BAB II

LANDASAN TEORI

2.1. Konsep Dasar Program

Di dalam sub bab berikut menjelaskan mengenai pengertian dari program, bahasa pemrograman, basis data, model pengembangan perangkat lunak yang digunakan.

2.1.1. Program

Menurut Prof. Dr. H. Sahyar, M.S. (2016:2) ”Program komputer adalah perintah-perintah atau intruksi yang disusun berdasarkan algoritma dengan bahasa pemrograman untuk penyelesaian suatu masalah”.

Untuk dapat membuat program maka kita harus mengetahui langkah-langkah yang diperlukan untuk membuat program, yaitu :

1. Mengetahui persoalan yang akan diselsaikan.
2. Menganalisa penyelesaian persoalan.
   a. Menentukan peubah (*variabel*) yang digunakan
   b. Menentukan tekhnik penyelesaiannya.

      misalnya : dengan rumus
3. Membuat algoritma dan atau *flowchart*-nya
4. Mengubah algoritma/*flowchart* tersebut kedalam bahasa pemrograman.
5. Menjalankan program dan mengevaluasinya.
6. Implementasi dan penggunaan program.

2.1.2. Bahasa Pemrograman

Menurut Prof. Dr. H. Sahyar, M.S. (2016:2) ”Bahasa pemrograman adalah perangkat lunak yang digunakan untuk menerjemahkan atau menuliskan algoritma dalam bentuk teks atau perintah-perintah yang dimengerti oleh komputer untuk menyelsaikan suatu masalah”.

Berdasarkan hierarki dikenal beberapa bahasa pemrograman diantaranya :

1. Bahasa pemrograman aras dasar seperti : bahasa mesin, dan bahasa *asembley.*
2. Bahasa pemrograman aras tengah seperti : Bahasa C
3. Bahasa pemrograman aras atas seperti : Pascal, Visual Basic, Mat Lab, dan lain-lain.

Bahasa pemrograman dapat menghasilkan file programm komputer yang dapat berdiri sendiri yang disebut *Compiler*, sedangkan bahasa pemrograman yang hanya bisa sebagai penerjemah kode-kode program disebut *Interpreter*.

BINA SARANA INFORMATIKA
UNIVERSITAS

2.1.3. Visual Basic 6.0

Menurut Aminudin (2016:2) ”Visual Basic 6.0 adalah bahasa pemrograman *event-driven* yang berasal dari BASIC”. *Event-driven* sendiri artinya program yang menunggu sampai adanya *respons* dari pemakai berupa kejadian tertentu, misalnya ketika tombol diklik atau menu dipilih. Ketika *event* terdeteksi, event yang berhubungan akan melakukan aksi dengan kode yang diberikan. Ada tiga edisi yang dikeluarkan microsoft, yaitu:

1. *Standard Edition*

   Standard Edition sangat direkomendasikan bagi pemula yang ingin mempelajari visual basic 6.0 versi ini mempunyai fasilitas sebagai berikut:

   a. Kemampuan aplikasi 32 bit yang berjalan di Microsoft Win 9x dan Win NT untuk pemula.
   b. Terdiri dari kontrol seperti *grid, tab, data bound*.
   c. Memuat *learn* Visul Basic *Now* dan *Online help.*
   d. Microsoft *Developer Network CD* berisi dokumentasi.

2. *Professional Edition*

   Professional Edition umumnya sudah cukup mendalami visual basic 6.0. Tidak banyak mengalami banyak perubahan dari *standard edition* hanya ada beberapa tambahan diantaranya:

   a. *ActiveX Control*, termasuk *internet control.*
   b. IIS(*Internet Information Server*).
   c. *Dynamic* HTML *Page Designer*

3. *Enterprise Edition*

   Lebih ditekankan untuk membuat aplikasi yang bersifat s*erver based*,tapi program aplikasi standar dapat berjalan dengan baik menggunakan versi ini, fasilitas yang ditambahkan adalah :

   a. *Application Performance Explorer.*
   b. IIS(*Internet Information Server*).
   c. *SQL Debugging.*
   d. *Visual Component Manager.*
   e. *Visual Database Toll.*

BINA SARANA INFORMATIKA UNIVERSITAS

2.1.4. Basis Data

Menurut Rosa & Shalahuddin (2016:43) ”Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”.

Adapun fungsi dari basis data itu sendiri adalah :

1. Mengelompokan data informasi sehingga lebih mudah dimengerti.
2. Mencegah terjadinya duplikasi data maupun inkonsistensi data.
3. Mempermudah proses penyimpanan, pengaksesan, pembaruan, dan menghapus data.
4. Menjaga kualitas data dan informasi yang diakses sesuai dengan data yang di*input*
5. Membantu proses penyimpanan data yang besar.

Pada intinya basis data adalah media untuk penyimpanan data agar dapat diakses dengan mudah dan cepat.

BINA SARANA INFORMATIKA UNIVERSITAS

2.1.5. XAMPP

Menurut Bayhaqi (2017:6) “XAMPP adalah perangkat lunak bebas(*free software)* yang mendukung untuk banyak sistem operasi, dan merupakan kompilasi dari beberapa program”. Fungsi XAMPP sendiri sebagai server yang berdiri sendiri(*locallhost*), yang terdiri dari beberapa program antara lain Apache HTTP Server, MySQL, PHP dan Perl.

2.1.6. PHP(*Hypertext Prepocessor*)

Menurut Bayhaqi (2017:7) ”PHP atau *hypertext preprocessor* adalah bahasa skrip pemrograman basis data yang dapat ditanamakan atau disiipkan ke dalam HTML”. PHP banyak digunakan untuk situs web yang dinamis PHP banyak digunakan untuk membangun sebuah CMS (*Content Management System*).”

Phpmyadmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk administrasi melalui *world wide web*. Phpmyadmin mendukung berbagai operasi MYSQL, diantaranya mengelola basis data, membuat tabel, membuat *fields*, membuat relasi, melihat indeks, dan lain-lain

BINA SARANA INFORMATIKA BSI UNIVERSITAS

2.1.7. *Crystal Report*

Menurut Aminudin (2016:158) “*Crystal Report* adalah form khusus berbentuk seperti format naskah yang ingin dicetak”. *Crystal Report* dapat berdiri sendiri dapat pula menjadi satu dengan *project visual*. Adapun cara mengkoneksikan *Crystal Report* dengan Visual Basic adalah sebagai berikut :

1. Pastikan *Crystal Report* sudah terinstall dan sudah membuat desain *report*nya, serta sudah menyimpan dalam satu folder dengan program visual basic yang telah dibuat.
2. Buka kembali program yang telah anda buat dan tambahkan tombol untuk cetak data.

3. Tambahkan objek *Crystal Report* di *tollbox* dengan cara pilih *project*-*Components* - pilih *tabControls* - pilih *Crystal Report Viewer Control* tekan OK.
4. Lalu masukan kode program untuk memasukan data di visual basic.

2.1.8. Model Pengembangan Perangkat Lunak

Metode yang digunakan pada pengembangan perangkat lunak ini menggunakan metode *waterfall* Rosa & Shalahuddin (2016:28) yang terbagi menjadi lima tahapan, yaitu:

1. Analisis kebutuhan perangkat lunak

   Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu didokumentasikan.
2. Desain *(Design)*

   Desain perangkat lunak merupakan proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Proses ini menstranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

BINA SARANA INFORMATIKA UNIVERSITAS

3. Pembuatan kode program

   Desain harus ditranslasikan kedalam programperangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dbuat pada tahap desain.

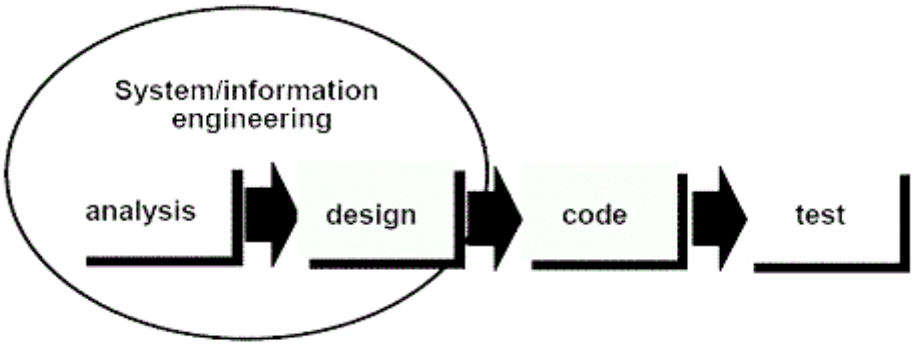
4. Pengujian *(Testing)*

   Pengujian fokus pada perangkat lunak secara fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan *(eror),* dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung *(support)* atau pemeliharaan *(maintenance).*

   Pada tahap ini juga tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan ini bisa terjadi karena kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

BINA SARANA INFORMATIKA UNIVERSITAS



Sumber : Rosa dan Shalahudin (2016:29)

Gambar II.1 Ilustrasi Model *Waterfall*

2.2. Peralatan Pendukung (*Tools Program)*

Didalam perancangan program penulis menggunakan beberapa peralatan pendukung diantaranya yaitu: *Entity Relantionship Diagram* (ERD), *Logical Record Structure* (LRS), pengkodean, *Hierarchy Input Process Output* (HIPO), *Flowchart*, implementasi dan pengujian unit. Berikut ini adalah landasan teori yang digunakan penulis sesuai pembahasan diatas.

2.2.1. *Entity Relationship Diagram* (ERD)

Menurut Rosa & Shalahuddin (2016:50) ”ERD *(Entity Relationship Diagram)* adalah bentuk paling awal dalam melakukan perancangan basis data relasional”.

Dalam penjelasannya ERD pertama kali dideskripsikan oleh Peter Chen yang dibuat sebagai bagian dari perangkat lunak. Notasi yang digunakan dalam ERD adalah:

1. Entitas / *Entity*

   Entitas / *Entity* merupakan data inti yang akan disimpan bakal tabel pada basis data benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer.

2. Relasi

   Relasi adalah yang menhubungkan antara entitas,biasanya diawali dengan kata kerja.

3. *Atribute*

   Atribute adalah *field* atau kolom data yang butuh disimpan dalam suatu entitas.

4. *Atribute* Kunci *Primer*

   Atribute Kunci *Primer* adalah *field* atau kolom data yang butuh disimpan dalam suatu entitas/*entity* dan digunakan sebagai kunci akses.

5. *Atribute* Multi Nilai / *Multivalue*

   Atribute Multi Nilai / *Multivalue* adalah *field* atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.

6. Asosiasi / *Association*

   Asosiasi / *Association* merupakan penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki *multiplicity* kemungkinan jumlah pemakaian.

BINA SARANA INFORMATIKA UNIVERSITAS

ERD biasanya memiliki tiga macam kardinalitas antara lain :

1. *One to one* yaitu*;* satu relasi menghubungkan dua buah entitas.
2. *One to many yaitu;* satu relasi menghubungkan tiga buah entitas.
3. *Many to many* yaitu; satu relasi menghubungkan banyak entitas.

2.2.2. *Logical Record Structure* (LRS)

Sebelum tabel dibentuk dari *field* atau *atribut* entitas secara fisik atau *level internal*, maka harus dibuatkan suatu bentuk relasional model yang dibuat secara *logic* atau *level external* dan konsep, dari pernyataan tersebut dibutuhkan yang disebut dengan *Logical Record Structure* (LRS).

Menurut Rosa & Shalahuddin (2016:58) *"Logical Record Structure* (LRS) merupakan konsep yang berkaitan dengan pandangan pemakai terhadap data yang disimpan dalam basis data".

Dalam pembuatan LRS terdapat tiga hal yang dapat mempengaruhi, yaitu:

1. Jika tingkat hubungan satu pada satu (*One to-one*), maka kunci salah satu entitas akan dijadikan kunci asing (*foreign key*) pada tabel yang lain dan kuci asing itu dijadikan kunci primer juga, dengan kata lain relasi tidak menjadi tabel sendiri.
2. Jika tingkat hubungan satu pada banyak (*one-to-many*), maka hubungan yang memiliki hubungan *one* akan dijadikan kunci primer di entitas yang memiliki hubungan *many*, dengan kata lain relasi tidak menjadi tabel sendiri.
3. Jika tingkat hubungan banyak pada banyak (*many-to-many*), maka hubungan menjadi sebuah tabel tersendiri dengan kunci primer adalah attribut yang menjadi kunci primer di kedua entitas yang direlasinya

2.2.3. Pengkodean *(Code)*

Menurut Shatu (2016:106) ”Kode memudahkan proses pengolahan data karena dengan kode data akan lebih mudah di identifikasi”.

Contoh kode untuk karyawan laki-laki diberi kode “L”, karyawan wanita dengan kode “W”.

Syarat kode yang baik memiliki beberapa faktor yang perlu dipertimbangkan yaitu :

1. Kode yang disusun perlu disesuaikan dengan metode proses data.
2. Setiap kode harus mewakili satu item sehingga tidak membingungkan.
3. Kode yang disusun harus memudahkan pemakai untuk mengingatnya.
4. Kode yang disusun harus *fleksibel*, dalam arti memungkinkan dilakukan perluasan tanpa perubahan menyeluruh.
5. Setiap kode harus menggunakan jumlah angka atau huruf yang sama.
6. Kode yang panjang perlu dipotong (*chunking*) agar memudahkan untuk diingat, misalnya : kode 60662582549 dapat dibuat 606-258-259.
7. Kode yang panjang perlu diberi kode yang merupakan *check* digit yaitu mengecek kebenaran kode.

Kode dapat dibuat dalam struktur kode yang berbeda. Setiap struktur mempunyai kelebihan dan kelemahan, oleh karena itu perlu struktur kode yang sesuai sehingga tujuan pemberian kode dapat tercapai.Berikut macam tipe kode yang dipakai :

1. Kode *Mnemonic*

   Merupakan singkatan dari karakteristik data. Misal pabrik sepatu, persediaan sepatu pria ukuran besar dapat dibuatkan kodenya SPB(Sepatu Pria Besar).

Bisa juga kode ini menggunakan kombinasi huruf dan angka, misal sepatu pria dengan ukuran 42 diberi kode SP42.

2. Kode Urut *(sequential Code)*

   Kode urut disebut juga dengan kode seri *(serial code)* merupakan kode urut antara satu kode dengan kode yang berikutnya.

   Contoh : 001 tas gunung

   002 jaket gunung

3. Kode Blok *(Block Code)*

   Kode blok mengklarifikasikan *item* kedalam kelompok blok tertentu yang mencerminkan satu klarifikasi tertentu atas dasar pemakaian maksimum yang diharapkan.

4. Kode Group *(Group Code)*

   Kode *group* membagi data ke dalam *group* tertentu.

5. Desimal kode

   Setiap kelompok data akan diberi nomor 0 sampai 9. Oleh karenanya pengelompokan data harus dilakukan maksimum dalam sepuluh kelompok.

BINA SARANA INFORMATIKA UNIVERSITAS

2.2.4 HIPO *(Hierarky Input Proses Output)*

Menurut Elistri, Wahyudi, & Supardi (2014:106) ”Diagram HIPO (Hierarkhi Input Proses Output) Merupakan serangkaian diagram yang terdiri dari serangkaian level yang mengalir dari atas kebawah yang menggambarkan sistem yang lebih detail. Diagram HIPO dirancang sebagai alat bantu dan alat dokumentasi yang digunakan untuk mengidentifikasi apa yang harus dilakukan untuk menyelesaikan suatu masalah”.

Adapun tingkatan yang terdapat pada diagram HIPO (*hirarky input proses output)* ada tiga tingkatan yaitu :

1. *visual table of content* (VTOC)

   Terdiri dari satu diagram *hirarky* atau lebih. Diagram ini menggambarkan hubungan dari fungsi-fungsi secara berjenjang. *Visual tabel of contents* menggambarkan seluruh program HIPO baik rinci maupun ringkasan yang terstruktur.

2. Diagram ringkasan/ *Overview diagram*

   Yaitu suatu seri diagram fungsional, yang masing-masing diagram dihubungkan dengan satu fungsi sistem

3. Diagram rinci/ *Detail diagram*

   Yaitu suatu seri diagram fungsional dan masing-masing diagram dihubungkan dengan sebuah sub fungsi sistem.

2.2.5. Diagram Alir Data *(Flowchart)*

Menurut Drs. Lamhot Sitorus (2015:14) *"Flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dengan simbol-simbol tertentu". Diagram alir ini akan menunjukan program secara logika, diagram alir ini selain dibutuhkan untuk alat komunikasi juga diperlukan sebagai dokumentasi.

Adapun aturan-aturan dalam perancangan diagram alir tersebut yaitu :

1. Diagram alir digambarkan dengan orientasi dari atas kebawah, dan dari kiri ke kanan.
2. Setiap kegiatan / proses dalam diagram alir harus dinyatakan secara *eksplisit*.
3. Setiap diagram alir harus dimulai dari satu *start state* dan berakhir pada satu atau lebih terminal akhir/*terminator/halt state*.

4. Gunakan *connector* dan *off-page connector state* dengan label yang sama untuk menunjukan hubungan antar *path* algoritma yang terputus atau terpotong, misalnya sebagai akibat pindah atau ganti halaman.

1. Bentuk *Flowchart*

Ada dua bentuk bagan alir yang akan dibahas di bentuk *flowchart* ini, yaitu bagan alir sebagai berikut :

a. *System Flowchart*

Adalah suatu Flowchart yang digunakan untuk menyelsaikan permasalahan pada sistem pengolahan data, dimana melibatkan beberapaa perangkat keras, seperti *magnetic disk,card* dll.

b. ***Program Flowchart***

Adalah suatu flowchart yang digunakan untuk menyelesaikan permasalahan dalam pembuatan program.

BINA SARANA INFORMATIKA
UNIVERSITAS

2. Tehnik Pembuatan

Ada dua tehnik cara pembuatan bagan alir yang akan dibahas di bentuk *flowchart* ini, yaitu sebagai berikut :

a. *General Way*

Tehnik pembuatan *Flowchart* dengan cara ini lazim digunakan dalam menyusun logika suatu program, yang menggunakan proses secara tidak langsung (*non-direct-loop*).

b. *Iteration Way*

Tehnik pembuatan *Flowchart* dengan cara ini biasanya dipakai untuk logika program yang cepat dan bentuk permasalahan yang *kompleks*.

2.2.6 Implementasi Dan Pengujian Unit

Menurut Rosa & Shalahuddin (2016:27) ”Implelentasi termasuk pada persiapan perangkat lunak pada lingkungan produksi (lingkungan pada *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase intergrasi dan pengujian, Adapun pengujian unit fokus pada usaha verifikasi pada unit yang terkecil pada desain perangkat lunak (komponen atau modul perangkat lunak)”.

Pengujian untuk validasi memiliki beberapa pendekatan sebagai berikut :

1. *Black Box Testing* (pengujian kotak hitam)

   Yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

BINA SARANA INFORMATIKA UNIVERSITAS