BAB III

OBJEK DAN METODE PENELITIAN

3.1. Objek Penelitian

Objek penelitian dilakukan di PD. PERTUB dengan tujuan untuk mempermudah dalam perancangan suatu Sitem Informasi Pembelian dan Penjualan

3.1.1 Sejarah Singkat Perusahaan

Pada tahun 1980 pemilik PD. PERTUB yaitu Jai Rusdiana bekerja pada suatu perusahaan produksi didaerah Bandung tepatnya didaerah kiara condong. Pada perusahaan tersebut pemilik bekerja sebagai karyawan pembuatan Paper Tube atau Rollan kain dan selama kurang lebih 5 tahun pemilik bekerja di perusahaan tersebut. Pada tahun 1985 pemilik keluar dari pekerjaannya dan berpindah pada perusahaan dagang Budi Agung yang masih terikat keluarga dan masih bergerak pada bidang yang sama yaitu pembuatan *Paper Tube* dan kurang lebih 6 tahun pemilik bekerja disana.

Faktor kebutuhan hidup dan dengan niat untuk memperbaiki kehidupannya pemilik berinisiatif untuk mendirikan usaha sendiri dalam pembuatan *Paper Tube.*bermodalkan pengalaman bertahun-tahun pemilik mulai merintis usahanya dengan mempersiapkan alat dan bahan sebagai bahan pembuatan *Paper Tube*. Terhitung sejak tahun 1990 sampai sekarang perusahaan yang di rintis pemilik yaitu Jai Rusdiana masih berjalan yang berlamat di Jl Caringin Blk 282B Dengan

nama Perusahaan dagang PERTUB. Nama pertub diambil dari singkatan *Paper Tube.*

3.1.2 Visi dan Misi Perusahaan

Visi dan Misi adalah suatu konsep perencanaan yang di sertai dengan tindakan sesuai dengan apa yang di rencanakan untuk mencapai suatu tujuan. Berikut adalah visi dan misi dari PD PERTUB adalah sebagai berikut :

3.1.2.1 Visi

Menjadi perusahaan yang unggul dan terpercaya

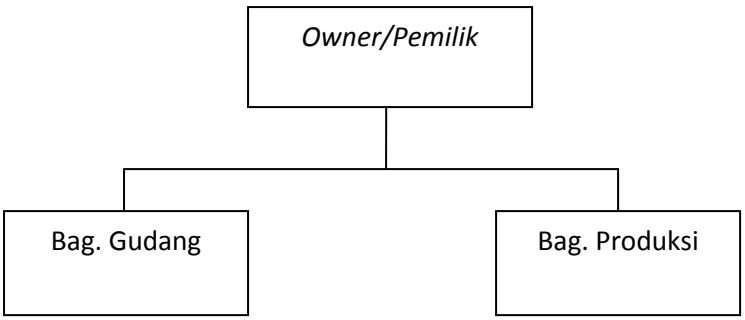
3.1.2.2 Misi

Menjadi suatu perusahaan dagang yang berkembang secara terus menerus dalam mengembangkan Sumber Daya Manusia, dan memberikan kualitas produksi yang berkualitas tinggi pada pelanggan.

3.1.3 Struktur Organisasi Perusahaan

Struktur organisasi adalah suatu susunan dan hubungan antara tiap bagian serta posisi yang ada pada suatu organisasi atau perusahaan dalam menjalankan kegiatan operasional untuk mencapai tujuan. Struktur organisasi menggambarkan dengan jelas pemisahan kegiatan pekerjaan antara yang satu dengan yang lain dan bagaimana hubungan akitvitas dan fungsi dibatasi. Dalam struktur organisasi yang baik harus menjelaskan hubungan wewenang siapa melapor kepada siapa.

Berikut adalah struktur organisasi pada PD. PERTUB



Gambar 3.1 Struktur organisasi pada PD. PERTUB (PD. PERTUB, 2013)

3.1.4 Deskripsi Tugas

1. Pemilik

    a) Memiliki wewenang dan pengambilan keputusan sepenuhnya atas kegiatan di bengkel.

    b) Bertanggung jawab penuh terhadap semua kegiatan perusahaan.

    c) Mengontrol seluruh kinerja karyawan

    d) Memberikan arahan terhadap pegawai untuk kemajuan usaha.

2. Bagian Gudang

    a) mengadakan penyediaan barang

    b) Mencatat semua penerimaan dan pengeluaran bahan baku

3. Bagian Produksi

    Bagian produksi ini bertanggung jawab untuk pembutan paper tube atas pesanan dari pelanggan

3.2. Metode Penelitian

Metode penelitian merupakan berisi uraian yang menggambarkan tahapan-tahapan dalam proses penelitian guna memecahkan masalah, penelitian dari awal perencanaan hingga tercapainya tujuan penelitian.

3.2.1 Desain Penelitian

Desain penelitian adalah semua proses yang diperlukan dalam perencanaan dan pelaksanaan penelitian, yang membantu dalam pengumpulan dan menganalisis data.

3.2.2 Jenis dan Metode Pengumpulan Data

Metodologi Penelitian yang digunakan dalam penulisan skripsi ini adalah sebagai berikut :

3.2.2.1 Sumber Data Primer

Data primer merupakan data yang didapat dari sumber informan pertama yaitu individu atau perseorangan. Dalam memperoleh data primer penulis melakukan observasi dan wawancara di tempat penelitian

1. Observasi

   Pengumpulan data data dengan cara terjun secara langsung terhadap objek yang akan diteliti. hasil observasi ke lapangan secara langsung dalam bentuk catatan tentang situasi dan kejadian

2. Wawancara

   Wawancara merupakan teknik pengumpulan data dengan cara dialog kepada narasumber. Pada tahap ini penulis melakukan wawancara dengan pemilik yaitu Jai Rusdiana. Tujuan dari wawacara ini adalah untuk mengetahui data-data yang dibutuhkan penulis seputar perusahaan tersebut.

3.2.2.2 Sumber Data Sekunder

Data sekunder antara lain disajikan dalam bentuk data-data, tabel-tabel, diagram-diagram, atau mengenai topik penelitian. Data ini digunakan untuk mendukung infomasi primer yang diperoleh baik dari dokumen, maupun dari observasi langsung ke lapangan. Adapun data sekunder yang digunakan yaitu sumber tertulis yang antara lain terdiri dari:

1. Sumber buku, Jurnal Ilmiah dan Internet

   Sumber tersebut merupakan sumber yang sangat berguna untuk menunjang penelitian, yakni sebagai tinjauan pustaka untuk menganalisis penelitian yang dilakukan.

2. Dokumen

   Mengumpulkan data sekunder dari PD. PERTUB melalui dokumen-dokumen serta hal-hal yang berkaitan dengan proses penelitian.

3.2.3 Metode Pendekatan dan Pengembangan Sistem

Berikut merupakan metode pendekatan dan pengembangan sistem yang dipakai dalam merancang dan membangun suatu aplikasi.

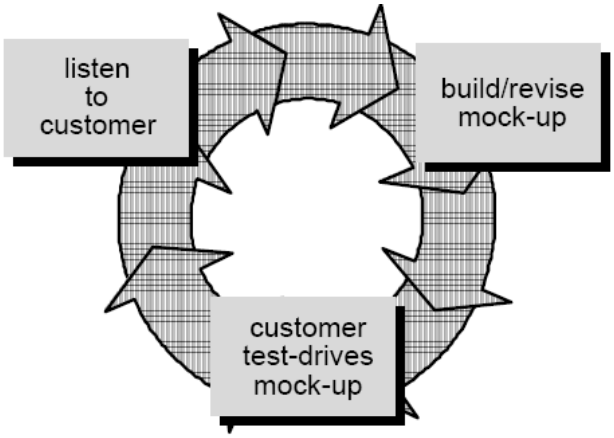
3.2.3.1 Metode Pendekatan Sistem

Dalam melakukan penelitian ini penulis menggunakan metode pendekatan yang dipakai adalah pemodelan berorientasi objek. Berorientasi objek adalah cara memandang persoalan menggunakan model-model yang diorganisasikan seputar konsep objek yang mengkombinasikan struktur data dan perilaku suatu entitas.

3.2.3.2 Metode Pengembangan Sistem

Metode pengembangan pada tahap ini yaitu menggunakan pendekatan prototipe *(prototyping).* Metode ini sangat baik digunakan untuk menyelesesaikan masalah kesalahpahaman antara *user* dan analis yang timbul akibat *user* tidak mampu mendefinisikan secara jelas kebutuhannya (Mulyanto, 2009).

Sebagian *user* kesulitan mengungkapkan keinginannya untuk mendapatkan aplikasi yang sesuai dengan kebutuhannya. Kesulitan ini yang perlu diselesaikan oleh analis dengan memahami kebutuhan *user* dan menerjemahkannya ke dalam bentuk model (prototipe). Model ini selanjutnya diperbaiki secara terus menerus sampai sesuai dengan kebutuhan *user.* Berikut adalah gambar prototype model.



Gambar 3.2. Prototyping Model (Yasmi Afrizal & Wahyuni, 2012)

Proses pada model *prototyping* yang digambarkan pada gambar 3.2 bisa dijelaskan sebagai berikut :

1. Pengumpulan kebutuhan : *developer* dan klien bertemu dan menetukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya.
2. Perancangan : perencanaan dilakukan cepat dan rancangan mewakili semua aspek *softwar*e yang diketahui, dan rancangan ini menjadi dasar pembuatan *prototype*.
3. Evaluasi *prototype* : klien mengevaluasi *prototype* yang dibuat dan digunakan untuk memperjelas kebuthan software.

Perulangan ketiga proses ini terus berlangsung hingga semua kebutuhan terpenuhi. Prototype-prototype ini dibuat untuk memuaskan kebutuhan klien dan untuk memahami kebutuhan klien lebih baik.

3.2.3.3 Alat Bantu Analisis dan Perancangan

Alat bantu dalam menganalisis dan perancangan pada tahap ini menggunakan pemodelan yang disebut *Unifield Modeling Language* (*UML).*

UML menurut Yasmi Afrizal, Wahyuni memaparkan bahwa UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan atau di hasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi, atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya.

Alat bantu yang digunakan dalam perancangan sistem adalah sebagai berikut:

1. *Usecase Diagram*

Sebuah bagian penting dari *Unified Modelling Language* (UML) adalah fasilitas yang memungkinkan kita untuk menggambar sebuah diagram, yaitu diagram use-case. *Use case* digunakan selama tahap analisis dalam sebuah proyek untuk mengidentifikasikan dan membagi fungsi dari sistem. Diagram ini membagi-bagi sistem menjadi *actor* dan *use cases*. Aktor mewakili peran-peran yang dapat dikerjakan oleh user di dalam sistem. *User-user* ini dapat berupa manusia, bagian *hardware* komputer atau bahkan *software* sistem yang lain. Salah satu kriteria adalah mereka harus berada di luar bagian dari *use case* sebuah sistem. Mereka harus menyediakan masukan (inputan) ke dalam sistem dan mereka akan mendapatkan keluaran (*output*) dari sistem tersebut.

Use case menjelaskan tentang cara kerja dari sistem setelah mendapat inputan dari actor. Cara kerja ini dijelaskan secara rinci. *Use case* menjelaskan tentang hal-hal seperti, masukan dari seorang actor dan bagaimana keluaran yang diterima oleh *actor* lain, dan cara kerja yang dapat merubah suatu masukan menjadi keluaran.

Use case merupakan suatu diagram yang sangat berguna bagi pihak analis saat mengelompokkan fungsi-fungsi dari sistem. Hubungan-hubungan antara cases dan diagram-diagram yang berhubungan dapat membantu pihak analis untuk membuat struktur dari sebuah sistem. Tetapi *use cases* bukanlah sebuah alat desain, mereka tidak digunakan untuk memspesifikasikan sebuah software atau mereka tidak melibatkan sebuah keberadaan kelas-kelas dan obyek-obyek. *Use case* merupakan sebuah deskripsi tentang sebuah sistem yang secara murni terpisah dari tahap desain sebuah *software*.

2. *Activity Diagram*

Activity diagrams 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, di mana 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. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

3. *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. *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/ metode dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message

4. *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 diagram,* dimana message dari level tertinggi memiliki nomor 1. Message dari level yang sama memiliki prefix yang sama.

5. *Class Diagram*

Class diagram digunakan untuk menampilkan kelas-kelas dan paket-paket di dalam sistem. *Class diagram* memberikan gambaran sistem secara statis dan relasi antar mereka. Biasanya, dibua beberapa *class diagram* untuk sistem tunggal. Beberapa *diagram* akan menampilkan subset dari kelas-kelas dan relasinya. Dapat dibuat beberapa diagram sesuai dengan yang diinginkan untuk mendapatkan gambaran lengkap terhadap sistem yang dibangun.

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 Diagram adalah *diagram* yang menggambarkan detail bagaimana komponen disebar kedalam *infrastruktur* sistem, dimana komponen akan terletak (pada mesin, *node*, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, *spesifikasi server*, dan hal-hal lain yang bersifat fisikal.

3.2.4 Pengujian Software

Pada tahap ini metode pengujian software yang digunakan adalah metode *Black Box Testing*. Konsep black box digunakan untuk mempresentasikan sistem yang cara kerja di dalamnya tidak tersedia untuk diinpeksi. Di dalam *Black Box*, item-item yang diuji dianggap "gelap" karena logikanya tidak diketahui, yang diketahui hanya apa yang masuk dan apa yang keluar dari *Black Box*

Pengujian black box merupakan pendekatan komplementer dari teknik white box.

Pengujian black box berusaha menemukan kesalahan dalam kategori :

a. Fungsi-fungsi yang tidak benar atau hilang,

b. Kesalahan interface,

c. Kesalahan dalam struktur data atau akses database eksternal,

d. Kesalahan kinerja,

e. Inisialisasi dan kesalahan terminasi..