

BAB II

LANDASAN TEORI

2.1. Pelayanan Jasa *Laundry*

Menurut (R. S. Hidayat & Winarno, 2014) mengemukakan bahwa “pelayanan jasa *laundry* merupakan sebuah bisnis yang berkaitan dengan jasa pencucian pakaian dengan mesin cuci maupun mesin pengering otomatis dan cairan pembersih serta pewangi khusus”.

2.2. Konsep Dasar Program

Pada BAB ini penulis membahas mengenai definisi serta uraian yang berkaitan dengan perancangan program pada Tugas Akhir penulis.

2.2.1. Program

Menurut (Kadir, 2014:8) mengatakan bahwa “program adalah kumpulan instruksi yang ditujukan untuk mengatur suatu tugas tertentu”. (Sukmaindrayana & Sidik, 2017) mengemukakan bahwa program adalah kumpulan intruksi yang digunakan untuk mengatur komputer agar melakukan suatu tindakan tertentu.

2.2.2. Bahasa Pemrograman

Menurut (Sahyar, 2016) “Bahasa pemrograman adalah perangkat lunak yang digunakan untuk menerjemahkan atau menuliskan algoritma dalam bentuk teks perintah-perintah yang dapat dimengerti oleh komputer untuk menyelesaikan masalah”.

1. Java

Menurut (Candenhead, 2016:36) Java adalah bahasa berorientasi objek, *platform*-netral, aman yang dirancang untuk lebih mudah dipelajari daripada C ++ dan lebih sulit untuk disalahgunakan daripada C dan C ++.

Menurut (Candenhead, 2016:38) mengemukakan bahwa pemrograman berorientasi objek atau *Object Oriented Programming* (OOP) adalah metodologi pengembangan perangkat lunak di mana sebuah program dikonseptualisasikan sebagai sekelompok objek yang bekerja bersama.

Objek dibuat dari templat yang disebut kelas, dan semua berisi data dan pernyataan yang diperlukan untuk menggunakan data itu.

2. Netbeans IDE

Menurut (Nofriadi, 2015) mengatakan bahwa “Netbeans merupakan sebuah aplikasi *Intergrated Development Environment* (IDE) yang berbasis Java dari Sun Microsystems yang berjalan diatas *swing* dan banyak digunakan sekarang sebagai *editor* untuk berbagai bahasa pemrograman”.

2.2.3. Basis Data

Basis data adalah sebuah sistem yang berfungsi untuk menyimpan dan mengolah sekumpulan data. (jubile enterprise, 2015:8)

1. MySQL

Menurut (Sari & Saputra, 2014) menyimpulkan bahwa “MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya”.

2. XAMPP

Menurut (Sari & Saputra, 2014) memaparkan bahwa “XAMPP adalah singkatan dari X (Program ini dapat dijalankan di banyak sistem operasi, seperti Windows, Linux, Mac OS, dan Solaris), Apache, MySQL, PHP, Perl. XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dalam paketnya sudah terdapat Apache (*web server*), MySQL (*database*), PHP (*server side scripting*), Perl, FTP *server*, PhpMyAdmin dan berbagai pustaka bantu lainnya”.

2.2.4. Model Pengembangan Perangkat Lunak

Menurut Pressman dalam (Sasmito, 2017) mengemukakan bahwa "Metode *Waterfall* merupakan model pengembangan sistem informasi yang sistematis dan sekuensial”.

Menurut Sommerville dalam (Sasmito, 2017) metode *Waterfall* memiliki tahapan-tahapan sebagai berikut :

1. *Requirements analysis and definition*

Layanan sistem, kendala, dan tujuan ditetapkan oleh hasil konsultasi dengan pengguna yang kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem.

2. *System and software design*

Tahapan Perancangan Sistem Mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat lunak dan hubungannya.

3. *Implementation and unit testing*

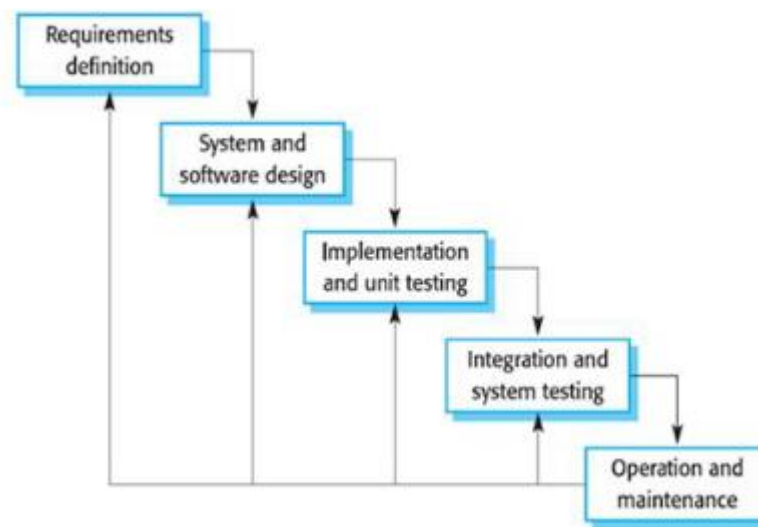
Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

4. *Integration and system testing*

Unit-unit individu program atau program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak.

5. *Operation and maintenance*

Biasanya (walaupun tidak selalu), tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. *Maintenance* melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.



Sumber : Sasmito (2017)

Gambar II.1
Metode *Waterfall*

2.3. *Tools Pendukung*

Terdapat beberapa peralatan pendukung yang digunakan pada setiap rancangan yang disusun dalam Tugas Akhir penulis.

2.3.1. *Entity Relationship Diagram (ERD)*

Menurut (Ratniasih, 2015) memaparkan bahwa “*Entity Relationship Diagram* adalah suatu alat untuk mempresentasikan model data yang ada pada sistem dimana didalamnya terdapat *entity* dan *relationship* yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta dari dunia nyata yang digambarkan dengan lebih sistematis dengan menggunakan diagram”.

Sedangkan menurut (Bouchrika, 2014:27) mengemukakan bahwa “*Entity Relationship Diagram (ERD)* adalah bahasa visual yang digunakan untuk menyediakan representasi grafis untuk struktur logis dari basis data”.

Menurut (Ratniasih, 2015) terdapat komponen-komponen yang terdapat di dalam *Entity Relationship Model* sebagai berikut :

1. Entitas

Entitas merupakan data inti yang akan disimpan bakal tabel pada basis data biasanya mengacu pada benda yang terlibat dalam aplikasi yang akan dibuat.

2. Atribut

Field atau kolom data yang butuh disimpan dalam suatu entitas.

3. Atribut kunci primer (*Primary Key*)

Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses *record* yang diinginkan biasanya berupa id atau kode yang tidak boleh sama atau jika lebih dari satu maka kombinasinya tidak boleh sama, kunci pemanggil benda seperti halnya nama pada manusia.

4. Atribut multi nilai (*Multivalue*)

Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.

5. Relasi

Relasi yang menghubungkan antar entitas biasanya diawali dengan kata-kerja; biasanya juga mendefinisikan relasi proses antar entitas atau benda yang terkait dengan aplikasi.

6. Asosiasi

Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki *multiplicity* kemungkinan jumlah pemakaian.

2.3.2. Logical Record Structure (LRS)

LRS (*Logical Record Structure*) merupakan representasi dari struktur *record-record* pada tabel-tabel yang terbentuk dari hasil antar himpunan entitas. (Sukmaindrayana & Sidik, 2017)

2.3.3. Pengkodean

Menurut Supranto dalam (Sukmaindrayana & Sidik, 2017) mengemukakan bahwa “Pengkodean (*coding*) adalah suatu kegiatan pemberian kode atau simbol pada keterangan-keterangan tertentu, kalau pengolahan akan dilakukan dengan komputer elektronik”.

Jogrianto dalam (Azis & Sarmidi, 2018) juga mengemukakan “Kode digunakan untuk tujuan mengklasifikasikan data, memasukan data ke dalam komputer dan mengambil bermacam-macam informasi yang berhubungan dengannya. Kode dapat dibentuk dari kumpulan angka, huruf, dan karakter-karakter khusus (misalnya %, /, &, \$, dan lain sebagainya)”. Dalam pengkodean (Aziz & Sarmidi, 2018) mengatakan bahwa :

- a. Angka merupakan simbol yang banyak digunakan pada sistem kode akan tetapi kode yang berbentuk angka lebih dari 6 digit akan sangat sulit untuk di ingat kode numerik (*numeric code*) menggunakan 10 macam kombinasi angka di dalam kode.
- b. Kode alfabetik (*alphabetic code*) menggunakan 26 kombinasi huruf untuk kodenya.
- c. Kode alphanumerik (*alphanumeric code*) merupakan kode yang menggunakan gabungan angka, huruf, dan karakter-karakter khusus meskipun kode numerik, alfabetik dan alphanumerik merupakan kode yang paling banyak digunakan di dalam sistem informasi, tetapi kode yang lain juga mulai banyak digunakan, seperti misalnya kode batang (*bar code*).

2.3.4. *Hierarchy Input Process Output (HIPO)*

Menurut (Trisyanto, 2017:82) HIPO (*Hierarchy Plus Input-Process-Output*) merupakan metodologi yang dikembangkan dan didukung oleh IBM. HIPO sebenarnya adalah alat dokumentasi program, akan tetapi sekarang HIPO juga banyak digunakan sebagai alat desain dan teknik dokumentasi dalam siklus pengembangan sistem.

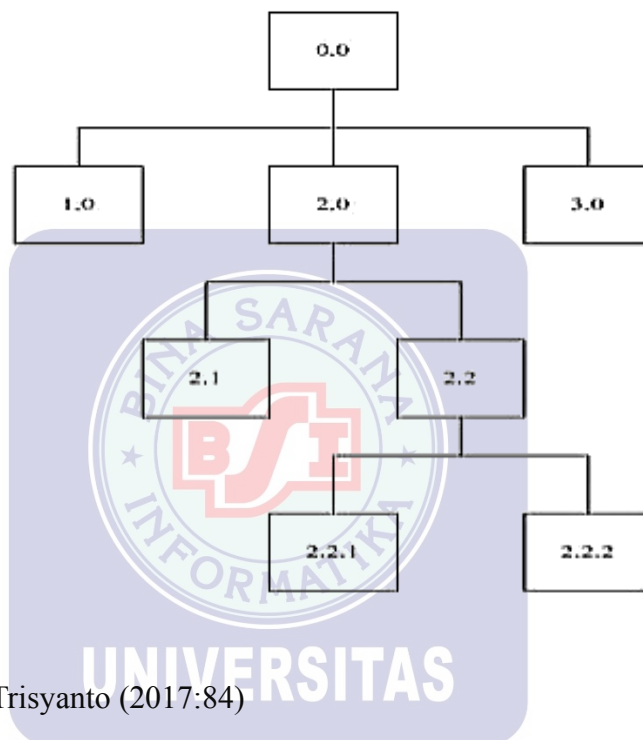
HIPO berbasis pada fungsi, yaitu tiap-tiap modul di dalam sistem digambarkan oleh fungsi utamanya. Fungsi-fungsi dari sistem digambarkan oleh

HIPO dalam tiga tingkatan. Untuk masing-masing tingkatan digambarkan dalam bentuk diagram tersendiri.

Dengan demikian HIPO menggunakan tiga macam diagram untuk masing-masing tingkatannya, yaitu sebagai berikut ini :

a. *Visual Table Of Contents (VTOC)*

Diagram ini menggambarkan hubungan dari fungsi-fungsi di sistem secara berjenjang.



Sumber: Trisyanto (2017:84)

Gambar II.2

Ilustrasi *Visual Table Of Contents*

b. *Overview Diagram*

Overview Diagram menunjukkan secara garis besar hubungan dari *input*, proses dan *output*. Bagian input menunjukkan *item-item* data yang akan digunakan oleh bagian proses, bagian proses berisi sejumlah langkah-langkah yang menggambarkan kerja dari fungsi, bagian *output* berisi dengan *item-item* data yang dihasilkan atau dimodifikasi oleh langkah-langkah proses.

c. *Detail Diagrams*

Detail diagrams merupakan diagram tingkatan yang paling rendah di diagram HIPO. Diagram ini berisi dengan elemen-elemen dasar dari paket yang menggambarkan secara rinci kerja dari fungsi.

(Azis & Sarmidi, 2018) mengatakan HIPO sebenarnya adalah alat dokumentasi program, akan tetapi sekarang HIPO juga banyak digunakan sebagai alat desain dan teknik dokumentasi dalam siklus pengembangan sistem HIPO berbasis pada fungsi, yaitu tiap-tiap modul didalam sistem digambarkan oleh fungsi utamanya.

2.3.5. Diagram Alir Program (*Flowchart*)

Flowchart adalah representasi grafik dari sistem informasi, proses-proses, aliran-aliran data logis, masukan-masukan, keluaran-keluaran, dan *file-file* juga entitas-entitas sistem operasi yang berhubungan dengan sistem informasi tersebut. Sistem *Flowchart* berisi kegiatan-kegiatan logis dan fisik dari siapa, apa, bagaimana, dan dimana proses informasi dan operasi proses terjadi. (Sari & Saputra, 2014)

2.3.6. *Black Box Testing*

Menurut (T. Hidayat & Muttaqin, 2018) *Black Box Testing* merupakan “pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program”.

Pengujian dilakukan untuk mengetahui fungsi-fungsi yang dibuat, masukan dan keluaran melalui kasus uji coba yang dibuat apakah sesuai dengan harapan atau tidak.