

## BAB II

### LANDASAN TEORI DAN TINJAUAN PUSTAKA

#### 2.1 Ayat Al-Qur'an yang Berkaitan dengan Penelitian

Bekerja adalah bentuk amalan ibadah yang memiliki nilai lebih dimata Allah SWT. Kerena dengan bekerja, kita menunjukkan usaha kita untuk mendapatkan rezeki sebagaimana yang telah diatur oleh Allah SWT.

Berikut ayat Al-Quran yang berhubungan dengan bekerja :

وَقُلْ أَعْمَلُوا فَسَيَرَى اللَّهُ عَمَلَكُمْ وَرَسُولُهُ وَالْمُؤْمِنُونَ وَسَتُرَدُّونَ إِلَىٰ عِلْمِ الْغَيْبِ  
وَالشَّهَادَةِ فَيُنَبِّئُكُم بِمَا كُنتُمْ تَعْمَلُونَ ١٠٥

Artinya : *Dan katakanlah, “Bekerjalah kamu, maka Allah akan melihat pekerjaanmu, begitu juga Rasul-Nya dan orang-orang Mukmin, dan kamu akan dikembalikan kepada (Allah) yang mengetahui yang gaib dan yang nyata, lalu diberitakan-Nya kepada kamu apa yang telah kamu kerjakan.”* (QS. At-Taubah [9]:105).

وَلَقَدْ مَكَّنَّاكُمْ فِي الْأَرْضِ وَجَعَلْنَا لَكُمْ فِيهَا مَعِيشًا قَلِيلًا مَّا تَشْكُرُونَ ١٠

Artinya : *“Dan sungguh, Kami telah menempatkan kamu di bumi dan disana Kami sediakan (sumber) penghidupan untukmu. (Tetapi) sedikit sekali kamu bersyukur.”* (QS. Al-A'raf [7]:10).

فَإِذَا قُضِيَتِ الصَّلَاةُ فَانْتَشِرُوا فِي الْأَرْضِ وَابْتَغُوا مِن فَضْلِ اللَّهِ وَاذْكُرُوا اللَّهَ  
كَثِيرًا لَّعَلَّكُمْ تُفْلِحُونَ ١٠

Artinya : *“Apabila shalat telah dilaksanakan, maka bertebaranlah kamu di bumi; carilah karunia Allah dan ingatlah Allah sebanyak-banyaknya agar kamu beruntung.”* (QS. Al-Jumu'ah [62]:10).

Ayat ini menjelaskan bahwa perintah bekerja telah diwajibkan semenjak nabi pertama, Adam as sampai nabi yang terakhir, Muhammad SAW. Perintah ini tetap berlaku kepada semua orang tanpa membeda-bedakan pangkat, status dan jabatan seseorang. Islam akan membukakan pintu kerja bagi setiap muslim agar ia dapat memilih pekerjaan yang sesuai dengan minatnya dan kemampuannya. Namun demikian masih banyak orang yang enggan untuk bekerja dan berusaha dengan alasan bertawakal kepada Allah SWT serta menunggu-nunggu rizki dari langit. Mereka telah salah memahami ajaran islam. Pasrah kepada Allah tidak berarti meninggalkan amal berupa bekerja. Seperti yang pernah rasul katakan : *“semaikanlah benih, kemudian mohonkanlah buah dari Rabbmu.”*

Allah memang telah berjanji akan memberikan rizki kepada semua makhluk-Nya. Akan tetapi janji ini seperti “cek kosong”, seseorang akan mendapatkan rizki kalau mau berusaha, berjalan dan bertebaran di penjuru-penjuru bumi. Karena Allah menciptakan bumi dan seisinya untuk kemakmuran manusia. Siapa yang mau berusaha dan bekerja ialah akan mendapat rizki dan rahmat dari Allah.

Hubungan dari semua ayat Al-Quran yang telah diuraikan dengan penelitian yaitu yang dilakukan adalah suatu pekerjaan haruslah dilakukan sesuai prosedur dan aturan yang berlaku agar mendapat hasil yang baik dari pekerjaan tersebut. Dan dengan adanya sistem kepegawaian para pegawai bisa memiliki kemampuan maksimal dalam menyelenggarakan tugas, fungsi dan kewajiban kepegawaian.

## **2.2 Teori Yang Berkaitan Dengan Sistem Informasi yang akan dibangun**

### **2.2.1 Sistem**

Menurut Nugroho (2010:17), Sistem dapat diartikan sebagai sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Hal pertama yang perlu diperhatikan dalam suatu sistem adalah elemen-elemennya. Tentunya setiap sistem memiliki elemen-elemennya sendiri, yang kombinasinya berbeda antara sistem yang satu dengan sistem yang lain. Namun demikian, susunan dasarnya tetap sama. Sedangkan Menurut Jugiyanto (2005), mendefinisikan sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

Maka dari itu dapat disimpulkan bahwa sistem adalah sekelompok unsur atau elemen yang berhubungan satu dengan yang lain untuk mencapai suatu tujuan.

#### **2.2.1.1 Karakteristik Sistem**

Menurut Jugiyanto (2005:3), suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu :

##### **a. Komponen Sistem**

komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem. Setiap subsistem mempunyai sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

b. Batasan Sistem

Batas sistem (*Boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya atau dengan lingkungan luarnya. Batas sistem dapat dipandang sebagai satu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

c. Lingkungan Luar Sistem

Lingkungan luar (*Environment*), dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi sistem. Lingkungan luar bersifat menguntungkan yang memberikan energi pada sistem sehingga harus selalu dijaga dan dipelihara. Selain itu lingkungan luar juga dapat merugikan sistem. Untuk dapat mempertahankan kelangsungan hidup sistem, maka lingkungan yang seperti ini harus dapat dikendalikan.

d. Penghubung Sistem

Merupakan media penghubung antara sub-sistem dengan sub-sistem lainnya. Melalui penghubung sumber-sumber daya dapat mengalir dari sub-sistem ke sub-sistem lainnya, sehingga saling berintegrasi membentuk satu kesatuan.

e. Masukan Sistem (*Input*)

Masukan adalah energi yang dimasukkan ke dalam sistem, dapat berupa masukan perawatan dan masukan signal. Masukan perawatan (*maintenance input*) adalah energi yang dimasukkan agar sistem tersebut dapat beroperasi. Masukan signal (*signal input*) adalah energi yang diproses untuk mendapatkan keluaran.

f. Keluaran Sistem (*Output*)

Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

g. Pengolah Sistem

Suatu sistem mempunyai suatu bagian pengolah yang dapat merubah masukan menjadi keluaran. Sasaran sistem, suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

### 2.2.2 Informasi

Menurut Jogiyanto (2005:8), Informasi dapat didefinisikan sebagai data yang diolah menjadi bentuk yang lebih beragam atau lebih berarti bagi pemakainya dan digunakan sebagai dasar untuk pengambilan keputusan. Sumber informasi adalah data. Data merupakan kenyataan dari kejadian-kejadian yang nyata.

Sedangkan menurut Nugroho (2010:17), Informasi merupakan salah satu elemen dalam manajemen perusahaan. Agar informasi dapat mengalir lancar, para manajer perlu menempatkan informasi dalam suatu kerangka sistem.

Informasi yang baik harus memenuhi kualifikasi yang antara lain :

1. Akurat

Maksud akurat ialah informasi tersebut bebas dari kesalahan dan bebas dari bias, bebas dari kesalahan berarti bahwa informasi tersebut benar-benar menyatakan apa yang harus dinyatakan. Bebas dari bias berarti bahwa informasi tersebut teliti.

## 2. Tepat Waktu

Jelas informasi harus diberikan pada waktu yang tepat. Informasi yang sudah kadaluarsa hanya bernilai sampah, sekalipun informasinya sama dan tidak berubah.

## 3. Relevan

Artinya bahwa informasi tersebut benar-benar sesuai kebutuhan pihak yang membutuhkan informasi. Misalnya, untuk menghitung dosis obat seorang pasien, dokter membutuhkan informasi mengenai berat badan pasien. Jadi, berat badan pasien adalah informasi yang relevan, sedangkan informasi tentang tinggi badan pasien bukanlah informasi yang relevan.

Maka dari itu dapat disimpulkan bahwa informasi adalah suatu data yang diolah untuk digunakan dalam proses pengambilan keputusan menjadi bentuk yang lebih berguna.

### **2.2.3 Sistem Informasi**

Menurut Nugroho (2010:17), Sistem informasi dapat didefinisikan sebagai integrasi antara orang, data, alat dan prosedur yang bekerja sama dalam mencapai suatu tujuan. Jadi, di dalam sistem informasi terdapat elemen orang, data, alat dan prosedur atau cara.

Menurut Al-Bahra (2013:14), Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial dan kegiatan strategis dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan–laporan yang diperlukan.

Sedangkan menurut Jugiyanto (2005:11), Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Dari berbagai definisi Sistem informasi menurut para ahli dapat disimpulkan bahwa Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dan kegiatan dari prosedur-prosedur yang diorganisasikan, bilamana dieksekusi akan menyediakan informasi untuk mendukung pengambilan keputusan.

#### **2.2.4 Kepegawaian**

Menurut Widjaja (2006:113) Pegawai adalah tenaga kerja manusia jasmaniah maupun rohaniah (mental dan pikiran) yang senantiasa dibutuhkan dan oleh karena itu merupakan salah satu modal pokok dalam usaha kerja sama untuk mencapai tujuan tertentu (organisasi).

Dari definisi diatas dapat diketahui bahwa pegawai merupakan modal pokok dalam suatu organisasi, baik itu organisasi pemerintahan maupun organisasi swasta. Dikatakan bahwa pegawai merupakan modal pokok dalam suatu organisasi karena berhasil tidaknya suatu organisasi dalam mencapai tujuannya tergantung pada pegawai yang memimpin dalam melaksanakan tugas-tugas yang ada dalam organisasi tersebut.

Manajemen kepegawaian lazim disebut personel management tata personalia atau pembinaan, sebab walaupun istilah-istilah tersebut nampaknya

berbeda namun pengertiannya sama. M. Manullang mendefinisikan mengenai pengertian manajemen kepegawaian dalam bukunya yang berjudul “ Dasar-Dasar Manajemen”, bahwa manajemen kepegawaian (personel manajemen) adalah seni dan ilmu perencanaan, pelaksanaan dan pengontrolan tenaga kerja untuk tercapainya tujuan yang telah ditentukan terlebih dahulu dengan adanya kepuasan hati pada diri para pegawai” . (Manullang, 11:2008)

Menurut wijadja (2006) Bagian kepegawaian adalah departemen pelayanan yang membantu para karyawan dan pimpinan organisasi. Fungsi bagian kepegawaian antara lain :

- a. Melaksanakan rencana penerimaan, penempatan dan pemberhentian.
- b. Mengolah pegawai sesuai kebutuhan organisasi.
- c. Melakukan pengawasan terhadap disiplin pegawai.
- d. Melakukan evaluasi semua kegiatan sesuai rencana.
- e. Memberikan laporan kegiatan sesuai hasil evaluasi.

### **2.2.5 Sistem Informasi Kepegawaian**

Berdasarkan Keputusan Menteri Dalam Negeri Nomor 17 tahun 2000 di jelaskan bahwa, sistem informasi kepegawaian (SIMPEG) adalah suatu totalitas yang terpadu dari perangkat pengolah meliputi pengumpulan prosedur, tenaga pengolah dan perangkat lunak; perangkat penyimpanan meliputi pusat data dan bank data serta perangkat komunikasi yang saling berkaitan, saling ketergantungan dan saling menentukan dalam rangka penyediaan informasi dibidang kepegawaian.



## 2.3 Metode Pengembangan Sistem

Menurut Pressman (2012:29) Software (perangkat lunak) adalah elemen kunci dalam evolusi sistem dan produk berbasis komputer. Selama lebih dari 50 tahun terakhir, perangkat lunak telah berkembang lebih jauh pemecahan masalah khusus dan dan alat analisis informasi untuk industri.

Rekayasa perangkat lunak mencakup proses, metode, dan alat-alat yang digunakan dalam membangun sistem secara tepat waktu dan berkualitas. Proses perangkat lunak mencakup lima kegiatan yaitu komunikasi, perencanaan, permodelan konstruksi dan peyerahan perangkat lunak ke pelanggan.

### 2.3.1 Agile Methods

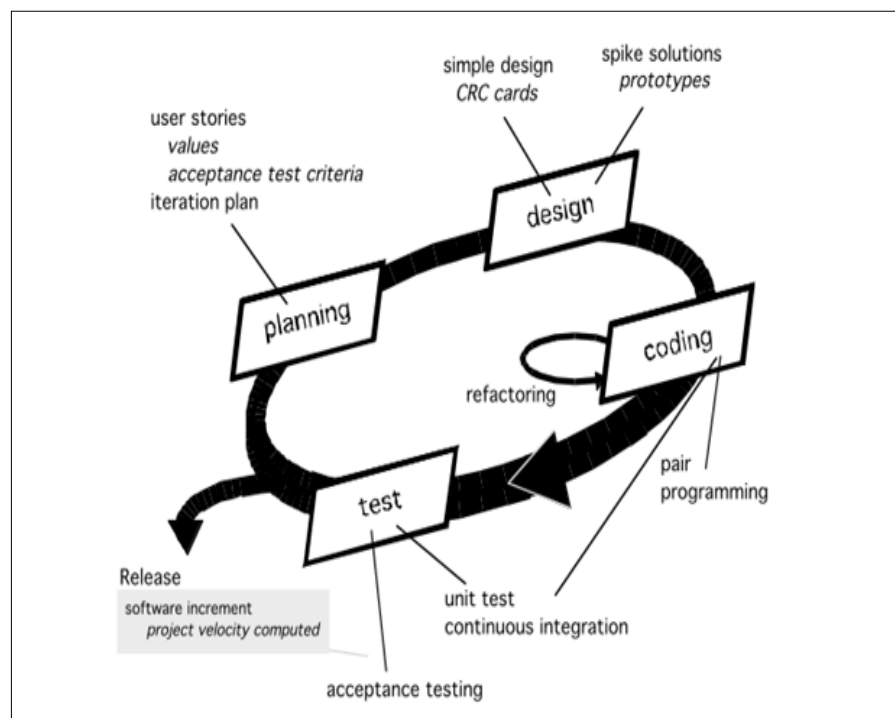
Menurut Pressman (2012:77), *Agile Development Methods* adalah sekelompok metodologi pengembangan perangkat lunak yang didasarkan pada prinsip-prinsip yang sama atau pengembangan sistem dalam jangka pendek yang memerlukan proses adaptasi (pada proyek dan kondisi-kondisi teknik yang berubah-ubah dengan cepat), rekayasawan harus mampu memberikan mekanisme yang cukup realitis untuk mendorong adanya disiplin yang seperlunya, atau harus dikategorikan sedemikian rupa sehingga menunjukkan “toleransi” bagi orang yang melakukan rekayasa perangkat lunak.

Menurut Pressman (2012:107), Metode *agile* menekankan empat isu utama yaitu pentingnya mengorganisasi tim mandiri yang memiliki kendali atas pekerjaan yang mereka lakukan, komunikasi dan kolaborasi antar anggota tim, pengakuan bahwa perubahan pada dasarnya merupakan sebuah kesempatan, dan menekankan pada penghantaran secara cepat dari perangkat lunak yang memuaskan pelanggan.

### 2.3.2 *Extreme Programming*

Menurut pressman (2012:86) *Extreme programming* adalah salah satu metodologi pengembangan cepat atau biasa disebut AGILE dalam pengembangan sistem informasi. Beck [Be04a] mendefinisikan lima nilai-nilai yang membentuk dasar bagi semua pekerjaan yang dibentuk sebagai bagian *Extreme programming* yaitu: komunikasi, kesederhanaan, umpan balik, keberanian, dan rasa hormat. Masing-masing nilai yang penting ini digunakan sebagai pendorong untuk kegiatan, tindakan, dan tugas XP yang spesifik.

*Extreme programming* menggunakan suatu pendekatan berorientasi objek sebagai paradigma yang diinginkan. Ada empat tahap pengembangan sistem dalam metodologi extreme programming digambarkan pada **Gambar 2.1**



Sumber : (Pressman, 2012:88)

**Gambar 2.1 Model *Extreme Programming***

Gambar ini menampilkan serangkaian tahapan pengembangan mengenai tahapan pada metode yang digunakan pada penelitian, yaitu:

1. Perencanaan (*Planning*).

Kegiatan perencanaan biasanya dimulai dengan mendengarkan, suatu kegiatan yang bertujuan untuk mengumpulkan kebutuhan-kebutuhan yang diperlukan untuk perangkat lunak yang akan dikembangkan. Aktivitas-aktivitas mendengarkan pada dasarnya mengarah kepada pembuatan serangkaian “cerita” (*user stories*) yang menggambarkan hasil yang diperlukan, fitur-fitur utama dan fungsionalitas.

2. Perancangan (*Design*).

Perancangan *extreme programming* dengan ketat mengikuti prinsip “tetap sederhana”. Perancangan XP akan memberikan panduan implementasi untuk suatu cerita ketika cerita ditulis, tidak kurang, tidak lebih. Rancangan-rancangan dan fungsionalitas tambahan (karena pengembang menganggap nantinya akan diperlukan) tidak terlalu disarankan. XP mendorong penggunaan kartu *CRC*, kartu *CRC* (*class responsibility collaborator*) digunakan untuk mengidentifikasi dan mengorganisasikan kelas berorientasi objek yang berkaitan dengan proses pengembangan perangkat lunak. Jika terdapat kesulitan untuk melakukan desain terhadap suatu cerita, XP menyarankan untuk membuat *prototype* dari desain tersebut. Disebut sebagai solusi *spike*, *prototype* rancangan diimplementasikan dan selanjutnya dievaluasi. Tujuannya adalah untuk mengurangi resiko-resiko yang akan timbul ketika pelaksanaan yang sesungguhnya dimulai. XP juga menyarankan *Refactoring*. *Refactoring* adalah proses mengubah sistem

perangkat lunak sedemikian rupa sehingga tidak mengubah perilaku eksternal kode, namun memperbaiki struktur internal yang ada di dalamnya.

### 3. Pengkodean (*Coding*)

Pada tahap ini, proses pengembangan tidak langsung melakukan implementasi terhadap desain yang telah dibuat, tetapi terlebih dahulu melakukan serangkaian unit pengujian untuk tiap-tiap cerita yang akan menjalankan setiap cerita yang disertakan pada rilis yang ada (peningkatan perangkat lunak). Setelah unit pengujian dibuat, pengembang perangkat lunak akan lebih mampu berkonsentrasi pada apa yang harus diimplementasikan supaya lurus dari unit pengujian tersebut. Tidak ada yang ditambahkan (tetap sederhana). Ketika kode-kode program telah selesai dituliskan, kode-kode program tersebut dapat langsung diuji menggunakan unit pengujian yang telah dirancang sebelumnya, sehingga bisa langsung memberikan umpan balik kepada para pengembang.

### 4. Pengujian (Test).

Pada tahap ini sistem yang telah dibangun dicoba apakah kebutuhan awal user atau user stories sudah dipenuhi dan apabila terpenuhi software siap dirilis.

## 2.4 Teori Yang Berhubungan Dengan Analisis Design

### 2.4.1 UML (*Unified Modeling Language*)

Permodelan adalah gambaran dari realita yang simpel dan dituangkan dalam bentuk pemetaan dengan aturan tertentu. UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mengidentifikasi requirement, membuat analisis dan desain, serta


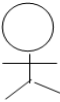

menggambarkan arsitektur dalam pemograman berorienttasi objek (Rosa & Shalahuddin, 2015:133).


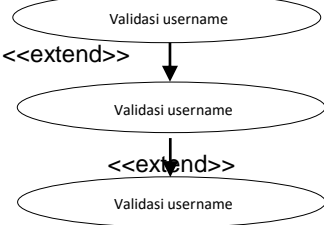

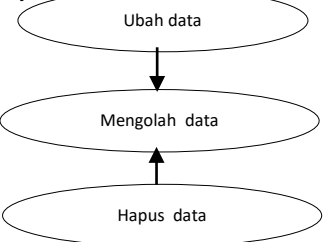

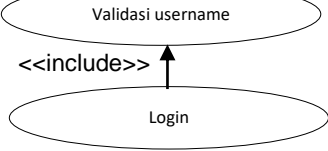
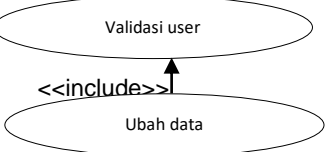
Pada perancangan sistem yang akan dibangun menggunakan *use case diagram*, *activty diagram* dan *class diagram*.

### 1. *Use case diagram*

Menurut Rosa dan Shalahuddin (2015:156), *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. *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Berikut adalah simbol-simbol yang ada pada *use case diagram* :

**Tabel 2.1** Tabel Simbol *Use Case Diagram*

No.	Simbol	Keterangan
1	<p><i>Use Case</i></p> 	Fungsionalitas yang disediakan sistem unit-unit yang saling bertukar pesan antara unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
2	<p>Aktor /actor</p> 	Orang atau 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: biasanya dinyatakan menggunakan kata benda di awal frase nama <i>Actor</i>
3	<p>Asosiasi/Association</p> 	Komunikasi antara aktor dengan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.

No.	Simbol	Keterangan
4	Ekstensi/ <i>extend</i>  <<extend>>  	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walaupun tanpa use case tambahan itu; mirip dengan prinsip inheritance pada pemograman berorientasi objek; biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, misal :  
5	Generalisasi/ <i>generalization</i>  	Hubungan generalisasi dan spesialisasi (umum – khusus ) antara dua buah Use Case dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya, misalnya :   <p>Arah panah mengarah pada use case yang menjadi generalisasinya (umum).</p>
6	Include  <<include>>  	Relasi use case tambahan ke sebuah use case di mana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini. Ada dua sudut pandang yang cukup besar mengenai include di use case : <ul style="list-style-type: none"> <li>• Include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan yang dijalankan misal pada kasus berikut :    </li> <li>• Include berarti use case yang tambahan akan selalu melakukan pengecekan apakah use case yang ditambahkan telah dijadikan sebelum use case tambahan dijalankan, misal pada kasus berikut :    </li> </ul> <p>Kedua interpretasi di atas dapat dianut salah satu keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>

Sumber : ( Rosa & Shalahuddin, 2015:156)



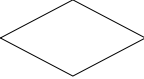

## 2. Activity diagram


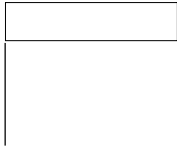
Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas mnggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi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

- a. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- b. Urutan atau pengelompokan tampilan sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan interface tampilan.
- c. Rancangan pengujian dimana setiap aktivias dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
- d. Rancangan menu yang ditampilkan pada perangkat lunak.

Berikut simbol-simbol yang ada pada *activity diagram* :

**Tabel 2.2** Tabel Simbol Activity Diagram

No	Simbol	Keterangan
1	Status Awal 	Status awal aktivitas system, sebuah diagram aktivitas memiliki sebuah awal
2	Aktivitas 	Aktivitas yang dilakukan system, aktivitas biasanyaawali dengan kata kerja
3	Percabangan/decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4	Penggabungan/ join 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

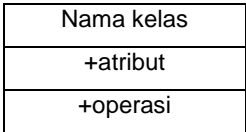





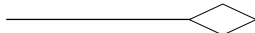
No	Simbol	Keterangan
5	Status Akhir 	Status Akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Sumber : (Rosa & Shalahuddin, 2015:162-163)

### 3. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut adalah simbol-simbol yang ada pada *class diagram* :

**Tabel 2.3** Tabel Simbol *Class Diagram*

No.	Simbol	Keterangan
1.	Kelas 	Kelas pada stuktur sistem
2.	Antarmuka/ interface  Nama_interface	Sama dengan konsep <i>interface</i> dalam pemograman <i>berorientasi objek</i>
3.	Asosiasi/ association 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4.	Asosiasi berarah/ <i>directed</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)
6.	Kebergantungan 	Kebergantungan antarkelas
7	Agregasi / aggregation 	Relasi antarkelas dengan makna semua-bagian (whole-part)

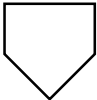
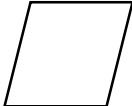

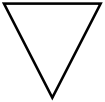
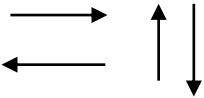
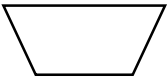

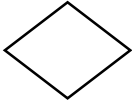
Sumber : (Rosa & Shalahuddin, 2014:141)


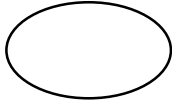


## 2.4.2 Flowchart

Menurut Jogiyanto, (2005:795) flowchart merupakan metode untuk menggambarkan tahap-tahap pemecahan masalah dengan mempresentasikan simbol-simbol tertentu dengan mudah dimengerti. Tujuan penggunaan flowchart adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi, dan jelas dengan menggunakan simbol-simbol yang standar, berikut Tabel 3.4 simbol-simbol Flowchart :

**Tabel 2.4** Simbol Flowchart

Simbol	Keterangan
	<b>Penghubung</b> Simbol untuk kelaur/masuk proses dalam lembar atau halaman lain
	<b>Input Output</b> Simbol yang menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya
	<b>Dokumen</b> Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak di kertas
	<b>On Line Storage</b> Simbol yang menyatakan bahwa data bahwa data di simbol ini akan disimpan
	<b>Simbol Garis Alir</b> Digunakan untuk menunjukkan arah selanjutnya yang akan di tuju dari simbol-simbol <i>flowchart</i>
	<b>Manual</b> Simbol yang menunjukan pengolahan yang tidak menggunakan komputer
	<b>Terminal</b> Simbol yang menunjukan untuk permulaan atau akhir suatu sistem
	<b>Kondisi</b> Simbol keputusan yang menunjukkan kondisi

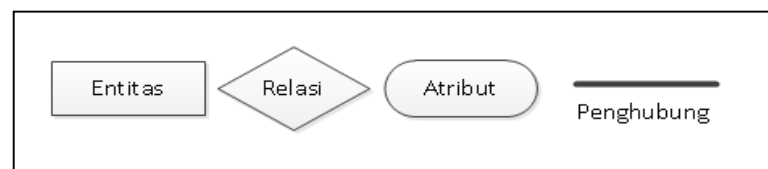
Simbol	Keterangan
	<b>Proses</b> Simbol yang menunjukkan pengolahan dilakukan oleh komputer
	<b>Penghubung</b> Simbol untuk keluar/masuk prosedur atau porses dalam lembar atau halaman yang masih sama

Sumber : (Jogiyanto, 2005:796)

### 2.4.3 ERD (*Entity Relationship Diagram*)

*ERD* adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. *ERD* merupakan model jaringan data yang menekankan pada struktur-struktur dan relationship data. Biasanya *ERD* ini digunakan oleh profesional sistem untuk berkomunikasi dengan pemakai eksekutif tingkat tinggi dalam suatu organisasi (seperti wakil presiden direktur dan manajer yang tidak tertarik pada pelaksanaan operasi-operasi sistem sehari-hari (Al-Bahra, 2013:142).

Model *Entity-Relationship* (ER) pada awalnya disampaikan oleh Peter di tahun 1976 sebagai suatu cara untuk menyatukan jaringan dan menggambarkan *relational database*. Singkatnya, model ER adalah sebuah model konseptual dari data yang menggambarkan keadaan sebenarnya dari *entities* dan *relationship* (Wahyudi, 2008:199). Adapun notasi dari *Entity Relationship Diagram* (ERD) seperti yang terdapat pada **Gambar 2.2**.



**Gambar 2.2 Notasi ERD**  
(Wahyudi, 2008:201)

*Entity* adalah sebutan lain dari *record* dan *entities* atau *set entity* adalah sebutan lain dari file. Kata entitas yang digunakan di sini dimaksudkan untuk menggambarkan *entities*, meskipun seharusnya ditulis himpunan entitas. *Entity* adalah sebutan lain dari *record* dan *entities* atau *set entity* adalah sebutan lain dari file. Kata entitas yang digunakan di sini dimaksudkan untuk menggambarkan *entities*, meskipun seharusnya ditulis himpunan entitas.

Relasi (relationship) adalah keterhubungan atau keterkaitan antara satu dengan satu atau lebih entitas lain. Bentuk hubungan bermacam-macam, yaitu (Nugroho, 2010:119):

1. *One to One*

Satu rekaman di sebuah entitas berhubungan dengan hanya satu rekaman di entitas lain.

2. *One to Many*

Satu rekaman di sebuah entitas berhubungan dengan lebih dari satu rekaman di entitas lain.

3. *Many to Many*

Lebih dari satu rekaman di sebuah entitas berhubungan dengan lebih dari satu rekaman di entitas lain.

Atribut atau *field* adalah suatu karakteristik yang biasa untuk menggambarkan seluruh atau sebagian dari *record*. Kata lain dari atribut adalah elemen data. Atribut dan entitas memiliki keterkaitan yang dapat digambarkan dengan notasi penghubung.

Penghubung atau konektor merupakan bentuk dari keterkaitan antara entitas, atribut, maupun dengan relasi. Penghubung dapat digambarkan dengan

garis yang menghubungkan notasi berdasarkan keterikatan yang dimiliki. Notasi entitas, relasi, atribut dan penghubung.

## **2.5 Teori Pendukung Lainnya**

### **2.5.1 PHP (*HyperText Preprocessor*)**

Menurut Nugroho (2008:114), PHP adalah produk *Open Source* yang dapat digunakan secara gratis tanpa harus membayar untuk menggunakannya. PHP merupakan bahasa standar yang digunakan dalam dunia *website*, PHP adalah bahasa program yang berbentuk skrip yang diletakkan didalam server *web*. Jika kita lihat sejarah mulanya PHP diciptakan dari ide Rasmus Lerdof untuk kebutuhan pribadinya, akan tetapi kemudian dikembangkan lagi sehingga menjadi sebuah bahasa yang disebut "Personal Home Page".

Menurut Winarno dkk (2013:1), PHP adalah bahasa pemrograman web bersifat server side, yang tujuannya untuk menghasilkan skrip yang akan di-generate dalam kode HTML yang merupakan bahasa standar web.

### **2.5.2 MySQL**

Menurut Nugroho (2008:91), *MySQL (My Structured Query Language)* adalah sebuah program pembuat pengelola *database* atau yang sering disebut dengan DBMS (*Database management System*), sifat dari DBMS ini adalah Open Source. Sedangkan, menurut Nugroho (2014:31) MySQL adalah software atau program aplikasi database, yaitu software yang dapat kita pakai untuk menyimpan data berupa informasi teks dan juga angka.

### 2.5.3 XAMPP

Menurut Nugroho (2008:74), *XAMPP* merupakan paket *PHP* yang berbasis Open Source yang dikembangkan oleh sebuah komunitas Open Source. Sedangkan, menurut Nurgoho (2014:1) dengan menginstal paket *XAMPP* maka sudah termasuk didalamnya: *PHP*, *MySQL* dan *Apache* sebagai webservernya.

Sedangkan menurut Riyanto (2014:11), *XAMPP* merupakan paket *PHP* yang berbasis Open Source yang dapat digunakan sebagai *tool* pembantu pengaplikasi berbasis *PHP*.

## 2.6 Pengujian (*Testing*)

Pengujian perangkat lunak merupakan persentase terbesar dari upaya teknis dalam proses perangkat lunak. Apapun jenis perangkat lunak yang anda bangun, strategi untuk perencanaan pengujian yang sistematis, pelaksanaan, dan kontrol dimulai dengan mempertimbangkan elemen-elemen kecil dalam perangkat lunak dan bergerak keluar terhadap program secara keseluruhan. Tujuan pengujian perangkat lunak adalah untuk menemukan kesalahan (Pressman, 2012:580). Dalam penelitian ini peneliti menggunakan *Black Box* sebagai alat untuk pengujian perangkat lunak.

### 2.6.1 *Black Box Testing*

Menurut Pressman (2012:597), Pengujian *Black Box Testing* atau Kotak Hitam berfokus pada persyaratan fungsional perangkat lunak. Artinya, teknik pengujian kotak hitam memungkinkan Anda untuk membuat beberapa kumpulan kondisi masukan yang sepenuhnya akan melakukan semua kebutuhan fungsional untuk program. Pengujian kotak hitam bukan tehnik alternatif untuk

kotak putih. Sebaliknya, ini merupakan pendekatan pelengkap yang mungkin dilakukan untuk mengungkap kelas kesalahan yang berbeda dan yang diungkap oleh metode kotak putih.

Pengujian kotak hitam berupaya untuk menemukan kesalahan dalam kategori berikut :

- a. Fungsi yang salah atau hilang
- b. Kesalahan antarmuka
- c. Kesalahan dalam struktur data atau akses basis data eksternal
- d. Kesalahan perilaku atau kinerja
- e. Kesalahan inisialisasi dan penghentian.

Contoh pengujian *black box testing* :

**Tabel 2.5 Contoh Pengujian *Black Box Testing***

Kelas uji	Skenario uji	Hasil yang diharapkan	Kesimpulan
Menu keluar	Ketika memilih tombol keluar	Keluar dari aplikasi	[√] Berhasil [ ] Tidak Berhasil

*Black box Testing* dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Sehingga untuk pengujian pada sistem kepegawaian ini peneliti lebih memilih menggunakan *black box testing* untuk menguji semua fungsi yang terdapat dalam sistem.

## 2.7 Tinjauan Pustaka

Beberapa penelitian yang sebelumnya yang berkaitan dengan sistem informasi kepegawaian. Salah satunya penelitian yang sejenis di lakukan oleh Mohamad Reza Fachlevi (2017) dengan judul “Perancangan Sistem Informasi

Kepegawaian Berbasis Website Di Bagian Kepegawaian SDN Binakarya 1 Kabupaten Garut” dengan hasil penelitian yaitu sebuah sistem informasi kepegawaian berbasis web. Dalam penelitian ini peneliti menggunakan metode prototype untuk pengembangan sistem, serta menggunakan pendekatan berorientasi objek (OOP) dalam merancang sistem informasi kepegawaian ini.

Selain itu penelitian lain yang dilakukan oleh Hilman Abdurahman Achmad, Tacbir Hendro Pudjiantoro, Faiza Renaldi (2017) dengan judul “Sistem Informasi Manajemen Kepegawaian Di Dinas Komunikasi Dan Informatika Bandung Barat”, hasil dari penelitian ini adalah sebuah rancangan sistem informasi manajemen berbasis web. Perancangan sistem pada penelitian ini menggunakan alat bantu UML, *MySQL*, dan menggunakan bahasa pemrograman *PHP* dan *framework codeigniter* serta menggunakan metode *waterfall* dalam pengembangan sistem.

Penelitian lain dengan judul Pendekatan Metodologi Extreme Programming Pada Aplikasi E-Commerce Berbasis M-Commerce (Studi Kasus : Toko Buku An’Nur di Pontianak) (2017) yang dilakukan oleh Eri Bayu Pratama. Hasil dari penelitian ini sebuah aplikasi yang berbasiskan M-commerce yang mampu meningkatkan penjualan buku melalui media *online* dengan kemudahan dalam pemesanan buku menggunakan *smartphone*. Dalam penelitian ini, peneliti menggunakan metode *extreme programming* .

Penelitian lain yang di lakukan oleh Bobby Heriyanto (2017) dengan judul Sistem Informasi Kepegawaian Pada Minimarket Kertapati Jaya Indah Palembang. Dengan hasil penelitian yaitu sebuah sistem informasi kepegawaian pada minimarket kertapati jaya indah Palembang yang mampu membeikan kemudahan

pengolahan data bagi pegawai maupun pemilik perusahaan. Dalam penelitian ini, peneliti menggunakan metode *waterfall* dan menggunakan bahasa pemrograman *PHP* serta *MySQL* sebagai databasenya.

Penelitian lain dengan judul Pendekatan Metodologi Extreme Programming Pada Aplikasi E-Commerce Berbasis M-Commerce (Studi Kasus Sistem Informasi Penjualan Alat-Alat Telekomunikasi ) (2016) yang dilakukan oleh Raisya Rahmi, Rika Perdanan sari, Rahmat suhatman. Hasil dari penelitian ini sebuah aplikasi yang berbasiskan M-commerce yang mampu meningkatkan penjualan alat-alat telekomunikasi melalui media *online* dengan kemudahan dalam pemesanan ala-alat telekomunikasi. Dalam penelitian ini, peneliti menggunakan metode *extreme programming* .

Penelitian selanjutnya dilakukan oleh Yesri Elva (2016) dengan judul Perancangan Sistem Informasi Kepegawaian Berbasis Web Pada Badan Kepegawaian Daerah (BKD) Dengan Menggunakan Bahasa Scripting *PHP* dan Database *MySQL*. Hasil dari penelitian ini yaitu sebuah sistem informasi kepegawaian berbasis web. Dalam penelitian ini menggunakan bahasa pemrograman *PHP* dan *HTML* serta *MySQL* untuk Databasenya.

Penelitian lain dengan judul Pengembangan Fungsi Pengajuan Cuti Karyawan Pada Sistem Absensi *Mobile* (2016) yang dilakukan oleh Fransiskus Adikara dengan hasil penelitian yaitu sistem operasi berbasis android yang mampu mempermudah karyawan dalam proses pengajuan cuti melalui mobile. Dalam peneliti ini, peneliti menggunakan menggunakan metode *Extreme Programming* serta menggunakan pendekatan berorientasi objek dalam perancangan sistem ini.



Penelitian lain dengan judul Sistem Informasi Pengajuan Cuti Pegawai Dinas Penghubungan Komunikasi Dan Informatikan Propinsi Sumatera Barat yang dilakukan oleh Keukeu Rohendi (2015) dengan hasil penelitian yaitu sebuah sistem informasi pengajuan cuti yang sudah terintegrasi yang mampu mempermudah bagian staff kepegawaian dalam proses pengolahan data cuti pegawai. Dalam penelitian ini peneliti menggunakan bahasa pemrograman *PHP* dan *MySQL* sebagai databasenya.

Selanjutnya penelitian yang berjudul Sistem Informasi Kepegawaian Menggunakan *Framework Yii* yang dilakukan oleh Wellian Susanto, Leon Andretti Abdillah, Susan Dian Purnamasari (2015) yang hasil dari penelitiannya yaitu sebuah sistem informasi kepegawaian seperti : pengolahan data pegawai, data pensiun, data diklat, data cuti, data daftar urut kepangkatan sehingga sub-sub bagian dapat memperoleh informasi kepegawaian dengan cepat. Dalam penelitian ini peneliti menggunakan metode pengembangan sistem RAD serta menggunakan *Framework Yii* dan *MySQL* sebagai databasenya.

Penelitian selanjutnya yang berjudul Rancang Bangun Sistem Pengolahan Supplier Invoice Accounting Menggunakan Metode *Extreme Programming* yang dilakukan oleh Agung Triayudi, Vidila Rosalina, Fildzah (2014), hasil dari penelitian ini yaitu sistem pengolahan supplier invoice accounting, sistem ini menggunakan metode *extreme programming*, menggunakan alat bantu pengembangan sistem yaitu DFD, *Context Diagram*, *Entity Relationship Diagram* (ERD) serta microsoft SQL 2005 sebagai databasenya.

Dalam penelitian ini bertujuan untuk membangun sistem informasi kepegawaian pada PT. Rambang Agro Jaya berbasis web. Pendekatan metode

pengembangan sistem yang digunakan dalam penelitian ini merupakan salah satu dari *Agile Methods* yaitu *extreme programming*, model ini cenderung menggunakan pendekatan berorientasi objek. Bahasa pemrograman yang digunakan dalam penelitian ini adalah *PHP* dan *MySQL* sebagai databasenya. Hasil dari penelitian ini yaitu sebuah sistem yang menghasilkan informasi tentang pengajuan cuti, pengajuan pensiun, pengajuan pengunduran diri, dan perjalanan dinas.