

**BAB IV**  
**METODOLOGI PENELITIAN**

**4.1. Jadwal dan tempat penelitian.**

**4.1.1. Jadwal Penelitian**

No.	URAIAN	Maret				April				Mei				Juni	
		Minggu Ke-													
		1	2	3	4	1	2	3	4	1	2	3	4	1	
1.	Identifikasi Kebutuhan														
2.	Membuat Prototype														
3.	Menguji Prototype														
4.	Pengkodean Sistem														
5.	Pengujian Sistem														

**4.1.2. Tempat Penelitian**

Riset ini dilakukan di PT SUMBER BAKTI UTAMA PALEMBANG. Perusahaan ini berlokasi di Jalan Tanjung Sari II No. 68 Bukit Sangkal Palembang.

## **4.2. Jenis Data**

### **4.2.1.Data Primer**

Menurut Sugiyono (2010:137), menyatakan bahwa Sumber primer adalah sumber data yang langsung memberikan data kepada pengumpulan data. Data yang dikumpulkan dan diolah sendiri atau seorang atau suatu organisasi langsung dari obyeknya. Dalam hal ini data primer diperoleh langsung dari PT Sumber Bakti Utama Palembang. Data yang dikumpulkan seperti calon data tenaga kerja , data keluar, data masuk , data keahlian, Pengalaman dan data seleksi .

### **4.2.2. Data Sekunder**

Menurut Sugiyono (2010:137), Data Sekunder adalah sumber data yang tidak langsung memberikan data kepada pengumpulan data, misalnya lewat orang lain atau lewat dokumen. Data yang diperoleh dalam bentuk sudah jadi, sudah dikumpulkan dan diolah pihak lain (biasanya sudah dipublikasikan). Data tersebut diperoleh dari PT Sumber Bakti Utama Palembang berupa Sejarah singkat, visi, dan misi, struktur organisasi, pembagian tugas dan wewenang.

## **4.3. Teknik Pengumpulan Data**

Adapun cara-cara yang dipakai penulis guna untuk pengumpulan data pada PT Sumber Bakti Utama Palembang adalah sebagai berikut :

#### **4.3.1 Observasi**

Menurut Sutabri (2012:97), merupakan teknik pengumpulan datadengan langsung melihat kegiatan yang dilakukan oleh user. Salah satu keuntungan dari pengamatan langsung atau observasi ini adalah bahwa sistem analis dapat lebih mengenai lingkungan fisik seperti tata letak ruangan serta peralatan dan formulir yang digunakan serta sangat membantu untuk melihat proses bisnis beserta kendala-kendalanya.

Pengamatan yang dilakukan secara langsung pada PT. Sumber Bakti Utama untuk mendapatkan data yang diperlukan seperti prosedur penyedia tenaga kerja.

#### **4.3.2 Wawancara (*Interview*)**

Menurut Sutabri (2012:89), adalah suatu teknik yang paling singkat untuk mendapatkan data, namun sangat tergantung pada kemampuan pribadi sistem analisis untuk dapat memanfaatkannya.

Pada metode ini penulis melakukan wawancara langsung dengan ibu Mardinatilopa, A.Md selaku staf bagian Personalia di PT. Sumber Bakti Utama guna mendapatkan informasi alur kerja sistem, permasalahan sistem sebelumnya dan apa yang dibutuhkan pada sistem yang akan dibangun .

#### **4.3.3 Dokumentasi**

Menurut Sugiyono (2013:204), dokumentasi merupakan catatan peristiwa yang sudah berlalu. Dokumen ini bisa berbentuk tulisan, gambar, atau karya-karya monumental dari seorang. Dokumen yang

berbentuk tulisan misalnya catatan harian, sejarah kehidupan (*life histories*), cerita, biografi, peraturan, kebijakan. Dokumen yang berbentuk gambar misalnya foto, gambar hidup, sketsa dan lain-lain.

Pada metode ini penulis juga melakukan dokumentasi yaitu dengan cara mengumpulkan data dari PT Sumber bakti utama seperti data calon tenaga kerja yang masuk dan keluar, Struktur organisasi dan profil perusahaan .

#### **4.3.4 Studi Pustaka**

Menurut Purnama (2010), studi kepustakaan merupakan metode pengumpulan data dengan cara mempelajari literature, paket modul dan panduan, buku-buku pedoman, buku-buku perpustakaan dan segala kepustakaan lainnya yang dianggap perlu dan mendukung. Cara ini Penulis lakukan dengan menggunakan buku serta jurnal sebagai referensi dan informasi untuk memperoleh konsep serta pengetahuan yang relevan dengan masalah yang akan diteliti.

#### **4.4.4. Metode Pengembangan**

Metode pengembangan sistem yang digunakan yaitu *Prototype*. Adapun tahapan-tahapan *Prototype* yang dilakukan dalam membangun aplikasi ini sebagai berikut;

##### **a. Identifikasi kebutuhan pemakai**

Tahap ini adalah tahap awal melakukan komunikasi dengan admin untuk mengidentifikasi kebutuhan yang diperlukan untuk

membangun sistem. Kebutuhan ini berupa data lowongan kerja, data rekrutmen, data penempatan

*b. Membangun Prototyping*

Setelah mendapatkan data, langkah selanjutnya membuat UML, Use Case, Diagram Activity, Dan Diagram class sebagai gambaran sistem yang akan dibangun.

*c. Menguji Prototyping*

Setelah membangun prototyping, maka gambaran sistem yang akan dibangun dievaluasi dengan tujuan untuk mendapatkan saran dan masukan dari pemakai. Jika pada tahap ini masih ada revisi maka harus dilakukan perbaikan.

*d. Pengkodean system*

Dalam tahap ini prototyping yang sudah disepakati selanjutnya diterjemahkan ke dalam bahasa pemrograman yang sesuai yaitu PHP dan database menggunakan MySQL.

*e. Pengujian system*

Setelah selesai di bangun dengan pemrograman, selanjutnya dilakukan proses pengujian sistem menggunakan metode *blackbox* dilakukan oleh admin.

#### **4.4. Jenis Penelitian**

##### **4.4.1. Penelitian Terapan**

Menurut Koesnaedi (2014: 82), Penelitian terapan adalah penelitian yang diarahkan untuk mendapatkan informasi yang dapat digunakan untuk memecahkan masalah. Penelitian terapan dilakukan dengan tujuan menerapkan, menguji, dan mengevaluasi masalah-maslah praktis sehingga dapat dimanfaatkan untuk kepentingan manusia, baik secara individual maupun kelompok. Masalah penelitian terapan ditetapkan untuk mencari solusi yang dapat dimanfaatkan manusia. Oleh karena itu, hasil pnelitiannya berupa jawaban nyata dan dapat dirasakan langsung oleh masyarakat yang dituju. Penelitian terapan dalam pendidikan, misalnya, berkaitan dengan peningkatan kualitas strategi, teknik, dan model pembelajaran, atau peningkatan minat dan motivasi masyarakat

#### **4.5. Alat dan Teknik Pengembangan Sistem**

##### **4.5.1. Alat Pengembangan Sistem**

Menurut Rosa A.S (2013:139), Pada perkembangan teknologi perangkat lunak, diperlukan adanya bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak.

Pada Perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik

pemrograman berorientasi objek, yaitu UML. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.

UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks- teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

#### **4.5.1.1 Diagram *Use Case* (*Use case Diagram*)**

Menurut Rosa A.S (2011 : 130), *use case* atau diagram *use case* merupakan pemodelan untuk melakukan (*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.

Adapun keterangan dari simbol – simbol *use case* dapat dilihat pada Tabel 4.2 sebagai berikut:

Tabel 5.2. Simbol *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Actor</i>	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 di awal frase nama aktor.
2.		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
3.		<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.
4.		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> .
5.		<i>Association</i>	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
6.		<i>Use Case</i>	Fungsionalitas yang disediakan oleh sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. Biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .

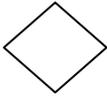
(Sumber : Rosa, 2011)

#### 4.5.1.2. Diagram Aktivitas (Activity Diagram)

Menurut Rosa A.S (2011 : 134), diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses sebuah 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.

Adapun keterangan dari simbol – simbol *activitydiagram* dapat dilihat pada Tabel 5.4 sebagai berikut.

**Tabel 5.4. Simbol Activity Diagram**

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Activity</i>	Aktivitas yang dilakukan sistem, aktivitas biasanya diawal dengan kata kerja.
NO	GAMBAR	NAMA	KETERANGAN
2.		<i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
3.		<i>Initial Node</i>	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
4.		<i>Activity Final Node</i>	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
5.		<i>Pengabungan</i> / <i>Join</i>	Asosiasi pengabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

(Sumber : Rosa, 2011)

#### 4.5.1.3. Diagram kelas (*Class Diagram*)

Menurut Rosa A.S (2011:122), diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Adapun keterangan dari symbol-simbol class diagram dapat dilihat pada Tabel 5.1 sebagai berikut :

**Tabel 5.1. Simbol *Class Diagram***

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Generalization</i>	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
2.		<i>NaryAssociation</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Class</i>	Kelas pada struktur sistem
4.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7.		<i>Association</i>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>

(Sumber : Rosa, 2011)

## 4.5. Teknik Pengembangan Sistem

Berikut adalah teknik pengembangan sistem yang penulis gunakan. Penulis menggunakan metodologi prototype.

### 4.5.1. Metode pengembangan sistem yang digunakan yaitu *Prototype*.

Adapun tahapan-tahapan *Prototype* yang dilakukan dalam membangun aplikasi ini sebagai berikut;

#### a. Identifikasi kebutuhan pemakai

Tahap ini adalah tahap awal melakukan komunikasi dengan admin untuk mengidentifikasi kebutuhan yang diperlukan untuk membangun sistem..

#### b. Membangun *Prototyping*

Setelah mendapatkan data, langkah selanjutnya membuat UML use case dan diagram activity sebagai gambaran sistem yang akan dibangun.

#### c. Menguji *Prototyping*

Setelah membangun prototyping, maka gambaran sistem yang akan dibangun dievaluasi dengan tujuan untuk mendapatkan saran dan masukan dari pemakai. Jika pada tahap ini masih ada revisi maka harus dilakukan perbaikan.

d. Pengkodean system

Dalam tahap ini prototyping yang sudah disepakati selanjutnya diterjemahkan ke dalam bahasa pemrograman yang sesuai yaitu PHP dan database menggunakan MySQL.

e. Pengujian system

Setelah selesai di bangun dengan pemrograman, selanjutnya dilakukan proses pengujian sistem menggunakan metode *blackbox* dilakukan oleh admin.

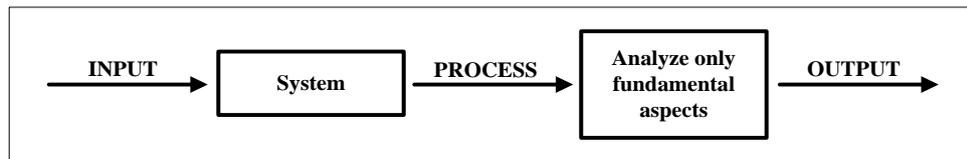
#### 4.6. Alat dan Teknik Pengujian

##### 4.6.1. Pengujian Kotak Hitam (*Black Box Testing*)

Menurut Roger S. Pressman (2010:75), Black Box Testing atau pengujian kotak hitam berfokus pada persyaratan fungsional dari perangkat lunak. Black box testing bukan merupakan alternatif dari pengujian white box testing. Sebaliknya, black box testing adalah pendekatan komplementer yang mungkin untuk mengungkap kelas yang berbeda dari kesalahan daripada metode white box testing.

Pengujian pada *Black Box* berusaha menemukan kesalahan seperti :

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.



*(Sumber: Santoso 2014)*

**Gambar 4.2 Teknik Pengujian *Black Box***