**BAB 1**

**PENDAHULUAN**

**1.1. Latar Belakang**

Teknologi informasi memberikan dampak yang cukup signifikan bagi peradaban manusia saat ini. Pemanfaatan *internet* sudah memasyarakat di setiap bidang kehidupan manusia sebagai media informasi yang paling luas, termasuk institusi pemerintah. Salah satu dokumen yang dipublikasikan melalui jaringan *internet* adalah *website.* Dampak kemajuan ilmu pengetahuan teknologi yaitu dengan perkembangan teknologi informasi saat ini sangat berarti bagi semua kalangan, baik berupa perusahaan swasta maupun di pemerintah. Informasi umumnya hanya diperoleh melalui media seperti surat kabar atau buku maupun media elektronik seperti radio dan televisi. Tetapi saat ini komputer merupakan suatu sarana informasi yang paling banyak digunakan oleh instansi perusahaan maupun perorangan. Dengan komputer masyarakat bisa mendapatkan informasi dan saling berkomunikasi dengan cepat, tepat, dan akurat

Dalam era globalisasi saat ini poliklinik dituntut untuk meningkatkan Kinerja dan daya saing sebagai badan usaha dengan tidak mengurangi misi sosial yang dibawahnya. Poliklinik harus merumuskan kebijakan-kebijakan strategis antara lain *efisien* dari dalam organisasi *management*. Peran serta harus mampu secara tepat mengambil keputusan untuk peningkatan pelayanan kepada masyarakat agar dapat menjadi organisasi yang *responsif, inovatif, efektif, efisien* dan menguntungkan. Poliklinik Polresta Palembang sebagai salah satu Poliklinik yang memiliki peran penting dalam pengembangan teknologi informasi di Palembang.

Di dalam penyimpanan dan pengolahan data pada Poliklinik Polresta Palembang pada saat ini masih manual berbentuk buku, ada pun prosedur yang dilakukan dalam menyimpan data di Poliklink Polresta Palembang saat ini pasien mengisi kartu atau formulir lalu ditulis kembali oleh petugas di dalam buku. Hal ini akan mempersulit para pegawai untuk mencari data pasien, obat, rekap medis, para medis dan SKD (Surat Keterangan Dokter) jika sewaktu-waktu data tersebut dibutuhkan.

Oleh karena itu penulis merasa perlu untuk membuat suatu sistem basis data yang dapat mengatasi kelemahan dan kekurangan dari pengolahan data sebelumnya. Dimana dengan sistem yang baru ini pengolahan data dapat dilakukan dengan cepat walaupun jumlah data yang dimasukkan banyak. Dengan kehadiran sistem basis data ini akan lebih *efisien* dalam mengerjakan laporan dan penyimpanan data secara baik.

Dari permasalahan di atas penulis mengambil kesimpulan untuk mengatasi masalah yang dialami oleh Poliklinik Polresta Palembang perlu adanya **“Perancangan Dan Implementasi Sistem Basis Data Berbasis *Web* Pada Poliklinik Polresta Palembang”.**

**1.2. Perumusan Masalah**

Berdasarkan uraian dan penjelasan diatas, maka permasalahan yang akan dirumuskan dalam penelitian adalah “Bagaimana Merancang dan mengimplementasikan Sistem Basis Data Pada Poliklinik Polresta Palembang ?”

**1.3. Batasan Masalah**

Untuk menghindari agar penyimpangan permasalahan dari rumusan yang ada, maka penulis hanya membatasi pada pengolahan data. Tampilan yang akan di tampilkan pada rancangan ini adalah

1. Merancang sistem pengolahan data pada para medis, pendataan pasien, data obat, rekap medis, dan data SKD (Surat Keterangan Dokter).
2. Mengimplementasikan sistem yang dirancang menggunakan bahasa pemrograman *PHP* dan *database* mengunakan *MySQL.*

**1.4. Tujuan dan Manfaat Penelitian**

**1.4.1. Tujuan Penelitian**

Tujuan dari penelitian ini adalah merancang dan Mengimplementasikan sistem basis data pada Poliklinik Polresta Palembang dengan menggunakan bahasa pemprograman *PHP* dan untuk databasenya mengunakan *database MySQL.* Agar dapat meningkatkan pelayanan pasien dan operasional di Poliklinik Polresta palembang.

**1.4.2. Manfaat Penelitian**

Adapun manfaat penelitian yang diambil penulis dalam penelitian ini adalah

1. **Bagi Poliklinik**

Dengan dibuatnya sistem basis data ini poliklinik pada polresta palembang ini dapat membantu dalam proses penyimpanan data pasien, obat, para medis, rekap medis dan data SKD (Surat Keterangan Dokter).

1. **Bagi Penulis**

Diharapkan dapat memberikan gambaran bahwa teknologi komputer dapat memberikan banyak keuntungan khususnya dalam penyimpanan data dan pengolahan data.

**1.5. Metodologi Penelitian**

**1.5.1. Waktu dan Tempat Penelitian**

Waktu penelitian ini direncanakan selama lima bulan yaitu dimulai dari bulan Oktober 2012 sampai bulan Februari 2013 di Poliklinik Polresta Palembang yang beralamat di jalan Gub. H. Achmad Bastari 01 Palembang 30252.

**1.5.2. Alat dan Bahan**

Alat dan bahan yang digunakan dalam penelitian ini meliputi *hardware* dan *software* yang diantaranya sebagai berikut:

1. *Hardware* berupa satu unit laptop Toshiba dengan spesifikasi :

1. *Processor* intel® Pentium
2. *Ram* 1 GB
3. *Hardisk* 320 GB

2. Printer canon PIXMA ip 2770

3. *Software*

1. *MySQL*
2. *PHP*
3. *Xampp*
4. *Dreamweaver*

**1.5.3. Metode Penelitian**

Metode yang digunakan adalah metode deskriptif. Metode deskriptif suatu metode dalam meneliti status sekelompok manusia, suatu objek, suatu set kondisi, suatu sistem pemikiran, ataupun suatu kelas peristiwa pada masa sekarang. Tujuan dari penelitian deskriptif ini adalah untuk membuat deskripsi, gambaran atau lukisan secara sistematis, faktual dan akurat mengenai fakta-fakta, sifat-sifat serta hubungan antar fenomena yang diselidiki.(Nazir,2003:54)

**1.5.4. Metode Pengumpulan Data**

Dalam melakukan penelitian untuk mendapatkan data maka metode yang dapat digunakan dalam proses pengumpulan data dengan cara :

1. Observasi

Teknik observasi dilakukan dengan mengamati dan mengambil data secara langsung di Poliklinik Polresta Palembang pada objek yang di teliti dengan mengadakan pengamatan dan pencatatan terhadap data yang diperlukan

1. Wawancara

Penulis melakukan wawancara atau tanya jawab secara langsung untuk mendapatkan informasi mengenai data yang ada di Poliklinik Polresta Palembang.

1. Studi pustaka

Mencari informasi tentang teori-teori yang berhubungan dengan penelitian dari sumber seperti buku-buku dan jurnal yang berkaitan dengan penelitian.

1. Dokumentasi

Penulisan mengumpulkan dokumentasi dalam bentuk data-data dan catatan.

**1.5.5. Metode Perancangan dan Implementasi Sistem Basis Data**

Dalam penelitian ini proses analisis dan perancangan basis data akan dilakukan dengan *siklus database life cycle* dengan tahapan-tahapan sebagai berikut (indrajani,2007:111):

1. *Database Planning*
2. *Definisi system*
3. *Requirement Collection and Analsys*
4. Perancangan Basis Data (*Design Database*)
5. Perancangan Basis Data *Conceptual*
6. Mengidentifikasi tipe entitas
7. Mengidentifikasi tipe relasi
8. Mengidentifikasi atribut dengan tipe entitas
9. Menentukan domain atribut
10. Menentukan atribut *candiadte key* dan *primary key*
11. Perancangan Basis Data *Logical*
12. Menghilangkan fitur yang tidak kompetibel dengan model data relasional
13. Menurunkan relasi untuk model data logical local
14. Validasi menggunakan normalisasi
15. Perancangan Basis Data *Fisikal*
16. Melakukan relasi dasar
17. Analisis transaksi
18. Estimasi kebutuhan disk space
19. Mendesain mekanisme keamanan

**BAB II**

**TINJAUAN PUSTAKA**

**2.1. Landasan Teori**

**2.1.1. Sistem Basis Data**

**2.1.1.1. Definisi Dari Sistem**

Kata “Sistem” selalu berkonotasi pada 3 (tiga) hal utama : komponen, ketergantungan dan tujuan. Artinya, setiap sistem akan selalu terdiri atas berbagai komponen yang saling berhubungan dan memiliki ketergantungan (*dependence*), dalam rangka mencapai satu tujuan tertentu. Dengan kata lain, bukanlah disebut sebuah sistem, jika hanya terdiri dari sebuah komponen, atau jika tidak memiliki hubungan antar komponen yang saling tergantung, atau jika tidak diniatkan untuk satu tujuan tertentu.(Fathansyah,2004:2).

Sistem adalah sebuah tatanan (keterpaduan) yang terdiri atas sejumlah komponen fungsional (dengan satuan fungsi/tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses/pekerjaan tertentu.(Fathansyah,2004:9)

**2.1.1.2. Pengertian Sistem Basis Data**

Sistem basis data adalah suatu sistem menyusun dan mengolah *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi/perusahaan sehinggah mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan.(Marlinda,2004:1)

**2.1.1.3. Komponen Sistem Basis Data**

Menurut Fathansyah(2004:10-13) komponen sistem basis data adalah

1. Perangkat keras *(hardware)*
2. Sistem operasi *(operating system)*
3. Basis data *database)*
4. Sistem pengelola basis data *(database management system/*DBMS*)*
5. Pemakai *(user)*
6. Aplikasi (perangkat lunak) lain

**2.1.1.4. DBMS *(Database Management System)***

Kumpulan *file* yang saling berkaitan dan program untuk pengelolanya disebut DBMS. DBMS merupakan kumpulan *software* yang mengkoordinasikan semua kegiatan yang berhubungan dengan basis data agar data dapat diakses/dipakai oleh pemakai*(user).* Tujuannya adalah efisiensi dan kenyamanan dalam memperoleh dan menyimpan informasi di dalam basis data.(Marlinda,2004:6)

**2.1.2. Basis Data**

**2.1.2.1. Definisi Basis Data**

Basis data terdiri atas 2 kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya. Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

* Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
* Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansasi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
* Kumpulan *file*/*table*/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronis.(Fathansyah,2004:2)

Basis data adalah suatu susunan/kumpulan data operasional lengkap dari suatu organisasi/dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya.(Marlinda,2004:1)

**2.1.2.2. Manfaat Basis Data**

Pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (objektif) (Fathansyah,2004:5-7) sebagai berikut :

1. Kecepatan dan kemudahan *(speed)*
2. Efisiensi ruang penyimpanan *(space)*
3. Keakuratan *(accuracy)*
4. Ketersediaan *(availability)*
5. Kelengkapan *(completeness)*
6. Keamanan *(security)*
7. Kebersamaan pemakaian *(sharability)*

**2.1.3. Normalisasi**

Normalisasi adalah suatu teknik dengan pendekatan bottom up yangdigunakan untuk mengidentifikasikan hubungan. Dimulai dari menguji hubungan, yaitu functional dependencies antara atribut. Pengertian lainnya adalah suatu teknik yang menghasilakan sekumpulan hubungn dengan sifat-sifat yang diinginkan dan memenuhi pada perusahaan. (Indrajani,2011:57)

Tujuan utama normalisasi adalah mengidenfikasikan kesesuaian hubungan yang mendukung data untuk memenuhi kebutuhan perusahaan. Adapun karekteristik hubungan tersebut mencakup:

1. Minimal jumlah atribut yang diperlukan untuk mendukung kebutuhan perusahaan.
2. Atribut dengan hubungan logika yang menjelaskan mengenai *functional de pendencies.*
3. Minimal duplikasi untuk tiap atribut

Menurut Nugroho (2011:199) terdapat 5 bentuk normalisasi yang secara garis besar digambarkan sebagai berikut:

1. Bentuk normal pertama (1NF/*First* Normal), Bentuk normal pertama adalah suatu bentuk relasi dimana atribut bernilai banyak telah dihilangkan sehingga akan dijumpai nilai tunggal.
2. Bentuk normal kedua (2NF), semua ketergantungan fungsional yang bersifat sebagian telah dihilangkan.
3. Bentuk normal ketiga (3NF), semua ketergantungan transitif telah dihilangkan.
4. Bentuk normal *Bouce-Codd*, semua Anomali yang tersisa dari hasil penyempurnaan ketergantungan fungsional sebelumnya telah dihilangkan.
5. Bentuk normal keempat (4NF), semua ketergantungan bernilai banyak telah dihilangkan.
6. Bentuk normal kelima (5NF), semua anomali yang tinggal telah dihilangkan.

**2.1.4. Desain Basis Data *(Database Design)***

Desain basis data adalah proses membuat desain yang akan mendukung operasional dan tujuan perusahaan :

1. Perancangan basis data *Conceptual*

Merupakan suatu proses pembentukan model yang berasal dari informasi yang digunakan dalam perusahaan yang bersifat independen dari keseluruhan aspek fisik. Model data tersebut dibangun dengan menggunakan informasi dalam spesifikasi kebutuhan *user* dan merupakan sumber informasi untuk fase desain logikal.

1. Perancangan basis data *Logikal*

Merupakan suatu proses pembentukan model yang berasal dari informasi yang digunakan dalam perusahaan yang berdasarkan model data tertentu, namun independen terhadap DBMS tertentu dan aspek fisik lainnya. Misalnya relasional. Model data konseptual yang telah dibuat sebelumnya, diperbaiki dan dipetakan kembali ke dalam model data logikal.

1. Perancangan basis data *Fisikal*

Merupakan proses yang menghasilkan deskripsi implementasi basis data pada penyimpanan sekunder. Menggambarkan struktur penyimpanan dan metode akses yang digunakan untuk mencapai akses yang efisien terhadap data. Dapat dikatakan juga, desain fisikal merupakan cara pembuatan menuju DBMS tertentu.

**2.1.5. Fase Perancangan Basis Data**

Tahapan dari perancangan basis data dalam penelitian ini terdapat pada siklus hidup pengembangan basis data berikut :

Database

Planning

System

Definition

Requirement

Collection &

Analysis

Database Design

DBMS Selection

(Optional)

Logical Database

Design

Application

Design

Physical Database

Design

Implementation

Prototyping

(Optional)

Data conversion & loading

Testing

Operational

Maintance

**Gambar 2.1** Siklus Hidup Pengembangan Basis Data

**2.1.6. Tahapan Perancangan Basis Data**

**2.1.6.1. Tahapan Perancangan Sistem**

Tahapan rancangan sistem basis data, dibagi menjadi dua bagian, yaitu perancangan basis data dan perancangan proses. Perancangan basis data merupakan langkah untuk menentukan basis data yang diharapkan dapat mewakili seluruh kebutuhan pengguna. Penyusunan basis data ini berdasarkan kamus aliran data yang telah dibuat pada tahapan sebelumnya. Perancangan basis data terdiri atas perancangan basis data secara konseptual, perancangan basis data secara logis, dan perancangan basis data secara fisik. Perancangan proses biasanya menghasilakan dokumentasi perancangan dalam bentuk spesifikasi dan bagan struktur sistem.(Kadir,2003:45)

**2.1.6.2. Tahapan Implementasi Sistem**

Tahap implementasi sistem mencakup pengkodean program, pengujian program, pemasangan program, dan juga pelatihan kepada pengguna. Setelah tahap ini berakhir maka akan sampai pada penggunaan. Dalam hal ini aplikasi mulai dioperasikan oleh pengguna untuk melakukan berbgai transaksi.(kadir,2003:61)

Setelah merancang secara logika dan fisik lengkap, kita dapat melaksanakan sistem basis data, perintah-perintah dalam DDL dan SDL *(storage definition language)* dan DBMS yang dipilih, dihimpun, dan digunakan untuk membuat skema basis data dan *file-file* basis data yang kosong. Sekarang basis data tersebut dimuat atau disatukan dengan datanya. Jika data harus diubah, maka diperlukan perubahan-perubahan yang mungkin untuk format ulang datanya yang kemudian dimasukan ke basis data yang baru. Transaksi-transaksi basis data sekarang harus dilaksanakan oleh para programmer aplikasi.(Marlinda,2004:114)

Implementasi sistem adalah meliputi penyiapan peralatan, penyusunan program, pelatihan karyawan, penyiapan jaringan, dan uji coba sistem. Implementasi meliputi berbagai kegiatan yang tidak lebih mudah dari desain sistem, yaitu meliputi:

1. Penyiapan *hardware* dan jaringan
2. Penyiapan *software* (pembuatan program)
3. Pemilihan dan pelatihan karyawan
4. Pengujian program dan prosedur
5. Pembuatan dokumentasi
6. Konversi data.

**2.1.7. Implementasi**

**2.1.7.1. Pengertian Implementasi**

Implementasi adalah tahap yang sangat penting bagi proses pencapaian tujuan, tahap ini berkaitan erat dengan jalan keluar yang telah direncanakan dan didesain untuk mendukung tuuan penyelenggaraan rencana maka tahap implementasi akan dikelompokkan menjadi 3, sesuai dengan tujuan, yaitu penyediaan informasi dan pengembangan strategi komunikasi, meningkatkan kapasitas pada mitra dalam hal pengelolaan informasi dan komunikasi, peningkatan sinergi dan kerja sama para dalam hal pengelolaan informasi dan komunikasi.(Afianto,2005:25)

**2.1.7.2. Implementasi Basis Data**

Tahapan implementasi basis data merupakan upaya untuk membangun basis data fisik yang ditempatkan dalam memori sekunder *(disk)* dengan bantuan DBMS *(database management system)* yang kita pilih. Tahap implementasi basis data diawali dengan melakukan transformasi dari model data yang telah selesai dibuat ke skema/struktur basis data sesuai dengan DBMS yang dipilih.(Fathansyah,2004:113)

Tahap implementasi sistem mencakup pengkodean program, pengujian program, pemasangan program, dan juga pelatihan kepada pengguna. Tahap implementasi basis data merupakan suatu upaya untuk membangun basis data fisik yang di tempatkan dalam memori sekunder *(disk)* dengan bantuan *DBMS (database manajement system).*

Setelah tahap implementasi selesai dilakukan akan terlihat bahwa kualitas dan bentuk perancangan basis data akan berpengaruh terhadap cara dan performasi pemanfaatan basis data.(Ladjamuddin,2004:42)

**2.1.8. *WEB***

**2.1.8.1 Pengertian *WEB (Website)***

*Website* adalah sebuah *situs* *web* (sering pula disingkat menjadi *situs* saja; *website, site*) adalah sebutan bagi sekelompok halaman *web* (*web page*), yang umunya merupakan bagian dari suatu nama *domain* *(domain name)* atau *sub domain*  di *World Wide Web (WWW*) di *internet*. *WWW* terdiri dari seluruh situs *web* yang tersediah kepada publik.(Dukom,2012:5)

**2.1.8.2 *WEB Database* (Basis Data *WEB*)**

Pemakaian *World Wide Web* sebagai alat untuk menyajikan dan mengumpulkan data telah berkembang jauh melampaui tampilan halaman *web* sederhana. Halaman-halaman *web* begitu sulit diatur pada mulanya dan hampir mustahil untuk dapat diperbaharui setiap waktu. Seiring dengan berjalannya waktu, volume dan struktur data yang disajikan pada *web* membutuhkan penyimpanan dan pengolahan ke dalam sebuah *database*/ basis datadan kemudian menghasilkan halaman-halaman *web* yang berbasis pada database tersebut. Jumlah *situs* yang mendapat keuntungan dari pendekatan metode ini kian hari kian meningkat. Berbagai *situs Web* yang menyajikan informasi tentang kumpulan data-data merupakan calon pengguna *Web database* ini. Data-data itu bisa berupa produk penjualan, foto-foto pribadi, *hyperlink*, atau apapun yang kita mau. Pendekatan yang umum adalah mendefenisikan sebuah *database*/basis data (misalnya: dengan *MySQL*), menambah *record,* menghapus, memperbarui atau mengelompokkan *database* dan langsung dibuat sebagai halaman *web* dengan cepat. Keuntungan pendekatan semacam ini adalah:

* Database lebih mudah dipelihara daripada memelihara halaman-halaman web individual.
* Menggunakan fasilitas pencarian database untuk kata kunci yang kita inginkan.
* Dengan database jadi lebih mudah untuk menyajikan data yang sama dalam cara yang berbeda, misalnya: data ditampilkan berdasarkan kategori, berdasarkan uraian, berdasarkan usia atau berdasarkan field lain dalam database.

**2.1.9. Sekilas Tentang *PHP***

*PHP* adalah salah satu bahasa pemprograman skrip yang dirancang untuk membangun aplikasi *web*. Ketika dipangil dari *web* *browser*, program yang ditulis dengan *PHP* akan di-*parsing* di dalam *web server* oleh *interpretr PHP* dan diterjemahkan ke dalam dokumen *HTML*, yang selanjutnya akan ditampilkan kembali ke *web browser.* Karena pemrosesan program *PHP* dilakukan di lingkungan *web server,* *PHP* dikatakan sebagai bahasa sisi *server (server-site).* Kode *PHP* tidak terlihat pada saat *user* memilih perintah *“View Source”* pada *web browser* yang mereka gunakan.(Budi,dkk,2010:41)

*PHP* diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1994. Awalnya, *PHP* digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *homepage*-nya. Rasmus Lerdorf adalah salah seorang pendukung *open source*. Oleh karna itu, ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis, kemudian menambah kemampuan *PHP* 1.0 dan meluncurkan *PHP* 2.0. Pada tahun 1996, *PHP* telah banyak digunakan dalam *website* di dunia. Sebuah kelompok pengembangan *software* yang terdiri dari Rasmus, Zeew Suraski, Andi Gutman, Stig Bakken, Shane Caraveo, dan Jim Winstead bekerja sama untuk menyempurnakan *PHP* 2.0. Akhirnya, pada tahun 1998, *PHP* 3.0 diluncurkan. Penyempurnaan terus dilakukan *PHP* 4.0. Tidak berhenti sampai disini kemampuan *PHP* terus bertambah, dan sampai saat ini versi terbaru *PHP* yang telah dikeluarkan adalah *PHP* 5.0.(Dianing,2012:5-6)

**2.1.10. Sekilas Tentang *MySQL***

*MySQL* merupakan sistem *database* yang banyak digunakan untuk pengembangan aplikasi *web.* Alasannya mungkin karena gratis, pengolahan datanya sederhana, memiliki tingkat keamanan yang bagus, mudah diperoleh, dan lain-lain(Budi,dkk,2010:216). *Mysql* juga berperan sebagai *client* sehingga sering disebut *database* *client/server,* yang *open sourse* dapat berjalan baik di *OS (Operating System)* manapun, dengan *platform windows* maupun *linux.* Selain itu *database* ini memiliki beberapa kelebihan dibanding *database* lain, di antaranya adalah

1. *MySQL* sebagai *Data Management System (DBMS)*
2. *MySQL* sebagai *Relation Database Management System* *(RDBMS)*
3. *MySQL* merupakan sebuah *database server,* jadi dengan menggunakan *database* ini anda dapat menghubungannya ke media internet sehingga dapat diakses dari jauh.

**2.1.11. Data *Flow Diagram (DFD)***

Data *flow* diagram/diagram alir data *(DFD/DAD)* adalah sebuah teknis grafis yang mengambarkan aliran informasi dan transformasi yang diaplikasikan pada saat bergerak dari *input* menjadi *output*. *DFD* dikenal sebagai grafik aliran data atau *bubble chart*. (Rosa dan Shalahuddin, 2011:64).

Tujuan data *flow diagram (DFD),* yaitu:

1. Untuk memberikan indikasi mengenai bagaimana data ditransformasikan pada saat data bergerak melalui sistem.
2. Untuk mengambarkan fungsi-fungsi *(subsistem)* yang mentransformasikan aliran data.

Simbol-simbol yang digunakan seperti yang terlihat pada table 2.1 berikut ini :

**Table 2.2** Simbol-Simbol Pada *DFD/DAD*

|  |  |  |
| --- | --- | --- |
| NO | Simbol | Keterangan |
| 1 |  | **Terminal**  Merupakan eksternal entity atau kesatuan luar yang merupakan sumber tujuan data. Terminator dapat digambarkan dengan suatu notasi kotak. |
| 2 |  | **Arus data** adalah alat untuk menunjukan arus data yang dapat berupa masukan atau hasil dari proses sistem mengalir antara proses, simpanan data dan kesatuan luar. Arah panah menggambarkan arah dari data. |
| 3 |  | **Proses** Menggambarkan bagian dari sistem yang mentranformasikan input data menjadi output data atau dapat diartikan sebagai kegiatan yang dilakukan oleh orang mesin atau komputer. |
| 4 |  | **Data store**  Merupakan sarana yang digunakan untuk menyimpan data. |

**2.1.12 *Entity Relation Diagram (ERD)***

Model *Entity relation diagram (ERD)* adalah notasi yang digunakan untuk melakukan aktivitas pemodelan data. Tujuan utama dari *Entity relation diagram (ERD)* adalah mewakili objek data dan hubungan mereka(Fathansyah,2012:81).

Komponen utama identifikasi untuk *Entity relation diagram (ERD)* berupa :

* 1. *Entitas*

Adalah reprensentasi dari hampir semua informasi gabungan yang harus dipahami oleh perangkat lunak, dengan informasi gabungan dapat diartikan sesuatu yang memiliki sejumlah sifat dan atribut yang berbeda. Objek data diwakili oleh sebuah persegi panjang yang diberi label. Objek data dihubungkan satu dengan yang lainnya, hubungan itu ditentukan oleh konteks masalah yang sedang dianalisis, objek data berupa :

1. Entitas eksternal (misal semua yang menghasilkan informasi)
2. Benda (berupa laporan)
3. Tempat (misal gudang ) dan sebagainya.
   1. Atribut

Atribut menetukan properti suatu objek data dan mengambil salah satu dari tiga karakteristik yang berbeda.

* 1. *Relationship*

Hubungan ditunjukkan dengan garis yang diberi label yang menghubungkan objek. Sambungan antara objek dan hubungan dibangun dengan menggunakan kardinalitas dan modalitas.

* 1. *link* adalahadalah tanda garis yang digunakan untuk menghubungkan komponen-komponen *ERD.*

**Table 2.3** Simbol-Simbol Pada *Entity Relation Diagram (ERD)*

|  |  |  |
| --- | --- | --- |
| No | Simbol | Keterangan |
| 1 |  | ***Entitas*** adalah suatu objek yang ada pada dunia nyata dan dapat dibedakan dari objek lainnya yang di definisikan secara unik. Entitas dapat berupa lingkungan elemen, *resource*, atau suatu transaksi yang sangat penting. *Entity* disimbolkan dengan persegi panjang. |
| 2 |  | ***Relationship*** adalah hubungan yang terjadi antara satu *entitas* atau lebih dan digambarkan dengan suatu prisma yang diberi label berbentuk kata kerja. |
| 3 |  | ***Link*** adalah tanda garis yang digunakan untuk menghubungkan komponen-komponen *ERD* |
| 4 |  | ***Atribut*** adalah menentukan property suatu objek data dan mengambil salah satu dari tiga karakteristik yang berbeda. |

**2.1.13 Flowchart**

Merupakan penggambaran secara grafik untuk mempermudah penyelesaian masalah khususnya yang perlu dipelajari dan dievaluasi lebih lanjut.

Jenis-jenis flowchart terdiri atas:

1. System Flowchart
2. Document Flowchart
3. Schematic Flowchart
4. Program Flowchart
5. Process Flowchart

**Tabel 2.4.** Notasi Document Flowchart

|  |  |  |  |
| --- | --- | --- | --- |
| SIMBOL  NOTASI | DESKRIPSI | SIMBOL  NOTASI | DESKRIPSI |
|  | Terminal showing source Or destination of documents And reports |  | Document flowline |
|  | Source document or report |  | On-page connector |
|  | Manual operation |  | Off-page connector |
|  | File for storing source documents and  reports |  | Accounting records  (journals, registers  logs, ledgers) |

**Gambar 2.4.** Notasi Document Flowchart

**2.2. Penelitian Sebelumnya**

Penelitian sebelumnya dilakukan dengan cara memberi informasi-informasi serta referensi-referensi mengenai hal yang akan diteliti. Dalam proses untuk mendukung dalam melaksanakan tugas akhir berikut ini jurnal yang telah melakukan penelitian sebelumnya sebagai referensi.

Menurut Marissa Tio Minar Butar-Butar dengan judul penelitian **PEMBANGUNAN APLIKASI PENGOLAHAN DATA PASIEN BERBASIS WEB (STUDI KASUS : RUMAH SAKIT IBU KARTINI KISARAN).** Rumah Sakit Ibu Kartini Kisaran adalah salah satu rumah sakit swasta yang telah berdiri sekian lama di kota Kisaran dan telah menanganin banyak pasien rawat inap dan rawat jalan. Sebagai rumah sakit besar yang memiliki banyak data dan informasi yang harus dikelolah, Rumah Sakit Ibu Kartini membutuhkan sebuah aplikasi berbasis Teknologi Informasi yang dapat menangani pengolahan data pasien rawat inap dan rawat jalan secara menyeluruh. Rumah Sakit Ibu Kartini memiliki tiga jenis pasien yaitu pasien BSP, NON BSP, dan pasien umum. Admin yang bertugas merasakan cukup sulit melakukan pendataan pasien dengan cara yang masih manual sesuai jenis pasien tersebut. Pencarian data pasien baik yang di rawat inap maupun yang dirawat jalan, pencarian ruangan kamar inap yang masih kosong, pembuatan laporan bulanan jumlah data pasien yang dirawat inap maupun yang dirawat jalan masih menggunakan cara manual, Sehingga membutuhkan waktu yang cukup lama dalam melakukan pendataan pasien, pencarian data pasien, pencarian ruangan kamar inap pasien, data pembuatan laporan bulanan. Oleh karena itu, proyek akhir ini bertujuan untuk memberikan solusi bagi permasalahan tersebut. Sistem yang akan dibuat di beri judul “Pembagunan Aplikasi Pengolahan Data Pasien Berbasis Web Pada Rumah Sakit Ibu Kartini ”. Sistem ini berupa aplikasi online yang mengunakan bahasa pemprograman php dengan framework codeigniter dan MySQL sebagai databasenya. Metode pengerjaan proyek ini menggunakan metode SDLC. Pada aplikasi ini terdapat fungsionalitas yang memudahkan admin untuk melakukan pengolahan data pasien yang terdapat didalamnya. Kata Kunci: Aplikasi, Pengolahan data, SDLC.

Menurut Emy Budi Susilowati dan Bambang Eka Purnama dengan judul penelitian **“ANALISIS DAN PERANCANGAN SISTEM INFORMASI PASIEN RUMAH SAKIT UMUM NIRMALA SURI SUKOHATJO”.** Salah satu perusahaan industri jasa yang membutuhkan komputer sebagai alat pemprosesan dan penyajian informasi adalah rumah sakit. Rumah sakit sebagai salah satu sebagai salah satu sektor yang berkembang dengan cepat seiring sengan bertambahnya jumlah penduduk. Kualitas pelayanan kepada pasien merupakan hal pokok yang menjadi titik sentral setiap industri jasa pelayanan kesehatan seperti poliklinik. Sistem yang dibuat dapat berjalan baik pada komputer dengan prosesor 486DX atau yang lebih baru, dengan memori 4 MB atau lebih VGA card 1MB atau lebih. Sistem yang dibuat mengandalkan objek basis data. Sehingga dengan data yang sudah tersimpan akan dapat diproses menurut kebutuhan dan diproses lebih lanjut lagi untuk keperluan strategis tertentu.

**BAB III**

**ANALISIS DAN PERANCANGAN**

**3.1. Analisis**

Pada tahap ini akan dijelaskan mengenai analisis sistem yang berjalan dan proses pengumpulan kebutuhan dan imformasi yang akan didukung oleh perancangan sistem basis data dan mengunakan informasi tersebut untuk mengidentifikasi kebutuhan *user* terhadap sistem yang baru.

**3.2. Analisis Sistem Yang Berjalan**

Hasil pengamatan dan analisis yang dilakukan, permaslahan yang ada pada sistem yang berjalan pada saat ini adalah masih kurang efektif dan efisiennya kepada calon pasien untuk berobat karna dalam registrasi data pasien masih mengunakan kartu atau formulir dan kartu tersebut dicatat kembali mengunakan buku lalu. Selain data pasien data lainnya seperti data rekap medis, SKD (Surat Keterangan Dokter), untuk data obat pendataan tidak mengunakan buku, data tersebut mengunakan microsoft office excel.

**3.2.1. *Flowchart* Dokumen Sistem yang Sedang Berjalan**

**Tabel 3.1.** Tabel *Flowchart* Dokumen Sistem yang Sedang Berjalan

Berdasarkan *Flowchart* Dokumen Sistem yang Sedang Berjalan bagian paramedis mencatat data pasien yang telah mendaftar, setelah di setujui maka pasien diperiksa oleh paramedis. Setelah paramedis memeriksa data pasien dicatat kembali di rekap medis yang terdiri dari data obat dan data pasien. Selanjutnya semua data dicatat laporan diserahkan oleh kepala kesehatan.

**3.3. Rancangan Sistem**

Pada rancangan sistem akan dijelaskan gambaran secara umum sistem basis data yang akan diimplementasikan dengan model desain data *flow diagram* (DFD) dan *Entity Relasional Diagram* (ERD).

**3.3.1.** **Data *Flow* Diagram**

Data *flow* diagram/diagram alir data *(DFD/DAD)* adalah sebuah teknis grafis yang mengambarkan aliran informasi dan transformasi yang diaplikasikan pada saat bergerak dari *input* menjadi *output*. *DFD* dikenal sebagai grafik aliran data atau *bubble chart*.

**3.3.1.1. Diagram Konteks**

Dalam diagram konteks terdiri dari 2 entitias yaitu bagian paramedis dab kepala kesehatan pada Poliklinik Polresta Palembang.

**Gambar 3.1** Diagram Konteks

Paramedis memberikan masukan data yang berupa data SKD, rekap medis, paramedis, obat dan pasien sedangkan sistem memberikan keluaran berupa laporan-laporan yang di sampaikan ke kepala kesehatan.

**3.3.1.1. Diagram Level O**

Dalam diagram level 0 sistem dibagi menjadi beberapa proses antara lain input data pasien , input data rekap medis, input data paramedis, input data obat, input data SKD, dan proses pembuatan laporan mengunakan beberapa tabel antara lain pasien, paramedis, rekap medis, obat, SKD (Surat Keterangan Dokter).

|  |
| --- |
|  |

**Gambar 3.2.** Diagram Level O

**3.3.2 *Entity Relational Diagram (ERD)***

*Entity relationship diagram* merupakan gambaran singkat dari proses terjadinya sistem, ERD terdiri dari 5 entitas yaitu pasien, rekap medis, paramedis, obat, SKD (Surat Keterangan Dokter) yang sebagai berikut :

|  |
| --- |
|  |

**Gambar 3.3.** gambar ERD

**3.4. Perencanaan Basis Data (*Database Planning)***

**3.4.1. Pernyataan Misi**

Perancangan basis data pada Poliklinik Polresta Palembang bertujuan untuk menyimpan data dan mengolah data. Perancangan basis data dimaksud untuk memberikan kemudahan pelayanan data terhadap aktifitas Poliklinik dimana data-data yang akan disimpan di dalam sebuah basis data yang nantinya akan digunakan secara bersama-sama seluruh bagian yang berhubungan dengan Poliklinik.

**3.4.2. Pernyataan Tujuan**

Pernyataan tujuan dalam perancangan basis data ini dijabarkan dalam beberapa kebutuhan akan informasi yang ada pada Poliklinik Polresta Palembang. Adapun kebutuhan akan proses basis data antara lain untuk mengelola data-data pasien, rekap medis, paramedis, obat, SKD (Surat Keterangan Dokter). Selain itu sistem dapat melakukan pencarian data yang berhubungan dengan data-data Poliklinik.

**3.5. Definisi Sistem (*System Definition)***

Sistem ini mencakup data-data Poliklinik yang ada pada Polresta Palembang.

Registrasi

Pasien

Rekap medis

Obat

Paramedis

PAUR KES

SKD (Surat Keterangan Dokter)

**Gambar 3.4**. Definisi Sistem

Keterangan :

* Pasien melakukan registrasi.
* Bagian registrasi berhubungan dengan paramedis.
* Para medis behubungan dengan pasien, obat, rekap medis,SKD (Surat Keterangan Dokter) dan memberikan laporan dari PAUR KES.
* Obat berhubungan dengan paramedis dan laporan obat dilihat oleh PAUR KES.
* Rekap medis berhubungan dengan paramedis dan laporan rekap medis dilihat oleh PAUR KES.
* SKD (Surat Keterangan Dokter) berhubungan dengan paramedis dan laporan SKD (Surat Keterangan Dokter) dilihat oleh PAUR KES.
* PAUR KES menerima laporan dari paramedis dan melihat laporan data obat, rekap medis, dan SKD (Surat Keterangan Dokter).

**3.6. Analisis dan Pengumpulan Kebutuhan**

Dalam analisis dan pengumpulan kebutuhan dapat dijabarkan dalam tabel berikut :

**Tabel 3.2.** Analisis dan Pengumpulan Kebutuhan

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Tipe Akses | Para Medis | PAUR KES |
| Pasien | Query | X |  |
| View | X |  |
| Surat | X |  |
| Laporan |  | X |
| Rekap Medis | Query | X |  |
| View |  | X |
| Surat |  |  |
| Laporan |  |  |
| Paramedis | Query | X | X |
| View | X | X |
| Surat |  |  |
| Laporan |  |  |
| Obat | Query |  |  |
| View | X |  |
| Surat |  |  |
| Laporan |  | X |
| SKD (Surat Keterangan Dokter) | Query | X |  |
| View |  |  |
| Surat |  |  |
| Laporan |  |  |

**3.7. Perancangan Basis Data (*Design Database)***

Dalam tahap perancangan basis data terdiri dari 3 tahapan yang disesuaikan dengan kebutuhan informasi dari Poliklinik Polresta Palembang. Tiga buah tahapan perancangan basis data tersebut, yaitu :

1. Perancangan basis data konseptual
2. Perancangan basis data logikal
3. Perancangan basis data fisikal

**3.7.1. Perancangan Basis Data Konseptual**

Perancangan basis data konseptual merupakan suatu proses pembentukan model yang berasal dari informasi yang digunakan dalam perusahaan yang bersifat independen dari kebutuhan aspek fisik. Langkah-langkah yang digunakan dalam perancangan basis data konseptual adalah sebagai berikut :

1. Mengidentifikasi tipe entitas
2. Mengidentifikasi tipe relasi
3. Mengidentifikasi atribut dengan tipe entitas
4. Menentukan domain atribut
5. Menentukan atribut *candidate key* dan *primary key*

**3.7.1.1. Identifikasi Tipe Entitas**

Entitas-entitas yang terkait dalam perancangan basis data pada Poliklinik Polresta Palembang dijelaskan dalam tabel 3.2.

**Tabel 3.3.** Identifikasi Tipe Entitas

|  |  |  |
| --- | --- | --- |
| Nama Entitas | Deskripsi | Kejadian |
| Pasien | Mendeskripsikan orang yang mendaftar untuk berobat | Setiap pasien dapat melakukan satu kali atau lebih pendaftaran |
| Paramedis | Mendeskripsikan kegiatan yang dilakukan pasien | Setiap para medis dapat dilakukan lebih dari satu kali kepada pasien |
| Obat | Mendeskripsikan obat yang diberikan kepada pasien | Obat dapat diberikan beberapa kali kepada pasien |
| Rekap medis | Mendeskripsikan histori pengobatan yang dilakukan kepada pasien | Setiap pasien memiliki lebih dari satu kali rekap medis |
| SKD (Surat Keterangan Dokter) | Mendeskripsikan memberikan keterangan pada dokter | Setiap pasien bisa lebih satu dari SKD |

**3.7.1.2. Identifikasi Tipe Relasi**

Setelah dilakukan dan didapat entitas-entitas yang terkait. Tahapan berikutnya yaitu melakukan identifikasi relasi atau hubungan antar entitas-entitas tersebut. Identifikasi tipe relasi dapat dilihat pada tabel 3.3.

**Tabel 3.4.** Identifikasi Tipe Relasi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nama Entitas | Multiplicity | Hubungan | Nama Entitas | Multiplicity |
| Pasien | 1..1 | Memiliki | Rekap Medis | 1..1 |
| Paramedis | 1..1  1..1 | Membuat  Membuat | Rekap Medis  SKD | 1..1  1..1 |
| Obat | 1..\* | Melibatkan | Rekap Medis | 1..\* |
| Rekap\_medis | 1..1  1..1  1..1 | Melibatkan  Melibatkan  Melibatkan | Pasien  Para Medis  Obat | 1..1  1..1  1..\* |
| SKD | 1..1 | Melibatkan | Paramedis | 1..1 |

**3.7.1.3. ERD Konseptual Awal**

Setelah dilakukan identifikasi tipe relasi terbentuk ERD konseptual awal yang sebagai berikut :

|  |
| --- |
|  |

**Gambar 3.5.** ERD Konseptual Awal

**3.7.1.4. Identifikasi Atribut Dari Tipe Entitas**

Atribut yang diperlukan untuk masing-masing entitas dalam perancangan basis data pada Poliklinik Polresta Palembang dijelaskan didalam tabel 3.4.

**Tabel 3.5.** Tabel Atribut Dari Tipe Entitas

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Nama Entitas | Atribut | Deskripsi | Tipe Data & Ukuran | Multi-Value | Null |
| 1 | Pasien | Kd\_pasien | Kode pasien | Varchar(5) | No | No |
| Nm\_pasien | Nama pasien | Varchar(30) | No | No |
| Pangkat\_nrp | Pangkat/NRP | Varchar(30) | No | No |
| Kesatuan | Kesatuan | Varchar(30) | No | No |
| Alamat | Alamat | Varchar(100) | No | No |
| Istri\_anak | Istri/anak | Varchar(100) | No | No |
| Anak\_ke | Anak ke | Varchar(2) | No | No |
| Umur | Umur | Int(11) | No | No |
| Jk | Jenis kelamin | Varchar(1) | No | No |
| Agama | Agama | Varchar(20) | No | No |
| 2 | Paramedis | Kd\_medis | Kode medis | Varchar(5) | No | No |
| Nama\_medis | Nama medis | Varchar(30) | No | No |
| Jk | Jenis kelamin | Varchar(1) | No | No |
| Jabatan | Jabatan | Varchar(100) | No | No |
| Tgl\_lahir | Tanggal lahir | Date | No | No |
| Agama | agama | Varchar(30) | No | No |
| 3 | Obat | Kd\_obat | Kode obat | Varchar(5) | No | No |
| Nm\_obat | Nama obat | Varchar(30) | No | No |
| Satuan | Satuan | Varchar(20) | No | No |
| Harga\_satuan | Harga satuan | Int(11) | No | No |
| 4 | Rekap\_medis | Kd\_rekap | Kode rekap | Varchar(5) | No | No |
| Tgl | Tanggal | Date | No | No |
| Kd\_pasien | Kode pasien | Varchar(5) | No | No |
| Keluhan | Keluhan | Varchar(100) | No | No |
| Kd\_medis | Kode medis | Varchar(5) | No | No |
| Kd\_obat | Kode obat | Varchar(5) | No | No |
| Jumlah\_obat | Jumlah obat | Int(20) | No | No |
| 5 | SKD | Kd\_surat | Kode surat | Varchar(5) | No | No |
| Tgl | Tanggal | Date | No | No |
| Nama | Nama | Varchar(30) | No | No |
| Pangkat\_nrp | Pangkat/NRP | Varchar(30) | No | No |
| Jabatan | Jabatan | Varchar(20) | No | No |
| Kesatuan | Kesatuan | Varchar(30) | No | No |
| Diagnosa | Diagnosa | Varchar(50) | No | No |
| Kd\_medis | Kode medis | Varchar(5) | No | No |

**3.7.1.5. Menentukan Domain Atribut**

Entitas-entitas yang telah dilakukan identifikasi tipe relasi kemudian dilakukan penentuan domain atribut. Domain atribut merupakan penentuan tipe data dari setiap atribut yang ada dalam tiap-tiap entitas.

**Tabel 3.6.** Tabel Domain Atribut

|  |  |  |  |
| --- | --- | --- | --- |
| NO | Nama Entitas | Atribut | Domain Atribut |
| 1 | Pasien | Kd\_pasien | Varchar(5) |
| Nm\_pasien | Varchar(30) |
| Pangkat\_nrp | Varchar(30) |
| Kesatuan | Varchar(30) |
| Alamat | Varchar(100) |
| Istri\_anak | Varchar(100) |
| Anak\_ke | Varchar(2) |
| Umur | Int(11) |
| Jk | Varchar(1) |
| Agama | Varchar(20) |
| 2 | Paramedis | Kd\_medis | Varchar(5) |
| Nama\_medis | Varchar(30) |
| Jk | Varchar(1) |
| Jabatan | Varchar(100) |
| Tgl\_lahir | Date |
| agama | Varchar(30) |
| 3 | Obat | Kd\_obat | Varchar(5) |
| Nm\_obat | Varchar(30) |
| Satuan | Varchar(20) |
| Harga\_satuan | Int(11) |
| 4 | Rekap\_medis | Kd\_rekap | Varchar(5) |
| Tgl | Date |
| Kd\_pasien | Varchar(5) |
| Keluhan | Varchar(100) |
| Kd\_medis | Varchar(5) |
| Kd\_obat | Varchar(5) |
| Jumlah\_obat | Int(20) |
| 5 | SKD | Kd\_surat | Varchar(5) |
| Tgl | Date |
| Nama | Varchar(30) |
| Pangkat\_nrp | Varchar(30) |
| Jabatan | Varchar(20) |
| Kesatuan | Varchar(30) |
| Diagnosa | Varchar(50) |
| Kd\_medis | Varchar(5) |

**3.7.1.6. Atribut-atribut *Candidate Key* dan *Primary Key***

Atribut yang terdapat didalam tiap-tiap entitas kemudian dilakukan pemberian *primary key*. Pemberian *primary key* diawali dengan dilakukan pengelompokkan terhadap atribut-atribut yang dimungkinkan untuk dijadikan *primary key*. Atribut yang tergolong kedalam calon *primary key* disebut *candidate key.*

**Tabel 3.7.** Atribute *Candidate key* dan *Primary key*

|  |  |  |
| --- | --- | --- |
| Nama entitas | Candidate key | Primary key |
| Pasien | Kd\_pasien | Kd\_pasien |
| Paramedis | Kd\_medis | Kd\_paramedis |
| Obat | Kd\_obat | Kd\_obat |
| Rekap\_medis | Kd\_rekap  Kd\_pasien  Kd\_obat  Kd\_medis | Kd\_rekapmedis |
| SKD | Kd\_surat  Kd\_paramedis | Kd\_surat |

**3.7.1.6. ERD Konseptual Dengan Primary Key**

Setelah menentukan setiap entitas-entitas yang dilakukan pemberian primary key terbentuk ERD konseptual dengan Primary key sebagai berikut :

|  |
| --- |
|  |

**Gambar 3.6.** ERD Konseptual dengan Primary Key

**3.7.2. Perancangan Basis Data Logical**

Perancangan basis data logikal merupakan lanjutan dari tahapan konseptual. Pada tahapan perancangan logikal akan menggambarkan apa yang sebenarnya disimpan dalam basis data dan hubungannya. Langkah-langkah yang digunakan dalam perancangan basis data logikal adalah sebagai berikut :

1. Menghilangkan fitur yang tidak kompatibel dengan model relasional.
2. Menurunkan relasi untuk model data *logical local.*
3. Validasi relasi menggunakan normalisasi.

**3.7.2.1. Menghilangkan Fitur Yang Tidak Kompatibel Dengan Relasional**

1. Menghilangkan tipe relasi *binary many to many* (\*:\*)

Pada model konseptual, terdapat beberapa relasi *biner many to many,* antara lain:

1. Relasi *many to many* antara entitas Rekapmedis dan Obat membentuk entitas baru yaitu Obat\_Rekap\_Medis
2. Menghilangkan Atribut *Multi-Valued*

Pada model konseptual, terdapat beberapa entitas yang memiliki atrinut nilai jamak *(multi-valued).* Hal tersebut tidak diperbolehkan didalam model relasioanl. Oleh karna itu proses dekomposisi pada atribut tersebut dan entitas yang terpisah. Proses menghilangkan *multi-valued* dapat dilihat pada gambar berikut :

1. Atribut *multi-valued* kd\_obat, jumlah\_obat pada entitas rekap medis

|  |
| --- |
|  |

**Gambar 3.7.** Penghilangan Atribut *Multi-valued*

**3.7.2.2. Menurunkan Relasi untuk Model Data *Logocal Local***

1. Tipe Entitas Kuat

Strong entity merupakan entitas yang tidak bergantung pada entitas lainnya. Berikut ini adalah himpunan entitas kuat :

1. **Pasien** (Kd\_pasien, Nm\_pasien, Pangkat\_nrp, Kesatuan, Alamat, Istri\_anak, Anak\_keUmur, Jk, Agama)

***Primary key*** kd\_pasien

1. **Paramedis** (Kd\_medis, Nama\_medis, Jk, Jabatan, Tgl\_lahir, agama)

***Primary key*** kd\_medis

1. **Obat** (Kd\_obat, Nm\_obat, Satuan,Harga\_satuan)

***Primary key*** kd\_obat

1. **Rekap\_medis** (Kd\_rekap,Tgl, Kd\_pasien, Keluhan, Kd\_medis)

***Primary key*** kd\_rekap

***Foreign key*** kd\_pasien ***references*** pasien (kd\_pasien)

***Foreign key*** kd\_medis ***references*** para medis (kd\_medis)

1. **SKD (**Kd\_surat, Tgl, Nama, Pangkat\_nrp, Jabatan, Kesatuan, Diagnosa, Kd\_medis

***Primary key*** kd\_surat

***Foreign key*** kd\_medis ***references*** rekap para medis (kd\_medis)

1. Tipe Entitas Lemah

Tipe entitas lemah merupakan entitas yang kemunculannnya tergantung pada ekesistensinya dalam sebuah relasi terhadap entitas lain (enitas kuat). Berikut adalah himpunan entitas lemah :

1. **Obat\_rekap\_medis** (kd\_rekap, kd\_obat, jumlah\_obat)

***Primary key*** kd\_rekap, kd\_obat

***Foreign key*** kd\_rekap ***references*** rekap medis (kd\_rekap)

***Foreign key*** kd\_obat ***references*** obat (kd\_obat)

1. Tipe Hubungan Biner *One-to-Many* (1:\*)

Untuk tiap relasi binary 1:\*, *entity* pada *“one side*” ditunjuk sebagai *parent entity* dan *entity pada “many side”* sebagai *child entity*. Untuk mewakili relasi ini, salinan dari atribut *primary key* pada *parent entity* dikirim ke dalam relasi yang *mewakili child entity*, untuk berlaku sebagai *foreign key*.

1. Tipe Hubungan Biner *One-to-one* (1:1)

Relasi binary 1:1, *entity* pada *“one side*” ditunjuk sebagai *parent entity* dan *entity pada “many side”* sebagai *child entity*. Untuk mewakili relasi ini, salinan dari atribut *primary key* pada *parent entity* dikirim ke dalam relasi yang *mewakili child entity*, untuk berlaku sebagai *foreign key*.

1. Tipe Hubungan Biner *many-to-many* (\*:\*)

Pada tipe hubungan biner many-to-many akan terbentuk entitas baru, dan untuk yang lain sama seperti pola diatas.

Berikut ini ditunjukan bagaimana entitas kuat mengirimkan primary key-nya ke dalam entitas lemah sehingga terjadi hubungan dan relasional yang berkaitan. Maka didapatkan hasil relasi dari proses menentukan model data logikal lokal seperti berikut :

**Tabel 3.8.** Entitas Kuat dan Lemah

|  |  |
| --- | --- |
| **Entitas** | **Atrribut** |
| **Pasien** | Kd\_pasien, Nm\_pasien, Pangkat\_nrp, Kesatuan, Alamat, Istri\_anak, Anak\_ke, Umur, Jk, Agama  ***Primary key*** kd\_pasien |
| **Paramedis** | Kd\_medis, Nama\_medis, Jk, Jabatan, Tgl\_lahir, agama  ***Primary key*** kd\_medis |
| **Obat** | Kd\_obat, Nm\_obat, Satuan,Harga\_satuan  ***Primary key*** kd\_obat |
| **Rekap\_medis** | Kd\_rekap,Tgl, Kd\_pasien, Keluhan, Kd\_medis  ***Primary key*** kd\_rekap  ***Foreign key*** kd\_pasien ***references*** pasien (kd\_pasien)  ***Foreign key*** kd\_medis ***references*** para medis (kd\_medis) |
| **SKD** | Kd\_surat, Tgl, Nama, Pangkat\_nrp, Jabatan, Kesatuan, Diagnosa, Kd\_medis  ***Primary key*** kd\_surat  ***Foreign key*** kd\_medis ***references*** rekap para medis (kd\_medis) |
| **Obat\_rekap\_medis** | kd\_rekap, kd\_obat, jumlah\_obat  ***Primary key*** kd\_rekap, kd\_obat  ***Foreign key*** kd\_rekap ***references*** rekap medis (kd\_rekap)  ***Foreign key*** kd\_obat ***references*** obat (kd\_obat) |

**3.7.2.3. Validasi Relasi dengan Normalisasi**

Proses normalisasi disini berfungsi untuk meminimalisasi penggunaan atribut yang berulang sehingga duplikasi pada tiap atribut semakin kecil. Adapun data-data yang memerlukan normalisasi adalah tabel Rekap medis yang sebagai berikut :

**1. UNF**

Bentuk relasi ini harus dihindari dalam peran cangun relasi dalam basis data. Relasi UNF mempunyai kriteria sebagi berikut :

1. Jika relasi mempunyai bentuk *non flat file* (terjadi akibat data disimpan sesuai dengan kedatangan, sehingga tidak memiliki struktur yang sama/tertentu, terjadi duplikasi atau tidak lengkap.
2. Jika relasi memuat set atribut berulang (*non single value*)
3. Jika relasi memuat atribut *non atomic value.*

**Tabel 3.9.** Relasi UNF

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Kd\_rekap | tgl | Kd\_pasien | Keluhan | Kd\_medis | Kd\_obat | Jumlah\_obat |
| 001 | 1-09-2012 | 011 | Batuk | 01 | 444 | 2 |
|  |  |  |  |  | 114 | 20 |
|  |  |  |  |  | 238 | 30 |
|  |  |  |  |  | 269 | 20 |
| 002 | 1-09-2012 | 012 | Lesu,nyeri nelan -+ 3hr | 03 | 292 | 10 |
|  |  |  |  |  | 231 | 10 |
|  |  |  |  |  | 79 | 10 |
|  |  |  |  |  | 111 | 12 |
| 003 | 1-09-2012 | 013 | Sakit pada kaki | 05 | 422 | 2 |
|  |  |  |  |  | 84 | 20 |
|  |  |  |  |  | 251 | 15 |
|  |  |  |  |  | 367 | 4 |
|  |  |  |  |  | 238 | 30 |

**2. 1NF**

Untuk mengubah relasi UNF menjadi bentuk 1NF, dapat dilakukan dengan cara sebagai berikut:

1. Melengkapi nilai-nilai dalam atribut.
2. Mengubah struktur relasi.

**Tabel 3.10.** Relasi 1NF

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Kd\_rekap | tgl | Kd\_pasien | Keluhan | Kd\_medis | Kd\_obat | Jumlah\_obat |
| 001 | 1-09-2012 | 011 | Batuk | 01 | 444 | 2 |
| 001 | 1-09-2012 | 011 | Batuk | 01 | 114 | 20 |
| 001 | 1-09-2012 | 011 | Batuk | 01 | 238 | 30 |
| 001 | 1-09-2012 | 011 | Batuk | 01 | 269 | 20 |
| 002 | 1-09-2012 | 012 | Lesu,nyeri nelan -+ 3hr | 03 | 292 | 10 |
| 002 | 1-09-2012 | 012 | Lesu,nyeri nelan -+ 3hr | 03 | 231 | 10 |
| 002 | 1-09-2012 | 012 | Lesu,nyeri nelan -+ 3hr | 03 | 79 | 10 |
| 002 | 1-09-2012 | 012 | Lesu,nyeri nelan -+ 3hr | 03 | 111 | 12 |
| 003 | 1-09-2012 | 013 | Sakit pada kaki | 05 | 422 | 2 |
| 003 | 1-09-2012 | 013 | Sakit pada kaki | 05 | 84 | 20 |
| 003 | 1-09-2012 | 013 | Sakit pada kaki | 05 | 251 | 15 |
| 003 | 1-09-2012 | 013 | Sakit pada kaki | 05 | 367 | 4 |
| 033 | 1-09-2012 | 013 | Sakit pada kaki | 05 | 238 | 30 |

**3. 2NF**

Untuk mengubah relasi 1NF menjadi 2NF, dapat dilkukan dengan mengubah struktur relasi dengan cara :

1. Identifikasi FD relasi baru
2. Berdasarkan informasi-informasi tersebut. Dekomposisi relasi 1NF menjadi relasi-relasi baru sesuai FD-nya.

**Tabel 3.11.** Relasi 2NF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Kd\_rekap | Tgl | Kd\_pasien | keluhan | Kd\_medis |
| 001 | 1-09-2012 | 011 | Batuk | 01 |
| 002 | 1-09-2012 | 012 | Lesu,nyeri nelan -+ 3hr | 03 |
| 003 | 1-09-2012 | 013 | Sakit pada kaki | 05 |

**4. 3NF**

Bentuk normal ketiga mengharuskan semua kolom pada tabel relasional tergantung hanya pada kunci utama. Sebuah tabel berada pada bentuk 3NF jika tabel suda berada pada 2NF dan setiap kolom yang bukan kunci tidak tergantung secara transitif pada kunci utamanya.

**Tabel 3.12.** Relasi 3NF

|  |  |  |
| --- | --- | --- |
| Kd\_rekap | Kd\_obat | Jumlah\_obat |
| 01 | 444 | 2 |
| 01 | 114 | 20 |
| 01 | 238 | 30 |
| 01 | 269 | 20 |
| 03 | 292 | 10 |
| 03 | 231 | 10 |
| 03 | 79 | 10 |
| 03 | 111 | 12 |
| 05 | 422 | 2 |
| 05 | 84 | 20 |
| 05 | 251 | 15 |
| 05 | 367 | 4 |
| 05 | 238 | 30 |

**3.7.2.4. Struktur Tabel Hasil Normalisasi**

Struktur tabel merupakan rancangan tabel untuk menjelaskan tipe, ukuran dari *field-field* yang terdapat pada tabel tersebut. Rancangan tabel dalam sistem basis data ini antara lain :

1. Nama tabel : Pasien

PK : kd\_pasien

**Tabel 3.13**. Tabel Pasien

|  |  |
| --- | --- |
| Nama Field | Type |
| Kd\_pasien | Varchar(5) |
| Nm\_pasien | Varchar(30) |
| Pangkat\_NRP | Varchar(35) |
| Kesatuan | Varchar(30) |
| Alamat | Varchar(100) |
| Istri\_anak | Varchar(100) |
| Anak\_ke | Varchar(2) |
| Umur | Int(11) |
| Jk | Varchar(1) |
| Agama | Varchar(20) |

1. Nama tabel : Paramedis

PK : kd\_medis

**Tabel 3.14**. Tabel Paramedis

|  |  |
| --- | --- |
| Nama Field | Type |
| Kd\_medis | Varchar(5) |
| Nama\_medis | Varchar(30) |
| Jk | Varchar(1) |
| Jabatan | Varchar(100) |
| Tanggal\_lahir | Date |
| Agama | Varchar(30) |

1. Nama tabel : SKD (Surat Keterangan Dokter)

PK : kd\_surat

FK : kd\_medis

**Tabel 3.15**. Tabel SKD (Surat Keterangan Dokter)

|  |  |
| --- | --- |
| Nama Field | Type |
| Kd\_surat | Varchar(5) |
| Tanggal | Date |
| Nama | Varchar(30) |
| Pangkat\_NRP | Varchar(30) |
| Jabatan | Varchar(20) |
| Kesatuan | Varchar(30) |
| Diagnosa | Varchar(50) |
| Kd\_medis | Varchar(5) |

1. Nama tabel : Obat

PK : kd\_obat

**Tabel 3.16**. Tabel Obat

|  |  |
| --- | --- |
| Nama Field | Type |
| Kd\_obat | Varchar(5) |
| Nm\_obat | Varchar(30) |
| Satuan | Varchar(20) |
| Harga\_satuan | Int(11) |

1. Nama tabel : Rekap Medis

PK : kd\_rekap

FK : kd\_pasien, kd\_medis

**Tabel 3.17**. Tabel Rekap Medis

|  |  |
| --- | --- |
| Nama Field | Type |
| Kd\_rekap | Varchar(5) |
| Tgl | Date |
| Kd\_pasien | Varchar(5) |
| Keluhan | Varchar(100) |
| Kd\_medis | Varchar(5) |

1. Nama tabel : Obat Rekap Medis

PK : kd\_rekap, kd\_obat

FK : kd\_rekap

**Tabel 3.18**. Tabel Obat Rekap Medis

|  |  |
| --- | --- |
| Nama Field | Type |
| Kd\_rekap | Varchar(5) |
| Kd\_obat | Varchar(5) |
| Jumlah\_obat | Int(11) |

**3.7.3. Perancangan Basis Data Fisikal**

Perancangan basis data fisikal merupakan proses yang menghasilakan deskripsi implementasi basis data pada penyimpanan sekunder. Tahapan-tahapan yang dilakukan didalam proses perancangan basis data fisikal ini adalah :

1. Melakukan desain relasi dasar
2. Analisis Transaksi
3. Estimasi kebutuhan disk space
4. Mendesain mekanisme keamanan

**3.7.3.1. Estimasi Kebutuhan *Disk Space***

Tujuan dari tahapan ini untuk memperkirakan besarnya tempat penyimpanan yang dibutuhkan basis data. Melakukan perkiraan berguna untuk mengetahui seberapa besar relasi dapat berkembang dan juga dapat mengatur pertumbuhan disk agar dapat menentukan bedar basis data pada masa depan.

**Tabel 3.19**. Tabel Ukuran Entitas

|  |  |  |  |
| --- | --- | --- | --- |
| Tabel Ukuran Entitas | | | |
| Tabel | Nama Atribut | Type Data | Ukuran |
| Pasien | Kd\_pasien | Varchar | 5 |
| Nm\_pasien | Varchar | 30 |
| Pangkat\_nrp | Varchar | 30 |
| Kesatuan | Varchar | 30 |
| Alamat | Varchar | 100 |
| Istri\_anak | Varchar | 100 |
| Anak\_ke | Varchar | 2 |
| Umur | Int | 11 |
| Jk | Varchar | 1 |
| Agama | Varchar | 20 |
| Kapasitas dari tabel pasien adalah 329 bytes  Diperkirakan selama 1 bulan terjadi penambahan 150 pasien  Dalam 5 bulan diperkirakan pertumbuhan tabel ini adalah 329\*150\*5=246750 bytes | | | |
| Para medis | Kd\_medis | varchar | 5 |
| Nama\_medis | Varchar | 30 |
| Jk | Varchar | 1 |
| Jabatan | Varchar | 100 |
| Tgl\_lahir | Date |  |
| Agama | Varchar | 30 |
| Kapasitas dari tabel para medis adalah 166 bytes | | | |
| Obat | Kd\_obat | Varchar | 5 |
| Nm\_obat | Varchar | 30 |
| Satuan | Varchar | 20 |
| Harga\_satuan | int | 11 |
| Kapasitas dari tabel obat adalah 66 bytes  Diperkirakan selama 1 bulan terjadi penambahan 500 obat  Dalam 5 bulan diperkirakan pertumbuhan tabel ini adalah 66\*500\*5= 165000 bytes | | | |
| SKD (Surat Keterangan Dokter) | Kd\_surat | Varchar | 5 |
| Tgl | Date |  |
| Nama | Varchar | 30 |
| Pangkat\_nrp | Varchar | 30 |
| Jabatan | Varchar | 20 |
| Kesatuan | Varchar | 30 |
| Diagnosa | Varchar | 50 |
| Kd\_medis | Varchar | 5 |
| Kapasitas dari tabel SKD adalah 170 bytes  Diperkirakan selama 1 tahun terjadi penambahan 100 surat  Dalam 5 tahun diperkirakan pertumbuhan tabel ini adalah 170\*100\*5= 85000 bytes | | | |
| Rekap medis | Kd\_rekap | Varchar | 5 |
| Tgl | Date |  |
| Kd\_pasien | Varchar | 5 |
| Keluhan | Varchar | 100 |
| Kd\_medis | Varchar | 5 |
| Kapasitas dari tabel Rekap medis adalah 119 bytes  Diperkirakan selama 1 bulan terjadi penambahan 250 rekap  Dalam 5 bulan diperkirakan pertumbuhan tabel ini adalah 119\*250\*5= 148750 bytes | | | |
| Obat rekap medis | Kd\_rekap | Varchar | 5 |
| Kd\_obat | Varchar | 5 |
| Jumlah\_obat | int | 11 |
| Kapasitas dari tabel obat Rekap medis adalah 19 bytes  Diperkirakan selama 1 bulan terjadi penambahan 250 rekap  Dalam 5 bulan diperkirakan pertumbuhan tabel ini adalah 19\*250\*5= 23750 bytes | | | |

**BAB IV**

**HASIL DAN PEMBAHASAN**

**4.1. Hasil**

Adapun hasil dari penelitian yang dilakukan ini ialah berupa rancangan basis data. Rancangan basis data ini memiliki tabel-tabel yang telah terbentuk dari proses fase-fase perancangan *konseptual, logical,* sampai *fisikal* yang telah diuraikan dibab sebelumnya dan diharapkan menjadi rancangan basis data yang baik, sehingga nantinya bisa dimanfaatkan dan untuk mempermudah perusahaan dalam pembuatan aplikasi pada Poliklinik Polresta Palembang nantinya.

Hasil dari perancangan basis data ini, dihasilkan 6 tabel basis data dan dimasukan ke dalam *database* yang bernama Poliklinik. Dan pengimplementasikan dari rancangan basis data yang terdiri atas 6 tabel yang dilakukan dengan menggunakan aplikasi XAMPP 1.6.3a dengan koneksi *server localhost.*

**Gambar 4.1.** Rancang Database Poliklinik Polresta Palembang

**4.2. Pembahasan**

Berikut ini akan dibahas mengenai rancangan basis data yang telah dibuat, pembahasan ini mencakup 6 struktur tabel yang telah terbentuk dengan disertai dengan tampilan tiap struktur tabel yang ada pada database Poliklinik.

**4.2.1. Tabel Master**

Tabel master adalah tabel yang terdiri dari tabel-tabel utama didalam database. Berikut tampilan dari tabel-tabel master pada database poliklinik :

**4.2.1. Tabel Pasien**

Tabel pasien merupakan tabel yang digunakan untuk menyimpan data pasien yang ada di poliklinik. Tabel ini terdiri dari 10 atribut yaitu kd\_pasien, nm\_pasien, pangkat\_nrp, kesatuan, alamat, istri\_anak, anak\_ke, umur, jk, agama. *Primary key* dari tabel ini ialah kd\_pasien.

**Tabel 4.1.** Struktur Tabel Pasien

|  |  |  |  |
| --- | --- | --- | --- |
| Nama Atribut | *Key* | *type* | *width* |
| Kd\_pasien | *Primary key* | Varchar | 5 |
| Nm\_pasien |  | Varchar | 30 |
| Pangkat\_nrp |  | Varchar | 30 |
| Kesatuan |  | Varchar | 30 |
| Alamat |  | Varchar | 100 |
| Istri\_anak |  | Varchar | 100 |
| Anak\_ke |  | Varchar | 2 |
| Umur |  | Int | 11 |
| Jk |  | Varchar | 1 |
| Agama |  | Varchar | 20 |
|  | | Total | 329 |

SQL *syntax create table* pada tabel barang :

CREATE TABLE `pasien` (

`kd\_pasien` VARCHAR( 5 ) NOT NULL ,

`nm\_pasien` VARCHAR( 30 ) NOT NULL ,

`pangkat\_nrp` VARCHAR( 30 ) NOT NULL ,

`kesatuan` VARCHAR( 30 ) NOT NULL ,

`alamat` VARCHAR( 100 ) NOT NULL ,

`istri\_anak` VARCHAR( 100 ) NOT NULL ,

`anak\_ke` VARCHAR( 2 ) NOT NULL ,

`umur` INT( 11 ) NOT NULL ,

`jk` VARCHAR( 1 ) NOT NULL ,

`agama` VARCHAR( 20 ) NOT NULL ,

PRIMARY KEY ( `kd\_pasien` )

) ENGINE = MYISAM ;

Hasil dari sql *syntax create table* diatas menghasilkan struktur tabel barang di xampp dibawah ini :

**Gambar 4.2.** Hasil Struktur Pasien

**4.2.2. Tabel Paramedis**

Tabel paramedis merupakan tabel yang digunakan untuk menyimpan data paramedis yang ada di poliklinik. Tabel ini terdiri dari 6 atribut yaitu kd\_medis, nama\_medis, jk, jabatan, tgl\_lahir, agama. *Primary key* dari tabel ini ialah kd\_medis.

**Tabel 4.2.** Struktur Tabel Paramedis

|  |  |  |  |
| --- | --- | --- | --- |
| Nama Atribut | *Key* | *type* | *width* |
| Kd\_medis | *Primary key* | Varchar | 5 |
| Nama\_medis |  | Varchar | 30 |
| Jk |  | Varchar | 1 |
| Jabatan |  | Varchar | 100 |
| Tgl\_lahir |  | Date |  |
| agama |  | Varchar | 30 |
|  | | Total | 166 |

SQL *syntax create table* pada tabel kategori paramedis

CREATE TABLE `paramedis` (

`kd\_medis` VARCHAR( 5 ) NOT NULL ,

`nama\_medis` VARCHAR( 30 ) NOT NULL ,

`jk` VARCHAR( 1 ) NOT NULL ,

`jabatan` VARCHAR( 100 ) NOT NULL ,

`tgl\_lahir` DATE NOT NULL ,

`agama` VARCHAR( 30 ) NOT NULL ,

PRIMARY KEY ( `kd\_medis` )

) ENGINE = MYISAM ;

Hasil dari sql *syntax create table* diatas menghasilkan struktur tabel kategori paramedis di xampp dibawah ini :

**Gambar 4.3.** Hasil Struktur Paramedis

**4.2.3. Tabel obat**

Tabel obat merupakan tabel yang digunakan untuk menyimpan data obat yang ada di poliklinik. Tabel ini terdiri dari 4 atribut yaitu kd\_obat, nm\_obat, jk, satuan, harga satuan. *Primary key* dari tabel ini ialah kd\_obat.

**Tabel 4.3.** Struktur Tabel Obat

|  |  |  |  |
| --- | --- | --- | --- |
| Nama Atribut | *Key* | *type* | *width* |
| Kd\_obat | *Primary key* | Varchar | 5 |
| Nm\_obat |  | Varchar | 30 |
| Satuan |  | Varchar | 20 |
| Harga\_satuan |  | Int | 11 |
|  | | Total | 66 |

SQL *syntax create table* pada tabel kategori barang :

CREATE TABLE `obat` (

`kd\_obat` VARCHAR( 5 ) NOT NULL ,

`nm\_obat` VARCHAR( 30 ) NOT NULL ,

`satuan` VARCHAR( 20 ) NOT NULL ,

`harga\_satuan` INT( 11 ) NOT NULL ,

PRIMARY KEY ( `kd\_obat` )

) ENGINE = MYISAM ;

Hasil dari sql *syntax create table* diatas menghasilkan struktur tabel kategori obat di xampp dibawah ini :

**Gambar 4.4.** Hasil Struktur Obat

**4.2.4. Tabel SKD (Surat Keterangan Dokter)**

Tabel SKD (Surat Keterangan Dokter) merupakan tabel yang digunakan untuk menyimpan data SKD yang ada di poliklinik. Tabel ini terdiri dari 8 atribut yaitu kd\_surat, tgl, nama, pangkat\_nrp, jabata, kesatuan,diagnosa, kd\_medis. *Primary key* dari tabel ini ialah kd\_surat dan atribut kd\_medis bertindak sebagi *foreign key*.

**Tabel 4.4.** Struktur Tabel SKD (Surat Keterangan Dokter)

|  |  |  |  |
| --- | --- | --- | --- |
| Nama Atribut | *Key* | *type* | *width* |
| Kd\_surat | *Primary key* | Varchart | 5 |
| Tgl |  | Date |  |
| Nama |  | Varchart | 30 |
| Pangkat\_nrp |  | Varchart | 30 |
| Jabatan |  | Varchart | 20 |
| Kesatuan |  | Varchart | 30 |
| Diagnosa |  | Varchart | 50 |
| Kd\_medis | *Foreign key* | Varchart | 5 |
|  | | Total | 170 |

SQL *syntax create table* pada tabel telpon pelanggan :

CREATE TABLE `skd` (

`kd\_surat` VARCHAR( 5 ) NOT NULL ,

`tgl` DATE NOT NULL ,

`nama` VARCHAR( 30 ) NOT NULL ,

`pangkat\_nrp` VARCHAR( 30 ) NOT NULL ,

`jabatan` VARCHAR( 20 ) NOT NULL ,

`kesatuan` VARCHAR( 30 ) NOT NULL ,

`diagnosa` VARCHAR( 50 ) NOT NULL ,

`kd\_medis` VARCHAR( 5 ) NOT NULL ,

PRIMARY KEY ( `kd\_surat` )

) ENGINE = MYISAM ;

Hasil dari sql *syntax create table* diatas menghasilkan struktur tabel SKD (Surat Keterangan Dokter) di xampp dibawah ini :

**Gambar 4.5.** Hasil Struktur SKD (Surat Keterangan Dokter)

**4.2.5. Tabel Rekap Medis**

Tabel rekap medis merupakan tabel yang digunakan untuk menyimpan data rekap medis yang ada di poliklinik. Tabel ini terdiri dari 5 atribut yaitu kd\_rekap, tgl, kd\_pasien, keluhan, kd\_medis. *Primary key* dari tabel ini ialah kd\_rekap, atribut kd\_pasien bertindak sebagi *foreign key* dan atribut kd\_medis bertindak sebagi *foreign key.*

**Tabel 4.5.** Struktur Tabel Rekap Medis

|  |  |  |  |
| --- | --- | --- | --- |
| Nama Atribut | *Key* | *type* | *width* |
| Kd\_rekap | *Primary key* | Varchar | 5 |
| Tgl |  | Date |  |
| Kd\_pasien | *Foreign key* | Varchar | 5 |
| Keluhan |  | Varchar | 100 |
| Kd\_pasien | *Foreign key* | Varchar | 5 |
|  | | Total | 115 |

SQL *syntax create table* pada tabel telpon pelanggan :

CREATE TABLE `rekap\_medis` (

`kd\_rekap` VARCHAR( 5 ) NOT NULL ,

`tgl` DATE NOT NULL ,

`kd\_pasien ` VARCHAR( 5 ) NOT NULL ,

`keluhan` VARCHAR( 100 ) NOT NULL ,

`kd\_medis` VARCHAR( 5 ) NOT NULL ,

PRIMARY KEY ( `kd\_rekap` )

) ENGINE = MYISAM

Hasil dari sql *syntax create table* diatas menghasilkan struktur tabel rekap medis di xampp dibawah ini :

**Gambar 4.6.** Hasil Struktur Rekap Medis

**4.2.6 Tabel Obat Rekap Medis**

Tabel rekap medis merupakan tabel yang digunakan untuk menyimpan data rekap medis yang ada di poliklinik. Tabel ini terdiri dari 3 atribut yaitu kd\_rekap, kd\_obat, jumlah\_obat. *Primary key* dari tabel ini ialah kd\_rekap dan kd\_obat, atribut kd\_rekap bertindak sebagi *foreign key*.

**Tabel 4.6.** Struktur Tabel Obat Rekap Medis

|  |  |  |  |
| --- | --- | --- | --- |
| Nama Atribut | *Key* | *type* | *width* |
| Kd\_rekap | *Primary key, foreign key* | Varchar | 5 |
| Kd\_obat | *Primary key* | Varchar | 5 |
| Jumlah\_obat |  | Int | 11 |
|  | | Total | 21 |

SQL *syntax create table* pada tabel telpon pelanggan :

CREATE TABLE `obat\_rekap\_medis` (

`kd\_rekap` VARCHAR( 5 ) NOT NULL ,

`kd\_obat` VARCHAR( 5 ) NOT NULL ,

`jumlah\_obat` INT( 11 ) NOT NULL ,

PRIMARY KEY ( `kd\_rekap` )

) ENGINE = MYISAM ;

Hasil dari sql *syntax create table* diatas menghasilkan struktur tabel obat rekap medis di xampp dibawah ini :

**Gambar 4.7.** Hasil Struktur Obat Rekap Medis

**4.3. Tampilan Interface**

Berikut ini akan membahas mengenai interface pada poliklinik yang telah disesuaikan dengan rancangan tabel basis data dan membuat sebuah form menggunakan bahasa pe,programan PHP.

**4.3.1. Tampilan Form Home (Menu)**

Tampilan ini merupakan interface pada tampilan menu Poliklinik Polresta Palembang.

**Gambar 4.8.** Tampilan Menu

**4.3.2. Tampilan Form Para Medis**

Tampilan ini merupakan interface pada tampilan form databases para medis di Poliklinik Polresta Palembang.

**Gambar 4.9.** Tampilan Form Para Medis

**4.3.3. Tampilan Form Obat**

Tampilan ini merupakan interface pada tampilan form databases obat di Poliklinik Polresta Palembang.

**Gambar 4.10.** Tampilan Form Obat

**4.3.4. Tampilan Form SKD (Surat Keterangan Dokter)**

Tampilan ini merupakan interface pada tampilan form databases SKD (Surat Keterangan Dokter).

di Poliklinik Polresta Palembang.

**Gambar 4.11.** Tampilan Form SKD (Surat Keterangan Dokter)

**BAB V**

**KESIMPULAN DAN SARAN**

**5.1. Kesimpulan**

Berdasarkan dari penelitian ini, penulis dapat mengambil kesimpulan sebagai berikut :

1. Dengan dilakukan Rancangan basis data diharapkan dapat menghasilkan suatu rancangan basis data Poliklinik dengan skema basis data yang baik dan dapat meminalisir duplikasi data.
2. Hasil dari rancangan basis data ini menghasilkan struktur basis data, dengan nama database Poliklinik dan terdiri atas 6 struktur tabel.

**5.2. Saran**

Dari kesimpulan diatas, ada beberapa saran yang penulis ingin sampaikan

1. Dengan rancangan basis data yang telah dihasilkan, instansi bisa memanfaatkan basis data tersebut dan mengembang aplikasi Poliklinik Polresta Palembang.
2. Penelitian ini disarankan dapat dipergunakan dan dikembangkan untuk penelitian selanjutnya.

**DAFTAR PUSTAKA**

Fathansyah. 2004. *Basis Data.* Bandung:Informatika

Hutabarat, Bernaridho. 2004. *Pengolahan Basis Data.* Yogyakarta:Andi

Ladjamuddin,Al-Bahra. 2004. *Konsep Sistem Basis Data dam Implementasi.* Yogyakarta: Graha Ilmu

Marlinda, Linda. 2004. *Sistem Basis Data.* Yogyakarta:Andi

P, Dianing. 2012. *Menguasai PHP MySQL Dalam Sekejap Tanpa Guru.* Jakarta:Kuncikom

Pressman, Rogers, S. 2002. *Rekayasa perangkat Lunak Buku I.* Yogyakarta:Andi

Raharjo, Buku, dkk. *Modul Pemprograman WEB (HTML, PHP, & MySQL).* Bandung:Modula

Indrajani. 2011. *Perancangan Basis Data Dalam ALL1.* Jakarta:PT Elex Media Komputindo