

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Beberapa penelitian sebelumnya dilakukan penulis lain yang dijadikan sebagai bahan pertimbangan dan berkaitan dengan penelitian yaitu yang dilakukan oleh Enjelia Tani, Belinda Bagre dan Stenly Adam pada tahun 2018. Tujuan dari penelitian ini adalah agar dapat memudahkan pihak perusahaan dalam melakukan pekerjaan lebih efisien dan untuk pengunjung dapat memperoleh informasi mengenai perusahaan Sederhana Karya Jaya (SKJ). Bagi perusahaan SKJ adalah dengan adanya sistem informasi yang akan dibuat dapat membantu memudahkan bagi pihak perusahaan untuk menyimpan data, mengakses informasi perusahaan dan mengolah data perusahaan. Metode pengembangan software yang digunakan dalam penelitian ini adalah metode air terjun (*Waterfall*). Sistem ini dibangun menggunakan bahasa pemrograman PHP dengan *Framework CodeIgniter* dan *data base MySQL*. Hasil dari penelitian ini yaitu Sistem Informasi Kepegawaian PT Sederhana Karya Jaya Berbasis Web yang dapat digunakan pegawai untuk melihat gaji, absen dan jadwal kerja pegawai walaupun dari tempat tinggal pegawai tersebut jauh dari perusahaan [1].

Penelitian serupa yang dilakukan oleh Olha Musa pada tahun 2020. Tujuan dari penelitian ini adalah Merancang sistem informasi kepegawaian berbasis *website* untuk menghasilkan sistem yang maksimal dalam pengelolaan data administrasi pegawai secara efektif dan efisien. Studi Kasus Dinas Pariwisata Dan Kebudayaan Kabupaten Tojo Una-Una Provinsi Sulawesi Tengah. Metode pengembangan *software* yang digunakan dalam penelitian ini adalah metode *research and development* dengan metode penelitian dan pengembangan. Metode pengujian yang digunakan yaitu *white box* dan *black box*. Hasil dari penelitian ini adalah Sistem Informasi Kepegawaian Berbasis Web pada Dinas Pariwisata Dan Kebudayaan yang dapat memudahkan bagian kepegawaian untuk melakukan pendataan pegawai beserta informasi usulan kenaikan pangkat pegawai dan mempermudah pegawai untuk melakukan pengusulan kenaikan pangkat dan mendapatkan informasi tentang usulan kenaikan pangkat [2].

Penelitian lainnya yang dikukan oleh Novan Wijaya, Adelia Rizky Febriyanti dan Anton Wibowo pada tahun 2020. Tujuan dari penelitian ini adalah untuk mempermudah dalam mengatur data kepegawaian pada PT. Pelayaran Sakti Inti Makmur sehingga menjadi lebih efektif dan efisien. Data kepegawaian meliputi menghitung masa kerja pegawai secara otomatis, membantu dalam mengurus surat izin, cuti, pengunduran diri, surat peringatan (SP) menjadi lebih mudah, dan mempermudah HRD dalam proses pembuatan laporan. Metode pengembangan *software* yang digunakan dalam penelitian ini adalah metode *iterative*, dimana dalam metode *iterative* meliputi tahap analisis, desain, pembuatan kode program dan pengujian. Pengujian sistem yang dilakukan menggunakan metode *black box testing* dimana hanya menguji unit dan antar muka dari sistem. Hasil dari penelitian ini adalah Sistem Pengelolaan Data Kepegawaian Berbasis Web Pada PT. Pelayaran Sakti Inti Makmur [3].

Penelitian selanjutnya yang dilakukan oleh Nova Noor Kamala Sari, Putu Bagus Adidyana Anugrah Putra dan Widiatry pada tahun 2020. Tujuan dari penelitian ini adalah untuk memfasilitasi pembuatan SKP dan PKP oleh pegawai dan pimpinan, mengajukan usulan pangkat untuk jabatan struktural, penyusunan Data Urut Kepangkatan, dan membantu *staff* administrasi mengelola data kepegawaian. Metode pengembangan *software* yang digunakan dalam penelitian ini adalah metode *waterfall* menurut Ian Sommerville. Metode *testing* yang digunakan website ini adalah metode *Black box*. Hasil dari penelitian ini adalah Sistem Informasi Kepegawaian UPT Kesatuan Pengelolaan Hutan Produksi Kapuas Tengah UNIT XI [4].

Penelitian selanjutnya yang dilakukan oleh Widya Lestari, Lia Andriani dan Fransiska Prihatini Sihotang pada tahun 2018. Tujuan dari penelitian ini adalah untuk mengurangi kemungkinan terjadinya kesalahan dalam melakukan perekapan absensi, melengkapi, serta memudahkan proses pengajuan cuti, mutasi, pension, dan kenaikan pangkat. Metode pengembangan *software* yang digunakan dalam penelitian ini adalah metode iterasi yang terdiri dari tahapan yang masing-masing tahapannya dapat dilakukan secara berulang hingga didapatkan hasil yang diinginkan. Sistem ini dibuat dengan menggunakan *database* yaitu MySQL dan program aplikasi menggunakan Bahasa pemrograman yaitu PHP. Hasil dari penelitian ini adalah Sistem Informasi Kepegawaian Pada Kejaksaan Negeri Palembang yang dapat membantu staf kepegawaian dalam melakukan pengelolaan absensi pegawai, pengajuan cuti, mutase, pensiun, dan kenaikan pangkat [5].

Pada penelitian ini, penulis bermaksud merancang sistem informasi data induk kepegawaian berbasis web yang dapat mempermudah petugas kepegawaian dalam melakukan pengelolaan data kepegawaian dan perubahan data pegawai yang ada di Politeknik Negeri Cilacap. Perbedaan penelitian ini dengan penelitian sebelumnya yaitu terdapat proses verifikasi yang dilakukan pengelola dan proses *export* pdf yang dilakukan dalam perekapan data.

Tabel 2. 1 Literatur Jurnal Penelitian Terdahulu

No	Judul	Penulis	Tahun	Metode	Hasil	Perbedaan
1	Perancangan Sistem Informasi Kepegawaian PT Sederhana Karya Jaya Berbasis Web	Enjelia Tani, Belinda Bagre dan Stenly Adam	2018	Metode air terjun (<i>Waterfall</i>).	Hasil dari penelitian ini yaitu Sistem Informasi Kepegawaian PT Sederhana Karya Jaya Berbasis Web yang dapat digunakan pegawai untuk melihat gaji, absen dan jadwal kerja pegawai walaupun dari tempat tinggal	-Sistem ini terdapat fitur melihat gaji dan jadwal pegawai

					pegawai tersebut jauh dari perusahaan	
2	Sistem Informasi Kepegawaian Berbasis Website pada Dinas Pariwisata Dan Kebudayaan	Olha Musa	2020	Metode <i>Research and Development</i> dengan metode Penelitian dan Pengembangan.	Hasil dari penelitian ini adalah Sistem Informasi Kepegawaian berbasis Web pada Dinas Pariwisata Dan Kebudayaan yang dapat memudahkan bagian kepegawaian untuk melakukan pendataan pegawai beserta informasi usulan kenaikan	-Menggunakan metode <i>Research and Development</i> -Menggunakan metode pengujian <i>white box</i> dan <i>black box</i>

					pangkat pegawai.	
3	Aplikasi Pengelolaan Data Kepegawaian Berbasis Web Pada PT. Pelayaran Sakti Inti Makmur Palembang	Novan Wijaya, Adelia Rizky Febriyanti dan Anton Wibowo	2020	Metode <i>iterative</i> .	Hasil dari penelitian ini adalah Sistem Pengelolaan Data Kepegawaian Berbasis Web Pada PT. Pelayaran Sakti Inti Makmur.	-Menggunakan metode <i>iterative</i>
4	Sistem Informasi Kepegawaian UPT Kesatuan Pengelolaan Hutan Produksi Kapuas	Nova Noor Kamala Sari, Putu Bagus Adidyana Anugrah Putra dan Widiatry	2020	Metode <i>waterfall</i> menurut Ian Sommer ville.	Hasil dari penelitian ini adalah Sistem Informasi Kepegawaian UPT Kesatuan Pengelolaan Hutan Produksi Kapuas	-Sistem ini terdapat fitur mengajukan usulan pangkat untuk jabatan struktural

	Tengah UNIT XI				Tengah UNIT XI.	
5	Sistem Informasi Kepegawaian Pada Kejaksaaan Negeri Palembang	Widya Lestari, Lia Andriani dan Fransiska Prihatini Sihotang	2018	Metode iterasi.	Hasil dari penelitian ini adalah Sistem Informasi Kepegawaian Pada Kejaksaaan Negeri Palembang yang dapat membantu staf kepegawaian dalam melakukan pengelolaan absensi pegawai, pengajuan cuti, mutase, pensiun, dan kenaikan pangkat.	-Mengguna kan metode iterasi -Sistem ini terdapat fitur pengajuan cuti, mutasi, pension, dan kenaikan pangkat

2.2 Landasan Teori

Landasan teori berisi hal-hal yang berkaitan dengan permasalahan dan ruang lingkup dalam pembuatan laporan.

2.2.1 Sistem Informasi

Sistem informasi adalah suatu asosiasi terdiri dari beberapa modul yang saling terintegrasi dimana menyajikan kumpulan informasi dan pengolahan data untuk disajikan sesuai dengan kebutuhan pengguna, *database*, *source code*, dan desain model yang mewakili dari suatu rancangan sistem informasi berguna untuk mempermudah dalam pengembangan dan perawatan [6].

Sistem informasi merupakan salah satu solusi dari permasalahan permasalahan yang dihadapi organisasi dan juga berguna untuk menghadapi tantangan dimasa saat ini. Selain itu sistem informasi berperan penting dalam melakukan pengambilan keputusan, maka dari itu diperlukan informasi yang tepat dan akurat sesuai dengan kebutuhan organisasi [7].

2.2.2 Kepegawaian

Kepegawaian berasal dari kata pegawai yang artinya adalah orang yang melakukan pekerjaan dengan mendapat imbalan jasa berupa gaji dan tunjangan dari pemerintah ataupun badan usaha swasta. Dalam sebuah organisasi, pegawai merupakan alat yang menggerakkan dan menggiatkan agar segala kegiatan organisasi organisasi dapat berjalan sesuai dengan tujuannya. Tanpa adanya manusia sebagai pegawai maka tujuan suatu organisasi / perusahaan yang telah ditentukan tidak akan tercapai sebagaimana yang diharapkan [8].

2.2.3 Rekayasa Perangkat Lunak

Istilah Rekayasa Perangkat Lunak (RPL) secara umum disepakati sebagai terjemahan dari istilah *Software Engineering*. Istilah *Software Engineering* dipopulerkan tahun 1968 pada *Software Engineering Conference* yang diselenggarakan oleh NATO. Beberapa pengertian menurut para ahli dibidang perangkat lunak, diantaranya yaitu pertama menurut Stephen R. Schach “Rekayasa perangkat lunak adalah sebuah disiplin dimana dalam menghasilkan perangkat lunak bebas dari kesalahan dan dalam pengiriman anggaran tepat waktu serta memuaskan keinginan pemakai.”, kedua menurut Fritz Bauzer “Rekayasa perangkat lunak adalah penetapan dan penggunaan prinsip rekayasa dalam rangka

memperoleh perangkat lunak yang dapat dipercaya dan dapat bekerja secara efisien pada mesin nyata.”, ketiga menurut Janner Simarmata” Rekayasa perangkat lunak atau satu bidang profesi yang mendalami cara-cara pengembangan perangkat lunak termasuk pembuatan, pemeliharaan aplikasi perangkat lunak dengan menerapkan teknologi dan praktik dari ilmu komputer, manajemen organisasi pengembangan perangkat lunak, dan bidang-bidang lainnya.” [9].

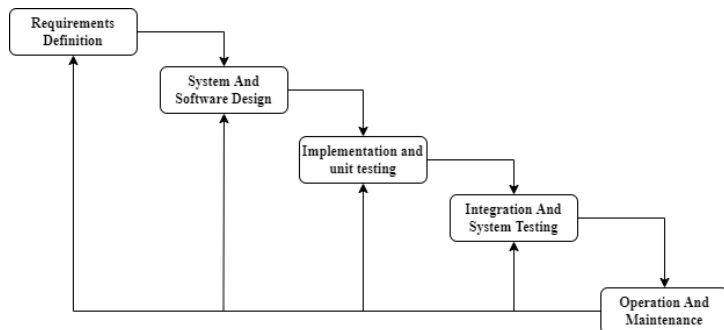
2.2.4 Rekayasa Web

Rekayasa web yaitu model rekayasa perangkat lunak yang digunakan untuk membuat web berkualitas tinggi [10]. Rekayasa web merupakan suatu model rekayasa perangkat lunak yang digunakan untuk perkembangan aplikasi-aplikasi berbasis web.

Berikut ini adalah metode dan tools yang digunakan :

1. Metode *Waterfall*

Metode *Waterfall* adalah salah satu model dalam pengembangan sistem Rekayasa Perangkat Lunak dimana klien dan pengembang dapat saling berkomunikasi dalam memenuhi kebutuhan sistem menurut Ian Sommerville. Metode ini banyak digunakan para pengembang sistem karena sangat membantu dalam mendefinisikan secara rinci kebutuhan sistem yang sesuai dengan keinginan klien [11].



Gambar 2. 1 Pemodelan *Waterfall*

Berikut rician penjelasan dari tahapan-tahapan metode *waterfall* :

a. *Requirements Definition*

Tahapan metode *waterfall* yang pertama yaitu tahap melakukan konsultasi dengan *user system* untuk menentukan pelayanan,

Batasan dan tujuan sistem. Persyaratan ini kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem.

- b. *System And Software Design*
Tahap yang selanjutnya adalah proses perancangan sistem membagi persyaratan dalam sistem perangkat keras atau perangkat lunak. Proses ini menentukan arsitektur sistem secara keseluruhan, mulai dari abstraksi sampai hubungan yang berelasi atau berkait.
- c. *Implementation and unit testing*
Tahapan metode *waterfall* yang berikutnya adalah proses dari menerjemahkan perancangan kedalam serangkaian program atau unit program. Pengujian dilakukan untuk meverifikasi setiap unit dapat bekerja dengan baik dan sesuai dengan spesifikasinya.
- d. *Integration And System Testing*
Tahap yang keempat yaitu tahap unit program individual diintegrasikan dan diuji sebagai sistem yang lengkap untuk menjamin bahwa persyaratan sistem telah terpenuhi.
- e. *Operation And Maintenance*
Tahapan metode *waterfall* yang terakhir, tahap ini adalah fase siklus hidup yang paling lama. Karena masuk kedalam proses penggunaan, sehingga perangkat lunak harus tetap dilakukan perawatan secara berkala, mencakup koreksi dari berbagai *error* yang tidak ditemukan pada tahap-tahap terdahulu.

Terkait dengan pengujian program, pengujian yang akan digunakan dalam pengembangan sistem ini adalah pengujian *black box*. Pengujian *black box* merupakan teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. Pengujian *black box* bekerja dengan mengabaikan struktur kontrol sehingga perhatiannya difokuskan pada informasi domain. Pengujian *black box* memungkinkan para pengembang perangkat lunak (*software*) untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program [12]. Pengujian *black box* berusaha untuk menemukan kesalahan dalam kategori sebagai berikut [12]:

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 terminal.

Berikut keuntungan dari menggunakan metode pengujian *black box*, yaitu [12]:

1. Penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu.
 2. Pengujian dilakukan dari sudut pandang pengguna, ini membantu untuk mengungkapkan ambiguitas atau inkonsistensi dalam spesifikasi persyaratan.
 3. Programmer dan tester keduanya saling bergantung satu sama lain.
- Metode pengujian *black box* ini juga memiliki kekurangan diantaranya yaitu [12] :

1. Uji kasus sulit diidentifikasi jika tanpa spesifikasi yang jelas.
2. Kemungkinan memiliki pengulangan tes yang sudah dilakukan oleh *programmer*.
3. Beberapa bagian *back end* tidak diuji sama sekali.

2. Tools / Alat Bantu


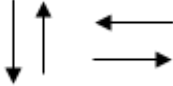




a. Flowchart





Flowchart merupakan urutan langkah-langkah pemecahan suatu masalah yang ditulis atau dilambangkan dengan simbol-simbol tertentu. *Flowchart* juga menggambarkan urutan logis dari suatu pemecahan masalah yang dituliskan dengan simbol-simbol yang sudah ditentukan. *Flowchart* akan mempresentasikan alur dalam suatu program secara logis [13]. Ada beberapa jenis flowchart diantaranya, yaitu :

1. Bagan alir sistem (*system flowchart*)
2. Bagan alir dokumen (*document flowchart*)
3. Bagan alir skematik (*schematic flowchart*)
4. Bagan alir program (*program flowchart*)
5. Bagan alir proses (*process flowchart*)

Flowchart mempunyai simbol-simbol tersendiri dari setiap anotasi-anotasi-geometri yang digunakan [13]. Simbol-simbol *flowchart* dapat dilihat pada Tabel 2.2 [13].

Tabel 2. 2 Simbol-Simbol Flowchart

No	Simbol	Fungsi
1	 Terminator Simbol	Simbol yang digunakan untuk permulaan (<i>start</i>) atau akhir (<i>end</i>) dari suatu kegiatan.
2	 <i>Flow Direction Symbol</i>	Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain (<i>connecting line</i>). Berfungsi juga untuk menunjukkan garis alir dari proses.
3	 <i>Processing Symbol</i>	Simbol yang digunakan untuk menunjukkan kegiatan yang dilakukan oleh komputer.
4	 <i>Manual Operation Symbol</i>	Simbol yang digunakan untuk menunjukkan kegiatan/proses yang tidak dilakukan oleh komputer.
5	 <i>Decision Symbol</i>	Simbol yang digunakan untuk memilih proses atau keputusan berdasarkan kondisi yang ada.
6	 <i>Input-output Symbol</i>	Simbol yang digunakan untuk menunjukkan proses <i>input-output</i> yang terjadi tanpa bergantung dari jenis peralatannya.

7	 <i>Document Symbol</i>	Simbol yang menunjukkan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> yang perlu cetak diatas kertas.
8	 <i>Predefined Process Symbol</i>	Simbol yang digunakan untuk menunjukkan hasil pelaksanaan suatu bagian prosedur.
9	 <i>Preparation Symbol</i>	Simbol yang digunakan untuk mempersiapkan penyimpanan didalam <i>storage</i> .
10	 <i>Manual Input Symbol</i>	Simbol yang digunakan untuk menunjukkan <i>input</i> data secara manual menggunakan <i>online keyboard</i> .




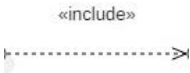

b. Unified Modeling Language (UML)

Unified Modelling Language (UML) adalah pemodelan berorientasi objek dalam melakukan perancangan sebuah sistem, akan tetapi dapat juga digunakan untuk permodelan aplikasi *procedural*. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak yang dimana aplikasi tersebut bisa berjalan pada komputer, sistem operasi dan jaringan apapun serta dapat ditulis dalam bahasa pemrograman apapun, karena UML juga menggunakan *class* dan *operation* didalam konsep dasarnya, maka dari itu lebih cocok untuk penulisan piranti lunak dibahasa pemrograman berorientasi objek. Meskipun demikian, UML tetap bisa digunakan untuk modeling aplikasi *procedural* dalam VB atau C++[14]. Beberapa diagram dari *Unified Modeling Language* (UML) antara lain :

1. Use Case Diagram

Use case diagram adalah diagram yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, diagram ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas). *Use case* diagram memiliki peran penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna, diagram ini juga bersifat statis [14]. Simbol-simbol *use case* dapat dilihat pada Tabel 2.3 [14].

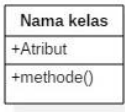




Tabel 2. 3 Simbol-Simbol Use Case


No	Simbol	Keterangan
1	 <i>Actor</i>	Menggambarkan tokoh atau seseorang yang berinteraksi dengan sistem. Dan dapat menerima dan memberi informasi pada sistem.
2	 <i>Use case</i>	Menjelaskan fungsi dari kegunaan sistem yang dirancang.
3	 <i>Association</i>	Menghubungkan antara <i>use case</i> dengan aktor tertentu.
4	 <i>Include</i>	Menunjukkan bahwa <i>use case</i> satu merupakan bagian dari <i>use case</i> lainnya.
5	 <i>Extend</i>	Menunjukkan arah panah secara putus-putus dari <i>use case</i> ke <i>base use case</i> .

2. Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem. Selain itu, *class diagram* menggambarkan struktur dan deskripsi kelas, *package* beserta hubungan antara satu dengan yang lain [14]. Simbol-simbol *class* dapat dilihat pada Tabel 2.4 [14].

Tabel 2. 4 Simbol-Simbol Class Diagram





No	Simbol	Keterangan
1	 <p style="text-align: center;"><i>Class</i></p>	Menggambarkan sebuah kelas pada sistem yang terbagi menjadi 3 bagian. Bagian atas adalah nama kelas. Bagian tengah adalah atribut kelas. Bagian bawah adalah <i>methode</i> dari kelas.
2	 <p style="text-align: center;"><i>Association</i></p>	Hubungan statis antar kelas. menggambarkan kelas yang memiliki atribut berupa kelas lain atau kelas yang harus mengetahui eksistensi kelas lain.
3	 <p style="text-align: center;"><i>Agregation</i></p>	Hubungan yang menyatakan bahwa suatu kelas menjadi atribut bagi kelas lain.
4	 <p style="text-align: center;"><i>Composition</i></p>	Bentuk khusus dari <i>agregation</i> dimana kelas yang menjadi bagian diciptakan setelah kelas <i>whole</i> dibuat.
5	 <p style="text-align: center;"><i>Generalization</i></p>	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).



6	 <i>Directed Assoaction</i>	Asosiasi dengan makna kelas yang satu digunakan oleh kelas yang lain.
---	---	---

3. Sequence Diagram

Sequence diagram adalah diagram yang menggambarkan interaksi antara objek di dalam dan sekitar sistem (termasuk *user*, *display*, dsb) berupa pesan yang digambarkan terhadap waktu, menggambarkan *scenario* / rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan keluaran (*output*) tertentu [14]. Simbol-simbol *sequence diagram* dapat dilihat pada Tabel 2.5 [14].

Tabel 2. 5 Simbol-Simbol Sequence Diagram

No	Simbol	Keterangan
1	 <i>Actor</i>	Orang yang berinteraksi dengan sistem.
2	 <i>Boundary</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.
3	 <i>Control</i>	Menggambarkan penghubung antara <i>boundary</i> dengan tabel.
4	 <i>Entity</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.

5	 <i>Message</i>	Mengindikasikan komunikasi antar objek.
6	 <i>Life Line</i>	Mengindikasikan keberadaan sebuah objek dalam basis waktu.

2.2.5 Pemrograman Berorientasi Objek

Pemrograman berorientasi objek (*Object Oriented Programming* atau disingkat OOP) merupakan suatu pemrograman yang menggunakan objek dan *class*. Pemrograman berorientasi objek ini memberikan kemudahan dalam pembuatan sebuah program [15]. Keuntungan yang bisa dapat dengan menggunakan pemrograman berorientasi objek ini [15], antara lain :

1. *Reusability*, kode yang dibuat dapat digunakan kembali.
2. *Extensibility*, programmer dapat membuat metode baru atau mengubah yang sudah ada sesuai yang diinginkan tanpa harus membuat kode dari awal
3. *Maintainability*, kode yang sudah dibuat lebih mudah untuk dikelola apabila aplikasi yang dibuat berskala besar yang memungkinkan adanya *error* dalam pengembangannya hal tersebut dapat diatasi dengan OOP karena pemrograman OOP sudah menggunakan konsep modularitas.

Objek dalam pemrograman berorientasi objek merupakan unit terkecil pemrograman yang memiliki data(sifat karakteristik) dan fungsi. Objek merupakan entitas dari sebuah keadaan, perilaku, dan identitas yang tugasnya sudah dirumuskan dalam suatu lingkup masalah, pendeklarasian objek dari sebuah *class* yang disebut dengan *instance*. *Class* merupakan wadah yang berisi pemodelan sebuah objek, mendiskripsikan karakteristik dan fungsi dari objek tersebut.

2.2.6 Basis Data

Basis data atau database merupakan sekumpulan informasi yang diatur agar lebih mudah dicari. Dalam arti umum basis data adalah sekumpulan data yang diproses dengan bantuan komputer yang memungkinkan data dapat diakses dengan mudah dan tepat, yang dapat digambarkan sebagai aktivitas dari satu atau lebih organisasi yang berelasi [16]. Jenis basis data yang sering digunakan :

1. Xampp

XAMPP merupakan sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolah data MySQL dikomputer lokal. XAMPP berperan sebagai *server web* pada komputer anda. XAMPP juga dapat disebut sebuah CPanel server *virtual*, yang dapat membantu anda melakukan *preview* sehingga dapat memodifikasi *website* tanpa harus *online* atau terakses dengan internet [16].

2. MySQL

MySQL adalah salah satu *Databases Management System* (DBMS) dari sekian banyak DBMS seperti Oracle, MS SQL, Postagre SQL, dan lainnya”. MySQL berfungsi untuk mengolah *database* menggunakan bahasa SQL. MySQL bersifat *open source* sehingga kita bisa menggunakannya secara gratis. Pemograman PHP juga sangat mendukung/*support* dengan *database* MySQL [16].

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah suatu bahasa (*language*) yang digunakan untuk mengakses data di dalam sebuah database relasional. SQL sering juga disebut dengan istilah *query*, dan bahasa SQL secara praktiknya digunakan sebagai bahasa standar untuk manajemen database relasional. Hingga saat ini hampir seluruh *server database* atau *software database* mengenal dan mengerti bahasa SQL. Dalam penggunaan SQL terdapat beberapa perintah yang berguna untuk mengakses dan memanajemen data yang terdapat dalam *database*. Secara garis besar, SQL *Server* mempunyai 3 (Tiga) jenis perintah SQL yaitu :

a. Data Definition Language (DDL)

DDL merupakan sub perintah dari bahasa SQL yang digunakan untuk membangun kerangka sebuah database, dalam hal ini database dan table. Terdapat tiga perintah penting dalam DDL, yaitu :

- 1.) *CREATE*: perintah yang digunakan untuk membuat, termasuk di dalamnya membuat *database* baru, tabel baru *view* baru, dan kolom baru.
 - 2.) *ALTER*: perintah yang berfungsi untuk mengubah struktur tabel yang telah dibuat. Mencakup di dalamnya mengubah nama tabel, menambah kolom, mengubah kolom, menghapus kolom, dan memberikan atribut pada kolom.
 - 3.) *DROP*: perintah yang berfungsi untuk menghapus *database* atau tabel.
- b. Data Manipulation Language (DML)
- DML merupakan sub perintah dari bahasa SQL yang digunakan untuk memanipulasi data dalam *database* yang telah dibuat. Terdapat 4 (Empat) perintah penting dalam DML, yaitu :
- 1.) *INSERT*: perintah yang digunakan untuk memasukkan data baru ke dalam sebuah tabel. Perintah ini tentu saja bisa dijalankan ketika *database* dan tabel sudah dibuat.
 - 2.) *SELECT*: perintah yang digunakan untuk mengambil dan menampilkan data dari tabel atau bahkan dari beberapa tabel dengan penggunaan relasi.
 - 3.) *UPDATE*: perintah yang digunakan untuk memperbaharui data pada sebuah tabel.
 - 4.) *DELETE*: perintah yang digunakan untuk menghapus data dari sebuah tabel.
- c. Data Control Language (DCL)
- DCL merupakan sub bahasa SQL yang berfungsi untuk melakukan pengontrolan data dan server databasenya, seperti manipulasi *user* dan hak akses (*priviledges*). Yang termasuk perintah dalam DCL ada 2 (Dua), yaitu :
- 1.) *GRANT*: perintah yang digunakan untuk memberikan hak akses oleh admin ke salah satu *user* atau pengguna. Hak akses tersebut bisa berupa hak membuat (*CREATE*), mengambil data (*SELECT*), menghapus data (*DELETE*), mengubah data (*UPDATE*), dan hak khusus lainnya yang berhubungan dengan sistem *database*.

- 2.) *REVOKE*: perintah yang digunakan untuk mencabut hak akses yang telah diberikan kepada *user*. Kebalikan dari perintah *GRANT*.

2.2.7 Usability

Usability adalah jenis kategori metode dalam evaluasi *usability* dengan mengobservasi pengguna pada sebuah desain kemudian diambil datanya dan dianalisis. Selama tes, *user* diminta menyelesaikan tugas, sementara pengamat melihat, mendengar dan mencatat. Tujuannya untuk mengidentifikasi masalah *usability*, mengumpulkan data kualitatif dan kuantitatif dan menentukan kepuasan pengguna dengan produk [17].

~Halaman ini sengaja dikosongkan~