

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Tinjauan Pustaka**

Tinjauan pustaka berisi tentang semua teori-teori yang berhubungan yang akan dibahas. Berikut adalah teori-teori yang penulis kumpulkan :

##### **2.1.1. Konsep Dasar Sistem Informasi**

Sistem Informasi adalah suatu sistem atau proses dimana data inputan diolah menjadi output yang diinginkan.

Sistem adalah kumpulan elemen yang saling berinteraksi satu sama lain yang membentuk satu kesatuan dalam usaha mencapai tujuan tertentu. Informasi adalah sebuah pernyataan yang menjelaskan suatu peristiwa suatu obyek sehingga manusia dapat membedakan sesuatu dengan yang lainnya. Informasi juga merupakan kumpulan data yang telah diolah yang memiliki arti [4].

Ketika mendefinisikan sistem, terdapat dua kelompok pendekatan, yang pertama yaitu menekankan pada prosedur dan menekankan pada komponen atau elemen. Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan bahwa sistem sebagai sebuah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran yang telah ditargetkan. Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam sistem. Selanjutnya, pendekatan yang lebih menekankan pada elemen atau komponen mendefinisikan bahwa sistem sebagai kumpulan dari elemen yang berinteraksi untuk mencapai suatu tujuan. [5]

## 1. Karakteristik Sistem

Karakteristik yang dimaksud adalah sebagai berikut :

### a. Komponen (*Components*)

Komponen sistem atau elemen sistem adalah semua hal yang menjadi bagian penyusun sistem, dapat berupa benda nyata ataupun abstrak..

### b. Batasan Sistem (*Boundary*)

Batasan sistem dibutuhkan untuk melainkan satu sistem dengan sistem yang lain agar tidak menyusahkan saat memberikan batasan scope tinjauan terhadap sistem.

### c. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem adalah hal-hal yang ada diluar sistem, dapat merugikan ataupun menguntungkan.

### d. Penghubung Sistem (*Interface*)

*Interface* adalah hal-hal yang menjadi penyambung antar bagian sistem. *Interface* menjadi media setiap komponen untuk saling berinteraksi dan berkomunikasi.

### e. Masukan Sistem (*Input*)

Masukan adalah bagian sistem yang akan diolah lebih lanjut untuk menghasilkan keluaran yang berguna.

### f. Pengolahan (*Processing*)

Pengolahan adalah bagian sistem yang sanget signifikan dalam mengolah masukan agar menghasilkan keluaran yang berguna.

### g. Keluaran (*Output*)

Keluaran adalah bagian sistem yang merupakan hasil dari bagian pengolahan

h. Sasaran (*Objective*) dan tujuan (*Goal*)

Supaya meraih sasaran dan tujuan sistem, setiap bagian dalam sistem harus dijaga.

i. Kendali (*Control*)

Agar bekerja sesuai dengan fungsi masing-masing, setiap komponen harus diperhatikan dan terkendali.

j. Umpan Balik (*Feedback*)

Umpan balik diperlukan oleh kontrol untuk menyadari adanya penyimpangan agar dapat dikembalikan pada kondisi normal.

2. Klasifikasi Sistem

Sistem informasi merupakan kumpulan antara bagian satu dengan lainnya yang bertujuan menghasilkan suatu informasi. Semua sistem mempunyai sasaran berbeda-beda dalam setiap permasalahan yang terjadi dari tiap sistem tersebut. Pada kasus tersebut, sistem mendapatkan klarifikasi dari berbagai sudut pandang. Ada 8 klarifikasi dalam sistem informasi ini, yaitu; sistem abstrak, sistem fisik, sistem alamiah, sistem buatan manusia, sistem deterministik, sistem probabilistik, sistem terbuka, dan sistem tertutup:

a. Sistem abstrak (*abstract system*)

Sistem ini berisi tentang gagasan atau konsep yang tidak tampak secara fisik. Misalnya sistem teologia. Maksudnya, pada sistem tersebut terjadi pemikiran hubungan antara manusia dengan Tuhan. Hal inilah terjadi sistem yang tidak tampak oleh manusia, akan tetapi hal itu bisa dirasakannya.

b. Sistem fisik (*physical system*)

Sistem ini adalah sistem yang ada dan nampak secara fisik. Contohnya sistem komputer, sistem transportasi, dan sistem sekolah. Pada sistem komputer,

terdapat elemen berupa peralatan yang berfungsi secara bersamaan sebagai alat dalam menjalankan pengolahan data. Begitu juga dengan sistem lainnya yang memiliki elemen-elemen tersendiri.

c. Sistem Alami (*natural system*)

Adalah sistem yang ada karena proses alam. Dalam sistem ini, tidak dibuat oleh manusia. Bagaikan sistem perputaran bumi, sistem tata surya, dan sistem terjadinya siang dan malam. Sistem tersebut terbuat tanpa ada campur tangan dari manusia.

d. Sistem buatan manusia (*human made system*)

Sistem ini dibuat dari hasil karya manusia yang melibatkan interaksi manusia dengan mesin, yang biasa disebut human-machine system atau ada yang menyebutnya machine system. Seperti contoh, sistem komputer. Dari sistem tersebut, mesin bergantung pada manusia yang mengoperasikannya..

e. Sistem deterministik (*deterministic system*)

Sistem ini adalah suatu sistem yang bergerak atau beroperasi dengan cara yang dapat diperkirakan secara tepat, dan dapat mengetahui interaksi yang terjadi pada setiap bagian-bagiannya. Misal pada sistem komputer dan *operation system* android. Yang dimana sistemnya akan bergerak dengan perkiraan sebuah kode-kode sistem yang berfungsi untuk menjalankan sistem satu dengan yang lain.

f. Sistem probabilistik (*probobalistic system*)

Adalah suatu sistem yang tidak dapat diperkirakan hasil akhirnya atau kondisi masa depannya secara tepat, karena memiliki unsur probabilitas-kemungkinan atau tidak menentu. Contohnya, pada sistem persediaan barang,

sistem pemilihan presiden, dsb. Sistem-sistem tersebut adalah sistem probabilistik karena tidak bisa diprediksi dalam hasilnya.

g. Sistem Terbuka (*open system*)

Sistem ini membolehkan terjadinya pertukaran materi dan energi antara sistem dengan lingkungan. Misalnya, tumbuhan dan hewan. Tumbuhan biasanya menyerap air dan karbon dioksida yang berasal dari lingkungan. Tumbuhan juga memerlukan kalor yang disorotkan oleh matahari. Dengan begitu, setiap sistem ini akan dipengaruhi oleh lingkungan.

h. Sistem Tertutup (*closed system*)

Berbeda dengan sistem terbuka, sistem tertutup merupakan kebalikannya. Yaitu adalah sistem yang tidak membolehkan terjadinya pertukaran materi antara sistem dan lingkungannya, dalam hal ini akan terjadi isolasi. Misal termos air panas. Dinding di dalam termos tersebut terbuat dari bahan isolator, yaitu bahan yang tidak menghantarkan panas. Dalam kesehariannya, memang banyak sistem terisolasi buatan yang tidak ideal. Minimal ada energi yang keluar, walaupun jumlahnya sangat kecil.

### 2.1.2. Definisi Upah dan Gaji

Menurut PP. No 8 tahun 1981 tentang Perlindungan Upah, upah adalah : *“suatu penerimaan sebagai imbalan dari pengusaha kepada tenaga kerja untuk suatu pekerjaan atau jasa yang telah atau akan dilakukan, dinyatakan atau dinilai dalam bentuk uang yang ditetapkan menurut suatu persetujuan atau peraturan perundang-undangan dan dibayarkan atas dasar suatu perjanjian kerja antara pengusaha dan pekerja termasuk tunjangan baik untuk pekerja sendiri maupun keluarganya”*.

Berbeda dari gaji yang bersifat tetap setiap bulannya dan akan terus diterima selama periode kerja, nilai dari upah umumnya telah disepakati di awal. Banyak

perusahaan swasta yang memberlakukan sistem perjanjian kerja waktu tertentu (PKWT), dan karyawan kontrak inilah yang mendapat imbalan upah, bukan gaji. Hal inilah yang membedakan antara gaji dan upah. [6]

### **2.1.3. Definisi Slip Gaji**

Slip gaji merupakan tanda bukti pemberian gaji dari perusahaan kepada seorang karyawan. Banyak tipe slip gaji tergantung pada isi dari gaji tersebut. Semakin tinggi jabatan seseorang maka semakin banyak intensif yang didapat, maka akan semakin banyak pula isi dari slip gaji tersebut. Slip gaji karyawan diperlukan oleh kedua belah pihak karena penggunaannya yang penting. Untuk karyawan, slip gaji berguna untuk digunakan sebagai acuan gaji yang akan diterima beserta insentif dan beberapa potongan terkait gaji seperti pajak. Untuk perusahaan, slip gaji berguna sebagai kontrol keuangan perusahaan agar perusahaan dapat mengetahui pengeluaran perusahaan selama 1 bulan-nya. [7]

### **2.1.4. Manfaat Slip Gaji**

Bagi Perusahaan :

- Slip gaji berfungsi sebagai bukti resmi bahwa perusahaan telah melakukan kewajibannya terhadap karyawan yang bersangkutan berkaitan dengan sejumlah nominal tertentu yang menjadi hak karyawan.
- Slip gaji berfungsi sebagai dokumentasi atas pencatatan pengeluaran gaji atau upah atas pengeluaran dana perusahaan.

Bagi Karyawan :

- Slip gaji berfungsi sebagai bukti kejelasan komponen proses penggajian bagi pegawai.
- Bukti sah untuk informasi potongan pajak dan penyetoran pajak penghasilan.

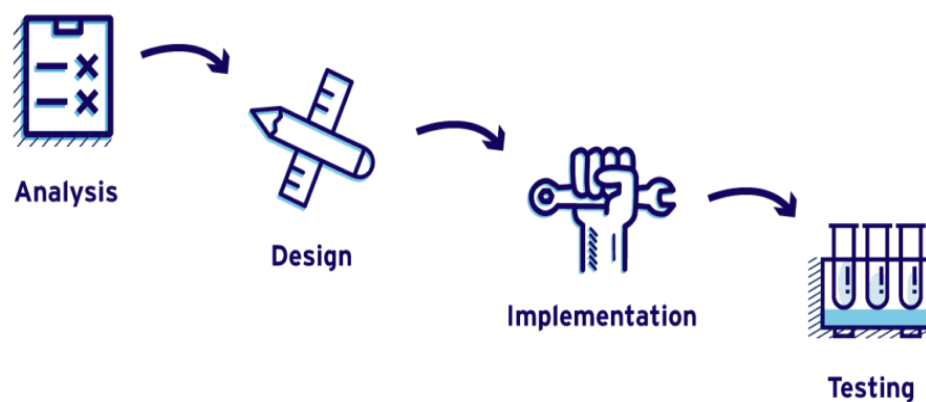
### 2.1.5. Definisi Cuti

Cuti merupakan keadaan tidak masuk kerja yang diizinkan dalam jangka waktu tertentu [8]

Cuti adalah hak Pegawai Negeri Sipil berupa izin tidak masuk kerja yang dapat di tunda dalam jangka waktu tertentu. Dalam pelaksanaanya cuti hanya dapat di tunda dalam jangka waktu tertentu dan apabila kepentingan dinas mendesak.

### 2.1.6. Model Waterfall

Metode Waterfall adalah suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan dipandang sebagai terus mengalir ke bawah (seperti *waterfall*) melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. [3]



Gambar II.1 Model Waterfall

#### 1. Analisa (*Analysis*)

Langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data pada tahap ini bisa dilakukan dalam sebuah penelitian, wawancara atau *study literature*. Seorang sistem analis dapat mencari informasi sebanyak mungkin dari user sehingga akan terbuat sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh user tersebut..

#### 2. Desain (*Design*)

Bagian ini akan membuat dokumen kebutuhan user atau bisa dikatakan sebagai data yang berhubungan dengan apa yang user inginkan dalam pembuatan sistem. Dokumen tersebut akan menjadi sebuah acuan sistem analis untuk mengubahnya ke dalam bahasa pemrogram

### 3. Implementasi (*Implementation*)

Bagian ini adalah penerjemahan desain menjadi bahasa yang bisa dikenali oleh computer. Dilakukan oleh para programmer yang akan menerjemahkan transaksi yang diminta oleh user. Tahapan ini lah yang merupakan tahapan paling nyata dalam mengerjakan suatu sistem. Maksudnya penggunaan komputer akan dimaksimalkan pada tahapan ini.

### 4. Pengujian (*Testing*)

Pada tahap ini fokus dibagian perangkat lunak dari segi logika dan fungsional dan meyakinkan bahwa semua bagian sudah diuji. Untuk meminimalisir kesalahan (*error*) dan memastikan hasil yang dihasilkan sesuai dengan yang diinginkan.

Dari kenyataan yang terjadi sangat jarang model *waterfall* dapat dilakukan sesuai alurnya dikarenakan halberikut:

- a. Perubahan spesifikasi perangkat lunak terjadi ditengah alur pengembangan.
- b. Sangat sulit bagi pengguna untuk mendefinisikan semua spesifikasi di awal alur pengembangan. Pengguna sering kali butuh contoh (*prototype*) untuk menjabarkan spesifikasi kebututhan sistem lebih lanjut.
- c. Client tidak mungkin bersabar mengakomodasi perubahan yang diperlukan di akhir alur pengembangan.



Walaupun banyak kelemahan yang dimiliki model *waterfall* tetapi model ini telah menjadi dasar dari model-model yang lain dalam melakukan perbaikan yang diperlukan di akhir alur pengembangan perangkat lunak.

Model *waterfall* digunakan saat kebutuhan pelanggan sudah sangat dipahami dan kemungkinan terjadinya perubahan pada pengembangan perangkat lunak kecil. Hal bagus dari model *waterfall* adalah struktur tahap pengembangan sistem jelas, dokumentasi dihasilkan disetiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap).

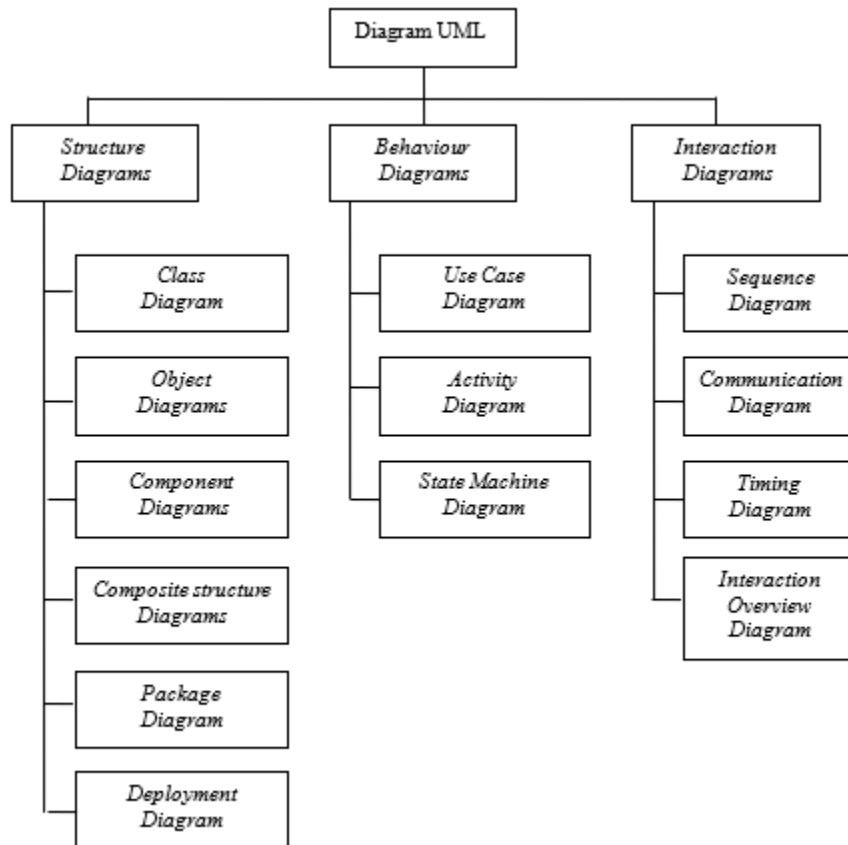
#### **2.1.7. Aplikasi Web**

Pada sebuah rekayasa perangkat lunak, suatu aplikasi web (bahasa Inggris: *web application* atau sering disebut *webapp*) adalah suatu aplikasi yang diakses menggunakan penjelajah web melalui suatu jaringan seperti Internet atau intranet. Aplikasi Web adalah suatu aplikasi *software* komputer yang dikode didalam bahasa yang didukung penjelajah web (seperti HTML, JavaScript, AJAX, Java, dll) dan tergantung pada penjelajah tersebut untuk menampilkan suatu aplikasi.

Aplikasi web telah menjadi populer dikarenakan kemudahan ketersediaanya aplikasi klien untuk mengaksesnya, penjelajah web, yang kadang disebut sebagai suatu *thin client* (klien tipis). Kemahiran dalam memperbaiki dan memelihara aplikasi web tanpa perlu mendistribusikan dan menginstal perangkat lunak pada kemungkinan ribuan komputer klien merupakan alasan pada popularitasnya. Aplikasi web yang sering digunakan contohnya webmail, toko ritel daring, lelang daring, wiki, papan diskusi, weblog, dsb.

### 2.1.8. Unified Modelling Language (UML)

UML (*Unified Modeling Language*) merupakan bahasa standar yang telah banyak digunakan pada dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, dan juga menunjukkan arsitektur dalam pemrograman berorientasi objek. [9]



Gambar II.2 - Diagram UML

Beberapa macam diagram yang digunakan adalah sebagai berikut :

#### 1. Diagram kelas (Class Diagram)

Diagram kelas atau *class diagram* adalah struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas mempunyai apa yang disebut atribut dan metode atau operasi.

#### 2. Diagram use-case (*Usecase Diagram*)

*Use case* diagram adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendefinisikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang nanti akan dibuat.

### 3. Diagram sekuen (*Sequence Diagram*)

Diagram sekuen merupakan tingkah suatu objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

### 4. Diagram aktifitas(*Activity Diagram*)

*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari suatu sistem atau proses bisnis atau menu yang ada terdapat pada perangkat lunak.

### 5. Diagram komponen(*Component Diagram*)

*Component diagram* terbuat untuk menunjukkan organisasi dan kebutuhan diantara kumpulan-kumpulan komponen dalam sebuah sistem.

### 6. Diagram deployment(*Deployment Diagram*)

*Deployment diagram* menggambarkan konfigurasi komponen pada proses eksekusi aplikasi.

#### 2.1.9. *Entity Relationship Diagram (ERD)*

ERD (*Entity Relationship Diagram*) adalah diagram yang menunjukkan hubungan antara entity satu dengan entity lainnya. Dalam hubungan tersebut terdapat beberapa jenis hubungan, yaitu :

1. *One to One* (Satu ke satu)

*One to One* yaitu setiap *entity* hanya boleh memiliki hubungan dengan satu *entity* yang lain.

## 2. *One to Many* (Satu ke Banyak)

*One to Many* yaitu hubungan antara satu entitas dengan beberapa entitas, dan begitu pula pada kebalikannya.

## 3. *Many to Many*

*Many to Many* yaitu hubungan antara beberapa *entity* yang memiliki lebih dari satu relasi dan membentuk tabel ketiga.

### 2.1.10. Struktur Navigasi

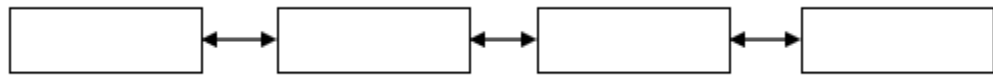
Struktur navigasi adalah suatu alur pada suatu program dan termasuk bagian terpenting dalam pembuatan aplikasi halaman *web*. Struktur navigasi ini merupakan perancangan hubungan dan rantai kerja dari beberapa lokasi yang berbeda dan dapat membantu mengorganisasikan seluruh elemen halaman web dan juga memudahkan untuk menganalisa keinteraktifan seluruh objek didalam halaman *web*.

Struktur navigasi digunakan untuk menunjukan secara garis besar isi dari seluruh *website* dan menggambarkan bagaimana hubungan antara isi dari *website* tersebut. Dengan struktur navigasi, dapat terlihat semua susunan isi dari sebuah *website* secara menyeluruh.

Struktur navigasi memiliki beberapa jenis. Yaitu :

#### 1. Navigasi Linier

Struktur ini menampilkan satu per satu tampilan layar dengan beruntun dan tidak adanya percabangan. Biasanya digunakan untuk presentasi karena kemudahan dalam hal penampilan data serta tidak membutuhkan banyak keinteraktifan di dalamnya. Tampilan yang ditunjukan adalah satu halaman sebelumnya atau satu halaman sesudahnya.

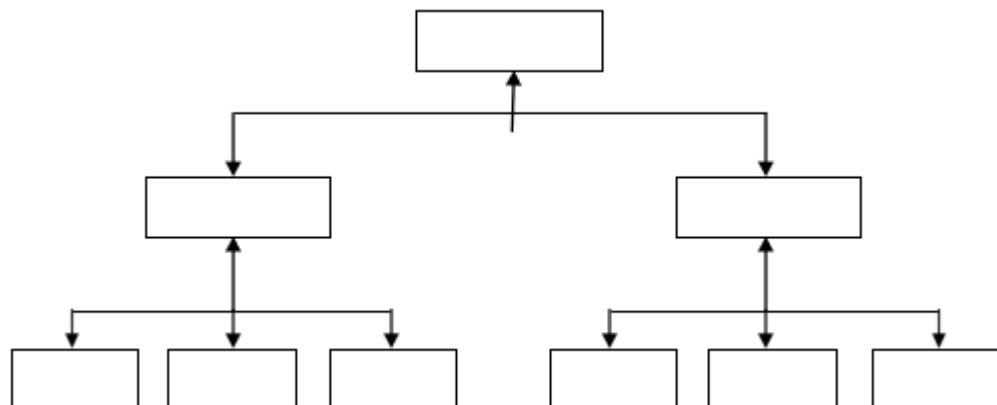


Gambar II.3 - Navigasi Linier

## 2. Navigasi Hirarki

Struktur ini biasa disebut juga struktur navigasi bercabang dan merupakan suatu struktur yang mengandalkan percabangan untuk menampilkan data atau gambar pada layar dengan kriteria tertentu.

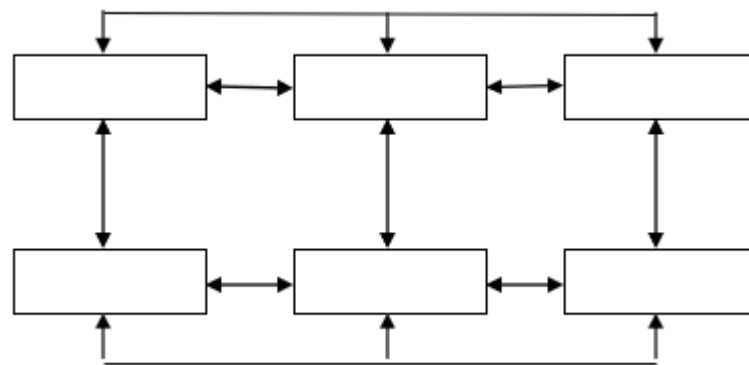
Tampilan pada menu pertama disebut *master page*. Halaman tersebut memiliki halaman bercabang yang biasa disebut dengan *slave page* (pendukung) dan jika dipilih menjadi halaman utama kedua.



Gambar II.4 - Navigasi Hirarki

## 3. Navigasi Non Linier

Struktur navigasi non linier (tidak berurutan) merupakan pengembangan dari struktur navigasi linier, tetapi navigasi ini memperkenankan adanya percabangan. Pada struktur ini kedudukan semua *page* sama, sehingga tidak dikenal dengan adanya master atau *slave page*.

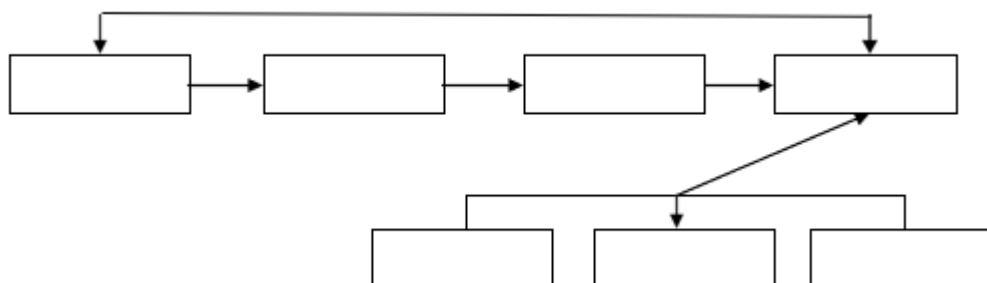


Gambar II.5 - Navigasi Non-Linier

#### 4. Navigasi Campuran

Adalah gabungan pada struktur sebelumnya serta disebut juga dengan struktur navigasi bebas. Artinya adalah jika suatu tampilan membutuhkan percabangan maka dibuat percabangan. Struktur ini sangat banyak digunakan dalam pembuatan aplikasi multimedia.

Pemakai dapat dengan bebas menelusuri program. Namun pada bagian tertentu gerakan dibatasi secara hirarki ataupun linier.



Gambar II.6 - Navigasi Campuran

#### 2.1.11. Pengujian *Black-Box* (*Black-Box Testing*)

Pengetesan perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program untuk mengetahui apakah fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan [10].

Black box adalah pengetesan yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Bisa digambarkan

seperti kita melihat suatu kotak hitam, kita hanya bisa melihat penampilan luarnya saja, tanpa tahu ada apa dibalik bungkus hitamnya. Sama halnya dengan pengujian black box, mengevaluasi hanya dari tampilan luarnya (interface nya), fungsionalitas nya tanpa mengetahui apa sesungguhnya yang terjadi dalam proses detail nya (hanya mengetahui input dan output).

Pengujian black box adalah pendekatan tambahan dari teknik white box, karena pengujian black box diharapkan dapat mendapatkan kelas kesalahan yang lebih luas dibandingkan teknik white box. Pengetesan black box berkonsentrasi pada pemeriksaan persyaratan fungsional pada perangkat lunak, untuk mendapatkan serangkaian kondisi input yang sesuai dengan persyaratan fungsional suatu program. Oleh sebab itu, pengetesan black box membolehkan pembuatan perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program.

#### **2.1.12. Web Server**

Web server menunggu *request* dari *client* yang menggunakan *browser* seperti Google Chrome, Mozilla Firefox, dll. Apabila terdapat sebuah *request* dari browser, maka *web server* akan mengolah atau memproses permintaan tersebut lalu mengirim hasil prosesnya berupa data yang diinginkan kembali ke browser [11].

Web server merupakan suatu *software* yang memberikan pelayanan berbasis data dan berfungsi menerima *request* dari HTTP atau HTTPS dari klien yang dikenal yang biasanya kita kenal dengan nama web browser dan untuk dikirimkan kembali yang dimana hasilnya halaman web dan pada umumnya akan berbentuk dokumen HTML. Itulah pengertian web server yang sebenarnya. Dengan kata lain web server

akan mengirimkan data dalam bentuk HTML pada permintaan web Browser sehingga akan terlihat seperti pada umumnya yaitu sebuah tampilan website.

### 2.1.13. Java

Java dikenal memiliki moto “*Write Once, Run Anywhere*”. Hal ini menunjukkan bahwa setiap kode program yang ditulis menggunakan Java bisa langsung dijalankan pada platform berbeda-beda tanpa perlu menyusun ulang [12].

Java adalah nama bahasa pemrograman yang terkenal di dunia. Seperti yang diketahui, program merupakan kumpulan instruksi yang ditujukan untuk komputer. Melalui program, komputer bisa diatur agar melakukan tugas tertentu sesuai yang ditentukan oleh *programmer*.

Dalam pemrograman Java perlu diketahui beberapa hal sebelum memulai mengerjakan program, berikut merupakan hal-hal yang perlu diketahui :

- **Case Sensitivity** berarti pengidentifikasi Hello dan hello akan memiliki arti yang berbeda pada Java.
- **Class Names**, Untuk semua nama class, huruf pertama harus kapital. Jika terdapat lebih dari satu kata yang digunakan untuk membentuk nama class, huruf pertama setiap kata harus dibuat kapital.  
Contoh class: MyFirstJavaClass
- **Method Names**, Semua nama Method harus diawali dengan huruf kecil. Jika ada lebih dari satu kata yang digunakan untuk membentuk nama method, maka huruf pertama harus dibuat dengan huruf kecil kemudian kata berikutnya harus dibuat dengan huruf kapital.  
Contoh public void myMethodName ()
- **Program File Name**, nama file program harus persis sesuai dengan nama class. Ketika menyimpan suatu file, file tersebut dapat disimpan menggunakan



nama class (ingat Java memiliki case sensitive) dan menambah '.java' di ujung nama (apabila nama file dan nama class tidak cocok, program tidak dapat di compile).

Contoh: 'MainPayrollApp' adalah nama class, maka file tersebut harus disimpan dengan 'MainPayrollApp.java'.

- **public static void main(String args[])**, program Java memulai proses dari method main(), yang merupakan bagian wajib dari setiap program Java.

#### 2.1.14. Maven

Maven berasal dari bahasa yiddi (*jewish german*) yang berarti *accumulator of knowledge* atau bahasa indonesianya penimpun ilmu [13].

Maven adalah *build automation tool* yang berguna untuk mempermudah *build* dan *manage* suatu project. Dengan menggunakan maven aplikasi dapat menggunakan *dependency* atau modul dengan mudah dan cepat.

#### 2.1.15. Mysql Server

MySQL adalah basis database manajer atau (*database management system*) atau DBMS yang multithread dan juga multi-user. MySQL telah diinstal sekitar 6 juta lebih di seluruh dunia. MySQL AB telah membuat MySQL menjadi *opens source* dibawah lisensi GNU *General Public License* (GPL), namun mereka juga menjual dibawah lisensi komersial untuk di mana penggunaannya tidak cocok dengan penggunaan GPL.

Database MySQL memiliki banyak tipe data untuk digunakan pada saat membuat tabel pada database. Tipe data tersebut berpengaruh dengan data yang akan kita simpan didalam tabel pada database. Secara garis besar, tipe data yang sering dipakai adalah tipe angka (INTEGER), texts (VARCHAR, TEXT) dan waktu (DATE, DATETIME, TIMESTAMP).

Memilih tipe data yang cocok nantinya akan memudahkan dalam pengaturan tabel, seperti penggunaan tipe data DATE yang akan otomatis membuat format standar DATE misalnya '2014-11-13'. Tipe data wajib ditetapkan pada saat pembuatan tabel, karena apabila dirubah pada saat sudah memiliki *record* akan berakibat mengacaukan isi tabel yang berakibat pada kehilangan data.

## **2.2. Penelitian Terkait**

Penulis telah menemukan beberapa penelitian terkait yang berhubungan dengan penelitian yang penulis lakukan, diantaranya sebagai berikut :

- Penelitian oleh Ahmat Josi yang berjudul “PERANCANGAN APLIKASI PENGGAJIAN PADA PERGURUAN TINGGI (Studi kasus sekolah tinggi XYZ)” pada tahun 2017 [1]. Penelitian ini dilakukan untuk membuat sistem penggajian lebih mudah untuk perguruan tinggi XYZ. Disini sang penulis menggunakan aplikasi java desktop. Berkat aplikasi yang dibuat proses perhitungan gaji menjadi cepat dan akurat, dari 1 minggu hingga beberapa menit saja, dan pengulangan perhitungan gaji yang dilakukan 3-7 kali hanya menjadi 1-2 kali saja.
- Penelitian oleh Andi Saryoko, Arie Kurniawan, Sopiyan Dalis dan Rahmat Hidayat yang berjudul “Sistem Informasi Human Capital management System Berbasis Web Pada PT. Asia Berjaya Mobilindo” 2019 [3]. Penelitian ini dilakukan untuk membuat web aplikasi penggajian otomatis untuk PT. Asia Berjaya Mobilindo. Sang penulis menggunakan bahasa pemrograman PHP untuk membuat web aplikasi tersebut. aplikasi tersebut memiliki fitur pengelolaan data karyawan, cuti karyawan, pencetakan slip gaji dan melihat absensi.

- Penelitian oleh Wahyu Nur Cholifah, Yulianingsih, dan Sri Melati Sagita yang berjudul “PENGUJIAN *BLACK BOX TESTING* PADA APLIKASI *ACTION & STRATEGY* BERBASIS ANDROID DENGAN TEKNOLOGI *PHONEGAP*” 2018 [10]. Penelitian ini dilakukan untuk mengetes aplikasi *Action & Strategy* yaitu aplikasi yang digunakan oleh anggota brimob untuk meningkatkan mutu dan kualitas dari setiap personil. Hasil penelitian testing tersebut berjalan lancar dan tidak memiliki masalah.

Dari penelitian terkait diatas dapat dirangkum pada tabel dibawah ini :

Tabel II-1 Penelitian Terkait

| No . | Nama Peniliti dan Tahun  | Judul   | Masalah  | Solusi   | Hasil   |
|------|--|---|--|--|---|
| 1    | Ahmat Josi, 2017   | PERANCANGAN APLIKASI PENGGAJIAN PADA PERGURUAN TINGGI (Studi kasus sekolah tinggi XYZ)        | Penghitungan dan penggajian pada perguruan tinggi XYZ membutuhkan waktu lama karena harus dilakukan secara manual          | Membuat aplikasi java desktop                  | Penggajian yang tadinya membutuhkan 1 minggu hanya menjadi beberapa menit saja, dan yang tadinya harus diulang berkali-kali hanya menjadi 1-2 kali saja |
| 2    | Andi Saryoko, Arie Kurniawan, Sopiyan Dalis dan Rahmat Hidayat, 2019 | Sistem Informasi Human Capital management System Berbasis Web Pada PT. Asia Berjaya Mobilindo | Perhitungan gaji, manajemen cuti, dan absensi di PT. Asia berjaya Mobilindo masih secara manual sehingga cukup menyulitkan | Membuat web aplikasi majamem en human resource | Memudahkan pengolahan data karyawan, data cuti, pencetakan slip gaji dan melihat absensi.   |

|   |   |   |   |                                  |  |
|---|---|---|---|----------------------------------|--|
|   |   |   | untuk melakukan rekapitulasi absensi, dan cuti.   |                                  |  |
| 3 | Wahyu Nur Cholifah, Yulianingsih, dan Sri Melati Sagita, 2018 | PENGUJIAN <i>BLACK BOX TESTING</i> PADA APLIKASI <i>ACTION &amp; STRATEGY</i> BERBASIS ANDROID DENGAN TEKNOLOGI <i>PHONEGAP</i> | Aplikasi <i>Action &amp; Strategy</i> yang akan digunakan belum tentu berguna untuk anggota brimob. | Melakukan testing pada aplikasi. | Aplikasi berjalan sesuai seperti yang diharapkan, tidak ditemukan kesalahan. |