BAB IV

HASIL DAN PEMBAHASAN

4.1 Perancangan Sistem

Perancangan adalah gambaran secara terperinci dari metode-metode perancangan dalam pembangunan suatu perangkat lunak. Perancangan sistem dibuat untuk menggambarkan secara jelas proses-proses yang diinginkan oleh pemakai (user). Perancangan sistem ini meliputi usecase diagram, skenario diagram, activity diagram, diagram sequence, collaboration diagram dan class diagram.

4.1.1 Tujuan Perancangan Sistem

Tujuan perancangan Sistem Informasi Pemasaran dan Pemesanan CV. AMADDA Berbasis Web ini adalah untuk memberikan gambaran yang jelas mengenai kebutuhan *user*. Adapun tujuan dari perancangan ini adalah :

1. Membangun aplikasi pemasaran dan pemesanan secara online dimana user atau calon konsumen dapat berkomunikasi dan melakukan proses pemesanan secara online.
2. Memperkenalkan produk yang ditawarkan yaitu Apartemen Easton Park ke masyarakat luas.
3. Membangun sistem informasi memanfaatkan teknologi internet, dengan menggunakan sistem unggah dan unduh yang dapat memfasilitasi manager pemasaran untuk memasarkan produknya.

4. Memberikan media pemesanan secara online untuk calon konsumen yang tertarik dengan produk yang ditawarkan CV. AMADDA.
5. Membantu pihak perusahaan dalam mengolah data pemesanan dari yang sebelumnya masih secara manual menjadi sistem yang terkomputerisasi.

4.1.2 Gambaran Umum Sistem yang Diusulkan

Perancangan sistem yang diusulkan berbeda dengan sistem yang sedang berjalan. Perubahannya ialah pengolahan data yang masih manual diubah menjadi terkomputerisasi. Pemasaran secara manualpun diganti menjadi berbasis website. Pada sistem yang diusulkan diharapkan dapat memberikan hasil sebagai berikut :

1. Dapat mempromosikan produk melalui website yang dirancang oleh penulis
2. Dapat membantu user untuk melakukan pemesanan secara online
3. Dapat meminimalisir kesalahan-kesalahan dalam mengolah data pemasaran dan pemesanan.
4. Dapat mencetak laporan pemesanan dan laporan unit tersedia.

4.1.3 Perancangan Prosedur yang Diusulkan

Perancangan prosedur merupakan gambaran kepada pengguna tentang sistem yang baru atau yang diusulkan oleh penulis. Pada perancangan prosedur yang diusulkan dijelaskan mengenai Use Case Diagram, Activity Diagram, Sequence Diagram, collaboration diagram, dan class diagram.

Berikut perancangan prosedur yang diusulkan :

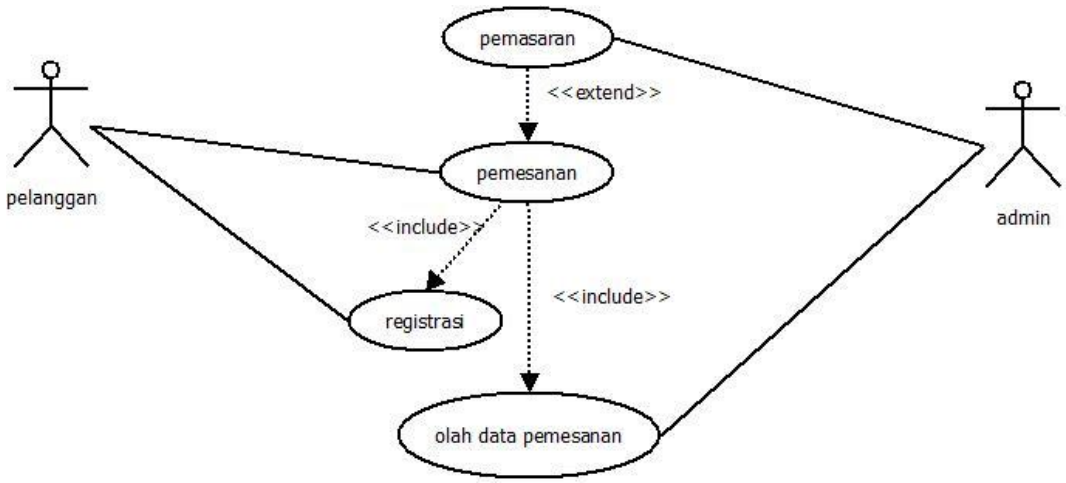
1. Pelanggan dapat mengakses menu-menu dari website yang telah disediakan. Menu yang disediakan tersebut ialah home, produk, fasilitas, lokasi, info & event, kontak, dan login.
2. Pelanggan dapat melihat unit apartemen yang tersedia tetapi untuk melakukan pemesanan pelanggan harus login, jika pelanggan tidak memiliki account maka pelanggan dapat melakukan registrasi.
3. Pelanggan dapat mengetahui status bookingnya.
4. Pelanggan dapat mencetak bukti pembayaran booking fee apabila pembayaran telah dilakukan.
5. Admin dapat menggunakan fasilitas yang disediakan sistem setelah melakukan login.
6. Admin dapat menginput data unit dan mengunggah (*upload*) nya ke menu produk.
7. Admin dapat menginput artikel ke menu info & event untuk memberikan berita-berita atau informasi terbaru.
8. Admin mengatur data keseluruhan user.
9. Admin dan direktur dapat mencetak Laporan unit tersedia dan Laporan Pemesanan.

4.1.3.1 Use Case Diagram

Use Case diagram (diagram use case) adalah deskripsi dari urutan aktifitas-aktifitas yang ditampilkan sistem dengan hasil yang terukur bagi suatu aktor. Use Case digambarkan dengan elips tegas yang berisi namanya. Use case

berisi tentang apa yang dilakukan oleh sistem atau apa yang terjadi pada sistem, bukan bagaimana sistem melakukan.

Berikut adalah usecase diagram sistem informasi pemasaran dan pemesanan yang diusulkan :



Gambar 4.1 Use case Sistem Informasi yang Diusulkan

4.1.3.2 Skenario Diagram

Skenario Use Case bertujuan untuk mendeskripsikan atau menjelaskan diagram use case. Berikut adalah skenario use case berdasarkan dari gambar 4.1.

1. Skenario Use Case Pemasaran

   No Use Case : 1

   Nama Use Case : Pemasaran

   Aktor : admin

   Deskripsi : Melakukan Pemasaran

Tabel 4.1 *Skenario Use Case* Pemasaran

Admin	Sistem
1. Login menurut hak akses	
	2. Menampilkan halaman utama admin
3. Memilih menu unit yang akan di pasarkan atau memilih menu edit info	
	4. Menyimpan data unit ke database
	5. Menampilkan data unit yang akan dipasarkan di halaman website dan halaman info.

2. Skenario Use Case Pemesanan

Aktor : pelanggan

Deskripsi : melakukan pemesanan unit

Tabel 4.2 Skenario Use Case Pemesanan

Pelanggan	Sistem
1. Mengakses menu produk	
	2. menampilkan unit apartemen Easton Park yang tersedia beserta harga dan fasilitasnya.
3. Memilih unit Apartemen Easton Park yang akan di booking	
	4. Menyimpan data pemesanan pelanggan
	5.Mengirim invoice pembayaran booking fee ke email pelanggan
6. Membayar booking fee dan melengkapi syarat-syarat pemesanan	
	7.Mengubah status pembayaran unit
	8.Mengirim bukti pembayaran booking fee
9.Menerima bukti pembayaran booking fee	

3. Nama Use Case : Registrasi

   Aktor : Pengunjung

   Deskripsi : Melakukan registrasi

Tabel 4.3 Skenario Use Case Registrasi

User	Sistem
1. Memilih menu registrasi	
2.Mengisi form registrasi	
	3. Menyimpan data form registrasi

4. Nama Use Case : Olah Data Pemesanan

   Aktor : Admin

   Deskripsi : Mengolah data pemesanan

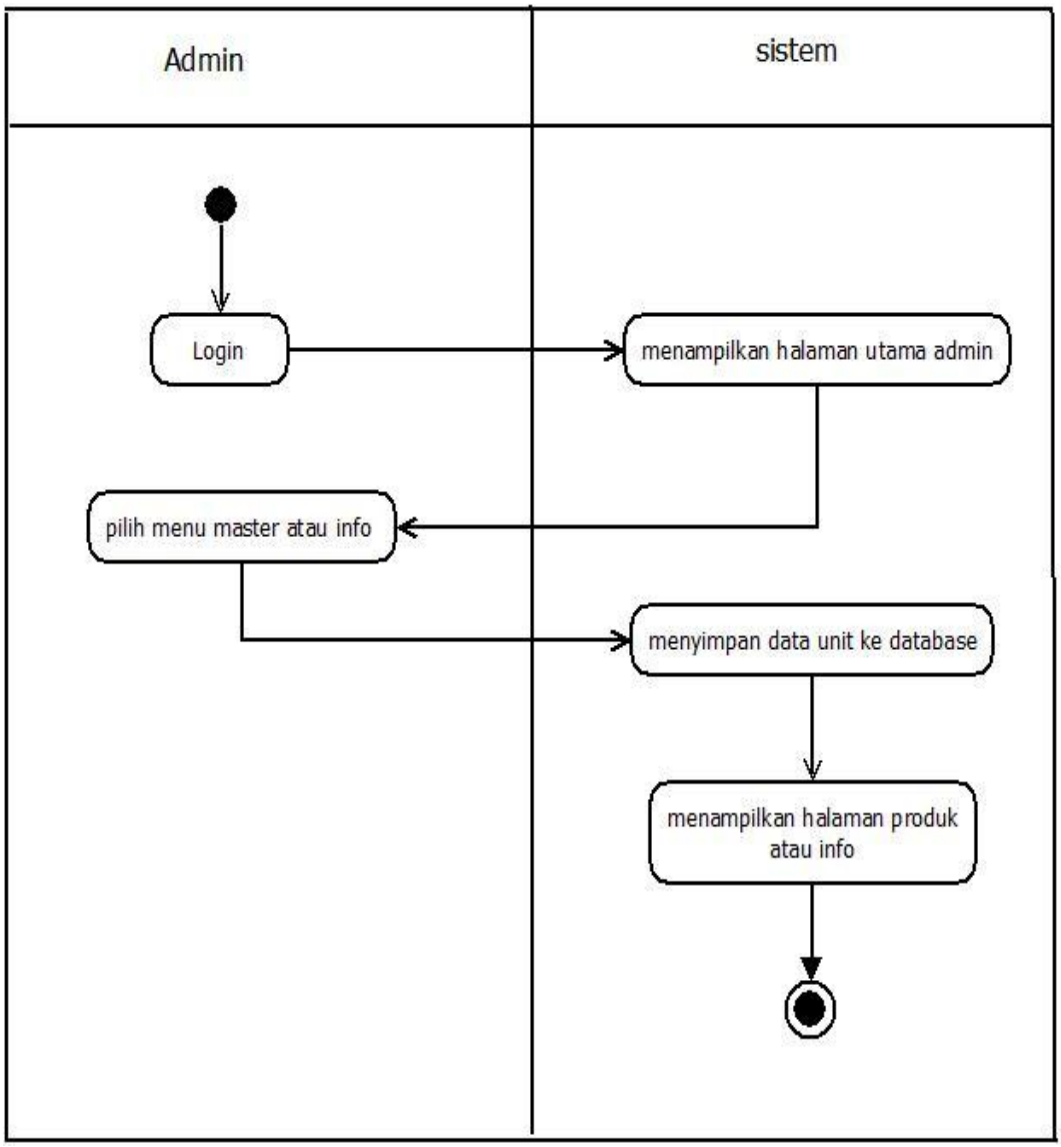
Tabel 4.4 Skenario Use Case Olah Data Pemesanan

Admin	Sistem
1. login sesuai hak akses	
	2. menampilkan menu yang di miliki sesuai hak akses, yaitu master data, users, invoice, laporan
3. memilih menu yang akan di kelola	
	4. menyimpan update data pemesanan

4.1.3.3. Activity Diagram

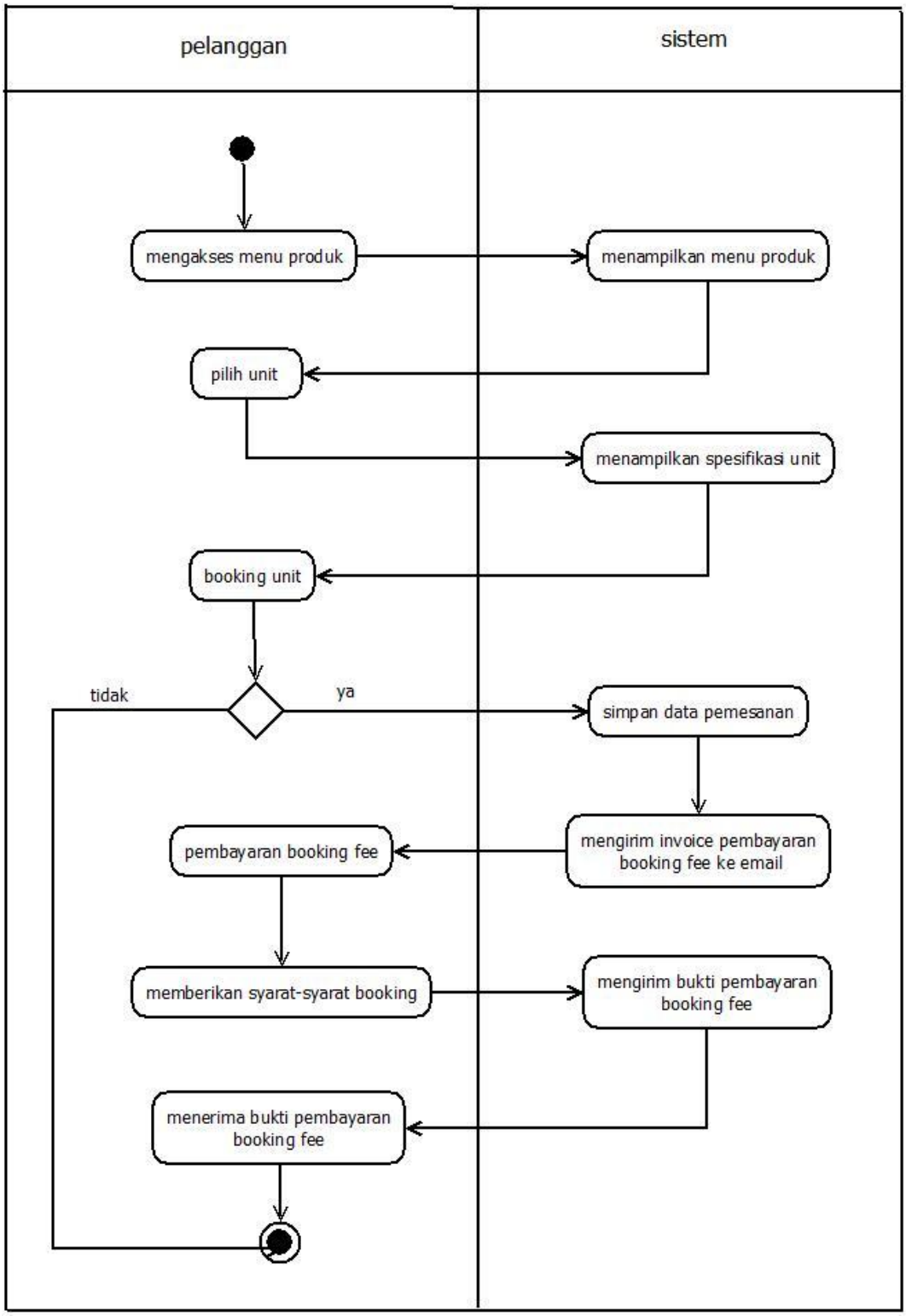
Diagram aktivitas atau *Activity Diagram* menggambarkan aliran fungsional sistem. Berikut ini adalah diagram aktivitas yang diusulkan penulis berdasarkan *use case diagram* yang dibuat sebelumnya.

1. Activity Diagram Pemasaran



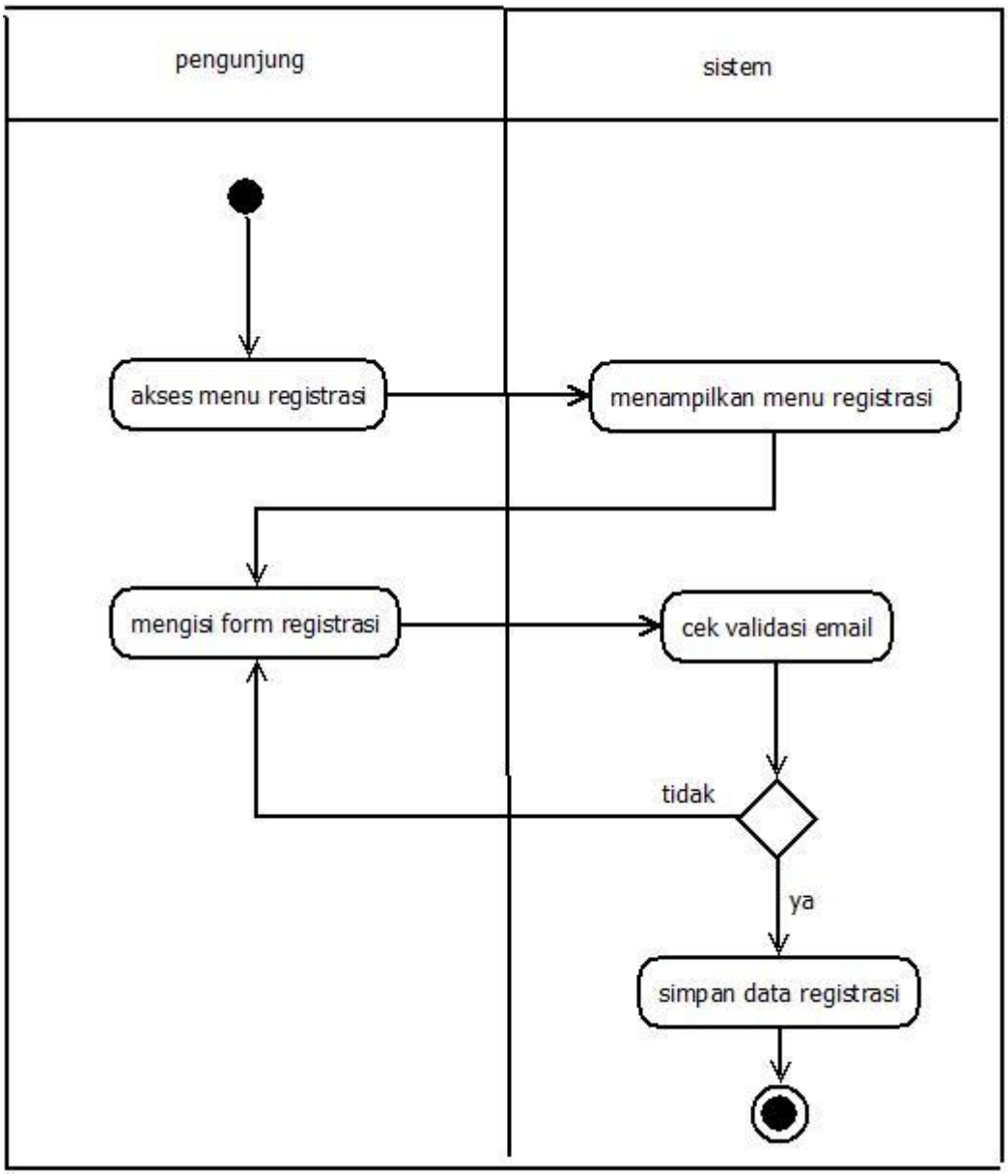
Gambar 4.2 *Activity Diagram Pemasaran*

2. Activity Diagram Pemesanan



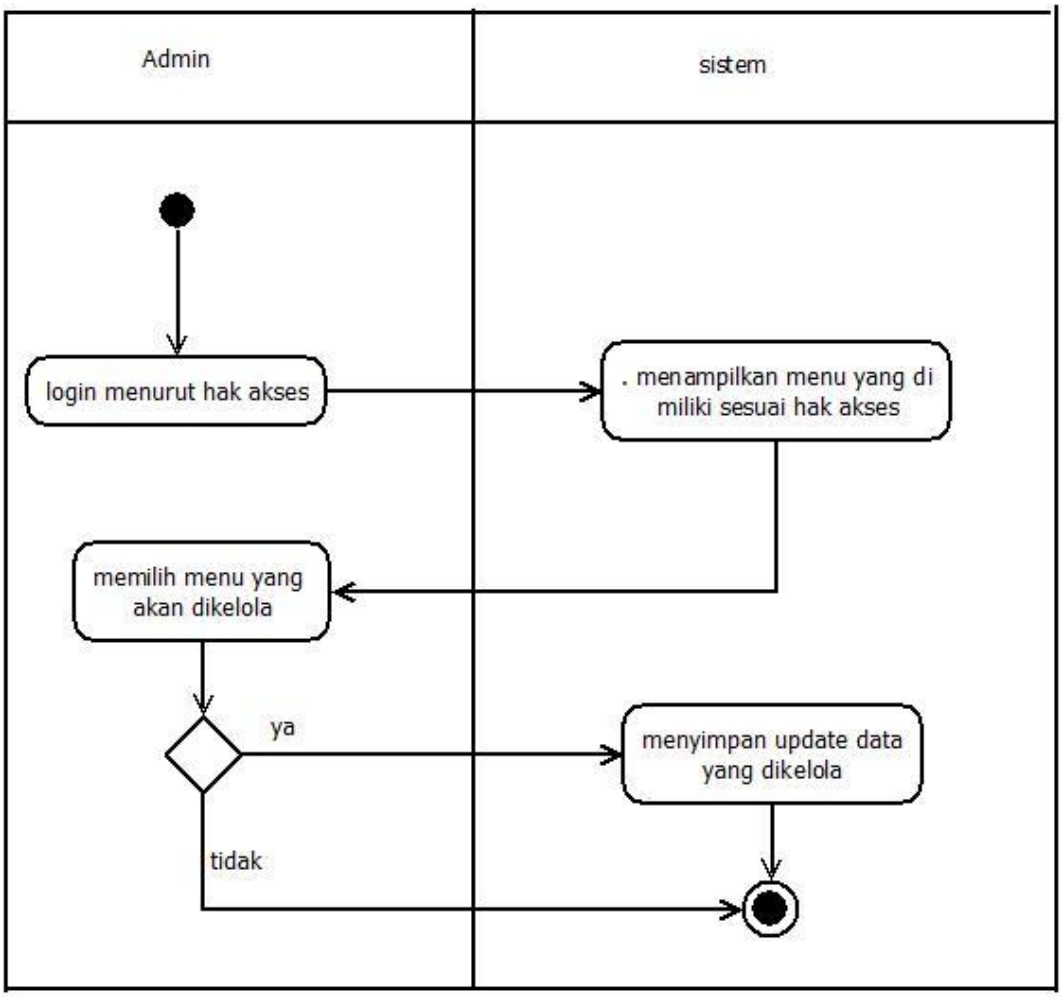
Gambar 4.3 *Activity Diagram Pemesanan*

3. Activity Diagram Registrasi



Gambar 4.4 *Activity Diagram Registrasi*

4. Activity Diagram Olah Data Pemesanan



Gambar 4.5 *Activity Diagram Olah Data Pemesanan*

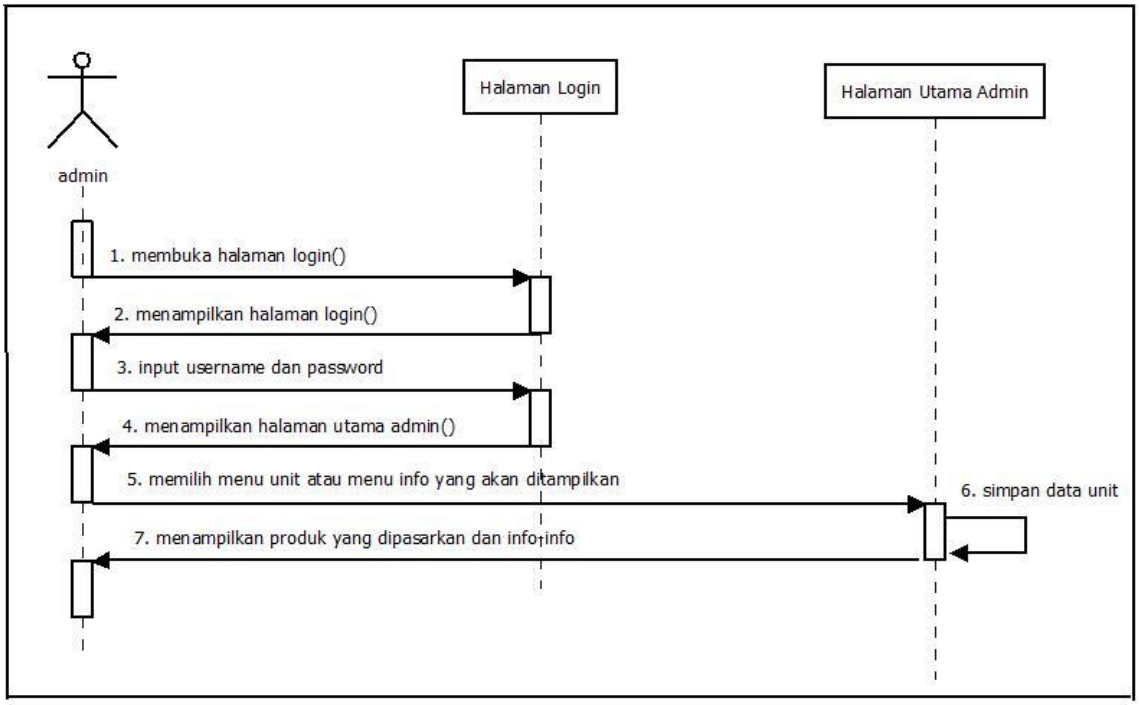
4.1.3.4. Sequence Diagram

Sequence diagram mendeskripsikan interaksi antara objek di dalam dan di sekitar sistem (termasuk user, tampilan dan sebagainya) yang disusun berdasarkan urutan waktu.

Pada sequence diagram ini menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan pada sistem sebagai respon dari sebuah event untuk menghasilkan output tertentu.

Banyaknya diagram sequence yang didefinisikan, bergantung pada proses use case yang telah dibuat. Berikut adalah sequence diagramnya :

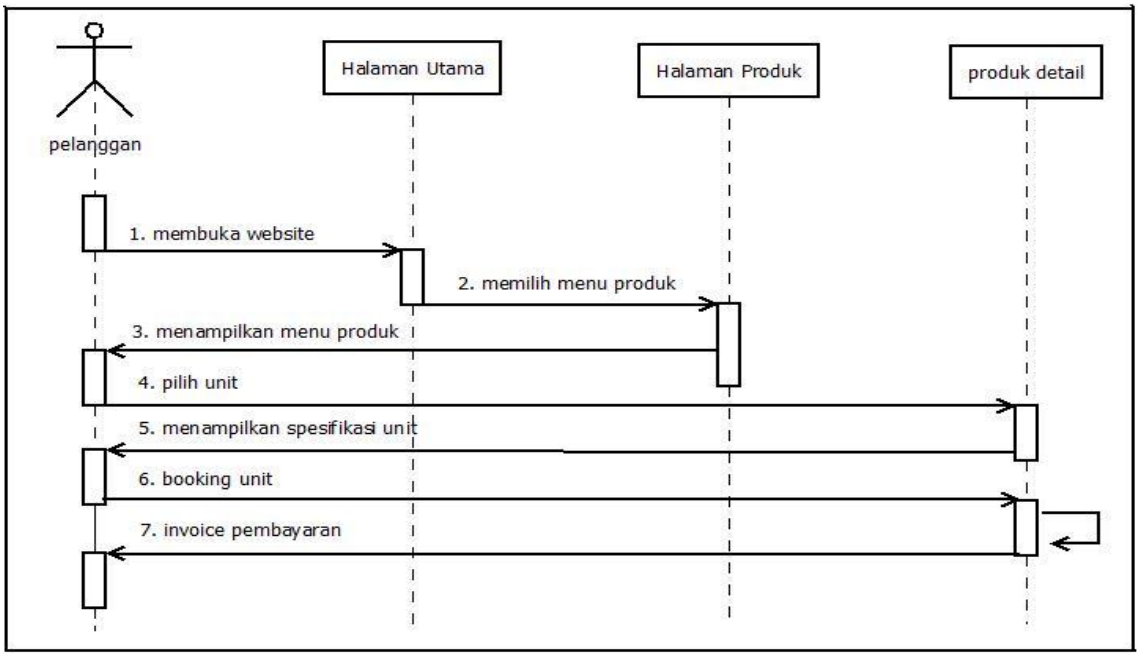
1. Diagram Sekuensial Pemasaran



Gambar 4.6 *Diagram Sekuensial Pemasaran*

Deskripsi : Proses pemasaran dilakukan secara online. Admin menginput data unit yang tersedia beserta info-info terbaru mengenai Apartemen Easton Park.

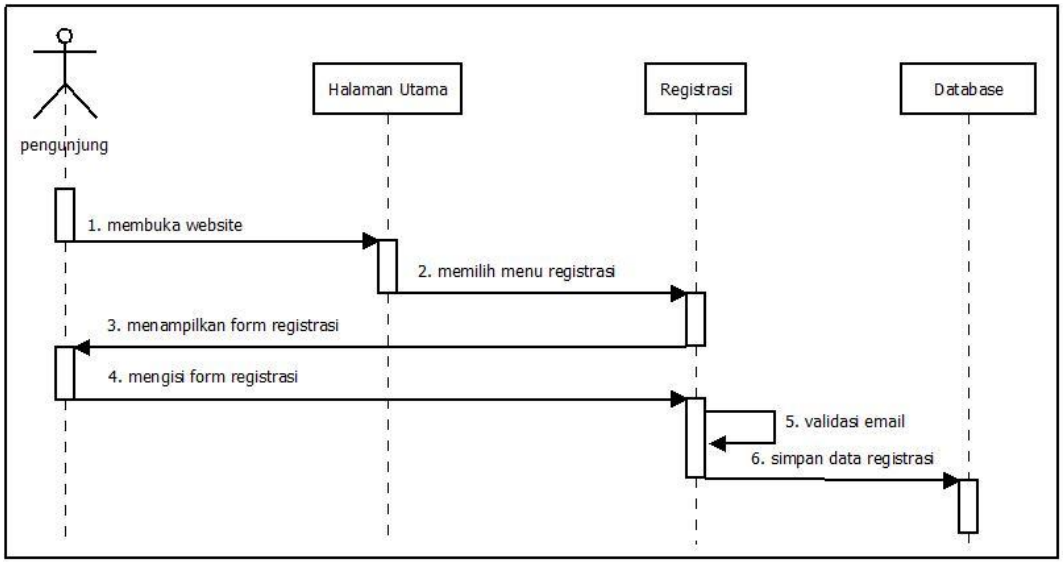
2. Diagram Sekuensial Pemesanan



Gambar 4.7 *Diagram Sekuensial Pemesanan*

Deskripsi : Proses pemesanan dilakukan secara online. Pelanggan dapat melakukan pemesanan unit Apartemen Easton Park dengan memilih unit yang tersedia.

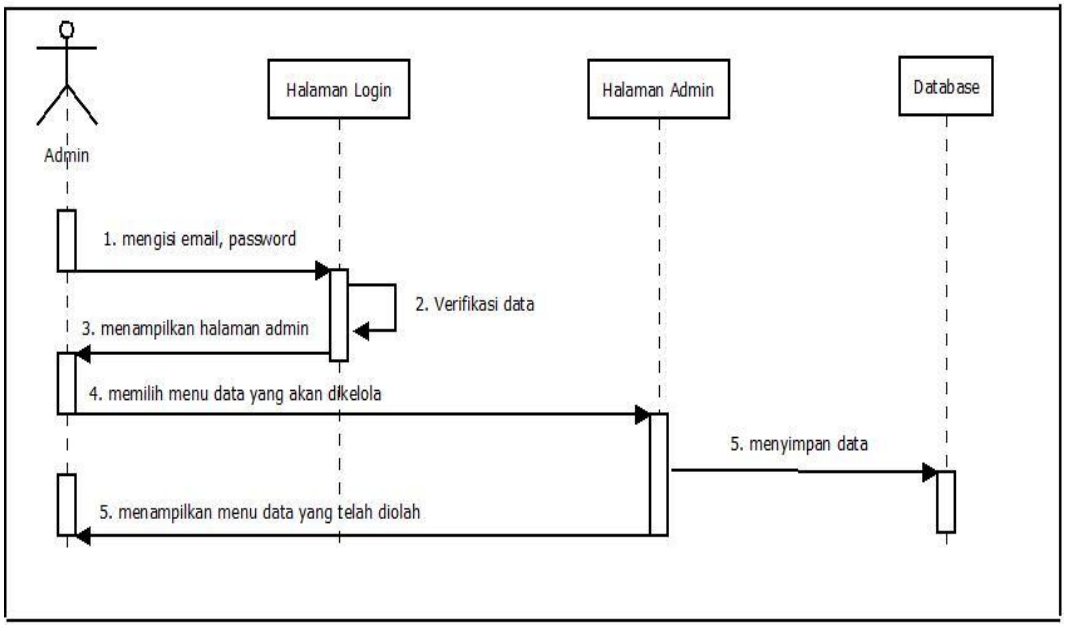
3. Diagram Sekuensial Registrasi



Gambar 4.8 *Diagram Sekuensial Registrasi*

Deskripsi : Proses mengisi form registrasi dilakukan pelanggan yang hendak melakukan pemesanan Unit Apartemen Easton Park. Form Registrasi ini berfungsi untuk menghindari penginputan secara asal oleh orang awam.

4. Diagram Sekuensial Olah Data Pemesanan

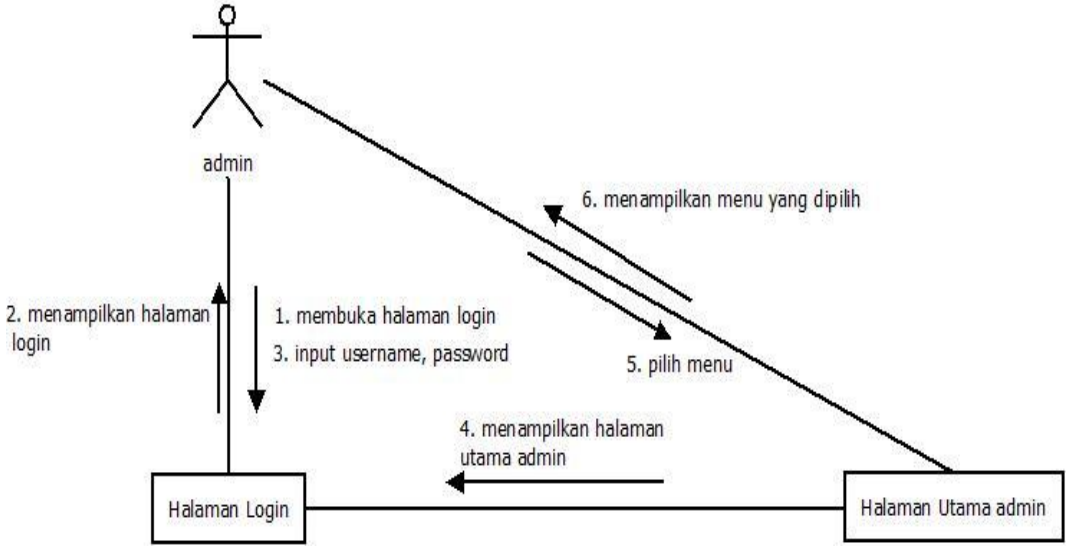


Gambar 4.9 *Diagram Sekuensial Olah Data Pemesanan*

Deskripsi : Proses mengolah data pemesanan hanya bisa diakses manager pemasaran. Manager pemasaran dapat mengelola data pemesanan mulai dari master data, user, dan invoice.

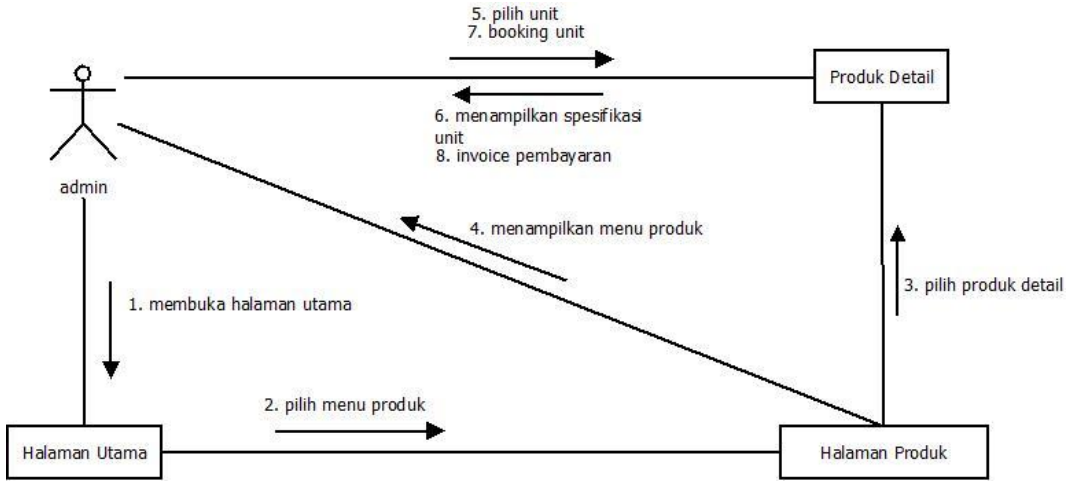
4.1.3.5.Collaboration Diagram

1. Collaboration Diagram Pemasaran



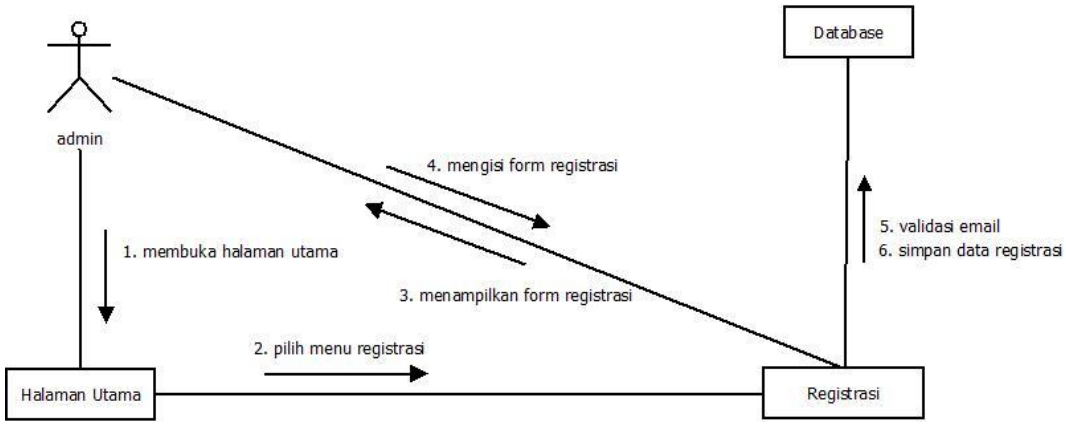
Gambar 4.10 *Collaboration Diagram Pemasaran*

2. Collaboration Diagram Pemesanan



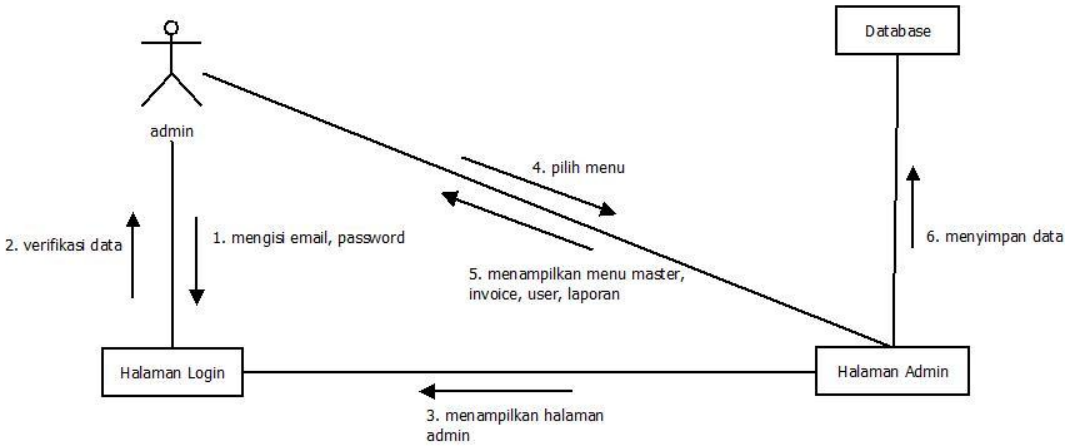
Gambar 4.11 *Collaboration Diagram Pemesanan*

3. Collaboration Diagram Registrasi



Gambar 4.12 *Collaboration Diagram Registrasi*

4. Collaboration Diagram Olah Data Pemesanan



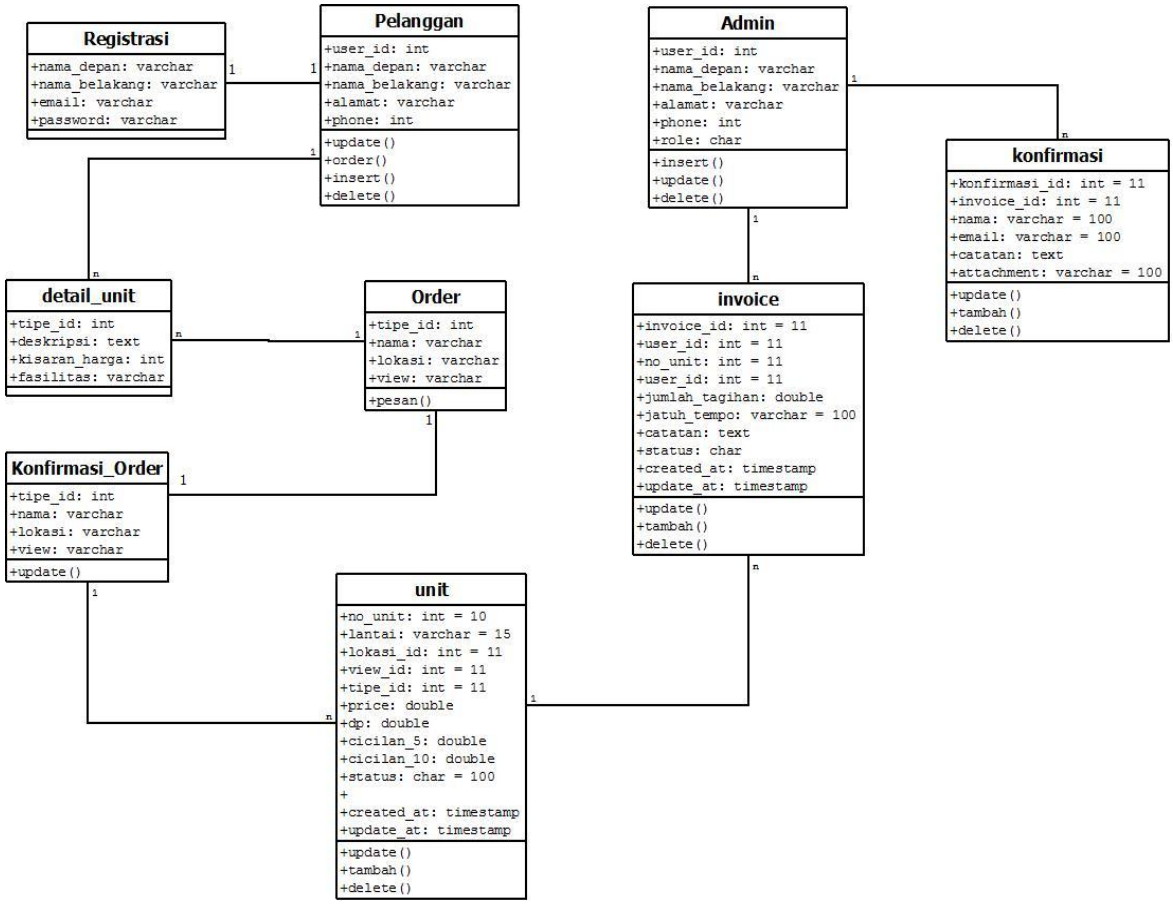
Gambar 4.13 *Collaboration Diagram Olah Data Pemesanan*

4.1.4 Perancangan Basis Data

4.1.4.1 Diagram Kelas

Class diagram membantu kita dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai.Class diagram memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap

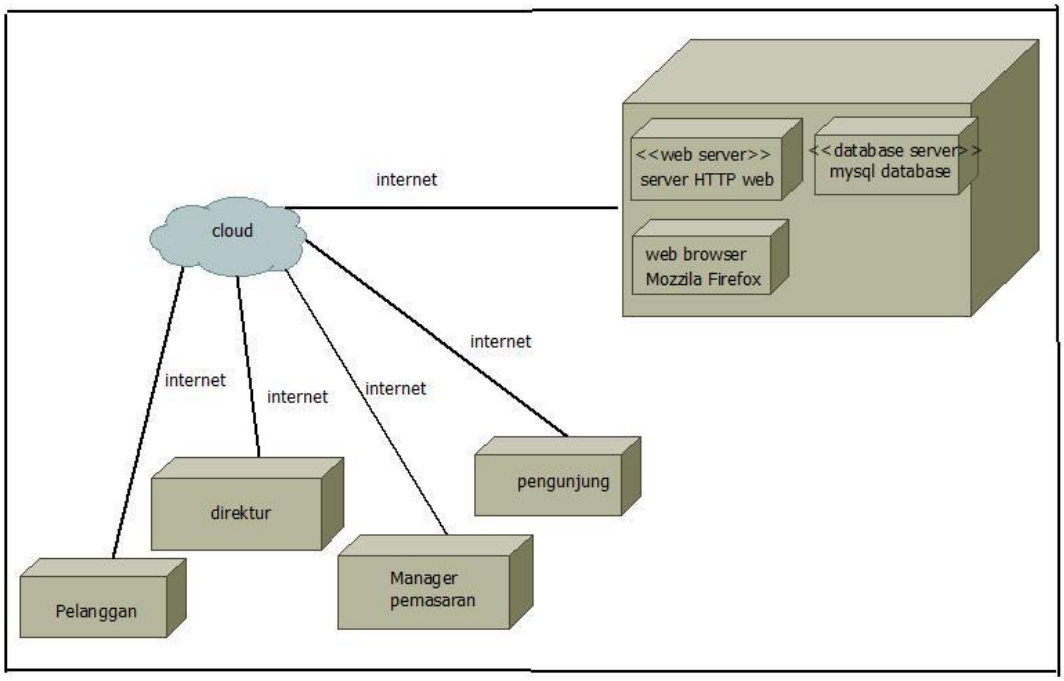
kelas didalam model desain (dalam logical view) dari suatu sistem. Selama proses analisa, class diagram memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem, selama tahap desain class diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur yang dibuat.



Gambar 4.14 *Diagram Class* Sistem Informasi yang diusulkan

4.1.4.2 Deployment Diagram

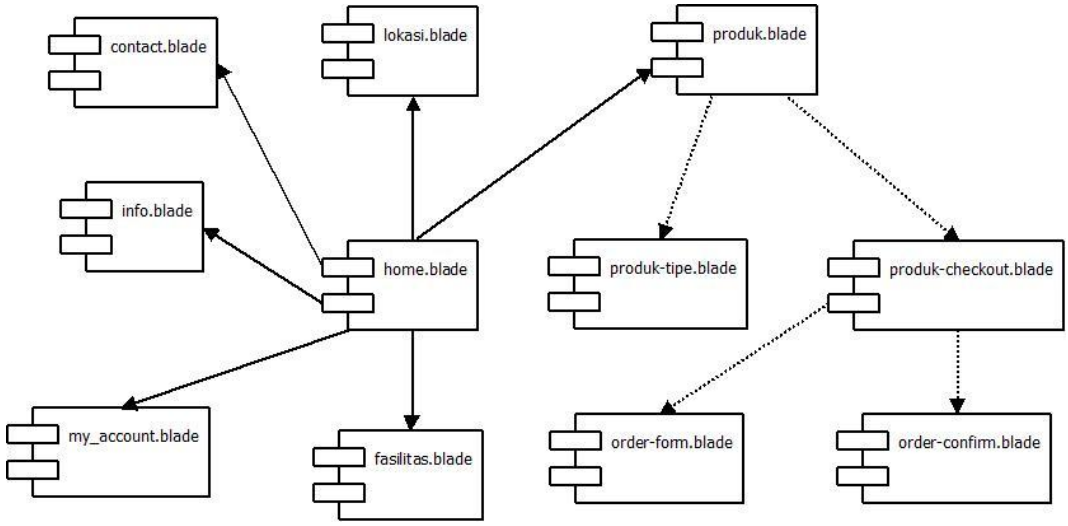
Deployment view merupakan suatu tampilan atau pandangan/kinerja dari sebuah sistem yang baru sesuai dengan perancangan data yang diambil dari beberapa objek.



Gambar 4.15 *Deployment Diagram* Sistem Informasi yang diusulkan

4.1.4.3 Component Diagram

Diagram Komponen atau *component diagram* menunjukan model secara fisik komponen perangkat lunak pada sistem dan hubungan antar mereka. Component adalah unit fisik yang nyata yang menjadi bagian dari deployment independent. Component ini diimplementasikan meskipun pada sistem yang kecil. Perancangan menu pada aplikasi yang diusulkan digambarkan oleh gambar component diagram berikut :



Gambar 4.16 *Component Diagram* Sistem Informasi yang diusulkan

4.1.4.4 Struktur File

1. Tabel Data Unit

   Nama Tabel : Unit

   Primary Key : no_unit

   Jumlah field :13

   Tabel 4.5 Tabel Data Unit

No	Nama Field	Type	Size	Keterangan
1	No_unit	Int	10	No urut unit
2	Lantai	Varchar	15	Posisi lantai unit
3	Lokasi_id	Integer	11	Lokasi unit

4	View_id	Integer	11	Tipe hadap unit
5	Tipe_id	Integer	11	Tipe unit apartemen
6	Price	Double		Harga unit
7	Dp	Double		Dp unit
8	Cicilan_5	double		Harga cicilan 5kali
9	Cicilan_10	Double		Harga cicilan 10kali
10	Status	char	1	Status booking
11	Created_at	Timpestamp		Tanggal dibuat
12	update_at	Timestamp		Tanggal update

2. Tabel Tipe Unit

Nama Tabel : Tipe_unit

Primary Key : tipe_id

Jumlah field :7

Tabel 4.6 Tabel Tipe Unit

No	Nama Field	Type	Size	Keterangan
1	Tipe_id	Integer	10	No urut tipe
2	Nama	varchar	100	Nama tipe
3	Sg	varchar	255	Luas tipe

4	Deskripsi	Text		Deskripsi tipe
5	Gambar	varchar	200	Gambar tipe
6	Created_at	Timestamp		Tanggal dibuat
7	Update_at	Timestamp		Tanggal update

3. Tabel view unit

Nama Tabel : view_unit

Primary Key : view_id

Jumlah field :6

Tabel 4.7 Tabel View Unit

No	Nama Field	Type	Size	Keterangan
1	View_unit	Integer	10	No urut tipe
2	Nama	varchar	100	Nama tipe
3	Deskripsi	Text		Deskripsi tipe
4	Gambar	Varchar	200	Gambar
5	Created_at	Timestamp		Tanggal dibuat
6	Update_at	Timestamp		Tanggal update

4. Tabel lokasi

Nama Tabel : lokasi

Primary Key : lokasi_id

Jumlah field :6

Tabel 4.8 Tabel lokasi

No	Nama Field	Type	Size	Keterangan
1	Lokasi_id	Integer	10	No urut tipe
2	Nama	varchar	100	Nama tipe
3	Deskripsi	Text		Deskripsi tipe
4	Gambar	Varchar	200	Gambar
5	Created_at	Timestamp		Tanggal dibuat
6	Update_at	Timestamp		Tanggal update

5. Tabel invoice

Nama Tabel : invoice

Primary Key :invoice_id

Jumlah field :8

Tabel 4.9 Tabel Invoice

No	Nama Field	Type	Size	Keterangan
1	Invoice_id	Integer	10	No urut invoice
2	User_id	integer	11	No urut user
3	Jumlah_tagihan	double		Jumlah tagihan
4	Jatuh_tempo	Varchar	200	Jatuh tempo booking
5	Catatan	text		Catatan
6	Status	Char	1	Status booking

7	Created_at	Timestamp		Tanggal dibuat
8	Update_at	Timestamp		Tanggal update

6. Tabel user

Nama Tabel : user

Primary Key : user_id

Jumlah field :12

Tabel 4.10 Tabel User

No	Nama Field	Type	Size	Keterangan
1	User_id	Int	10	No urut user
2	Nama_depan	Varchar	100	Nama depan
3	Nama_belakang	Varchar	100	Nama belakang
4	Email	Varchar	100	Email
5	Nik	Integer	11	Nik
6	Phone	Varchar	20	Phone
7	Alamat	Varchar	250	Alamat
8	Password	Varchar	150	Password
9	Role	Char	255	Pilihan user
10	Remember_token	Varchar	100	Pengingat

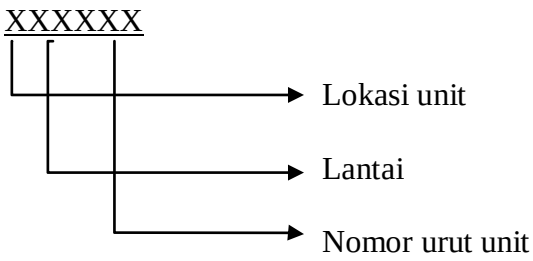
11	Created_at	Timpestamp		Tanggal dibuat
12	update_at	Timestamp		Tanggal update

4.1.4.5 Kodefikasi

Kodefikasi digunakan sebagai identitas untuk setiap data yang akan diinput dan untuk mengidentifikasi suatu objek secara singkat. Kode dibuat dalam bentuk gabungan angka dan huruf. Pengkodean di CV.AMADDA bertujuan untuk mempermudah dalam memasukkan data dan dalam melakukan pencarian data.

Adapun rincian kodefikasi data yang ada pada website pemasaran dan pemesanan apartemen Easton Park di CV.AMADDA dapat dilihat dibawah ini :

1. No_unit



Contoh : C01011

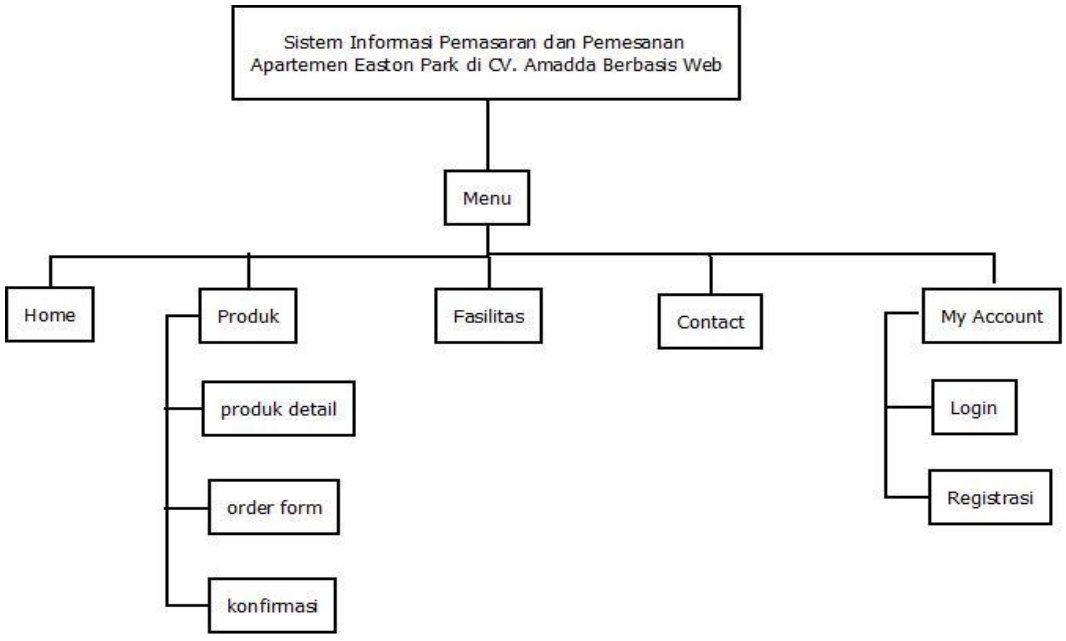
4.2 Perancangan Antar Muka

Perancangan antar muka merupakan penggambaran tampilan (interface) sebuah sistem yang akan digunakan secara langsung oleh pengguna, dan juga merupakan penggambaran interaksi yang dapat dilakukan oleh pengguna dalam sistem. Adapun dalam perancangan antar muka ini terdapat beberapa bagian yang harus dilakukan, yaitu dari menentukan rancangan input dan output pada setiap fungsi yang telah ditentukan.

4.2.1 Struktur Menu

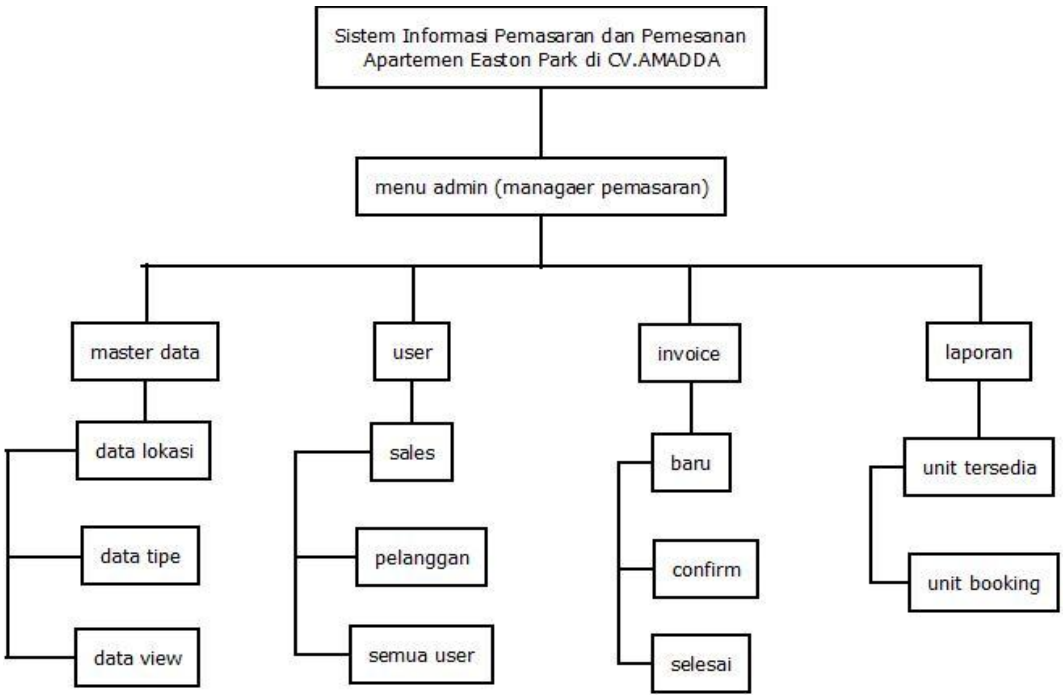
Pada program yang di rancang oleh penulis di gunakan struktur menu, hal ini bertujuan untuk memudahkan pemakai dalam berhubungan dengan sistem komputer. Struktur menu ini terdiri dari menu utama yang terbagi dalam beberapa sub bagian. Berikut ini adalah struktur menu website yang diusulkan.

1. Struktur menu halaman utama



Gambar *4.17* Struktur Menu Halaman Utama

2. Struktur menu halaman admin



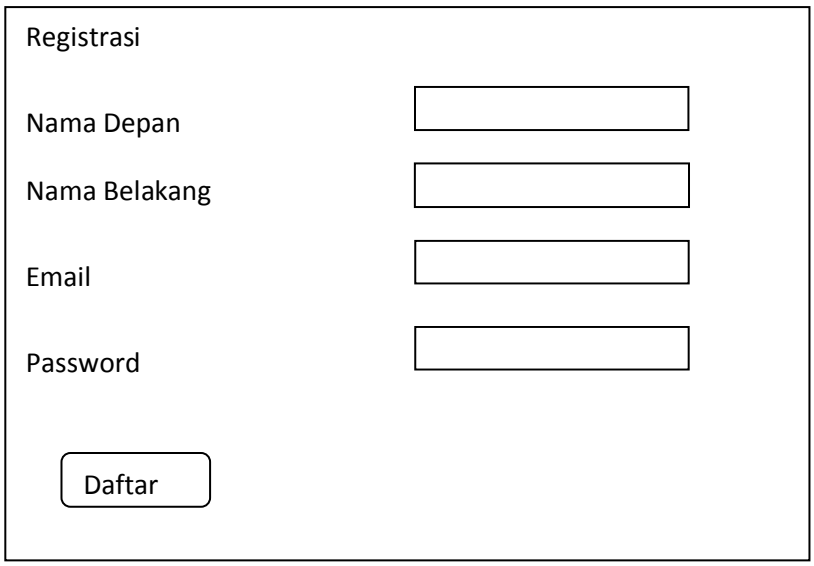
Gambar ***4.18*** Struktur Menu Halaman Admin

4.2.2 Perancangan Input

Perancangan input berfungsi untuk memasukkan data ke database. Berikut ini adalah perancangan input yang akan dibuat pada Sistem Informasi Pemasaran dan Pemesanan Apartemen Easton Park di CV.AMADDA :

1. Perancangan Registrasi

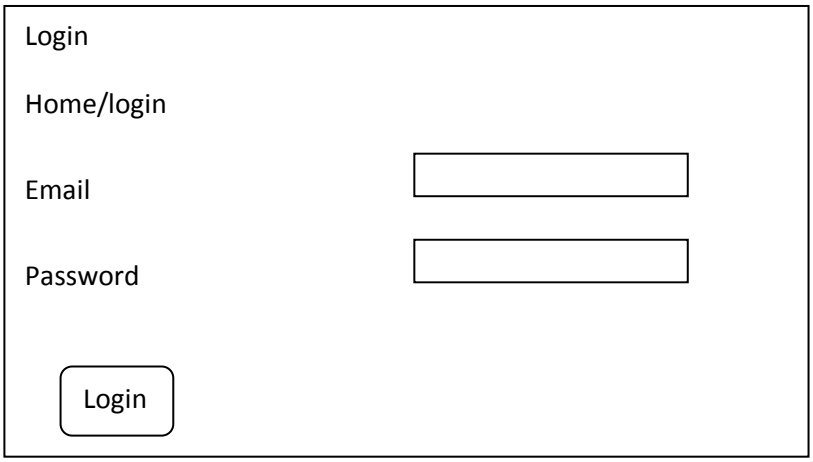
   Merupakan form registrasi yang harus diisi oleh pengunjung yang ingin melakukan pemesanan tetapi belum memiliki account untuk login.



Gambar 4.19 Perancangan Registrasi

2. Perancangan Login

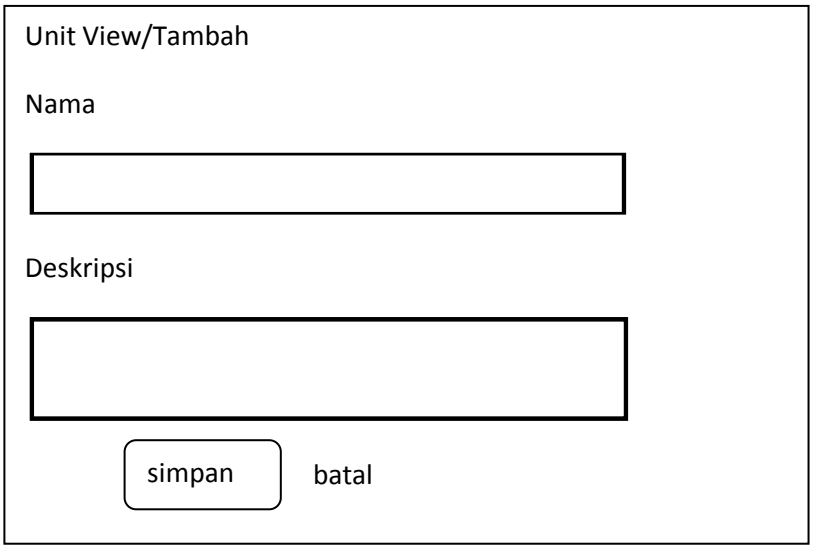
Login merupakan langkah awal untuk menyeleksi user yang dapat mengakses admin panel ataupun proses seleksi pelanggan yang akan melakukan pemesanan.



Gambar 4.20 Perancangan Login

3. Rancangan Master Data View

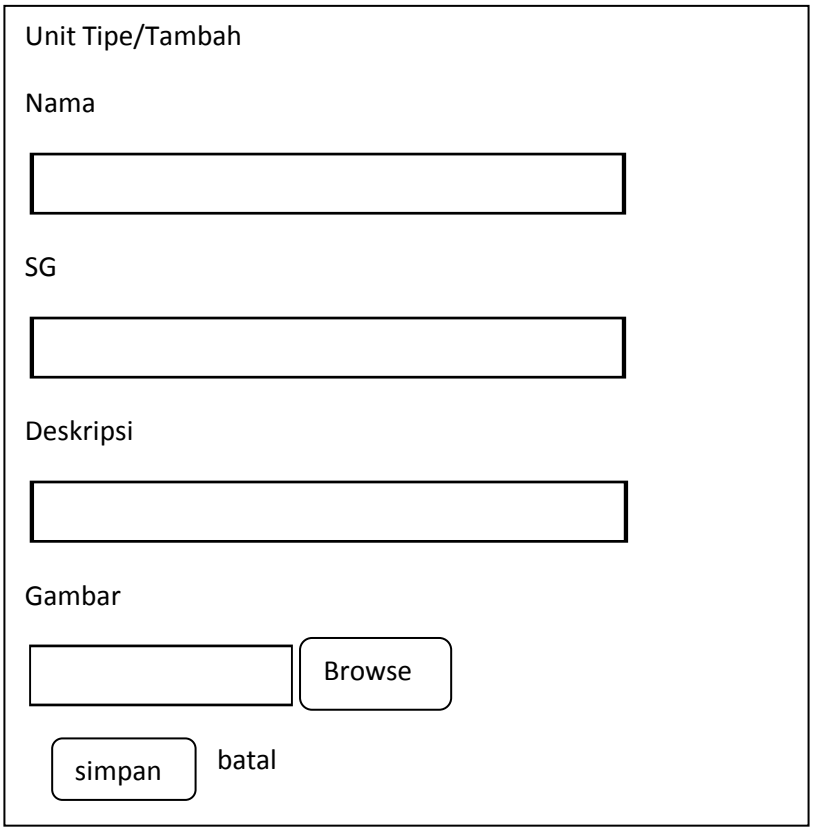
   Master data view merupakan form yang digunakan untuk input data view atau hadap unit.



Gambar 4.21 Perancangan Input Master Data View

4. Rancangan Master Data Tipe

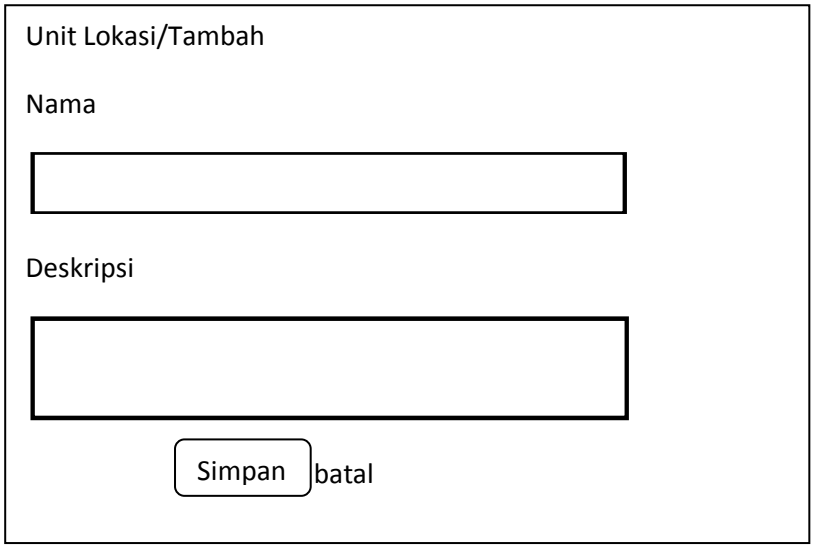
   Master Data Tipe merupakan form untuk mengisi Tipe unit yang baru dengan Luas SG unit tersebut. Beserta Gambar unit yang akan di upload.



Gambar 4.22 Perancangan Master Data Tipe

5. Rancangan Master Data Lokasi

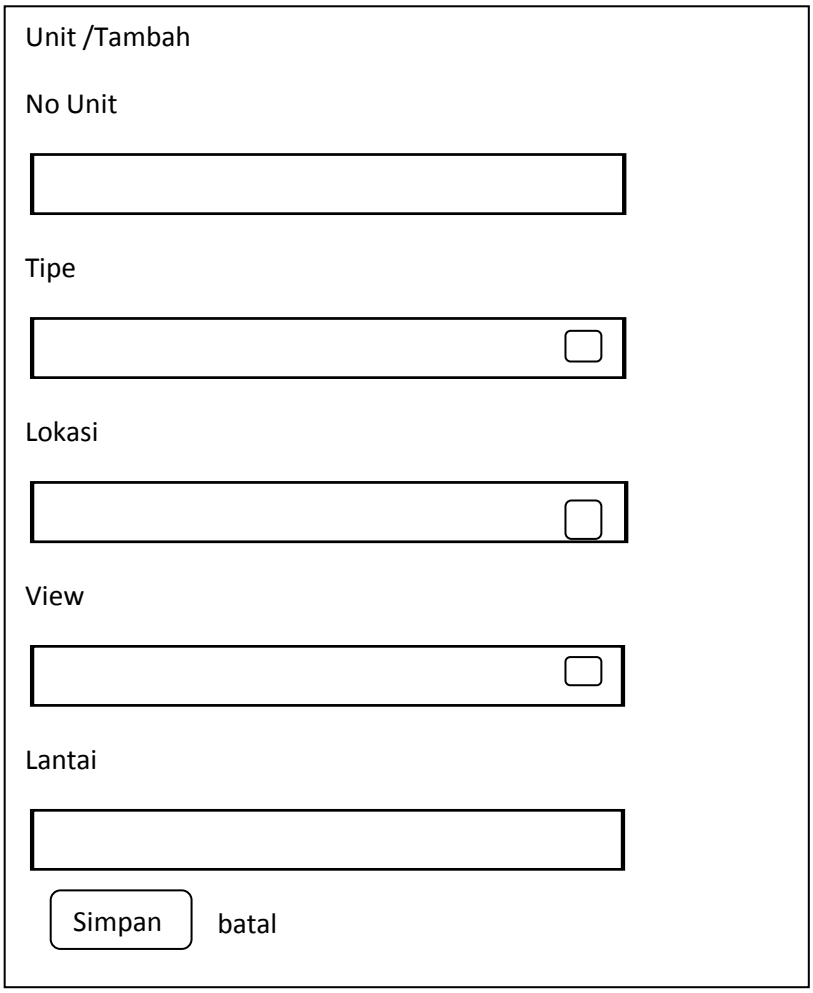
Master data Lokasi merupakan form untuk mengisi Lokasi gedung dari unit.



Gambar 4.23 Perancangan Master Data Lokasi

6. Rancangan Master Data Unit

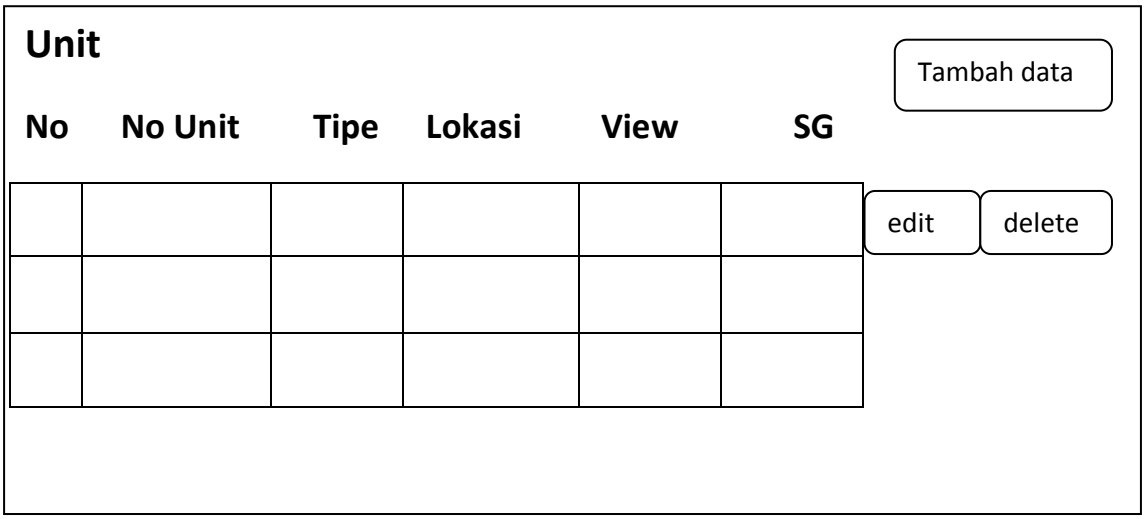
Master data unit merupakan form untuk mengisi data unit yang baru.



Gambar 4.24 Perancangan Master Data Unit

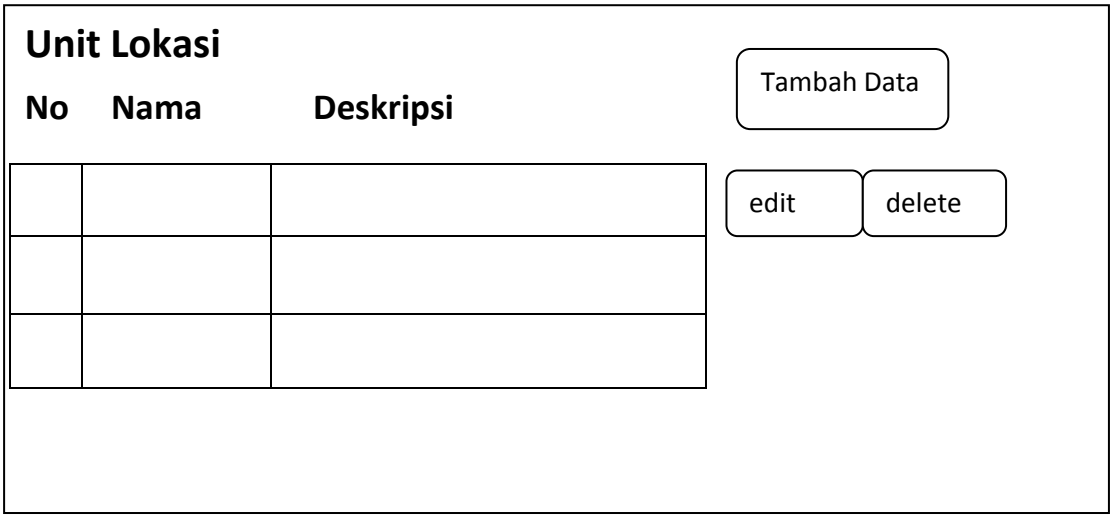
4.2.3 Perancangan Output

1. Output Data Unit



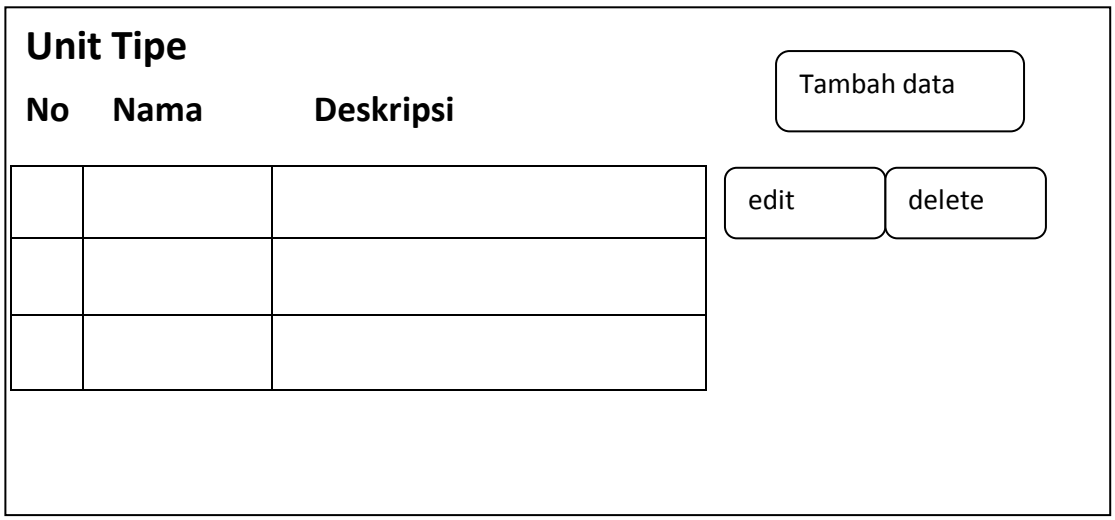
Gambar 4.25 Output Data Unit

2. Output Data Unit Lokasi



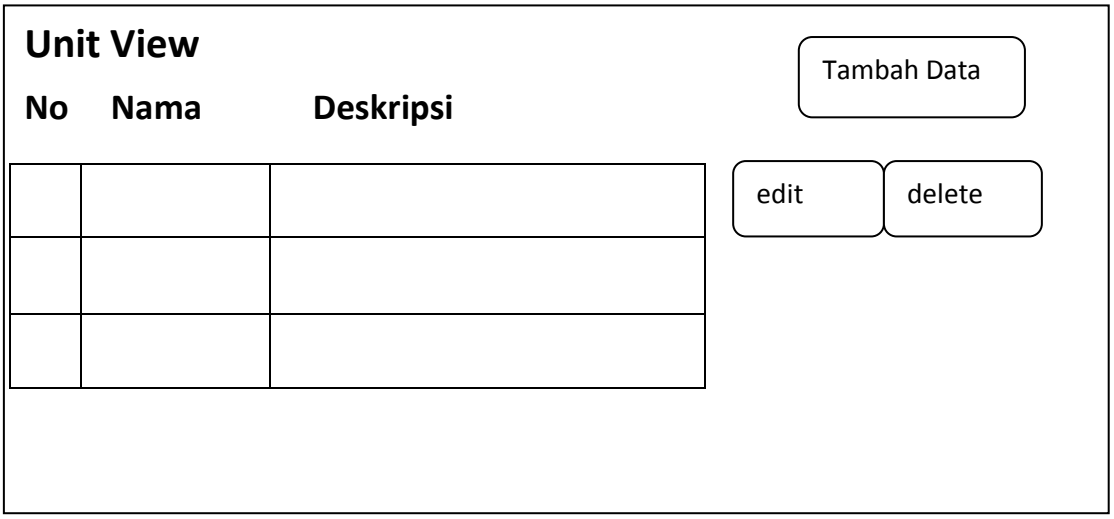
Gambar 4.26 Output Data Unit Lokasi

3. Output Data Tipe



Gambar 4.27 Output Master Data Tipe

4. Output Data View

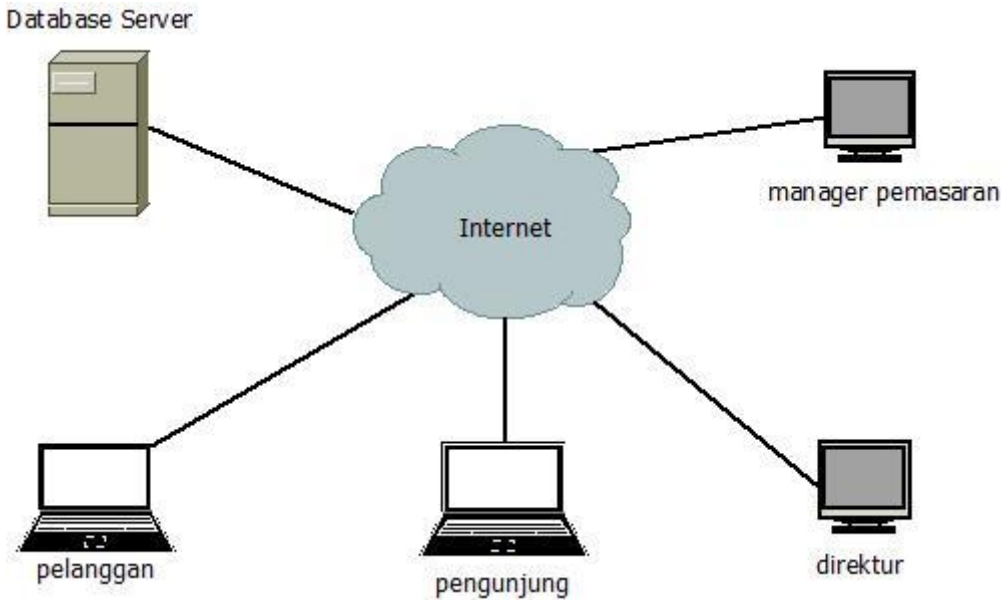


Gambar 4.28 Output Master Data View

4.3 Perancangan Arsitektur Jaringan

Perancangan arsitektur jaringan yang akan digunakan pada CV.AMADDA dalam pembangunan sistem informasi ini yaitu dengan menggunakan akses jaringan internet atau bisa disebut berbasis *online*.

User (pengunjung, pelanggan, manager pemasaran dan direktur) dapat mengakses website dimanapun dan kapanpun asalkan adanya akses internet. Berikut ini adalah gambar perancangan arsitektur jaringan yang diusulkan.



Gambar 4.29 Arsitektur Jaringan CV.AMADDA

4.4 Implementasi

. Implementasi merupakan kumpulan dari elemen-elemen yang telah didesain kedalam bentuk pemograman untuk menghasilkan suatu tujuan yang dibuat berdasarkan kebutuhan yang telah dianalisis sebelumnya. Implementasi RUP Sistem Informasi Pemasaran dan Pemesanan Apartemen Easton Park di CV. AMADDA ini dibentuk menggunakan bahasa

pemrograman PHP, dengan basis data yang digunakan adalah MySQL. Bahasa pemograman PHP dapat dijalankan di berbagai Sistem Operasi tetapi penulis disini menggunakan Windows & dikarenakan cukup banyak dan sering digunakan.

4.4.1 Batasan Implementasi

Batasan implementasi dari pembuatan aplikasi sistem informasi akademik ini, sebagai berikut :

1. Pada sistem informasi pemasaran dan pemesanan berbasis web ini, penulis hanya mengambil batasan tentang pengelolaan data unit dan laporan.
2. Pada implementasi database, penulis menggunakan sistem basis data MySQL

4.4.2 Implementasi Perangkat Lunak

Perangkat lunak digunakan untuk mendukung kinerja sistem operasi dan aplikasi *database*. Adapun perangkat lunak yang digunakan adalah sebagai berikut :

1. Implementasi perangkat lunak yang digunakan oleh perancang sistem :
    a. Adobe Dreamweaver
    b. Sublime text 2
    c. Xampp
    d. Windows 7
    e. Git Bash

f. Mozzila Firefox

2. Implementasi perangkat lunak yang digunakan oleh pengguna sistem :
   a. browser
   b. minimal windows XP

4.4.3 Implementasi Perangkat Keras

Perangkat keras yang di gunakan oleh perancangan pengembangan sistem dan pengguna sistem yaitu sebagai berikut :

1. Pada sisi perancang sistem :
   a. Menggunakan laptop processor Intel Core i3
   b. RAM 2GB
   c. Hardisk untuk penampungan data minimal 500GB
   d. Mouse
   e. LAN atau Modem
2. Pada sisi pengguna sistem di jelaskan perangkat keras yang di butuhkan berdasarkan kebutuhan minimal yang harus terpenuhi yaitu :
   a. Menggunakan processor minimal intel dual core 2.2 Ghz
   b. RAM minimal 1GB
   c. Mouse, Keyboard, dan monitor sebagai peralatan antar muka.
   d. Modem.

4.4.4 Implementasi Basis Data (sintaks SQL)

-- phpMyAdmin SQL Dump

-- version 4.1.12

```
-- http://www.phpmyadmin.net

--

-- Host: 127.0.0.1

-- Generation Time: Jul 08, 2014 at 07:01 PM

-- Server version: 5.6.16

-- PHP Version: 5.5.11


SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";

SET time_zone = "+00:00";



/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS
*/;

/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;

/*!40101 SET NAMES utf8 */;
```

```
--

-- Database: `apartement`

--



-- --------------------------------------------------------



--

-- Table structure for table `invoice`

--



CREATE TABLE IF NOT EXISTS `invoice` (

  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,

  `user_id` int(11) NOT NULL,

  `jumlah_tagihan` double NOT NULL,

  `jatuh_tempo` date NOT NULL,

  `catatan` text COLLATE utf8_unicode_ci NOT NULL,

  `status` char(1) COLLATE utf8_unicode_ci NOT NULL,

  `created_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',

  `updated_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
```

```
  PRIMARY KEY (`id`)

) ENGINE=InnoDB  DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
AUTO_INCREMENT=9 ;


--

-- Dumping data for table `invoice`

--


INSERT INTO `invoice` (`id`, `user_id`, `jumlah_tagihan`, `jatuh_tempo`,
`catatan`, `status`, `created_at`, `updated_at`) VALUES

(1, 1, 5000000, '2014-07-10', 'Tipe: London, Lokasi: wing1, View: mountain', '0',
'2014-07-03 11:42:11', '2014-07-03 11:42:11'),

(2, 1, 5000000, '2014-07-11', 'Tipe: London, Lokasi: wing1, View: mountain', '0',
'2014-07-03 18:11:07', '2014-07-03 18:11:07'),

-- --------------------------------------------------------


--

-- Table structure for table `lokasi`

--
```

```
CREATE TABLE IF NOT EXISTS `lokasi` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `deskripsi` text COLLATE utf8_unicode_ci NOT NULL,
  `gambar` varchar(200) COLLATE utf8_unicode_ci NOT NULL,
  `created_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `updated_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
AUTO_INCREMENT=2 ;

--
-- Dumping data for table `lokasi`
--

INSERT INTO `lokasi` (`id`, `nama`, `deskripsi`, `gambar`, `created_at`,
`updated_at`) VALUES
(1, 'wing1', 'sdas', '', '2014-07-03 11:35:47', '2014-07-03 11:35:47');
```

```
-- --------------------------------------------------------

--

-- Table structure for table `migrations`

--

CREATE TABLE IF NOT EXISTS `migrations` (
  `migration` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `batch` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

--

-- Dumping data for table `migrations`

--

INSERT INTO `migrations` (`migration`, `batch`) VALUES
('2014_06_19_221746_create_table_users', 1),
('2014_06_19_222744_create_table_units', 1),
('2014_06_19_230332_create_table_locations', 1),
```

```
('2014_06_19_230606_create_table_unit_types', 1),

('2014_06_19_230807_create_table_unit_views', 1),

('2014_06_23_152332_move_field_sg', 2),

('2014_06_23_153125_update_detail_unit', 2),

('2014_06_24_132726_update_user_role_detail', 2),

('2014_07_01_152031_create_invoice_table', 2);



-- --------------------------------------------------------



--

-- Table structure for table `tipe_unit`

--



CREATE TABLE IF NOT EXISTS `tipe_unit` (

  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,

  `nama` varchar(100) COLLATE utf8_unicode_ci NOT NULL,

  `sg` varchar(255) COLLATE utf8_unicode_ci NOT NULL,

  `deskripsi` text COLLATE utf8_unicode_ci NOT NULL,

  `gambar` varchar(200) COLLATE utf8_unicode_ci NOT NULL,
```

```
  `created_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `updated_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
AUTO_INCREMENT=4 ;

--
-- Dumping data for table `tipe_unit`
--

INSERT INTO `tipe_unit` (`id`, `nama`, `sg`, `deskripsi`, `gambar`, `created_at`,
`updated_at`) VALUES
(1, 'London', '39', 'abcd', 'UNIT SC=PARIS.jpg', '2014-07-03 11:30:24', '2014-07-
03 11:37:40'),
-- --------------------------------------------------------

--
-- Table structure for table `unit`
--
CREATE TABLE IF NOT EXISTS `unit` (
```

```
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `no_unit` varchar(15) COLLATE utf8_unicode_ci NOT NULL,
  `lantai` varchar(10) COLLATE utf8_unicode_ci NOT NULL,
  `lokasi_id` int(11) NOT NULL,
  `view_id` int(11) NOT NULL,
  `tipe_id` int(11) NOT NULL,
  `price` double NOT NULL,
  `dp` double NOT NULL,
  `cicilan_5` double NOT NULL,
  `cicilan_10` double NOT NULL,
  `status` char(1) COLLATE utf8_unicode_ci NOT NULL,
  `created_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `updated_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`id`),
  UNIQUE KEY `unit_no_unit_unique` (`no_unit`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
AUTO_INCREMENT=1 ;


-- --------------------------------------------------------
```

```
--

-- Table structure for table `user`

--


CREATE TABLE IF NOT EXISTS `user` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama_depan` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `nama_belakang` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `email` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `nik` int(11) NOT NULL,
  `phone` varchar(20) COLLATE utf8_unicode_ci NOT NULL,
  `alamat` varchar(250) COLLATE utf8_unicode_ci NOT NULL,
  `password` varchar(150) COLLATE utf8_unicode_ci NOT NULL,
  `role` char(255) COLLATE utf8_unicode_ci NOT NULL,
  `remember_token` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `created_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `updated_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`id`),
```

```
  UNIQUE KEY `user_email_unique` (`email`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
AUTO_INCREMENT=6 ;


--
-- Dumping data for table `user`
--
-- Table structure for table `view_unit`
--


CREATE TABLE IF NOT EXISTS `view_unit` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama` varchar(100) COLLATE utf8_unicode_ci NOT NULL,
  `deskripsi` text COLLATE utf8_unicode_ci NOT NULL,
  `gambar` varchar(200) COLLATE utf8_unicode_ci NOT NULL,
  `created_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `updated_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`id`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
AUTO_INCREMENT

--

-- Dumping data for table `view_unit`

--

--

-- Constraints for dumped tables

--

--

-- Constraints for table `unit`

--

ALTER TABLE `unit`

 ADD CONSTRAINT `unit_ibfk_1` FOREIGN KEY (`id`) REFERENCES
`tipe_unit` (`id`);

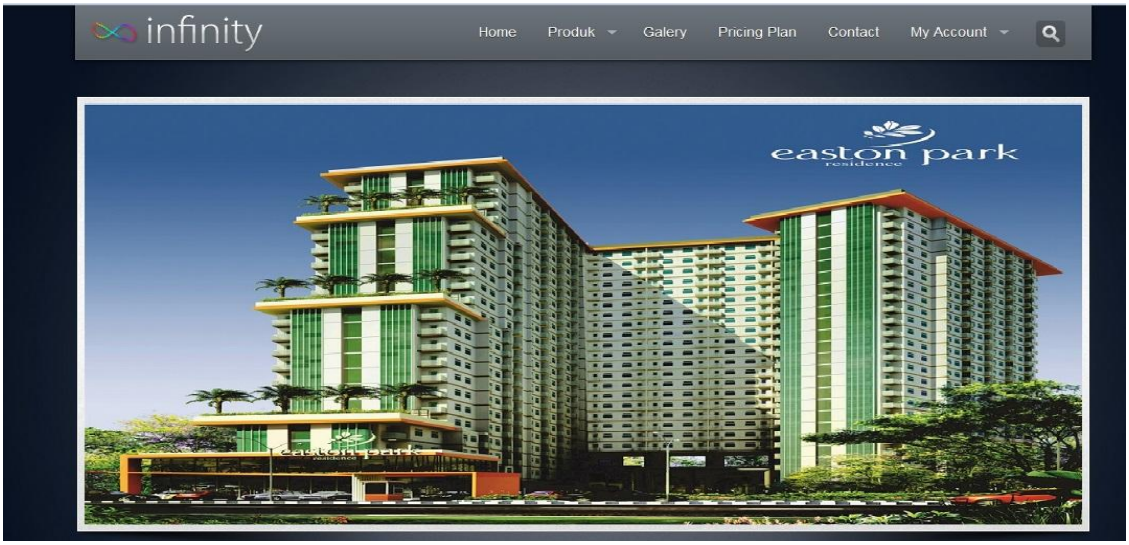

/*!40101                                                    SET
CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101                                                    SET
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
```

/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

4.4.5 Implementasi Antar Muka

1. Implementasi Antar Muka Halaman Utama



Gambar 4.30 Halaman Utama

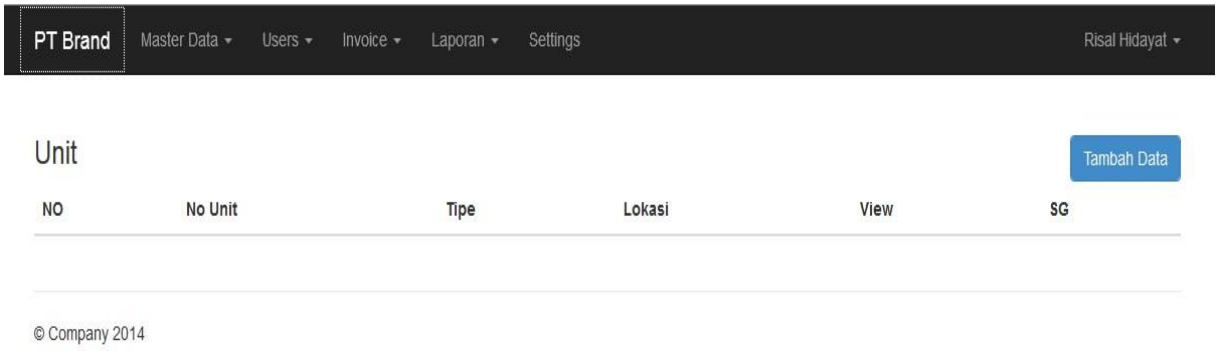
Halaman Utama merupakan halaman awal dari website. Pada halaman ini tidak ada batasan hak akses, jadi setiap orang dapat mengakses halaman ini beserta fasilitas yang ada di halaman ini. Berikut rincian penjelasan dari beberapa menu yang ada pada halaman utama :

Tabel 4.11 Rincian menu Halaman Utama

Menu	Sub Menu	Deskripsi	Nama File
Home		Menu tampilan awal website	Home.blade

Produk	Produk Detail Order Form Konfirmasi	Menampilkan unit yang ditawarkan beserta pemesanannya	Produk.blade
Fasilitas		Menampilkan Fasilitas yang dimiliki Easton Park	Fasilitas.blade
Contact		Menampilkan form untuk mengirim pesan	Contact.blade
My Account		Menampilkan form login dan registrasi	My_account.blade

2. Implementasi Antar Muka Halaman Admin



Gambar 4.31 Halaman Admin

Halaman admin merupakan tampilan awal untuk admin (manager pemasaran). Pada halaman ini akses dibatasi, hanya manager pemasaran saja yang dapat mengakses halaman ini beserta fasilitas-fasilitasnya. Adapun penjelasan setiap menunya akan dijelaskan pada tabel dibawah ini.

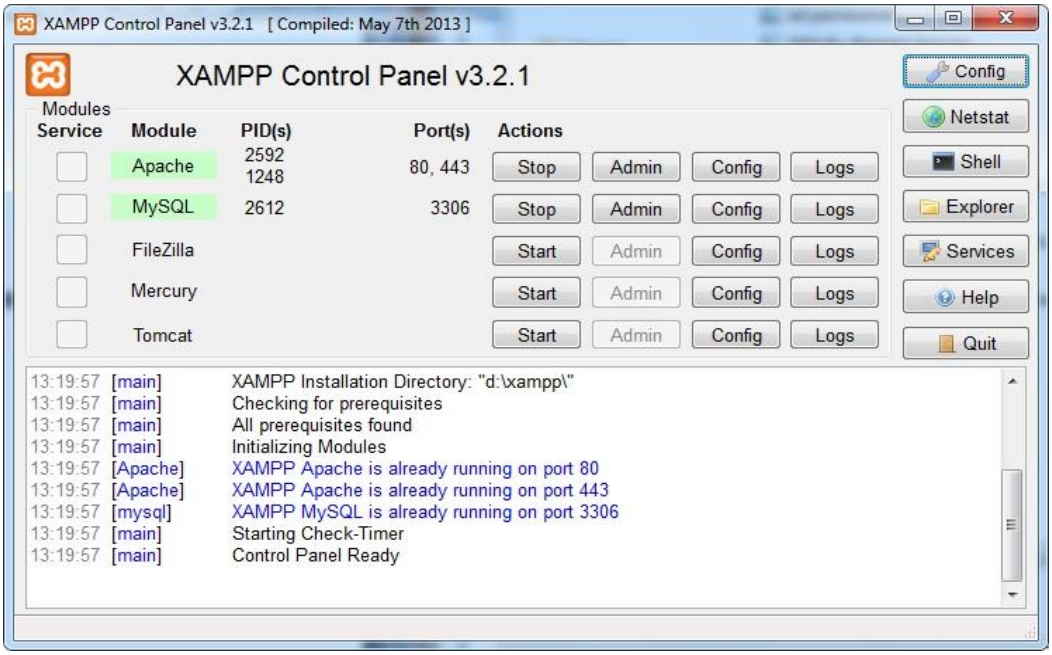
Tabel 4.12 Rincian Menu Halaman Admin

Menu	**Sub Menu**	**Deskripsi**	**Nama File**
Master data	Data View	Menampilkan form data untuk input data view	UnitView.php
	Data Lokasi	Menampilkan form data Lokasi	UnitLocation.php
	Data Tipe	Menampilkan form data tipe unit	UnitType.php
	Data Unit	Menampilkan form unit	Unit.php
Users		Menampilkan user yang dapat login	User.php
Invoice		Menampilkan pemesanan pending atau diterima	Invoice.php
Laporan		Menampilkan laporan	Laporan.php

4.4.6 Implementasi Instalasi Program

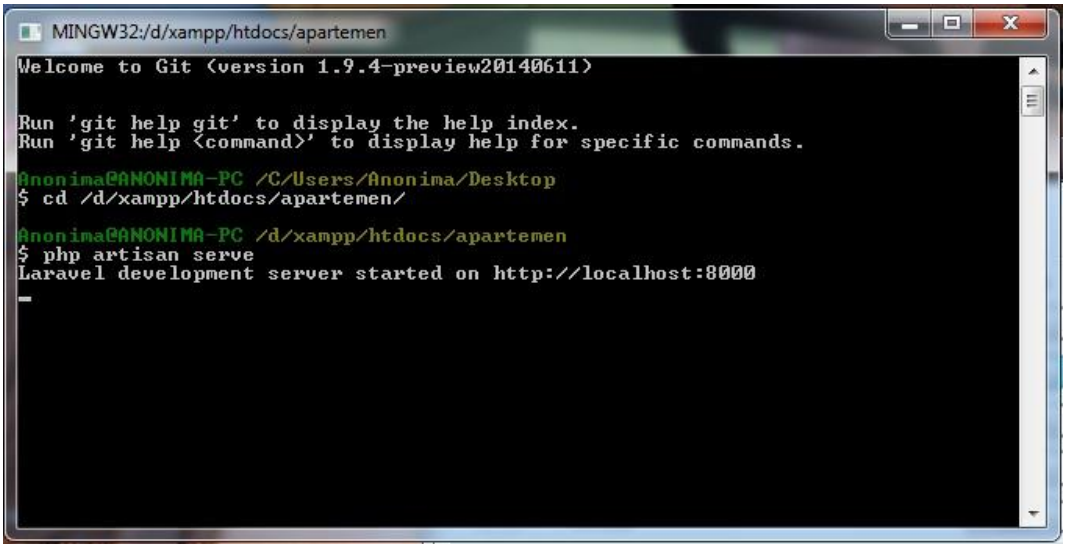
Berikut ini adalah implementasi instalasi program untuk menggunakan program Sistem Informasi Pemasaran dan Pemesanan Apartemen Easton Park di CV. AMADDA. Langkah-langkahnya adalah sebagai berikut :

1. Klik XAMPP Control Panel Application pada desktop.
2. Klik Start Apache dan MySql pada XAMPP Control Panel Application seperti pada gambar dibawah ini.



Gambar 4.32 *Running* Xampp

3. Buka Git bash.



Gambar 4.33 Halaman Git Bash

4. Ketikkan perintah berikut :

   cd /d/xampp/htdocs/apartemen/

   php artisan serve

5. Buka browser, gunakan mozzila firefox lalu ketikkan alamat http://localhost:8000/

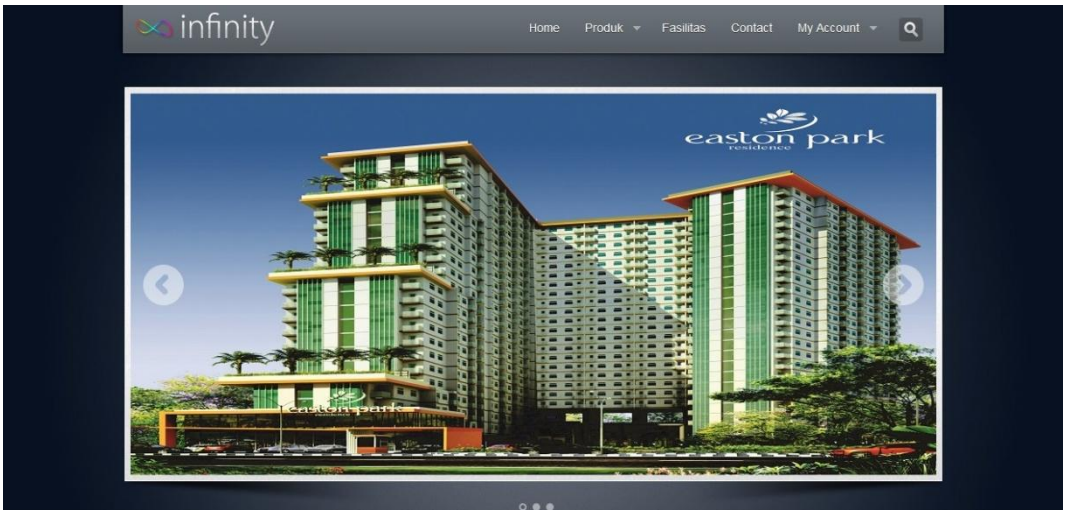
4.4.7 Penggunaan Program

Penggunaan program bertujuan untuk menerangkan cara penggunaan atau cara user menggunakan fungsionalitas dari Sistem Informasi Pemasaran dan Pemesanan Apartemen Easton Park di CV.AMADDA. Penjelasan penggunaan program ini akan disesuaikan berdasarkan hak akses. Adapun berikut ini adalah cara penggunaan program Sistem Informasi Pemasaran dan Pemesanan Apartemen Easton Park di CV.AMADDA :

4.4.7.1 Penggunaan Program Pengunjung

1. Home

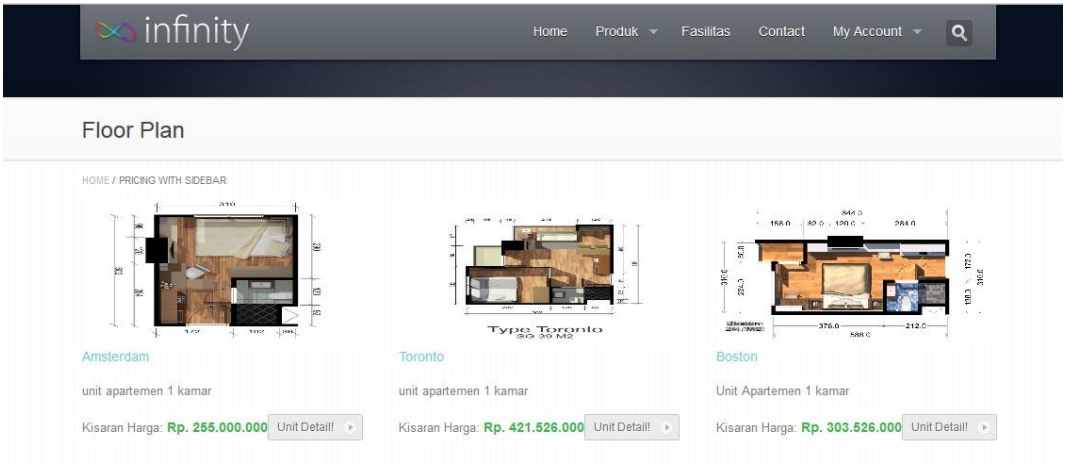
   Home merupakan tampilan awal pada saat membuka website



Gambar 4.34 Halaman Home

2. Produk

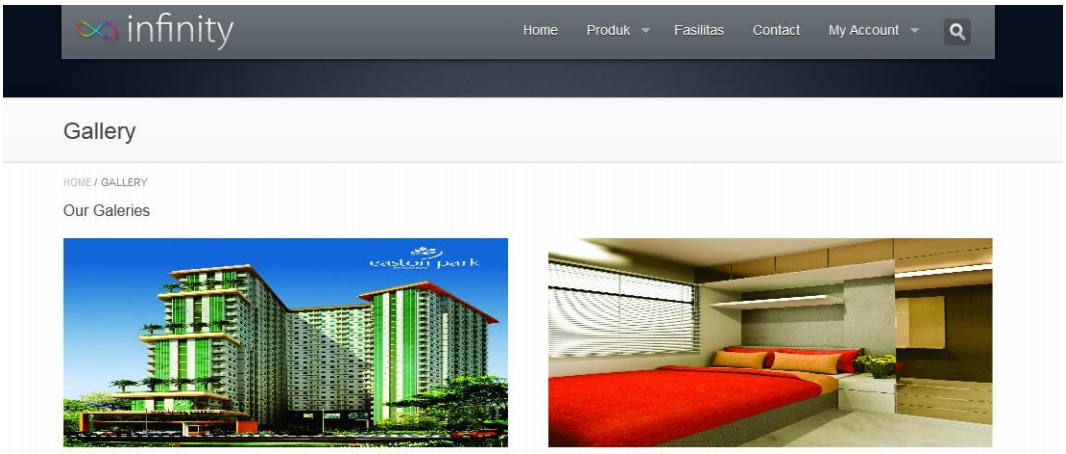
   Di menu produk, pengunjung dapat melihat tipe apartemen Easton Park yang tersedia.



Gambar 4.35 Halaman Produk

3. Fasilitas

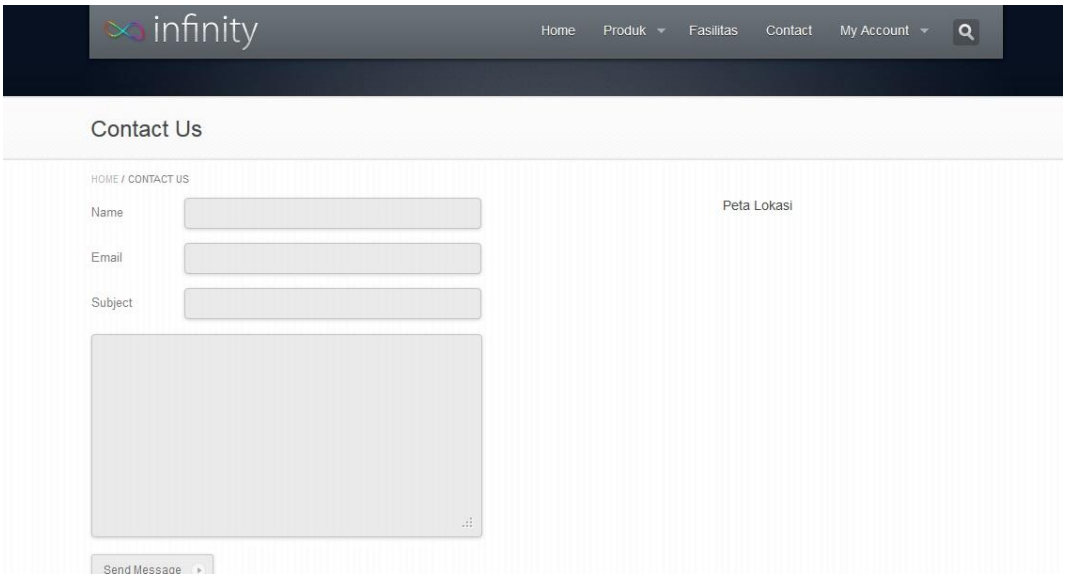
   Di menu Fasilitas, pengunjung dapat melihat fasilitas apa saja yang ditawarkan Easton Park.



Gambar 4.36 Halaman Fasilitas

4. Contact

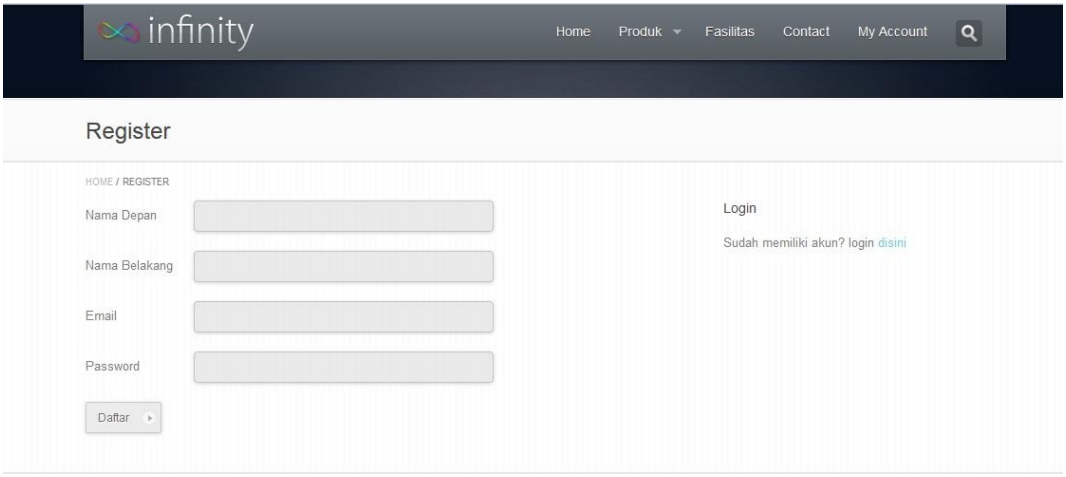
   Menulis menambahkan menu contact, untuk mempermudah interaksi dengan pengunjung. Pesan yang dikirim pengunjung akan langsung masuk ke email admin (manager pemasaran)



Gambar 4.37 Halaman Contact

5. My account

   Halaman My account dibagi dua form, Login dan registrasi. Pengunjung tidak dapat melakukan login. Untuk itu pengunjung harus melakukan registrasi.

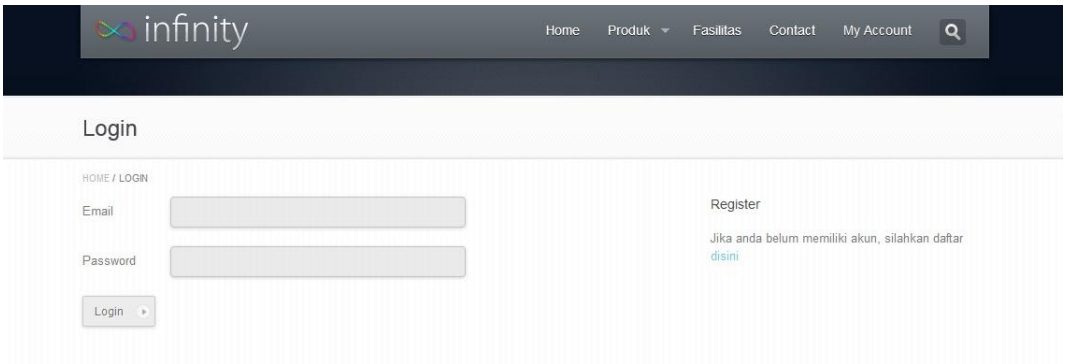


Gambar 4.38 Halaman Register

4.4.7.2 Penggunaan Program Pelanggan

1. Login

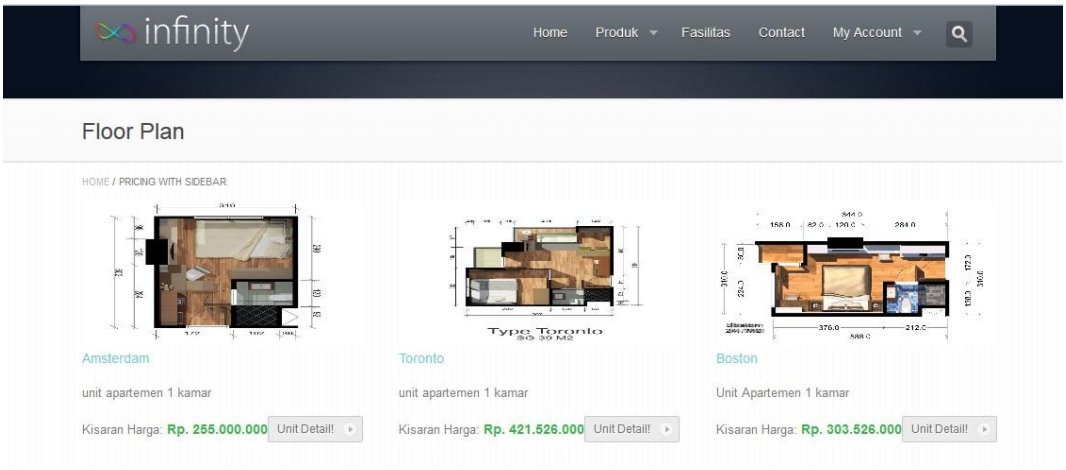
   Pelanggan memasukkan email dan password sesuai dengan form pendaftaran.



Gambar 4.39 Halaman Login Pelanggan

2. Produk

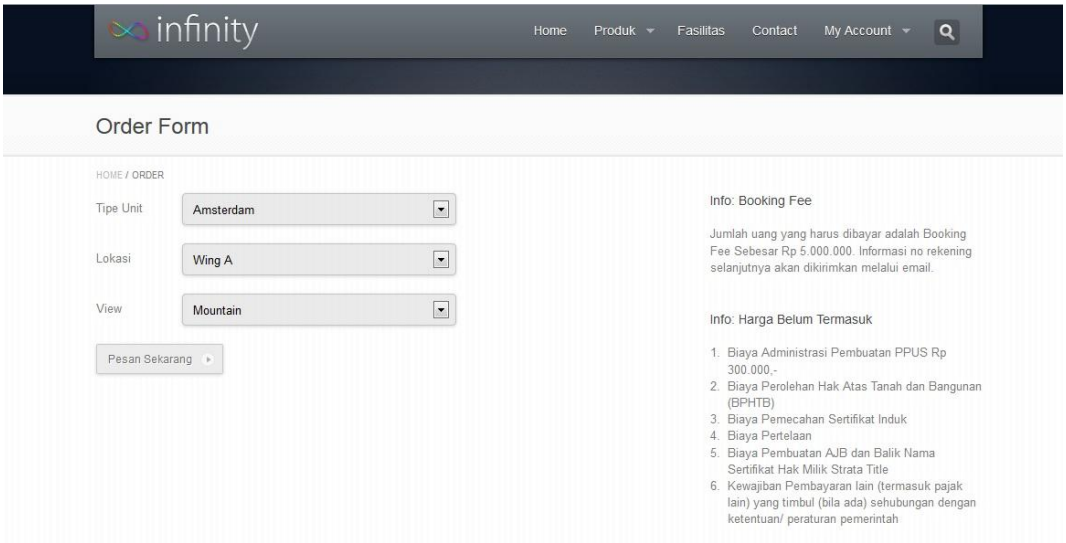
   Pelanggan dapat melakukan transaksi pemesanan dengan memilih unit yang akan di book.



Gambar 4.40 Halaman Jenis Produk

3. Order form

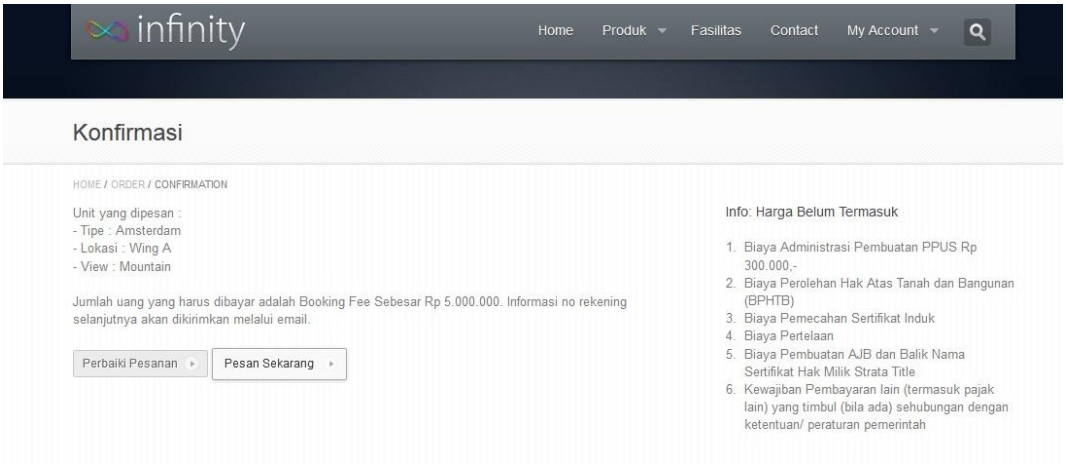
Setelah pelanggan memilih unit yang akan di booking. Maka pelanggan akan memasuki halaman order form untuk melihat list dari unit yang akan di pesan.



Gambar 4.41 Halaman Order Form

4. Konfirmasi

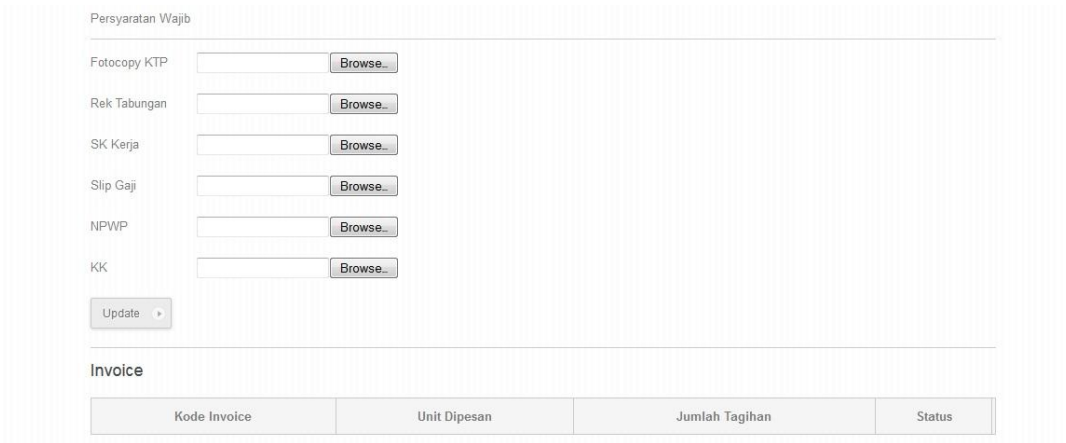
Setelah melihat list dari order form, kemudian pelanggan akan memasuki halaman konfirmasi untuk memastikan unit yang akan dipesan.



Gambar 4.42 Halaman Konfirmasi

5. Persyaratan

Untuk melakukan pemesanan, pelanggan harus melengkapi syarat-syarat pemesanan. Ketika syarat-syarat lengkap dan sudah melakukan pembayaran booking fee maka manager pemasaran akan menyatakan status unit yang di booking.

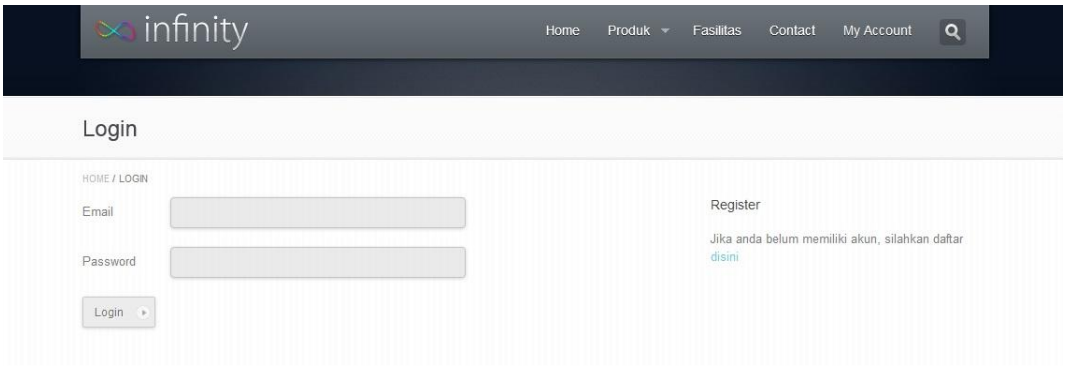


Gambar 4.43 Halaman Persyaratan

4.4.7.3 Penggunaan Program Admin

1. Login

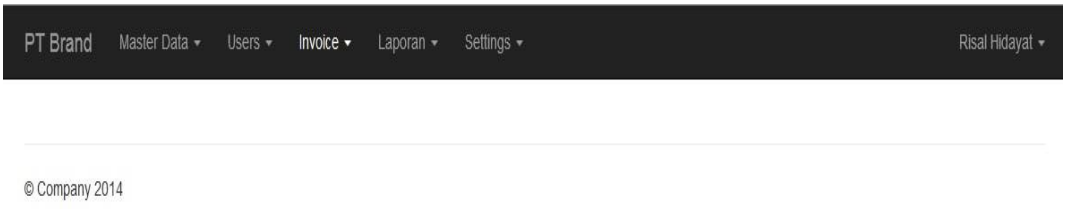
   Admin melakukan input email dan password.



Gambar 4.44 Halaman Login admin

2. Menu admin

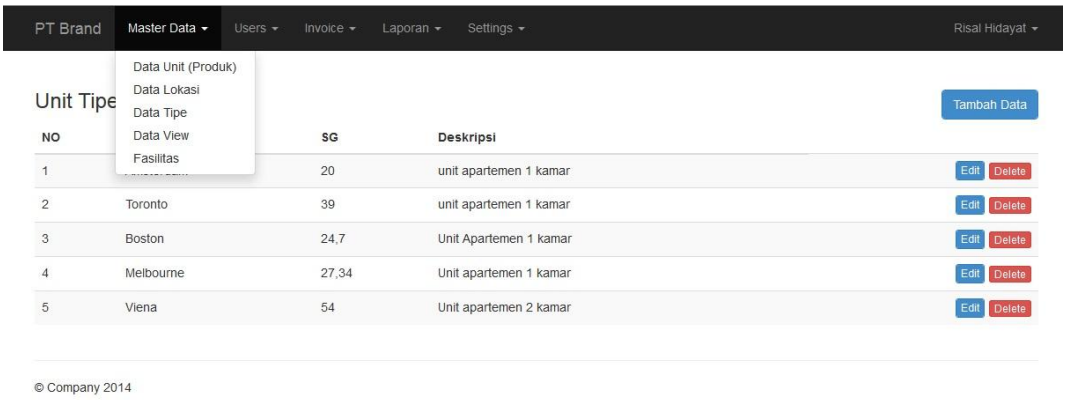
   Admin memasuki halaman admin setelah melakukan login



Gambar 4.45 Halaman menu utama admin

3. Master

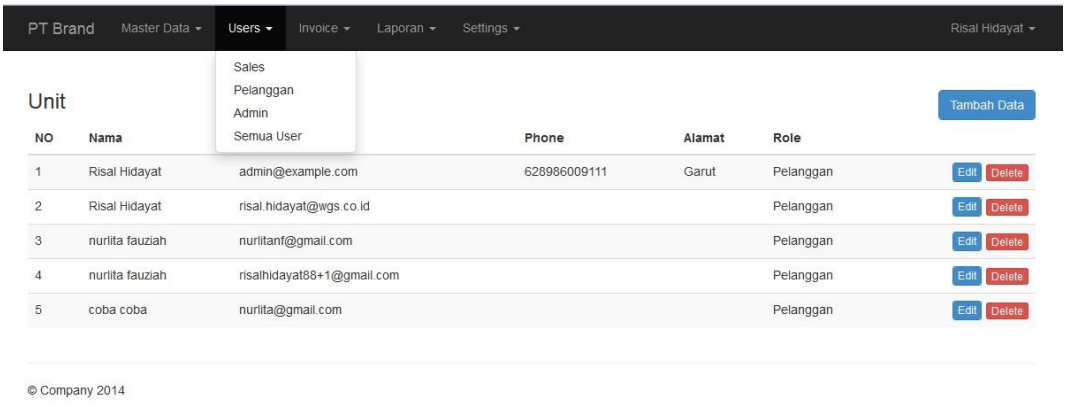
Manager pemasaran dapat mengolah data master seperti data unit, data lokasi, data tipe, data view, dan fasilitas.



Gambar 4.46 Halaman Master

4. Users

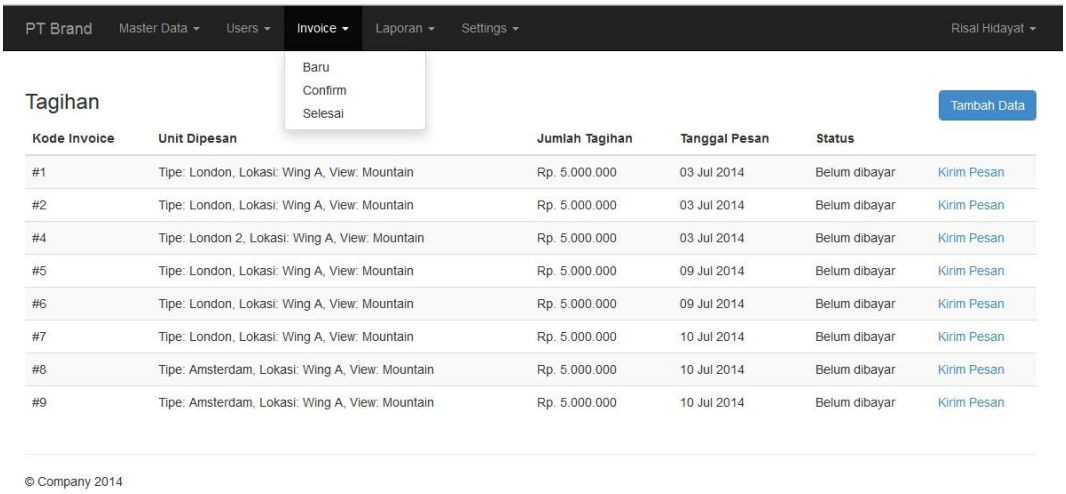
Manager pemasaran dapat mengolah data user dari mulai pelanggan, sales, dan admin.



Gambar 4.47 Halaman users

5. Invoice

Manager pemasaran dapat mengolah data invoice.



Gambar 4.48 Halaman Invoice

4.5 Pengujian

Pengujian perangkat lunak adalah proses untuk mencari kesalahan pada setiap item perangkat lunak, mencatat hasilnya, mengevaluasi setiap aspek pada setiap komponen sistem dan mengevaluasi semua fasilitas dari perangkat lunak yang dikembangkan.

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Tujuan dari pengujian adalah untuk menjamin perangkat lunak yang di bangun memiliki kualitas yang handal, yaitu mampu mempresentasikan kajian pokok dari spesifikasi, analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri.

4.5.1 Rencana Pengujian

Pengujian perangkat lunak Sistem Informasi Pemasaran dan Pemesanan ini menggunakan metode black box. Pengujian black box berfokus pada persyaratan fungsional perangkat lunak yang dibuat. Berikut ini adalah hasil dari

pengujian dari Sistem Informasi Pemasaran dan Pemesanan Apartemen Easton Park di CV. AMADDA Berbasis Web :

4.5.1.1 Rencana Pengujian Halaman Admin

Tabel 4.13 Rencana Pengujian Halaman Admin

Kelas Uji	**Butir Uji**	**Metode Pengujian**
Login	Login berhasil : email, password yang dimasukan terdaftar di database.	Black box
	Login gagal : email, password yang dimasukan tidak terdaftar di database	Black box
Pengujian pengolahan data master unit	Proses tambah, ubah, hapus data master	Black box
Pengujian laporan pemesanan dan unit tersedia	Cetak laporan pemesanan dan unit tersedia	Black box

4.5.1.2 Rencana Pengujian Halaman Utama

Tabel 4.14 Rencana Pengujian Halaman Utama

Kelas Uji	Butir Uji	Metode Pengujian
Menu konfirmasi produk	Proses pemesanan berhasil	Black box
Registrasi	Proses registrasi valid dan tersimpan ke database	Black box

4.5.1.3 Rencana Pengujian Halaman Pelanggan

Tabel 4.15 *Rencana Pengujian* Halaman Pelanggan

Kelas Uji	Butir Uji	Metode
Login Pelanggan	Login berhasil : email, password yang dimasukan terdaftar di database.	Black box
	Login gagal : email, password yang dimasukan tidak terdaftar di database	Black box
Profil	Menampilkan form my account	Black box
	Update data pelanggan	
	Simpan perubahan	

4.5.2 Kasus dan Hasil Pengujian

Kasus dan hasil pengujian dibuat untuk mengetahui apakah Sistem Informasi Pemasaran dan Pemesanan Apartemen Easton Park di CV. AMADDA Berbasis Web dapat berjalan dengan baik atau tidak.

4.5.2.1 Kelas Uji Halaman Admin

4.5.2.1.1 Pengujian Login Admin

Tabel 4.16 *Pengujian* Halaman Login Admin

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : risalhidayat88@gmail.com Password : 12345678	Dapat login dan masuk ke halaman utama admin	Dapat login dan masuk ke halaman utama admin	[X] Diterima [ ] Ditolak
Login	Dapat masuk ke halaman utama admin	Tombol login dapat berfungsi sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : bebas Password : bebas	Tidak dapat login dan menampilkan pesan kesalahan	User tidak dapat login jika salah email dan password	[X] Diterima [ ] Ditolak

4.5.2.1.2 Pengujian Data Master Unit

Tabel 4.17 *Pengujian* Data Master Unit

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tipe : Toronto Lokasi : Wing A View : Mountain SG : 20	Data Berhasil disimpan ke database	Data berhasil disimpan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tipe : <kosong> Lokasi : <kosong> View : <kosong> SG : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan “the field is required”	[X] Diterima [ ] Ditolak

4.5.2.1.3 Pengujian Laporan

Tabel 4.18 *Pengujian* Laporan

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
No Unit : a0101b Tipe : Toronto Lokasi : Wing A View : Mountain	Data Berhasil disimpan ke database	Data berhasil disimpan	[X] Diterima [ ] Ditolak

SG : 20 Status : Tersedia			
Cetak Laporan	Dapat mencetak Laporan	Tombol cetak berfungsi	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
No unit :<kosong> Tipe : <kosong> Lokasi : <kosong> View : <kosong> SG : <kosong> Status: <kosong>	Data tidak tersimpan	Sistem menampilkan pesan “the no unit field is required”	[X] Diterima [ ] Ditolak

4.5.2.2 Kelas Uji Halaman Utama

4.5.2.2.1 Menu Konfirmasi Produk

Tabel 4.19 *Pengujian* Menu Konfirmasi Produk

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tipe Unit : Amsterdam Lokasi : Wing A	Data Berhasil tersimpan dan tampil di account	Data berhasil disimpan	[X] Diterima [ ] Ditolak

View : Mountain	pelanggan		
Perbaiki Pesanan	Dapat mengedit data pesanan	Tombol perbaiki pesanan berfungsi	[X] Diterima [ ] Ditolak
Pesan Sekarang	Tampil halaman account pelanggan	Tombol pesan sekarang berfungsi	[X] Diterima [ ] Ditolak

4.5.2.2.2 Registrasi

Tabel 4.20 *Pengujian* Registrasi

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama depan : nurlita Nama belakang : fauziah Email : nurlitanf@gmail.com Password : 12345	Data Berhasil disimpan ke database	Data berhasil disimpan	[X] Diterima [ ] Ditolak

Daftar	Tampil halaman utama pelanggan	Tombol daftar berfungsi	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama depan : <kosong> Nama belakang : <kosong> Email : <kosong> Password : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan “the field is required”	[X] Diterima [ ] Ditolak

4.5.2.3 Kelas Uji Halaman Pelanggan

4.5.2.3.1 Login Pelanggan

Tabel 4.21 *Pengujian* Login Pelanggan

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : nurlitanf@gmail.com Password : 12345	Dapat login dan masuk ke halaman utama pelanggan	Dapat login dan masuk ke halaman utama pelanggan	[X] Diterima [ ] Ditolak

Login	Dapat masuk ke halaman utama pelanggan	Tombol login dapat berfungsi sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : bebas Password : bebas	Tidak dapat login dan menampilkan pesan kesalahan	User tidak dapat login jika salah email dan atau password	[X] Diterima [ ] Ditolak

4.5.2.3.2 Profil Pelanggan

Tabel 4.22 *Pengujian* Profil Pelanggan

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : nurlitanf@gmail.com Nama depan : nurlita Nama belakang : fauziah Alamat : Komp.puteraco	Data Berhasil disimpan ke database	Data berhasil disimpan	[X] Diterima [ ] Ditolak

No tlp : 0899682868			
Update	Tampil halaman utama pelanggan	Tombol Update berfungsi	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masuk	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : nurlitanf@gmail.com Nama depan : <kosong> Nama belakang : fauziah Alamat : Komp.puteraco No tlp : 0899682868	Data tidak tersimpan	Sistem menampilkan pesan “the nama field is required”	[X] Diterima [ ] Ditolak

4.5.3 Kesimpulan Hasil Pengujian

Berdasarkan rencana Pengujian diatas penulis dapat mengambil kesimpulan mengenai bagaimana cara pengujian Sistem Informasi yang dibangun oleh penulis. Penulis dapat memperbaiki kesalahan-kesalahan pada saat Sistem Informasi telah di Implementasikan di CV.AMADDA.