BAB III

OBJEK DAN METODE PENELITIAN

3.1. Objek Penelitian

Objek penelitian ini berisikan tentang gambaran tempat penulis melakukan penelitian, diantaranya tentang sejarah, struktur organisasi dan deskripsi pekerjaan pada Perusahaan Home Industri Zadini Shoes yang terletak di Jalan Terusan Cibaduyut (Situtarate) No. 27 Bandung 40239.

3.1.1. Sejarah Singkat Zadini Shoes

Zadini shoes didirikan oleh Bapak Jajat Sudrajat sejak tahun 1990, bergerak dalam bidang usaha pengelolaan, produksi dan penjualan sepatu yang termasuk dalam *home industry* karena perusahaan industri termasuk dalam skala menengah. berlokasi dijalan terusan Cibaduyut (Situtarare) No. 27 Bandung. Zadini Shoes memproduksi dan menjual sepatu dengan berbagai macam bentuk atau model dengan bahan yang berasal dari kulit ataupun *non* kulit. Perusahaan Home industri Zadini Shoes juga melayani *order* (pesanan) dari perusahaan lain ataupun menjual langsung kepada konsumen.

3.1.2. Visi dan Misi Zadini Shoes

3.1.2.1. Visi Zadini Shoes

Menjadi salah satu perusahaan sepatu casual terbaik di Bandung dan dapat memasarkan dalam skala nasional dan internasional.

3.1.2.2. Misi Zadini Shoes

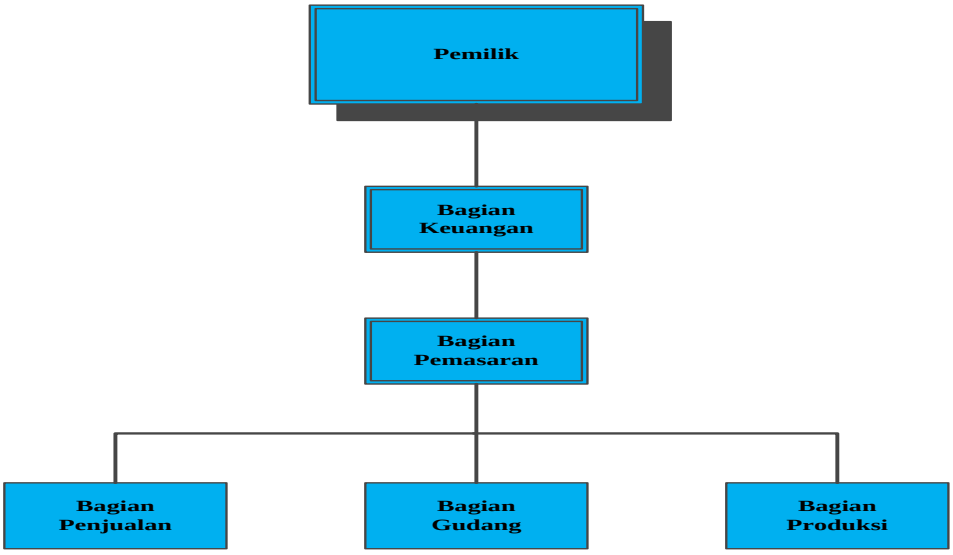
1. Membuat sepatu dengan kualitas terbaik
2. Berbisnis dengan jujur dan mentaati hukum yang berlaku
3. Menyiapkan dan membina SDM yang ahli dan berkemampuan di bidangnya.

3.1.3 Struktur Organisasi

Struktur organisasi mempunyai arti penting karena struktur organisasi merupakan bentuk atau pola formal kegiatan dan hubungan antara berbagai bagian-bagian didalam suatu perusahaan. Dengan mengetahui struktur organisasi dapat diperoleh gambaran tentang bagianbagian yang ada di dalamnya, apa peranan masing-masing bagian tersebut dan wewenang serta tanggung jawabnya dalam melaksanakan tugasnya.

STRUKTUR ORGANISASI

Zadini Shoes



Gambar 3.1 Struktur Organisasi

(Sumber : Data Zadini Shoes *Bandung)*

3.1.4 Deskripsi Kerja

Deskripsi kerja adalah suatu rincian yang menunjukan posisi, tanggung jawab, wewenang, fungsi dan tugas tugas yang harus dilakukan. Deskripsi tugas perlu dibuat agar masing – masing bagian mengerti akan kedudukan didalam organisasi. Adapun uraian jabatan pada Perusahaan Zadini Shoes Bandung, yaitu :

1) Pemilik
    a. Menerima laporan secara berkala dari Penanggung jawab keuangan yang berkaitan dengan perkembangan Perusahaan.
    b. Memiliki wewenang sebagai pengambil keputusan setiap kegiatan perusahaan.
2) Bagian Keuangan
    a. Bertanggung jawab atas semua kegiatan keuangan
    b. Bertanggung jawab atas nominal uang yang dipegang
3) Bagian Pemasaran
    a. Bertanggung jawab atas kegiatan pemasaran yang dilakukan oleh Perusahaan.
    b. Mengkoordinir semua aktifitas perusahaan dalam memberikan pelayanan kepada semua pelanggan yang diarahkan untuk pemenuhan kepuasan pelanggan dan meningkatkan jumlah pelanggan.
4) Bagian Penjualan
    a. Membuat laporan penjualan secara berkala untuk pimpinan.
    b. Mengolah transaksi penjualan.
5) Bagian Gudang
    a. Melakukan pengecekan barang yang ada

b. Membuat laporan persediaan barang

6) Bagian Produksi

a. Mengkoordinasi kegiatan produksi sehingga proses produksi dapat berjalan dengan lancar.

b. Mengambil segala keputusan yang berhubungan dengan apa yang akan diproduksi atas persetujuan *owner*.

3.2 Metode Penelitian

Metode penelitian adalah sebuah metode atau cara yang akan dilakukan pada sebuah penelitian baik dalam hasil akhir akan menghasilkan sebuah produk, riset maupun produk dan riset, serta melibatkan individu sebagai objek penelitian

Metode penelitian yang dilakukan dalam pengumpulan data diantaranya :

a. Observasi (Pengamatan)

Yaitu suatu pengamatan yang sistematis terhadap objek yang dituju secara langsung yang dilakukan dengan indera mata.

b. Interview (Wawancara)

Pada metode ini peneliti mengadakan interview atau wawancara langsung dengan pihak yang berwenang dalam hal ini adalah pimpinan Perusahaan Zadini Shoes.

3.2.1. Desain Penelitian

Dalam melakukan suatu penelitian diperlukan perencanaan penelitian agar penelitian yang dilakukan dapat berjalan dengan baik, sistematis serta efektif. Desain penelitian adalah semua proses yang diperlukan dalam perencanaan dan pelaksanaan penelitian.

Dari uraian diatas maka dapat dikatakan bahwa, desain penelitian merupakan semua proses penelitian yang dilakukan oleh penulis dalam melaksanakan penelitian mulai dari perencanaan sampai dengan pelaksanaan penelitian yang dilakukan pada waktu tertentu.

Dalam penelitian ini, penulis menetapkan desain penelitian yang lebih luas, yang mencakup proses-proses berikut ini :

1. Identifikasi masalah
2. Merumuskan masalah
3. Memilih metode pengumpulan data
4. Pelaporan hasil penelitian termasuk proses penelitian dan interpretasikan data.

3.2.2. Jenis dan Metode Pengumpulan Data

Metode pengumpulan data meliputi penelitian lapangan dapat berupa data primer dan sekunder adalah sebagai berikut :

3.2.2.1. Sumber Data Primer

a. Observasi adalah metode pengumpulan data melalui pengamatan langsung atau peninjauan secara cermat dan langsung di lapangan atau lokasi penelitian. Dalam hal ini, peneliti dengan berpedoman kepada desain penelitiannya perlu mengunjungi lokasi penelitian untuk mengamati langsung berbagai hal atau kondisi yang ada di lapangan. Penemuan ilmu pengetahuan selalu dimulai dengan observasi dan kembali kepada observasi untuk membuktikan kebenaran ilmu pengetahuan tersebut.

b. Wawancara adalah Dialog langsung dengan orang yang terkait dengan objek yang bersangkutan yang sekiranya dapat membantu dalam melengkapi data-data yang diperlukan.

3.2.2.2. Sumber Data Sekunder

Pengumpulan data dengan cara mempelajari data yang telah tersedia atau dikumpulkan terlebih dahulu oleh pihak lain seperti buku-buku.

Cara yang digunakan 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. Metode ini digunakan untuk pengumpulan data yang berhubungan dengan sejarah, tujuan, kegiatan dan struktur organisasi,

3.2.3 Metode Pendekatan dan Pengembangan Sistem

Metode pendekatan dan pengembangan sistem yang akan digunakan oleh penulis bertujuan untuk memudahkan penulis untuk menyelesaikan masalah yang terdapat pada penelitian ini, penjelasan lebih lengkap adalah sebagai berikut :

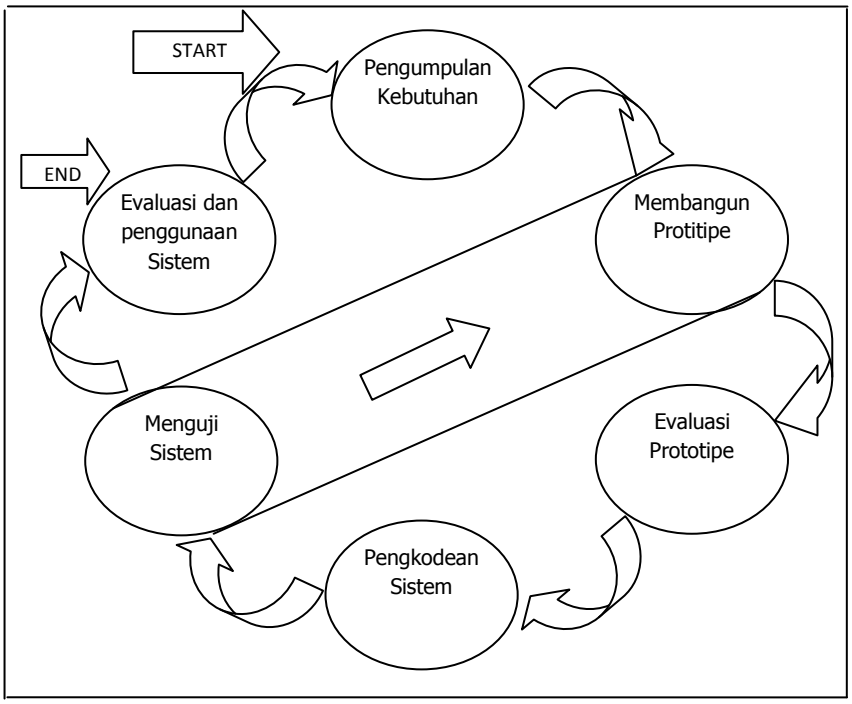
3.2.3.1 Metode Pendekatan Sistem

Metode Pendekatan Sistem Metode pendekatan sistem yang di gunakan adalah pendekatan dengan Object Oriented yang menggnakan AOO (Analisis Object Oriented) dan DOO yang di visualisasikan dengan UML dan diantara nya adalah sebagai berikut : UseCase diagram, Activity

diagram, Class Diagram, Sequence diagram, Class Diagram, Component diagram ,dan Deployment diagram

3.2.3.2 Metode Pengembangan Sistem

Metode yang digunakan dalam membangun perangkat lunak dalam penyusunan skripsi ini adalah **Prototipe** karena tidak melakukan identifikasi kebutuhan output, pemrosesan ataupun input detail, tidak memiliki kepastian efisiensi algoritma, tidak memiliki kemampuan penyesuaian dari sebuah sistem operasi, atau bentuk - bentuk yang harus dilakukan oleh interaksi manusia dan mesin.



Gambar 3.2 prototipe Model

[Sumber:"Software Engineering Laboratory" Modul Praktikum Rekayasa Perangkat Lunak. 2004]

Tahapan-tahapan Prototyping

Tahapan-tahapan dalam Prototyping adalah sebagai berikut:

1) Pengumpulan kebutuhan

Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasikan semua kebutuhan, dan garis besar sistem yang akan dibuat.

2) Membangun prototyping

   Membangun prototyping dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat input dan format output)

3) Evaluasi protoptyping

   Evaluasi ini dilakukan oleh pelanggan apakah prototyping yang sudah dibangun sudah sesuai dengan keinginann pelanggan. Jika sudah sesuai maka langkah 4 akan diambil. Jika tidak prototyping direvisi dengan mengulangu langkah 1, 2 , dan 3.

4) Pengkodean sistem

   Dalam tahap ini prototyping yang sudah di sepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai

5) Menguji sistem

   Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan.

6) Evaluasi Sistem

   Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan . Perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan .

Secara ideal prototipe berfungsi sebagai sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. Bila prototipe yang sedang

bekerja dibangun pengembang harus mempergunakan fragmen-fragmen program yang ada yang memungkinkan program yang bekerja untuk dimunculkan secara cepat.

3.2.3.3 Alat Bantu Analisis dan Perancangan

1) Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang di harapkan dari sebuah sistem. Sebuah usecase mempresentasikan sebuah interaksi antara aktor dengan sistem. Usecase merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng create sebuah daftar belanja, dan sebagainya. Seorang atau sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan- pekerjaan tertentu.

Use case diagram dapat sangat membantu bila kita sedang menyusun requirment sebuah sistem, mengkomunikasikan rancangan dengan klien, merancang test case untuk semua feature yang ada pada sistem. Sebuah usecase dapat meng include fungsionalitas usecase lain sebagai bagian dari proses dalam dirinya. Secara umum di asumsikan bahwa use case yang di include oleh lebih dari satu usecase lain,sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang common. Sebuah use case juga dapat meng extend use case lain dengan behaviournya sendiri. Sementara hubungan generalisasi antar use case menunjukan bahwa use case yang merupakan spesialisasi dari yang lain.

2) Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan disain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk manipulasi keadaan tersebut (metoda/fungsi). Class diagram menggambarkan struktur dan deskripsi class, Package dan objek serta hubungan satu sama lain seperti contaiment pewarisan asosiasi dan lain – lain.

Class memiliki 3 area pokok :

1. Nama (stereotype)
2. Atribut
3. Metoda

Atribut dan metoda dapat memiliki sifat sebagai berikut :

a. Private, tidak dapat dipanggil dari luar class yang bersangkutan.
b. Protected,hanya dapat dipanggil oleh class yang bersangkutan dengan anak- anak yang mewarisi nya.
c. Public, dapat di panggil oleh siapa saja.

Class dapat merupakan implementasi dari sebuah interface, yaitu class abstrack yang hanya memiliki metoda. Inteface tidak dapat langsung di instansikan, tetapi harus diimplementasikan dahulu menjadi sebuah class. Dengan demikian interface mendukung resolusi metoda pada saat runtime.

Sesuai perkembangan class model, class dapat dikelompokkan menjadi package. Kita juga dapat membuat diagram yang terdiri atas package. Hubungan antar class

1. Assosiasi, yaitu hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class yang memiliki atribut berupa class lain atau class yang harus menunjukan arah query class lain. Panah navigability menunjukan arah query antar class.
2. 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 di warisi nya. Kebalikan dari pewarisan adalah generalisasi.
3. Agregasi, yaitu hubungan yang menyatakan bagian (“terdiri atas..”)
4. Hubungan dinamis, yaitu rangkaian pesan (message) yang di passing dari satu class kepada class lain.hubungan dinamis dapat di gambarkan dengan menggunakan sequence diagram yang akan dijelaskan kemudian.

3) Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang di rancang, 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 usecase atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara usecase menggambarkan bagaimana aktor menggunakan system untuk melakukan aktivitas. Sama seperti state, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. Decision digunakan untuk menggambarkan behaviour pada kondisi tertentu.

Untuk mengilustrasikan proses-proses paralel (fork dan join) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal, atau vertikal. Activity diagram dapat dibagi menjadi beberapa object swimlane untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

4) Sequence Diagram

Sequnce diagram menggambarkan interaksi antara objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya.) berapa message yang di gambarkan terhadap waktu. Sequence diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait.). Sequence diagram biasa nya digunaka untuk

menggambarkan skenario atau rangkaia langkah-langkah yang di lakukan sebagi respon dari sebuah event untuk menghasilkan output tertentu. Diawali dengan men trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal output apa yang di hasilkan. Masing-masing objek, termasuk aktor, memiliki lifeline vertikal. Message di gambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, message akan dipetakan menjadi operasi/metoda class. Activation br menunjukan lama nya eksekusi sebuah proses, biasanya diawali dengan diterima nya sebuah message. Untuk objek-objek yang memiliki sifat khusus, standar UML mendefenisikan icon khusus untuk object boundary, controller dan persistent entity.

5) Component Diagram

Component diagram menggambarkan struktur dari hubungan antra komponen piranti lunak, termasuk ketergantungan (dependency) diantaranya. 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 runtime. Umumnya komponen terbentuk beberapa class dan atau package, tetapi dapat juga dari komponenkomponen yang lebih kecil. Komponen dapat juga berupa interface, yaitu kumpulan yang disediakan sebuah komponen untuk komponen lain.

6) Deployment Diagram

Deployment/physical diagram menggambarkan detail bagaimana komponen di deploy dalam infrastruktur sistem, dimana 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 mendeploy komponen dalam lingkungan sebenarnya. Hubungan antar node (misalnya TCP/IP) dan requirement dapat juga di defenisikan dalam diagram ini.

3.2.4. Pengujian Software

Pengujian perangkat lunak adalah proses pengecekan terhadap suatu perangkat lunak, apakah perangkat lunak yang dihasilkan sudah dapat dijalankan sesuai dengan standar perangkat lunak yang dihasilkan sudah dapat dijalankan sesuai dengan standar tertentu. Standar yang menjadi acuan dapat berupa menurut instansi tertentu ataupun disesuaikan dengan keperluan costumer dan user. Meningkatnya visibilitas perangkat lunak sebagai suatu elemen sistem dan biaya yang muncul akibat kegagalan perangkat lunak.

Pada saat ini telah banyak berkembang berbagai metode desain pengujian perangkat lunak. Metode-metode tersebut memberikan kepada pengembang sebuah pendekatan yang sistematik dalam pengujian. Yang lebih penting pengujian ini memberikan mekanisme yang dapat membantu memastikan kelengkapan dalam pengujian dan memberikan kemungkinan tertinggi untuk mengungkap kesalahan pada perangkat lunak. Dalam

penelitian ini metode yang akan digunakan dalam pengujian perangkat lunak yang dihasilkan adalah dengan menggunakan metode *black-box*.

Metode pengujian yang akan dilakukan penulis pada pengujian perangkat lunak yang akan dibangun adalah metode pengujian *black-box*, dimana pada pengujian *black-box* ini hanya berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian didapatkan serangkaian kondisi input yang sepenuhnya mengginakan semua persyaratan fungsional untuk suatu program. Metode pengujian ini dilakukan untuk menemukan kesalahan yang mungkin terjadi pada sebuah perangkat lunak seperti :

1. Fungsi-fungsi tidak benar atau hilang.
2. Kesalahan antar muka
3. Kesalahan dalam struktur data atau akses basis data eksternal
4. Kesalahan kinerja.
5. Kesalahan inisialisasi dan kesalahan terminasi.