BAB III

OBJEK DAN METODE PENELITIAN

3.1. Objek Penelitian

Tempat yang dijadikan penulis sebagai objek penelitian yakni sebuah bidang jasa pelayanan penyewaan perlengkapan bayi. Untuk melihat lebih jelas gambaran mengenai objek penelitian tepatnya di Little Joy, maka penulis membahas mengenai sejarah, visi dan misi, struktur organisasi dan deskripsi pekerjaan dari Little Joy tersebut.

3.1.1. Sejarah Singkat Perusahaan

Little Joy merupakan perusahaan yang bergerak di bidang Jasa Persewaan perlengkapan bayi, yang berdiri pada tanggal 19 Juli 2010 bertempat di Jl. Soekarno Hatta No. 366 Bandung 40235. Pendirinya adalah Ibu Lily Virgiyanti.



Gambar 3.1 Logo Perusahaan Little Joy

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

3.1.2. Visi dan Misi Perusahaan

Visi Perusahaan

" Menjadi Penyedia Jasa Penyewaan Perlengkapan Bayi yang profesional, kreatif, dan terbaik se-Bandung Raya".

Misi Perusahaan

1. Meningkatkan produktivitas dan profesionalitas karyawan ditunjang dengan fasilitas perusahaan yang memadai.
2. Menyediakan Sarana dan Prasarana Perlengkapan bayi yang lengkap.
3. Memberikan pelayanan terbaik kepada konsumen.

3.1.3. Struktur Organisasi Perusahaan

Struktur organisasi Little Joy dapat dilihat pada gambar di bawah ini.



Gambar 3.2 Struktur Organisasi Little Joy

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

3.1.4. Deskripsi Tugas

Agar setiap karyawan dapat menjalankan fungsi serta tugasnya masing-masing dengan baik maka perlu dijabarkan secara jelas tentang deskripsi kerjanya masing-masing yang sesuai dengan posisi jabatan yang diambil di dalam perusahaan tersebut. Adapun deskripsi kerja masing-masing bagian yang ada di dalam Little Joy adalah sebagai berikut :

1. Pemilik/Bagian Administrasi
   a. Mempunyai tugas mengawasi , menghitung inventori dan menetapkan harga jual.
   b. Mengambil tindakan atau keputusan yang dianggap perlu untuk menghadapi atau menangani permasalahan yang timbul di Little Joy yang dapat mempengaruhi kinerja perusahaan.
2. Penjaga toko

   Mempunyai tugas melayani penyewa barang-barang perlengkapan bayi yang ada ditoko.
3. Tukang antar jemput barang

   Mempunyai tugas mengantarkan atau menjemput barang pesanan kepada penyewa perlengkapan bayi.

3.2. Metode Penelitian

Metode suatu kerja untuk melakukan suatu tindakan, atau suatu kerangka berfikir menyusun gagasan, yang beraturan, berarah dan berkonteks dengan maksud dan tujuan.

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

Penulis menggunakan metode pendekatan analisis dan pemograman notasi yang berorientasi objek, sebagai metode untuk mencari pemecahan permasalahan di Little Joy. Sehingga dapat mendapatkan solusi dan pemecahan masalahnya dengan di dasari dari data-data yang telah ada.

3.2.1. Desain Penelitian

Desain penelitian yang digunakan dalam membuat sistem informasi penyewaan perlengkapan bayi di Little Joy yaitu metode deskriptif. Metode deskriptif yaitu penelitian yang bertujuan untuk memperoleh ciri-ciri variable, dimana dalam penelitian ini adalah untuk memperoleh gambaran tentang kinerja program yang dirancang dan di implementasikan kepada pengguna (user) dengan pendekatan studi kasus pada Little Joy.

3.2.2. Jenis dan Metode Pengumpulan Data

Dalam pengembangan sistem tentunya diperlukan data yang akurat sesuai dengan sistem yang akan dikembangkan, oleh karena itu dibutuhkan suatu teknik pengumpulan data yang tepat. Teknik pengumpulan data yang digunakan penulis dalam tahap analisa dan tahap perancangan sebagai berikut :

3.2.2.1. Sumber Data Primer

Data primer dalam penelitian di dapatkan dari hasil wawancara dengan pemilik Little Joy. Data yang didapatkan dari hasil wawancara adalah sejarah perusahaan, prosedur penyewaan, pemesanan, pembatalan pemesanan dan penitipan jual.

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

3.2.2.2. Sumber Data Sekunder (dokumentasi)

Teknik pengumpulan data sekunder didapatkan dengan cara mengumpulkan dokumen-dokumen yang dimiliki perusahaan yang dapat digunakan dalam tahap analisis sistem. Adapun dokumen-dokumennya yaitu:

1. Formulir Penyewaan Barang
2. Formulir Penitipan Jual Barang
3. Formulir Pemesanan/Booking
4. Bukti Pembayaran (Deposit, Sewa, Biaya Administrasi Penitipan Jual, Biaya Pembatalan Pemesanan)
5. Laporan Data Barang

3.2.3. Metode Pendekatan dan Pengembangan Sistem

Metode pendekatan dan pengembangan sistem merupakan suatu metode yang akan digunakan dalam melakukan perancangan sistem informasi.

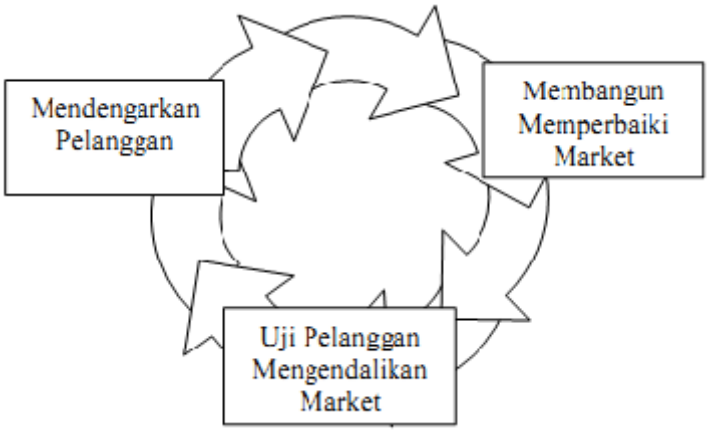
3.2.3.1. Metode Pendekatan Sistem

Metode pendekatan yang digunakan adalah metode pendekatan analisis dan pemograman notasi yang berorientasi objek, dimana akan membahas permasalahan di organisasi tersebut agar dapat dipecahkan dan hasil dari sistem akan mudah untuk dipelihara, lebih memuaskan pemakainya, mempunyai dokumentasi yang baik, tepat waktu, dan dapat meningkatkan produktivitas dan kualitasnya akan lebih baik.

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

3.2.3.2. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan adalah dengan menggunakan metode prototipe. Prototipe paradigma dimulai dengan mengumpulkan kebutuhan. Pengembangan dan pelanggan bertemu dan mendefinisikan keseluruhan sistem yang akan dibuat, mengidentifikasi segala kebutuhan yang diketahui, kemudian melakukan perancangan.



Gambar 3.3 Prototipe Paradigma

Sumber (Rekayasa Perangkat Lunak, Roger S. Pressman, Ph.D).

2002 : 40)

Menurut Roger S. Pressman, Ph.D. (2002 : 41) Prototipe bisa berfungsi sebagai sistem yang pertama, memang benar bahwa baik pelanggan maupun pengembang menyukai paradigma prototipe. Para pemakai merasa enak dengan sistem aktual, sedangkan pengembang sistem membangunnya dengan segera.

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

3.2.3.3. Alat Bantu Analisis dan Perancangan

Pembangunan sistem informasi dengan berorientasi objek merupakan suatu perancangan yang berbeda dengan yang berorientasi data. Namun secara konteks perancangan ini digunakan untuk membangun sistem informasi sesuai kebutuhan dari pemakai (*user*) sistem informasi.

Menurut Martin Fowler (2004:1) "Unified Modeling Language (UML) adalah keluarga notasi grafis yng didukung oleh meta-model tunggal, yang membantu pedeskripsian dan desain system perangkat lunak, khususny system yang dibangun mengguanakan pemograman berorientasi objek (OO)".

UML sendiri telah menyediakan alat-alat bantu berupa diagram visual yang menggambarkan berbagai aspek yang ada di dalam sistem. Adapun alat-alat bantu diagram yang disediakan di dalam UML diantaranya, yaitu :

1. *Use Case* Diagram

   Menurut Martin Fowler (2004:141) "Use case adalah teknik untuk merekam persyaratan fungsional sebuah system yang berinteraksi antara para pengguna system dengan system itu sendiri, dengan member sebuah narasi tentang bagaimn system tersebut digunakan".

   Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah "apa" yang diperbuat sistem, dan bukan "bagaimana". Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan system. *Use case* merupakan sebuah pekerjaan tertentu, yaitu seorang aktor adalah sebuah entitas manusia atau mesin yang

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

2. *Activity* Diagram

Menurut Martin Fowler (2004:163) "Activity diagram adalah teknik untuk menggambarkan llogika procedural, proses bisnis dan jalur kerja".

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan *state diagram* khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih

3. *Class* Diagram

Class diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class*

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Hubungan Antar *Class* adalah sebagai berikut :

1. Asosiasi, yaitu hubungan statis antar *class*. Umumnya menggambarkan *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain. Panah *navigability* menunjukkan arah *query* antar *class*.
2. Agregasi, yaitu hubungan yang menyatakan bagian ("terdiri atas..").
3. Pewarisan, yaitu hubungan hirarkis antar *class*. *Class* dapat diturunkan dari *class* lain dan mewarisi semua atribut dan metoda *class* asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari *class* yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.

Hubungan dinamis, yaitu rangkaian pesan (*message*) yang di-*passing* dari satu *class* kepada *class* lain. Hubungan dinamis dapat digambarkan dengan menggunakan *sequence diagram* yang akan dijelaskan kemudian.

4. *Sequence* Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu.

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

Sequence diagram terdiri atar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message.

5. *Collaboration* Diagram

   Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, di mana *message* dari level tertinggi memiliki nomor 1. Messages dari level yang sama memiliki prefiks yang sama.

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

6. *Component* Diagram

   Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya. Komponen piranti lunak adalah modul berisi *code*, baik berisi *source code* maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time, link time*, maupun *run time*. Umumnya komponen terbentuk dari beberapa *class* dan/atau *package*, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface*, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain.

7. *Deployment* Diagram

   Deployment/physical diagram menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisikal. Sebuah *node* adalah server, *workstation*, atau piranti keras lain yang digunakan untuk men-*deploy* komponen dalam lingkungan sebenarnya. Hubungan antar *node* (misalnya TCP/IP) dan *requirement* dapat juga didefinisikan dalam diagram ini.

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional

3.2.4. Pengujian Software

Dalam tahap pengujian ini penulis menggunakan pengujian black box. Alasannya karena pengujian black box dapat mengetahui apakah perangkat lunak yang dibuat dapat berfungsi dengan benar dan telah sesuai dengan yang diharapkan. Pengujian black box adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika intern perangkat lunak. 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 berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian black-box memungkinkan perekayasa perangkat lunak ,mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian black-box berusaha menemukan kesalahan dalam kategori sebagai berikut :

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.

Created with nitroPDF professional
download the free trial online at nitropdf.com/professional