BAB III

OBJEK DAN METODE PENELITIAN

Pada bab III ini akan dibahas mengenai objek dan metode penelitian dimana terdapat metode pendekatan dan pengembangan sistem yang dibangun.

3.1 Objek Penelitian

Objek penelitian ini berisikan tentang objek – objek yang akan diteliti oleh penulis yang berhubungan dengan program aplikasi yang akan dibangun oleh penulis. Adapun objek – objek yang dicatat oleh penulis adalah sebagai berikut :

3.1.1 Sejarah Singkat Perusahaan

Pada tahun 1981, bidan Sri Sumarni masih belum mendirikan atau membuka tempat praktek sendiri. Beliau menerima panggilan dari rumah ke rumah untuk membantu melahirkan. Namun pada tahun 1982, bidan Sri Sumarni membuka tempat praktek di kediamannya yaitu di Jalan Tubagus Ismail 5 RT 001/ RW 009 Nomor 70 Bandung. Alasan dia membuka tempat praktek ini adalah untuk mengamalkan keahlian yang beliau miliki.

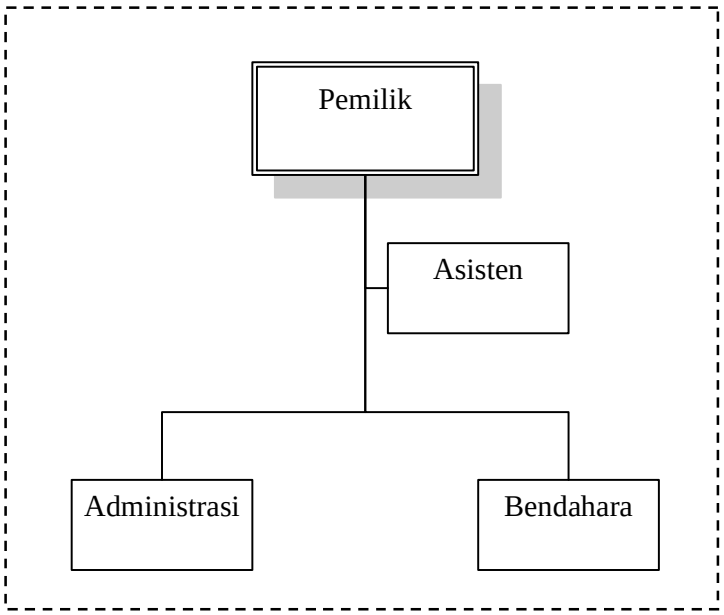
3.1.2 Visi dan Misi Perusahaan

Visi dan misi bidan Sri Sumarni dalam membuka tempat praktek ini adalah sebagai berikut :

1. Memanfaatkan ilmu kebidanan yang telah didapatkan selama sekolah dan sebagai suatu bentuk usaha.
2. Memberikan pelayanan semaksimal mungkin kepada pasien dengan penuh rasa kasih sayang dan tanggung jawab.

3.1.3 Struktur Organisasi Perusahaan

Struktur organisasi dalam tempat praktek bidan Sri Sumarni dapat dilihat pada gambar di bawah ini.



Gambar 3.1 Struktur Organisasi

3.1.4 Deskripsi Tugas (*Job Description*)

Adapun deskripsi tugas / *job description* untuk setiap bagian yang terlibat dalam perancangan sistem informasi ini adalah sebagai berikut :

1. Pemilik

   Orang yang memiliki perusahaan tersebut, dalam objek penelitian ini pemilik adalah bidan Sri Sumarni sendiri. Bertugas melayani pasien secara langsung dengan penuh rasa tanggung jawab.

2. Asisten

   Orang yang bertugas untuk membantu bidan Sri Sumarni dalam melayani pasien.

3. Administrasi

   Orang yang bertugas untuk melayani pendaftaran pasien.

4. Bendahara

   Orang yang bertugas untuk melayani pembayaran pasien.

3.2 Metode Penelitian

Metode penelitian yang digunakan adalah metode deskriptif. Metode deskriptif adalah memaparkan secara lengkap dan membuat gambaran secara sistematis mengenai keadaan tertentu.

3.2.1 Desain Penelitian

Untuk memperoleh data yang lengkap dan benar akan data yang dibutuhkan maka metode penelitian yang digunakan penulis dalam penyusunan

laporan ini adalah metode deskriptif yaitu suatu bentuk pengumpulan data yang bertujuan menggambarkan, menjelaskan suatu masalah kemudian menganalisis.

3.2.2 Jenis dan Metode Pengumpulan Data

Data yang diperlukan dalam penelitian ini berasal dari dua sumber yaitu data primer dan data sekunder. Data primer dikumpulkan dengan melakukan penelitian diloasi secara langsung. Data sekunder dikumpulkan melalui penelitian kepustakaan seperti dokumen – dokumen.

3.2.2.1 Sumber Data Primer

Data primer yang digunakan adalah data yang didapatkan melalui identifikasi dan informasi langsung dari pihak Bidan Sri Sumarni untuk mendapatkan data yang dapat mendukung penelitian ini. Sumber data primer dalam penelitian ini diperoleh melalui wawancara.

1. **Metode Wawancara / *interview***

   Dengan cara mengajukan pertanyaan dalam wawancara langsung ataupun tertulis, yaitu dengan mewawancarai pihak bidan Sri Sumarni untuk mendapatkan data maupun yang dianggap akan menambah perolehan atau kelengkapan data sehingga mempermudah penyelesaian laporan dan penelitian.

3.2.2.2 Sumber Data Sekunder

Data sekunder merupakan cara pengumpulan data dengan cara mempelajari data yang telah tersedia atau dikumpulkan / diberikan oleh pihak yang bersangkutan kepada penulis.

Cara yang digunakan adalah untuk mengumpulkan data sekunder adalah metode dokumentasi, yaitu dengan mengumpulkan data dan informasi yang diperlukan dari sumber – sumber kebanyakan dari materi sejenis dokumen yang berkenaan dengan masalah yang diteliti.

3.2.3 Metode Pendekatan dan Pengembangan Sistem

Suatu penelitian tidak akan berjalan dengan baik apabila tidak dilakukan dalam suatu proses yang teratur dan terarah. Oleh karena itu diperlukan suatu metode pendekatan dan pengembangan sistem untuk melaksanakan suatu penelitian. Metode pendekatan dan pengembangan sistem yang digunakan adalah metode deskriptif, yaitu metode dengan mengumpulkan, menjelaskan, menganalisis data yang diperoleh dan menggali permasalahan yang mungkin ada, dengan harapan memperoleh pengetahuan baru.

3.2.3.1 Metode Pendekatan Sistem

Metode pendekatan sistem yang digunakan adalah metode analisis dan perancangan berorientasi objek. Analisis berorientasi objek adalah tahapan untuk menganilisis spesifikasi atau kebutuhan akan sistem yang akan dibangun dengan

konsep berorientasi objek, apakah benar kebutuhan yang ada dapat diimplementasikan menjadi sebuah sistem berorientasi objek.

3.2.3.2 Metode Pengembangan Sistem

Penelitian ini menggunakan pendekatan pengembangan perangkat lunak dengan metode *prototype*, karena metode ini berfungsi sebagai sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak.

Model *prototype* dapat digunakan untuk menyambungkan kesalahpahaman *user* mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang diinginkan *user* kepada pengembang perangkat lunak.

Berikut adalah gambar dari model prototipe.



Gambar 3.2 Model Pengembangan *Prototype*

(Sumber : Rosa A.S & M. Shalahuddin 2011 : 30)

Model prototipe dimulai dari mengumpulkan kebutuhan *user* terhadap perangkat lunak yang akan dibuat. Lalu dibuatlah program prototipe agar *user* lebih terbayang dengan apa yang sebenarnya diinginkan. Program prototipe biasanya merupakan program yang belum jadi. Program ini biasanya merupakan program simulasi alur perangkat lunak sehingga tampak seperti perangkat lunak yang sudah jadi. Program prototipe ini dievaluasi oleh *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan *user*.

Mock – up adalah sesuatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi atau keperluan lain. Sebuah *mock – up* disebut sebagai prototipe perangkat lunak jika menyediakan atau mampu mendemonstrasikan sebagian besar fungsi sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak.

Seiring dengan mengembangkan prototipe maka sistem perangkat lunak yang sebenarnya dikembangkan juga sehingga sesuai dengan kebutuhan *user*.

3.2.3.3 Alat Bantu Analisis dan Perancangan

Pada tahap ini sistem akan dirancang dengan memanfaatkan alat bantu seperti berikut :

1) ***Use Case Diagram***

   Diagram *use case* merupakan pemodelan untuk kegiatan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang

akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi – fungsi itu.

Ada dua hal utama dalam *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

a) Aktor merupakan orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang tapi aktor belum tentu merupakan orang.
b) *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit – unit yang saling bertukar pesan antar unit atau aktor.

2) *Class Diagram*

Diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

a) Atribut merupakan variabel – variabel yang dimiliki suatu kelas.
b) Operasi atau metode adalah fungsi – fungsi yang dimiliki oleh suatu kelas.

Kelas – kelas yang ada pada struktur sistem harus dapat melakukan fungsi – fungsi sesuai dengan kebutuhan sistem. Susunan struktur

kelas yang baik pada diagram kelas sebaiknya memiliki jenis – jenis kelas berikut :

a) Kelas main

   Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

b) Kelas yang menangani tampilan sistem

   Kelas yang mendefinisikan dan mengatur tampilan ke pengguna.

c) Kelas yang diambil dari pendefinisian *use case*

   Kelas yang menangani fungsi – fungsi yang harus ada diambil dari pendefinisian *use case*.

d) Kelas yang diambil dari pendefinisian data

   Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Dalam mendefinisikan metode yang ada di dalam kelasperlu memperhatikan apa yang disebut dengan *cohesion* dan *coupling*. *Cohesion* adalah seberapa dekat keterkaitan intruksi di dalam sebuah metode terkait satu sama lain sedangkan *coupling* adalah ukuran seberapa dekat keterkaitan intruksi antara metode yang satu dengan metode yang lain dalam sebuah kelas. Sebagai aturan secara umum

maka sebuah metode yang dibuat harus memiliki kadar *cohension* yang kuat dan kadar *coupling* yang lemah.

3) *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek – objek yang terlibat dalam sebuah *use case* beserta metode – metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Banyaknya diagram sekuen yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

4) *Activity Diagram*

Diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal – hal berikut :

a) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
b) Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
c) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.

5) *Component Diagram*

Diagram komponen dibuat untuk menunjukkan organisasi dan ketergantungan di antara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem. Diagram komponen juga dapat digunakan unutk memodelkan hal – hal berikut :

a) *Source code* program perangkat lunak
b) Komponen *executable* yang dilepas ke *user*
c) Basis data secara fisik
d) Sistem yang harus beradaptasi dengan sistem lain

e) *Framework* sistem, *framework* pada perangkat lunak merupakan kerangka kerja yang dibuat untuk memudahkan pengembangan dan pemeliharaan aplikasi.

Komponen dasar yang biasanya ada dalam suatu sistem adalah sebagai berikut :

a) Komponen *user interface* yang menangani tampilan
b) Komponen *bussiness procesiing* yang menangani fungsi – fungsi proses bisnis
c) Komponen *data* yang menangani manipulasi data
d) Komponen *security* yang menangani keamanan sistem.

Komponen lebih terfokus pada penggolongan secara umum fungsi – fungsi yang diperlukan.

6) *Deployment Diagram*

Diagram *deployment* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal – hal berikut :

a) Sistem tambahan (*embedded system*) yang menggambarkan *device, node* dan *hardware*.
b) Sistem *client / server*.
c) Sistem terdistribusi murni.
d) Rekayasa ulang aplikasi.

3.2.4 Pengujian *Software*

Pada penelitian ini, *software* diuji dengan menggunakan *black box testing* atau pengujian *black box*. Pengujian *black box* merupakan pendekatan komplementer dari teknik *white box*, karena pengujian *black box* diharapkan mampu mengungkap kelas kesalahan yang lebih luas dibandingka teknik *white box*. Pengujian *black box* berfokus pada pengujian persyaratan fungsional perangkat lunak, untuk mendapatkan serangkaian kondisi input yang sesuai dengan persyaratan fungsional suatu program.

Pengujian *black box* adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian *black box* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dibangkitkan, dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak dicek apakah telah sesuai dengan yang diharapkan.

Pengujian *black box* berusaha menemukan kesalahan dalam kategori :

1. Fungsi – fungsi yang tidak benar atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau akses *database eksternal*
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi

Berbeda dengan pengujian *white box*, pengujian *black box* cenderung diaplikasikan selama tahap akhir pengujian. Pengujian *black box* harus dapat menjawab pertanyaan sebagai berikut :

1. Bagaimana validitas fungsional diuji
2. Kelas input apa yang akan membuat kasus pengujian menjadi lebih baik
3. Apakah sistem akan sangat sensitif terhadap input tertentu
4. Bagaimana batasan dari suatu data diisolasi
5. Kecepatan data apa dan *volume* data apa yang akan ditoleransi oleh sistem
6. Apa pengaruh kombinasi tertentu dari data terahadap sistem operasi