

## BAB IV

### METODE PENELITIAN

#### 4.1. Lokasi dan Waktu Penelitian

##### 4.1.1. Tempat Penelitian

Penelitian ini dilakukan di Rumah Sakit Umum Daerah (RSUD) Sekayu, yang beralamat di Jl. Kol. Wahid Udin Lingkungan I Kel. Kayuara Kec. Sekayu Kab.Musi Banyuasin 30711.

##### 4.1.2. Waktu Penelitian

Penelitian ini dilaksanakan pada bulan September sampai bulan Desember 2017. Waktu yang dibutuhkan dalam membangun Sistem Pendukung Keputusan Penilaian Kinerja Pegawai Pada RSUD Sekayu Menggunakan Metode SMART. Adapun jadwal penelitian dapat dilihat pada Tabel 4.1 :

**Tabel 4.1. Jadwal Penelitian**

No	Aktifitas	Periode																
		September				Oktober				Nopember				Desem ber				
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
1	Analisis Kebutuhan <i>User</i>																	
2	Membuat <i>prototype</i>																	
3	Menyesuaika n <i>Prototype</i> Dengan Keinginan <i>User</i>																	
4	Membuat Sistem Baru																	
5	Melakukan <i>Testing</i>																	



### **4.3. Teknik Pengumpulan Data**

Dalam penulisan laporan ini penulis menggunakan beberapa metode teknik pengumpulan data, yang terdiri dari :

#### **4.3.1. Observasi (pengamatan)**

Menurut Nazir (2014:154), observasi adalah cara pengambilan data dengan menggunakan mata tanpa ada pertolongan alat standar lain untuk keperluan tersebut.

Observasi yang dilakukan penulis yaitu melakukan pengamatan secara langsung pada Rumah Sakit Umum Daerah (RSUD) Sekayu untuk mengetahui sistem penilaian karyawan yang dilakuakn pada RSUD Sekayu, hasil pengamatan penulis penilaian karyawan RSUD Sekayu masih menggunakan *Microsoft Excel* dalam pengolahan datanya.

#### **4.3.2. Interview (wawancara)**

Menurut Nazir (2014:170), wawancara adalah proses memperoleh keterangan untuk tujuan penelitian dengan cara tanya jawab, sambil bertatap muka antara si penanya atau pewawancara dengan si penjawab atau responden dengan menggunakan alat.

Wawancara dilakukan secara tanya jawab langsung kepada ibu Shinta Novita Lizza A yang mempunyai wewenang untuk memberikan data atau informasi yang dibutuhkan oleh penulis yaitu berupa informasi sistem penilaian kinerja pegawai pada RSUD Sekayu.

#### 4.3.3. Studi Pustaka

Menurut Nazir (2014:79), Studi kepustakaan adalah teknik pengumpulan data dengan mengadakan studi penelaah terhadap buku-buku, literatur-literatur, catatan-catatan, dan laporan-laporan yang ada hubungannya dengan masalah yang dipecahkan.

Studi pustaka yang dilakukan penulis yaitu membaca buku-buku perpustakaan dan toko-toko buku untuk mengutip dari buku yang penulis baca tersebut sebagai pelengkap teori-teori pendukung yang bersangkutan dengan topik penelitian.

#### 4.3.4. Dokumentasi

Menurut Jogiyanto (2008:121), dokumentasi adalah hal yang sangat penting dilakukan karena akan menjadi acuan tahapan operasi dan pemeliharaan. Teknik pengumpulan data dengan dokumentasi yaitu dengan cara mengumpulkan dokumen-dokumen yang berkaitan dengan penyusunan laporan skripsi.

Pada metode ini penulis mendapatkan dokumen-dokumen yang terkait dari bagian *Human Resources Development (HRD)* pada Rumah Sakit Umum Daerah (RSUD) Sekayu seperti profile, sejarah, struktur organisasi, visi & misi serta data pegawai, dan data penilaian kinerja pegawai.

## 4.4. Alat Dan Teknik Pengembangan Sistem

### 4.4.1. Alat Pengembangan Sistem

Menurut Simarmata (2010:223), *UML-based Web Engineering* (UWE) merupakan pendekatan metode-metode logis untuk pengembangan aplikasi Web yang berdasarkan pada *unified Process* (Jacobsen, dkk, 2000). UWE terutama berdasarkan pada konsep yang paling berhubungan dan disediakan oleh metode-metode lainnya, tetapi ditentukan oleh notasi UML (*UML profile*) yang direkatkan pada teknik diagram yang diajukan oleh UML dan menetapkan proses desain semiotomatis dan sistematis.

#### 1. *Use Case Diagram*

Menurut Rosa, AS dan Shalahuddin (2014:155), *Use case diagram* merupakan permodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah 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. Simbol-simbol *use case diagram* di tunjukkan pada tabel 4.2 di bawah ini:

Tabel 4.2. Elemen-elemen *use case diagram*

Simbol	Deskripsi
<p data-bbox="587 427 699 454"><i>use case</i></p> 	<p data-bbox="890 421 1362 667">Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i>.</p>
 <p data-bbox="507 790 715 817">Actor atau Role</p>	<p data-bbox="890 674 1362 1048">Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang,tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan benda diawal frase nama aktor.</p>
<p data-bbox="507 1059 799 1086">Asosiasi / <i>Association</i></p> 	<p data-bbox="890 1059 1362 1216">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>
<p data-bbox="579 1232 799 1258"><i>Ekstensi / extend</i></p> 	<p data-bbox="890 1232 1362 1854">Relasi <i>use case</i> tambahan sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang samadengan <i>use case</i> yang ditambahkan; misal arah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi extend-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
<p data-bbox="528 1865 719 1937">Generalisasi / <i>generalization</i></p>	<p data-bbox="890 1865 1362 1977">Hubungan generalisasi dan spesialisasi (umum – khusus ) antara dua buah <i>use case</i> dimana fungsi</p>

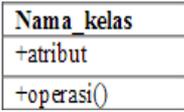
	yang satu adalah fungsi yang lebih umum dari lainnya.
<p><i>Menggunakan / include / uses</i></p> <p style="text-align: center;"><code>&lt;&lt;extend&gt;&gt;</code></p> <p style="text-align: center;"></p>	Relasi <i>use case</i> tambahan sebuah <i>use case</i> dimana <i>use case</i> yang ditambah memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

Sumber : Rosa, AS dan Shalahuddin (2014:156).

## 2. Class Diagram

Menurut Rosa (2014:141), Diagram *class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau oprasional. Simbol-simbol *class* diagram di tunjukkan pada tabel 4.3. di bawah ini:

**Tabel 4.3. Elemen-elemen *class* diagram**

Deskripsi	Simbol
Kelas pada struktur sistem.	Kelas / <i>class</i> : 
Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.	Antar muka / <i>Interface</i> :  <b>Nama_interface</b>
Relasi antara kelas dengan makna umum, sosisasi biasanya disertai dengan <i>multiplicity</i> .	Assosiasi / <i>Association</i> : 
Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain asosiasi biasanya juga disertai dengan <i>multiplicity</i> .	Asosiasi berarah / <i>Directed Association</i> : 
Relasi antar kelas dengan makna generalisasi-spesialisasi (umum	Generalisasi :

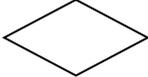
husus)	
Relasi anatar kelas dengan makna kebergantungan antar kelas.	Kebergantungan / antarkelas 
Relasi antar kelas dengan makna semua bagian.	Agregasi / <i>Aggregatio</i> : 

Sumber : Rosa, AS dan Shalahuddin (2014:146)

### 3. Activity Diagram

Menurut Rosa (2014:161), Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran 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. Simbol-simbol *activity* diagram di tunjukkan pada tabel 4.4:

#### 4. Tabel 4.4. Simbol Class Diagram

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah setatus awal.
<i>aktivitas</i> 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
<i>Percabangan / decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
<i>Pengabungan / join</i> 	Asosiasi pengembangan dimana lebih dari satu aktivitas digabungkan menjadi satu.

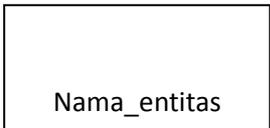
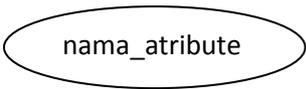
<p><i>Status akhir</i></p> 	<p>Status ahir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.</p>
--------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------

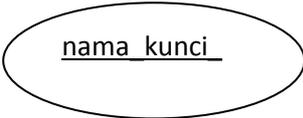
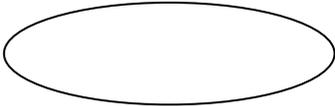
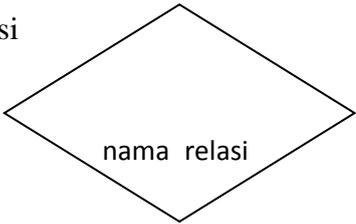
*Sumber* : Rosa, AS dan Shalahuddin (2014:162).

### 5. *Entity Relationship Diagram (ERD)*

Menurut Rosa A.S (2011:49), *Entity Relationship Diagram (ERD)* adalah bentuk paling awal dalam melakukan perancangan basis data relasional. Jika menggunakan OODMBS maka perancangan ERD tidak perlu dilakukan. Simbol-simbol *entity relationship diagram (ERD)* ditunjukkan pada tabel 4.5

**Tabel 2.3. Simbol – simbol *Entity Relationship Diagram (ERD)***

Simbol	Keterangan
<p><i>Entitas / entity</i></p> 	<p>Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data</p>
<p>Atribut</p> 	<p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas</p>

Simbol	Keterangan
<p>Atribut kunci primer</p> 	<p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id</p>
<p>Atribut multivali/<i>multivalue</i></p> 	<p><i>Field</i> atau kolom data yang butuh disimpan dalam entitas yang dapat dimiliki nilai lebih dari satu</p>
<p>Relasi</p> 	<p>Relasi yang menghubungkan antarentitas; biasanya diawali dengan kata kerja</p>
<p>Asosiasi / <i>association</i></p> 	<p>Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian</p>

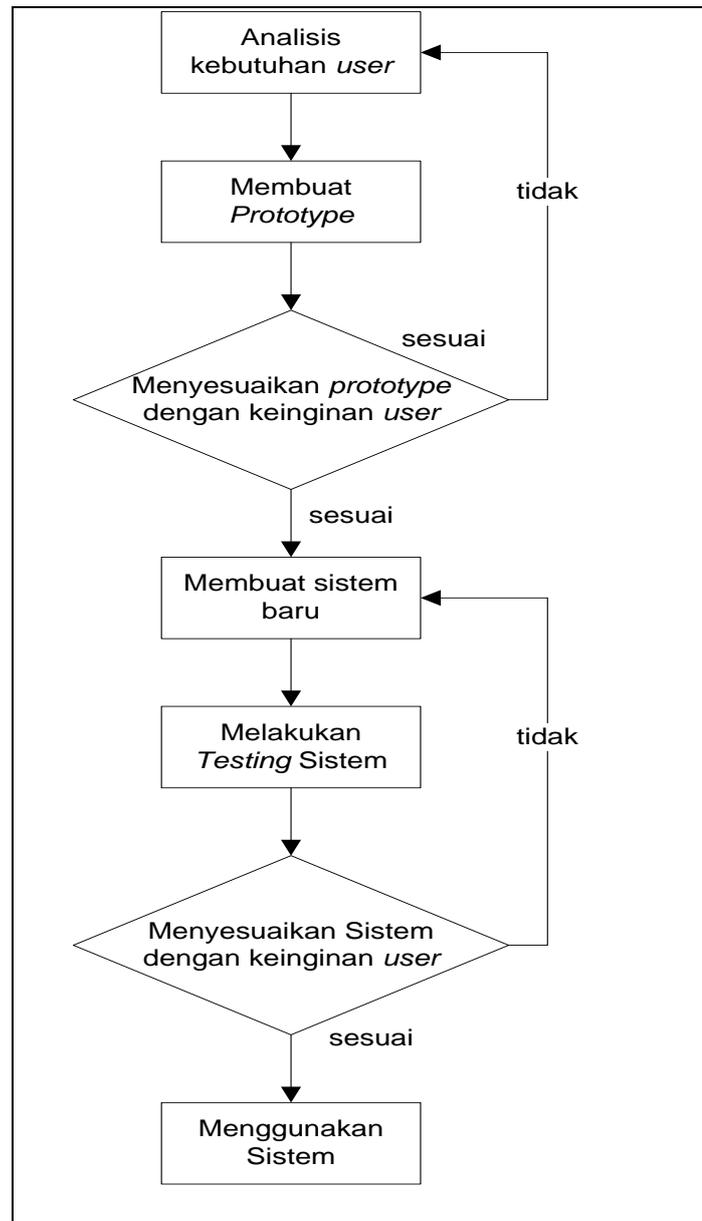
Sumber : Rosa, AS dan Shalahuddin (2011:49)

#### 4.4.2. Metode Pengembangan Sistem

Metode pengembangan sistem yang dipakai penulis untuk merancang sistem pendukung keputusan ini menggunakan metode *prototype*, metode ini merupakan salah satu metode yang banyak digunakan dalam pengembangan perangkat lunak.

Menurut Mulyani (2016:26), *prototyping* merupakan teknik pengembangan sistem yang menggunakan *prototype* untuk menggambarkan sistem, sehingga pengguna atau pemilik sistem mempunyai gambaran pengembangan sistem yang akan dilakukannya. Metode *prototype* terbagi menjadi 2 (dua) tipe yaitu *Evolutionary Prototype* dan *Requirement Prototype*.

*Requirement Prototype* merupakan *Prototype* yang dibangun oleh pengembang dengan mendefinisikan fungsi dan prosedur sistem dimana pengguna atau pemilik sistem tidak bisa mendefinisikan sistem tersebut. Pada penelitian ini Untuk membangun aplikasi yang telah direncanakan, penulis menggunakan *prototype* tipe *Requirement Prototype* seperti diperlihatkan pada gambar 4.1 berikut ini :



(Sumber: Mulyani,2016)

**Gambar 4.1 Model *Requirement Prototype***

Ini Menunjukkan pendekatan *prototype* pada umumnya dan melibatkan beberapa langkah berikut :

1. Analisis kebutuhan *user*

Pada tahapan ini peneliti melakukan identifikasi masalah yang terdapat pada RSUD Sekayu dan melanjutkan membuat alur sistem penilaian kinerja pegawai yang dilakukan RSUD Sekayu dengan menggunakan *flowchart*, dan pada tahap terakhir peneliti menganalisis kebutuhan apa saja yang di inginkan setiap pengguna.

2. Membuat *prototype*

Pada tahapan ini peneliti membuat *prototype* dari hasil analisis yang telah dilakukan pada tahap sebelumnya yaitu :

- a. Membuat rancangan *use case diagram* yang menjelaskan kegiatan yang ada pada setiap pengguna.
- b. Membuat rancangan *activity diagram* yang menjelaskan aktivitas-aktivitas yang dilakukan setiap pengguna sistem dengan terperinci.
- c. Membuat rancangan *class diagram* untuk mendukung perancangan aplikasi yang akan dibangun. Peneliti menggambarkan class dan atribut yang dimiliki oleh setiap class serta hubungan pada sistem.
- d. Membuat desain database atau rancangan tabel-tabel yang dibutuhkan untuk membangun sistem.

e. Membuat desain interface aplikasi, pada tahap ini peneliti menggambarkan setiap tampilan aplikasi yang dibangun.

3. Menyesuaikan *prototype* dengan keinginan *user*

Pada tahap ini peneliti akan membuat *pototype* baru apabila terjadi kesalahan pada *prototype* yang dibangun sesuai dengan keinginan *user*.

4. Membuat sistem baru

Pada tahapan ini peneliti mulai membangun sistem yang sudah di setujui oleh pihak RSUD Sekayu pada tahapan sebelumnya.

5. Melakukan *testing* sistem

Pada tahapan ini peneliti melakukan testing dengan menggunakan metode black box, pada setiap sistem yang dibangun apakah sudah sesuai dengan fungsinya.

6. Menyesuaikan sistem dengan keinginan *user*

Pada tahapan ini peneliti menyesuaikan sistem yang sudah dibangun dengan keinginan user apakah terdapat kekurangan atau tidak.

7. Menggunakan sistem

Pada tahapan ini sistem yang dibangun siap digunakan oleh *user*.

## **4.5. Teknik Pengujian Sistem**

### **4.5.1. *Black Box***

Pada tahapan ini penulis melakukan pengujian mulai dari proses login sampai proses menyimpan data ke dalam *database*, tabel yang diuji diantaranya sebagai berikut: tabel data pegawai, tabel data penilaian pegawai, tabel pegawai sudah direnking.