BAB IV

HASIL DAN PEMBAHASAN

4.1. Perancangan Sistem

Perancangan sistem merupakan tahap selanjutnya dari fase pembuatan sistem informasi yang diusulkan setelah mengetahui bagaimana kondisi sistem yang sedang berjalan untuk memperbaiki kelemahan yang ada didalam sistem tersebut.

Pada perancangan sistem ini penulis mempersiapkan pembangunan sistem informasi zoom petcare bandung berbasis web .

4.1.1. Tujuan perancangan sistem

Tujuan dari perancangan sistem ini adalah untuk memberikan gambaran secara umum kepada pemakai mengenai sistem informasi yang baru. Sistem informasi baru yang dirancang bertujuan untuk mempermudah dan mempercepat proses aktivitas pekerjaan pada Zoom Petcare.

Adapun tujuan lain dari perancangan sistem informasi ini adalah sebagai berikut :

1. Merubah proses manual menjadi terkomputerisasi dengan adanya sistem informasi yang akan dibangun.

2. Merubah penyimpanan dokumen yang masih menggunakan kertas dan buku menjadi tersimpan dalam database sistem.

3. Memudahkan pencarian data pasien dan data rekam medis pasien karena data tersimpan dalam database sistem.

4. Mempercepat pembuatan laporan yang dapat langsung dilihat dan dicetak tanpa harus merekap data dengan mencatatnya secara manual.

4.1.2. Gambaran umum sistem yang diusulkan

Gambaran umum ini menjelaskan secara singkat mengenai sistem yang akan diusulkan yang akan mempresentasikan seluruh bentuk bagian yang ada di dalamnya dan akan memperjelas perbedaan antara sistem yang sedang berjalan dengan sistem yang diusulkan.

Untuk asumsi prosedur awal, data pasien, tindakan, dan obat di aplikasi ini dianggap telah ada. Dimana sebelumnya resepsionis meng*input* data pasien sebagai proses pendaftaran, kartu rekam medis pasien, data tindakan dan data obat.

Jadi sistem yang diusulkan merupakan sebuah aplikasi perangkat lunak yaitu Sistem Informasi Pelayanan Pelanggan yang menyediakan fitur pengolahan data dimulai dari pendaftaran pasien menjadi kartu rekam medis pasien, setelah kartu rekam medis pasien terisi bisa dibuatkan nota pada pasien tersebut. Sehinggga dengan adanya aplikasi Sistem informasi yang diusulkan ini resepsionis tidak perlu mencatat manual dan menghitung pendapatan secara manual lagi.

4.1.3. Perancangan prosedur yang diusulkan

Perancangan prosedur yang diusulkan adalah jika sistem yang lama masih menggunakan cara manual dan dokumen atau data-data berbentuk buku, maka di sistem informasi yang akan dirancang akan memaksimalkan alat bantu komputer yang ada.

Berikut adalah perancangan prosedur yang diusulkan dalam perancangan:

1. **Prosedur pendaftaran**

   Pasien datang ke bagian resepsionis untuk melakukan pendaftaran, petugas akan meminta identitas pasien yang akan di *input*kan ke dalam sistem untuk dibuatkan kartu pasien.

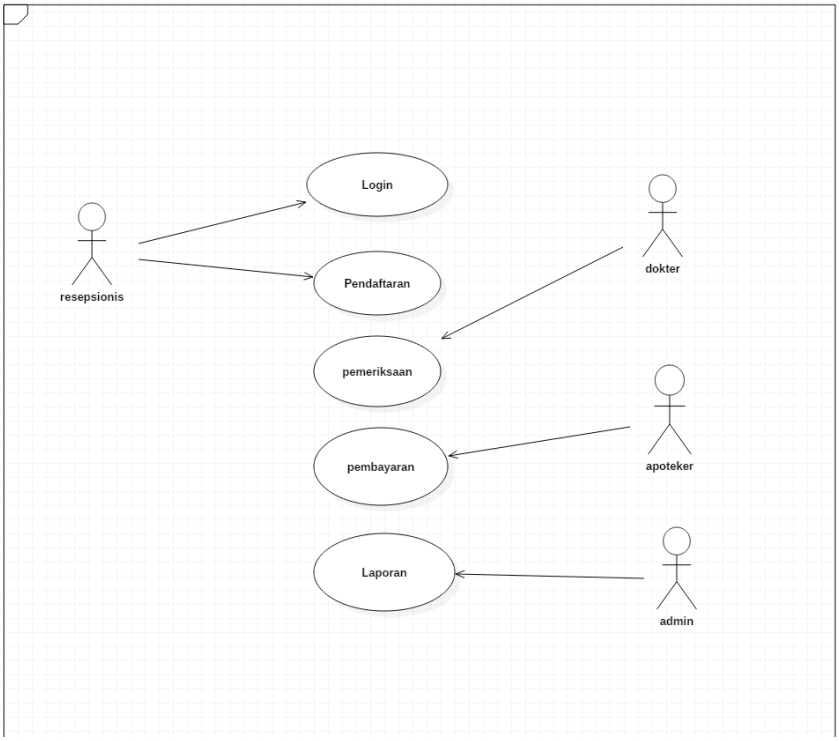
2. **Prosedur pemeriksaan dan pencatatan rekam medis pasien**

Dokter memeriksa dan mendengarkan keluhan pasien, lalu dokter meng*input*kan diagnosa, keluhan, resep obat dan tindakan ke dalam sistem sebagai proses rekam medis.

3 Prosedur pembayaran dan pengambilan obat

Pasien mendatangi bagian apoteker dan memberikan kartu pasien. Apoteker meng*input*kan nomer pendaftaran pasien lalu memberikan total biaya obat yang harus di bayar sesuai resep yang diberikan oleh dokter.Setelah membayar , pasien akan di berikan struk pembayaran obat beserta obat yang telah di bayar.

4.1.3.1. *Usecase diagram*



Gambar 4.1 *Usecase diagram* Yang Diusulkan

a Definisi aktor dan deskripsinya

Tabel 4.1 Definisi Aktor dan Deskripsinya

NO	Aktor	Deskripsi
1	Resepsionis	Orang yang memiliki akses dan bertugas melakakukan pengolahan data pendaftaran.
2	Dokter	Orang yang memiliki akses dan bertugas mencatat data rekam medis pasien sesuai dengan diagnosa/pemeriksaan yang dilakukan lalu mencatat tindakan dan resep obat, dan membuat laporan.
3	Apoteker	Orang yang memiliki akses dan bertugas melakukan pengolahan data obat, dan pembayaran

4	Admin	Orang yang memiliki akses dan bertugas dalam melakukan pengolahan data pasien,data tindakan,data penyakit,data pegawai dan laporan

b Definisi *usecase* dan deskripsinya

Tabel 4.2 Definisi *Usecase* dan Deskripsinya

NO	*Usecase*	Deskripsi
1	*Login*	Merupakan proses dimana aktor melakukan *login* agar bisa mendapatkan akses pada aplikasi.
2	Pendaftaran	Merupakan proses dimana resepsionis melakukan pencatatan data pasien untuk pendaftaran dan mencetak kartu pasien dan kartu pendaftaran.
3	Pemeriksaan	Merupakan proses dimana dokter mencatat hasil dari diagnosa dan keluhan dari pasien serta mencatat tindakan dan resep obat yang akan di berikan kepada pasien
4	Pembayaran	Merupakan proses dimana resepsionis menghitung total biaya dari tindakan dan obat yang diberikan kepada pasien.
5	Laporan	Merupakan proses dimana aktor membuat laporan dari tiap proses yang dilakukan.

4.1.3.2. Skenario usecase

Sekenario *usecase* mendeskripsikan aktor-aktor yang melakukan prosedur dalam sistem dan menjelaskan respon yang ditanggapi oleh sistem tersebut terhadap prosedur yang dilakukan oleh aktor. Berikut adalah sekenario *usecase* nya :

1 Skenario *Usecase Login*

Tabel 4.3 Skenario *Usecase Login*

Identifikasi	
Nomor	1
Nama	*Login*
Tujuan	Mendapatkan akses dari aplikasi
Aktor	Resepsionis, Dokter, Apoteker,Admin
Skenario Utama	
Aktor	Reaksi sistem
	1 Menampilkan menu *login*
2 Aktor memasukan *username* dan *password*.	
	3 Validasi data *login* dengan data *username* dan *password* yang ada di

	database.
	4 Masuk ke halaman utama sesuai akses yang di berikan.
Skenario Alternatif (*username* dan *password* salah)	
	1 Menampilkan menu *login*
2 Aktor memasukan *username* dan *password*.	
	3 Validasi data *login* dengan data *username* dan *password* yang ada di database.
	4 Menampilkan pesan kesalahan *username* atau *password* salah.
5 Aktor memasukan *username* dan *password*.	
	6 Masuk ke halaman utama sesuai akses yang di berikan.

2 Skenario *Usecase* melakukan pendaftaran

Tabel 4.4 Skenario *Usecase* Melakukan Pendaftaran

Identifikasi	
Nomor	2
Nama	Melakukan Pendaftaran
Tujuan	Men*input*kan data pasien baru, mendaftarkan pasien dan mencetak kartu pasien
Aktor	Resepsionis
Skenario Utama	
Aktor	Reaksi sistem

1 Memilih menu pasien lama	
	2 Menampilkan form cari pasien.
3 Masukkan nama pasien atau id pasien yang akan di id pasien yang akan di daftarkan.	
	4 Menampilkan nama atau id pasien yang terekam oleh database.
5 Pilih pasien yang akan didaftarkan.	
6 Pilih tombol simpan di form pendaftaran.	
Skenario Alternatif (pasien baru)	
1 Memilih tombol pasien baru.	2 Menampilkan form pasien baru.
3 Masukan identitas pasien yang akan didaftarkan.	
4 Pilih menu simpan di form pasien baru.	
	5 Menyimpan identitas pasien dan mencetak kartu pasien.
6 Pilih tombol simpan di	

form pendaftaran.	

3 Skenario *Usecase* Pemeriksaan

Tabel 4.5 Skenario *Usecase* Pemeriksaan

Identifikasi	
Nomor	3
Nama	Pencatatan Rekam Medis
Tujuan	Mencatat hasil diagnosa, keluhan, tindakan , dan resep obat yang diberikan
Aktor	Dokter
Skenario Utama	
Aktor	Reaksi sistem
	1 Menampilkan list pasien yang mendaftar
2 Memilih pasien yang akan di diagnosa berdasarkan nomer daftar.	

	3 Menampilkan form pemeriksaan,diagnose, kode penyakit, tindakan.
4 Meng*input*kan data keluhan, diagnosa, tindakanan yang akan diberikan, serta resep obat.	
5 Menyimpan data diagnosa.	
	6 Menyimpan data diagnosa kedalam database.

4 Skenario *Usecase* Pembayaran

Tabel 4.6 Skenario *Usecase* Pembayaran

Identifikasi	
Nomor	4
Nama	Pembayaran
Tujuan	Melakukan perhitungan total biaya dari tindakan dan obat yang diberikan
Aktor	Apoteker
Skenario Utama	

Aktor	Reaksi sistem
	1 Menampilkan list pasien yang selesai di periksa.
2 Memilih pasien.	
	3 Menampilkan data yang diberikan beserta total biaya pembayarannya.
4 Mencetak nota pembayaran.	
5 Memilih tombol pembayaran	
	6 Menyimpan data di database

5. **Skenario *Usecase* Laporan**

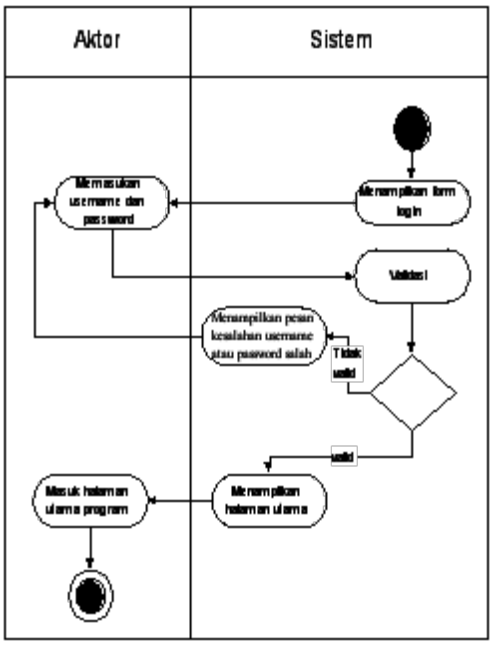
Tabel 4.7 Skenario *Usecase* Laporan

Identifikasi	
Nomor	5
Nama	Laporan
Tujuan	Menjelaskan proses pembuatan laporan
Aktor	Admin

Skenario Utama	
Aktor	Reaksi sistem
1 Memilih menu laporan	
2 Memilih tanggal laporan	
	3 Menampilkan data laporan
4 Mencetak Laporan	

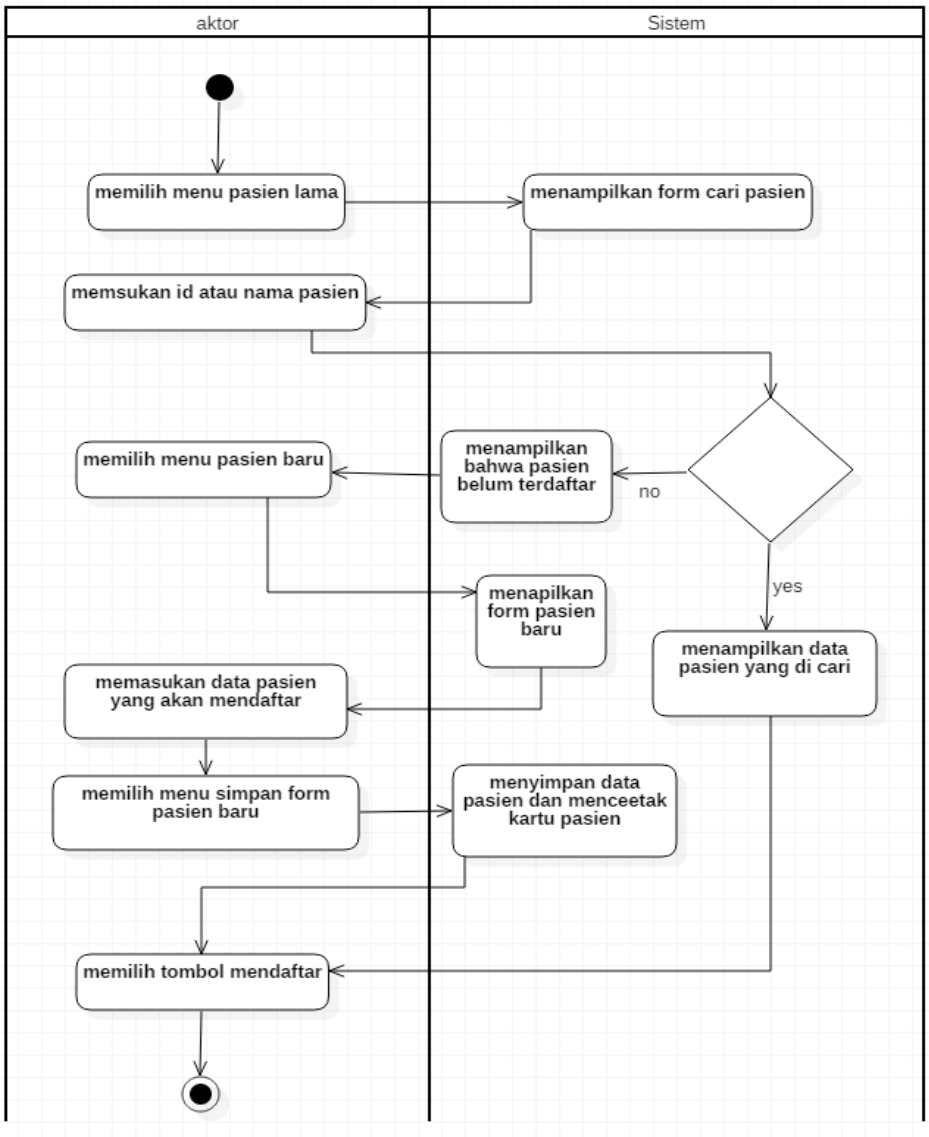
4.1.3.3. Activity diagram

Activity diagram menggambarkan berbagai alur setiap aktifitas yang terdapat dalam sistem yang sedang di rancang. Dari bagaimana alur masing-masing berawal, terdapatnya decision dalam aktifitas, sampai berakhirnya aktifitas tersebut. Adanya *activity diagram* sangat diperlukan agar pengguna sistem dapat mengetahui setiap kegiatan yang terdapat dalam sistem.

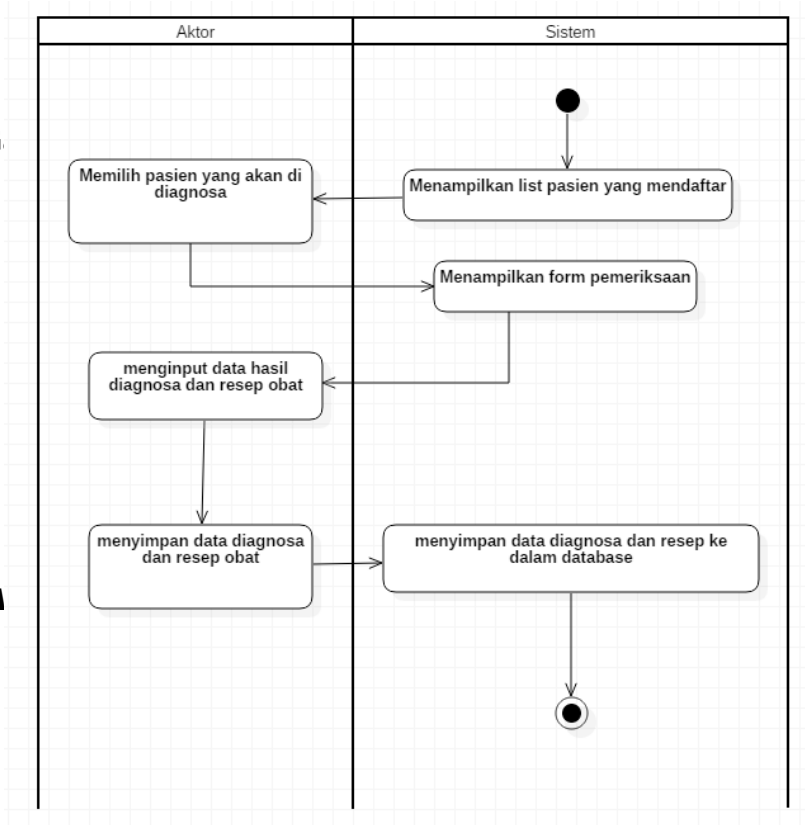


1 ***Activity diagram Login***

Gambar 4.2 ***Activity diagram Login***


aktor
Sistem
memilih menu pasien lama
menampilkan form cari pasien
memsukan id atau nama pasien
memilih menu pasien baru
menampilkan bahwa pasien belum terdaftar
no
yes
menapilkan form pasien baru
menampilkan data pasien yang di cari
memasukan data pasien yang akan mendaftar
memilih menu simpan form pasien baru
menyimpan data pasien dan menceetak kartu pasien
memilih tombol mendaftar

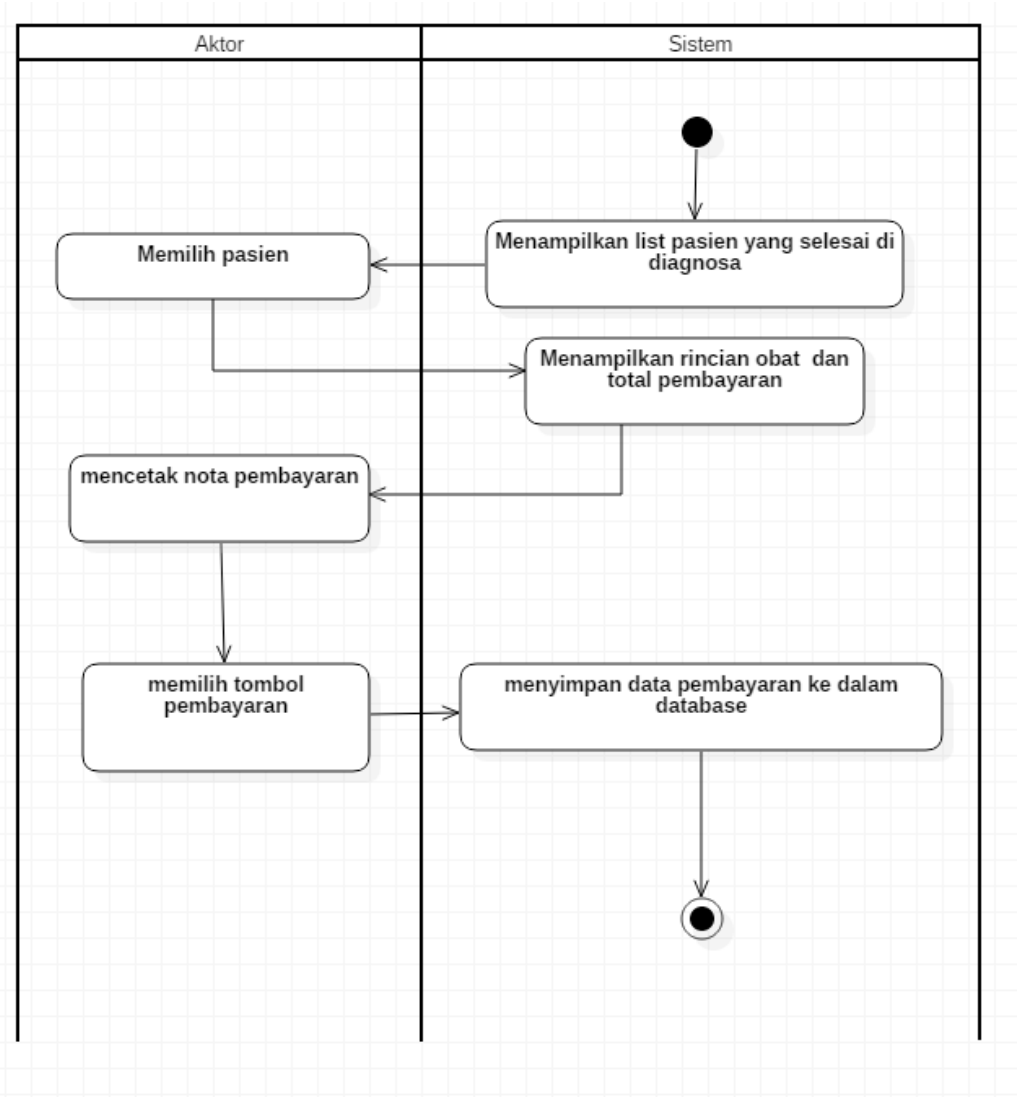
G ran


Aktor
Sistem
Memilih pasien yang akan di diagnosa
Menampilkan list pasien yang mendaftar
Menampilkan form pemeriksaan
menginput data hasil diagnosa dan resep obat
menyimpan data diagnosa dan resep obat
menyimpan data diagnosa dan resep ke dalam database

3 *A*

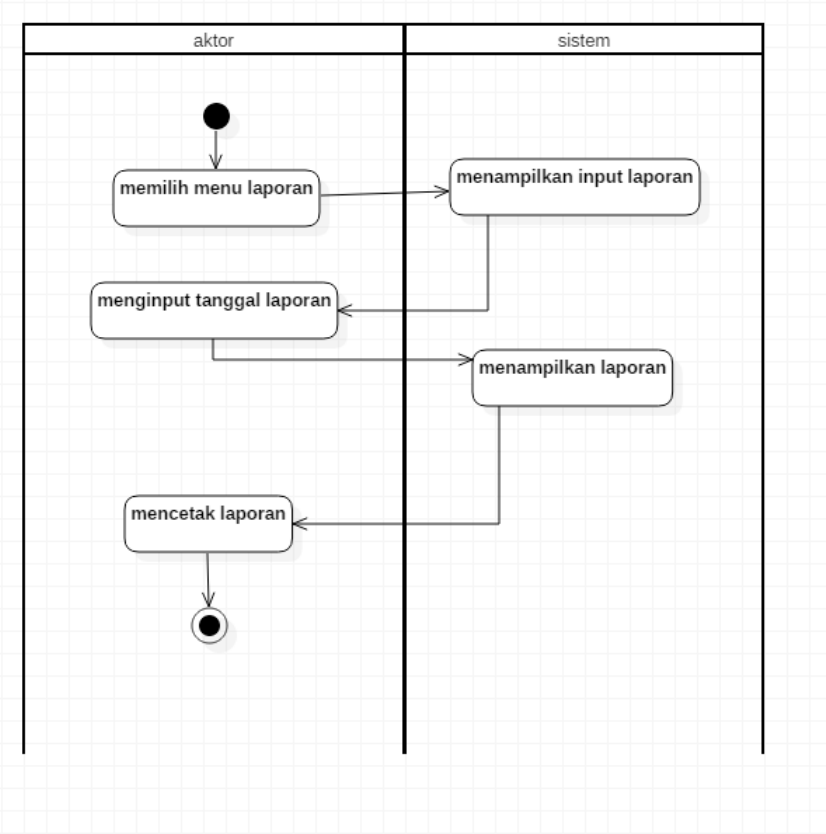
Gambar 4.4 *Activity diagram* Pencatatan Rekam Medis

4 *Activity diagram* Pembayaran



Gambar 4.5 *Activity diagram* Pembayaran Tindakan

5 *Activity diagram* Laporan

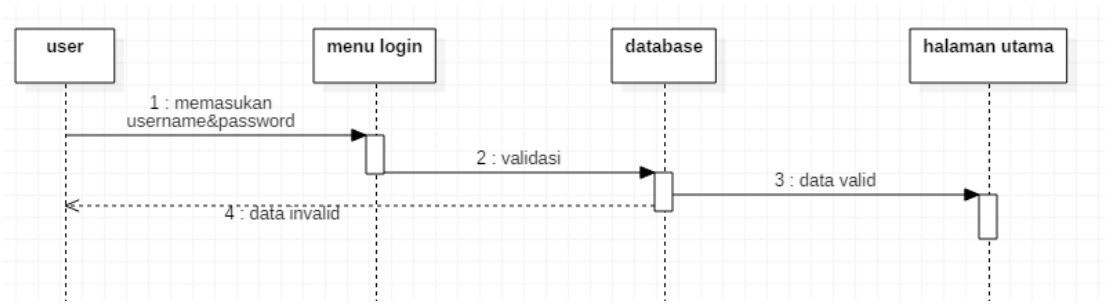


Gambar 4.6 *Activity diagram* Cetak Laporan

4.1.3.4. Sequence diagram

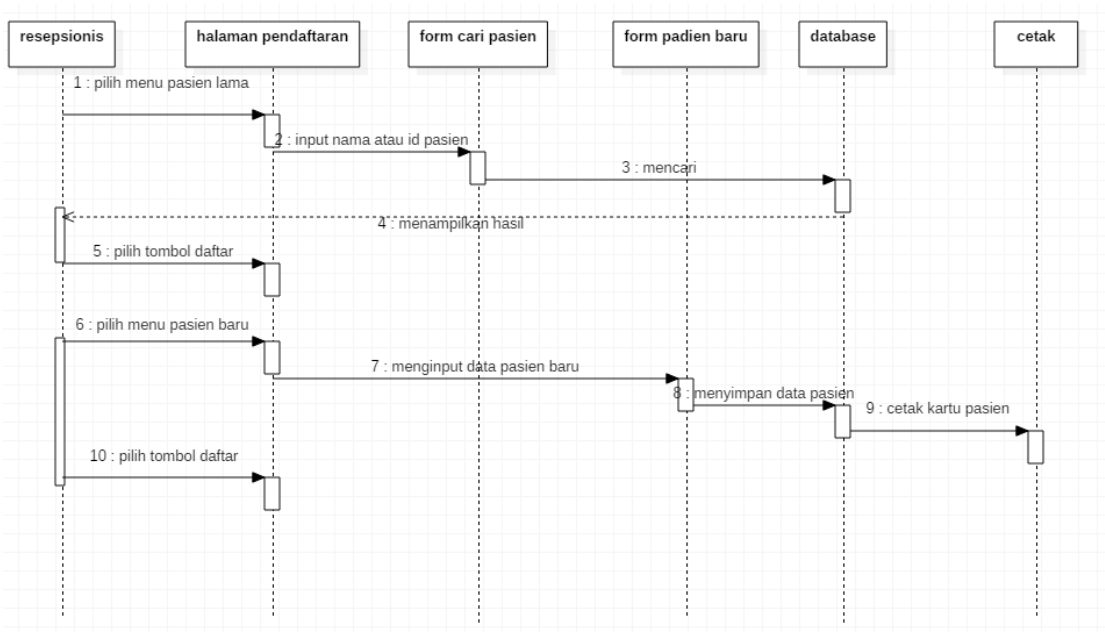
Sequence diagram menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem.

1 *Sequence diagram Login*



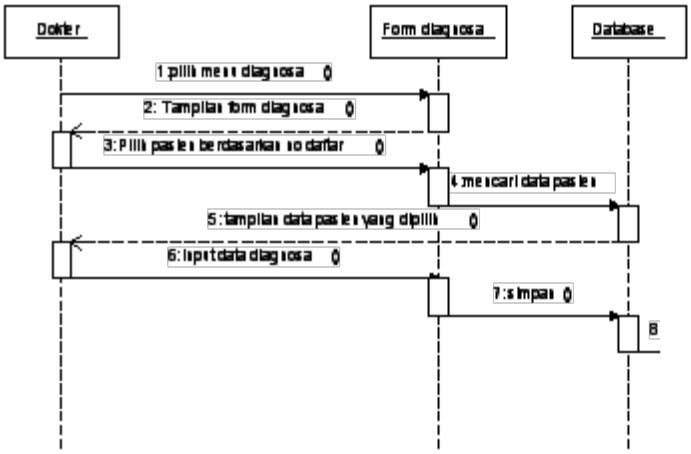
Gambar 4.7 *Sequence diagram Login*

2 *Sequence diagram* Melakukan Pendaftaran



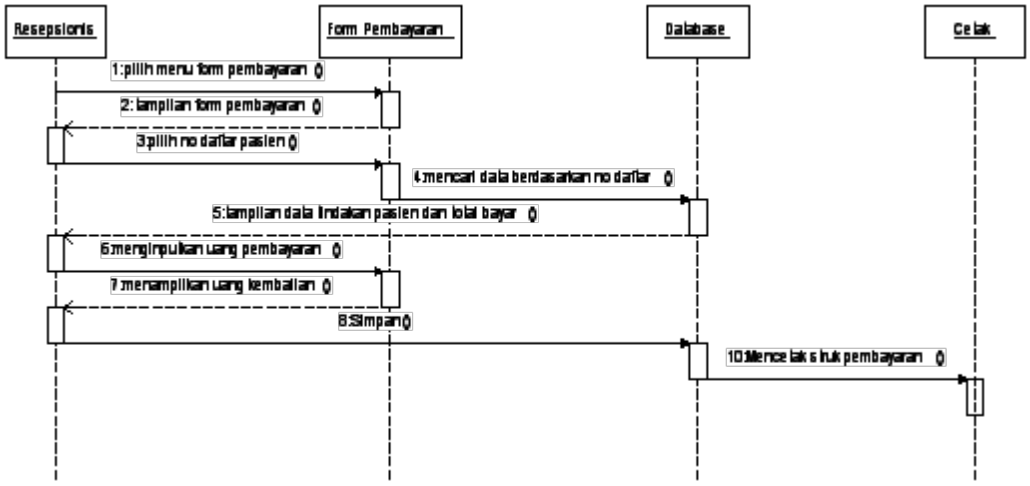
Gambar 4.8 *Sequence diagram* Melakukan Pendaftaran

3 *Sequence diagram* Pencatatan wdRekam Medis



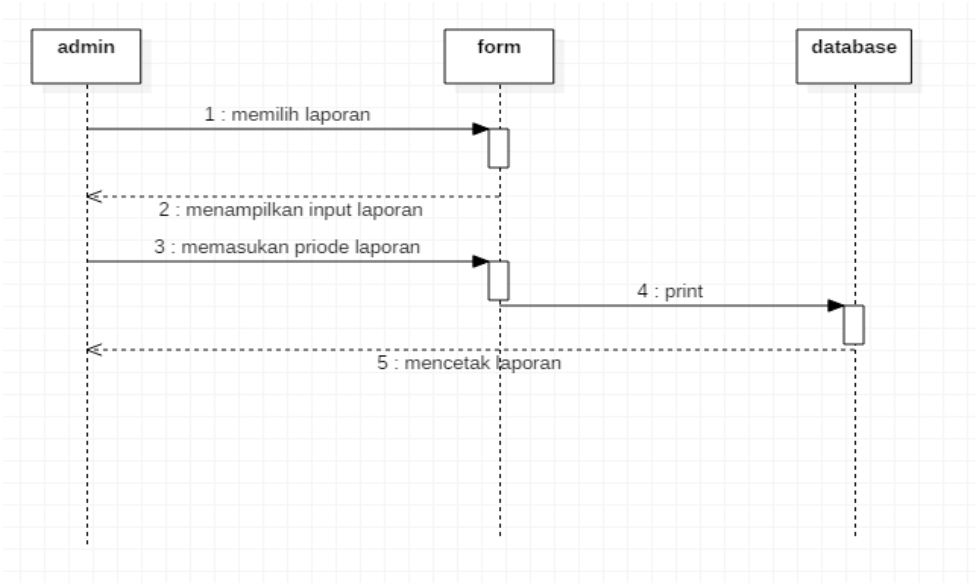
Gambar 4.9 *Sequence diagram* Pencatatan Rekam Medis

4 *Sequence diagram* Pembayaran



Gambar 4.10 *Sequence diagram* Pembayaran Tindakan

5 *Sequence diagram* Laporan

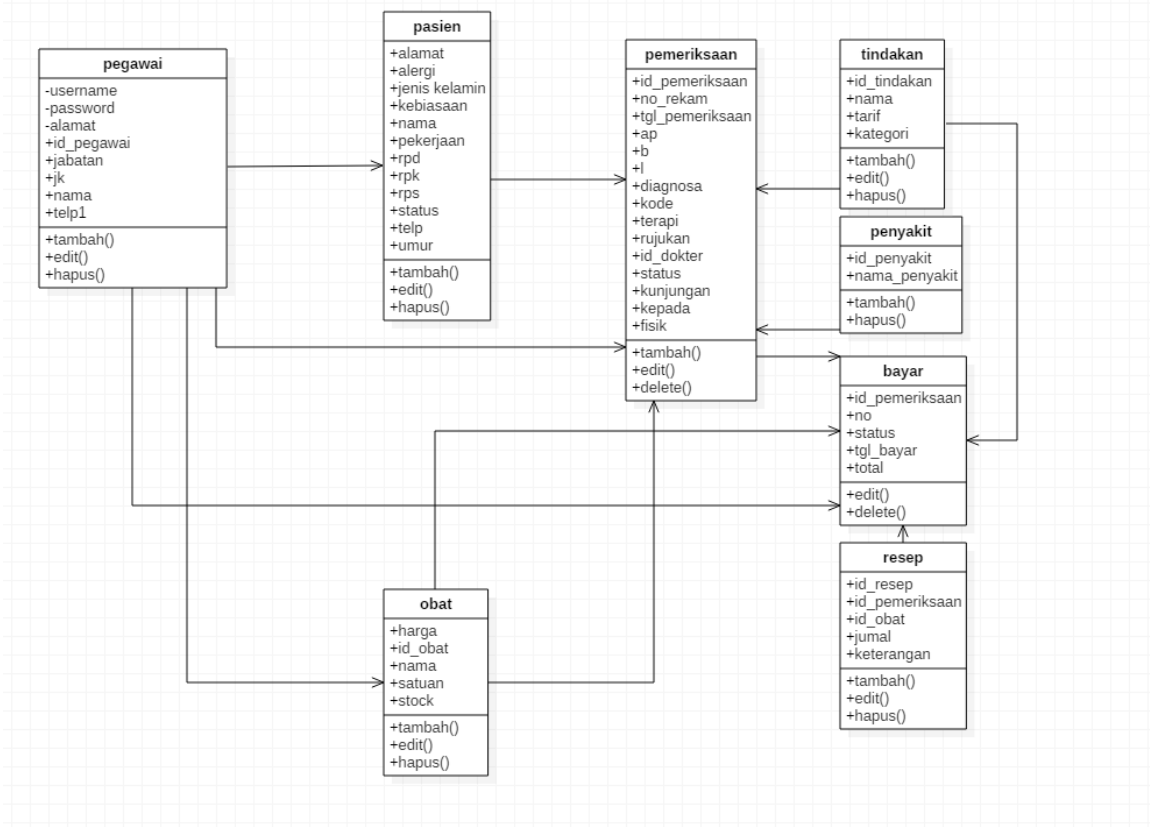


Gambar 4.11 *Sequence diagram* Cetak Laporan

4.1.4. Perancangan basis data

4.1.4.1. *Class diagram*

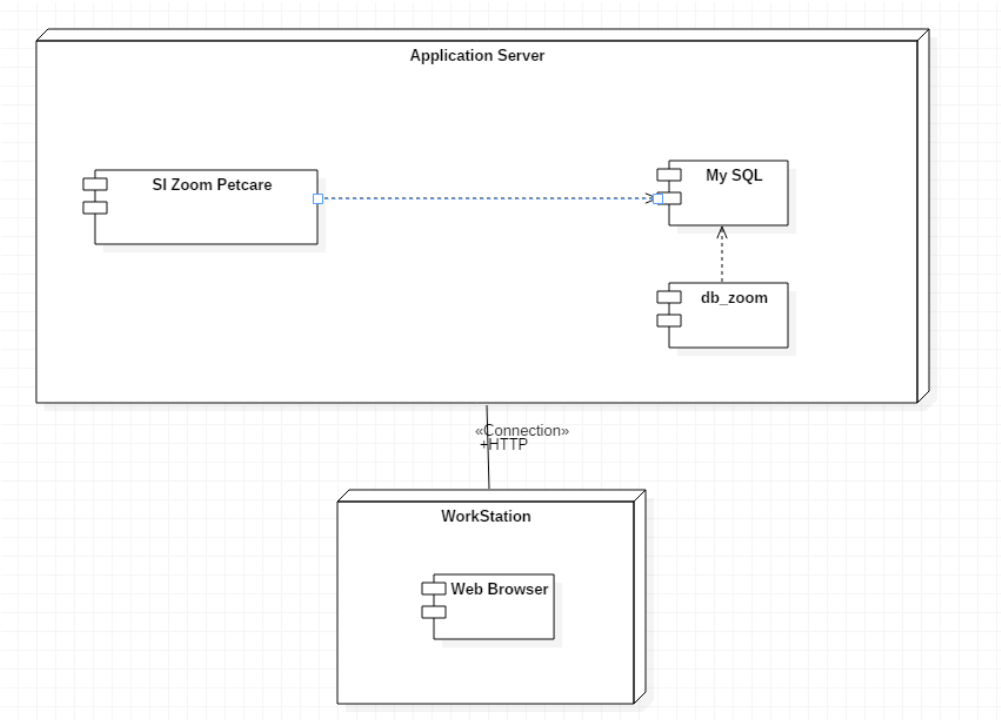
Class diagram adalah sebuah spesifikasi dari fungsionalitas yang menghasilkan objek dan merupakan inti dari pengembanggan aplikasi ini. Diagram ini juga merupakan gambaran keadaan atribut atau properti dari sistem yang melakukan manipulasi fungsi atau metode.



Gambar 4.12 *Class diagram* Yang Diusulkan

4.1.4.2. Deployment diagram

Deployment diagram adalah diagram untuk mengembangkan dari suatu sistem yang telah diciptakan sehingga dapat ditelusuri apabila terjadi kesalahan baik dalam perancangan maupun implementasinya.



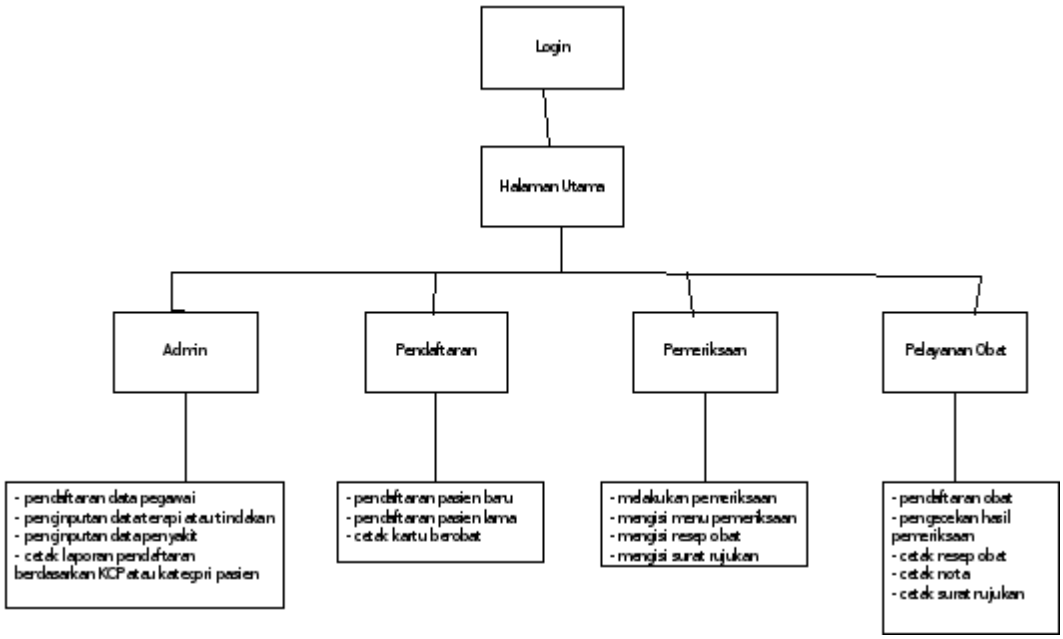
Gambar 4.13 *Deployment Diagram* Yang Diusulkan

4.2 Perancangan Antar Muka

Perancangan antar muka atau disebut juga *interface* aplikasi yang diusulkan dalam Sistem Informasi Zoom Petcare Bandung.

1 Struktur Menu

Dalam perancangan antar muka akan dibentuk dan digambarkan dalam struktur menu, adapun rancanngan struktur menus tersebut yaitu :



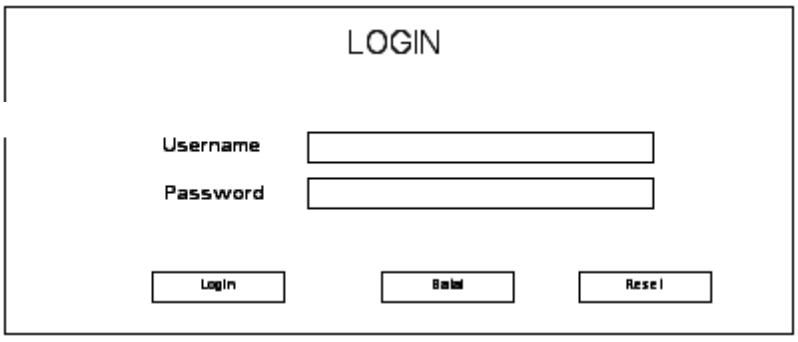
Gambar 4.14 Struktur Menu

2 Perancangan Input

Perancangan input dibuat untuk gambaran struktur penginputan yang akan di usulkan untuk pelayanan pasien, yaitu :

1 Input Data Login

Gambaran form login sistem informasi pelayanan pasien sebagai berikut :



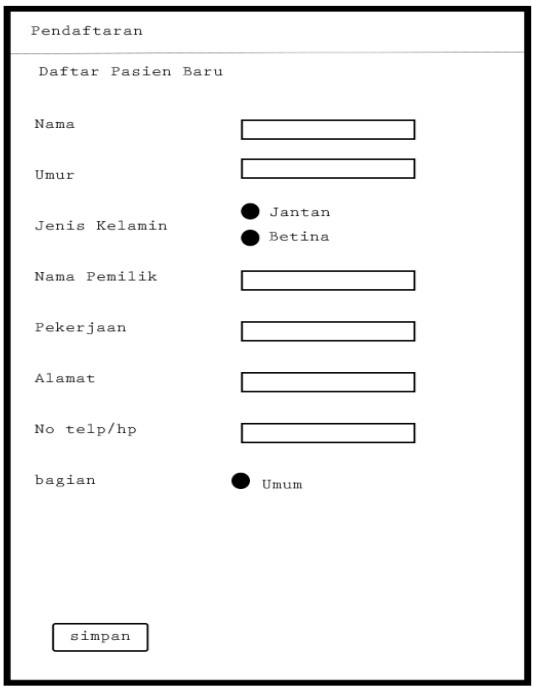
Gambar 4.15 Tampilan Login

Tabel 4.8 Tabel keterangan perancangan form login

Tombol / Fitur	Keterangan
Username	Kolom mengisi username
Password	Kolom mengisi password
Login	Tombol untuk submit login

2 Input Data Pasien Baru

Gambaran form Input data pasien baru yang akan mendaftar



Gambar 4.16 Form input data pasien baru

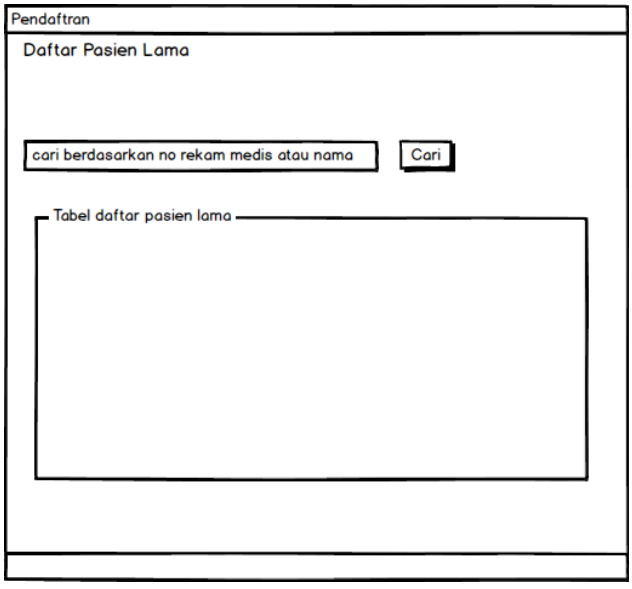
Tabel 4.9 Keterangan **Form input data pasien baru**

Tombol / Fitur	Keterangan
Nama	Kolom mengisi nama peliharaan
Umur	Kolom mengisi umur hewan peliharaan
Jenis Kelamin	Tombol untuk memilih jenis kelamin hewan peliharaan
Nama Pemilik	Kolom mengisi nama pemilik
Pekerjaan	Kolom mengisi pekerjaan pemilik
Alamat	Kolom mengisi alamat pemilik
No. telp	Kolom mengisi nomer telepon pemilik
Bagian	Tombol untuk memilih bagian yang akan menangani pasien

Simpan	Tombol untuk menyimpan data

3 Form Input Data Pasien Lama

Gambaran form Input data pasien lama yang akan mendaftar



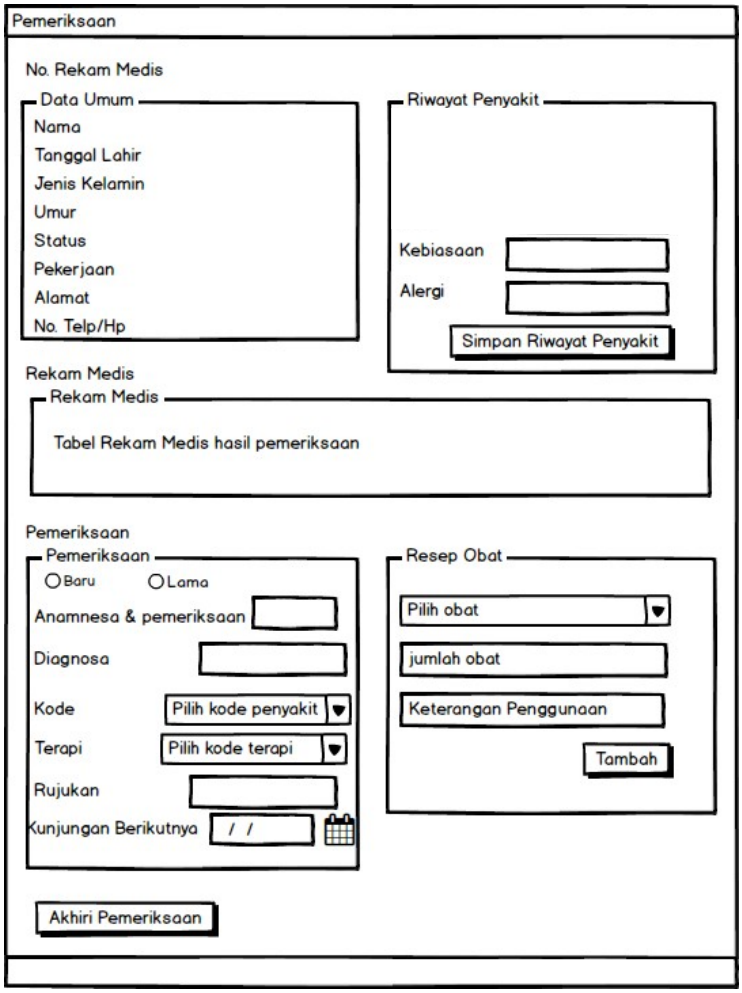
Gambar 4.17 Form input data pasien lama

Tabel 4.10 Form input data pasien lama

Tombol / Fitur	Keterangan
Kolom cari	Kolom mengisi no rekam medis atau nama pasien
Cari	Tombol untuk mencari pasien
Table daftar	Daftar hasil pencarian

4 Form Input Pemeriksaan Pasien

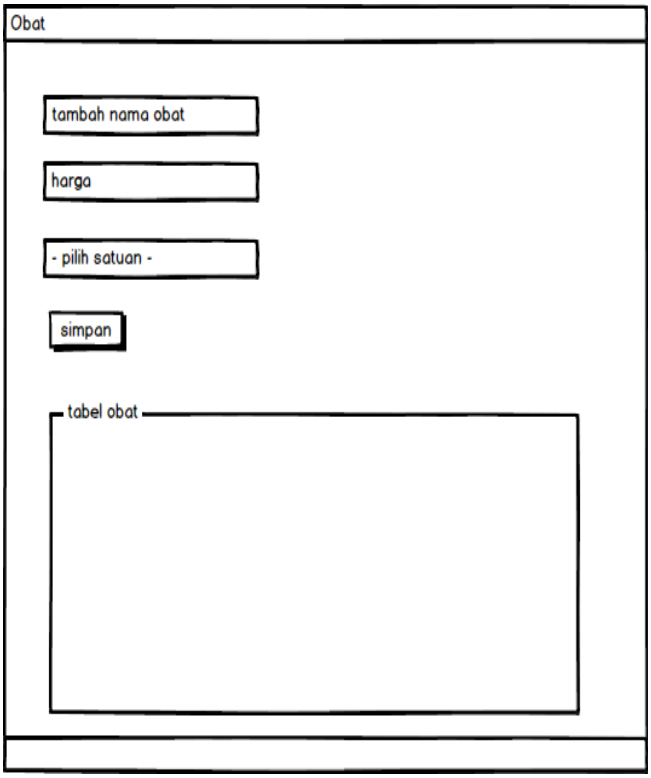
Gambaran Form Input untuk pemeriksaan pasien pada user dokter



Gambar 4.18 Form input untuk pemeriksaan pasien

5 Form Input Data obat

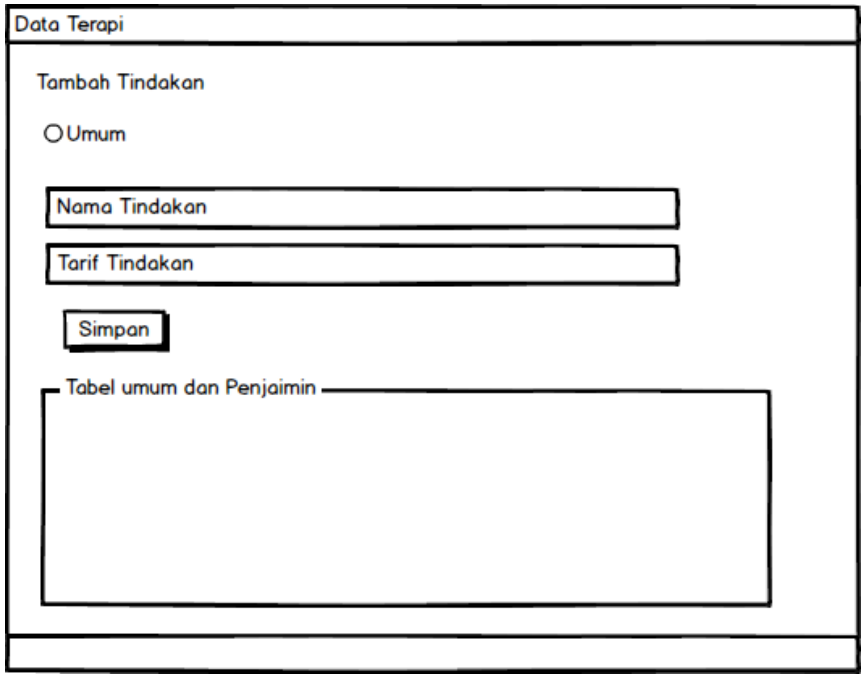
Gambaran form untuk penambahan data obat



Gambar 4.19 Form untuk penambahan data obat

6 Form Input data terapi

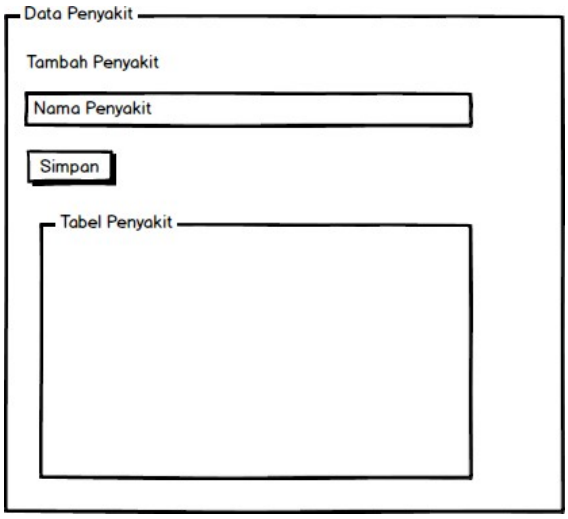
Gambaran form untuk input data terapi



Gambar 4.20 Form untuk input data terapi

7 Form Input Data Penyakit

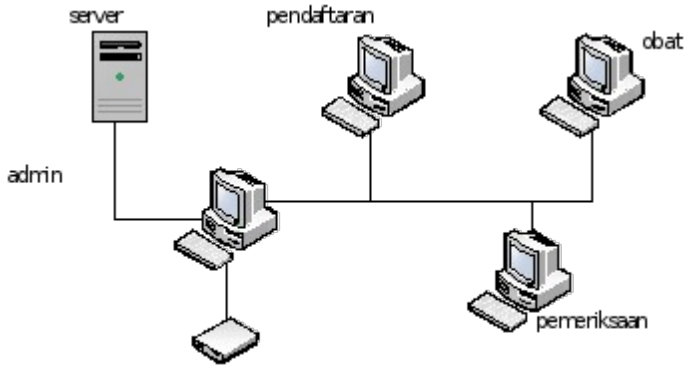
Gambaran form untuk input data penyakit



Gambar 4.21 Form untuk input data penyakit

4.3 Perancangan Arsitektur Jaringan

Perancangan arsitektur jaringan yang di bangun atau digunakan pada sistem yang diusulkan adalah arsitektur jaringan menggunakan jaringan *local area network* (LAN) dengan topologi bus, karena dengan menggunakan jaringan tersebut sudah cukup memenuhi kebutuhan sistem dan selain iyu juga dalam pengalikasian jaringannya cukup mudah. Berikut perancangan arsitektur jaringan :



Gambar 4.22 Perancangan Arsitektur Jaringan

4.4. Pengujian

Dalam penelitian ini pengujian merupakan bagian penting dalam siklus pembuatan atau pengembangan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian perangkat lunak ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas dan dapat berjalan sesuai kebutuhan *user*.

4.4.1. Rencana pengujian

Pengujian perangkat lunak ini menggunakan metode pengujian *Black Box*. Pengujian *Black Box* berfokus pada persyaratan atau kebutuhan fungsional perangkat lunak yang dibuat.

4.4.2. Kasus dan hasil pengujian

Tabel 4.11 Kasus dan Hasil Pengujian

NO	Fungsi Yang Diuji	Cara Pengujian	Hasil yang diharapkan	Hasil Pengujian
1	*Login*	Mengisi *username*, *password* dan menekan tombol *Login*	Menampilkan halaman utama	
2	Menambahkan data obat	Memilih tombol baru, lalu mengisi form *input* data obat	Data obat berhasil di tambah	
3	Menambahkan data tindakan	Memilih tombol tambah, lalu mengisi nama tindakan dan biaya, lalu memilih tombol simpan	Data tindakan berhasil di tambah	

4	Menambahkan data *user*	Memilih tombol tambah, lalu memilih posisi dan mengisi kode *user*, nama, *username*, *password*, alamat, dan no telepon, lalu memilih tombol simpan	Data *user* berhasil di tambah	
5	Menambahkan data pendaftaran	Memilih tombol tambah, lalu memilih tombol cari atau baru, lalu memilih tombol simpan	Data pendaftaran berhasil ditambah	
6	Menambahkan data diagnosa	Memilih tombol tambah, memilih tombol cari pasien, lalu mengisi keluhan, dan tindakan, lalu memilih tombol cari tindakan dan memilih tindakan yang di berikan, lalu memilih tombol cari obat dan memilih obat yang di berikan, lalu memilih tombol simpan	Data diagnosa berhasil di simpan dan dicetak	
7	Mengedit data pasien	Memilih nama pasien di tabel pasien, memilih tombol edit,	Data pasien berhasil di	

		mengisi data yang akan di edit, memilih tombol simpan.	update	
8	Membuat laporan kunjungan	Memilih tanggal awal dan tanggal akhir, memilih tombol cetak laporan	Laporan kunjungan berhasil di cetak	

4.4.3. Kesimpulan hasil pengujian

Berdasarkan hasil pengujian perangkat lunak ini dengan kasus uji sample di atas dapat di tarik kesimpulan bahwa perancangan dari perangkat lunak Sistem Informasi Zoom Petcare Bandung dan laporan-laporan yang dibutuhkan dapat berjalan dan berfungsi dengan baik, selain itu secara fungsional perangkat lunak ini mengeluarkan hasil sesuai dengan apa yang diharapkan sebelumnya.

4.5. Implementasi

Implementasi ialah proses penerapan rancangan program yang telah dibuat ke dalam sebuah aplikasi pemrograman sesuai dengan yang di harapkan dari program aplikasi tersebut. Kegiatan implementasi dilakukan dengan dasar yang telah direncanakan dalam tahap perancangan.

4.5.1. Implementasi perangkat lunak

Dalam membangun sistem informasi pelayanan pasien pada Zoom Petcare mempunyai kebutuhan perangkat lunak yang mendukung dalam pembuatan sistem ini, yaitu :

1 Sistem operasi : Microsoft Windows 8.1

2 HTML editor : Sublime Text 3

3 Web Server : XAMPP v3.2.2

4 Databaase Server : MySQL

5 Web Browser : UCBrowser

4.5.2. Implementasi perangkat keras

Selain perangkat lunak, kita juga membutuhkan perangkat keras untuk dapat menjalankan beberapa perangkat lunak. Minimum spesifikasi *hardware* untuk adpat membuat dan menjalankan aplikasi ini adalah sebagai berikut :

1 Processor Intel duo core 1.80 GHz.

2 Minimum RAM 1 Gb.

3 VGA minimum 256 Mb.

4 Kapasitas free Hardisk 80 Gb.

5 Monitor 15 inch, mouse, dan keyboard sebagai peralatan antarmuka.

4.5.3 Implementasi Basis Data

Implementasi basis data merupakan perancangan *database* yang dibuat menggunakan bahasa *SQL (Structure Query Language).* Berikut adalah implementasi *database* menggunakan aplikasi *MySql* dalam sistem informasi Zoom Petcare Bandung.

Table 4.12 Basis Data

Create database ‘pasien’		
No	Nama Tabel	Syntax SQL
1	pasien	CREATE TABLE ‘pasien’(‘no_rekam’ varchar(9) NOT NULL, ‘nama’ varchar(40) NOT NULL, ‘umur varchar NOT NULL, ‘jk’ char(1) NOT NULL, ‘namapemilik’ varchar(1) NOT NULL, ‘pekerjaan’ text NOT NULL, ‘alamat’ text NOT NULL, ‘telp’ int(12) NOT NULL, ‘bagian’ varchar(11) NOT NULL, ‘kebiasaan’ varchar(50) NOT NULL, ‘alergi’ varchar(50) NOT NULL,
2	Pegawai	CREATE TABLE ‘pegawai’ (‘id_pegawai’ varchar(4) NOT NULL, ‘nama’ varchar(35) NOT NULL, ‘jk’ char(1) NOT NULL, ‘alamat’ text NOT NULL, ‘telp1’ varchar(13) NOT NULL, ‘jabatan’ varchar(15) NOT NULL,

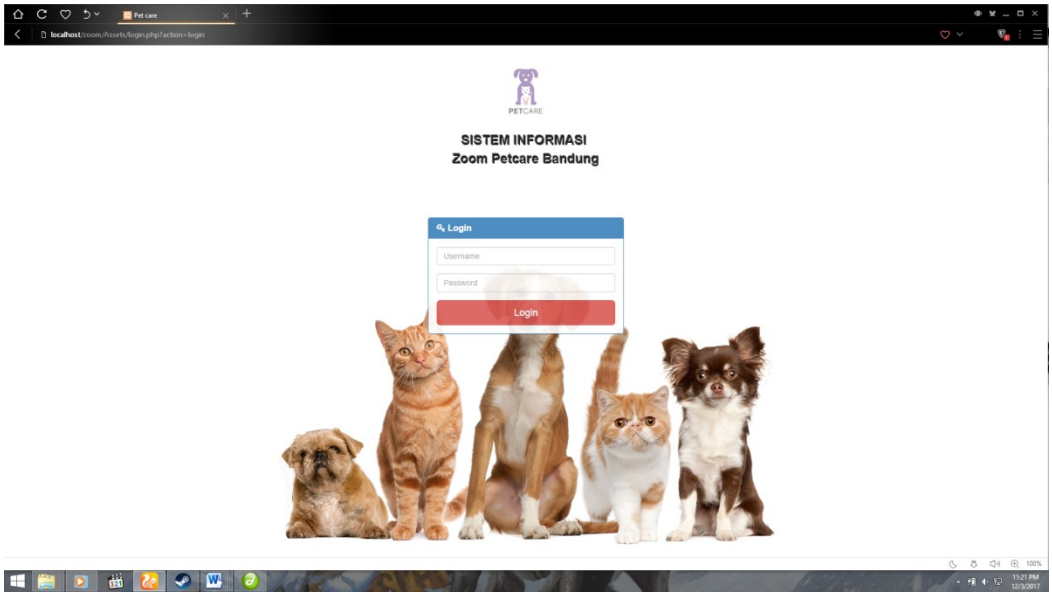
		‘username’ varchar(4) NOT NULL, ‘password’ varchar(8) NOTNULL)
3	Obat	CREATE TABLE ‘obat’ (‘id_obat’ varchar(5) NOT NULL, ‘nama’ varchar(50) NOT NULL, ‘satuan’ varchar(10) NOT NULL, ‘stok’ int(11) NOT NULL, ‘harga’ int(11) NOT NULL)
4	Pemeriksaan	CREATE TABLE ‘pemeriksaan’ (‘id_pemeriksaan’ int(11) NOT NULL, ‘id_pasien’ int(11) NOT NULL, ‘tgl_pemeriksaan’ datetime NOT NULL, ‘diagnosa’ text NOT NULL, ‘kode’ varchar(10) NOT NULL, ‘terapi’ varchar(50) NOT NULL, ‘rujukan’ varchar(50) NOT NULL, ‘id_dokter’ int(11) NOT NULL, ‘status’ int(11) NOT NULL, ‘kunjungan’ date NOT NULL, ‘kepada’ text NOT NULL, ‘fisik’ text NOT NULL,
5	Penyakit	CREATE TABLE ‘penyakit’ (

		‘id_penyakit’ varchar(5) NOT NULL, ‘nama_penyakit’ varchar(100) NOT NULL)
6	Resep	CREATE TABLE ‘resep’ (‘id_resep’ int(11) NOT NULL, ‘id_pemeriksaan’ int(11) NOT NULL, ‘id_obat’ int(11) NOT NULL, ‘jumlah’ int(11) NOT NULL, ‘keterangan’ varchar(10) NOT NULL)
7	Tindakan	CREATE TABLE ‘tindakan’ (‘id_tindakan’ varchar(4) NOT NULL, ‘nama’ varchar(100) NOT NULL, ‘tarif’ int(11) NOT NULL, ‘kategori’ varchar(11) NOT NULL)
8	Bayar	CREATE TABLE ‘bayar’ (‘no’ int(5) NOT NULL, ‘id_pemeriksaan’ int(11) NOT NULL, ‘tgl_bayar’ date NOT NULL, ‘total’ int(11) NOT NULL, ‘status’ char(1) NOT NULL)

4.5.4. Implementasi antar muka

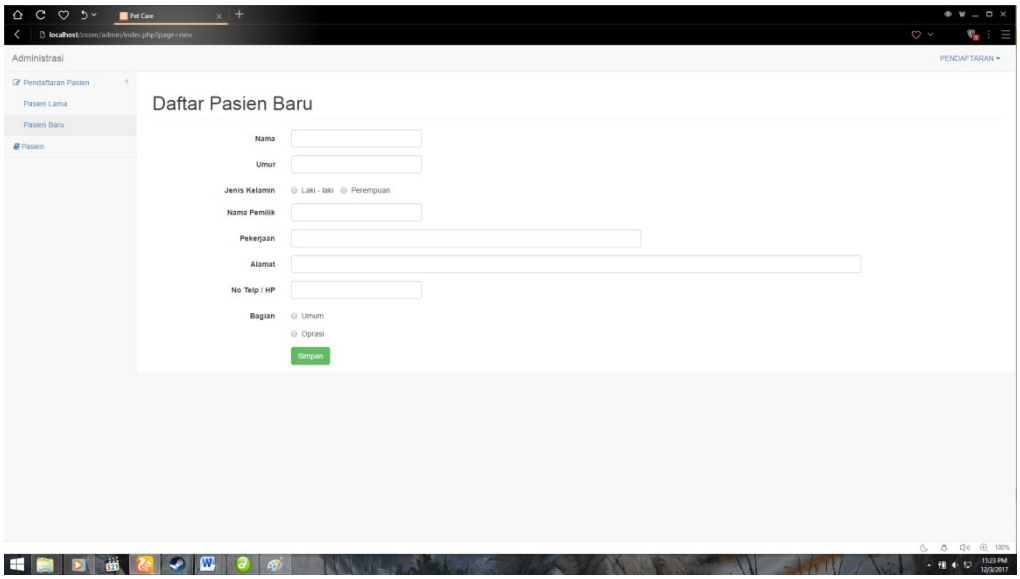
1 Login

Sebelum menggunakan aplikasi ini user diharuskan untuk login terlebih dahulu untuk nantinya memiliki hak akses sesuai dengan bagian yang dibutuhkan.



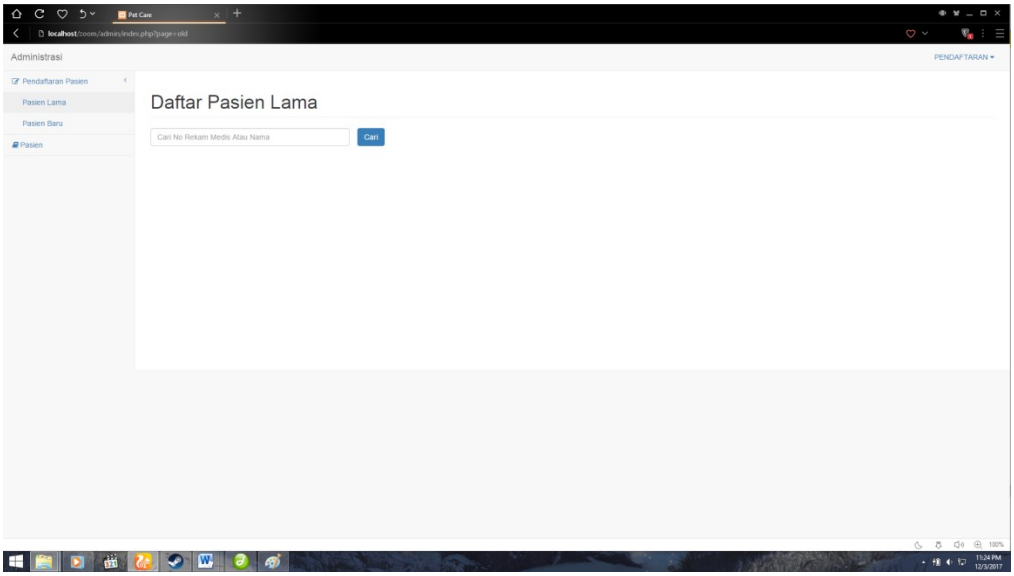
Gambar 4.23 Tampilan Login

2 Menu pendaftaran pasien baru



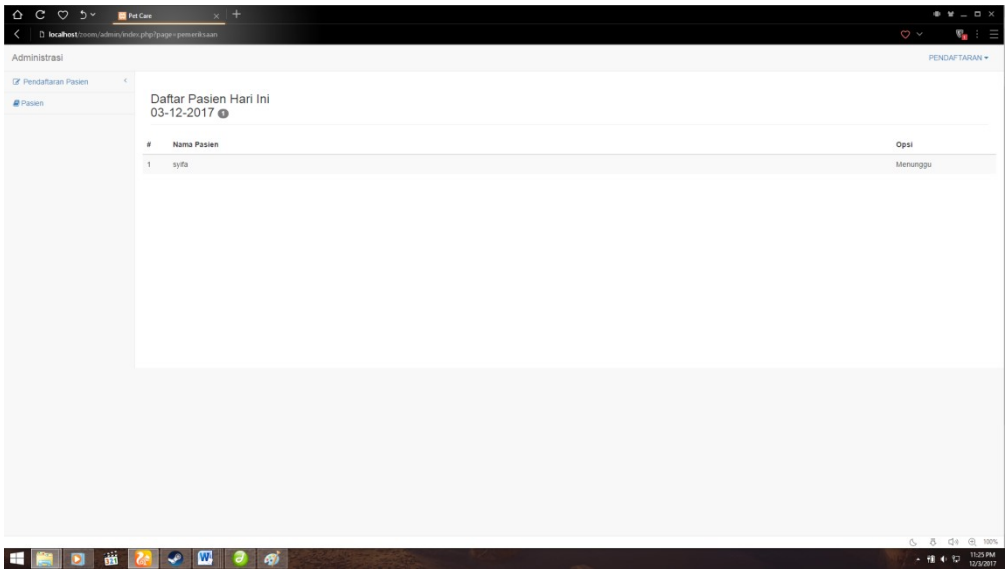
Gambar 4.24 Tampilan Daftar Pasien Baru

3 Menu pendaftaran pasien lama



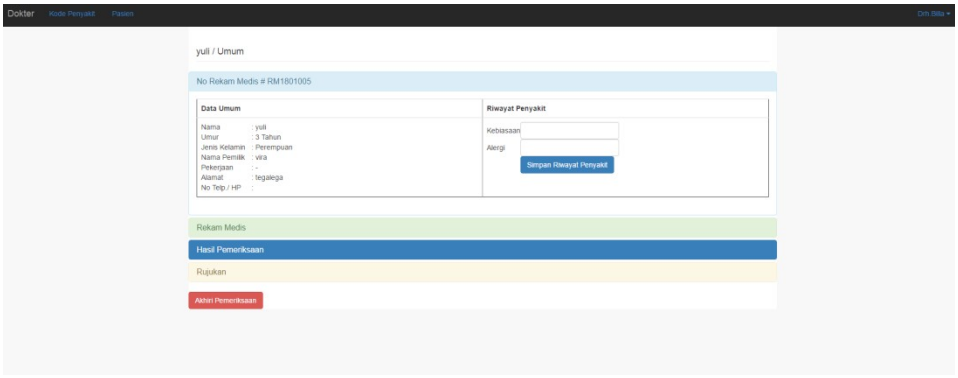
Gambar 4.25 Tampilan Daftar Pasien Lama

4 Waiting list Pemeriksaan



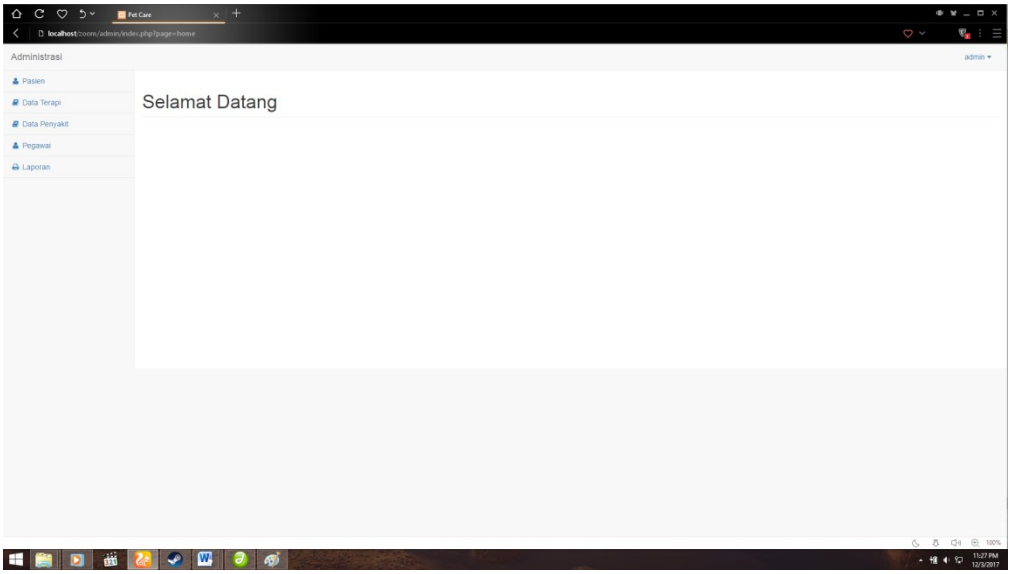
Gambar 4.26 Tampilan Waiting list pemeriksaan

5 Menu input pemeriksaan



Gambar 4.27 Tampilan Input Pemeriksaan

6 Menu halaman admin



Gambar 4.28 Menu halaman admin

7 Laporan pemeriksaan

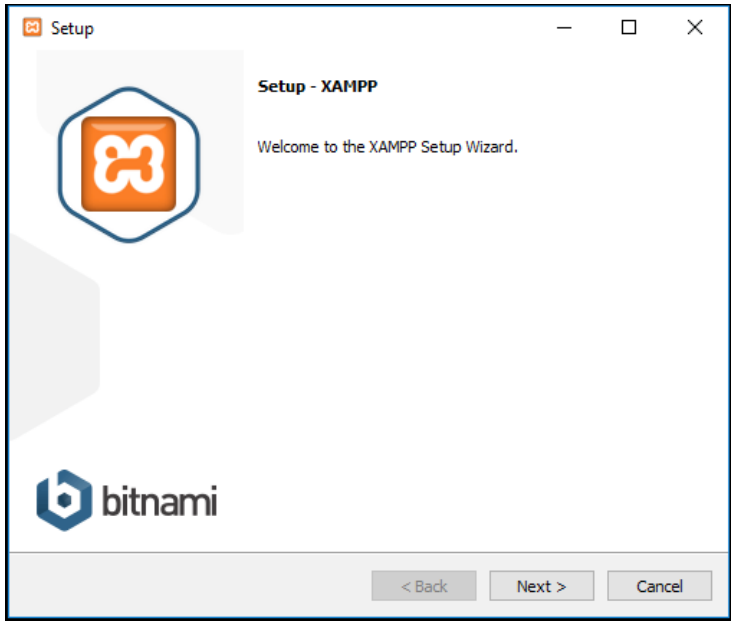


Gambar 4.29 Laporan Pendaftaran Pasien Umum

4.5.5 Implementasi Instalasi Program

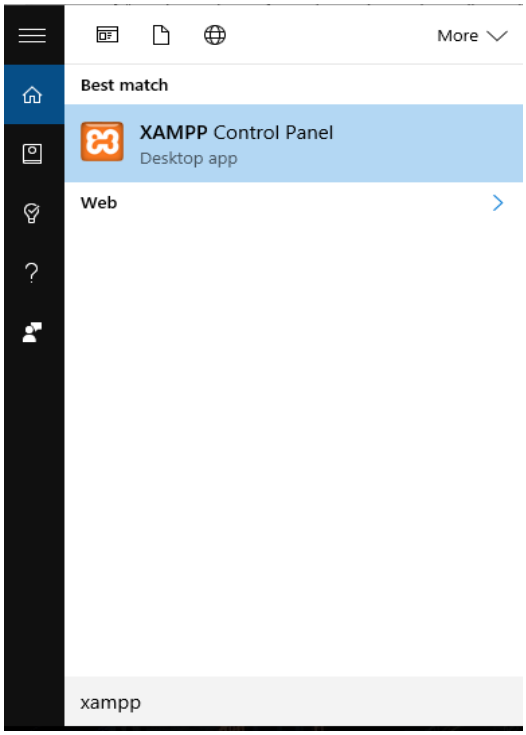
Berikut ini adalah tahapan instalasi program yang akan dijelakan oleh penulis, langkah-langkahnya sebagai berikut:

a Klik app Xampp untuk mulai install, kemudian akan muncul tampilan sebagai berikut untuk selanjutnya next – next saja ikuti langkah instalasinya dan penyimpanan instalannya atau direktorinya :



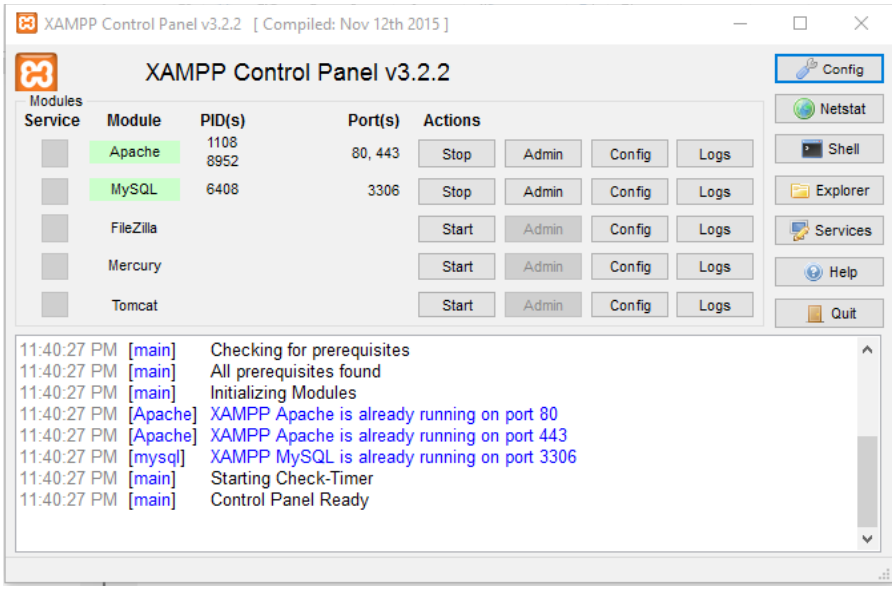
Gambar 4.30 Install XAMPP

b Kemudian setelah selesai proses penginstalan, lalu cari pada menu search program xampp



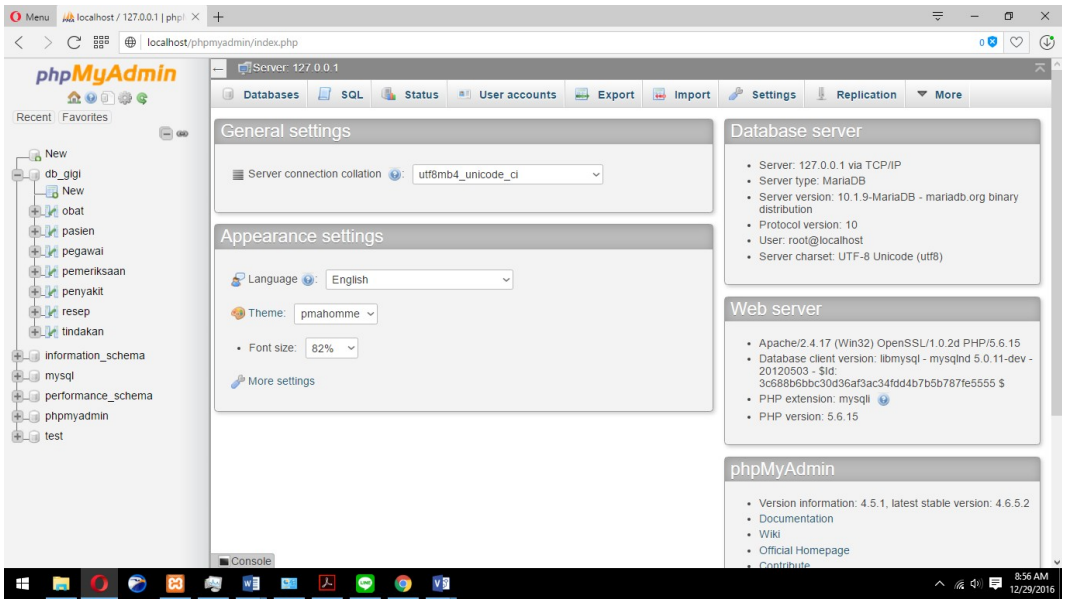
Gambar 4.31 Menjalankan XAMPP

c Kemudian aktifkan denga cara klik start Apache dan MySQL



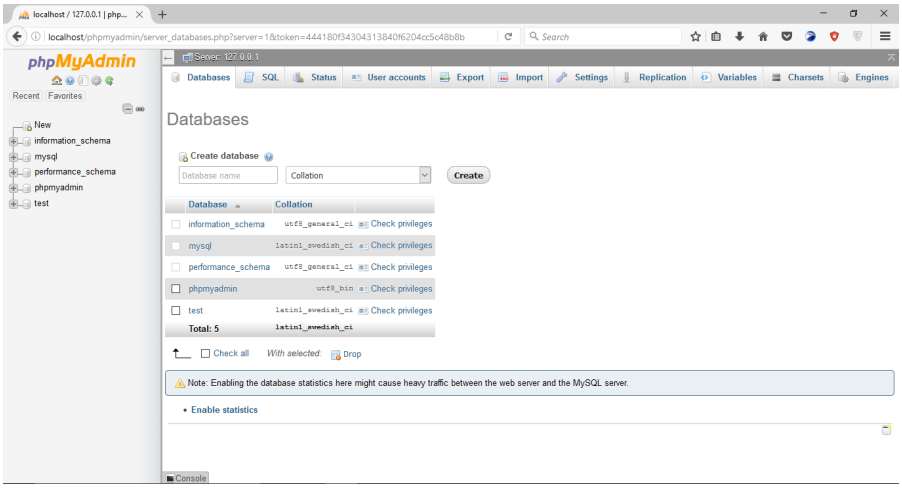
Gambar 4.32 Mengaktifkan XAMPP Control Panel

d buka Google Chrome, opera, atau *web* browser lainnya dan ketikan alamat http://localhost/phpmyadmin/ pada url



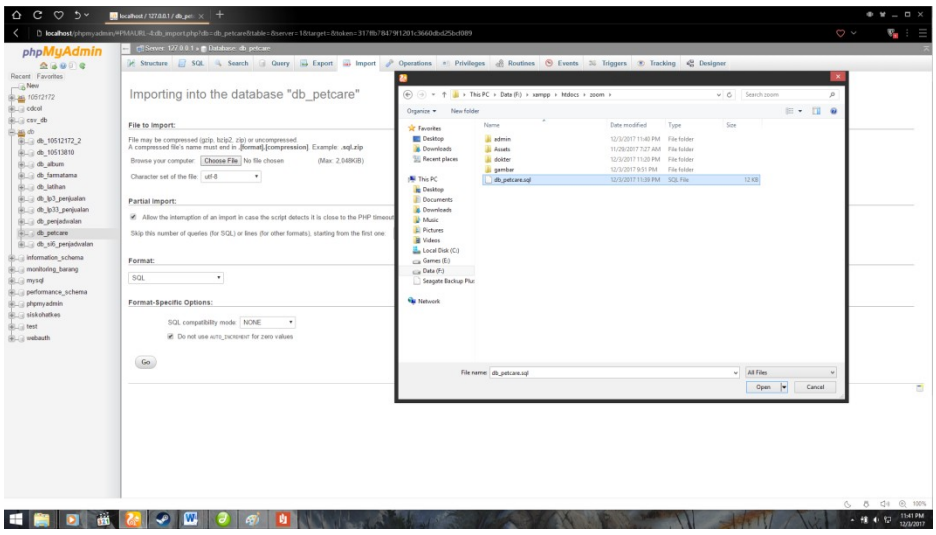
Gambar 4.33 Tampilan Utama phpMyAdmin

e kemudian buat database dengan nama database db_petcare



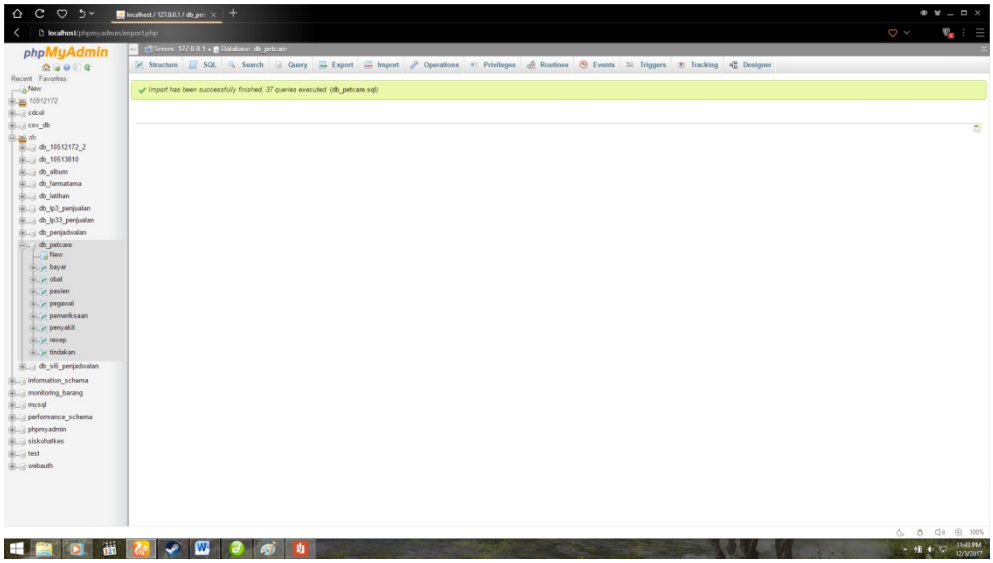
Gambar 4.34 Membuat Database

f selanjutnya pilih import database db_petcare, sebelumnya simpan program di folder xampp/htdocs/



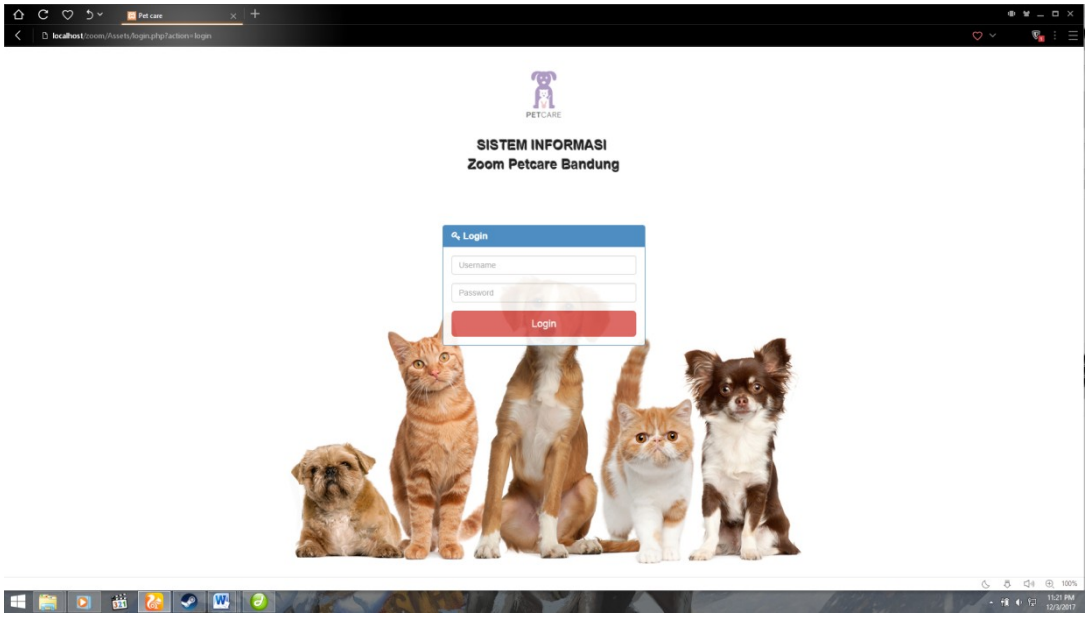
Gambar 4.35 Import Database

g Database berhasil terimport dan tabel yang berada di database tersebut sudah muncul



Gambar 4.36 Database Berhasil Terimport

h Dan yang terakhir untuk masuk pada program ketikkan pada web browser http://localhost/petcare/Assets/login.php?action=login



Gambar 4.37 Halaman Utama Program