

BAB II

LANDASAN TEORI

2.1 Rancang

Rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen – komponen sistem diimplementasikan (Pressman, 2002). Perancangan adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah–masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik (Ladjamudin, 2005).

2.2 Bangun

Bangun atau pembangunan sistem adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian (Pressman, 2002).

2.3 Sistem

(Tata Sutabri, 2012) mendefinisikan Sistem adalah sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu.

Menurut Tata Sutabri sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu :

1. **Komponen sistem (*Component*)**

Susatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan.

2. **Batasan sistem (*Boundary*)**

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkung luarnya. Batasan

sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

3. Lingkungan luar sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang memengaruhi operasi sistem tersebut disebut lingkungan luar sistem. lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau interface. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain.

5. Masukan sistem (*Input*)

Energi yang dimasukkan kedalam sistem tersebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

6. Keluaran sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi input bagi subsistem lain.

7. Pengolahan sistem (*Proses*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya adalah sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

8. Sasaran sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran dan tujuan yang telah direncanakan.

2.4 Monitoring

Monitoring atau pengawasan sebagai mendeterminasi apa yang telah dilaksanakan, maksudnya mengevaluasi prestasi kerja dan apabila perlu, menerapkan tindakan – tindakan korektif sehingga hasil pekerjaan sesuai dengan rencana yang telah ditetapkan. (George , 2008).

2.5 Aplikasi Mobile

Menurut (Safaat, 2012) Perangkat lunak aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna. Aplikasi mobile berasal dari kata *application* dan *mobile*. *Application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program siap pakai yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju sedangkan mobile dapat di artikan sebagai perpindahan dari suatu tempat ke tempat yang lain.

Aplikasi ini dapat diakses melalui perangkat nirkabel seperti pager, seperti telepon seluler dan PDA. Adapun karakteristiknya yaitu:

1. Ukuran yang kecil : Perangkat *mobile* memiliki ukuran yang kecil. Konsumen menginginkan perangkat yang terkecil untuk kenyamanan dan mobilitas mereka.
2. Memory yang terbatas : Perangkat *mobile* juga memiliki memory yang kecil, yaitu *primary (RAM)* dan *secondary (disk)*.
3. Daya proses yang terbatas : Sistem *mobile* tidaklah setangguh rekan mereka yaitu *desktop*.
4. Mengonsumsi daya yang rendah : Perangkat *mobile* menghabiskan sedikit daya dibandingkan dengan mesin desktop.

5. Kuat dan dapat diandalkan : Karena perangkat *mobile* selalu dibawa kemana saja, mereka harus cukup kuat untuk menghadapi benturan-benturan, gerakan, dan sesekali tetesan-tetesan air.
6. Konektivitas yang terbatas : Perangkat *mobile* memiliki bandwidth rendah, beberapa dari mereka bahkan tidak tersambung.
7. Masa hidup yang pendek : Perangkat-perangkat konsumen ini menyala dalam hitungan detik kebanyakan dari mereka selalu menyala.

2.6 Konsep Data Informasi

2.6.1. Data

Data dapat didefinisikan sebagai bahan keterangan tentang kejadian-kejadian nyata atau fakta-fakta yang dirumuskan dalam kelompok lambang tertentu yang tidak acak yang menunjukkan jumlah, tindakan, atau hal (Sutanta, 2004).

2.6.2. Informasi

Informasi adalah data yang telah diklasifikasi atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya mengolah data dari tak berguna menjadi berguna bagi penerimanya. Nilai informasi dilukiskan paling berarti dalam konteks sebuah keputusan (Tata Sutabri, 2012).

2.7 World Wide Web (WWW)

World Wide Web (WWW), lebih dikenal dengan web yang merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet dengan fasilitas hypertext untuk menampilkan data berupa text, gambar, suara, animasi dan data multimedia lainnya (Kustiyahningsih & Anamisa, 2011). Sehingga web pada awalnya adalah ruang informasi dalam internet, dengan menggunakan teknologi hypertext, pemakai dituntut menemukan informasi dengan mengikuti

link yang disediakan dalam dokumen web yang ditampilkan dalam web browser. Situs atau web dapat dikategorikan menjadi dua yaitu :

1. Web Statis

Web statis adalah web yang menampilkan informasi-informasi yang sifatnya statis(tetap). Disebut statis karena pengguna tidak dapat berinteraksi dengan web tersebut.

2. Web Dinamis

Web dinamis adalah web yang menampilkan informasi serta dapat berinteraksi dengan pengguna. Web yang dinamis memungkinkan pengguna untuk berinteraksi menggunakan form sehingga dapat mengolah informasi yang ditampilkan.

2.8 URL

URL (Universal Resource Locator) adalah konsep nama file standar yang diperluas dengan jaringan untuk menentukan lokasi informasi pada web browser. Nama file ini tidak hanya menunjukkan direktori dan nama filenya (Kustiyahningsih & Anamisa, 2011). URL dapat disediakan (ada atau diakses) dengan berbagai metode, dan bisa jadi bukan sekedar file karena URL dapat menunjukkan query-query, dokumen yang disimpan dalam database, hasil dari suatu perintah finger atau archie atau apapun yang berkaitan dengan data hasil proses.

2.9 HTTP

HTTP (Hypertext Transfer Protocol) adalah suatu protokol yang menentukan aturan yang perlu diikuti oleh web browser dalam meminta atau mengambil suatu dokumen dan menyediakan dokumen yang diminta oleh browser. Server HTTP umumnya digunakan untuk melayani dokumen hypertext, karena HTTP adalah protokol dengan overhead yang sangat rendah (Kustiyahningsih & Anamisa, 2011).

2.10 Perangkat Lunak yang Digunakan

2.10.1. Notepad ++

Notepad++ merupakan sebuah penyunting teks dan penyunting kode sumber yang tersedia dalam beberapa sistem operasi seperti Windows, Linux, dan Mac OS X. Notepad ++ menggunakan komponen Scintilla untuk dapat menampilkan dan menyunting teks dan berkas kode sumber berbagai bahasa pemrograman. Notepad ++ didistribusikan sebagai perangkat lunak bebas.

2.10.2. Framework Bootstrap

Menurut (Husein, 2013) *Bootstrap* merupakan *framework* ataupun *tools* untuk membuat aplikasi *website* ataupun situs *web responsive* secara cepat, mudah dan gratis, karena *website* yang dibangun oleh peneliti merupakan *website* yang dapat diakses dalam perangkat *mobile* ataupun *personal computer*. *Bootstrap* terdiri dari CSS dan HTML untuk menghasilkan *Grid, Layout, Typography, Table, Form, Navigation*, dan lain lain.

2.10.3. XAMPP

XAMPP merupakan aplikasi server yang menggabungkan beberapa aplikasi *server* yang biasa digunakan di *web server*. Berikut beberapa komponen – komponen yang terdapat pada XAMPP, yaitu Apache (*web server*), MySQL (*database server*), Filezila FTP *server*, Mercury Mail (*mail server*), Php MyAdmin (*web-based interface MySQL*).

2.10.4. Dreamweaver CS5

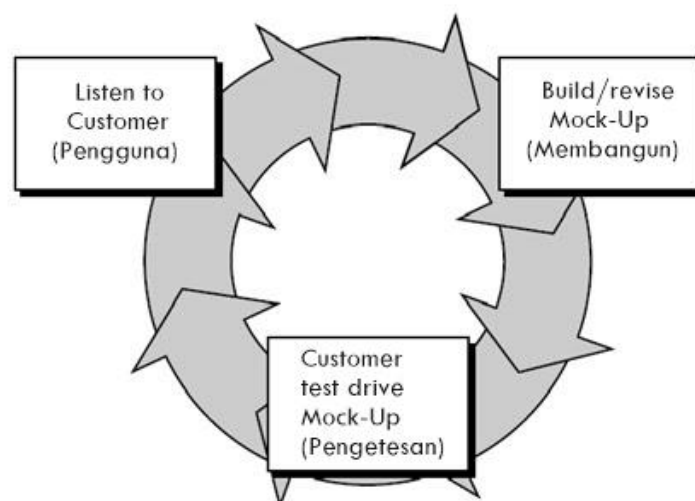
Dreamweaver merupakan software aplikasi yang digunakan sebagai *HTML editor* profesional WYSIWYG (*What You See Is What You Get*), yang intinya adalah bahwa anda tidak harus memberikan keleluasaan kepada anda untuk menggunakan sebagai media penulisan bahasa pemrograman *web*. Dengan kemampuan

fasilitas yang optimal dalam jendela *design* membuat program ini memberikan kemudahan untuk mendesain *web* meskipun untuk para *web designer* pemula (Andi Offset, 2010). Sedangkan kemampuan *Dreamweaver* untuk berinteraksi dengan beberapa bahasa pemrograman seperti *PHP*, *ASP*, *JavaScript*.

2.11 Metode Pengembangan Perangkat Lunak

2.11.1 Metode Prototype

Menurut (Rosa dan Salahuddin, 2015) Model prototipe dapat digunakan untuk menyambung ketidak pahaman pelanggan mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak. Model prototype (*prototyping model*) dimulai dari mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat. Lalu dibuatlan program prototipe agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program ini biasanya menyediakan tampilan dengan simulasi alur perangkat lunak sehingga tampak seperti perangkat lunak yang sudah jadi. Program prototipe ini dievaluasi oleh pelanggan atau *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user*.



Gambar 2.1 Model *Prototype*

Tahapan yang dilakukan saat mengembangkan sistem dengan Model *Prototype* berdasarkan Gambar 2.1, yaitu :

- 1) *Listen to customer* (Pengguna)
Pengembang mendengarkan kebutuhan pengguna sebagai pemakai sistem perangkat lunak (*user*) untuk menganalisis serta mengembangkan kebutuhan *user*.
- 2) *Build/revise mock-up* (Membangun)
Mock-up adalah suatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi, atau keperluan lain. *Mock-up* disebut sebagai prototipe perangkat lunak jika menyediakan atau mampu mendemostrasikan sebagian besar fungsi dari sistem perangkat lunak.
- 3) *Customer test drives mock-up* (Pengetesan)
Customer melakukan pengujian terhadap *mock-up* yang telah dibuat. Pengujian *mock-up* yang dilakukan berdasarkan keinginan *customer*.

2.11.2 *UML (Unified Modeling Language)*

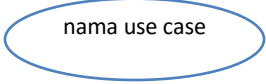


(Rosa dan Salahuddin, 2015) mendefinisikan Unified Modelling Language (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan system piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML dapat dibuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun.




1. *Use Case Diagram*

Use case atau diagram use case merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antar satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan apa siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dandapatdipahami. ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case:

- 1) Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- 2) Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar aktor.

Tabel 2.1 Simbol *Use Case Diagram*

Simbol	Deskripsi
<p><i>Use case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit untuk bertukar pesan antar unit, biasanya dinyatakan dengan menggunakan awal frase nama <i>use case</i> .
<p>Asosiasi/<i>association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
<p>Generalisasi/<i>generalization</i></p> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.




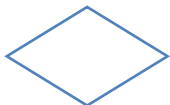
<p>Menggunakan <i>include</i></p> <p><<<i>include</i>>></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i> :</p> <ol style="list-style-type: none"> 1) <i>include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan 2) <i>include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan.
<p>Aktor/<i>actor</i></p>  <p>nama <i>actor</i></p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar 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.</p>
<p>Menggunakan Ekstensi/<i>extend</i></p> <p><<<i>extend</i>>></p> 	<p>Relasi <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>


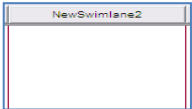
2. Activity Diagram

(Rosa dan Shalahuddin, 2015) diagram aktivitas atau *activity diagram* menggambarkan *workflow* (alir kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

- 1) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- 2) Urutan atau pengelompokan tampilan dari sistem/*uses interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- 3) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan pengujian yang didefinisikan kasus ujinya.
- 4) Rancangan menu yang ditampilkan pada perangkat lunak.

Tabel 2.2 Simbol Activity Diagram

Simbol	Keterangan
<i>Start state</i> 	Menunjukkan dimulainya suatu <i>workflow</i> pada sebuah <i>activity diagram</i> .
<i>End state</i> 	Menggambarkan akhir atau terminal dari pada sebuah <i>activity diagram</i> .
<i>State transition</i> 	Menunjukkan kegiatan apa berikutnya setelah suatu kegiatan sebelumnya
<i>Decision</i> 	Suatu titik/point pada <i>activity diagram</i> yang mengindikasikan suatu kondisi dimana ada kemungkinan perbedaan transisi.




<i>Activity</i> 	Menunjukkan aktivitas yang ada dalam system
<i>Swimlane</i> 	Membantu untuk lebih memahami siapa atau apa yang memulai aktivitas.

3. *Sequence Diagram*

(Rosa dan Salahuddin, 2015), *Sequence* diagram secara grafis menggambarkan bagaimana object berinteraksi dengan satu sama lain melalui pesan pada eksekusi sebuah *use case* atau proses. Diagram ini mengilustrasikan bagaimana pesan terkirim dan diterima di antara objek.

Adapun simbol–simbol dari *sequence* diagram sebagai berikut:

Tabel 2.3 Simbol *Sequence Diagram*

Nama Komponen	Komponen	Simbol
<i>Lifeline</i>	<i>Lifeline</i> mengindikasikan keberadaan sebuah <i>object</i> dalam basis waktu. Notasi untuk <i>Lifeline</i> adalah garis putus-putus vertikal yang ditarik dari sebuah <i>object</i> .	
<i>Activation</i>	<i>Activation</i> dinotasikan sebagai sebuah kontak segi empat yang gambar pada sebuah <i>lifeline</i> . Mengindikasikan sebuah <i>object</i> yang akan melakukan sebuah aksi.	
<i>Message</i>	<i>Message</i> , digambarkan dengan anak panah <i>horizontal</i> antara <i>activation message</i> mengindikasikan komunikasi antara <i>object-object</i> .	

2.12 Penelitian terdahulu

Tabel 2.4 berikut ini merupakan penelitian terdahulu dari sistem monitoring berbasis mobile

Tabel 2.4 Penelitian Terdahulu

No	Nama	Judul	Terbit/Tahun	Abstrak
1	Rian Mamora	Rekayasa Perangkat Lunak Sistem Monitoring Guru pada Perguruan Taman Siswa Bandar Lampung	Teknik Informatika, IBI Darmajaya Bandar Lampung/2013	Pelaksanaan pemantauan dilakukan dengan evaluasi kepada para guru yang dilakukan oleh Kepala Sekolah. Tujuan dari aplikasi sistem pemantauan guru adalah sebagai masukan dalam bentuk jawaban untuk pertanyaan yang diajukan kepada siswa yang diajar. Mengelola data dan mengevaluasi guru untuk meningkatkan kualitas pendidikan dan juga menciptakan kompetensi guru dalam mempertahankan dan meningkatkan pengajaran. Oleh karena itu penulis menganalisa dan mendesain "rekayasa perangkat lunak guru Sistem Monitoring di taman siswa lembaga bandar lampung
2	Desmi Haldi	Sistem Penjadwalan Mata Pelajaran pada SMAN 9 Bandar Lampung	Teknik Informatika, IBI Darmajaya Bandar Lampung/2011	SMA N 9 Bandar Lampung merupakan institusi pendidikan yang ada di Bandar Lampung. Salah satu kelengkapan yang ada adalah penjadwalan mata pelajaran . Penjadwalan yang dilakukan dengan cara menentukan hari, jam sesuai dengan kebutuhan kelas, jumlah mata pelajaran dan guru. Namun dalam pelaksanaan sering terjadi permasalahan. Dengan terciptanya sistem ini diharapkan masalah-masalah tersebut bisa diatasi dengan baik .