

BAB II

LANDASAN TEORI

2.1. Konsep Dasar *Web*

2.1.1. *Website*

Menurut Abdul Kadir “*Web* merupakan salah satu sumber daya *internet* yang berkembang pesat, pendistribusian informasi *web* dilakukan pendekatan *hyperlink*, yang memungkinkan suatu teks, gambar, ataupun objek yang lain menjadi acuan untuk membuka halaman-halaman yang lain.” (Harison & Syarif, 2016).

“*Website* adalah halaman web yang saling berhubungan yang berisi kumpulan informasi yang berupa teks, gambar, animasi, audio dan video bisa diakses melalui jalur koneksi *internet* yang dibuat untuk personal, organisasi dan perusahaan.” (Adiwisastra & Hikmah, 2020:1).

Menurut Arief *website* adalah kumpulan halaman-halaman yang digunakan untuk mempublikasikan informasi berisikan dokumen-dokumen multimedia berupa teks, gambar dan program multimedia lainnya berupa animasi (gambar gerak, tulisan gerak), suara dan atau gabungan dari semuanya itu baik yang bersifat statis atau dinamis (Permana, 2016).

2.1.2. *Internet*

“*Internet* adalah jaringan komputer yang sangat luas, menjangkau seluruh dunia dan memungkinkan terjadinya pertukaran informasi/data.” (Farlina et al., 2020:2).

“*Internet* atau *interconnection networking* (keterhubungan antar jaringan) merupakan jaringan komputer yang terluas, dengan cakupan seluruh planet bumi.” (Irawan, 2018).

“*Internet* adalah jaringan atau sistem pada jaringan komputer yang saling berhubungan dengan menggunakan sistem *global transmission control protocol* atau *internet protocol suite* (TCP/IP) sebagai protokol pertukaran paket untuk melayani miliaran pengguna diseluruh dunia.” (Prastyo & Wahyuni, 2019).

2.1.3. *Web Browser*

“*Web Browser* adalah aplikasi yang digunakan untuk berselancar di dunia *internet*. *Browser* dapat memandu pengguna *internet* untuk berpindah antar situs *web* dengan mudah.” (Farlina et al., 2020:3).

“*Web browser* adalah *software* yang digunakan untuk menampilkan tampilan halaman *web* dan informasi dari *server web*. *Software* ini kini telah dikembangkan dengan menggunakan *user interface grafis*, sehingga pemakai data melakukan ‘*point and click*’ untuk pindah antar dokumen.” (Adiwisastra & Hikmah, 2020:5).

Menurut Kustiyaningsih “*Web browser* adalah *software* yang digunakan untuk menampilkan informasi dari *server web*.” (Sukisno & Wuni, 2017).

2.1.4. *Web Server*

Web server yaitu Komputer yang digunakan untuk menyimpan dokumen-dokumen *web*, komputer ini akan melayani permintaan dokumen *web* dari kliennya. *Web browser* seperti *explorer* atau *navigator* berkomunikasi melalui jaringan (termasuk jaringan *internet*) dengan *web*, menggunakan HTTP. *Browser* akan mengirikan *request* ke *server* untuk meminta dokumen tertentu atau layanan lain yang di sediakan oleh *server*. *Server* memberikan dokumen satu layanan jika tersedia juga dengan menggunakan *protocol* HTTP. Salah satu aplikasi *server localhost* serta yang paling banyak digunakan dan cukup familiar di kalangan *web developer* saat ini adalah XAMPP (Adiwisastra & Hikmah, 2020:5).

“*Web server* adalah perangkat lunak yang berfungsi sebagai penerima permintaan yang dikirimkan melalui *browser*, kemudian memberikan tanggapan permintaan dalam bentuk halaman situs *web* atau lebih umumnya dalam dokumen HTML.” (Farlina et al., 2020:4).

2.1.5. HTTP (*Hypertext Transfer Protocol*)

“HTTP adalah suatu *protocol* yang menyampaikan berbagai informasi dari *server website* untuk ditampilkan kepada *user* atau pemakai melalui *web browser*.” (Adiwisastro & Hikmah, 2020:2).

“HTTP adalah bagian URL yang mengidentifikasi lokasi web dan digunakan dalam protokol HTML.” (Farlina et al., 2020:3).

2.1.6. PHP (*Hypertext Preprocessor*)

“PHP adalah salah satu bahasa pemrograman *script* yang dirancang untuk membangun aplikasi *web*.” (Sabaruddin & Jayanti, 2019:3).

“PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah *script* HTML.” (Firman et al., 2016).

PHP merupakan salah satu aplikasi eksternal yang bisa digunakan oleh *server web*, sehingga *server web* tidak sekedar untuk memberikan layanan dokumen HTML saja, tetapi juga menjadi program yang menerima masukan dari luar dan memberikan keluaran yang berasal dari *database* ataupun sumber data lainnya menjadi dokumen HTML (Irawan, 2018).

2.1.7. CSS (*Cascading Style Sheets*)

“CSS adalah singkatan dari *Cascading Style Sheets* yaitu kumpulan kode program yang digunakan untuk mendesain atau mempercantik tampilan halaman HTML.” (Adiwisastra & Hikmah, 2020:87).

“*Cascading Style Sheets* (CSS) merupakan salah satu bahasa pemrograman *web* untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam.” (Farlina et al., 2020:65).

Menurut Prasetio “CSS adalah suatu teknologi yang digunakan untuk memperindah halaman *website*.” (Sagita & Sugiarto, 2016).

2.1.8. *Javascript*

Menurut Susanto & Oscar “*Javascript* adalah bahasa pemrograman yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam *browser* menjadi lebih interaktif, tidak sekedar indah saja.” (Farlina et al., 2020:75).

“*Javascript* adalah bahasa pemrograman web yang bersifat *Client Side* Programming language. *Client side* programming language adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh client. Aplikasi *client* yang dimaksud merujuk kepada web *browser* seperti *Google Chrome* dan *Mozilla Firefox*. JavaScript digunakan untuk membuat interaksi atau menambah fitur web dinamis kedalam sebuah web.” (Adiwisastra & Hikmah, 2020:63).

Menurut Prasetio “*Javascript* adalah program dalam bentuk *script* yang dijalankan oleh interpreter yang telah ditanamkan kedalam *browser web*, sehingga *browser web* dapat mengeksekusi program *javascript*.” (Sagita & Sugiarto, 2016).

2.1.8. HTML (*Hypertext Markup Language*)

Menurut Winarno “*Hypertext Markup language (HTML)* adalah sebuah bahasa pemrograman untuk menampilkan konten di *website*. Dan disebut juga sebagai bahasa pemrograman yang bebas artinya dalam pengembangannya dapat dikembangkan bersama secara global.” (Adiwisatra & Hikmah, 2020:11).

Menurut Anhar HTML (*Hypertext Markup Language*) adalah sekumpulan symbol-simbol atau tag-tag yang dituliskan dalam sebuah file yang digunakan untuk menampilkan halaman pada *web browser*. Tag-tag HTML selalu diawali dengan `<x>` dan diakhiri `</x>` dimana x tag HTML itu seperti b, I, u, dan lain-lain (Farlina et al., 2020:11).

Menurut Prasetio “HTML (*hypertext markup language*) adalah bahasa pemrograman yang digunakan untuk mendesain sebuah halaman.” (Sagita & Sugiarto, 2016).

2.1.9. JQuery

“*JQuery* merupakan sebuah *Javascript Library* atau bisa disebut juga sebagai perpustakaan dari kumpulan kode/listing *Javascript* yang siap pakai. Dalam arti sederhana, *jquery* dapat digunakan untuk meringkas sebuah listing *javascript* yang panjang dalam sebuah proyek pembuatan *website*.” (Harison & Syarif, 2016).

“*JQuery* merupakan program yang berjalan pada sisi *server* dan akan ditampilkan pada *Browser Web*. *JQuery* dapat berjalan di dalam HTML, atau bahasa pemrograman berbasis *web* lainnya.” (Harison & Syarif, 2016).

Menurut Saputra dan Agustin “*Jquery* merupakan salah satu teknik atau kumpulan *library javascript* yang sangat terkenal dengan animasinya. Dengan sedikit sentuhan, animasi dalam *website* mudah kita ciptakan. *Jquery* dapat dianggap sebagai *framework javascript*.” (Sagita & Sugiarto, 2016).

2.1.10. Adobe Dreamweaver

“*Adobe Dreamweaver* adalah perangkat lunak terkemuka untuk desain *web* yang menyediakan kemampuan *visual* yang intuitif termasuk pada tingkat kode, yang digunakan untuk membuat dan mengedit *website* HTML serta aplikasi *mobile* seperti *smartphone*, *tablet* dan perangkat lainnya.”(Irawan, 2018).

“*Dreamweaver* adalah *software* aplikasi yang digunakan sebagai HTML *editor profesional* untuk mendisain *web* secara *virtual*.” (Suhartono, 2015).

Menurut Sibero *Dreamweaver* merupakan sebuah produk *web developer* yang dikembangkan oleh *Adobe System Inc*, sebelumnya produk *Dreamweaver* dikembangkan oleh *Macromedia Inc*, yang kemudian sampai saat ini perkembangannya diteruskan oleh *Adobe System Inc*, *Dreamweaver* dikembangkan dan dirilis dengan kode nama *Creative Suit* (CS) (Sukisno & Wuni, 2017).

2.1.11. Basis Data

“Basis data merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implisit, sehingga data yang terkumpul secara acak dan tanpa mempunyai arti, tidak dapat disebut sebagai *database*.” (Harison & Syarif, 2016).

Menurut Kenneth “Basis data atau *database* adalah sekumpulan data organisasi untuk melayani banyak aplikasi secara efisien dengan memusatkan data dan mengendalikan *redundancy data*.” (Purnama, 2016:118).

Menurut Fathansyah “Basis data adalah himpunan kelompok data atau arsip yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.” (Sagita & Sugiarto, 2016).

2.1.12. *MySQL (My Structure Query Language)*

Menurut Kustiyahningsih “*MySQL* adalah sebuah basis data yang mengandung satu atau jumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel. Tabel terdiri dari sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel.” (Firman et al., 2016).

“*MySQL* merupakan *software Relational Database Management System (RDBMS)* yang digunakan untuk mengelola *database* dengan cepat, dapat menampung data dengan jumlah yang sangat besar, dapat diakses oleh banyak *user (multiuser)* dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*).” (Irawan, 2018).

“*MySQL (My Structure Query Language)* adalah suatu *RDBMS (Relational Database Management System)* yaitu aplikasi sistem yang menjalankan fungsi pengelolaan data.” (Prastyo & Wahyuni, 2019).

2.1.13. *XAMPP*

Menurut Wahana mengemukakan bahwa “*XAMPP* adalah salah satu paket instalasi *apache*, *PHP*, dan *MySQL* secara instant yang dapat digunakan untuk membantu proses instalasi ketiga produk tersebut”. (Agus & Safitri, 2015).

Menurut Puspitasari “*Xampp* adalah sebuah *software web browser* yang didalamnya sudah tersedia *database server MySQL* dan *support PHP programming*.” (Sukisno & Wuni, 2017).

2.1.14. *PhpMyAdmin*

“*PhpMyadmin* merupakan aplikasi berbasis web yang ditulis menggunakan bahasa pemrograman *PHP*. Selain memakai *MySQL* untuk menjalankannya,

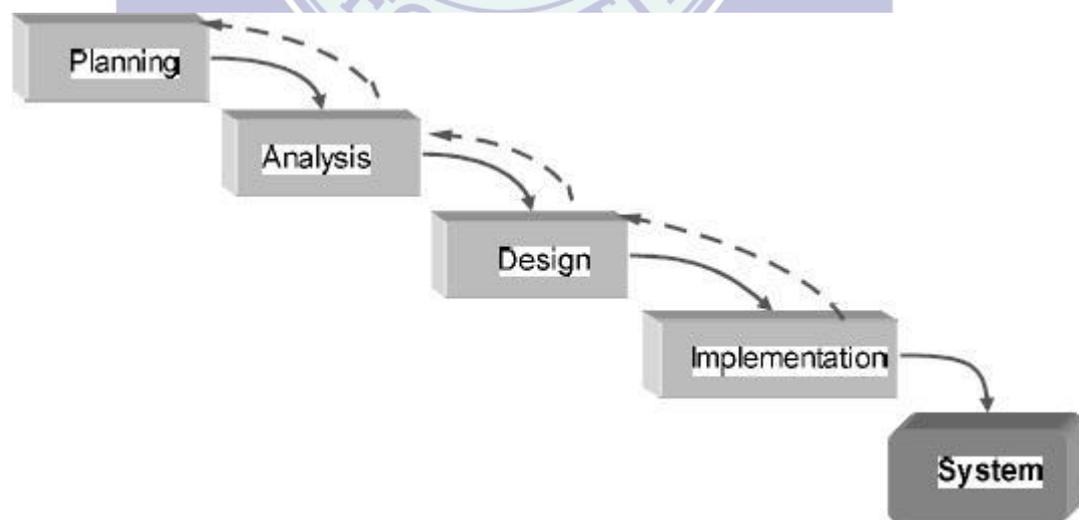
PhpMyadmin memerlukan *web server* yang sering digunakan berpasangan dengan PHP yaitu *Apache Web Server*.” (Achyani & Saumi, 2019).

“*PhpMyadmin* merupakan aplikasi *web PHP* sebagai *administrator MySQL*, *PhpMyadmin* mendukung aktifitas *MySQL* sebagai pengelola data, tabel, relasi antar tabel dan sebagainya.” (Prastyo & Wahyuni, 2019).

2.1.15. Model Pengembangan Perangkat Lunak

Menurut Sukamto & Salahuddin “Model *System Development Life Cycle* (SDLC) sering juga disebut sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*).” (Achyani & Saumi, 2019).

Menurut Susanto “SDLC (*System Development Life Cycle*) adalah salah satu metode pengembangan sistem informasi yang populer pada saat sistem informasi pertama kali dibuat.” (Budi et al., 2016).



Sumber: (Budi et al., 2016)

Gambar II.1.
Metode Waterfall

“Model air terjun (*waterfall*) merupakan penyedia pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain pengodean, pengujian dan tahap pendukung (*support*).” (Prastyo & Wahyuni, 2019).

Metode *Waterfall* adalah metode pengembangan perangkat lunak dengan pendekatan sekuensial dengan cakupan aktivitas (Budi et al., 2016), sebagai berikut:

1. Rekayasa Sistem dan Analisis (*System Engineering and Analysis*)

Karena perangkat lunak adalah bagian dari sistem yang lebih besar, pekerjaan dimulai dari pembentukan kebutuhan-kebutuhan untuk seluruh elemen sistem dan kemudian memilah mana yang untuk pengembangan perangkat lunak. Hal ini penting, ketika perangkat lunak harus berkomunikasi dengan *hardware*, orang dan basis data.

2. Analisis Kebutuhan Perangkat Lunak (*Software Requirement Analysis*)

Pengumpulan kebutuhan dengan fokus pada perangkat lunak, yang meliputi: *domain* informasi, fungsi yang dibutuhkan, untuk kerja/performansi dan antarmuka. Hasilnya harus didokumentasikan dan di-*review* ke pelanggan.

3. Perancangan (*Design*)

Ada empat atribut untuk program yaitu: struktur data, arsitektur perangkat lunak, prosedur detail dan karakteristik antarmuka. Proses desain mengubah kebutuhan-kebutuhan menjadi bentuk karakteristik yang dimengerti perangkat lunak sebelum dimulai penulisan program. Desain ini harus terdokumentasi dengan baik dan menjadi bagian dari konfigurasi perangkat lunak.

4. Pembuatan Kode (*Coding*)

Penterjemahan perancangan ke bentuk yang dapat dimengerti oleh mesin, dengan menggunakan bahasa pemrograman.

5. Pengujian (*Testing*)

Setelah kode program selesai testing dapat dilakukan. Testing memfokuskan pada logika internal dari perangkat lunak, fungsi eksternal dan mencari segala kemungkinan kesalahan dan memeriksa apakah sesuai dengan hasil yang diinginkan.

6. Pemeliharaan (*Maintenance*)

Merupakan bagian paling akhir dari siklus pengembangan dan dilakukan setelah perangkat lunak dipergunakan, meliputi kegiatan-kegiatan: *Corrective Maintenance*, *Adaptive Maintenance* dan *Perfektive Maintenance*.

2.2. Teori Pendukung

2.2.1. Struktur Navigasi

“Struktur navigasi adalah alur dari suatu program yang rancangan hubungannya (rantai kerja) dari beberapa area yang berbeda dan dapat membantu mengorganisasikan seluruh elemen pembuatan *website*.” (Adiwisastro & Hikmah, 2020:2).

Menurut Suyanto, “Struktur navigasi adalah situs *web* melibatkan sistem navigasi situs *web* secara keseluruhan dan *desain interface* situs *web* tersebut, navigasi memudahkan jalan yang mudah ketika menjelajahi situs *web*”. (Farlina et al., 2020:5).

Menurut Binanto terdapat empat struktur dasar navigasi yang digunakan pada produk multimedia (Farlina et al., 2020:5), yaitu:

1. Struktur Navigasi *Linier*

Struktur navigasi *linier* hanya mempunyai satu rangkaian cerita yang berurut, yang menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang dapat ditampilkan pada struktur jenis ini adalah satu halaman sebelumnya atau satu halaman sesudahnya, tidak dapat dua halaman sebelumnya atau dua halaman sesudahnya.



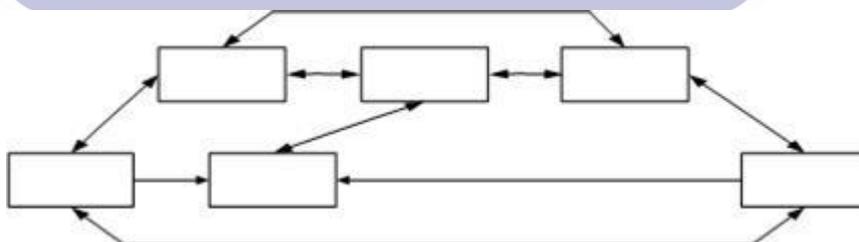
Sumber: (Farlina et al., 2020:5)

Gambar II.2.

Struktur Navigasi *Linier*

2. Struktur Navigasi *Non-Linier*

Struktur navigasi *non-linier* atau struktur tidak berurut merupakan pengembangan dari struktur navigasi *linier*. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan yang dibuat pada struktur *non-linier* ini berbeda dengan percabangan pada struktur *hirarki*, karena pada percabangan *non-linier* ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada *Master Page* dan *Slave Page*.



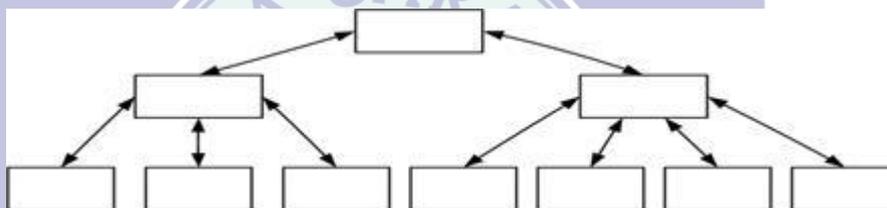
Sumber: (Farlina et al., 2020:6)

Gambar II.3.

Struktur Navigasi *Non-Linier*

3. Struktur Navigasi *Hirarki*

Struktur navigasi *hirarki* biasa disebut struktur bercabang, merupakan suatu struktur yang mengandalkan percabangan untuk menampilkan data berdasarkan kriteria tertentu. Tampilan pada menu pertama akan disebut sebagai *Master Page* (halaman utama pertama), halaman utama ini mempunyai halaman percabangan yang disebut *Slave Page* (halaman pendukung). Jika salah satu halaman pendukung dipilih atau diaktifkan, maka tampilan tersebut akan bernama *Master Page* (halaman utama kedua), dan seterusnya. Pada struktur navigasi ini tidak diperkenankan adanya tampilan secara *linier*.

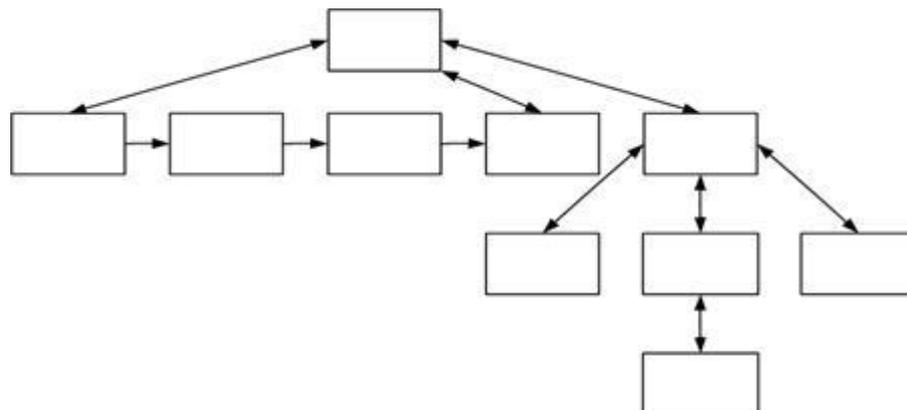


Sumber: (Farlina et al., 2020:6)

Gambar II.4.
Struktur Navigasi *Hirarki*

4. Struktur Navigasi Campuran

Struktur navigasi campuran merupakan gabungan dari ketiga struktur sebelumnya yaitu *linier*, *non-linier* dan *hirarki*. Struktur navigasi ini juga biasa disebut dengan struktur navigasi bebas. Struktur navigasi ini banyak digunakan dalam pembuatan *website* karena struktur ini dapat digunakan dalam pembuatan *website* sehingga dapat memberikan ke-interaksian yang lebih tinggi.



Sumber: (Farlina et al., 2020:7)

Gambar II.5.

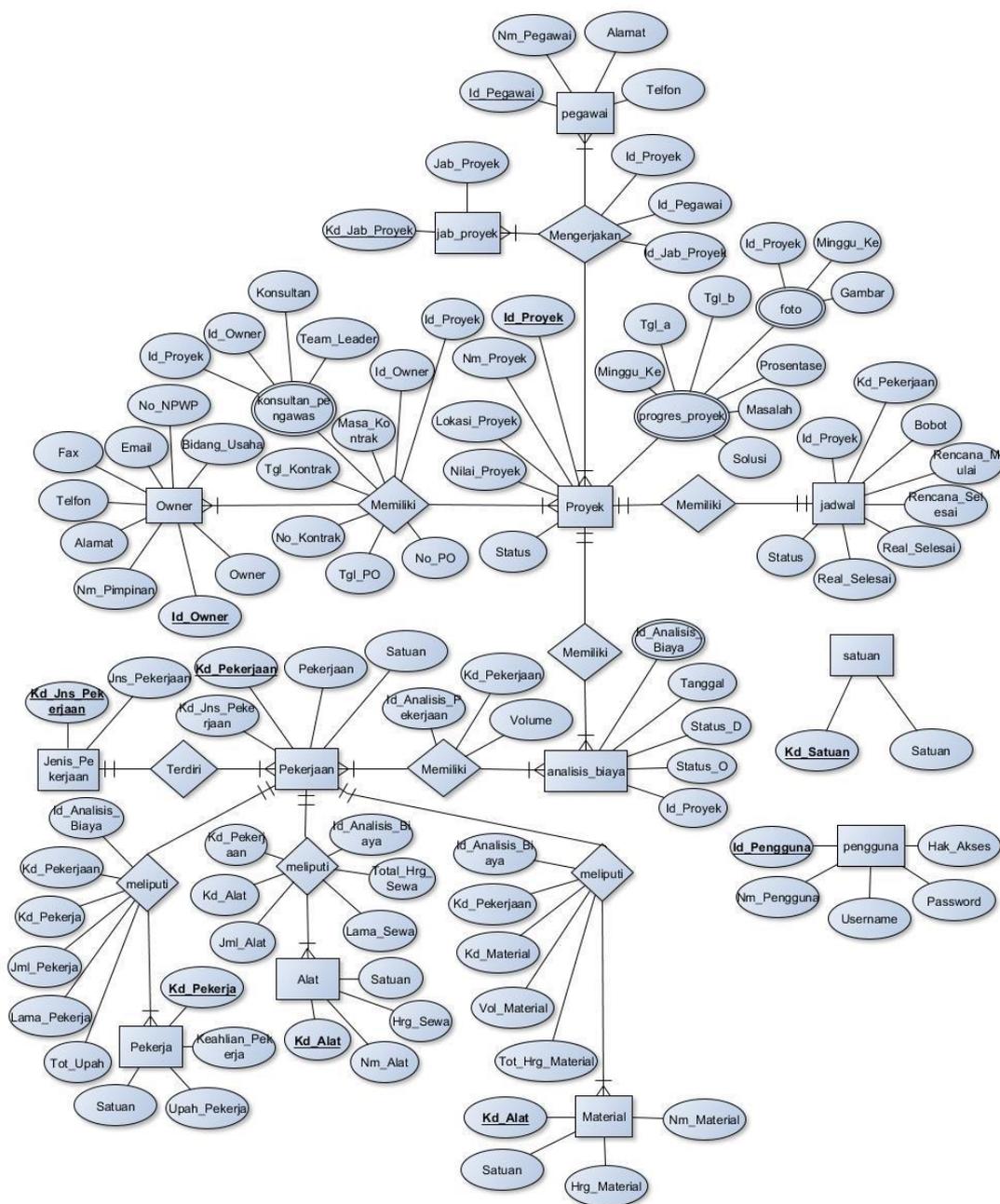
Struktur Navigasi Campuran

2.2.2. ERD (*Entity Relationship Diagram*)

“ERD adalah model data untuk menggambarkan hubungan antara satu entitas dengan entitas lain yang mempunyai relasi (hubungan) dengan batasan-batasan.” (Harison & Syarif, 2016).

ERD (*Entity Relationship Diagram*) adalah suatu model penyajian data dengan menggunakan *entity* dan *relationship*, ERD menggambarkan model konseptual untuk menggambarkan struktur logis dari basis data berbasis grafis yang bertujuan agar basis data dapat dipahami dan dirancang dengan mudah (Setiyani, 2018:184).

Menurut Utami “*Entity Relationship Diagram* adalah gambaran relasi antar table dengan tujuan untuk memperjelas hubungan antar tabel penyimpanan. ERD terdiri atas sekumpulan objek dasar yaitu entitas dan hubungan antar entitas-entitas yang saling berhubungan.” (Prihantara et al., 2018).



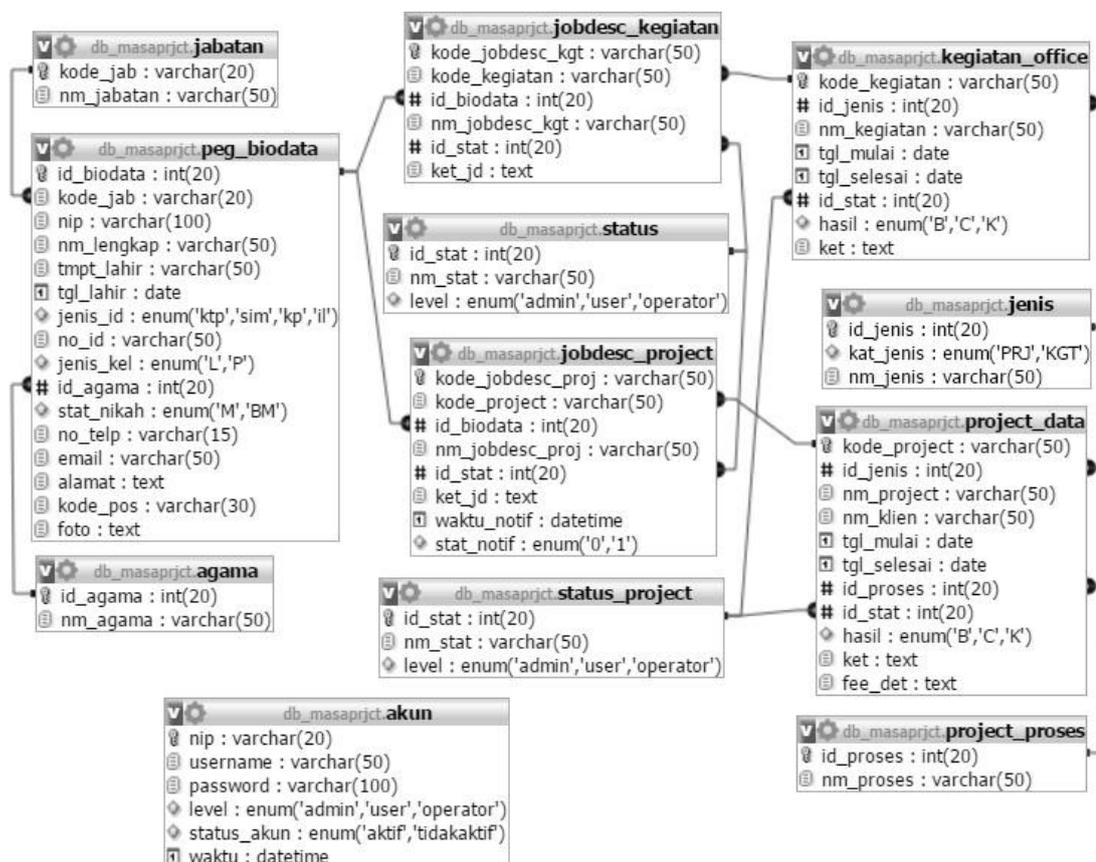
Sumber: (Prihantara et al., 2018)

Gambar II.6.

Ilustrasi Entity Relationship Diagram

2.2.3. LRS (Logical Record Structure)

“Logical Record Structure (LRS) merupakan representasi dari struktur record pada tabel-tabel yang terbentuk dari hasil antara himpunan entitas.” (Achyani & Saumi, 2019).



Sumber: (Permana, 2016)

Gambar II.7.

Ilustrasi Logical Record Structure

2.2.4. Implementasi dan Pengujian Web

Menurut Rizky *Black-Box testing* (pengujian kotak hitam) adalah tipe *testing* yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya, sehingga para *tester* memandang perangkat lunak seperti layaknya sebuah kotak hitam yang tidak penting dilihat isinya, tapi cukup dikenali proses *testing* dibagian luar (Sagita & Sugiarto, 2016).

“*Black box testing* merupakan strategi pengujian yang memperhatikan atau memfokuskan kepada faktor fungsionalitas dan spesifikasi perangkat lunak.” (Setiyani, 2018:300).

Berikut ini adalah langkah-langkah dari proses black box testing (Setiyani, 2018:300), yaitu:

1. Menganalisis kebutuhan spesifikasi dari perangkat lunak.
2. Pemilihan jenis *input* yang memungkinkan menghasilkan *output* salah pada perangkat lunak yang sedang diuji.
3. Menentukan output untuk suatu jenis input.
4. Pengujian dilakukan dengan input-input yang telah benar-benar diseleksi.
5. Perbandingan output yang dihasilkan dengan output yang diharapkan.
6. Menentukan fungsionalitas yang seharusnya pada perangkat lunak yang sedang diuji.

Pengujian *black bos testing* dapat dilakukan pada setiap *level* pembangunan sistem yaitu dimulai dari *unit*, *integration*, *system* dan *acceptance*. Teknik-teknik dalam *black box testing* (Setiyani, 2018:301), adalah sebagai berikut:

1. *Equivalence Partitioning*

Teknik ini merupakan teknik pengujian *software* yang melibatkan pembagian nilai *input* ke dalam bagian nilai *valid* dan tidak *valid* dan memilih perwakilan dari masing-masing data tes.

2. *Boundary Value Analysis*

Teknik ini merupakan teknik pengujian *software* yang melibatkan penentuan-penentuan nilai *input* dan memilih beberapa nilai dari batasan tersebut baik luar maupun dalam batasan-batasan tersebut sebagai data tes.

3. Cause Effect Graphic

Teknik ini merupakan teknik pengujian *software* yang melibatkan pengidentifikasian sebab-sebab (kondisi *input*) dan akibat-akibat (kondisi *output*) menghasilkan kasus-kasus tes.

Tabel II.1

Contoh Pengujian *Black Box Testing*

No	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Nama Kota dan Ongkos Kirim tidak diisi kemudian klik tombol simpan data	Nama Kota: (kosong) Ongkir: (kosong)	Sistem akan menolak akses <i>user</i> dan menampilkan "Semua Bidang Wajib Diisi"	Sesuai harapan	Valid
2	Mengetikan Nama Kota dan Ongkos Kirim tidak diisi atau kosong kemudian klik tombol simpan data	Nama Kota: Jakarta Ongkos Kirim: (kosong)	Sistem akan menolak akses <i>user</i> dan menampilkan "Semua Bidang Wajib Diisi"	Sesuai harapan	Valid
3.	Nama Kota tidak diisi dan Ongkos Kirim diisi dengan benar kemudian klik tombol simpan data	Nama Kota: (kosong) Ongkos Kirim: 29000	Sistem akan menolak akses <i>user</i> dan menampilkan "Semua Bidang Wajib Diisi"	Sesuai harapan	Valid
4.	Nama Kota sudah ada di <i>database</i> kemudian klik tombol simpan data	Nama Kota: Jakarta (sudah ada) Ongkos Kirim: 29000	Sistem akan menolak akses <i>user</i> dan menampilkan "Kota Sudah Terdaftar"	Sesuai harapan	Valid
5.	Nama Kota dan Ongkos Kirim diisi dengan benar kemudian klik tombol simpan data	Nama Kota: Bandung Ongkos Kirim: 32000	Sistem menerima akses <i>user</i> dan menampilkan "Kota Berhasil Disimpan"	Sesuai harapan	Valid

Sumber: (Utami & Khasanah, 2018)