

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

(Jati, 2013) membuat aplikasi penjadwalan dengan judul “Jadwal Matakuliah Berbasis *Web* dan *Android*” yang dilakukan oleh sistem. Tujuannya adalah mempermudah mahasiswa untuk mendapatkan informasi jadwal mata kuliah dan pembuatan aplikasi *android* berupa data data jadwal mata kuliah yang sederhana dibuat dengan *MySQL* yang dikoneksikan dengan *web server* dan *web browser* sebagai *compliernya*. Pada tugas akhir ini dikembangkan aplikasi penjadwalan mata kuliah berbasis *web* dan *android* dan pengguna yang menggunakan perangkat *handphone* sistem operasi *android* dapat terbantu untuk mengakses informasi yang diberikan oleh aplikasi ini dan yang tidak memiliki *handphone* sistem informasi dapat mengakses melalui *web browser*.

(Manurung, 2013) adalah mahasiswa Politeknik Negeri Medan Program Studi Manajemen Informatika Jurusan Teknik Elektro yang melakukan penelitian di Politeknik Negeri Medan dengan judul “Sistem Informasi Jadwal Kuliah Berbasis Web Pada Jurusan Teknik Elektro Politeknik Negeri Medan”. Kekurangan dari Sistem ini adalah setiap dosen harus bersedia mengajar kapan saja dan di ruang manapun.

(Iswanto, 2012) membuat sebuah sistem dalam tugas akhir yang berjudul “Sistem Informasi Pengumuman Perkuliahan Mahasiswa Berbasis *Web*”. Dimana sistem informasi tersebut dapat membantu untuk mengelola pengumuman

perkuliahan dan menampilkan informasi tersebut dilayar monitor. Dalam pembuatan sitem informasi ini menggunakan lima kekurangan dari sistem informasi ini selain menampilkan pengumuman hanya secara umum dan juga mahasiswa harus datang ke kampus untuk mengetahui pengumuman perkuliahan.

Berdasarkan pada tiga penelitian di atas peneliti akan membangun suatu sistem aplikasi berbasis *web service* yang nantinya bukan hanya sebagai media yang membantu petugas yang memberikan informasi namun juga menjadi media untuk menyampaikan informasi kepada mahasiswa program studi Teknik Informatika secara *realtime*. Seiring berkembangnya Teknik Informatika, petugas memanfaatkan *internet* untuk menyampaikan penjadwalan acara yang ada di program studi Teknik Informatika di dukung juga oleh mahasiswa sebagai yang membutuhkan informasi dengan menggunakan *internet* sebagai sumber informasi utama mengenai penjadwalan acara yang ada di program studi Teknik Informatika.

2.2 Rancang – Bangun

Menurut (Jogiyanto, 2001), Rancang bangun adalah sebuah tahap analisis dari perkembangan sistem yang merupakan definisi dari kubutuhan fungsional serta menggambarkan suatu sistem yang dibentuk berupa penggambaran, perencanaan dan pembuatan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh, termasuk konfigurasi dari komponen perangkat keras dan perangkat lunak dari suatu sistem.

Sedangkan menurut ahli (McLeod, 2002), Rancang sistem adalah penentuan proses dan data yang diperoleh oleh sistem baru. Sedangkan perancangan adalah

suatu kegiatan yang bertujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah dari pemilihan sistem terbaik, dan pengertian bangun atau pembanguna sistem adalah kegiatan menciptakan kegiatan baru atau memperbaiki sistem yang ada baik keseluruhan atau sebagian teori ini menurut (Ladjamudin, 2005).

Dengan demikian arti dari rancang bangun adalah kegiatan menerjemahkan hasil dari analisa ke dalam bentuk perangkat lunak lalu menciptakan sistem atau memperbaiki sistem yang ada.

2.3 Web Service

Menurut (Deitel, 2004), *Web service* merupakan sebuah teknologi yang meliputi kumpulan standar yang memungkinkan dua aplikasi saking berkomunikasi dan bertukar data di internet. Sedangkan menurut W3C *Web services Architecture Working Group* pengertian Web service merupakan sebuah sistem yang dibuat untuk mendukung interaksi mesin ke mesin melalui sebuah jaringan (Web Services Architecture Working Group, 2004).

2.4 Jenis – Jenis Web Service

Jenis *web service* dapat dibagi menjadi dua, yaitu REST dan SOAP.

2.4.1 Representational State Transfer (REST)

Menurut (Deitel e. a., 2012), *Representational State Transfer (REST)* mengacu ke gaya arsitektur untuk mengimplementasi *web services*. *Web services* seperti ini sering disebut dengan RESTful *web services*. RESTful *web services* di implementasi dengan menggunakan standard web. Setiap method pada RESTful

web services diidentifikasi oleh URL yang unik. Dengan demikian ketika server menerima sebuah *request* maka *server* akan segera mengetahui operasi yang akan digunakan. Setiap operasi khusus mungkin akan

tersimpan pada *cached locally* oleh *browser* ketika *service* dipanggil dengan GET *request* sehingga *request* berikutnya dengan operasi yang sama dipercepat dengan *loading* dari hasil *browser chache*. Restful *web services* merupakan alternatif selain dari SOAP *web service*.

2.4.2 Simple Object Access Protokol (SOAP)

Menurut (Kuntoro, 2005), *Simple Object Access Protocol* (SOAP) merupakan sebuah sistem protokol yang merupakan paduan dari HTTP (*Hypertext Transfer Protocol*) dan XML (*Extensible Markup Language*).

2.5 HTML (*Hypertext Markup Language*)

Menurut (Nugroho, 2006), (HTML) *Hypertext Markup Language* merupakan bahasa pemformatan teks dokumen pada jaringan komputer atau yang disebut “*World Wide Web*”. Sedangkan menurut Arief (2011), HTML merupakan salah satu format yang digunakan dalam pembuatan aplikasi yang berjalan pada halaman *web*.

2.6 Web Browser

Menurut (Arief, 2011), *Web Browser* adalah perangkat lunak atau *software* yang digunakan untuk menjalankan dokumen-dokumen *web* dengan cara diterjemahkan. Prosesnya dilakukan oleh komponen yang ada didalamnya yang

disebut *web browser*. Beberapa jenis *web browser* yang ada diantaranya *Mozilla Firefox, Google chrome, Opera, Safari*

2.7 *Java Script*

Javascript menurut (Sunyoto, 2007) adalah bahasa scripting yang populer di internet dan dapat bekerja di sebagian besar *browser* populer seperti *Internet Explorer (IE), Mozilla Firefox, Netscape* dan *Opera*. Kalau dilihat dari suku kata terdiri dari dua suku kata yaitu *Java* dan *Script*. *Java* merupakan bahasa pemrograman berorientasi objek dan *Script* adalah serangkaian intruksi program

2.8 *JQuery*

JQuery adalah *library Javascript* yang dibuat untuk memudahkan pembuatan *website* dengan HTML yang berjalan di sisi *Client*. Kumpulan kode/fungsi Javascript siap pakai, sehingga mempermudah dan mempercepat kita dalam membuat kode *javascript*.

Menurut (Aloysius, 2011) *jQuery* adalah librari atau kumpulan kode *Javascript* siap pakai. Keunggulan menggunakan *jQuery* dibandingkan dengan *Javascript* standar, yaitu menyederhanakan kode *Javascript* dengan cara memanggil fungsi-fungsi yang disediakan oleh *jQuery*. *Javascript* sendiri merupakan bahasa *Scripting* yang bekerja disisi *Client/Browser* sehingga *website* bisa lebih interaktif.

Menurut (Alexsander, 2011) “*jQuery* adalah salah satu *javascript* framework terbaik saat ini. *jQuery* dikembangkan oleh John Resig pada tahun 2006 di BarCamp NYC. Pada awal perkembangannya, *jQuery* pertama dibuat untuk

meringkas penggunaan CSS *Selector* dalam suatu pustaka fungsi. *jQuery* memiliki ciri khas pada penggunaan perintahnya, prefix untuk *jQuery* dengan tanda \$ kemudian dilanjutkan dengan fungsi atau perintah.

2.9 PHP (*Perl Hypertext Preprocessor*)

(Arief, 2011) (Arief, 2011) PHP (*Perl Hypertext Preprocessor*) adalah bahasa *server-side-scripting*. Sebuah *website* akan lebih interaktif dan dinamis bila menggunakan program PHP, maka sintaks dan perintah-perintah PHP akan dieksekusi di *server* kemudian hasilnya akan dikirimkan ke browser dengan format HTML.

Sedangkan menurut (Bunafit, 2006), PHP atau singkatan dari *Personal Home Page* merupakan bahasa skrip yang tertanam dalam HTML untuk dieksekusi bersifat *server side*. PHP termasuk dalam *open source product*, sehingga *source code* PHP dapat diubah dan didistribusikan secara bebas. Versi terbaru PHP dapat diunduh secara gratis melalui situs resmi PHP : <http://www.php.net>.

PHP dapat berjalan pada berbagai *web server* dan juga dapat berjalan di banyak sistem operasi. Salah satu keunggulan PHP adalah kemampuannya untuk koneksi ke berbagai macam jenis basis data. Sehingga dapat membuat halaman dinamis pada *web* dinamis.

2.10 Yii Framework

Menurut *Oxford English Dictionary Framework* mempunyai arti struktur dasar yang melandasi sebuah sistem. Sehingga keseluruhan sistem bergerak berdasarkan kerangka struktur dasar yang telah dibentuk sebelumnya.

Menurut (Roger , 2012), *framework* merupakan kerangka kode yang dapat disempurnakan dengan *classes* yang spesifik atau dengan fungsi yang dirancang untuk mengatasi masalah yang dihadapi.

Keuntungan dalam menggunakan *framework*

- a. Dalam penggunaan skrip yang sudah dibuat dan dites oleh penggunaan *programmer* lain, sehingga dapat mengurangi waktu pengujian pemogramaan.
- b. Memiliki aspek keamanan dan bukan hanya fungsionalitas saja karena secara langsung *framework* ini sudah dikembangkan para ahli.
- c. Dapat mengarahkan *programmer* melakukan pemecahan masalah spesifik dan telah terbukti karena telah dihadapi oleh *programmer* lain.

Kekurangan *framework* Yii

- a. Para *programmer* mungkin akan menemukan batasan-batasan ketika merancang aplikasi menggunakan *framework*.
- b. Kemungkinan akan membutuhkan waktu yang lama dan menambah biaya *development* apabila *framework* yang digunakan belum terdokumentasi dengan baik dan kurang *support*.

Fitur Yii Framework

Yii mempunyai dua buah kode generator yaitu *yiic* dan *yii*. *Yiic* adalah kode generator masih menggunakan *command line* *yiic*. Sedangkan *Gii* adalah kode generator yang berbasis *web*. Yii sudah dilengkapi dengan sebuah kode

generator berbasis *web* yang disebut Gii dari versi 1.1.2. pada saat membuat kerangka dasar aplikasi *web*, maka dengan mengetik sebaris perintah pada yiic maka yiic akan membuatnya. Apabila sudah merancang *database* lengkap dengan tabel-tabel, maka sudah dapat langsung membuat sistem tampilannya *insert*, *update* dan *delete* dengan menggunakan Gii. Artinya di sini langsung dapat membuat aplikasi dasar tanpa perlu koding. Berikut beberapa fitur lainnya dari Yii :

2.11 MVC (*Model View Controller*)

MVC adalah pola asitektur yang memisahkan tiga komponen utama logis komponen *Model*, *View*, *Controller* secara terstruktur. Salah satu penggunaan metode MVC adalah *framework Yii* pada PHP. Berikut ini adalah bagian bagian dari MVC :

a. Model

Model bukan sebuah *database* tetapi merupakan bagian (fungsi-fungsi) yang berhubungan langsung dengan *database* seperti memasukan data, pembaruan data, dan lain-lain namun tidak dapat berhubungan langsung dengan bagian *view*.

b. View

View merupakan bagian kode program yang mengatur tampilan ke pengguna. Pada aplikasi *web* bagian *view* biasanya berupa file *template* HTML, yang diatur oleh *controller*. *View* berfungsi untuk meneruma data kepada pengguna. Bagian ini mamilki akses langsung pada bagian model

c. *Controller*

Controller merupakan bagian yang mengatur hubungan antara bagian model dan *view*. *Controller* berfungsi menerima dan *request* dan data dari *user* kemudian menentukan apa yang diproses aplikasi.

2.12 XAMPP

Menurut (Betha, 2012), dalam bukunya yang berjudul Pemrograman Web Dengan PHP, menyebutkan bahwa :

“XAMPP (X(*windows/linux*) Apache MySQL PHP dan Perl) merupakan paket server web PHP dan database MySQL yang paling populer dikalangan pengembang web dengan menggunakan PHP dan MySQL sebagai databasenya”.

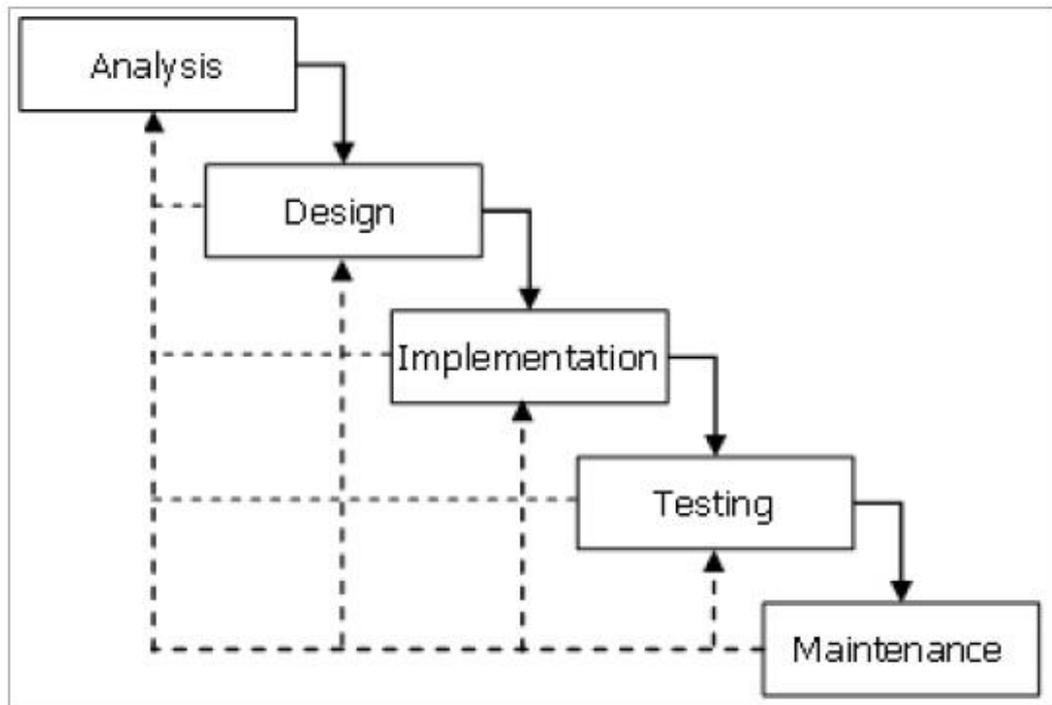
2.13 Unified Modeling Language (UML)

Menurut Britton dan Doake, (2005), *Unified Modeling Language (UML)* adalah satu kumpulan diagram, yang dirancang secara khusus untuk pengembangan berorientasi objek, dan telah menjadi standar industri untuk pemodelan sistem berorientasi objek.

2.14 Metode Pengembangan Sistem

Sebagai pengembangan pada sistem penelitian ini maka akan digunakan model SDLC (*Software Development Life Cycle*) pengembangan atau rekayasa

website, berikut ini adalah gambaran dari model SDLC :



Gambar 2.1 Metode Pengembangan Sistem (SDLC)

a. Tahap Analisis

Pada tahap ini dilakukan proses perencanaan menyangkut studi tentang kebutuhan pengguna kelayakan baik secara teknik maupun secara teknologi.

b. Tahap Perancangan

Pada tahap ini menyangkut sistem dimana akan diberikannya solusi dari segala masalah yang ada setelah dilakukannya tahap analisis.

c. Tahap Implementasi

Pada tahapan ini akan dilakukan implementasi yang ada pada tahap perncanaan kesituasi nyata dengan pemilihan perangkat keras dan penyusunan desain.

d. Tahap Pengujian

Pada tahap ini dilakukan proses pengujian terhadap *website* yang sudah dibuat apakah masih ada kekurangan dan kelemahan.

e. Tahap Pemeliharaan

Pada tahap ini dilakukan perawatan dan pemeliharaan *web*, jika diperlukan perbaikan kecil kemudian jika priode sistem sudah habis akan masuk lagi pada tahap perencanaan.

2.15 Use Case Diagram

Menurut Britton dan (Briton , 2005), *Use-Case Diagram* adalah diagram yang secara grafis menggambarkan interaksi antara *user* dengan sistem. Simbol umum digunakan dapat dilihat pada Tabel 2.1

Tabel 2.1 Tabel Simbol Use Case

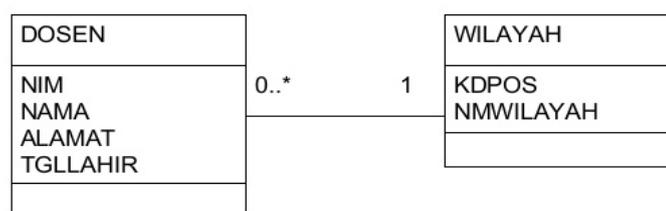
| SIMBOL | NAMA | KETERANGAN |
|---|----------------------|---|
|  | <i>Actor</i> | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> . |
|  | <i>Dependency</i> | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang tidak mandiri. |
|  | <i>Generalation</i> | Hubungan dimana objek anak (<i>descendation</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>). |
|  | <i>Use Case</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |
|  | <i>Collaboration</i> | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya |
|  | <i>Association</i> | Apa yang menghubungkann antara objek satu dengan objek lainnya. |
|  | <i>System</i> | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |

2.16 Class Diagram

Class Diagram adalah diagram yang menunjukkan class-class yang ada dari sebuah sistem dan hubungannya secara logika. Class diagram menggambarkan struktur statis dari sebuah sistem. Karena itu class diagram merupakan tulang punggung atau kekuatan dasar dari hampir setiap metode berorientasi objek termasuk UML. (Hendri, 2008)

Class diagram merupakan diagram statis. *Class* diagram mewakili pandangan statis dari aplikasi. *Class* diagram tidak hanya digunakan untuk memvisualisasikan, menggambarkan, dan mendokumentasikan perbedaan aspek dari sistem, tetapi juga untuk membangun kode *executable* dari aplikasi *software*.

Class diagram menampilkan koleksi dari *class*, *interface*, *associations*, *collaborations*, dan *constraints*. Hal ini juga dikenal sebagai diagram struktural. Untuk contoh gambar *class diagram* ada pada gambar 2.2 berikut



Gambar 2.2 Contoh *Class Diagram*