BAB III

OBJEK DAN METODE PENELITIAN

3.1 Objek Penelitian

Berdasarkan data dari distro & clothing after school, objek penelitiannya adalah sebagai berikut:

3.1.1 Sejarah Singkat Perusahaan

Distro & clothing after school didirikan tahun 2011 pada bulan agustus yang bergerak dalam bidang penjualan yang berbagai macam pakaian untuk remaja. Meskipun distro ini baru berdiri, sudah banyak brand-brand yang ada di toko kami diantaranya ouval research, cosmic, airplane, nepz, dead heartz, warmaple, airwaves, dll.

Adapun didirikannya distro ini berawal dari semakin berkembangnya trend dan mode yang berjalan begitu cepat khususnya di kota purwakarta, pada awal berdirinya distro after school hanyalah distro yang menjual khusus fashion untuk remaja. Dengan banyaknya bermunculan distro-distro di purwakarta, after school akan memberikan pelayanan kepada konsumen semaksimal mungkin, sehingga dapat menarik minat para konsumen. Produk yang kami pasarkan dengan upaya untuk memenuhi kebutuhan konsumen.

3.1.2 Visi dan Misi Clothing & Distro After School

Visi

Distro after school ingin mengembangkan produk ini menjadi produk yang berkualitas juga menjadi produk yang diminati oleh para kaum muda baik pria maupun wanita yang selalu mengikuti *trend fashion* dalam kesehariannya.

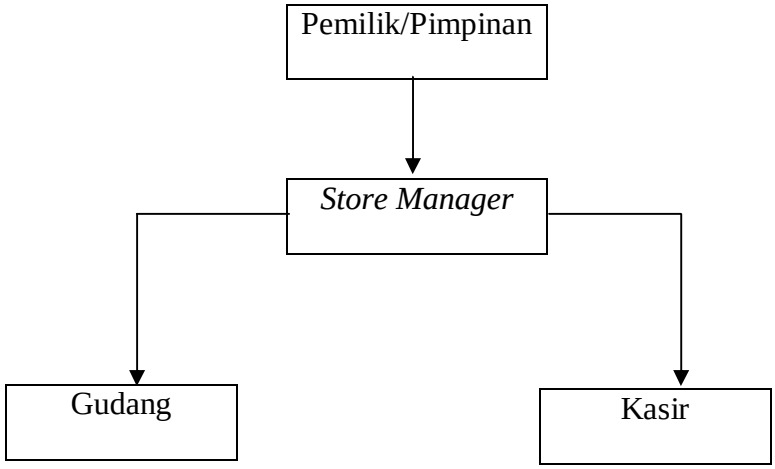
Misi

a. Menghasilkan produk terbaik sehingga banyak konsumen yang berminat untuk memakainya.
b. Fokus bisnis tertuju pada kegiatan produksi barang yang sesuai dengan spesifikasi dan permintaan konsumen.
c. Meningkatkan pelayanan terbaik kepada konsumen.

3.1.3 Struktur Organisasi Perusahaan

Dalam struktur organisasi ini akan dijelaskan mengenai rangkaian aktivitas yang terdapat di dalam perusahaan tersebut. Struktur organisasi merupakan susunan dari setiap jabatan yang sudah stabil dan merupakan salah satu faktor yang mempengaruhi orang yang tergabung dalam organisasi tersebut. Dengan adanya struktur organisasi, maka setiap pimipinan atau bawahan mengetahui dengan jelas sampai dimana tugas dan fungsi yang harus dilaksanakan atau batas wewenang dan tanggung jawab yang harus dilaksanakan. Pembagian kerja adalah suatu keharusan didalam organisasi agar tidak terjadi tumpang tindih pekerjaan. Pembagian kerja bukan hanya perlu dilihat dari manfaat yang diperoleh dari

penerapan spesialisasi tetapi mewujudkan penempatan orang yang tetap pada jabatan yang tepat. Untuk lebih jelasnya terlihat pada struktur organisasi berikut:



Gambar 3.1 Struktur organisasi clothing & distro after school.
(Sumber: *Store manager*)

3.1.4. Deskripsi Tugas

Pada distri after school ini tugas dan wewenang dari masing-masing bagian dalam perusahaan dapat diuraikan sebagai berikut:

1. Pemilik/Pemimpin
    a. Mendapatkan laporan operasional dari seluruh kegiatan perusahaan termasuk arus keluar masuknya barang dan keuangan.
    b. Mengkoordinasikan operasional secara keseluruhan dan memberikan tugas dan wewenangnya kepada masing-masing personal yang terkait dalam perusahaan.

2. *Store manager*
    a. Memimpin dan mengkoordinasikan seluruh kegiatan dan memberikan tugas-tugas kepada bawahannya pada masing-masing toko dalam melaksanakan tugas-tugas setiap personalnya.
    b. Memberikan laporan kepada pemimpin perusahaan mengenai kegiatan operasional pada toko yang dibawahinya.
    c. Mempersiapkan data keuangan untuk kepentingan penyusunan anggaran.
3. Gudang
    a. Mencatat setiap ada pengeluaran barang atau seluruh barang yang terjual.
    b. Memberikan pelayanan terbaik kepada konsumen.
4. Kasir
    a. Mmbuat data barang baru yang masuk daru *supplier*.
    b. Mencatat setiap barang yang terjual
    c. Memberikan pelayanan terbaik kepada konsumen.

3.2. Metedologi Penelitian

3.2.1. Desain Penelitian

Metedologi penelitian merupakan ilmu tentang kerangka kerja pada penelitian yang bersistem. Sistem penelitian dikerjakan secara kontekstual, konteks penelitian tersusun atas unsur-unsur yang menjadi pangkal beranjak pemikiran yang membentuk gagasan dasar konsep. Menurut M. Nazir dalam bukunya metode penelitian menyebutkan definisi desain penelitian adalah sebagai berikut: “Desain penelitian adalah semua proses yang diperlukan dalam perencanaan

dan pelaksanaan penelitian. Berdasarkan definisi di atas penulis dapat menyimpulkan bahwa *desain* penelitian adalah proses perencanaan yang dilakukan sebelum melaksanakan penelitian.

Penulis memilih jenis desain penelitian dengan data primer/sekunder, karena jika peneliti membutuhkan data primer maka peneliti dapat langsung *observasi* ke tempat penelitian sedangkan jika membutuhkan data sekunder maka peneliti dapat mengevaluasi kembali data-data primer yang didapat dari tempat penelitian tersebut.

3.2.2 Jenis dan Metode Pengumpulan Data

3.2.2.1 Sumber Data Primer

Adapun metode yang digunakan dalam pembuatan skripsi ini adalah : Metode Penelitian, yaitu :

a. Observasi

   Yaitu merupakan sutau teknik pengumpulan data yang efektif untuk mempelajari sistem, dengan cara mengamati langsung objek penelitian yaitu di clothing & distro after school purwakarta.

b. Wawancara

   Yaitu sutu teknik pengumpulan data dengan cara tanya jawab langsung mengenai data yang diperlukan dari masalah yang akan diangkat.

3.2.2.2 Sumber Data Sekunder

Data yang diperoleh secara tidak langsung dari sumber penelitian, dapat dilakukan dengan studi pustaka yaitu mengumpulkan data dengan cara membaca buku-buku tertulis lainnya untuk menyusun laporan tugas akhir ini.

3.2.3 Metode Pendekatan Dan Pengembangan Sistem

3.2.3.1 Metode Pengembangan Sistem

Metode yang digunakan dalam pengembangan untuk membangun sistem informasi ini yaitu metode *prototype* yang dimana *prototype* merupakan suatu metode dalam pemgembangan sistem yang menggunakan pendekatan untuk membuat suatu program dengan cepat dan bertahap dan *prototype* juga membuat proses pengembangan sistem informasi menjadi lebih baik cepat dan lebih muda. Dimana tahapan-tahapan yang harus dilaksanakan adalah sebagai berikut:

1. Identifikasi kebutuhan sistem

Pada tahap ini merupakan tahap awal dalam membangun sebuah sistem informasi, dimana antara pemakai sistem (*user*) dan pengembang sistem bertemu. User dijelaskan tentang kebutuhan sistem yang akan dibangun oleh pengembang sistem.

2. Membuat *prototype*

Setelah menganalisa sistem yang akan dibangun, pengembangan sistem mulai membuat *prototype*.

3. Menguji *prototype*

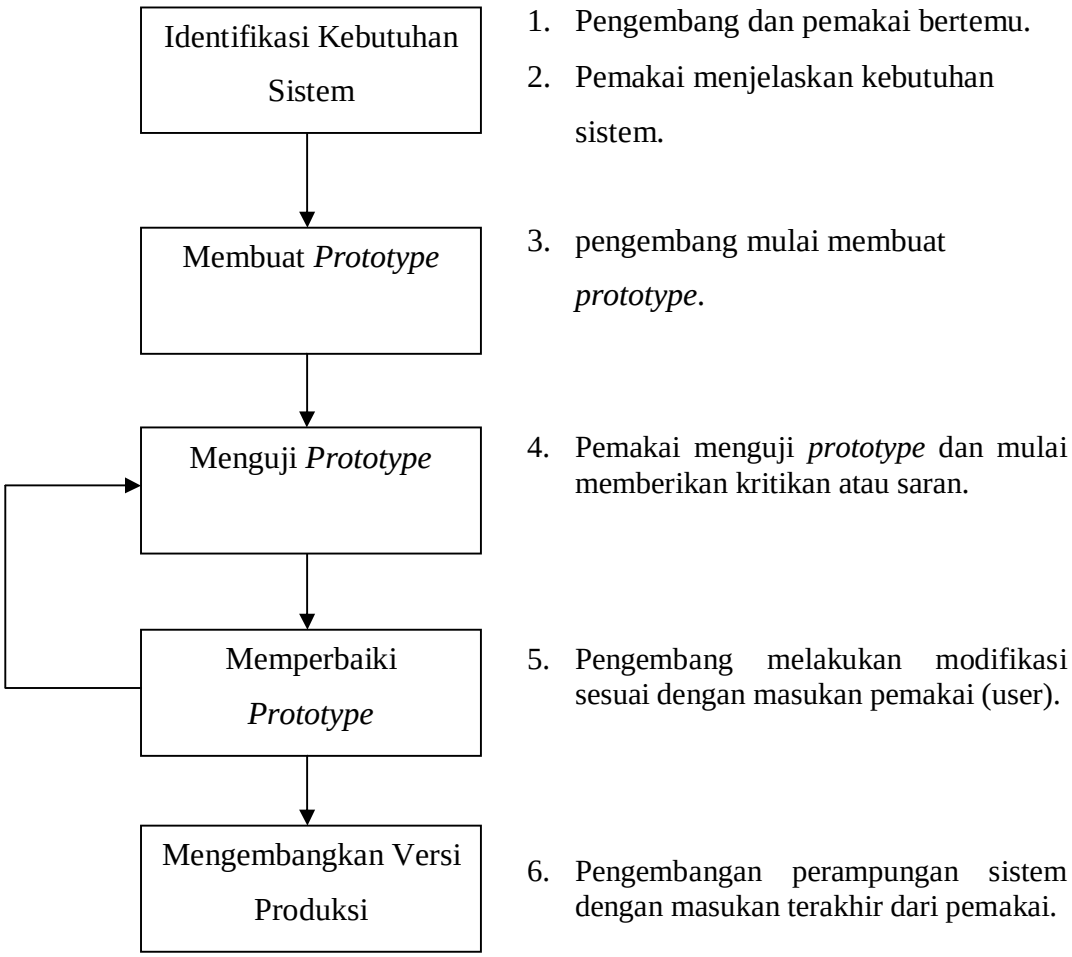
Setelah tahap ini pembuatan *prototype* selesai, kemudian pengembang sistem dan *user* melakukan pengujian program agar program dapat digunakan sesuai dengan kebutuhan, dan *user* memberikan saran atau masukan bila terdapat kekurangan pada program.

4. Memperbaiki *prototype*

Pada tahap ini pengembang sistem melakukan perbaikan dan modifikasi sesuai dengan masukan dan saran dari *user*.

5. Mengembangkan versi produksi

Pada tahap ini pengembang sistem menyelesaikan sistem yang telah dibuatnya sesuai masukan atau saran terakhir dari pemakai sistem. Dibawah ini adalah tahapan pendekatan *prototype* yang ditujukkan pada gambar berikut:



Gambar 3.2 Tahapan Pendekatan *Prototype*
[Sumber :[Susanto Azhar 2004]]

Kelebihan:

Prototype	*waterfall*
1. Metode ini cukup efektif sebagai paradigma dalam rekayasa perangkar lunak. Kuncinya adalah mendapatkan kebutuha dan aturan yang jelas yang disetujui pelanggan dan pembuat perangkat lunak. Walaupun pada umumnya prototype akan dihilangkan dan dibuat perangkat yang sebenarnya.	1. Metode ini lebih baik jika kebutuhan sudah diketahui dengan baik.

Kekurangan:

Prototype	*waterfall*
1. Pelanggan kadang tidak menyadari bahwa mungkin saja *prototype* dibuat terburu-buru dan rancangan tidak tersusun dengan rapi. 2. Pengembang kadang-kadang membuat suatu implementasi sembarangan karena ingin selesai dengan cepat.	1. Pada kenyataannya jarang mengikuti urutan *sekuensial* seperti pada teori *interasi* sering terjadi menyebabkan masalah yang baru. 2. Pelanggan harus sabar karena pembuat perangkat lunak akan dimulai pada tahap desain sudah sesuai.

3.2.3.2 Alat Bantu Analisis dan Perancangan

Alat bantu analisis dan perancangan yang digunakan adalah metode analisis terstruktur berorientasi data. Adapun metode desain yang ada pada langkah perancangan ini akan digambarkan dalam bentuk bagan alir dokumen (*flowmap*), *diagram konteks, data flow diagram,* dan *kamus data*.

1. Flow map

Diagram alir dokumen (*flowmap*) merupakan gambaran hubungan antaraentitas yang terlihat berupa aliran-aliran dokumen yang ada. Bagai alir dokumen disebut juga bagan alir formulir yang merupakan bagan alir yang menunjukkan arus laporan dan formulir.

2. Diagram konteks

Diagram kontek nerupakan suatu diagram alir yang tingkat tinggi yang menggambarkan seluruh jaringan, masukan dan keluaran. *sistem* yang dimaksud adalah untuk menggambarkan sistem yang sedang berjalan. mengidentifikasikan awal dan akhir data awal dan akhir yang masuk dan keluaran sistem. Diagram ini merupakan gambaran umum *sistem* yang nantinya akan kita buat. secara uraian mah dapat dikatakan bahwa *diagram kontek* itu berisi siapa saja yang memberikan data (inputan) ke *sistem* serta kepada siapa data informasi yang harus dihasilkan *sistem.*

3. Data flow diagram

Data flow diagram merupakan alat perancangan sistem yangberorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh professional

sistem kepada pemakai maupun pembuat program. Dalam *DFD* dibahas fungsi apa saja yang diperlukan oleh suatu sistem dan aliran data yang terdapat diantara proses didalamnya. *DFD* berguna sebagai alat untuk memverifikasi apakah sistem yang akan dibangun sudah memenuhi kriteria yang diinginkan user atau belum.

Data Flow Diagram dapat dikembangkan dari level yang paling rendah ke level yang paling tinggi. *DFD* level 0 merupakan pengembangan dari diagram konteks, *DFD* level 1 merupakan pengembangan dari *DFD* level 0. Tiap proses dari *DFD* dapat dikembangkan lagi menjadi lebih detil sampai proses-proses tersebut tidak dapat dikembangkan lagi.

4. Kamus data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan *sistem,* sehingga user dan analis *sistem* mempunyai pengertian yang sama tentang *input*, *output*, dan komponen data. Kamus data ini sangat membantu analis *sistem* dalam mendefinisikan data yang mengalir di dalam *sistem,* sehingga pendefinisian data itu dapat dilakukan dengan lengkap dan terstruktur. Pembentukan kamus data dilaksanakan dalam tahap analisis dan perancangan suatu *sistem.*

Pada tahap analisis, kamus data merupakan alat komunikasi antara *user* dan analis *sistem* tentang data yang mengalir di dalam sistem, yaitu tentang data yang masuk ke *sistem* dan tentang informasi yang dibutuhkan oleh user. Sementara itu pada tahap perancangan *sistem* kamus data digunakan untuk merancang *input*, laporan dan *database*. Pembentukan kamus data didasarkan atas alur data yang terdapat pada *DFD.* Alur data pada *DFD* ini bersifat global, dalam

arti hanya menunjukan nama alur datanya tanpa menunjukan struktur dari alur data itu. Untuk menunjukan struktur dari alur data secara terinci maka dibentuklah kamus data yang didasarkan pada alur data di dalam *DFD*.

5. Perancangan *basis data*

A. Normalisasi

Normalisasi merupakan cara atau proses untuk mengidentifikasi "*tabel*" kelompok atribut yang memiliki ketergantungan yang sangat tinggi antar atribut satu dengan atribut yang lainnya. Perancangan basis data diperlukan agar kita bisa memiliki babis data yang kompak dan efisien dalam ruang penyimpanan, cepat dalam mengakses dan mudah dalam pemanipulasia (ubah, tambah, hapus) data.

Langkah-langkah pembentukkan normalisasi antara lain:

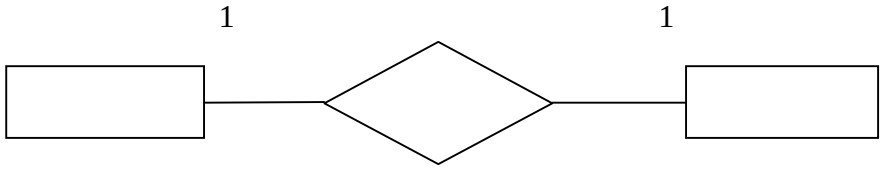
a. Bentuk tidak normal (*unnormalized form*)
b. Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti format tertentu, dapat saja data tidak lengkap atau terduplikasi. Data dikumpulan ada adanya sesuai dengan penginputan.
c. Bentuk norma pertama (*first normal form*)
d. pada tahap ini dilakukan penghilangan beberapa group elemen yang agar menjadi satu harga tunggal yang berinteraksi di diantara setiap baris pada suatu tabel, dan setiap atribut harus mempunyai nilai data.
e. Bentuk normal kedua (*second normal form*)
f. Normalisasi ketiga berfungsi bahwa relasi harus dalam bentuk normal kedua dan tidak boleh ada ketergantungan antara *field-field* non kunci.

B. Tabel relasi

Dalam sebuah *database,* data tabel memiliki sebuah *filed* yang memiliki nilai unik untuk setiap *field* baris. *Field* ini ditandai dengan *icon* bergambar kunci didepan namanya, baris-baris yang berhubungan pada tabel mengulangi kunci *primer (primary key)* dari baris yang sihubungkannya pada tabel lain, salinan dari kunci primer didalam tabel – tabel yang lain disebut dengan kunci asing. Kunci asing ini tidak perlu bersifat unik dan semua *field* yang bisa menjadi kunci asing yang membuat sebuah *field* merupakan kunci asing adalah jika dia sesuai dengan kunci *primer*. Pada relasi tabel terdapat 3 macam hubungan yaitu:

a. One to one

Tingkat hubungan ini menunjukkan tingkat hubungan satu ke satu, dinyatakan dengan satu kejadian pada *entitas* pertama, dan hanya mempunya satu hubungan dengan satu kejadian pada cerita yang kedua dan sebaliknya. Artinya setiap *tupelo* pada *entitas* A berhubungan dengan paling banyak satu tuelo pada *entitas* B, dan begitu juga sebaliknya setiap *tupelo* pada *entitas* B berhubungan dengan paling banyak satu *tupelo* pada *entitas* A.

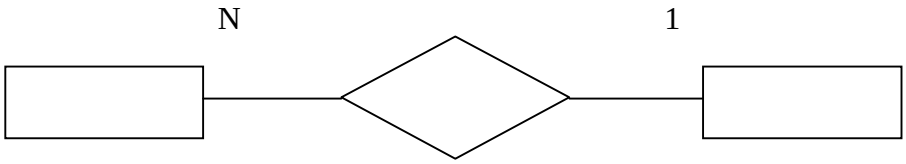


Gambar 3.3 : Relasi satu ke satu *(one to one)*

b. Many to one

Tingkat hubungan dari satu ke banyak adalah sama dengan banyak kesatu, tergantung dari arah mana hubungan tersebut untuk satu kejadian pada *entitas* yang pertama dapat mempunyai banyak hubungan dengan kejadian pada

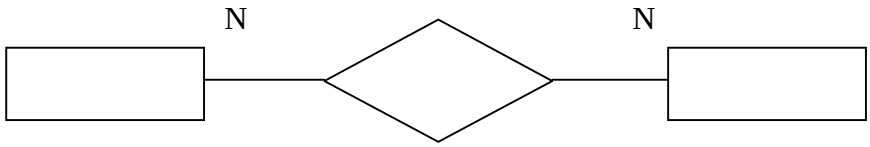
entitas kedua. Sebaliknya, satu kejadian pada *entitas* kedua hanya dapat mempunyai satu hubungan dengan satu kejadian pada *entitas* pertama.



Gambar 3.4: Relasi banyak ke satu *(many to one)*

c. Many to many

Tingkat hubungan dari banyak ke banyak terjadi jika tiap kejadian pada seluruh *entitas* akan mempunyai banyak hubungan dengan kejadian pada *entitas* lainnya, dilihat dari sisi *entitas* yang pertama maupun yang dilihat dari sisi entitas yang kedua.

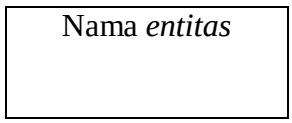


Gambar 3.5: Relasi banyak ke banyak *(many to many)*

C. Entity relasionship diagram

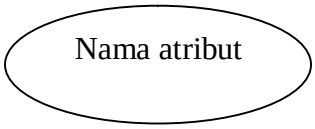
Entity relasionship diagram (*ERD*) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. *ERD* merupakan model jaringan data yang menekankan pada struktur-struktur dan *relationship data*. Tujuan utama digunakannya *ERD* adalah menunjukkan objek-objek (*entitas*) apa saja yang ingin dilibatkan dalam sebuah basis data dan bagaimana hubungan yang terjadi diantara objek tersebut.

1. Persegi panjang, menyatakan himpunan entitas.



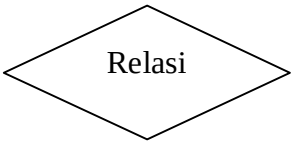
Entitas

2. Lingkaran atau elips, menyatakan atribut yang berfungsi sebagai key.



Atribut

3. Belah ketupat, menyatakan himpunan relasi.

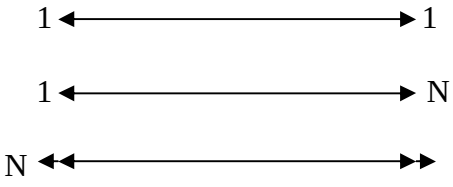


Relasi

4. Garis, sebagai penghubunga antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atributnya.

Garis

5. Kardinalitas relasi dapat dintyatakan dengan banyaknya garis cabang atau dengan pemakai angka (1 dan 1 untuk relasi satu ke satu, 1 dan N untuk relasi satu ka banyak atau N dan N untuk relasi banyak ke banyak.



Kardinalitas relasi.

D. Entity

Entity merupakan elemen yang disebut atribut dan berfungsi mendeskripsika karakter *entitas*. Pada *ERD entity* digunakan dengan sebuah bentuk persegi panjang. *Entity* diberi sama dengan kata benda dan dapat dikelompokkan dalam empat jens nama yaitu orang, benda, kejadian (terdapat unsur waktu didalamnya).

3.2.4. Pengujian *Software*

Pengujian perangkat lunak (*software*) menggunakan metode pengujian *Black Box*. Metode uji coba *Black Box* memfokuskan pada keperluan fungsional dari *software*. Ujicoba *Black Box* berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya :

1. Fungsi-fungsi yang salah atau hilang.
2. Kesalahan *interface.*
3. Kesalahan dalam struktur data atau akses *database eksternal.*
4. Kesalahan performa.
5. Kesalahan *inisialisasi* dan *terminasi.*

Uji coba didesain untuk dapat menjawab pertanyaan-pertanyaan berikut :

1. Bagaimana *validitas* fungsionalnya diuji?
2. Jenis input seperti apa yang akan menghasilkan kasus uji yang baik ?
3. Apakah sistem secara khusus sensitif terhadap nilai input tertentu ?
4. Bagaimana batasan-batasan kelas data diisolasi?
5. Berapa *rasio* data dan jumlah data yang dapat ditoleransi oleh *sistem*?
6. Apa akibat yang timbul dari kombinasi spesifik data pada operasi *sistem*?

Dengan mengaplikasikan ujicoba *Black Box*, diharapkan dapat menghasilkan sekumpulan kasus uji yang memenuhi kriteria berikut:

1. Kasus uji yang berkurang jika jumlahnya lebih dari 1, maka jumlah dari uji kasus tambahan harus didesain untuk mencapai uji coba yang cukup beralasan.
2. Kasus uji yang memberitahukan sesuatu tentang keberadaan atau tidaknya suatu jenis kesalahan, daripada kesalahan yang terhubung hanya dengan suatu uji coba yang spesifik.