BAB II
LANDASAN TEORI

2.1 Perancangan Sistem

Menurut Al Fatta (2011), perancangan adalah :

> "suatu kegiatan untuk membentuk membuat sketsa struktur kegiatan atau pekerjaan dari suatu analisis ke dalam suatu perencanaan untuk dapat diterapkan dalam suatu bentuk nyata dapat disebut sebagai perancangan".

Menurut Rosa dan Shalahuddin (2014), perancangan adalah :

> "suatu upaya untuk mengonstruksi sebuah sistem yang memberikan kepuasan (mungkin informal) akan spesifikasi kebutuhan fungsional, memenuhi target, memenuhi kebutuhan secara *implisit* atau *eksplisit* dari segi performasi maupun penggunaan sumber daya, kepuasan batasan pada proses design dari segi biaya, waktu, dan perangkat disebut sebagai perancangan".

Jadi dapat disimpulkan bahwa perancangan adalah suatu kegiatan dalam memberikan penyampaian terkait hasil analisis yang akan di terapkan dengan sketsa perancangan dari sistem yang akan dibangun.

2.2 Mobile Application

Menurut Turban (2012, 277)*mobile application* yaitu istilah yang digunakan untuk mendeskripsikan aplikasi internet yang berjalan pada *smartphone* piranti*mobile* lainnya.

Mobile Application adalah sebuah aplikasi yang memungkinkan untuk melakukan mobilitas dengan menggunakan perlengkapan seperti Telepon seluler atau *Handphone*. Dengan menggunakan aplikasi *mobile*, maka dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, *browsing* dan lain sebagainya. Pemanfaatan aplikasi *mobile* untuk hiburan paling banyak digemari oleh hampir 70% pengguna telepon seluler, karena dengan memanfaatkan adanya fitur game, music *player*, sampai video player membuat kita menjadi semakin mudah menikmati hiburan kapan saja dan dimanapun.

2.3 Aplikasi

Istilah aplikasi pada dasarnya berasal dari bahasa inggris yaitu dari kata *application* yang berarti penerapan ataupun penggunaan.Jika ditinjau secara istilah, aplikasi berarti suatu program yang telah siap untuk digunakan.Aplikasi secara sengaja dibuat untuk melakukan suatu fungsi bagi pemakai jasa.Semua jenis aplikasi lainnya yang dipakai untuk sebuah sasaran yang dituju.

Aplikasi adalah alat bantu untuk mempermudah dan mempercepat proses pekerjaan dan bukan merupakan beban bagi penggunanya. (Ibisa, 2010)

Menurut Jogiyanto H.M. (2005), aplikasi adalah:

“Penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output”

2.4 Metode Pengumpulan Data

Dalam penelitiaan ini, peneliti menggunakan 3 teknik pengumpulan data, yaitu :

1. Wawancara

   Wawancara merupakan salah satu usaha atau percobaan secara sistematis, untuk mengumpulkan informasi yang kita butuhkan, yaitu dengan cara memberikan pertanyaan-pertanyaan kepada pihak yang menurut kita dapat dijadikan sumber data yang relevan dengan penelitian kita. Tujuan dari wawancara adalah untuk memperoleh informasi yang lebih akurat dan lengkap, untuk menyusun sistem yang baru agar sesuai dengan kebutuhan sistem organisasi.

2. Observasi

   Observasi merupakan pengumpulan data yang dilakukan dengan cara mendatangi langsung dan mempelajari sistem yang terjadi di Toko Moris Jaya secara nyata ditempat kejadian.

3. Studi Pustaka

   Studi pustaka yaitu salah satu teknik pengumpulan data yang dilakukan dengan cara membaca, mengutip, dan membuat serta mempelajari catatan, buku-buku dan literatur- literatur yang bersumber pada bahan-bahan pustaka yang mendukung dan dapat dijadikan bahan referensi terkait dengan penelitian.

2.5 Metodologi Pengembangan Sistem

Metodelogi pengembangan sistem pada penelitian ini menggunakan metode *Prototype*. Menurut Adi Nugroho (2011)metode *prototype* dibuat saat pengguna tidak tahu pasti apa yang mereka inginkan baik rincian masukannya, rincian proses dan rincian keluaran yang diinginkan untuk itu dibuatlah *Prototype* kepada pengguna. Kemudian pengguna menyarankan perbaikan-perbaikan jika terdapat kekurangan sistem yang perlu diperbaiki. Adapun tahapan-tahapannya adalah sebagai berikut :

1. Interaksi dengan pengguna

   Pada tahapan ini penyusun menganalisis apa yang ingin pengguna dapatkan dari sistem/perangkat lunak itu. Sehingga aplikasi yang dihasilkan sesuai dengan kebutuhan pengguna dan sistem.

2. Membuat *Prototype*

   Pada tahapan ini akan dibuat sebuah *Prototype* aplikasi berbasis *WAP* berdasarkan atas kebutuhan pengguna dan sistem pada tahap interaksi dengan pengguna.

3. Menguji *Prototype*

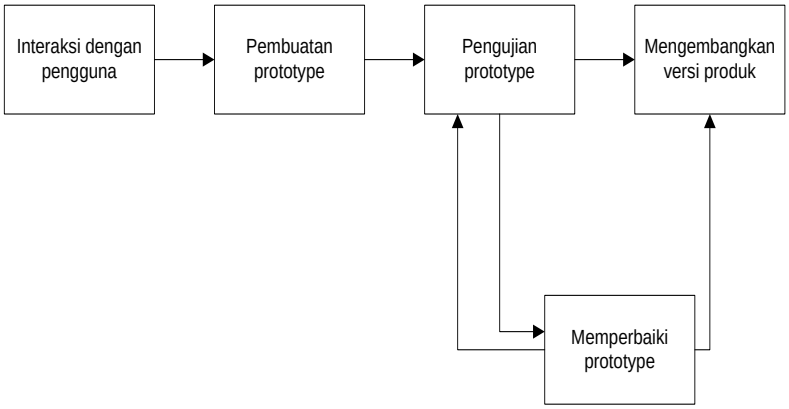
   Tahapan ini adalah proses penilaian terhadap *Prototype* yang telah dibuat apakah sesuai dengan kebutuhan atau tidak jika tidak maka *Prototype* akan diperbaiki.

4. Memperbaiki *Prototype*

Setelah ditemukan letak kesalahan dari *Prototype* yang dirancang pada tahapan ini penyusun akan membuat atau memperbaiki *Prototype* yang ada setelah itu akan di uji lagi sehingga *Prototype* sesuai dengan keinginan pengguna.

5. Mengembangkan Versi Produk setelah aplikasi dapat berjalan dan memenuhi kebutuhan sistem maka aplikasi ini siap dipakai.

Metode *Prototype* ini dapat digambarkan sebagai berikut :



Gambar 2.1 Metode *Prototype* (Adi Nugroho,2011: 78)

Menurut O'Brien (2011), *Prototype* memiliki tahapan tahapan sebagai berikut :

a. Identifikasi kebutuhan *end user*

Pada tahap ini para pemakai akhir mengidentifikasi kebutuhan bisnis mereka dan menilai kelayakan beberapa alternatif solusi sistem informasi. Pengguna sistem informasi dan vendor mengidentifikasikan semua kebutuhan, dan garis besar sistem yang akan dibuat beserta alternatif solusi sistem.

b. Membangun *prototype business system*

Pada tahap ini para pemakai akhir atau pakar sistem informasi menggunakan alat pengembangan aplikasi untuk secara interaktif mendesain dan menguji *prototype* berbagai komponen sistem informasi yang memenuhi kebutuhan para pemakai akhir. Membangun*prototyping* aplikasi pengembangan dengan membuat model sebagai uji coba yang mewakili kebutuhan pengguna secara garis besar.

c. Revisi *prototype* kedalam bentuk yang mendekati kebutuhan *end user*

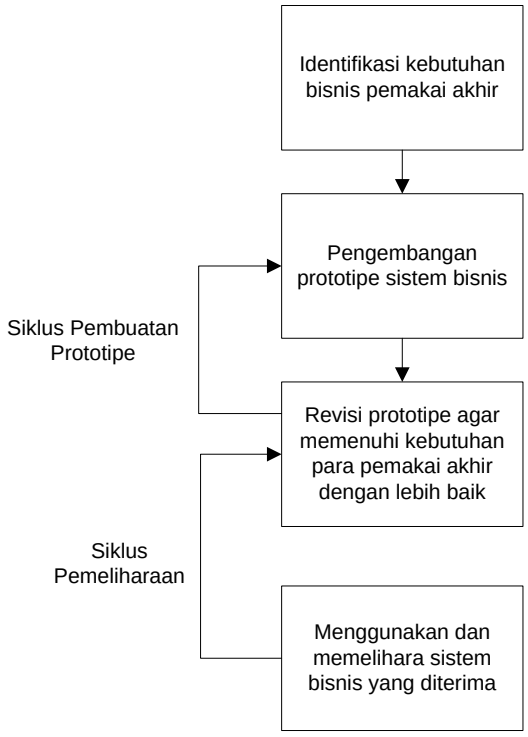
Model sistem bisnis diuji coba, dievaluasi dan dimodifikasi berulang-ulang hingga dapat diterima oleh pengguna dan dirasakan oleh pengguna telah sesuai dengan kebutuhan.

d. Menggunakan dan memelihara *business system* yang telah diterima

Dalam tahap ini sistem bisnis yang telah disepakati antara pengguna dan *vendor* dapat dimodifikasi dengan mudahkarena sebagian besar dokumentasi dari sistem telah tersimpan.

Model *prototype* dimulai dengan pengumpulan kebutuhan. Pengembang dan *user* bertemu dan mendefinisikan obyektif keseluruhan dari *software*, mengidentifikasi segala kebutuhan yang diketahui, dan area garis besar dimana definisi lebih jauh merupakan keharusan kemudian dilakukan perencanaan secara cepat, Sehingga *prototyping* dikenal dengan *Rapid Application Development* (RAD).

Skema tahapan pengembangan sistem informasi dengan pendekatan *prototyping* menurut O'Brien :



Gambar 1.1 Skema tahapan pengembangan sistem informasi dengan pendekatan *prototyping*

2.6 Metode Pengolahan Data

Pengertian dari pengolahan data atau data processing merupakan manipulasi data ke bentuk yang lebih informative atau berupa informasi.Informasi merupakan hasil dari kegiatan pengolahan suatu data dalam bentuk tertentu yang lebih berarti dari suatu kegiatan atau suatu peristiwa.

Terdapat empat penggolongan alat pengolahan data yang bisa anda ketahui diantaranya peralatan manual atau alat sederhana untuk mengolah data dengan factor terpenting adalah dalam penggunaan alat dengan tenaga tangan manusia.Selanjutnya adalah peralatan mekanik yaitu peralatan yang sudah lebih modern dan dalam bentuk mekanik dan digerakkan oleh tangan manual.

Peralatan berikutnya adalah dengan peralatan mekanik elektronik yang digerakkkan secara otomatis dengan motor elektronik.Dan alat yang terakhir adalah peralatan elektronik yang dikerjakan secara elektronik penuh tanpa bantuan tangan manusia.

2.6.1 Jenis data

Data menurut jenisnya ada dua yaitu data kualitatif dan data kuantitatif sebagai berikut:

1. Data kualitatif

   Data yang berhubungan dengan kategorisasi, karakteristik berwujud pertanyaan atau berupa kata-kata.Contonya wanita itu cantik, pria itu tampan, baik, buruk, rumah itu besar dan sebagainya. Data ini biasanya didapat dari wawancara yang bersifat subyektif sebab data tersebut ditapsirkan lain oleh

orang yang berbeda. Data kualitatif dapat diangkakan dalam bentuk ordinal atau rangking.

2. Data kuantitatif

   Data yang berwujud angka-angka.Contohnya ; yang diterima menjadi PNS 150 orang, penghasilan klinik bersalin 1 milyar/ bulan. Data ini diperoleh dari pengukuran langsung maupun dari angka-angka yang diperoleh dengan mengubah data kualitatif menjadi data kuantitatif.Data kuantitatif bersifat objektif dan bisa ditafsirkan oleh semua orang.

2.7 Pemodelan Berorientasi Objek

2.7.1 Pengertian Pemodelan Beriorentasi Objek

Pendekan berorientasi objek merupakan adalah suatu teknik atau cara pendekatan dalam melihat permasalahan dan sistem (sistem perangkat lunak, sistem informasi atau sistem lainnya). Sisitem berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan objek – objek dunia nyata Dalam rekayasa perangkat lunak, konsep pendekatan berorientasi objek dapat diterapkan pada tahapan analisis, perancangan, pemrograman dan pengujian perangkat lunak. Ada bergbagai teknik yang digunakan pada masing – masing tahapan tersebut, dengan aturan dan pemodelan tertentu.Sistem berorientasi objek merupakan sebuah sistem yang dibangun dengan berdasarkan metode berorientasi objek.Metode berorientasi objek adalah sebuah sistem yang komponennya dibungkus (diendkapsulisasi) menjadi kelompok data dan fungsi. Setiap komponen dalam sistem dapat mewarisi sifat atribut dan sifat

dari komponen lainnya dan dapat berinteraksi satu sama lainnya. Rosa A.S. dan M. Shalahuddin dalam bukunya Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek (2013:103)

2.7.2 Ciri – Ciri Pemodelan Berorientasi Objek

1. **Kelas (class)**

   Kelas adalah kumpulan objek – objek dengan karakteristik yang sama. Sebuah kelas akan mempunyai sifat (atribut), kelakuan (operasi atau meode), hubungan (relationship) dan arti. Secara teknis, kelas adalah sebuah stuktur tertentu dalam pembuatan peragkat lunak.Kelas merupakan bentuk struktur pada kode pemrogram yang merupakan metodologi berorientasi objek. Kelas secara fisik adalah berkas atau file yang berisi kode program, dimana kode program merupakan semua hal yang terkait dengan nama kelas. Rosa A.S. dan M. Shalahuddin (2013:104)

2. **Objek (object)**

   Objek adalah suatu yang mewakili dunia nyata seperti benda, manusia,tempat, struktur, status atau hal lainnya yan abstrak. Ojek murupakan suatu entitas yang mampu menyimpan informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan atau berpengaruh pada suatu objeknya.Objek mempunyai siklus hidup yaitu diciptakan, dimanpulasi, dan dihancurkan.

   Secara teknis, sebuah kelas saat program dieksekusi maka akan dibuat sebuah objek. Objek dilihat dari segi teknis elemen pada saat runtime yang akan diciptakan, dimanipulasi dan dihancurkan saat dieksekusi maka sebuah objek

hanya ada pada saat sebuah program di eksekusi. Rosa A.S. dan M. Shalahuddin (2013:106).

3. **Metode (method)**

   Operasi atau metode merupakan fungsi atau transformasi yang dapat dilakukan terhadap objek atau dilakukan oleh objek.Metode berasala dari event, aktivitas atau aksi keadaan, fungsi atau kelakuan dunia nyata. Contoh meode misalnya read, write, move, copy dan sebagainya. Kelas sebaiknya memiliki metode get dan set untuk setiap atribut agar konsep enkapsulasi tetap terjaga. Metode get digunakan untuk memberi akses kelas lain dalam mengakses atribut, dan set adalah metode yang digunakan untuk mengisi atribut, agar kelas lain tidak mengakses atribut secara langsung. Rosa A.S. dan M. Shalahuddin (2013:107)

4. **Atribut (attribute)**

   Atribut dari sebuah kelas adalah variabel global yang dimiliki sebuah kelas.Atribut dapat berupa nilai atau elemen – elemen data yang dimiliki oleh objek dalam kelas objek.Atribut dimiliki secara individu oleh sebuah objek.Atribut sebaiknya bersifat private untuk menjaga konsep enkapsulasi. Rosa A.S. dan M. Shalahuddin (2013:108)

5. **Enkapsulasi (encapsulation)**

   Pembungkusan atribut data dan pelayanan (perasi-operasi) yang dipunyai oleh objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerjanya. Rosa A.S. dan M. Shalahuddin (2013:108)

6. **Antarmuka (interface)**

Antarmuka atau interface sangat mirip dengan kelas, tapi tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa isi. Deklarasi metode pada sebuah interface dapat diimplementasikan oleh kelas lain. Metode pada antar muka yang diimplementasikan pada suatu kelas harus sama persis dengan yang ada pada antarmuka. Antarmuka atau interface biasanya digunakan agar kelas yang lain tidak mengakses angsung ke suatu kelas tapi mengakses antarmukanya. Rosa A.S. dan M. Shalahuddin (2013:109)

7. Package

Package adalah sebuah kontainer atau kemasan yang dapat digunakan untuk mengelompokan kelas – kelas sehingga memungkinkan beberapa kelas yang bernama sama disimpan dalam package yang berbeda. Rosa A.S. dan M. Shalahuddin (2013:110)

2.7.3 UML (*Unified Modeling Language*)

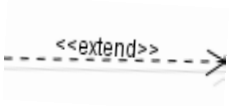
Menurut Rosa & Shalahuddin (2013:137) UML adalah:*Unified Modeling Language (UML)* merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan.Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

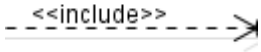
2.7.3.1 *Use Case Diagram*

Merupakan pemodelanuntuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikansebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.secara kasar *use case* digunakan

untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. (Rosa A.S dan M. Shalahuddin, 2013)

Tabel 2.1 Simbol *Usecase Diagram*(Rosa A.S dan M. Shalahuddin, 2013)

No	Simbol	Keterangan
1.	*UseCase*	Fungsionalitas yang disediakan sistem sebagai unit-ubit yang saling bertukar pesan antara unit atau aktor.
2.	*Actor*	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, walaupun simbol aktor adalah orang namun aktor belum tentu merupakan orang. Biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3.	*Association*	Komunikasi antara aktor dan use case yang berpartisipasi pada usecase atau use case memiliki interaksi dengan aktor.
4.	Ekstensi/ *Extend* 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walaupun tanpa use case tambahan itu.
5.	*Generalization*	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi

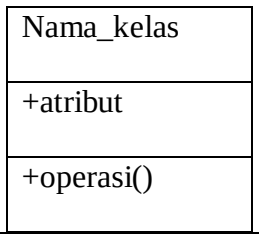
		yang satu adalah fungsi yang lebih umum dari lainnya.
6.	*Include* 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini.

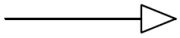
2.7.3.2 *Class Diagram*

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yangakan dibuat untuk membangun sistem.Kelas memilikiapa yang disebut atribut dan metode atau operasi.(Rosa A.S dan M. Shalahuddin, 2013).

Sebuah kelas diagram terdiri dari sejumlah kelas yang dihubungkan dengan garis yang menunjukan hubungan antar kelas.

Tabel 2.2 Simbol *Class Diagram*(Rosa A.S dan M. Shalahuddin, 2013)

Simbol	**Keterangan**
Kelas/Class 	Kelas pada struktur system
Antar muka/*interface* 	Sama dengan konsep *interface* dalam pemograman berorientasi objek

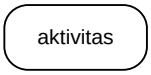
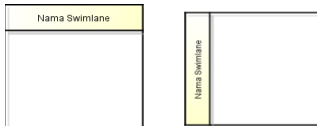
Simbol	Keterangan
Asosiasi/*association* 	Relasi antar kelas dengan makna umum ,asosiasi biasanya juga disertai dengan *multiplicity*.
Asosiasi berarah/*directed* 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity*.
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan atau *Dependency* 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi/*aggregation* 	Relasi antar kelas dengan makna semua-bagian(whole-part)

2.7.3.3 *Activity Diagram*

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan kator,jadi aktivitas yang dapat dilakukan oleh sistem. (Rosa A.S dan M. Shalahuddin, 2013)

Berikut ini adalah simbol-simbol yang ada pada diagram aktivitas:

Tabel 2.3 Simbol *Activity Diagram*(Rosa A.S dan M. Shalahuddin, 2013)

Simbol	Keterangan
Status awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ *decision*	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan/ *join*	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir	Status akhir yang dilakukan sistem,sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

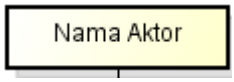
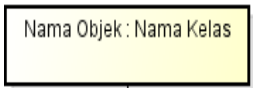
2.7.3.4 *Sequence Diagram*

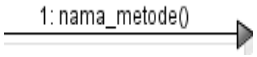
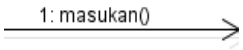
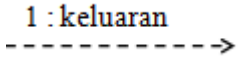
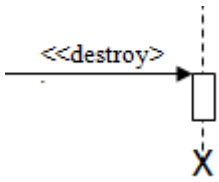
Diagram sequence menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek atau message yang dikirimkan dan diterima

antar objek.oleh karena itu untuk menggambarkan diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *usecase* berserta metode-metode yang dimiliki kelas. (Rosa A.S dan M. Shalahuddin, 2013)

Berikut ini adalah simbol-simbol yang ada pada diagram aktivitas:

Tabel 2.4 Simbol*Sequence Diagram*(Rosa A.S dan M. Shalahuddin, 2013)

Simbol	Keterangan
Aktor atau  Tanpa waktu aktif	Orang,proses,atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang;biasanya dinyatakan menggunakan kata benda diawal *frase* nama aktor.
Garis hidup 	Menyatakan kehidupan suatu objek
Objek  	Menyatukan objek yang berinteraksi pesan
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.

Simbol	Keterangan
Pesan tipe *create* <<*create*>>	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
Pesan tipe *call* 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
Pesan tipe *send* 	Menyatakan bahwa suatu objek mengirim data /masukan/informasi ke objek lainnya , arah panah mengarah pada objek yang dikirim.
Pesan tipe *return* 	Menyatakan bahwa suatu objek yang telah menjalankan sesuatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu.
Pesan Tipe *Destroy* 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada *create* maka ada *destroy*.

2.8 Alat Implementasi Sistem

2.8.1 *MySQL*

(Loka Dwirtara) MySQL bersifat RDBMS (Relational Database Management Sistem) yang memungkinkan seorang admin dapat menyimpan banyak informasi ke tael-tabel , dimana tabel-tabel tersebut saling berkaitan satu sama lain. Keuntungan RDBMS sendiri adalah kita dapat memecah database ke dalam tabel-

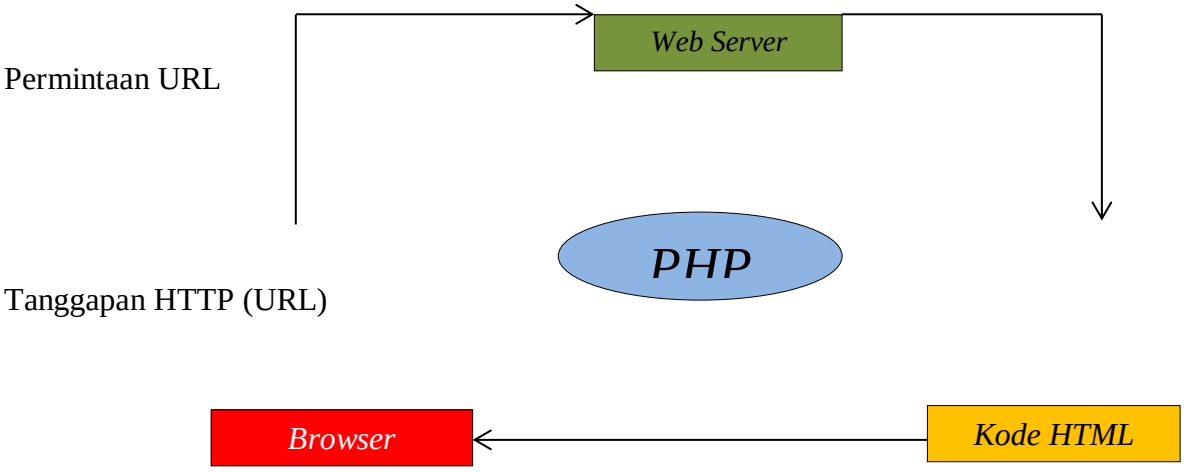
tabel yang berbeda, setiap tabel memiliki informasi yang berkaitan dengan tabel yang lainya.

MySQL adalah *multi userdatabase* yang menggunakan bahasa *Structured Query Language* (SQL). MySQL dalam operasi *client-server*melibatkan server daemon MySQL di sisi *server*dan sebagai macam program serta library yang berjalan di sisi client MySQL mampu menangani data yag cukup besar.

2.9 PHP

(Rohi Abdulloh,2015) PHP singkatan *Hyper Text Preprocessor* yang merupakan *server-side programming* ,yaitu Bahasa pemograman yang di proses di sisi server. Fungsi utama PHP dalam membangun website adalah untuk melakukan pengolahan data pada database. Data website akan dimasukan ke database, diedit, dihapus, dan ditampilkan pada website yang diatur oleh PHP. Pengembangan demi pengembangan terus berlanjut , ratusan fungsi ditambahkan sebagai fitur dari bahasa PHP, dan di awal tahun 1999, netcraft mencatat, ditemukan 1.000.000 situs di dunia telah menggunakan PHP. Ini membuktikan bahwa PHP merupakan bahasa yang paling populer di gunakan oleh dunia web development. Hal ini mengagetkan para developernya termaksud rumus sendiri, dan tentunya sangat diluar dugaan sang pembuatnya . Kemudian Zeev Suraski dan Andi Gutsman selaku *core depelover* (programer inti) mencoba untuk menulis ulang PHP. Dan diintegrasikan dengan meggunakan Zend scripting engine, dan mengubah jalan alur operasi PHP. Dan semua fitur baru tersebut di rilis dalam PHP 4.13 juli 2004 evolusi PHP , PHP telah megalami banyak sekali perbaikan di segala sisi . dan wajar jika netcraft

mengumumkan PHP sebagai bahasa web populer didunia , karena tercatat 19 juta domain telah menggunakan PHP sebagai server side scriptingnya.



Gambar **2.2**Proses kerja *PHP*

2.10 Web

Yang dimaksud dengan aplikasi *web* atau aplikasi berbasis *web* adalah aplikasi yang dijalankan melalui *browser*. Aplikasi seperti ini pertama kali dibangun hanya dengan menggunakan bahasa yang disebut dengan HTML (*HyperText Markup Language*) dan protokol yang digunakan dinamakan HTTP (*HyperText Transfer Protokol*).

Menurut Oktavian (2010:13)

"HTML adalah suratu bahasa yang dikenali oleh *web browser* untuk menampilkan informasi dengan lebih menarik dibandingkan dengan tulisan text biasa *(plain text)."*

HTML merupakan singkatan dari *HyperTextMarkup Language* adalah script untuk menyusun dokumen-dokumen *Web*. Dokumen HTML disimpan dalam

format teks reguler dan mengandung tag-tag yang memerintahkan *webbrowser* untuk mengeksekusi perintah-perintah yang dispesifikasikan

Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML. Pada saat ini, banyak skrip seperti itu antara lain yaitu PHP, ASP, ASP.NET sedangkan contoh yang berupa objek antara lain adalah *applet (Java)*.

2.11 Xampp

Fungsi Xampp sendiri adalah sebagai server yang berdiri sendiri *(localhost)*, yang terdiri beberapa program antara lain: *Apache HTTP Server*, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama Xampp sendiri merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah untuk digunakan yang dapat menampilkan halaman web yang dinamis. Untuk mendapatkannya Xampp anda dapat mendownload langsung dari web resminya**.**