

BAB 2

TINJAUAN PUSTAKA

2.1. Website

Suatu Web adalah salah satu aplikasi yang berisikan dokumen–dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan dan untuk mengakses menggunakan perangkat lunak yang disebut browser [1]. Untuk membuka sebuah website maka pengguna harus memiliki perangkat (komputer, smartphone) yang terkoneksi dengan internet atau intranet. Halaman website atau web umumnya berbentuk dokumen dalam format Hyper Text Markup Language (HTML), yang dapat diakses melalui HTTP atau HTTPS, suatu protokol yang menyampaikan berbagai informasi dari server website untuk ditampilkan kepada para user atau pemakai melalui web browser [2].

Sebuah website memiliki alamat URL yang unik dan spesifik yang disebut dengan domain. Misalnya domain Maxmanroe.com, Google.com, Facebook.com, dan lain-lain. Website dapat diakses dengan menggunakan browser dan koneksi internet. Namun, ada beberapa website yang bisa diakses menggunakan jaringan lokal (LAN). Dan adapun jenis-jenis [2]

2.1.1 Website Statis

Bersifat statis apabila isi informasi website tetap, jarang berubah, dan isi informasinya searah hanya dari pemilik website. Bersifat dinamis apabila isi informasi website selalu berubah-ubah, dan isi informasinya interaktif dua arah berasal dari pemilik serta pengguna website. Contoh website statis adalah berisi profil perusahaan, sedangkan website dinamis adalah seperti Friendster, Multiply, dll. Dalam sisi pengembangannya, website statis hanya bisa diupdate oleh pemiliknya saja, sedangkan website dinamis bisa diupdate oleh pengguna maupun pemilik [3].

2.1.2 Website Dinamis

Website dinamis adalah jenis website yang dirancang khusus untuk dapat menampilkan update konten sesering mungkin. Website dinamis dapat disesuaikan dengan kebutuhan, baik dari sisi tampilannya maupun dari sisi fiturnya. Website dinamis umumnya dirancang dengan konsep visual dan kemampuan interaksi tinggi dengan pengguna. Beberapa fitur yang bisa ditambahkan pada website dinamis yaitu, kolom komentar, fitur live chatting, formulir pendaftaran, dan lain-lain [2]. Beberapa contoh website dinamis:

1. Blog/ website pribadi
2. Situs E-commerce
3. Website Portal
4. Situs Berita
5. Katalog online
6. Dan lain-lain

2.1.3 Website Interaktif

Pengertian website interaktif adalah jenis website yang digunakan untuk tujuan berinteraksi dengan orang lain secara online. Umumnya pengguna website interaktif adalah komunitas atau pengguna internet aktif. Beberapa contoh website interaktif:

1. Situs Media sosial
2. Situs forum online
3. Blog

2.2. Web Browser

Web browser adalah suatu program, software, aplikasi atau perangkat yang digunakan untuk mencari atau menjelajahi informasi di dalam jaringan internet dari sebuah media seperti situs, blog, jejaring sosial, dan lain sebagainya yang tersimpan di dalam internet. Tentunya kamu tidak asing lagi dengan nama-nama web browser seperti Chrome, Firefox, Opera, Safari dan lain-lain. Merupakan sebuah aplikasi atau perangkat lunak yang digunakan mengambil, menyajikan, dan melintasi berbagai sumber informasi yang ada didalam jaringan internet (world wide web). Sumber-sumber informasi yang dimaksud itu terdiri dari berbagai macam bentuk seperti misalnya berupa halaman situs, gambar, video, infografis, konten, dan yang lainnya [2].

2.3. Dreamweaver

Adobe Dreamweaver adalah program yang digunakan untuk membuat atau menyunting halaman web. Software Dreamweaver dikeluarkan oleh Adobe System. Aplikasi ini banyak digunakan oleh para programmer, desainer dan developer web dikarenakan kemudahan dalam penggunaannya, kelengkapan fiturnya dan juga dukungannya terhadap teknologi terkini.

Adobe Dreamweaver menyediakan fitur editor WYSIWYG (What You See is What You Get) atau dalam bahasa kesehariannya disebut Design View. Maksudnya adalah, tampilan hasil akhir web kita nanti akan sama dengan tampilan pada saat proses perancangan halaman web.

Dengan segala fitur yang ada pada Adobe Dreamweaver, membuat suatu web bukanlah hal yang sulit. Kita tidak perlu menguasai berbagai macam bahasa pemrograman web seperti HTML, CSS, Javascript, PHP, dan sebagainya. Cukup mengetahui dasar dasarnya saja, karena didalam aplikasi ini sudah disediakan alat alat otomatis.

Selain itu, aplikasi ini juga menyediakan 3 macam tampilan yaitu Code View, Design View dan Split View. Code View cocok untuk para programmer yang terbiasa dengan kode kode pemrograman web. Sedangkan Design View cocok

untuk para Designer yang terbiasa dengan visual. Jika ingin menggunakan keduanya, bisa memilih Split View [2].

2.4. Pemrograman Berorientasi Objek

Pendekatan berorientasi objek merupakan suatu teknik atau cara pendekatan dalam melihat permasalahan dan sistem (sistem perangkat lunak, sistem informasi, atau sistem lainnya). Pendekatan berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan objek-objek dunia nyata. Ada banyak cara untuk mengabstraksikan dan memodelkan objek-objek tersebut, mulai dari abstraksi objek, kelas, hubungan antar kelas sampai abstraksi sistem. Saat mengabstraksikan dan memodelkan objek ini, data dan proses-proses yang dimiliki oleh objek akan dienkapsulasi (dibungkus) menjadi satu kesatuan [4].

Dalam rekayasa perangkat lunak, konsep pendekatan berorientasi objek dapat diterapkan pada tahap analisis, perancangan, pemrograman, dan pengujian perangkat lunak. Ada berbagai teknik yang dapat digunakan pada masing-masing tahap tersebut, dengan aturan dan alat bantu pemodelan tertentu [4].

Sistem berorientasi objek merupakan sebuah sistem yang komponennya dibungkus (dienkapsulasi) menjadi kelompok data dan fungsi. Setiap komponen dalam sistem tersebut dapat mewarisi atribut, sifat, dan komponen lainnya yang dapat berinteraksi satu sama lain. Terdapat beberapa konsep utama pada metodologi berorientasi objek, diantaranya [4]:

- 1) Kelas (*class*), kumpulan objek-objek dengan karakteristik yang sama. Kelas merupakan definisi statik dari himpunan objek yang sama yang mungkin lahir atau diciptakan dari kelas tersebut. Sebuah kelas akan mempunyai sifat (atribut), kelakuan (operasi/metode), hubungan (*relationship*), dan arti. Suatu kelas dapat diturunkan dari kelas yang lain, dimana atribut dari kelas semula dapat diwariskan ke kelas yang baru.
- 2) Objek (*object*), abstraksi dari sesuatu yang mewakili dunia nyata seperti benda, manusia, satuan organisasi, tempat, kejadian, struktur, status, atau hal-hal lain yang bersifat abstrak. Objek merupakan suatu entitas yang mampu menyimpan

informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan atau dapat berpengaruh pada status objeknya. Objek mempunyai siklus hidup yaitu diciptakan, dimanipulasi, dan dihancurkan.

- 3) Abstraksi (*abstraction*), prinsip untuk merepresentasikan dunia nyata yang kompleks menjadi suatu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.
- 4) Enkapsulasi (*encapsulation*), pembungkusan atribut data dan layanan (operasi-operasi) yang dimiliki objek untuk menyembunyikan implementasi dari objek sehingga objek lain tidak mengetahui cara kerjanya.
- 5) Pewarisan (*inheritance*), mekanisme yang memungkinkan suatu objek mewarisi sebagian atau seluruh definisi dan objek lain sebagai bagian dari dirinya.
- 6) Polimorfisme (*polymorphism*), kemampuan suatu objek untuk digunakan dibanyak tujuan yang berbeda dengan nama yang sama sehingga menghemat baris program.

2.5 *Unified Modelling Language (UML)*

Unified Modeling Language (UML) adalah bahasa yang digunakan untuk menspesifikasikan, memvisualisasikan, dan mendokumentasikan artifak dari sebuah sistem berorientasi objek yang dalam tahap pengembangan. UML menyediakan standar defacto dalam analisis sistem berorientasi objek dan desain yang didasarkan dari pengalaman pengguna .

UML menawarkan sebuah standar untuk merancang model sebuah system. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi, dimana aplikasi tersebut dapat berjalan pada piranti keras, *system* operasi dan jaringan apapun, sertra ditulis dalam bahasa pemrograman apapun [4].

2.5.1 *Use Case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. Di dalam *Use Case* terdapat aktor. Seorang / sebuah aktor adalah

sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. Elemen-elemen yang ada pada diagram use case adalah aktor, use case, dan hubungan ketergantungan, generalisasi dan asosiasi [4].

a. Aktor

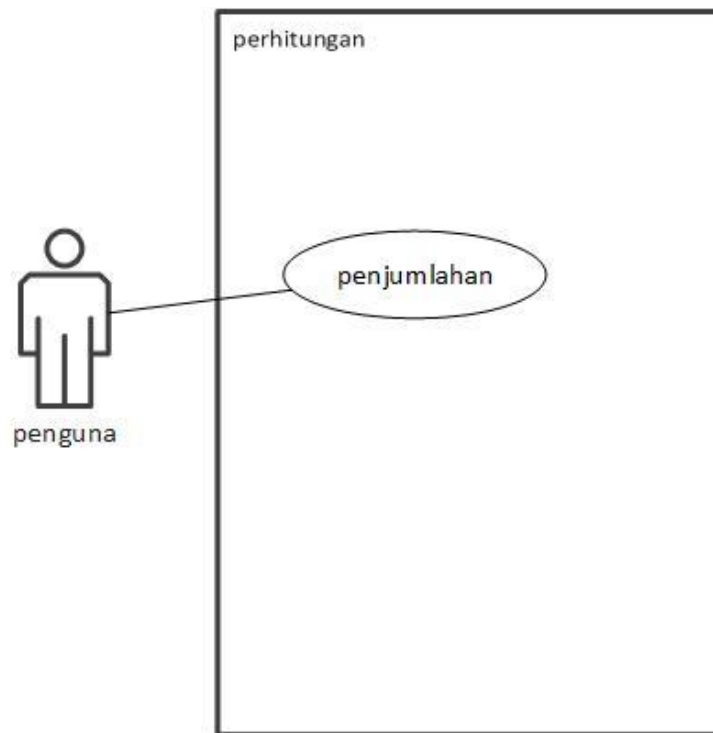
Aktor adalah pemakai sistem, dapat berupa manusia atau sistem terotomatisasi lain. Aktor adalah sesuatu atau seseorang yang berinteraksi dengan sistem, yaitu siapa atau apa yang menggunakan sistem. Aktor berkomunikasi dengan sistem lewat pengiriman dan penerimaan pesan. Use case selalu diawali oleh aktor yang mengirim pesan.

b. Use Case

Use case adalah cara spesifikasi penggunaan sistem oleh aktor. Use case menspesifikasikan perilaku sistem atau bagian sistem dan merupakan deskripsi sekumpulan sekuen aksi termasuk varian-varian yang dilakukan sistem untuk memproduksi hasil atau nilai ke aktor.

c. Hubungan antar Use Case

Keterhubungan antar *use case* dengan *use case* lain berupa generalisasi antara *use case*, yaitu *include* dan *extend*. *Extend* adalah relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu. *Include* adalah relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya.



Gambar 2.1 Contoh Use Case Diagram

2.5.2 Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain.

Class memiliki 3 area pokok [4]:

1. Nama memiliki fungsi mempresentasikan identitas dari sebuah *class*.
2. Atribut memiliki fungsi mempresentasikan atribut-atribut yang ada di dalam suatu *class*.
3. Metoda memiliki fungsi mempresentasikan metode-metode yang ada pada suatu *class*.

Atribut dan metoda dapat memiliki salah satu sifat berikut:

1. *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan.

2. *Protected*, hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya.
3. *Public*, dapat dipanggil oleh siapa saja.

Setiap *class* pasti memiliki hubungan dengan *class* lain. Terdapat 4 cara menghubungkan satu *class* dengan *class* lain tergantung hubungan *class – class* tersebut. Hubungan tersebut direpresentasikan dengan sebuah panah diantara *class* tersebut. Adapun 4 hubungan antar *class* pada *class diagram* sebagai berikut [4]:

1. Asosiasi

Asosiasi adalah hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain. Panah navigability menunjukkan arah query antar class.

2. Agregasi

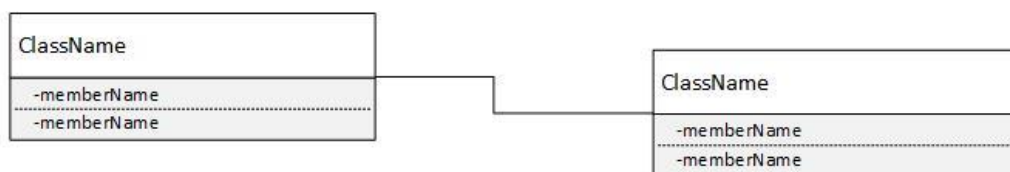
Agregasi adalah hubungan yang menyatakan bagian “terdiri atas”.

3. Pewarisan

Pewarisan adalah hubungan hirarkis antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru sehingga ia disebut anak dari class yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.

4. Hubungan dinamis

Hubungan dinamis adalah rangkaian pesan yang di passing dari satu class kepada class lain. Hubungan dinamsi dapat digambarkan dengan menggunakan sequence diagram.

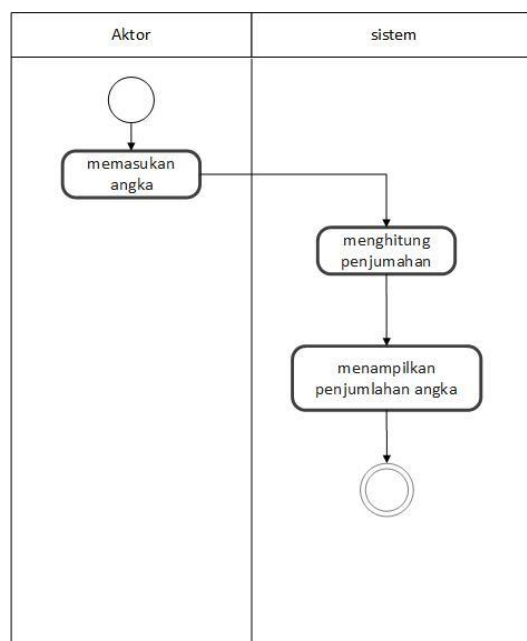


Gambar 2.2 Contoh Class Diagram

2.5.3 Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi [4].

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas [4].



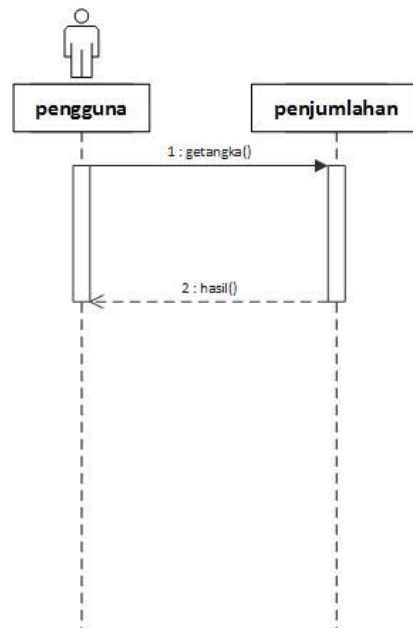
Gambar 2.3 Contoh Activity Diagram

2.5.4 Sequence Diagram

Sequence diagram dapat menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait) [4].

Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang

dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class* [4].



Gambar 2.4 Contoh Sequence Diagram

2.6. Instagram

Instagram adalah sosial media berbasis gambar yang memberikan layanan berbagi foto atau video secara online. Instagram berasal dari pengertian dari keseluruhan fungsi aplikasi ini. Kata "insta" berasal dari kata "instan", seperti kamera polaroid yang pada masanya lebih dikenal dengan sebutan "foto instan". Instagram juga dapat menampilkan foto-foto secara instan, seperti polaroid di dalam tampilannya. Sedangkan untuk kata "gram" berasal dari kata "telegram" yang cara kerjanya untuk mengirimkan informasi kepada orang lain dengan cepat. Sama halnya dengan Instagram yang dapat mengunggah foto dengan menggunakan jaringan Internet, sehingga informasi yang ingin disampaikan dapat diterima dengan cepat. Oleh karena itulah Instagram merupakan lakuran dari kata instan dan telegram.

Instagram adalah sebuah aplikasi media sosial yang memungkinkan pengguna untuk mengambil foto dan video. Foto atau video yang dibagikan nantinya akan terpampang di Feed pengguna lain yang menjadi teman Anda. Pada Instagram sendiri menerapkan sistem pertemanan yang menggunakan istilah following dan follower. Following berarti Anda mengikuti pengguna, sedangkan Follower berarti pengguna lain yang mengikuti Anda. Selanjutnya setiap pengguna dapat berinteraksi dengan cara memberikan komentar dan memberikan respon suka terhadap foto yang dibagikan dengan menggunakan simbol berupa love. Di Indonesia sendiri Instagram telah menjadi salah satu aplikasi dengan pengunduh terbanyak mulai dari kalangan masyarakat biasa, pejabat, artis, hingga sampai kepada kalangan civitas akademika yaitu mahasiswa sendiri. Karena kepopulerannya itulah sehingga Instagram mempunyai peran terhadap gaya hidup penggunanya [5].

Setelah diluncurkan pada tahun 2010, Instagram dengan cepat mendapatkan popularitas, dengan satu juta pengguna terdaftar dalam dua bulan, 10 juta dalam setahun dan akhirnya 800 juta pada September 2017. Pada April 2012, Facebook memperoleh keuntungan sekitar US 1 miliar dolar Hingga Oktober 2015, lebih dari 40 miliar foto telah diunggah ke layanan. Instagram dapat digunakan pada smartphone, iPhone, iPad atau iPod Touch versi apapun dengan sistem operasi iOS

2.7.API (Application Programming Interface)

Pengertian API (Application Programming Interface) Antarmuka pemrograman aplikasi (Application Programming Interface/API) adalah sekumpulan perintah, fungsi, dan protocol yang dapat digunakan oleh programmer saat membangun perangkat lunak untuk system operasi tertentu. API memungkinkan programmer untuk menggunakan fungsi standar untuk berinteraksi dengan system operasi.

API dapat menjelaskan cara sebuah tugas (task) tertentu dilakukan. Dalam pemrograman procedural seperti bahasa C, aksi biasanya dilakukan dengan media pemanggilan fungsi. Karena itu, API biasanya menyertakan penjelasan dari fungsi/rutin yang disediakannya.

API menyediakan fungsi dan perintah dengan bahasa yang lebih terstruktur dan lebih mudah untuk dipahami oleh programmer bila dibandingkan dengan System Calls, hal ini penting untuk aspek editing dan pengembangan, sehingga programmer dapat mengembangkan sistem dengan mudah. API juga dapat digunakan pada Sistem Operasi mana saja asalkan sudah ada paket-paket API nya.

Dalam contoh program sederhana, dibutuhkan setidaknya ribuan system calls per detik. Oleh karena itu Kebanyakan programmer membuat aplikasi dengan menggunakan Application Programming Interface(API). Dalam API itu terdapat fungsi-fungsi/perintah-perintah untuk menggantikan bahasa yang digunakan dalam system calls dengan bahasa yang lebih terstruktur dan mudah dimengerti oleh programmer. Dengan API, siapa pun sekarang dapat mengembangkan aplikasi mereka sendiri menggunakan data yang disimpan [6].

2.7.1 *API Instagram*

Instagram sekarang sudah menjadi salah satu platform web paling populer dalam kategori berbagi foto dan video kesayangan. Untuk membantu pengguna mengintegrasikan jaringan ini jauh dalam kehidupan sehari-hari, instagram telah merilis sebuah aplikasi publik biasa disebut Application Programming Interface (API). memungkinkan pengguna untuk mendapatkan informasi profil dasar, foto, dan video di akun instagram. API ini ditujukan untuk pengguna Instagram non-Bisnis dan non-Pencipta. Jika Anda ingin membuat aplikasi yang memungkinkan pengguna untuk mempublikasikan media, memoderasi komentar, mengidentifikasi @mentioned dan media hashtag, atau mendapatkan data tentang pengguna Instagram lainnya [7].

2.8. *JSON (Javascript Object Notation)*

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun

karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dan lain-lain. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data. Kelebihan format data JSON adalah lebih unggul dibandingkan XML mulai dari kecepatan, penulisan yang lebih gampang dan coding untuk parsing yang lebih ringkas dan sederhana [8].

2.9. Webservice

Web service dapat digambarkan sebagai sebuah sistem yang dirancang untuk dapat mendukung interaksi komunikasi antar mesinmesin pada suatu jaringan. Teknologi web service memungkinkan pengguna untuk dapat menghubungkan berbagai jenis software yang memiliki platform dan sistem operasi yang sama atau berbeda sekalipun. Jadi, pengguna tetap mendapatkan sebuah potongan informasi dari suatu website tanpa harus mengunjungi website tersebut, cukup dengan mengetahui fungsi/method web service yang disediakan oleh web itu. Pengguna dapat mengakses potongan informasi itu dengan meletakkan fungsi/method itu pada sebuah aplikasi [9].

2.10. Codeigniter

CodeIgniter Sebuah framework php yang bersifat open source dan menggunakan metode MVC (Model, View, Controller) untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal. Dalam situs resmi codeigniter, (Official Website CodeIgniter,2002) menyebutkan bahwa codeigniter merupakan framework PHP yang kuat dan sedikit bug. Codeigniter ini dibangun untuk para pengembang dengan bahasa pemrogram PHP yang membutuhkan alat untuk membuat web dengan fitur lengkap.

Framework Codeigniter dikembangkan oleh Rick Ellis, CEO Ellislab, Inc. kelebihan dari framework codeigniter jika dibandingkan dengan framework lain adalah sebagai berikut [10] :

1. Gratis (Open-Source)

Kerangka kerja Codeigniter memiliki lisensi dibawah Apache/BSD open-source sehingga bersifat bebas atau gratis.

2. Berukuran kecil

Ukuran yang kecil merupakan keunggulan tersendiri jika dibandingkan framework lain yang berukuran besar dan membutuhkan resource yang besar dan juga dalam eksekusi maupun penyimpanannya.

3. Menggunakan konsep M-V-C

Codeigniter merupakan konsep M-V-C (Model-View-Controller) yang memungkinkan pemisahan antara layer application-logic dan presentation. Dengan konsep ini kode PHP, query Mysql, Javascript dan CSS dapat saling dipisah-pisahkan sehingga ukuran file menjadi lebih kecil dan lebih mudah dalam perbaikan kedepannya atau maintenance.

