

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Gambaran umum Sistem Informasi**

Sistem informasi akademik pada umumnya berhubungan dengan menampilkan KRS dan KHS serta informasi tentang universitas tersebut.

Hampir semua universitas dan instansi-instansi saat ini menggunakan sistem informasi berbasis *website*. Sebagai contoh Universitas Lampung yang hampir semua informasi tentang kegiatan kampus, pengumuman dapat dilihat melalui Siakad *online* sehingga dapat diakses di manapun dan kapanpun tanpa harus datang langsung ke kampus untuk mendapatkan informasi yang dicari.

Sistem SMS *gateway* yang biasanya dikembangkan juga menguntungkan mahasiswa tetapi memiliki kelemahan karena hanya fokus pada satu informasi saja dan proses perulangan (kirim dan terima) SMS yang banyak.

Seiring majunya jaman dunia selular bukan tidak mungkin menghadirkan layanan *website* melalui WAP yang juga dapat dikembangkannya sistem informasi universitas untuk melakukan pengisian KRS dan melihat KHS.

## 2.2 Metode Prototyping

Menurut Yazuhara :

Menurut literatur, yang dimaksud dengan prototipe (*prototype*) adalah "model pertama", yang sering digunakan oleh perusahaan industri yang memproduksi barang secara massal. Tetapi dalam kaitannya dengan sistem informasi definisi kedua dari Webster yang menyebutkan bahwa "*prototype is an individual that exhibits the essential peatures of later type*". Bila diaplikasikan dalam pengembangan sistem informasi manajemen dapat berarti bahwa prototipe tersebut adalah sistem informasi yang menggambarkan hal-hal penting dari sistem informasi yang akan datang. Prototipe sistem informasi bukan merupakan sesuatu yang lengkap, tetapi sesuatu yang harus dimodifikasi kembali, dikembangkan, ditambahkan atau digabungkan dengan sistem informasi yang lain bila perlu.

Dalam beberapa hal pengembangan *software* berbeda dengan produk-produk manufaktur, setiap tahap atau fase pengembangan sistem informasi merupakan bagian yang tidak terpisahkan dari seluruh proses yang harus dilakukan. Proses ini umumnya hanya untuk satu produk dan karakteristik dari produk tersebut tidak dapat ditentukan secara pasti seperti produk manufaktur, sehingga penggunaan "model pertama" bagi pengembangan *software* tidak tepat. Istilah *prototyping* dalam hubungannya dengan

pengembangan *software* sistem informasi manajemen lebih merupakan suatu proses bukan prototipe sebagai suatu produk.

### **2.2.1 Karakteristik metode *prototyping***

Menurut Yazuhara :

Ada empat langkah yang menjadi karakteristik metode *prototyping* yaitu :

1. Pemilahan fungsi

Mengacu pada pemilahan fungsi yang harus ditampilkan oleh *prototyping*. Pemilahan harus selalu dilakukan berdasarkan pada tugas-tugas yang relevan yang sesuai dengan contoh kasus yang diperagakan.

2. Penyusunan sistem informasi bertujuan untuk memenuhi permintaan akan tersedianya *prototype*.

3. Evaluasi

Yaitu pengujian yang dilakukan untuk mendapatkan umpan balik dari konsumen.

4. Penggunaan selanjutnya

### **2.2.2. Jenis-jenis *prototyping***

Menurut Yazuhara :

Ada 4 jenis dari *prototyping* yaitu :

1. *Feasibility prototyping* – digunakan untuk menguji kelayakan dari teknologi yang digunakan untuk sistem informasi yang disusun.

2. *Requirement prototyping* – digunakan untuk mengetahui kebutuhan aktivitas bisnis *user*.
3. *Desain Prototyping* – digunakan untuk mendorong perancangan sistem informasi yang akan digunakan.
4. *Implementation prototyping* – merupakan lanjutan dari rancangan prototype, prototype ini langsung disusun sebagai suatu sistem informasi yang digunakan.

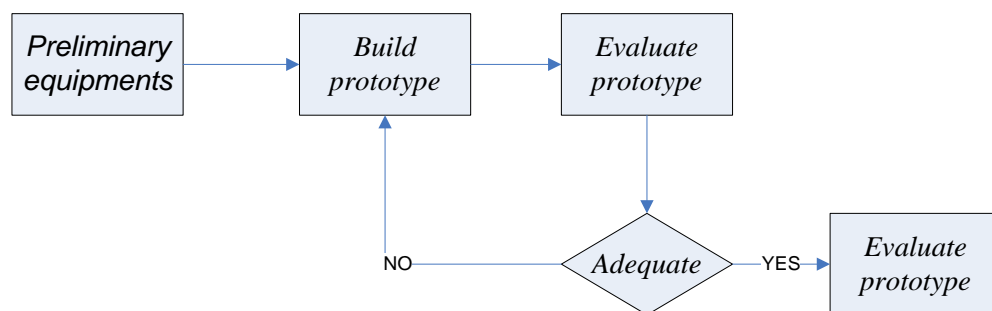
### 2.2.3 Metode pendekatan *prototyping*

Menurut Karmila :

Ada tiga metode pendekatan *prototyping* yaitu :

#### 1. *Throw away prototyping*

Prototype dibuat dan dites, pengalaman yang diperoleh dari latihan ini digunakan untuk membuat produk *final*-nya, tetapi *prototype*-nya dibuang. Proses ini dapat dilihat pada gambar 2.1

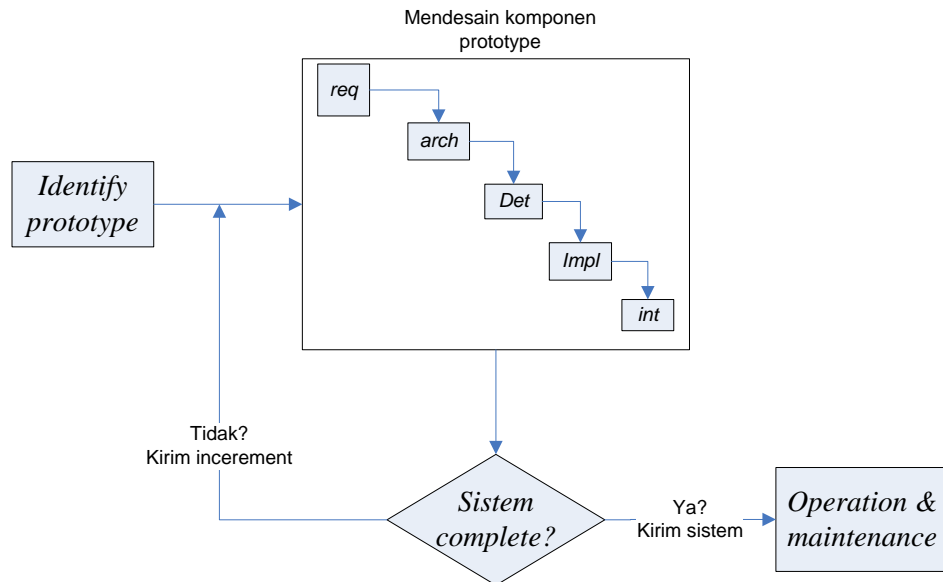


Gambar 2.1 *Throw away prototype*

#### 2. *Incremental*

Produk *final*-nya dibuat dalam bentuk komponen-komponen yang terpisah, dan dikerjakan satu komponen dalam satu saat.

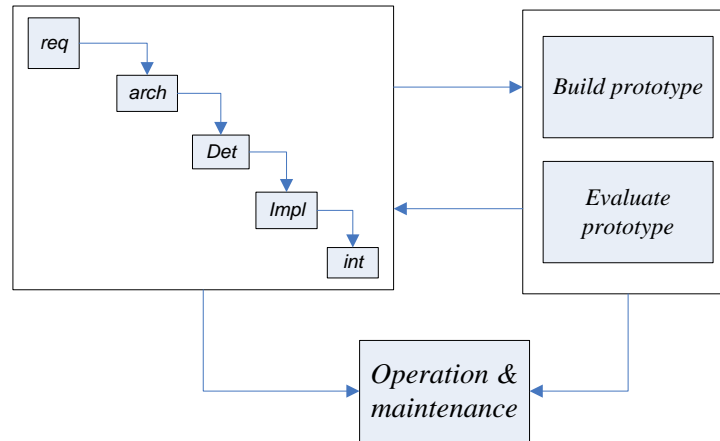
Sebetulnya hanya ada satu rancangan *final* dari sistem, tetapi sistem ini dibagi-bagi dalam komponen-komponen terpisah yang lebih kecil. Proses ini dapat dilihat pada gambar 2.2



Gambar 2.2 *Incremental prototype*

### 3. *Evolutionary*

Pada metode ini, *prototypenya* tidak dibuang, melainkan digunakan sebagai dasar untuk iterasi perancangan selanjutnya. Dalam hal ini, sistem yang sesungguhnya dipandang sebagai evolusi dari versi awal yang terbatas menuju produk akhirnya. Proses ini dapat dilihat pada gambar 2.3



Gambar 2.3 *Evolutionary prototype*

Sumber :

<http://karmila.staff.gunadarma.ac.id/Downloads/files/2614/prototype.doc>

#### 2.2.4 Keunggulan dan kelemahan metode *prototyping*

Menurut Yazuhara :

Metode prototyping memiliki keunggulan dan kelemahan sebagai berikut :

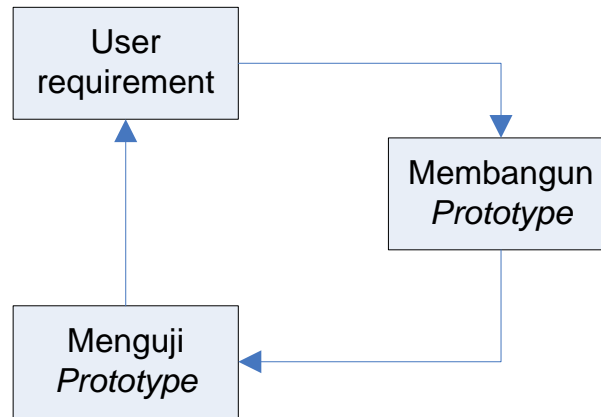
Keunggulannya :

1. *End user* dapat berpartisipasi aktif.
2. Penentuan kebutuhan lebih mudah diwujudkan.
3. Mempersingkat waktu pengembangan sistem informasi.

Kelemahannya :

1. Proses analisis dan perancangan terlalu singkat
2. Mengesampingkan alternatif pemecahan masalah
3. Bisanya kurang fleksible dalam menghadapi perubahan
4. *Prototype* yang dihasilkan tidak selamanya mudah dirubah
5. *Prototype* terlalu cepat selesai

Dapat disimpulkan bahwa terdapat tiga proses penting dalam pengembangan sistem menggunakan *prototype* yang dapat digambarkan pada gambar 2.4.



Gambar 2.4 Tiga proses penting dalam prototyping

### 2.3 *Wireless Application Protocol (WAP)*

Menurut Udin Suchaini :

Dasar aplikasi pemrograman *Wireless Application Protocol (WAP)* *WAP* merupakan sebuah protokol aplikasi komunikasi, digunakan sebagai media untuk mengakses sebuah informasi dan sebagai biro jasa. *WAP* dirancang untuk *micro browser*, untuk *handle* sebuah perangkat *mobile* atau telepon seluler yang dapat digunakan untuk menciptakan jaringan telepon seluler. *WAP* sendiri merupakan turunan dari aplikasi *XML* versi 1.0. *WAP server* dapat dibangun dengan menggunakan aplikasi *Apache Web Server*, sedangkan aplikasi *WAP* dapat dibangun dengan menggunakan bahasa pemrograman *WML*. *WAP* memformat data internet agar bisa dibaca dengan perangkat telepon seluler. Permintaan data (*request*) dari ponsel dikirimkan ke *WAP server*

melalui *WAP gateway*. *WAP (Wireless Application Protocol)* merupakan layanan telepon selular dengan standar internet. *WAP* didukung perusahaan Ericsson, Nokia and Motorola. *WAP micro browser* merupakan perangkat lunak yang berkapasitas kecil yang digunakan untuk membuat perintah pada perangkat keras, *CPU* dan memori sehingga memberikan informasi yang ditulis dengan bahasa pemrograman *WML*. *Micro browser* ini juga bisa menampilkan informasi yang ditulis dengan menggunakan *Javascript*, *WMLScript* dan *PHP* yang dikolaborasikan dengan bahasa pemrograman.

Untuk merancang sebuah *WAP* yang dinamis diperlukan beberapa bahan pendukung, misalnya:

- a) *Hardware* untuk *server*.
- b) Sistem operasi *Windows*, *Linux*, *OS/2*, *Mac OS* atau *Unix*.
- c) *WAP Hosting DNS (Domain Name Sistem)*.
- d) *WAP Gateway*, fungsi *WAP gateway* dioperasikan oleh perusahaan operator selular.
- e) *Web Server* seperti *Personal Web Server*, *Internet Information Server*, atau *Apache*, *Server Internet* yang mampu diakses oleh *WAP browser*. Informasi pada *web server* dibuat dengan menggunakan pemrograman *website* dan *WML* dan berhubungan dengan *database*.
- f) *Database Management Sistem* seperti *MySQL*, *Microsoft SQL Server*, *Oracle*, *Informix*, atau *Sybase Database*.



- g) Bahasa pemrograman seperti *PHP*, *Perl*, *Microsoft Active Server Page (ASP)*, *Java Server Page (JSP)*, *Allaire Cold Fusion*.
- h) Yang tidak kalah penting adalah *WAP browser* pada ponsel, untuk mengakses *WAP* pada telepon seluler. Untuk melakukan pembuatan aplikasi *WAP*, *WAP emulator* digunakan sebagai alat bantu pengganti *mobile phone* atau telepon seluler. Perangkat ini merupakan suatu *software* yang dapat digunakan untuk simulasi pada media telepon seluler. Ada banyak *WAP emulator* yang tersedia di internet, dan juga banyak yang dapat di-*download* secara gratis, seperti *M3 Gate*, *Winwap Browser* dan lain sejenisnya.

#### **2.4 Dynamic Database Mobile**

Menurut Unang Achlison :

*Database mobile* adalah data tidak terhubung (*disconnected database*) ke jaringan komputer namun akses data bisa di mana-mana. *Ad-hoc database* memeriksa perubahan data yang diakses oleh *user mobile (mobile host)*. *Broadcast disks* terhubung ke *database mobile* dan melayani *user mobile*.

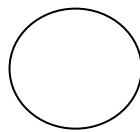
#### **2.5 Data Flow Diagram**

*Data Flow Diagram (DFD)* adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data di mana komponen-komponen asal, tujuan, dan penyimpanan dari data tersebut.

DFD dapat digunakan untuk dua hal utama, yaitu untuk membuat dokumentasi dari sistem informasi yang ada, atau untuk menyusun dokumentasi untuk sistem informasi yang baru.

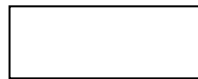
DFD terdiri dari 4 (empat) simbol. Simbol ini digunakan sebagai elemen sistem, arus data dan penyimpanan data. Berikut ini merupakan simbol-simbol yang ada pada DFD :

**a. Proses.**



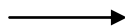
Simbol proses menggambarkan kegiatan yang mengubah input menjadi output. Tiap proses diberikan nama atau label dari kegiatan tersebut dengan menggunakan kata kerja.

**b. External Entity.**



Simbol *external entity* biasanya merupakan pelaku, juga menandakan titik berakhirnya sistem.

**c. Arus data.**



Arus data menggambarkan kegiatan apa yang dilakukan oleh external entity kepada sistem, kepada *data store / master data*, ataupun ke proses selanjutnya.

d. **Data store** atau penyimpanan data.



Data store adalah suatu penampungan data dalam DFD

## 2.6 Entity Relationship Diagram

Menurut Rahmat :

*Entity Relationship Diagram* (ERD) merupakan suatu model untuk menjelaskan hubungan antar data dalam **basis data** berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.

ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarannya digunakan beberapa notasi dan simbol.

Pada dasarnya ada tiga simbol yang digunakan, yaitu :

### a. Entiti

Entiti merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain (Fathansyah, 1999: 30).

Simbol dari entiti ini biasanya digambarkan dengan persegi panjang.

### b. Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasi isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol elips.

### c. Hubungan atau Relasi

Relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dalam satu **basis data** yaitu (Abdul Kadir, 2002: 48) :

#### 1. Satu ke satu (*One to one*)

Hubungan relasi satu ke satu yaitu setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B.

#### 2. Satu ke banyak (*One to many*)

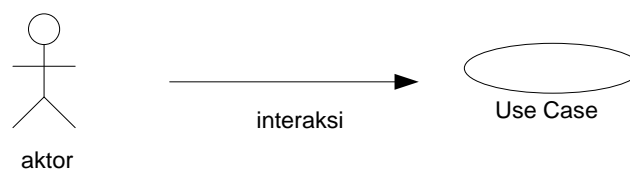
Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A.

#### 3. Banyak ke banyak (*Many to many*)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B.

## 2.7 Use case

*Use case* diagram menggambarkan pekerjaan sistem, tanpa menjelaskan secara detail bagaimana sistem itu melakukannya. Dalam *use case* juga menggambarkan interaksi apa saja antara pengguna dan sistem. Membuat *use case* diagram termasuk dalam tahap menganalisa *user requirement*.



Gambar 2.5 Komponen dalam *use case*

## **2.8 General Packet Radio Services (GPRS)**

Menurut Udin Suchaini :

*GPRS* mendistribusikan paket data akses internet sampai 114Kbps. Transfer data menuju jaringan internet (*web server*) melalui jaringan *GPRS* Selular. Layanan yang ada pada internet dapat diakses melalui *GPRS*, karena *protocol GPRS* sama seperti internet. Dalam jaringan *GSM*, membutuhkan modul *GGSN* (*Gateway GPRS Service Node*) dan *SGSN* (*Serving GPRS Service Node*). *GGSN* bertindak sebagai *gateway* antara jaringan *GPRS* dan jaringan data publik seperti *Internet Protocol (IP)*.

## **2.9 Pemrograman Wireless Markup Language (WML)**

Menurut Andi :

*WML* adalah kepanjangan dari *Wireless Markup Language*, yaitu bahasa pemrograman yang digunakan dalam aplikasi berbasis XML (*eXtensible Markup Language*). *WML* ini adalah bahasa pemrograman yang digunakan dalam aplikasi *wireless*. *WML* merupakan analogi dari *HTML* yang berjalan pada protokol nirkabel.

Menurut Udin Suchaini :

*WML* merupakan bahasa pemrograman untuk membuat sebuah halaman yang dapat diperlihatkan atau ditampilkan pada *WAP browser*. *WML* merupakan bahasa pemrograman turunan dari *HTML*, tetapi didasarkan pada bahasa *XML*. Halaman *WML*

disebut dengan *deck*, yaitu kumpulan dari *card* (Utomo, 2006:1). Aplikasi dari *WAP* dapat digunakan untuk informasi kereta, pesawat, pemesanan tiket, informasi nomor telepon, alamat, dan masih banyak lagi.

a. Struktur penulisan bahasa *WML*

Cara penulisan program *WML* berbeda dengan program *HTML*, yang membedakan adalah *script* tersebut adalah program *WML* yang dimulai dengan *tag* `<wml>` dan diakhiri dengan *tag* `</wml>`. Program *WML* ditulis dengan versi *XML*, sehingga dalam penulisan harus dideklarasikan ke dalam bentuk *XML*. Apabila tidak, maka *WAP emulator* atau telepon seluler tidak dapat mengenalinya. *WAP server* dapat dibangun dengan menggunakan aplikasi *Apache Webserver*, sedangkan aplikasi *WAP* dapat dibangun dengan menggunakan bahasa pemrograman *WML*.

*WML scripts* adalah bahasa pemrograman yang digunakan pada aplikasi *WAP*, yang mana *WML* dan *HTML* ada kemiripan.

Bentuk dasar penulisan *tag* `<wml>` adalah sebagai berikut.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.3//EN"
"http://www.wapforum.org/DTD/wml13.dtd" >
<wml>
<card>
(Lokasi eksekusi data)
</card>
```

```
</wml>
```

Penulisan *script* di atas untuk melakukan eksekusi secara umum. Warna dari *script code* tiap bahasa pemrograman berbeda. Sedangkan untuk *script code WML* yang biasa digunakan memiliki warna biru tua seperti di atas.

#### b. Konfigurasi untuk mendukung *WML*

Konfigurasi penting untuk penambahan parameter yang menyatakan dukungan untuk *WAP*, yaitu menambahkan beberapa parameter untuk file ekstensi yang terkait dengan program *WAP*. Ekstensi-ekstensi program *WAP* yang digunakan antara lain *.wml*, *.wbmp*, *.wmls*, atau *.php* yang terdapat pada *httpd.conf*. Parameter tersebut diletakkan pada baris setelah parameter *Addtype* yang berkaitan dengan *PHP*.

## 2.10 Pemrograman *PHP*

Menurut Udin Suchaini :

*PHP* adalah bahasa pemrograman berbasis *website*. Bahasa ini mempunyai kelebihan yaitu komabilitasnya dengan berbagai macam jenis *database*, dukungan dengan berbagai macam jenis sistem operasi.

*PHP* lebih cocok dan umum digunakan jika digabungkan dengan *database MySQL*. *MySQL* dengan *PHP* seakan-akan dua hal yang tidak dapat dipisahkan. Tentunya untuk dapat menggunakan

keduanya dibutuhkan tingkat kemampuan *programming* tertentu.

Bentuk dasar. Penulisan *script code PHP* adalah sebagai berikut :

```
<?php  
echo"....."; //lokasi eksekusi data  

```

*Script* di atas merupakan salah satu dari cara penulisan eksekusi dari bahasa pemrograman *PHP*. Perbedaan warna terlihat jelas dengan fungsi masing-masing yang berbeda. `<?php ?>` berwarna merah, hal ini menunjukkan bahwa *script code* yang dieksekusi menggunakan bahasa pemrograman *PHP*. *Script echo* merupakan struktur kontrol menampilkan tulisan. *Script* ini berwarna biru, sedangkan isi dari *echo* akan berwarna merah dan diakhiri dengan tanda titik koma. Penulisan lokasi eksekusi data diawali dengan tanda dua buah garis miring ke kanan (`//`) dimaksudkan bahwa penulisan ini tidak akan dieksekusi pada saat melakukan *compiler* terhadap *script* yang dijalankan, serta terlihat berwarna oranye secara otomatis. Bahasa pemrograman *PHP* banyak digunakan oleh *programmer* berlatar belakang *C/C++* karena kemiripan *syntax* atau bahasanya. *Open source*, karenanya gratis dan bebas. *Database* pasangannya biasanya *MySQL*, dijalankan bersama *Webserver Apache* di atas *Operating Sistem Linux*.