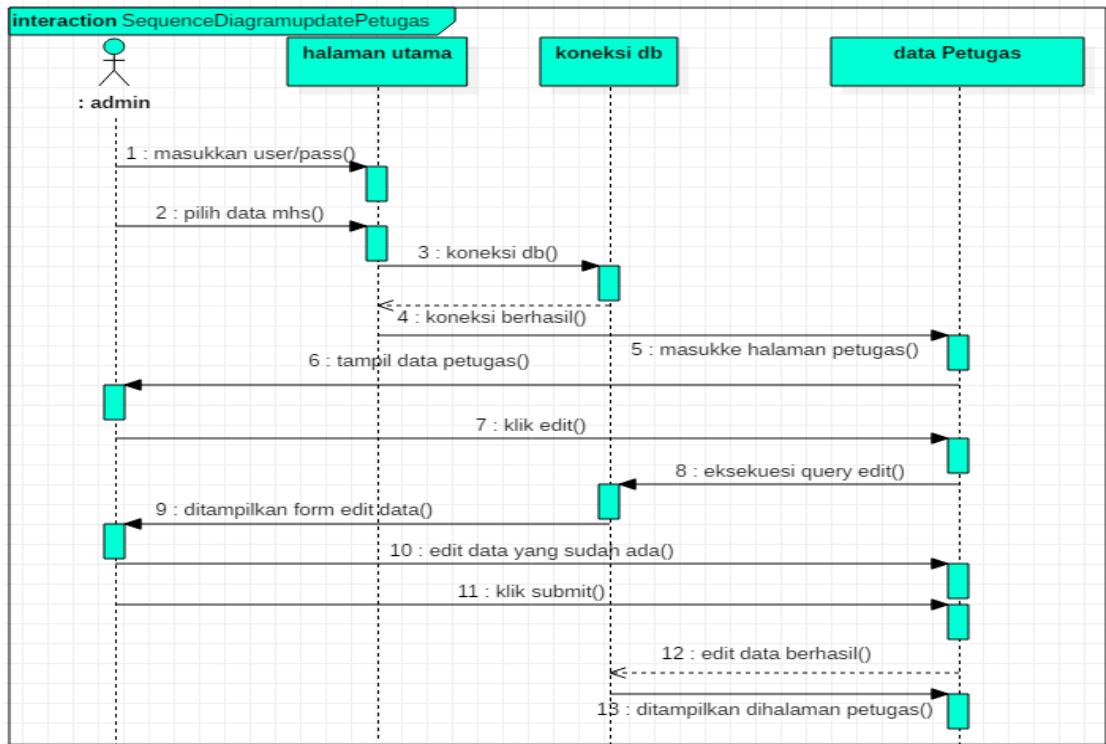
5.2.3.8 Sequence Diagram Tambah Petugas



Gambar 71.16 Sequence Diagram Tambah Petugas

Sequence diagram ini menjelaskan proses Tambah Petugas. Admin mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian admin memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data user dan klik tambah petugas. Nanti akan ditampilkan *form* data petugas di mana petugas dapat menambahkan petugas baru.

5.2.3.9 Sequence Diagram Update Petugas



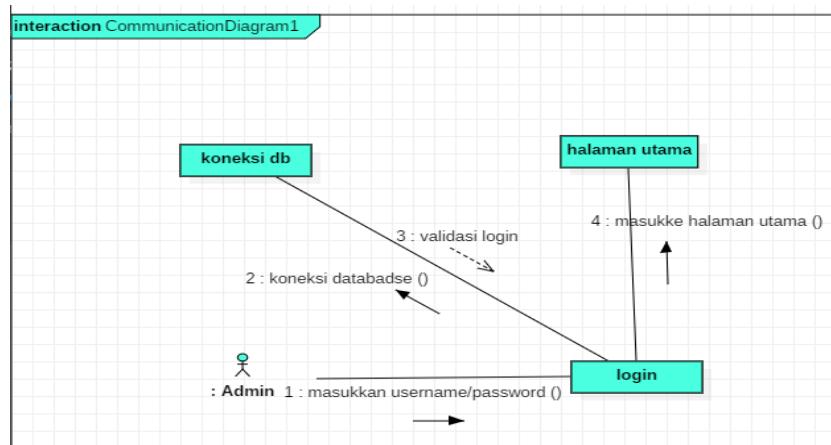
Gambar 72.17 Sequence Diagram Update Petugas

Sequence diagram ini menjelaskan proses *Update Petugas*. Admin mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian admin memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data user dan akan ditampilkan data user yang sudah ada yang dapat diubah.

5.2.4 Collaboration Diagram

Diagram kolaborasi mengelompokkan *message* pada kumpulan diagram sekuen menjadi sebuah diagram. *Collaboration diagram* juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*.

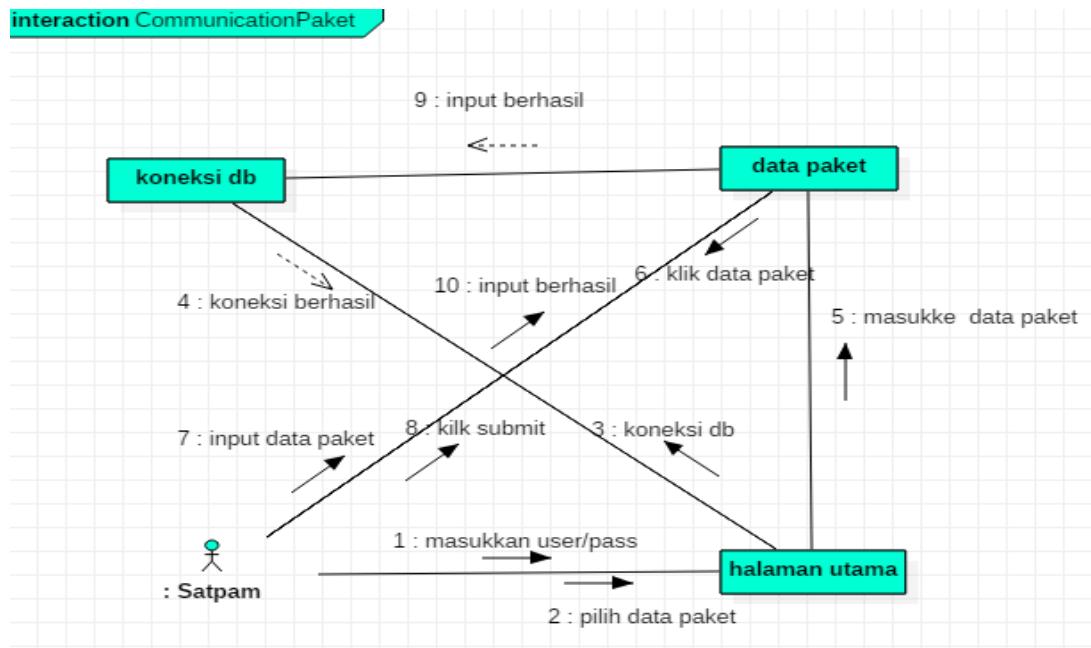
5.2.4.1 Collaboration Diagram Login



Gambar 73.18 Collaboration Diagram Login

Collaboration diagram login ini menjelaskan proses *Login*. Petugas mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian petugas memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama.

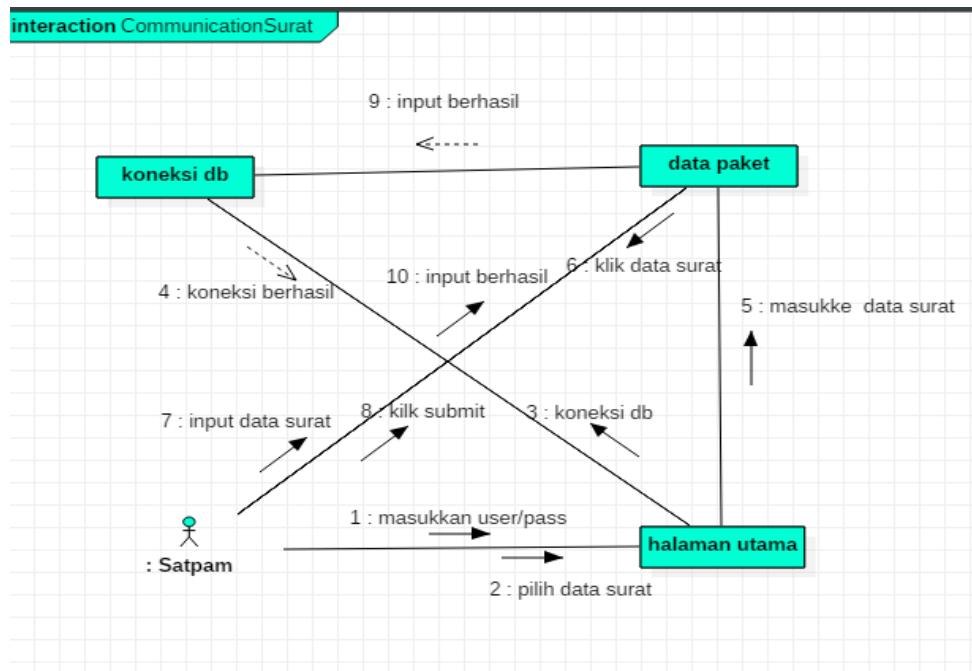
5.2.4.2 Collaboration Diagram Paket



Gambar 74.19 Collaboration Diagram Paket

Collaboration diagram ini menjelaskan proses Tambah Paket. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data paket dan klik tambah paket. Nanti akan ditampilkan *form* data paket di mana satpam dapat menambahkan paket baru.

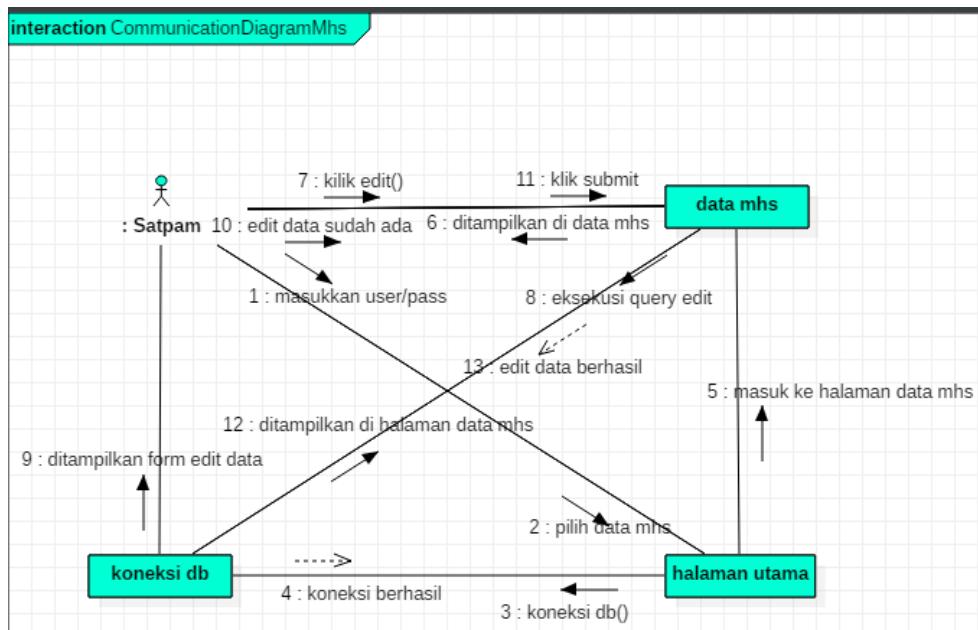
5.2.4.3 Collaboration Diagram Surat



Gambar 75.20 Collaboration Diagram Surat

Collaboration diagram ini menjelaskan proses Tambah Surat. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data surat dan klik tambah surat. Nanti akan ditampilkan *form* data surat di mana satpam dapat menambahkan paket baru.

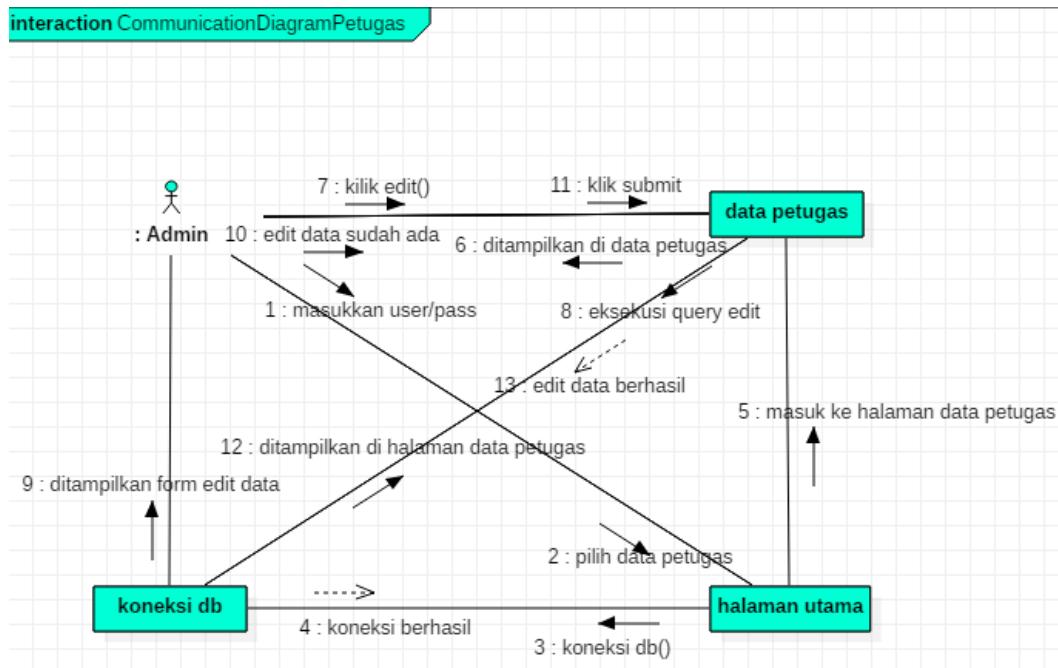
5.2.4.6 Collaboration Update Penerima



Gambar 76..23 Collaboration Diagram update Penerima

Collaboration diagram ini menjelaskan proses update penerima. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data paket dan klik update penerima. Nanti akan ditampilkan *form* data surat di mana satpam dapat mengubah data penerima baru.

5.2.4.7 Collaboration Update Petugas

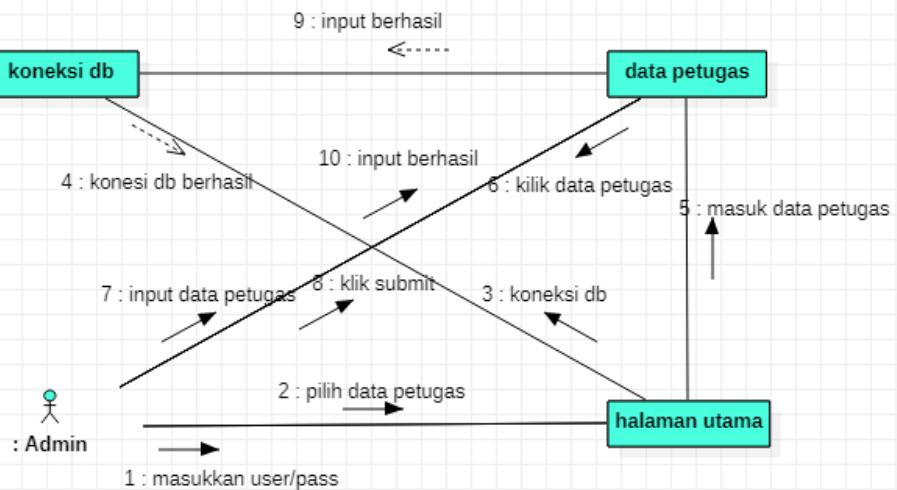


Gambar 77.24 Collaboration Diagram update Petugas

Collaboration diagram ini menjelaskan proses update Petugas. Satpam mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian satpam memasukkan *username* dan *password*. Jika sesuai maka akan didapatkan tampilan halaman utama. Lalu pilih menu data paket dan klik ubah mahasiswa. Nanti akan ditampilkan *form* data surat di mana satpam dapat mengubah data siswa baru.

5.2.4.8 Collaboration Update tambah Petugas

interaction CommunicationDiagramtambahPtgs

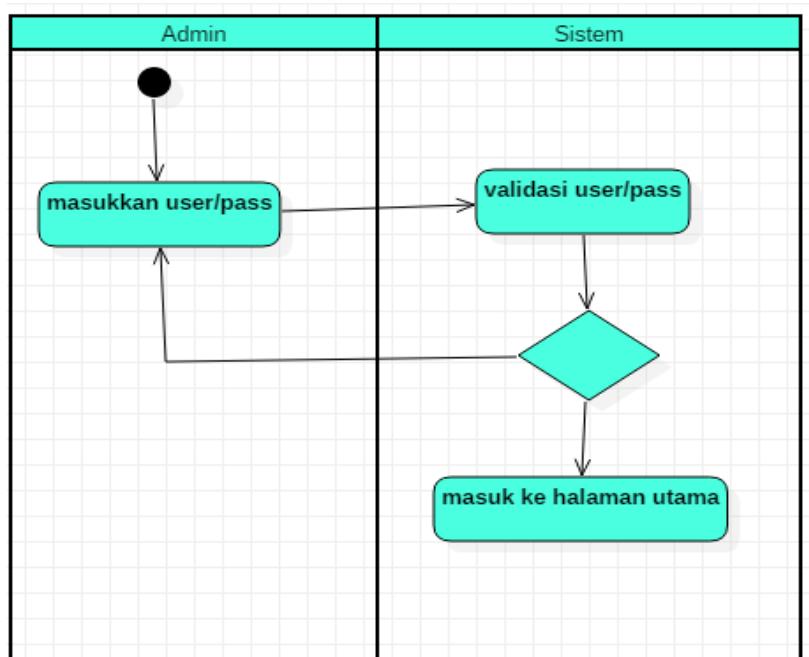


Gambar 78.25 Collaboration Diagram tambah Petugas

5.2.5 Activity Diagram

Activity Diagram menggambarkan *workflow* suatu sistem atau proses bisnis. Diagram ini menunjukkan suatu alur kegiatan suatu berurutan. Diagram ini digunakan untuk mendeskripsikan kegiatan-kegiatan dalam sebuah operasi meskipun juga dapat digunakan untuk mendeskripsikan alur kegiatan yang lainnya seperti *use case* atau suatu interaksi.

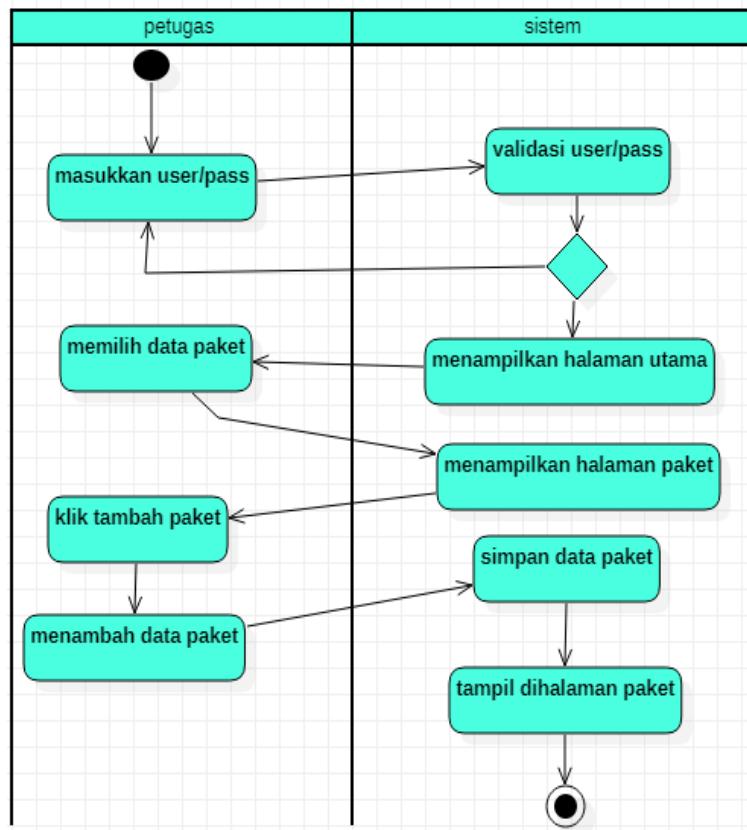
5.2.5.1 Activity Diagram Login



Gambar 79.26 Activity Diagram Login

Admin membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama.

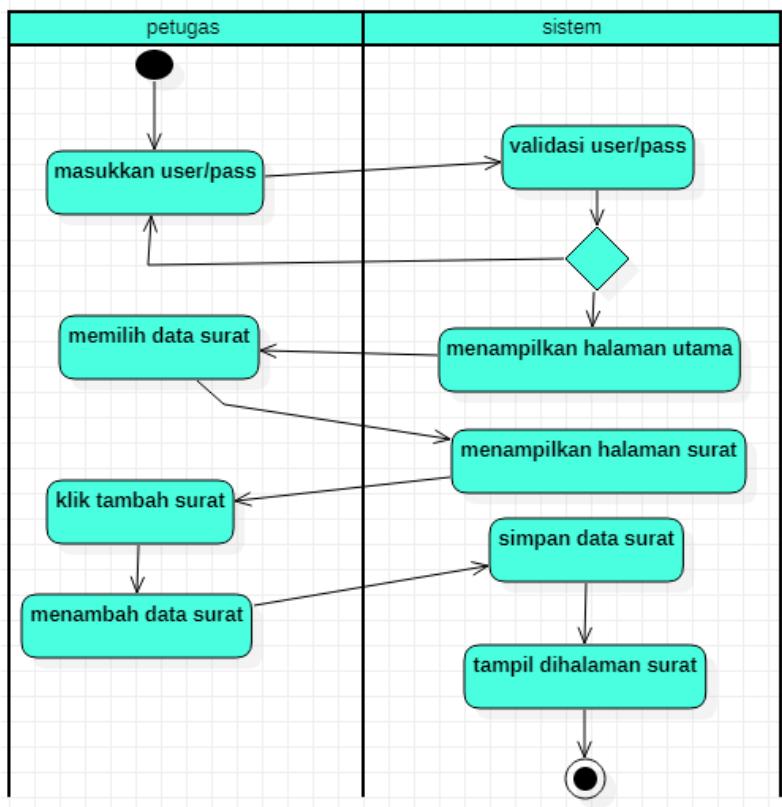
5.2.5.2 Activity Diagram tambah Paket



Gambar 80.27 Activity Diagram tambah Paket

Petugas membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data paket dan klik tambah paket. Nanti akan ditampilkan *form* data petugas di mana petugas dapat menambahkan paket baru.

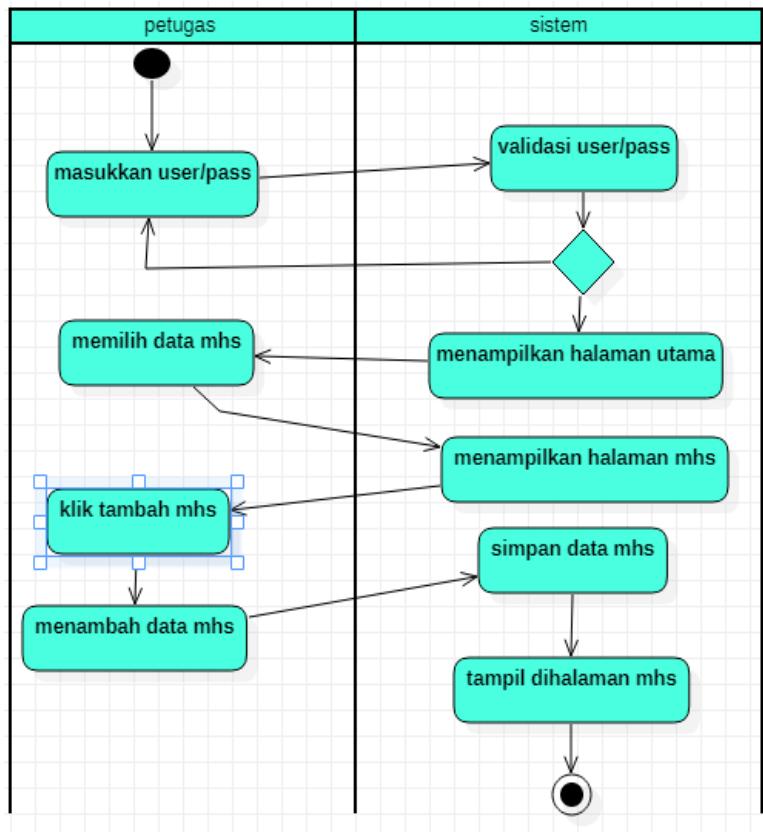
5.2.5.3 Activity Diagram tambah Surat



Gambar 81.28 Activity Diagram tambah Surat

Petugas membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data surat dan klik tambah surat. Nanti akan ditampilkan *form* data petugas di mana petugas dapat menambahkan surat baru.

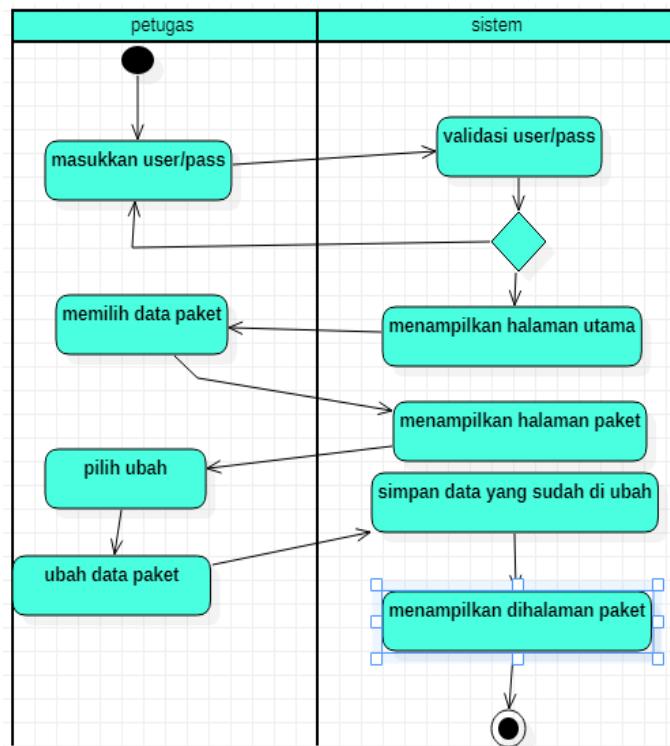
5.2.5.4 Activity Diagram tambah Penerima



Gambar 82.29 Activity Diagram tambah Penerima

Petugas membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data penerima dan klik tambah penerima. Nanti akan ditampilkan *form* data petugas di mana petugas dapat menambahkan penerima baru.

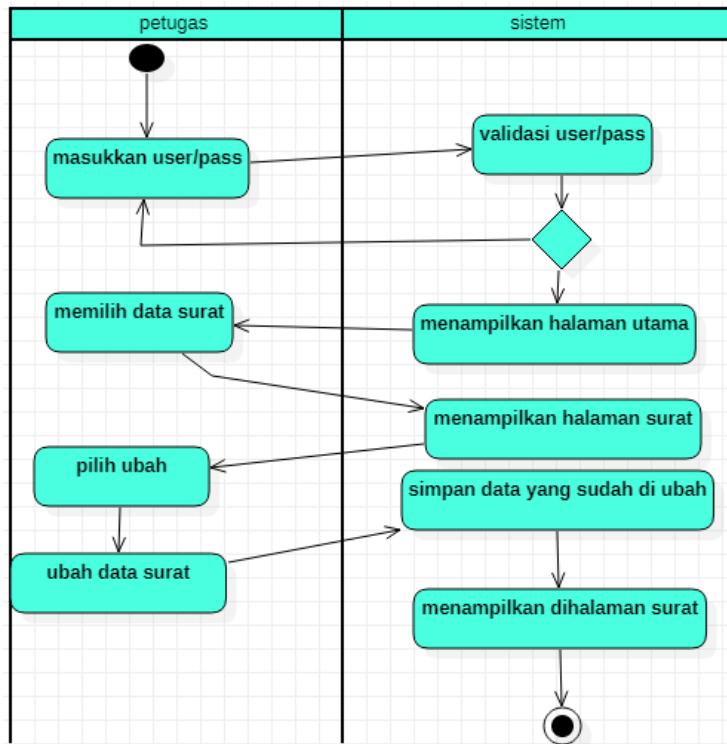
5.2.5.5 Activity Diagram Update Paket



Gambar 83.30 Activity Diagram Update Paket

Petugas membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data paket dan akan ditampilkan data paket yang sudah ada yang dapat diubah.

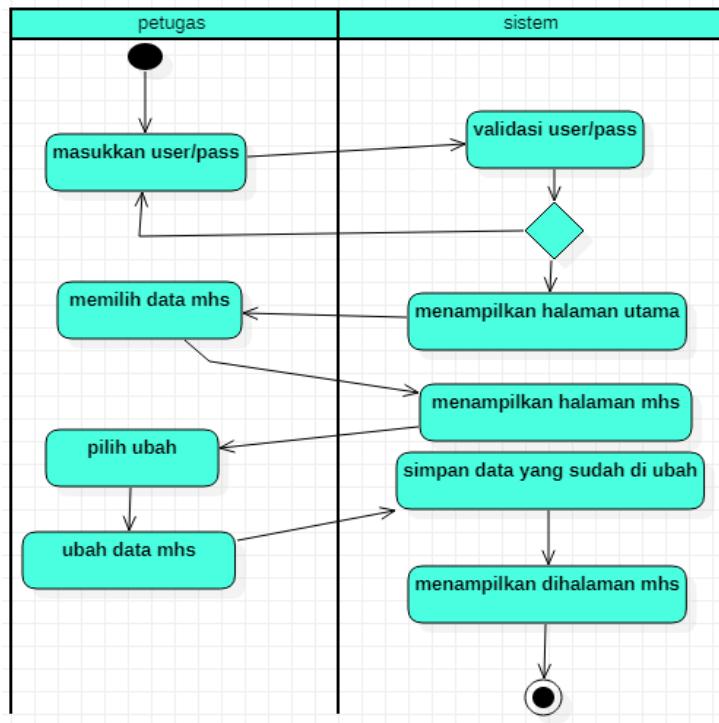
5.2.5.6 Activity Diagram Update Surat



Gambar 84.31 Activity Diagram Update Surat

Petugas membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data surat dan akan ditampilkan data surat yang sudah ada yang dapat diubah.

5.2.5.7 Activity Diagram Update Penerima



Gambar 85.32 Activity Diagram Update Penerima

Petugas membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data penerima dan akan ditampilkan data mahasiswa yang sudah ada yang dapat diubah.

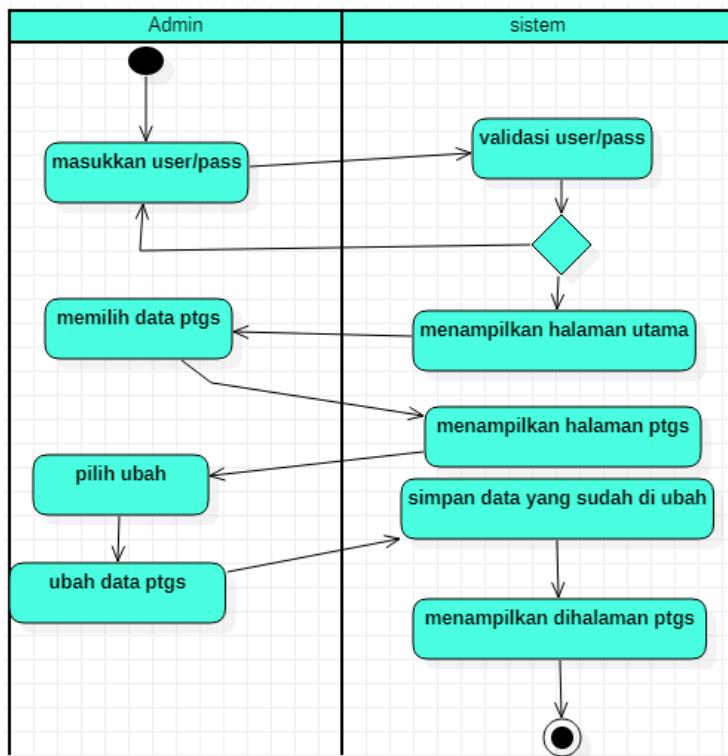
5.2.5.8 Activity Diagram tambah Petugas



Gambar 86.33 Activity Diagram tambah Petugas

Petugas membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data petugas dan akan ditampilkan data petugas yang sudah ada yang dapat diubah.

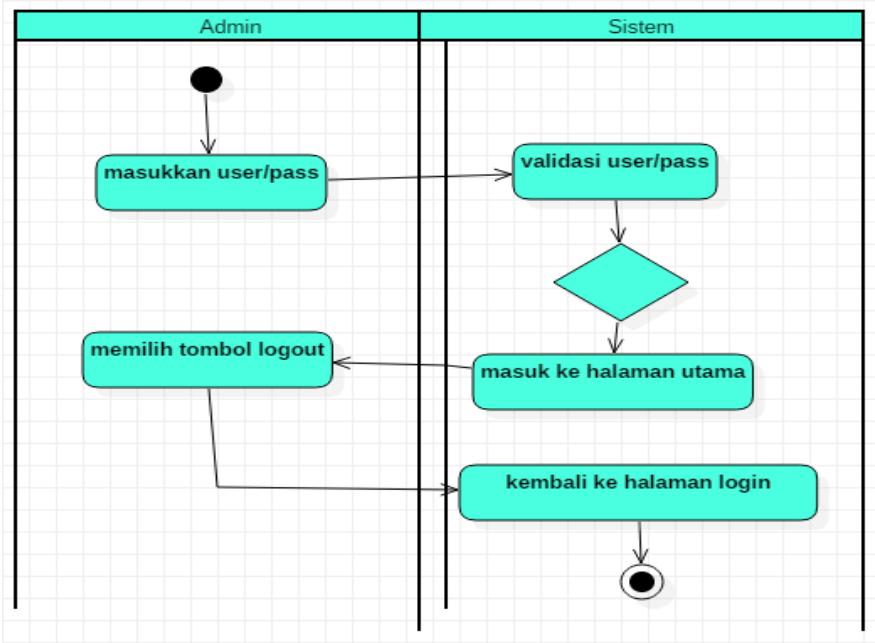
5.2.5.9 Activity Diagram Update Petugas



Gambar 87.34 Activity Diagram Update Petugas

Petugas membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data petugas dan akan ditampilkan data petugas yang sudah ada yang dapat diubah.

5.2.5.10 Activity Diagram Logout



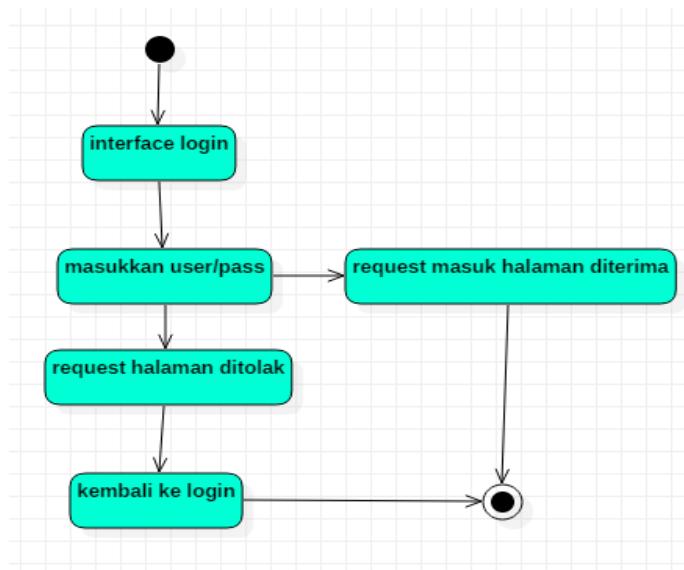
Gambar 88.35 Activity Diagram Logout

Admin membuka halaman *login* lalu memasukan *username* dan *password*, jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Admin dapat melakukan *logout* jika sudah tidak ada lagi yang akan dikerjakan.

5.2.6 Statechart Diagram

Diagram status digunakan untuk menyatakan kondisi (status) sebuah objek pada sistem informasi berjalan. Diagram ini diadopsi dari penggambaran kondisi mesin status (*state machine*) yang menggambarkan status apa saja yang dialami oleh mesin.

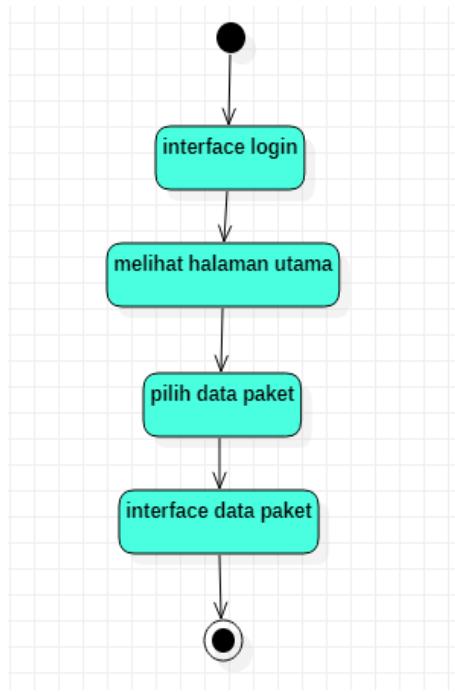
5.2.6.1 Statechart Diagram Login



Gambar 89.36 Statechart Diagram Login

Statechart diagram login ini menjelaskan proses *Login*. Petugas mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian petugas memasukkan *username* dan *password* pada *interface login*. Jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama.

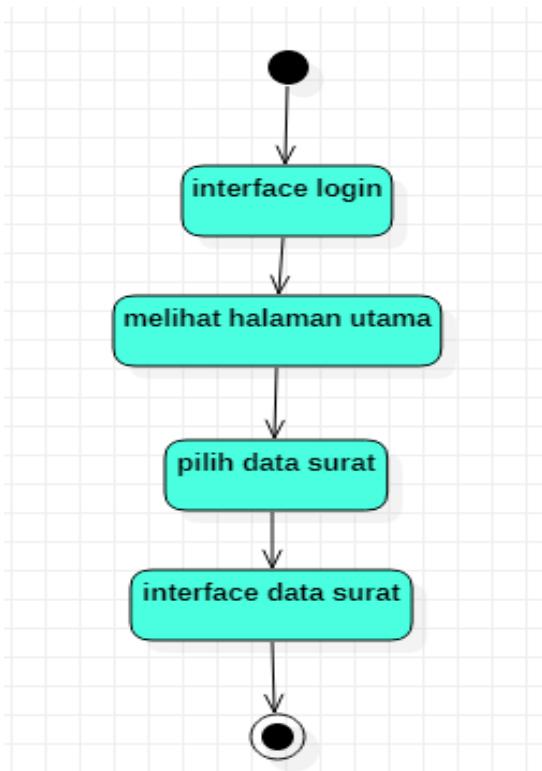
5.2.6.2 Statechart Diagram tambah Paket



Gambar 90.37 Statechart Diagram tambah Paket

Statechart diagram login ini menjelaskan proses Tambah Paket. Petugas mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian petugas memasukkan *username* dan *password* pada *interface login*. Jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data paket dan klik tambah paket. Nanti akan ditampilkan *form* data paket di mana petugas dapat menambahkan paket baru.

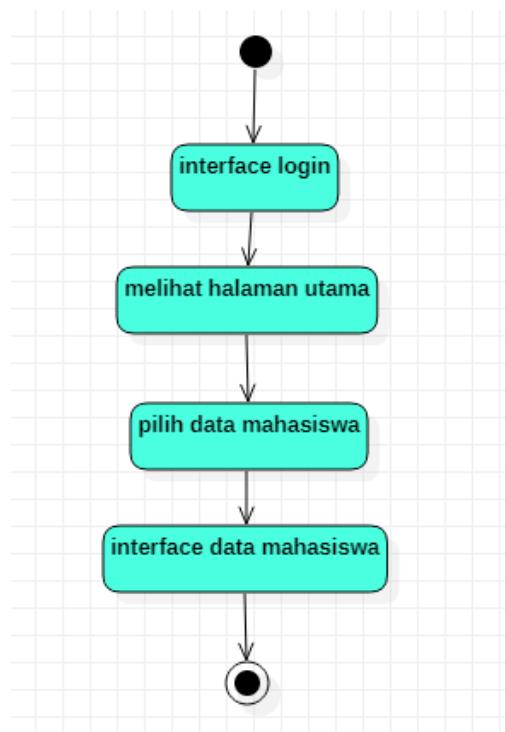
5.2.6.3 Statechart Diagram tambah Surat



Gambar 91.38 Statechart Diagram tambah Surat

Statechart diagram login ini menjelaskan proses Tambah Paket. Petugas mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian petugas memasukkan *username* dan *password* pada *interface login*. Jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu surat buku dan klik tambah surat. Nanti akan ditampilkan *form data surat* di mana petugas dapat menambahkan surat baru.

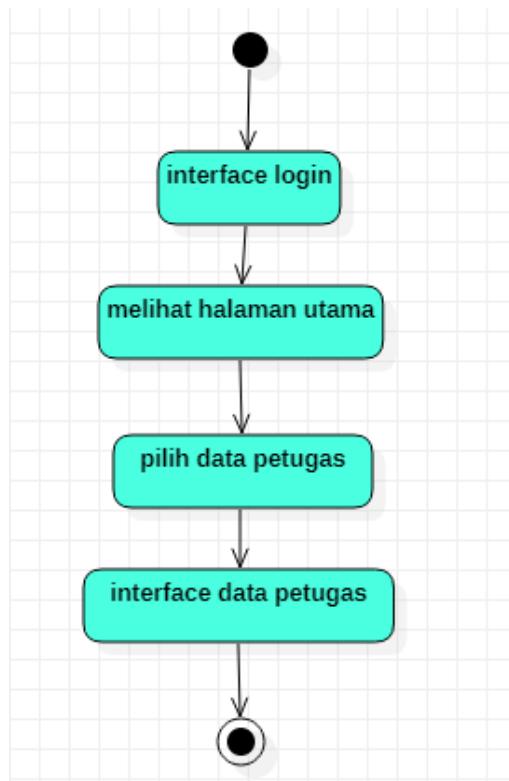
5.2.6.4 Statechart Diagram tambah Penerima



Gambar 92.39 Statechart Diagram tambah Penerima

Statechart diagram login ini menjelaskan proses Tambah Mahasiswa. Petugas mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian petugas memasukkan *username* dan *password* pada *interface login*. Jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data mahasiswa dan klik tambah mahasiswa. Nanti akan ditampilkan *form* data mahasiswa di mana petugas dapat menambahkan mahasiswa baru.

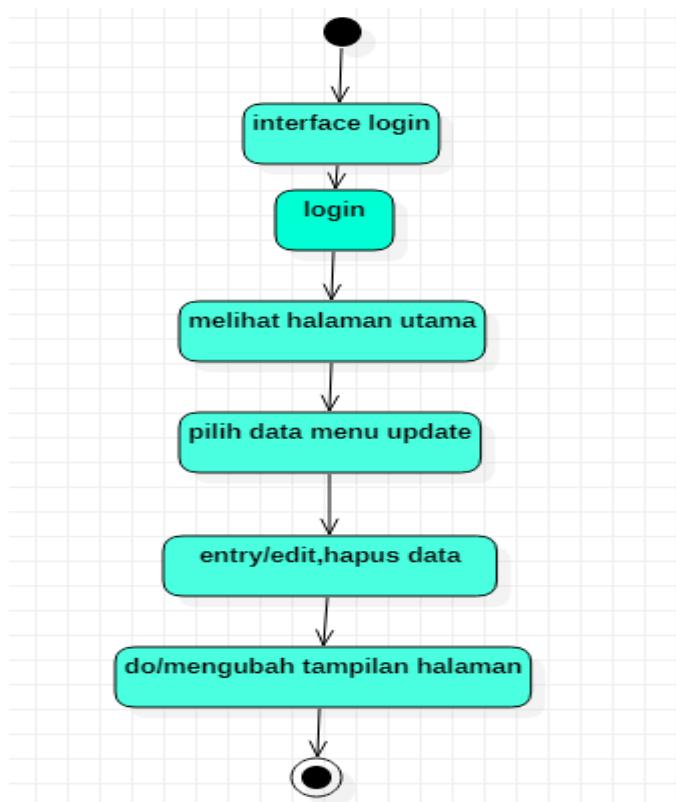
5.2.6.5 Statechart Diagram tambah Petugas



Gambar 93.40 Statechart Diagram tambah Petugas

Statechart diagram login ini menjelaskan proses Tambah Petugas. Admin mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian petugas memasukkan *username* dan *password* pada *interface login*. Jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data petugas dan klik tambah petugas. Nanti akan ditampilkan *form* data petugas di mana admin dapat menambahkan petugas baru.

5.2.6.6 Statechart Diagram Update Data

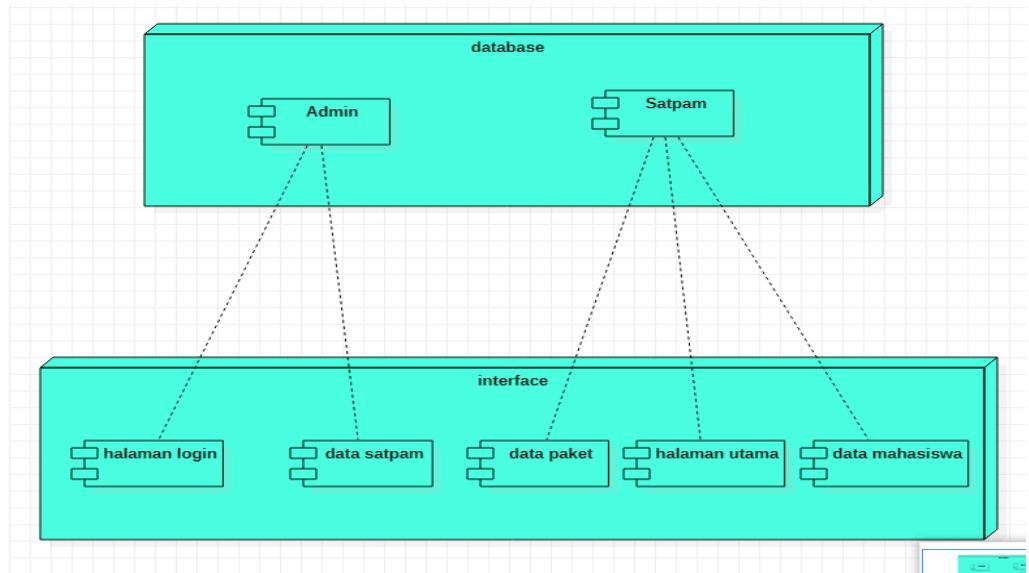


Gambar 94.41 Statechart Diagram Update Data

Statechart diagram login ini menjelaskan proses update data. Admin mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian admin memasukkan *username* dan *password* pada *interface login*. Jika *login* tidak *valid* maka *login* gagal dan jika *valid* maka sistem akan menampilkan halaman utama. Lalu pilih menu data user dan klik tambah petugas. Nanti akan ditampilkan *form* data petugas di mana admin dapat menambahkan petugas baru.

5.2.7 Component Diagram

Component diagram digunakan untuk menunjukkan organisasi dan ketergantungan di antara kumpulan komponen dalam sebuah sistem



Gambar 95.42 Component Diagram

5.2.8 Deployment Diagram

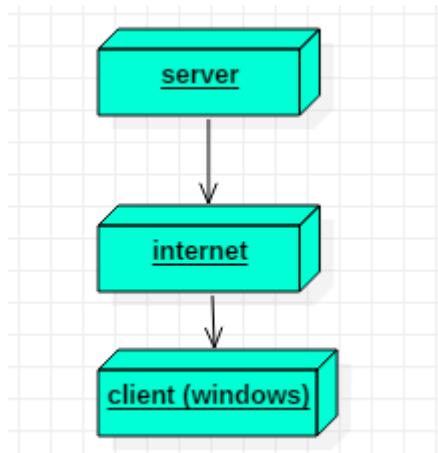
Deployment diagram menunjukkan tata letak sebuah sistem secara fisik, menampilkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware* yang digunakan untuk mengimplementasikan sebuah sistem dan keterhubungan antara komponen-komponen *hardware* tersebut. *Deployment diagram* dapat digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur fisik sebuah sistem. Berikut *deployment diagram* yang menggambarkan susunan fisik dan perangkat lunak dan sistem.

5.2.8.1 Deployment Diagram Software



Gambar 96.43 Deployment Diagram Software

5.2.8.2 Deployment Diagram Hardware

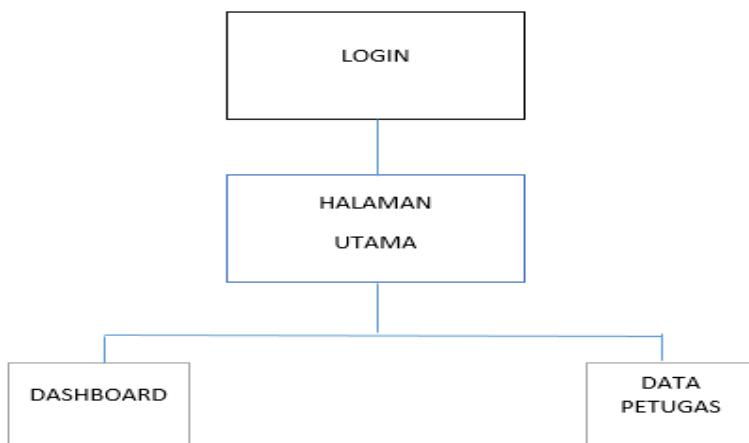


Gambar 97.44 Deployment Diagram Hardware

5.2.9 Struktur Menu

Berikut struktur menu sistem informasi peminjaman dan pengembalian buku di perpustakaan berbasis web :

5.2.9.1 Struktur Menu pada Admin



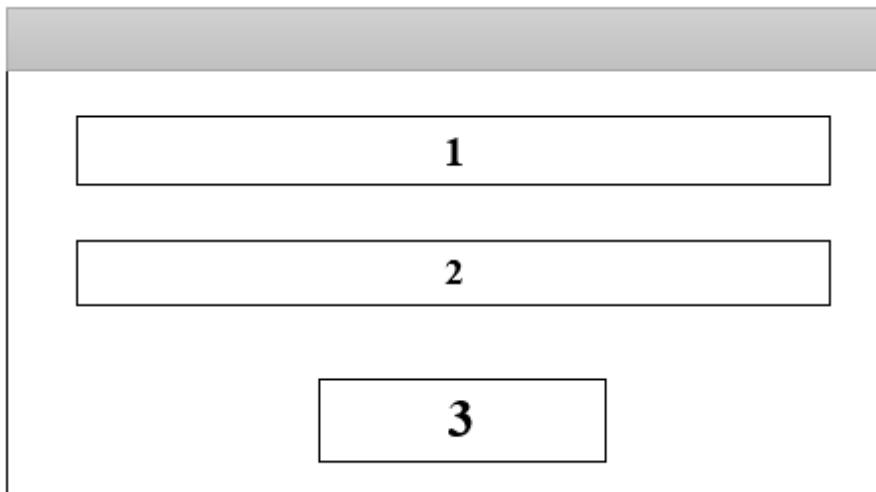
Gambar 5.45 Struktur Menu pada Admin

Fungsi-fungsi yang di rancang pada tahap ini bertujuan untuk memudahkan pengoprasian program tersebut. Admin melakukan login untuk masuk ke halaman utama. Halaman utama hanya menampilkan menu dashboard dan data petugas.

5.2.9.2 Struktur Menu pada Petugas

Pada perancangan aplikasi website ini kami mempunyai beberapa antar muka antara lain :

1. Tampilan Login



Gambar 5.46 Tampilan Login

Pada tahap *login*, petugas memasukkan *username* dan *password* di form *login*. Lalu klik *Login Now*, maka proses *login* akan di validasi oleh sistem.

Keterangan :

1. *Username*
2. *Password*
3. *Login Now*

Algoritma

If login untuk petugas

Masukkan username dan password petugas

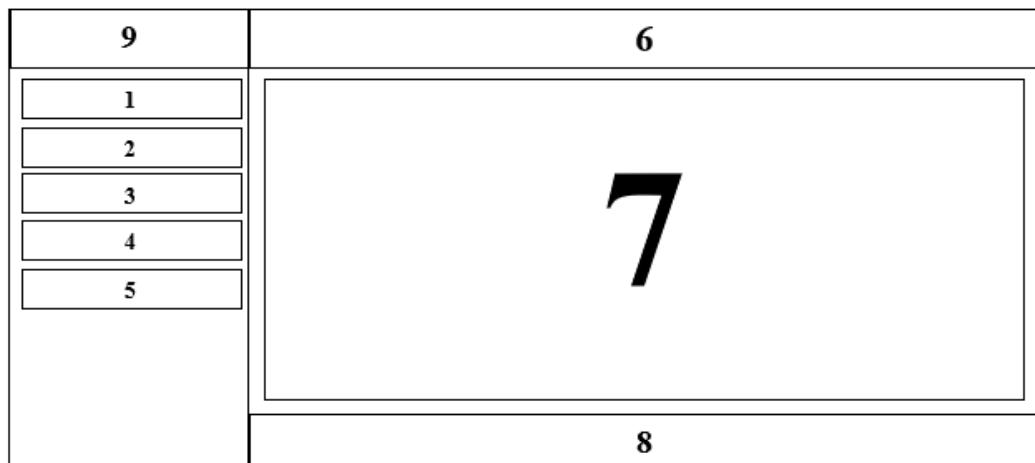
Else if login untuk admin

Masukkan username dan pass word admin

End if

End

1. Tampilan Halaman Data Barang



Gambar 5.45 Tampilan Halaman Utama pada Data Barang

Pada tahap ini, petugas dapat memilih menu pada *side* menu.

Keterangan :

1. *Dashboard*
2. Data paket
3. Data surat
4. Data tamu
5. *History*
6. *Navigation Top*
7. *Content/Main Page*
8. *Footer*
9. *Header*

Algoritma

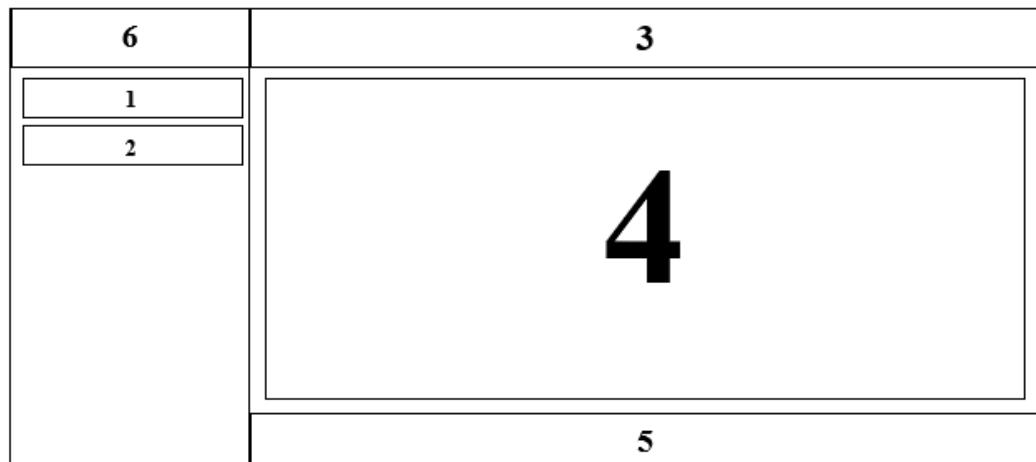
If login petugas berhasil

 Menampilkan halaman utama petugas

End if

End

2. Tampilan Halaman Utama pada Petugas



Gambar 5.46 Tampilan Halaman Utama pada petugas

Pada tahap ini, petugas dapat memilih menu pada *side menu*.

Keterangan :

1. *Dashboard*
2. Data Devinisi
3. *Navigation Top*
4. *Content/Main Page*
5. *Footer*
6. *Header*

Algoritma

If login admin berhasil

 Menampilkan halaman utama Petugas

End if

End

3. Tampilan Paket

The screenshot shows a form with a title 'Tampilan Paket'. Below the title are seven input fields arranged in three rows. Row 1 contains three fields labeled 1, 2, and 3. Row 2 contains two fields labeled 4 and 5. Row 3 contains one large field labeled 6. A small field labeled 7 is located below the large field 6.

Gambar 5.47 Tampilan Paket

Pada tahap paket, petugas memasukkan nama, jenis kantor, telepon, tanggal edit dan hapus di formpaket. Lalu klik *Submit*, maka proses paket akan disimpan di *database*.

Keterangan :

1. Nama
2. Jenis kantor
3. Telepon
4. Tanggal
5. Edit
6. Hapus
7. *Submit*

Algoritma

If pilih menu data anggota

 Menampilkan data anggota

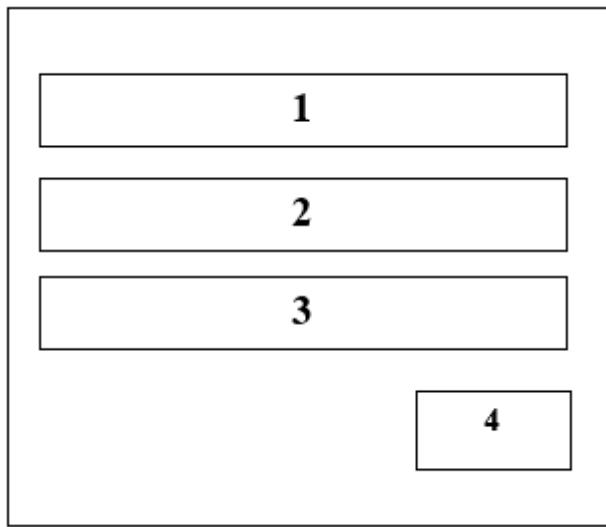
Else if pilih tambah anggota

 Masukkan nama, jenis kantor, telepon ,tanggal, edit, hapus

End if

End

4. Tampilan Data Kategori



Gambar 5.48 Tampilan Data Kategori

Pada tahap *new*, petugas memasukkan nama anggota, judul buku dan tanggal kembali di form *new*. Lalu klik *Submit*, maka proses *input* data master akan disimpan di *database*.

Keterangan :

1. Jenis Kantor
2. Nama Pengirim
3. Tanggal
4. *Submit*

Algoritma

If pilih data peminjaman

 Menampilkan data peminjaman

Else if pilih input peminjaman

 Masukkan jenis kantor, nama pengirim, tanggal

End if

BAB VI

IMPLEMENTASI

6.1 Implementasi

Implementasi adalah penerapan fungsi dari sebuah aplikasi yang sebelumnya telah dirancang sedemikian rupa demi kebutuhan suatu proses.

6.1.1 Lingkungan Implementasi

Untuk memaksimalkan kinerja dari pembuatan *website* itu sendiri, berikut ini adalah kriteria implementasi perangkat lunak dan perangkat keras yang dibutuhkan saat pembuatan *website*.

1. Perangkat Lunak

Table 13.1 Spesifikasi Perangkat Lunak

No	Jenis	Keterangan
1	Sistem Operasi	: Windows 10
2	Bahasa Pemrograman	: PHP (<i>Framework Bootstrap</i>)
3	Database	: MySQL
4	Perangkat Lunak	: Sublime Text, Google Chrome (<i>Browser</i>)

2. Perangkat Keras

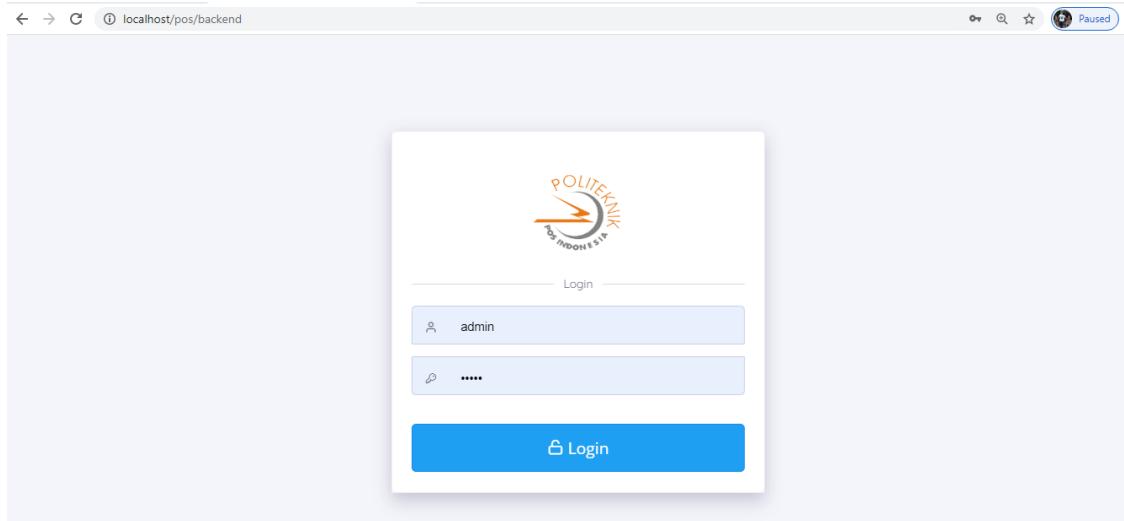
Table 14.2 Spesifikasi Perangkat Keras

No	Jenis	Keterangan
1	<i>Processor</i>	: Intel® Core™ i3-6100U
2	<i>RAM</i>	: 4 GB
3	<i>Hardisk Esternal</i>	: 1 TB
4	<i>HDD</i>	: 500 GB
5	<i>VGA</i>	: NVIDIA GeForce 920M

6.2 Pembahasan Hasil Implementasi

Implementasi dari *website* ini, dijelaskan secara umum dari *mainpage* setiap menu, antara lain : *Login*, *Dashboard*, Data Tamu, Data Paket, Data Surat, *History Penglolaan*, *Form Tambah Buku Tamu*, *Form Tambah Paket*, *Form Input Surat*, Koneksi. Berikut adalah tampilan antar muka:

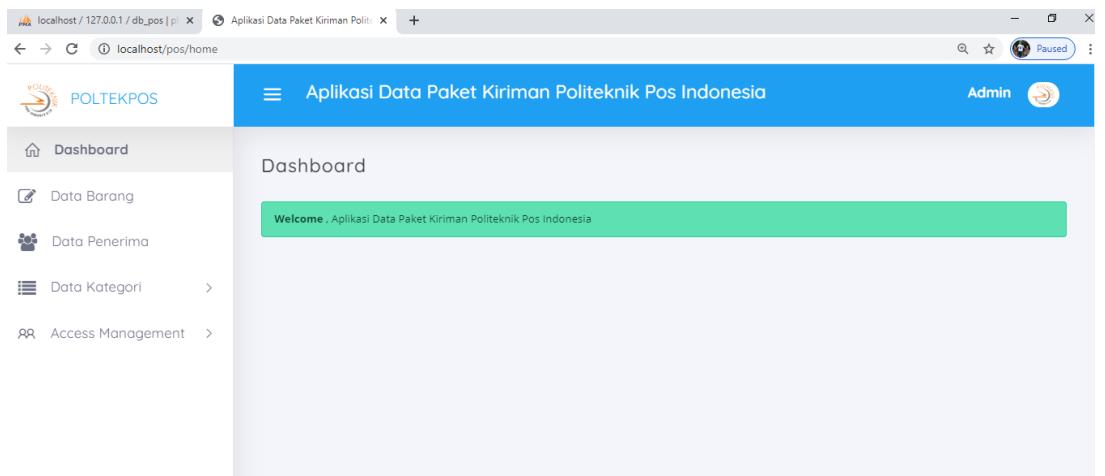
6.2.1 Halaman Login Admin



Gambar 98.1 Halaman Login Admin

Login memasukkan *username* dan *password*. Klik *Login Now*. Jika berhasil maka akan tampil ke Halaman Utama.

6.2.2 Halaman Dasboard pada Halaman Admin



Gambar 99.2 Halaman Dasboard

Dasboard adalah tampilan awal pada aplikasi data paket. Dimana terdapat Data Barang, Data Penerima, Data Kategori dan Access Managemen.

6.2.3 Halaman Data Barang

No	Nama Pengirim	Kategori Paket	Nama Penerima	Email Penerima	Status Barang	Petugas Penerima	Tanggal
1.	jono	Barang	andi	andi@gmail.com	Menunggu	Widia Wijaya	Jun 2021
2.	feny	Surat	rani	rani@gmail.com	Menunggu	Widia Wijaya	Karr 2021
3.	Kurnia Sandi	Barang	Riki Karnovi	rikikarnovi@gmail.com	Menunggu	Pedi Sumantri	Karr 2021
4.	Riky Martin	Barang	maria goretti	maria@gmail.com	Menunggu	Anwar	Karr 2021
5.	Cio	Barang	lidwina	lidwinagulo18@gmail.com	Menunggu	Widia Wijaya	Karr

Gambar 100.3 Halaman Data Barang

Data Barang adalah list data yang ada di post satpam yang dilakukan oleh petugas pos satpam. Data tersebut dapat mengkonfirmasi pengiriman paket. Setelah itu jika ingin menambah data lagi klik Tambahkan Data. Status email akan menunggu jika hosting pada aplikasi tidak dijalankan. Dan setatus akan berubah apabila hosting pada aplikasi jalan. Status tersebut adalah dimana si penerima paket atau barang sudah menerima notifikasi melalui email bahwa barang yang dikirim sudah sampai.

6.2.4 Halaman From Data Barang

The screenshot shows a web-based application interface for managing package delivery data. On the left, there's a sidebar with links for Dashboard, Data Barang, Data Penerima, Data Kategori, and Access Management. The main area has a title 'Data Barang' and a table titled 'List Data'. A modal window titled 'Form Data Barang' is open in the center. This modal contains fields for 'Nama Pengirim *', 'Telepon Pengirim *', 'Penerima *', 'Kategori Paket *', and 'Petugas *', each with dropdown menus. Below these fields is a text area labeled 'Keterangan'. At the bottom of the modal are two buttons: 'Close' and 'Save changes'. In the background, the table 'List Data' shows five rows of package information, each with columns for Status Barang, Petugas Penerima, Tanggal, and an 'Activate Windows' link.

No	Nama Pengirim	Kategori Paket	Penerima	Alamat	Tanggal	Activate Windows
1.	John Doe	-Pilih-	-Pilih-	-Pilih-	2021-01-15	Go to Settings to activate Windows.
2.	Female	-Pilih-	-Pilih-	-Pilih-	2021-01-15	Widia Wijaya
3.	Kurnia Sandi	Barang	Riki Karnovi	rikikarnovi@gmail.com	2021-01-15	Widia Wijaya
4.	Riky Martin	Barang	maria goretti	maria@gmail.com	2021-01-15	Pedi Sumantri
5.	Cio	Barang	lidwina	lidwinagulo18@gmail.com	2021-01-15	Anwar

Gambar 101.4 Halaman From Data Barang

From data Barang adalah pengisian data-data pengirim dan penerima yang dilakukan oleh petugas pos satpam. Data tersebut dapat mengkonfirmasi pengiriman paket. Setelah itu jika data akan bertambah pada halaman Data Barang. Jika data tidak diisi dengan lengkap maka data tidak dapat disimpan dan data diminta untuk di lengkapi terlebih dahulu.

6.2.5 Halaman Data Penerima

No	Nama	Email	Telepon		
1.	lidwina	lidwinagulo18@gmail.com	082311950685		
2.	Sri Kurnia Ningsih	sri@gmail.com	081325678912		
3.	maria goretti	maria@gmail.com	085276541908		
4.	Riki Karnovi	rikikarnovi@gmail.com	08136548926		
5.	Riko	ingnasrico@gmail.com	083190950251		
6.	rani	rani@gmail.com	08132456789		
7.	andi	andi@gmail.com	08132647898		

Gambar 102.5 Halaman Data Penerima

Data Penerima adalah list data penerima yang dilakukan oleh petugas pos satpam. Data tersebut untuk menambahkan nama-nama data penerima yang nantinya akan di panggil saat pengisian data barang. Jika ingin menambahkan data penerima klik Tambahkan Data.

6.2.6 Halaman Form Data Penerima

Form Data Penerima

Nama *	<input type="text"/>
Kategori *	<input type="text"/> Mahasiswa <input type="text"/> Mahasiswa <input type="text"/> Dosen <input type="text"/> Staff
Email *	<input type="text"/>
Telepon *	<input type="text"/>

Gambar 103.6 Halaman Data Penerima

Data Penerima adalah pendataan data penerima yang dilakukan oleh petugas pos satpam. Data tersebut dapat mengkonfirmasi penambahan data penerima. Setelah itu data yang sudah diisikan dan lengkap akan ditambahkan pada halaman Data Penerima. Sebagai kategori ada pemilihan apakah sebagai mahasiswa, dosen, atau staf.

6.2.7 Halaman Data Kategori Penerima

The screenshot shows a web application interface for managing recipient categories. The top navigation bar includes the logo 'POLTEKPOS', the title 'Aplikasi Data Paket Kiriman Politeknik Pos Indonesia', and a user 'Admin'. The left sidebar menu lists 'Dashboard', 'Data Barang', 'Data Penerima', 'Data Kategori' (which is currently selected), and 'Kategori Paket'. Below the sidebar is a link 'Access Management >'. The main content area is titled 'Kategori Penerima' and contains a table titled 'List Kategori Penerima'. The table has columns 'No' and 'Nama Kategori'. The data is as follows:

No	Nama Kategori	Action	Action
1.	Mahasiswa		
2.	Dosen		
3.	Staff		

Buttons for 'Reload' and 'Tambah Data' are located at the top right of the table area. A search bar labeled 'Search Text' is also present.

Gambar 104.7 Halaman Data Kategori Penerima

Data Kategori adalah list category penerima yang dilakukan oleh petugas pos satpam. Data tersebut dapat mengkonfirmasi list katgori penerima. Setelah itu jika ingin menambahkan category klik Tambahkan Data.

6.2.8 Halaman Data Kategori Paket

The screenshot shows a web application interface for managing package categories. The top navigation bar includes links for 'Dashboard', 'Data Barang', 'Data Penerima', 'Data Kategori' (which is currently selected), and 'Access Management'. The main title is 'Aplikasi Data Paket Kiriman Politeknik Pos Indonesia'. On the right, there are buttons for 'Admin' and a user profile icon. The central area is titled 'Kategori Paket' and contains a table titled 'List Kategori Paket'. The table has three rows, each representing a category: 'Surat' (No. 1), 'Barang' (No. 2), and 'tamu' (No. 3). Each row includes edit and delete icons. A 'Search Text' input field is located at the top right of the table area. Navigation controls like 'Reload' and 'Tambah Data' are also present.

Gambar 105.8 Halaman Data Kategori Paket

Data Kategori Paket adalah list category paket yang dilakukan oleh petugas pos satpam. Data tersebut dapat mengkonfirmasi list katgori paket. Setelah itu jika ingin menambahkan category klik Tambahkan Data.

6.2.9 Halaman User Management

The screenshot shows the User Management section of the application. On the left, there's a sidebar with icons for Dashboard, Data Barang, Data Penerima, Data Kategori, Access Management (which is expanded to show User Management), and a back arrow. The main area has a blue header with the title 'Aplikasi Data Paket Kiriman Politeknik Pos Indonesia' and a user 'Admin'. Below the header is a search bar with 'Pencarian User Management', 'Keyword', 'Role', and a dropdown for 'Semua Data'. There are 'Reload' and 'Tambah Data' buttons. The central part is titled 'List User Management' and contains a table with the following data:

No	User Name	Nama	Role	Email	Update Password	Status	Action
1.	superuser	Superuser	Super User	admin@admin.com	20-03-2019	Aktif	
2.	doron	Doron	Satpam	-		Aktif	
3.	admin	admin	Admin	-		Aktif	
4.	Dadang	Dadang Koswara	Admin	dadang@gmail.com		Aktif	
5.	pedi	Pedi Sumantri	Satpam	pedi@gmail.com		Aktif	
6.	widia	Widia Wijaya	Satpam	widia18@gmail.com		Aktif (inactive)	

Gambar 106.9 Halaman User Management

User Management adalah list *user management* yang dilakukan oleh petugas pos satpam. Data tersebut dapat mengkonfirmasi list *user management*. Setelah itu jika ingin menambahkan data klik Tambahkan Data. Dimana data tersebut adalah data-data petugas yang bertugas untuk mengelola aplikasi pengelolaan data barang yang ada di politeknik pos Indonesia. Dan petugas tersebut apakah masih aktif atau tidak.

6.2.10 Halaman From User Management

The screenshot shows the 'User Management' form in a web application. At the top right, there is a header bar with the title 'Aplikasi Data Paket Kiriman Politeknik Pos Indonesia' and a user status 'Admin'. On the left, a sidebar menu includes 'Dashboard', 'Data Barang', 'Data Penerima', 'Data Kategori', 'Access Management', and 'User Management', with 'User Management' currently selected. The main form area has a title 'Form User Management' and fields for 'Username', 'Nama', 'Nomor HP / Telepon', 'Email', 'Jenis Kelamin' (with a dropdown menu showing '- Pilih -'), 'Role' (with a dropdown menu showing 'Super User'), and 'Status User' (with a dropdown menu showing 'Aktif'). Below these fields is a table titled 'Nama Menu' with columns for View, Create, Edit, Delete, Export, Duplicate, and Unlock. The table lists three items: 'Dashboard' (all permissions checked), 'Data Barang' (View, Create, Edit, Delete checked), and 'Data Penerima' (View, Create, Edit, Delete checked). To the right of the table is a message: 'Activate Windows Go to Settings to activate Windows.' Below the table is another table titled 'Data Penerima' with various permission checkboxes. At the bottom of the form are two buttons: a red 'Kembali' button and a green 'Simpan' button.

Gambar 107.10 Halaman From User Management

From User Management adalah list *from user management* yang dilakukan oleh petugas pos satpam. Data tersebut dapat mengkonfirmasi list *from user management*. Setelah itu akan ada pilihan untuk dia sebagai admin atau sebagai super admin dan apabila dia sebagai admin atau super admin aka nada petunjuk bahwa petugas itu hanya bias melakukna apa saja pada aplikasi tersebut. Dimana data tersebut adalah data-data petugas yang bertugas untuk mengelola aplikasi pengelolaan data barang yang ada di politeknik pos Indonesia dan data tersebut akan di tambahkan ke halaman *user management*.

6.3 Pengujian dan Hasil Pengujian

Setelah dilakukan perancangan dan implementasi ke dalam program, selanjutnya akan dilakukan simulasi untuk pengujian aplikasi dengan tujuan memperoleh perbandingan antara hasil perancangan dengan hasil yang dicapai.

6.3.1 Identifikasi dan Rencana Pengujian

Table 15.3 Identifikasi dan Rencana Pengujian

Nomo r	Kelas Uji	Butir Uji	Identifika si	Tingkat Pengujia n	Jenis Pengujia n	Jadwal
UC- 01	Akses Websit	Akses Buka	VD_01	Pengujia n	Blackbox	13 Desember20
UC- 02	e	Website		Intrenshi p I		19
UC- 04	Input Data	Input Data Penerima	ID_01	Pengujia n	Blackbox	13 Desember
				Intrenshi p I		2019
UC- 03		Input Data Kategori	ID_02	Pengujia n	Blackbox	13 Desember
				Intrenshi p I		2019
UC- 05		Input Access Manageme nt	ID_03	Pengujia n	Blackbox	13 Desember
				Intrenshi p I		2019

Keterangan Identifikasi Pengujian :

1. VD : Validasi
2. ID : Input Data
3. HD : Hapus Data
4. TD : Tampil Data

6.3.2 Hasil Pengujian

6.3.2.1 Kelas Uji Validasi

Table 16.4 Butir Uji Buka Website

Identifikasi	VD_01
Deskripsi	Pengujian dengan membuka halaman per halaman website
Prosedur Pengujian	Hanya membuka halaman per halaman
Masukan	-
Keluaran yang diharapkan	<i>Load</i> halaman tidak memakan waktu lama
Kriteria Evaluasi Hasil	Setiap halaman yang terdapat pada website
Hasil yang Didapat	<i>Load</i> halaman tidak terlalu lama
Simpulan	Diterima

6.3.2.2 Kelas Uji Input Data Penerima

Table 17.5 Butir Uji Data Penerima

Identifikasi	ID_01
Deskripsi	Penginputan data penerima
Prosedur Pengujian	Isi <i>form</i> dengan data yang benar
Masukan	Jenis kantor, nama pengirim, tanggal
Keluaran yang diharapkan	Menampilkan hasil <i>inputan</i>

Kriteria Evaluasi Hasil	Informasi data penrima yang <i>diinput</i> ditampilkan
Hasil yang Didapat	Sesuai dengan keluaran yang diharapkan
Simpulan	Diterima

6.3.2.3 Kelas Uji Input Data Master Devisi

Table 18.6 Butir Uji Input Data Master Devisi

Identifikasi	ID_02
Deskripsi	Penginputan data devisi
Prosedur Pengujian	Isi <i>form</i> dengan data yang benar
Masukan	Nama Devisi
Keluaran yang diharapkan	Menampilkan hasil <i>inputan</i>
Kriteria Evaluasi Hasil	Informasi data buku yang <i>diinput</i> ditampilkan
Hasil yang Didapat	Sesuai dengan keluaran yang diharapkan
Simpulan	Diterima

6.3.2.4 Kelas Uji Input Data Rekap Paket

Table 19.7 Butir Uji Input Data Rekap Paket

Identifikasi	ID_03
Deskripsi	Penginputan data Rekap
Prosedur Pengujian	Isi <i>form</i> dengan data yang benar
Masukan	Jenis Kantor, Jumlah
Keluaran yang diharapkan	Menampilkan hasil <i>inputan</i>
Kriteria Evaluasi Hasil	Informasi data asi yang di <input type="text"/> input ditampilkan
Hasil yang Didapat	Sesuai dengan keluaran yang diharapkan
Simpulan	Diterima

6.3.2.5 Kelas Uji Input Data Rekap Per Devisi

Table 20.8 Butir Uji Input Data Rekap Per Devisi

ID_04	
Identifikasi	
Deskripsi	Penginputan data rekap per devisi
Prosedur Pengujian	Isi <i>form</i> dengan data yang benar
Masukan	Nama <i>Devisi</i>
Keluaran yang diharapkan	Menampilkan hasil <i>inputan</i>
Kriteria Evaluasi Hasil	Informasi data <i>user</i> yang <i>diinput</i> ditampilkan
Hasil yang Didapat	Sesuai dengan keluaran yang diharapkan
Simpulan	Diterima

6.3.2.6 Kelas Uji Data Laporan

Table 21.9 Butir Uji Data Laporan

ID_04	
Identifikasi	
Deskripsi	Data Laporan
Prosedur Pengujian	Lihat Master Data,Data Surat
Lihat	Nama Pengirim, jenis kantor,tanggal, nama penerima, Hp_Penerima, keterangan

Keluaran yang diharapkan	Menampilkan hasil
Kriteria Evaluasi Hasil	Masuk ke halaman data laporan
Hasil yang Didapat	Sesuai dengan keluaran yang diharapkan
Simpulan	Diterima

6.3.2.7 Kelas Uji Data Historis

Table 22.9 Butir Uji Data Historis

Identifikasi	ID_05
Deskripsi	Data Historis
Prosedur Pengujian	Buat data Priode
Masukan	Lihat laporan
Keluaran yang diharapkan	Menampilkan laporan
Kriteria Evaluasi Hasil	Masuk ke halaman historis laporan
Hasil yang Didapat	Sesuai dengan keluaran yang diharapkan
Simpulan	Diterima

