**BAB II**

**LANDASAN TEORI**

**2.1 Konsep Dasar Web**

1. ***Website***

Menurut Bowo (2013:6) “*Website* adalah halaman informasi yang disediakan melalui internet sehingga bisa diakses seluruh dunia selama terkoneksi dengan jaringan internet”. *Website* merupakan komponen atau kumpulan komponen yang terdiri dari teks, gambar, suara, video, animasi dan lain sebagainya. Sehingga lebih merupakan media informasi yang menarik untuk dikunjungi.

1. ***Web Browser***

Menurut Sidik dan Pohan(2014:5) “*Web Browser* adalah software yang digunakan untuk menampilkan informasi dari *server web*.” Software kini telah dikembangkan dengan menggunakan *user interface grafis* sehingga pemakai dapat dengan mudah melakukan *point* dan *click* untuk pindah antar dokumen.

1. ***Web Server***

Menurut Sadeli (2014:2) “*Web Server* adalah sebuah perangkat lunak server yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan *Web Server Browser* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman web yang umumnya berbentuk dokumen HTML”.

1. ***Uniform Resource Locator* *(URL)***

Menurut Bowo (2013:16) “*Uniform Resource Locator* *(URL)* merupakan suatu rangkaian karakter dengan format dan standart tertentu yang digunakan untuk menunjukkan alamat suatu sumber seperti dokumen dan gambar di internet”. *URL* merupakan *link* atau *tautan* menuju ke halaman atau konten dalam sebuah *website*.

Sebagian orang sering salah persepsi menyamakan *domain* dengan *URL,* padahal keduanya berbeda. Dalam kontek sebuah *website*, domain seperti yang kita pelajari sebelumnya mengacu pada sistem penamaan *server* atau *website*. Sedangkan *URL* mengacu pada peng-alamatan sebuah file, dokumen atau konten dalam sebuah *website.*

**2.1.2 Bahasa Pemrograman**

Menurut dipraja (2014:26), *programming language* (bahasa pemograman) merupakan suatu sintak untuk mendefinisikan program komputer, bahasa ini memungkinkan seorang *programming* dapat membuat suatu program aplikasi, contohnya: *borland delphi.*

1. ***HyperText Markup Language (HTML)***

Menurut Rozi (2016:17) “*HyperText Markup Language (HTML)* adalah bahasa pengkodean yang digunakan untuk membuat halaman web agar bisa ditampilkan melalui *web browser*”.

1. ***HyperText Processor (PHP)***

Menurut Wahana Komputer (2012:88) ***HyperText Processor (PHP)*** atau sering disebut PHP merupakan bahasa pemograman berbasis server-side yang dapat melakukan parsing script php menjadi script web sehingga dari sisi client menghasilkan suatu tampilan yang menarik. PHP merupakan perkembangan dari FI atau Form Interface yang dibuat oleh Rasmus Lerdoff pada tahun 1995.

1. ***Cassanding Style Sheets* (*CSS*)**

Menurut Rozi (2016:69) “*Cassanding Style Sheets* *(CSS)* adalah bahasa pengkodean yang digunakan untuk menata gaya tampilan halaman web agar lebih cantik dan indah saat ditampilkan di *web browser*.

1. ***Dreamweaver CS5***

Menurut Gary B.Shelly, Dolores j. Wel T.Campbell (2012:8) “*Dreamweaver* adalah perangkat lunak web editor keluaran *Adobe System* yang digunakan untuk membangun dan mendesain suatu website dengan fitur-fitur yang menarik dan kemudahan dalam penggunaanya.

Selain mudah digunakan dalam membuat dan mendesain web, *dreamweaver* juga menambah fleksibilitasnya dengan bahasa pemrograman web yang lainnya, dan dapat berinteraksi pada beberapa perangkat lunak lainnya, dan tidak salah jika perangkat lunak ini menjadi pilihan utama bagi para *web desainer* amatir maupun *expert*.

Pada *dreamweaver cs5* terdapat beberapa fitur baru yang keren yang dapat digunakan, mulai dari css, pengguna fitur canggih *ajax* dan *jquery* menggunakan browser widget serta pengujian tampilan web menggunakan *web adobe browser lab*.

1. ***Apache,MySql,Php,Perl (XAMPP)***

Menurut Sidik (2014:72) “*XAMPP* merupakan paket *server web php* dan database *mysql* yang paling popular dikalangan pengembangan web dengan menggunakan *php* dan *mysql* sebagai databasenya”.

**2.1.3 Basis Data**

Menurut Sukamto R.A dan Shalahuddin M (2015:43) “Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau membuat informasi tersedia saat dibutuhkan”.

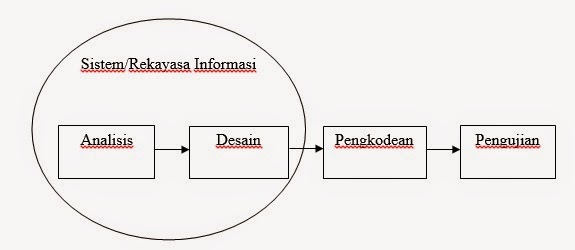
1. ***My Structure Query Language* (*My SQL)***

Menurut Raharjo (2015:355) “*My SQL Software* yang bersifat open source yang paling popular, digunakan untuk menyimpan data dari aplikasi berbasis web”.

Dalam paket distribusinya, PHP menyediakan tiga pustaka (*library*) yang dapat diguanakan untuk mengakses database *MySql* secara mudah, yaitu *php\_mysql.dll*, *php\_mysqlli.dll*, dan *php pdo\_mysql.dll*. Masing-masing pustaka tersebut memiliki daftar fungsi maupun kelas yang berbeda dalam penggunaanya.

**2.1.4 Model Pengembangan Perangkat Lunak**

Menurut Sukamto R.A dan Shalahuddin M (2015:28) “Model SDLC air terjun *(waterfall)* sering juga disebut model Sequensial Linier atau alur hidup klasik (*classic life cycle*)”.



Sumber : Sukamto R.A dan Shalahuddin M (2015:28)

**Gambar II.1**

***Waterfalll***

Gambar menjelaskan bahwa metode *waterfall* menekan pada sebuah keterurutan dalam proses pengembangan perangkat lunak. Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan dalam metode *waterfall.*

1. **Analisis Kebutuhan Perangkat Lunak**

Proses pengumpulan kebutuhan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk di dokumentasikan.

1. **Desain**

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat di implementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu di dokumentasikan.

1. **Pembuatan Kode Program**

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

1. **Pengujian**

Pengujian fokus pada perangkat lunak secara dari segi logic dan fungsional serta memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisirkan kesalahan *(error)* dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

1. **Pendukung (*Support)* Atau Pemeliharaan (*Maintenance*)**

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya 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.

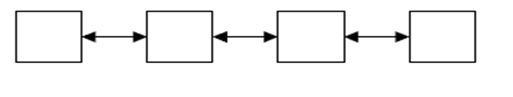
**2.2 Teori Pendukung**

1. **Struktur Navigasi**

Menurut Sutopo (2007:6) menyimpulkan bahwa “struktur navigasi adalah struktur atau alur suatu program yang merupakan rancangan hubungan dan rantai kerja dari beberapa area yang berbeda dan dapat membantu mengorganisasikan seluruh elemen pembuatan *website*”. Menentukan struktur navigasi merupakan halaman yang sebaiknya dilakukan sebelum membuat sebuah web. Ada empat macam bentuk dasar dari struktur navigasi yang biasa digunakan dalam proses pembuatan website, yaitu:

1. **Struktur Navigasi *Linear***

Struktur navigasi *linear* 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.



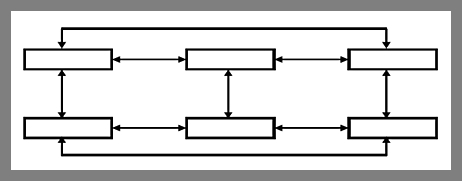
Sumber : Sutopo (2007:6)

**Gambar II.2.**

***Struktur Navigasi Linear***

1. **Struktur Navigasi *Non Linear***

Struktur navigasi *non linear* merupakan pengembangan dari struktur navigasi linear. Pada struktur ini diperkenankan membuat navigasi bercabang. Percabangan pada struktur non linear ini berbeda dengan percabangan pada struktur hirarki, karena pada percabangan non linear ini walaupun terdapat percabangan, tetap tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada *master page* dan *slave page*.



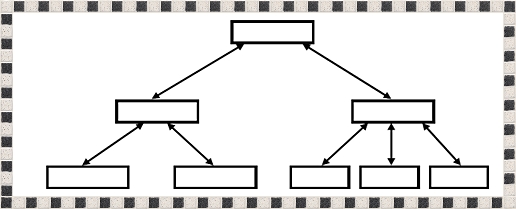
Sumber : Sutopo (2007:6)

**Gambar II.3.**

***Struktur Navigasi Non Linear***

1. **Struktur Navigasi *Hirarki***

Struktur navigasi *hirarki* biasa disebut dengan struktur bercabang, merupakan suatu struktur yang mengandalkan percabangan untuk menampilkan data berdasarkan kriteria tertentu. Tampilan pada menu satu 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 navigasi ini tidak diperkenalkan adanya tampilan secara linear.



Sumber : Sutopo (2007:6)

**Gambar II.4.**

***Struktur Navigasi Hirarki***

**2.2.2  *Enterprise Relationship Diagram (ERD)***

Menurut Sukamto dan Salahuddin (2015:53) “*Enterprise Relationship Diagram (ERD)* adalah bentuk paling awal dalam melakukan perancangan basis data relasional”.

1. Menurut Sukamto dan Salahuddin (2015:50) elemen-elemen diagram hubungan entitas, terdiri dari :
2. **Entitas (*Entity*)**

Entitas adalah suatu yang nyata atau abstrak dimana kita akan menyimpan data. Ada 4 kelas entitas yaitu misalnya pegawai, pembayaran, sekolah dan buku.

1. **Relasi (*Relationship*)**

Relasi adalah yang menghubungkan antar entitas, biasanya diawali dengan kata kerja

1. **Atribut**

Atribut adalah ciri umum semua atau sebagian besar instansi pada entitas tertentu. Sebutan lain dari atribut adalah property, elemen data dan field. Misalnya nama, alamat, nomor pegawai dan gaji adalah atribut entitas pegawai. Sebuah atribut atau kombinasi atribut yang mengidentifikasikan satu dan hanya satu instansi suatu entitas disebut kunci utama atau pengenal. Misalnya nomor pegawai adalah kunci utama untuk pegawai.

1. Menurut Kursini (2007:22) istilah yang harus dipahami sebelum merancang basis data dengan menggunakan model *Entity Relationship Diagram,* yaitu :
2. **Super Key**

Satu atau lebih atribut yang dapat dibedakan setiap baris data dalam tabel secara unik

1. **Kandidat Key**

Merupakan kumpulan atribut minimal yang membedakan setiap baris data dalam table secara unik. Untuk bisa menjadi kandidat key, suatu atribut harus memenuhi persyaratan sebagai berikut :

1. Untuk satu nilai hanya mengidentifikasi satu baris dalam satu relasi (unik)
2. Tidak memiliki subset yang juga merupakan kunci relasi
3. Tidak dapat bernilai null
4. **Kunci Relasi (*Relation Key*) / Kunci Utama (*Primary Key*)**

Nilai dari kunci relasi harus mengidentifikasikan sebuah baris yang unik didalam sebuah relasi. Kunci relasi terdiri dari satu atau lebih atribut-atribut relasi. Agar bisa menjadi *primary key*, sebuah atribut harus lah memenuhi persyaratan sebagai kandidat key.

1. **Kunci Alternatif (*Alternatif Key*)**

Kunci yang tidak ada didunia nyata, tetapi diadakan dan dijadikan *primary ke*y. Kunci alternatif dibuat ketika tidak ada satupun atribut dalam sebuah relasi yang bisa mewakili relasi tersebut atau ada yang bisa menjadi kandidat key tetapi tidak cukup efektif untuk digunakan sebagai primary key.

1. **Komposit Key**

Komposit key merupakan primary key yang terdiri lebih dari satu atribut.

1. **Foreign Key (FK)**

Istilah FK juga banyak digunakan dalam perancangan. Sebuah Fk adalah sekumpulan atribut dalam suatu relasi ( missal A ) sedemikian sehingga kumpulan atribut ini bukan kunci relasi A tetapi merupakan kunci dari relasi lain.

1. **Kardinalitas Pemetaan**

Kardinalitas atau pemetaan atau rasio kardinalitas menunjukan jumlah entity yang dihubungkan ke satu entity lain dengan suatu relationship sets. Kardinalitas pemetaan meliputi :

1. Hubungan satu ke satu *(one to one*)

Yaitu satu entity dalam A dihubungkan dengan maksimum satu entity.

1. Hubungan satu ke banyak (*one to many*)

Yaitu satu entity dalam A dihubungkan dengan sejumlah entity dalam entity dalam B dihubungkan dengan maksimum satu entity dalam A.

1. Hubungan banyak ke satu (*many to one*)

Yaitu satu entity dalam A dihubungkan dalam maksimum satu entity B. satu entity dalam B dapat dihubungkan dengan sejumlah entity dalam A.

1. Hubungan banyak ke banyak (*many to many*)

Satu entity dalam A dihubungkan dengan sejumlah entity dalam entity dalam B dihubungkan dengan sejumlah entity dalam A.

1. **Derajat Relationship**

Menurut Fathansyah (2007:88) “ Derajat atau kardinalitas relasi itu mewakili hubungan (korespondesi) maksimum yang boleh terjadi antara himpunan entitas yang satu terhadap himpunan entitas yang lain”. Telah dijelaskan pula bahwa derajat relasi ini akan sangat diperlukan pada saat kita ingin mengimplementasikan sebuah rancangan basis data.

1. ***Logical Record Structured (LRS)***

Menurut Sinarmata (2007:12) “LRS dibentuk dengan nomor dari tipe record, beberapa tipe record digambarkan dengan kotak persegi panjang dan dengan nama yang unik”. LRS terdiri dari link-link diantara tipe record. Link ini menunjukan arah dari satu tipe record lainnya.

* + 1. **Pengujian Web**

Menurut Sukamto R.A dan Shalahuddin M (2015:272) “Pengujian adalah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan”. Aktifitas pengujian terdiri dari satu set atau sekumpulan langkah dimana dapat menempatkan desain kusus uji yang spesifik dan metode pengujian.

Metode pengujian yang digunakan yaitu *Black Box Testing* (pengujian kotak hitam). Menurut Sukamto R.A dan Shalahuddin M (2015:275) “*Black Box Testing* yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program”.

Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukkan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.