

BAB III

LANDASAN TEORI

3.1 Rekrutmen

Flippo (1984) mendefinisikan sebagai berikut: *“Penarikan calon pegawai atau tenaga kerja adalah proses pencarian tenaga kerja yang dilakukan secara seksama, sehingga dapat merangsang mereka untuk mau melamar jabatan tertentu yang ditawarkan oleh organisasi”*.

Werther dan Davis (1996) mendefinisikan sebagai *“proses untuk mendapatkan dan untuk merangsang pelamar-pelamar yang mempunyai kemampuan agar mnejadi pegawai.”*

Menurut Musselman dan Hughes (1981) mendefinisikan sebagai *“Penarikan calon pegawai adalah proses pembentukan sekumpulan pelamar yang memiliki kualitas tertentu.”*

Dari ketiga definisi tersebut dapat ditarik kesimpulan bahwa penarikan calon pegawai adalah serangkaian kegiatan yang dilakukan secara terencana, guna memperoleh calon-calon pegawai yang memenuhi syarat-syarat yang dituntut oleh suatu jabatan tertentu, yang dibutuhkan oleh suatu organisasi. (Amin, 2011)

3.2 Program Aplikasi

Program merupakan ekpresi, pernyataan kombinasi yang disusun dan dirangkai menjadi satu kesatuan prosedur yang berupa urutan langkah untuk menyelesaikan masalah yang diimplementasikan dengan menggunakan bahasa pemrograman, sehingga dapat dieksekusi oleh komputer. Aplikasi merupakan

penerapan, menyimpan sesuatu hal, data, permasalahan pekerjaan kedalam suatu sarana atau media yang digunakan untuk menerapkan atau mengimplementasikan hal atau permasalahan tersebut sehingga berubah menjadi suatu bentuk yang baru tanpa menghilangkan nilai-nilai dasar dari hal, data, permasalahan atau pekerjaan. Jadi dalam hal ini hanya bentuk dari tampilan data yang berubah, sedangkan isi yang memuat dalam data tersebut tidak mengalami perubahan.

Berdasarkan pengertian dari program dan aplikasi diatas maka dapat disimpulkan, program aplikasi adalah sederetan kode yang digunakan untuk mengatur komputer agar dapat melakukan pekerjaan sesuai dengan keinginan dari permasalahan pengguna (Jogiyanto, 2000).

3.3 *Software Development Life Cycle*

Daur hidup pengembangan system, yang dalam bahasa asing disebut dengan *software development life cycle* (SDLC) berfungsi untuk menggambarkan tahapan-tahapan utama dan langkah-langkah untuk pengembangan sistem.

3.3.1 Analisis

Tahap analisis adalah sebuah proses investigasi terhadap sistem yang sedang berjalan dengan tujuan untuk mendapatkan jawaban mengenai pengguna sistem, cara kerja sistem dan waktu penggunaan sistem. Dari proses analisa ini akan didapatkan cara untuk membangun sistem baru (Melengo, 2013).

3.3.2 Desain

Desain merupakan sebuah proses perancangan suatu sistem yang akan di bangun baik itu dari sisi desain *layout* atau tampilanya (nila artistik dan estetika), berdasarkan dari analisa sebelumnya (Permana, 2013). Dalam desain, ada empat macam desain yaitu:

A. Desain Proses

Desain proses menggambarkan bagaimana mengubah hasil analisis kedalam bentuk *Context Diagram* dan *Data Flow Diagram*.

A.1. Context Diagram

Context Diagram merupakan langkah pertama dalam desain proses. Pada *Context Diagram* dijelaskan sistem apa yang dibuat dan entitas eksternal apa saja yang terlibat. Dalam *Context Diagram* harus ada arus data yang masuk dan arus data yang keluar (Kendall, 2003).

A.2. Data Flow Diagram

DFD level 0 adalah langkah selanjutnya setelah *Context Diagram*. Pada langkah ini, digambarkan proses-proses yang terjadi dalam sistem informasi. Setelah selesai maka akan di dekomposisi menjadi DFD Level 1 yang merupakan penjelasan dari DFD level 0. Pada proses ini dijelaskan proses apa saja yang dilakukan pada setiap proses yang terdapat di DFD level 0 (Kendall, 2003).

B. Desain Data

Desain data adalah aktivitas pertama dan terpenting dari empat aktivitas desain yang dilakukan selama rekayasa perangkat lunak. Proses pemilihan

struktur dalam menentukan desain yang paling efisien sesuai kebutuhan. Tujuannya adalah untuk mendapatkan struktur data yang baik sehingga diperoleh program yang lebih modular dan mengurangi kompleksitas pengembangan perangkat lunak (Astari, 2013).

Bagian desain data terdiri dari tiga bagian yaitu:

B.1. ER-Model

ER-Model (Komputer, 2013) adalah model data konseptual tingkat tinggi untuk perancangan basis data. Model data konseptual adalah himpunan konsep yang mendeskripsikan struktur basis data, transaksi pengambilan dan pembaruan basis data. Konsep paling dasar di model ER adalah entitas, relasi dan atribut. Komponen-komponen utama *ER-Model* adalah:

1. Entitas (*entity*) yang digunakan untuk memodelkan objek-objek yang berada di perusahaan/lingkungan.
2. Relasi (*relationship*) yang dipakai untuk memodelkan koneksi/hubungan di antara entitas-entitas.
3. Atribut-atribut (properti-properti), memodelkan properti-properti dari entitas dan relasi.
4. Konstrain-konstrain (batasan-batasan) integritas, konstrain-konstrain ketentuan validitas.

Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain. Sebuah kursi yang kita

duduki, seseorang yang menjadi pegawai di sebuah perusahaan dan sebuah mobil yang melintas di depan kita adalah entitas.

Sekelompok entitas yang sejenis dan berada dalam lingkup yang sama membentuk sebuah himpunan entitas (*entity sets*). Sederhananya, entitas menunjuk pada individualitas suatu objek, sedang himpunan entitas menunjuk pada rumpun (*family*) dari individu tersebut.

Seorang pasien, misalnya akan dimasukkan dalam himpunan entitas pasien. Sedang seorang dokter akan ditempatkan dalam himpunan entitas dokter. Dalam berbagai pembahasan atau literatur, penyebutan himpunan entitas (yang kurang praktis) ini seringkali digantikan dengan sebutan entitas saja. Karena itu sering ditemui, penggunaan istilah entitas (*entity*) di sebuah literatur sebenarnya menunjuk pada himpunan entitas.

B.2. Normalisasi

Normalisasi merupakan teknik analisis data yang mengorganisasikan atribut-atribut data dengan cara mengelompokkan sehingga terbentuk entitas yang *non-redundant*, stabil, dan fleksibel. Normalisasi dilakukan sebagai uji coba pada suatu tabel secara berkelanjutan untuk menentukan apakah tabel itu sudah baik, yaitu dapat dilakukan proses *insert*, *update*, *delete*, dan modifikasi pada satu atau beberapa atribut tanpa mempengaruhi integritas data dalam relasi tersebut. Pada proses normalisasi terhadap tabel pada database dapat dilakukan dengan tiga tahap normalisasi antara lain:

1. Bentuk Tidak Normal

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti format tertentu, dapat saja data tidak lengkap atau terduplikasi. Data dikumpulkan apa adanya sesuai dengan saat menginput.

2. Bentuk Normal Pertama (1NF)

Pada tahap ini dilakukan penghilangan beberapa kelompok elemen yang berulang agar menjadi satu harga tunggal yang berinteraksi di antara setiap baris pada suatu tabel, dan setiap atribut harus mempunyai nilai data yang atomic (bersifat *atomic-value*). Atom adalah zat terkecil yang masih memiliki sifat induknya, bila terpecah lagi maka ia tidak memiliki sifat induknya.

Syarat normal ke satu (1-NF) antara lain:

- a. Setiap data dibentuk dalam *flat-file*, data dibentuk dalam satu *record* demi satu *record* nilai dari atribut berupa “*atomic value*”.
- b. Tidak ada seperangkat atribut yang berulang atau bernilai ganda.
- c. Telah ditentukannya *primary key* untuk tabel / relasi tersebut.
- d. Setiap atribut hanya memiliki satu pengertian.

3. Bentuk Normal Kedua (2NF)

Bentuk normal kedua didasari atas konsep *full functional dependency* (ketergantungan fungsional sepenuhnya) yang dapat didefinisikan sebagai berikut. Jika A adalah atribut dari suatu relasi, B dikatakan *full functional dependency* (memiliki ketergantungan fungsional) terhadap A, tetapi tidak secara tepat memiliki ketergantungan fungsional dari subset (himpunan bagian) A. Syarat normal kedua (2-NF) sebagai berikut:

- a. Bentuk data telah memenuhi kriteria bentuk normal kesatu.

- b. Atribut bukan kunci (*non-key attribute*) harus memiliki ketergantungan fungsional sepenuhnya pada kunci utama atau *primary key*.

4. Bentuk Normal Ketiga (3NF)

Walaupun relasi 2-NF memiliki redundansi yang lebih sedikit dari pada relasi 1-NF, namun relasi tersebut masih mungkin mengalami kendala bila terjadi anomali peremajaan (*update*) terhadap relasi tersebut. Jika kita hanya mengupdate satu baris saja, sementara baris yang lainnya tidak, maka data didalam basis data tersebut menjadi tidak konsisten atau tidak teratur. Anomali terhadap peremajaan ini disebabkan oleh suatu ketergantungan transitif (*transitive dependency*). Kita harus menghilangkan ketergantungan tersebut dengan melakukan normalisasi ketiga (3-NF). Syarat normal ketiga (3-NF) adalah sebagai berikut:

- a. Bentuk data telah memenuhi kriteria bentuk normal kedua.
- b. Atribut bukan kunci (*non-key attribute*) harus tidak memiliki ketergantungan transitif, dengan kata lain suatu atribut bukan kunci tidak boleh memiliki ketergantungan fungsional (*functional dependency*) terhadap atribut bukan kunci lainnya, seluruh atribut bukan kunci pada suatu relasi hanya memiliki ketergantungan fungsional terhadap *primary key* di relasi itu saja (Jewyer, 2013).

C. Desain Antar Muka

C.1. Perangkat Keras

Perangkat keras komputer atau *hardware* adalah perangkat pada komputer yang berbentuk fisik (dapat disentuh). Perangkat komputer sendiri

dibedakan menjadi dua yaitu perangkat keras dan perangkat lunak. Perangkat keras sendiri berfungsi dengan baik dikarenakan adanya perangkat lunak sebagai sistem yang menjalankannya.

Agar proses pembelajaran mengenai komputer lebih terarah adapun pengelompokan perangkat keras agar lebih mudah sesuai dengan fungsinya pada komputer. Berikut ini pengelompokan perangkat keras komputer:

1. *Input Device*: Perangkat masukan.
2. *Process Device*: Perangkat yang menjalankan proses sistem komputer.
3. *Output Device*: Perangkat keluaran, menghubungkan sistem keluar.
4. *Storage Device*: Perangkat untuk menyimpan. (Affandi, Perangkat Keras Komputer, 2013)

C.2. Perangkat Lunak

Karena disebut sebagai perangkat lunak, maka sifatnya pun berbeda dengan [perangkat keras](#). Jika perangkat keras adalah komponen yang nyata yang dapat dilihat dan disentuh oleh secara langsung manusia, maka perangkat lunak tidak dapat disentuh dan dilihat secara fisik, perangkat lunak memang tidak tampak secara fisik dan tidak berwujud benda namun bisa untuk dioperasikan.

[Pengertian perangkat lunak komputer](#) adalah sekumpulan data elektronik yang disimpan dan diatur oleh komputer, data elektronik yang disimpan oleh komputer itu dapat berupa program atau instruksi yang akan menjalankan suatu perintah. Melalui **perangkat lunak** inilah suatu komputer dapat menjalankan suatu perintah (Vicky, 2013).

C.3. Jaringan Komputer

Jaringan komputer adalah sebuah sistem yang terdiri atas beberapa komputer yang didesain sedemikian rupa sebagaimana tujuan utamanya yakni untuk dapat berbagi sumber daya (CPU, printer), berkomunikasi (pesan instan, surel), dan dapat mengakses informasi (situs web). Sementara menurut pembagiannya, jaringan komputer dapat dibedakan menjadi dua jenis, yakni jaringan terdistribusi dan jaringan tersentralisasi. Manfaat jaringan komputer:

1. Menjadikan sistem komputer menjadi lebih mudah dioperasikan dan fleksibel.
2. Banyak aplikasi pendukung yang dapat dijalankan di berbagai macam jenis komputer.
3. Tidak menjadikan ketergantungan terhadap satu jenis komputer (sehingga jenis komputer lain bisa dikoordinasikan secara bersamaan).
4. Reabilitas tinggi dan dapat membagi sumber daya.
5. Memperluas pendayagunaan sistem operasi.
6. Memperluas kemudahan berkomunikasi.
7. Memudahkan kecepatan mengakses informasi. (Prasty, 2013)

D. Desain Fisik

D.1. *Physical Data Model*

PDM merupakan gambaran secara detail basis data dalam bentuk fisik. Penggambaran rancangan PDM memperlihatkan struktur penyimpanan data yang benar pada basis data yang digunakan sesungguhnya (Andiyono, 2013).

D.2. Kamus Data

Kamus data berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan secara detail dan mengorganisasi semua elemen data yang digunakan dalam sistem secara persis sehingga pemakai dan penganalisis sistem mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses. Kamus data sering disebut juga dengan sistem *data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi.

Dengan menggunakan kamus data, analisis sistem adapat mendefinisikan data yang mengalir di sistem dengan lengkap. Pada tahap analisis, kamus data digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir dari sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem (Assafiiyah, 2013).

3.3.3 Pengujian

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan spesifikasi, desain dan pengkodean. Meningkatnya visibilitas (kemampuan) perangkat lunak sebagai suatu elemen sistem dan “biaya” yang muncul akibat kegagalan perangkat lunak, memotivasi dilakukannya perencanaan yang baik melalui pengujian yang teliti. Pada dasarnya, pengujian merupakan satu langkah dalam proses rekayasa perangkat lunak yang dapat dianggap sebagai hal yang merusak daripada membangun.

A. *Black Box Testing*

Test case ini bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya, apakah pemasukan data keluaran telah berjalan sebagaimana yang diharapkan dan apakah informasi yang disimpan secara eksternal selalu dijaga kemutakhirannya.

Teknik pengujian *black-box* berfokus pada domain informasi dari perangkat lunak, dengan melakukan *test case* dengan mempartisi domain masukan dari suatu program dengan cara yang memberikan cakupan pengujian yang mendalam.

Metode pengujian *graph-based* mengeksplorasi hubungan antara dan tingkah laku objek-objek program. Partisi ekivalensi membagi domain masukan ke dalam kelas data yang mungkin untuk melakukan fungsi perangkat lunak tertentu. Analisis nilai batas memeriksa kemampuan program untuk menangani data pada batas yang dapat diterima.

Metode pengujian yang terspesialisasi meliputi sejumlah luas kemampuan perangkat lunak dan area aplikasi. Antar muka pengguna, arsitektur *client/server*, dokumentasi dan fasilitas *help* dan sistem *real time* masing-masing membutuhkan pedoman dan teknik khusus untuk pengujian perangkat lunak (Alfiyanto, 2013)