BAB III

OBJEK DAN METODE PENELITIAN

3.1 Objek Penelitian

Objek pada penelitian ini adalah Aplikasi Penjualan dan Pemesanan Berbasis *Web* pada *Everglow Concept Store* Penelitian dilakukan pada *Everglow Concept Store* yang beralamatkan di Jln.Gegerkalong Hilir No.156 Bandung. Everglow Concept Store merupakan perusahan yang bergerak dibidang penjualan baju.

3.1.1 Sejarah Singkat Perusahaan

Pada Tahun 2007 *Everglow Concept Store* berdiri, awalnya hanya berupa bisnis rumahan dan hanya menerima pembuatan kaos dan tempat sablon biasa saja, pengerjaannyapun dilakukan oleh 3 orang pemuda yang memiliki ambisi untuk mengembangkan bisnis distro di daerah Lembang. Tidak lama ketika pertengahan Tahun 2007, *Everglow Concept Store* diajak untuk bekerja sama dengan merk dagang lain, sehingga selain memasarkan produknya sendiri ternyata *Everglow Concept Store* juga turut membantu memasarkan produk milik distro lain.

Hingga pada suatu saat awal Tahun 2008, *Everglow Concept Store* memberanikan diri untuk membuka sebuah Distro, tentunya hal ini akan sangat bermanfaat untuk menjadi sebuah tempat penjualan yang strategis. Selain itu, juga akan menjadi solusi sebagai penjualan produk sendiri maupun produk distro yang

dikirim oleh distro lain. Sehingga terbentuklah sebuah tempat bernama *Everglow Concept Store* yang berada di Jalan Gegerkalong Hilir No.156 Bandung.

3.1.2 Visi dan Misi Perusahaan

Perlu rasanya badan usaha atau perusahaan memiliki visi dan misi sebagai pedoman, suatu awal tujuan pengembangan usahanya kedepan, dengan menitik beratkan pada visi dan misi dari perusahaan tersebut. Berikut ini adalah visi dan misi *Everglow Concept Store* :

a. Visi

Visi Distro *Everglow Concept Store* dalam mengembangkan bisnis/usahanya adalah:

Support Cloting for National maksudnya; Distro *Everglow Concept Store* menginginkan produk yang telah dibuatnya dapat dikenal, khususnya oleh kalangan masyarakat setempat, umumnya dapat dikenal oleh seluruh kalangan masyarakat di Indonesia.

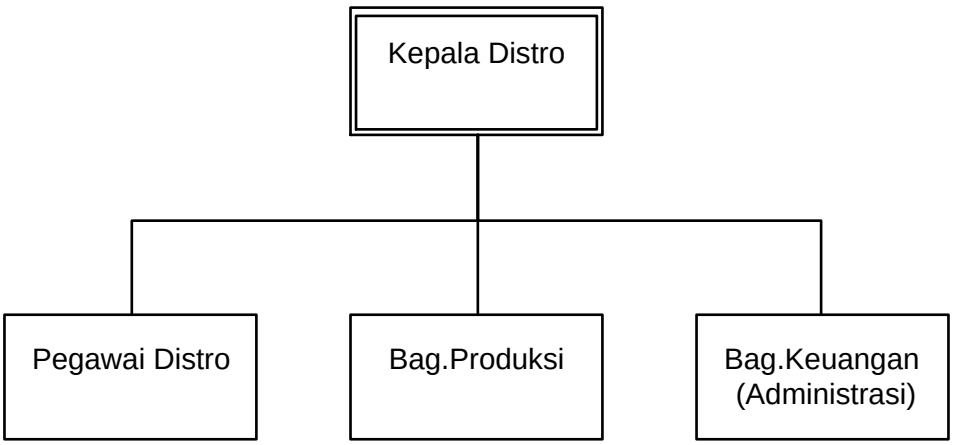
b. Misi

Misi Distro *Everglow Concept Store* adalah:

1. Menjaga mutu dan kualitas produk dengan harga yang terjangkau.
2. Memberikan kenyamanan, keamanan, yang bersahabat untuk menjalin kebersamaan antar sesama.
3. Memberikan akses terbaik guna melayani *customer* sebaik-baiknya.

3.1.3 Struktur Organisasi Perusahaan

Setiap perusahaan pasti memiliki struktur organisasi yang masing-masingnya berbeda, antara perusahaan satu dengan perusahaan yang lain. Struktur organisasi dibuat agar susunan kerja, wewenang, dan tanggung jawab suatu bagian tertentu dapat terlihat dengan jelas. Begitu juga dengan Distro *Everglow Concept Store*, sedangkan pada struktur organisasi yang akan penulis terapkan di sini hanya yang berhubungan dengan struktur alur program yang sedang penulis rancang saat ini, dapat dijelaskan sebagai berikut :



Gambar 3.1 Struktur Organisasi Distro *Everglow Concept Store*

Sumber : Dokumentasi Distro *Everglow Concept Store*, 2012

3.1.4 Deskripsi Tugas

Agar setiap pegawai dapat menjalankan fungsi serta tugasnya masing-masing dengan baik, maka perlu dijabarkan secara jelas tentang deskripsi kerja yang sesuai dengan posisi jabatan yang diambil dalam perusahaan tersebut. Adapun deskripsi kerja masing-masing bagian yang ada pada Distro *Everglow Concept Store* adalah sebagai berikut :

1. Kepala Distro

   Bertugas membuat laporan-laporan barang dagang, aktifitas pegawai, mencatat data transaksi penjualan dan pembelian, serta mencatat hutang piutang perusahaan.

2. Keuangan (Administrasi)

   Bertugas membuat data keuangan gaji pegawai, menghitung Laba/Rugi, serta menganalisa perekonomian perusahaan.

3. Bag. Produksi

   Bertugas mengawasi keluar–masuk barang kedalam gudang dan mencatat laporannya serta melakukan pembelian barang dan membuat laporannya.

4. Pegawai Distro

   Bertugas melayani dan menangani hanya bagian penjualan barang ke konsumen dan melihat persediaan barang.

3.2 Metode Penelitian

Dalam menyusun usulan penelitian ini penulis menggunakan metode penelitian yang digunakan di antaranya adalah sebagai berikut:

3.2.1 Desain Penelitian

Desain penelitian yang digunakan yaitu penelitian deskriptif dimana penelitian deskriptif merupakan metode penelitian yang berusaha menggambarkan dan menginterpretasi objek sesuai dengan apa adanya. Oleh karena itu, peneliti menggunakan metode ini sebagai suatu rancangan penelitian karena metode ini

juga menggambarkan suatu keadaan dimana peristiwa dan kondisi yang diteliti itu sedang berjalan.

3.2.2 Jenis dan Metode Pengumpulan Data

Jenis dan metode pengumpulan data yang dilakukan oleh penulis terhadap *Everglow Concept Store* yaitu menggunakan data primer dan data sekunder.

3.2.2.1 Sumber Data Primer

Data primer adalah data yang diambil secara langsung dari objek penelitian oleh peneliti perorangan maupun organisasi.

1. Wawancara

   Metode wawancara merupakan tanya jawab yang dilaksanakan penulis dengan pihak pribadi sumber. Dalam teknis wawancara ini penulis berperan sebagai pewawancara dan mewawancarai pemilik perusahaan ini. Penulis sendiri mendapatkan data primer dengan mewawancarai langsung pemimpin dan staf di *Everglow Concept Store* dengan pertanyaan-pertanyaan seputar penjualan yang dilakukan oleh perusahaan.

2. Observasi

   Metode ini merupakan teknik pengumpulan data dengan cara penulis langsung survei ke tempat penelitian yaitu guna mengetahui sistem yang sedang berjalan untuk acuan pembuatan sistem yang lebih baik lagi.

3.2.2.2 Sumber Data Sekunder

Data sekunder adalah data yang didapat tidak secara langsung dari objek penelitian. Peneliti melakukan pengumpulan data dengan menyimpan dokumen

bukti pembayaran. Selain itu penyusun mengadakan studi pustaka, merupakan teknik yang dilakukan oleh penyusun dengan cara membaca buku-buku sumber, catatan perkuliahan dan latihan-latihan yang berhubungan dengan topik yang dibahas.

3.2.3 Metode Pendekatan dan Pengembangan Sistem

Metode pendekatan dan pengembangan sistem adalah hal yang sangat penting untuk dijelaskan. Adapun metode pendekatan dan pengembangan sistem yang digunakan oleh penulis, yaitu:

3.2.3.1 Metode Pendekatan Sistem

Metode pendekatan sistem yang penulis gunakan pada penelitian ini adalah menggunakan metode berorientasi objek atau *object oriented* yang merupakan suatu paradigma baru dalam rekayasa perangkat lunak. Pendekatan berorientasi objek merupakan suatu teknik atau cara pendekatan dalam melihat permasalahan dalam sistem (sistem perangkat lunak, sistem informasi, atau sistem lainnya). Pendekatan berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan objek-objek dunia nyata. Rosa A.S-M.Shalahuddin (2011:86). Alat-alat yang digunakan dalam pendekatan analisis dan pemrograman berorientasi objek yaitu dengan notasi UML dengan membuat sembilan diagram yaitu, *Use case diagram, Activity diagram, Sequence diagram, Class diagram, Component diagram, Deployment diagram.*

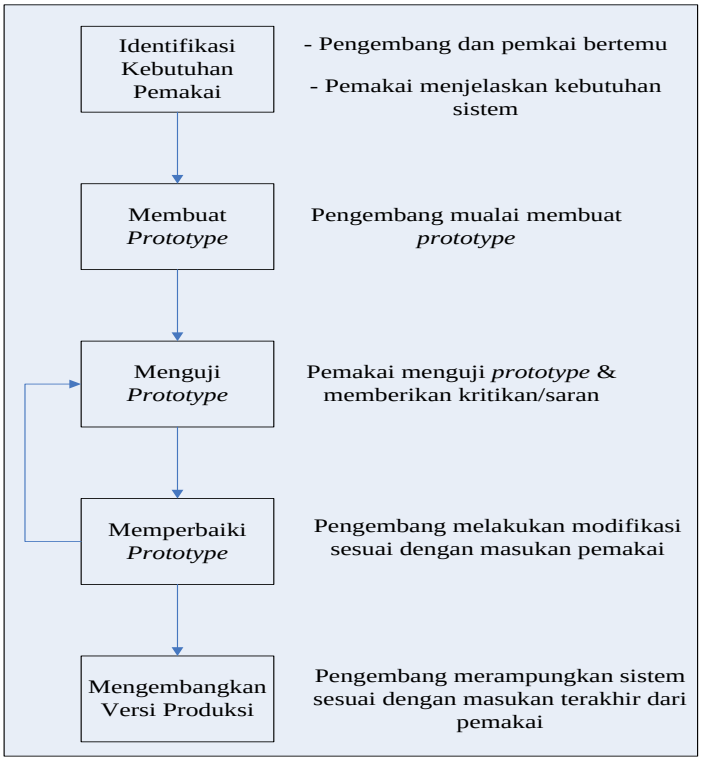
3.2.3.2 Metode Pengembangan Sistem

Metode yang digunakan untuk proses pengemangan perangkat lunak yang akan dibangun adalah model *prototype*. *Prototype* merupakan suatu metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat sesuatu program dengan cepat dan bertahap sehingga segera dapat dievaluasi oleh pemakai**.** Selain itu *prototype* juga membuat proses pengembangan sistem informasi menjadi lebih cepat dan lebih mudah, terutama pada keadaan kebutuhan pemakai sulit untuk diidentifikasi.

Secara garis besar, sasaran *prototype* adalah sebagai berikut :

1. Mengurangi waktu sebelum pemakai melihat sesuatu yang konkret dari usaha pengembangan sistem.
2. Menyediakan umpan balik yang cepat dari pemakai ke pengembang.
3. Membantu menggambarkan kebutuhan pmakai dengan kesalahan yang lebih sedikit.
4. Meningkatkan pemahaman pengembang dan pemakai terhadap sasaran yang seharusnya dicapai oleh sistem.
5. Menjadikan keterlibatan pemakai sangat berarti dalam analisis dan desain sistem.

Berikut ini merupakan mekanisme pengembangan sistem dengan menggunakan *Prototype:*



Gambar 3.2 Mekanisme Pengembangan Sistem dengan *Prototype*

Sumber : (Kadir, 2003:416)

Tahapan-tahapan *prototyping* yaitu :

1. Pengumpulan Kebutuhan

   Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasikan semua kebutuhan, dan garis besar system 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 keinginan pelanggan. Jika sudah sesuai maka langkah 4 akan diambil, jika tidak *prototyping* direvisi dengan mengulangi 1, 2 dan 3.

4. Mengkodekan 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. Pengujian ini dilakukan dengan *White Box, Black Box, Basis Path,* pengujian arsitektur dan lain-lain

6. Evaluasi Sistem

   Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan . Jika ya, langkah 7 dilakukan; jika tidak, ulangi langkah 4 dan 5.

7. Menggunakan sistem

   Perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan.

Kelebihan *Prototype*

1. Pendefinisian kebutuhan pemakai menjadi lebih baik karena keterlibatan pemakai yang lebih intensif.

2. Meningkatkan kepuasan pemakai dan mengurangi resiko pemakai tidak menggunakan sistem mengingat keterlibatan mereka yang sangat tinggi sehinga sistem memenuhi kebutuhan mereka dengan lebih baik.
3. Mempersingkat waktu pengembangan.
4. Memperkecil kesalahan disebabkan pada setiap versi prototipe, kesalahan segera terdeteksi oleh pemakai.
5. Pemakai memiliki kesempatan yang lebih banyak dalam meminta perubahan-perubahan.
6. Menghemat biaya.

Kelemahan *Prototype*

1. *Prototype* hanya bisa berhasil jika pemakai bersungguh-sungguh dalam menyediakan waktu dan pikiran untuk menggarap *prototype*.
2. Kemungkinan dokumentasi terabaikan karena pengembang lebih berkonsentrasi pada pengujian dan pembuatan *prototype*.
3. Mengingat target waktu yang pendek, ada kemungkinan sistem yang dibuat tidak lengkap dan bahkan sistem kurang teruji.
4. Jika terlalu banyak proses pengulangan dalam membuat *prototype*, ada kemungkinan pemakai menjadi jenuh dan memberikan reaksi yang negatif.
5. Apabila tidak dikelola dengan baik, *prototype* menjadi tidak pernah berakhir. Hal ini disebabkan permintaan terhadap perubahan terlalu mudah untuk dipenuhi.

3.2.3.3 Alat Bantu Analisis dan Perancangan

Sesuai dengan metode pendekatan sistem yang digunakan yaitu metode berorientasi objek, maka penulis memakai pemodelan dengan notasi UML *(Unified Modeling Language).* Untuk mendapatkan banyak pandangan terhadap sistem informasi yang akan dibangun, UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem. Ada 6 diagram yang digunakan oleh penulis, yaitu:

a. Diagram *Use Case*

Diagram Use Case atau *use case diagram* merupakan pemodelan untuk kelakuan *(behaviour)* 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. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

b. Diagram *Activity*

Activity Diagram atau 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.

c. Diagram *Sequential*

Diagram sequential atau *sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar 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 *sequence* 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 sequence sehingga semakin banyak *use case* yang didefinisikan maka diagram *sequence* yang harus dibuat juga semakin banyak

d. Diagram *Class*

Class diagram atau kelas diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut (variabel-variabel yang dimiliki oleh suatu kelas) dan metode atau operasi (fungsi-fungsi yang dimiliki oleh suatu kelas). Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem.

e. Diagram *Component*

Component diagram atau komponen diagram 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. Komponen dasar yang biasanya ada dalam suatu sistem adalah komponen *user interface* yang menangani tampilan, komponen *business processing* yang menangani fungsi-fungsi proses bisnis, komponen data yang menangani manipulasi data, dan komponen *security* yang menangani keamanan sistem.

f. Diagram *Deployment*

Deployment diagram atau *deployment* diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut:

1. Sistem tambahan *(embedded system)* yang menggambarkan rancangan *device*, *node*, dan *hardware*.
2. Sistem *client/server*
3. Sistem terdistribusi murni
4. Rekayasa ulang aplikasi

3.2.4 Pengujian *Software*

Pengujian perangkat lunak *(software)* adalah elemen kritis dari jaminan kualitas perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi, desain, dan pengkodean.

Pengujian software yang penulis gunakan dalam penelitian ini adalah *black box*. Untuk lebih jelasnya bisa lihat di bawah ini :

Faktor Pengujian *Black Box*:

1. *Graph-based* testing adalah graf yang mewakili hubungan antar objek pada modul sehingga tiap objek dan hubungannya tersebut dapat diuji.
2. *Equivalence Partitioning* adalah pembagian domain masukan dari program menjadi kelas data yang dibuatkan kasus ujinya.
3. *Boundary Value Analysis* adalah pemilihan kasus uji dengan mencari batas-batas esktrim dari kelas data.
4. *Comparison Testing* adalah digunakan untuk system yang menganut *redundancy* kasus uji yang dirancang untuk satu versi perangkat lunak dijadikan masukkan pada pengujian versi perangkat lunak lainnya.

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* bukan merupakan *alternative* dari teknik white box, tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan dari pada metode *white box*.