



## **BAB II**

# **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Beberapa penelitian tentang Permohonan Cuti Pegawai dengan hasil yang relevan antara lain: Muhammad Al Khusnul Rizki, A Ferico [7], melakukan sebuah penelitian pada Pengadilan Tata Usaha Negara Bandar Lampung, dimana Pengadilan Tata Usaha Bandar Lampung memiliki sebuah masalah pada proses permohonan cuti yang masih bersifat konvensional. Sistem konvensional yang dimaksud adalah dengan menggunakan form kertas pengajuan cuti dalam mengajukan permohonan cuti. Sehingga dibutuhkan suatu sistem yang dapat membantu proses permohonan cuti. Sistem ini berbasis website menggunakan PHP 7 dan menggunakan *database* MySQL. Manfaat yang didapat dari sistem informasi ini yaitu mampu menghasilkan informasi yang cepat, dan tepat waktu sehingga memudahkan dalam proses permohonan cuti.

Penelitian lain yang dilakukan oleh Oktaviani V [8], pada PT. Aneka Gas Industri TBK yang memiliki permasalahan berupa proses perhitungan serta pencatatan sisa cuti yang dicatat menggunakan aplikasi Microsoft Excel yang mana bisa terjadi *human error* karena pencatatan tersebut dilakukan oleh manusia. Masalah lain yang dihadapi yakni apabila karyawan yang mengajukan permohonan cuti lebih dari satu orang, maka akan mengakibatkan lambatnya proses pengajuan cuti karyawan. Sistem ini dikembangkan menggunakan metode *System Development Life Cycle* (SDLC). Manfaat adanya Sistem Informasi Cuti Karyawan dapat membantu personalia dalam mengelola data cuti karyawan, membuat laporan rutin dan memperoleh informasi dengan mudah dan cepat, dan mengurangi terjadinya kesalahan-kesalahan oleh *human error*.

Penelitian lain telah dilakukan oleh Iin Seprina, Evi Yulianingsih [9], pada SMA Az-Zahra yang memiliki permasalahan yaitu proses pengajuan cuti masih menerapkan sistem konvensional

di mana pegawai diwajibkan mengisi formulir pengajuan surat cuti kebagian SDM sehingga harus menunggu terlebih dahulu surat balasan dari pihak SDM, proses tersebut kurang efisien. Sistem ini dikembangkan menggunakan metode RUP (*Rational Unified Process*) dan menggunakan pemodelan visual *Unified Modeling Language* (UML) dalam membuat perancangan desain. Manfaat yang didapat dari sistem pengajuan cuti yaitu karyawan dapat dilakukan secara online dengan berbasis Web Mobile sehingga dapat lebih mudah bagi karyawan untuk mengajukan permohonan cuti tanpa perlu datang langsung ke bagian HRD dan pegawai dapat mengetahui berapa jumlah sisa cuti yang telah diambil tanpa perlu menanyakan langsung ke HRD.

Penelitian lain juga telah dilakukan oleh Zahrani Khairunnisa, Fitri Susanti, Robbi Hendriyanto [10], pada Kantor Sekretariat Kota Administrasi Jakarta Utara yang memiliki permasalahan pada proses persetujuan cuti yang mana Walikota atau Sekretaris Kota yang berhak memberi persetujuan tidak selalu berada di tempat atau lingkungan kantor, hal tersebut membuat tertundanya proses pengajuan cuti, serta masalah lain terdapat pada saat pembuatan laporan tahunan yang mana berkas-berkas cuti bisa hilang atau tercecer dikarenakan berkas tersebut disimpan dalam flashdisk, komputer atau *hardisk* yang mana data berkas tersebut bisa saja rusak. Sistem ini dikembangkan menggunakan metode SDLC (*System Development Life Cycle*) model *Waterfall*, dengan basis website menggunakan *framework* CodeIgniter. Manfaat yang didapat dari sistem ini adalah mempercepat proses persetujuan yang diberikan oleh Walikota atau Sekretaris Kota dimanapun dan kapanpun, serta lebih terjaganya berkas – berkas laporan tahunan karena data tersebut sudah tersimpan didalam *database*.

Penelitian berikutnya dilakukan oleh Rohmat Taufiq, Angga Aditya Permana, Mifta Alliandry Marfino [11], pada PT Tribuana Gasindo yang memiliki permasalahan pada pengecekan histori formulir cuti pada masing-masing pegawai yang mana masih konvensional. Masalah lain terdapat pada dokumen permohonan cuti yang jarang adanya *back-up* data dokumen. Sistem ini dikembangkan dengan menggunakan bahasa pemrograman PHP dan berbasis website serta menggunakan pengujian *black box* yang mana bertujuan untuk menguji apakah setiap fungsi di dalam program dapat berjalan dengan benar. Manfaat yang didapat dari sistem ini adalah pengajuan

cuti karyawan yang dapat lebih efektif dalam pengelolaan data cuti karyawan dan dapat mengatasi masalah *human error*. Dari penelitian yang akan dilakukan memiliki suatu perbedaan dengan penelitian-penelitian sebelumnya. Perbedaan tersebut diantaranya yaitu penggunaan Framework Laravel dalam pembangunan sistem ini, yang mana menggunakan bahasa pemrograman PHP (*Hypertext Preprocessor*) dan MySQL sebagai aplikasi *database*. Tahap pengembangan sistem ini menggunakan metode SDLC (*System Development Life Cycle*) model *Prototype*. Serta terdapat perbedaan lain yang terletak pada fitur *approval* pengajuan cuti yaitu adalah *two step approval*.

**Tabel 2.1** Perbandingan Penelitian Sistem

No	Judul	Peneliti	Perbedaan	Keterbaruan
1.	Rancang Bangun Aplikasi E-Cuti Pegawai Berbasis Website (Studi Kasus: Pengadilan Tata Usaha Negara) (2021)	Muhammad Al Khusnul Rizki, A Ferico OP	Framework Code Igniter  Metode Extreme Programming (XP)	Framework Laravel  Metode Prototype
2.	Perancangan Sistem Informasi Cuti Karyawan Pada PT. Aneka Gas Industri TBK.	Muhammad, Vinnuri Oktaviani	Sistem berbasis desktop  Tidak menggunakan <i>emailgateway</i>	Sistem berbasis Web  Menggunakan <i>emailgateway</i>

	Pekanbaru (2019)			
3.	Penerapan Metode RUP Untuk Sistem Pengajuan Cuti Pegawai di SMA Az-Zahra Palembang Berbasis Web Mobile (2022)	Iin Seprina, Evi Yulianingsih	Menggunakan Metode RUP Menggunakan bahasa pemrograman PHP tanpa Framework  Tidak menggunakan <i>emailgateway</i>	Menggunakan Metode <i>Prototype</i> Menggunakan Framework Laravel  Menggunakan <i>emailgateway</i>
4.	Aplikasi Pengelolaan Cuti Pegawai Berbasis Web Studi Kasus: Kantor Sekretariat Kota Administrasi Jakarta Utara (2019)	Zahrani Khairunnisa, Fitri Susanti, Robbi Hendriyanto	Menggunakan Metode <i>Waterfall</i>  Menggunakan Framework CodeIgniter  Tidak menggunakan <i>emailgateway</i>	Menggunakan Metode <i>Prototype</i>  Menggunakan Framework Laravel  Menggunakan <i>emailgateway</i>
5.	Rancang Bangun Sistem Informasi Pengajuan Cuti Berbasis Web Pada PT.	Rohmat Taufiq, Angga Aditya Permana, Mifta Alliandry Marfino	Menggunakan bahasa pemrograman PHP tanpa Framework  Tidak menggunakan	Menggunakan Framework Laravel  Menggunakan

	Tribuana Gasindo (2022)		<i>emailgatewa y</i>  Menggunaka n Metode <i>SDLC</i>	<i>emailgatewa y</i>  Menggunak an Metode <i>SDLC</i> model Prototype
--	-------------------------	--	---	---

## 2.2 Landasan Teori

### 2.2.1 Sistem Informasi

Sistem informasi adalah suatu sistem yang ada didalam suatu perusahaan atau organisasi yang berjalan secara terorganisir untuk mengumpulkan, memasukan, dan memroses data dan penyimpanannya, mengelola, mengontrol dan melaporkannya untuk digunakan sebagai sarana mencapai tujuan. Pada saat ini organisasi banyak dipengaruhi oleh perkembangan sistem informasi yang mana ingin mencapai tahap sistem informasi yang cepat, relevan, serta akurat. Sistem informasi juga merupakan suatu sistem didalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, serta bersifat manajerial dan merupakan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Jadi dapat disimpulkan bahwa sistem informasi adalah serangkaian langkah-langkah dalam menggabungkan subsistem untuk mempertemukan kebutuhan organisasi dengan laporan yang dibutuhkan [12].

Adapun definisi sistem informasi menurut beberapa ahli adalah sebagai berikut:

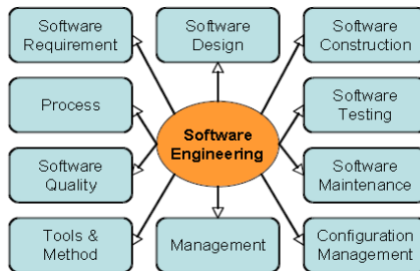
1. Menurut Laudon bahwa sistem informasi adalah suatu sistem yang mendukung pengambilan keputusan dan pengawasan pada sebuah organisasi yang berisi serangkaian komponen yang saling terhubung, seperti mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi [13].

2. Menurut Jogiyanto, sistem informasi adalah hasil olah data yang berguna serta bermakna untuk digunakan sebagai penjelasan untuk mengambil suatu keputusan [14].
3. Menurut Alter, sistem informasi adalah suatu kombinasi dari informasi, prosedur kerja, manusia dan teknologi informasi yang terorganisasi untuk mencapai tujuan organisasi [15].

### 2.2.2 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak (*software*) adalah suatu perintah program dalam komputer yang berisi proses yang digunakan untuk menghasilkan perangkat lunak yang dapat bekerja lebih efisien dan efektif sesuai dengan perintah dan kemauan user [16].

Ruang lingkup dari RPL dapat digambarkan sebagai berikut [17]:



**Gambar 2.1** Ruang Lingkup RPL

1. *Software Requirment* merupakan kegiatan yang dilakukan untuk mengidentifikasi dan menganalisis kebutuhan perangkat lunak. Hasil akhir tahapan ini adalah spesifikasi dan model perangkat lunak.
2. *Software Design* adalah tahapan perancangan arsitektur, komponen, antar muka, dan karakteristik lain dari perangkat lunak.
3. *Software Construction* berhubungan dengan detail pengembangan perangkat lunak, termasuk algoritma, pengkodean, pengujian dan pencarian kesalahan.

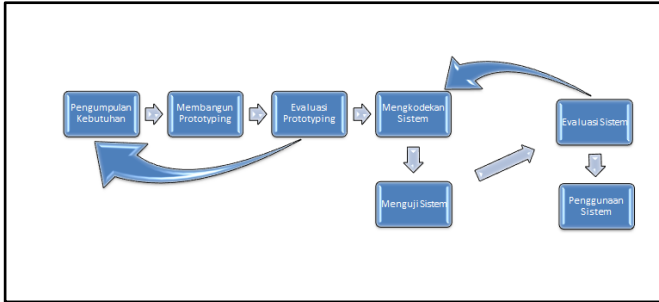
4. *Software Testing* meliputi pengujian pada keseluruhan perilaku perangkat lunak.
5. *Software Maintenance* mencakup upaya-upaya perawatan ketika perangkat lunak telah dioperasikan.
6. *Software Configuration Management* berhubungan dengan usaha perubahan konfigurasi perangkat lunak untuk memenuhi kebutuhan tertentu.
7. *Software Engineering Management* berkaitan dengan pengelolaan dan pengukuran RPL, termasuk perencanaan proyek perangkat lunak.
8. *Software Engineering Tools and Methods* mencakup kajian teoritis tentang alat bantu dan metode RPL.
9. *Software Engineering Process* berhubungan dengan definisi, implementasi pengukuran, pengelolaan, perubahan dan perbaikan proses RPL.
10. *Software Quality* menitik beratkan pada kualitas dan daur hidup perangkat lunak.

#### **A. Metode Pengembangan Sistem**

Tahap pengembangan sistem dalam penelitian ini menggunakan SDLC (*System Development Life Cycle*) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan untuk mengembangkan sistem perangkat lunak sebelumnya berdasarkan *best practice* atau cara-cara yang sudah disebut baik. SDLC yang digunakan dalam pengembangan sistem menggunakan SDLC model *prototyping* menurut Pressman [18].

Metode *prototyping* adalah sebuah model pengembangan sistem dengan model perancangan kilat lalu dievaluasi. Pengembangan sistem dikerjakan secara cepat dan juga melibatkan interaksi dengan *stakeholder*, berikut adalah tahap-tahap dalam model *prototyping* [19].





**Gambar 2.2** Model *Prototyping*

1. **Pengumpulan Kebutuhan**  
Pada tahap ini pelanggan dan pengembang bersama-sama mendefinisikan format dan kebutuhan perangkat lunak secara keseluruhan, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.
2. **Membangun *Prototyping***  
Yang dilakukan pada tahap ini adalah membuat desain sementara yang berpusat pada penyajian kepada pelanggan, misalnya; dengan membuat contoh input dan output.
3. **Evaluasi *Prototyping***  
Jika *prototype* yang sudah dibangun sesuai dengan keinginan pelanggan maka dilakukan evaluasi. Jika sudah sesuai akan dilanjutkan ketahap pengkodean (*coding*), namun jika belum sesuai maka akan diperbaiki dengan mengulang kembali tahap 1, tahap 2, dan tahap 3.
4. **Mengkodekan Sistem**  
Tahap ini lakukan jika *prototype* yang dibuat sudah diterima dan disepakati yang dilanjutkan dengan pengkodean dengan bahasa pemrograman yang sesuai
5. **Menguji Sistem**  
Sistem yang telah selesai dibuat kemudian akan dilakukan pengujian dengan menggunakan metode *black-box*. Tujuannya adalah untuk mengetahui apakah sistem yang dibuat sudah dapat berfungsi dengan baik dan benar sesuai yang diharapkan.
6. **Evaluasi Sistem**

Tahap ini dilakukan untuk mengetahui apakah sistem yang dibangun telah sesuai dengan yang diharapkan. Jika sudah memenuhi dapat dilakukan tahap selanjutnya, namun jika tidak maka perlu dilakukan pengulangan pada tahap 4 dan tahap 5.

#### 7. Menggunakan Sistem

Sistem yang dibangun siap digunakan oleh pelanggan setelah tahap pengujian berhasil dan diterima oleh pelanggan.

### **B. Metode Pengujian Sistem**

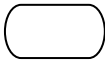
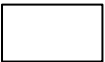
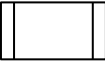

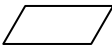
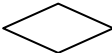
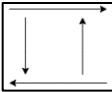
Tahap pengujian sistem ini menggunakan metode *Black box testing*. *Black box testing* merupakan pengujian perangkat lunak yang berfokus pada detail kebutuhan perangkat lunak seperti tampilan perangkat lunak dan fungsionalitas perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak sudah berfungsi dengan benar. Pengujian *Black box testing* bertujuan untuk menemukan kesalahan dalam kategori kesalahan antarmuka, kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi dan terminasi [20].

### **C. Alat Bantu atau Tools**






#### **1. Flowchart**

*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan dari suatu program. *Flowchart* adalah bentuk gambar/diagram yang mempunyai aliran satu atau dua arah secara sekuensial. Berikut Tabel 2.2 dan 2.3 merupakan simbol-simbol yang terdapat pada *flowchart* :

**Tabel 2.2** Simbol-Simbol *Flowchart* [21]

No.	Simbol	Nama	Keterangan
1.		<i>Terminal</i>	Memulai dan mengakhiri suatu program
2.		<i>Proses</i>	Proses perhitungan atau proses pengolahan data
3.		<i>Predefined Process</i> (sub program)	Permulaan sub program atau proses pengolahan data
4.		<i>Preparation</i>	Proses inisialisasi atau pemberian harga awal
5.		<i>Input – Output</i>	Memasukan data maupun menunjukkan hasil dari suatu <i>process</i> tanpa tergantung dengan jenis peralatannya
6.		<i>Decision</i>	Memilih proses berdasarkan kondisi yang ada.
7.		<i>Flow</i>	Menghubungkan antara simbol satu dengan simbol yang lain atau menyatakan jalannya arus dalam suatu proses. Simbol arus ini sering disebut juga dengan <i>connecting line</i> .

**Tabel 2.3** Simbol-Simbol *Flowchart* [21]

No	Simbol	Nama	Keterangan
1.		<i>Document</i>	Merupakan simbol untuk data yang terbentuk informasi.
2.		<i>On Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman
3.		<i>Off Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda
4.		<i>Manual Operation</i>	Menunjukkan pengolahan yang tidak dilakukan oleh komputer/pc.
5.		<i>Manual Input</i>	Memasukan data secara manual <i>on-line keyboard</i> .

## 2. *UML (Unified Modeling Language)*


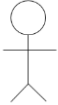




*UML (Unified Modeling Language)* merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggambarkan diagram dan teks-teks pendukung [22]. *UML* memiliki banyak diagram, berikut penjelasan mengenai diagram yang ada di *UML*.

### a. *Use Case Diagram*

*Use Case Diagram* merupakan pemodelan yang memiliki kemampuan dalam menggambarkan interaksi diantara aktor dan sistem. *Use case* digunakan untuk mengetahui fungsi

apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu [23]. Berikut tabel 2.4 adalah simbol-simbol yang ada pada diagram *use case diagram*:

**Tabel 2.4** Simbol-Simbol Use Case Diagram [24]


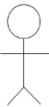




No.	Simbol	Nama	Keterangan
1.		<i>Use case</i>	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
2.		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
3.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5.		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas

b. *Sequence Diagram*

*Sequence Diagram* digunakan untuk menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima

oleh objek. Banyaknya diagram sekuen yang akan dibangun sesuai dengan pendefinisian usecase yang memiliki proses sendiri [25]. Berikut adalah tabel 2.5 adalah simbol-simbol yang ada pada *Sequence Diagram* :

**Tabel 2.5** Simbol-Simbol Sequence Diagram [24]

No.	Simbol	Nama	Keterangan
1.		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Actor</i>	Menggambarkan <i>user</i> atau pengguna.
3.		<i>Message</i>	Spesifikasi dari komunikasi antar <i>objek</i> yang memuat informasi – informasi tentang aktivitas yang terjadi.
4.		<i>Boundary</i>	Menggambarkan sebuah <i>form</i> .
5.		<i>Control Class</i>	Menghubungkan <i>boundary</i> dengan Tabel.
6.		<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.

### 2.2.3 Pemrograman Berorientasi Objek

Metodelogi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan informasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metodologi berorientasi objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas. Metodologi berorientasi objek merupakan rangkaian aktivitas analisis orientasi objek, pemrograman berorientasi objek, dan pengujian berorientasi objek [26].

### 2.2.4 Web

*Web* adalah kumpulan halaman yang saling terhubung dan memiliki keterkaitan dengan berkas-berkasnya yang tersimpan pada *web server*. *Web* juga dapat berisikan dokumen terkait multimedia seperti gambar, suara, animasi dan video yang didalamnya menggunakan *protocol* HTTP (*Hypertext Transfer Protocol*) yang dapat diakses menggunakan *browser* [27]. *Web* memiliki pola desain arsitektur yaitu *MVC* serta memiliki beberapa *framework* untuk pengembangan suatu web.

#### a. *Web Server*

*Web server* adalah perangkat lunak server yang berfungsi untuk menerima permintaan dari klien dalam bentuk situs web melalui HTTP atau HTTPS yaitu melalui jaringan *local* atau melalui jaringan *internet*. Ada beberapa *web server* yang tersedia diantaranya yaitu *Apache* dan *IIS* (*Internet Information Service*). *Web server* yang digunakan penulis adalah *Apache* [28].

#### b. *Web Browser*

Pengertian *web browser* adalah perangkat lunak untuk mengidentifikasi sumber informasi, yang berupa halaman website, gambar, video atau jenis konten lainnya. Fungsi *Web Browser* adalah mempermudah dalam pencarian informasi di internet, menyimpan berbagai data di internet, membuka file menggunakan ekstensi khusus, menjaga keamanan website, serta

mendukung pemakaian search engine. Jenis-jenis *browser* yang biasa digunakan adalah Google, Chrome, Mozilla, Firefox, dan masih banyak yang lainnya [29].

### 2.2.5 PHP

PHP merupakan singkatan dari PHP *Hypertext Preprocessor* yang digunakan sebagai bahasa *script server-side* dalam pengembangan web yang disisipkan pada dokumen HTML. PHP merupakan software *opensource* yang disebar dan dilisensikan secara gratis. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya serta *web server* yang mendukung PHP dapat ditemukan dimana mana salah satunya apache. PHP adalah bahasa *opensource* yang dapat digunakan di beberapa mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah *system* [30].

### 2.2.6 Framework

*Framework* adalah kumpulan pernyataan atau instruksi yang dikumpulkan dalam *class* dan *function* dengan fungsi-fungsi tertentu untuk mempermudah dalam pemanggilan *syntax* serta mempermudah *developer* atau *programmer* dalam mengembangkan suatu sistem tanpa harus membuat membuat fungsi atau *class* dari awal, dan menghemat waktu pengerjaan [31].

#### a. *Model-View-Controller (MVC)*

*Model-View-Controller (MVC)* adalah sebuah konsep untuk meng-enkapsulasi data bersama dengan pemrosesan (*model*), mengisolasi dari proses manipulasi (*controller*) dan tampilan (*view*) untuk direpresentasikan pada sebuah *user interface*. *MVC* merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web [32].

#### b. *Framework Laravel*

*Framework Laravel* adalah sebuah framework PHP yang menggunakan konsep *MVC (model view controller)*. *Laravel* adalah pengembangan *website* berbasis *MVP* yang ditulis dalam



*PHP* yang dirancang untuk meningkatkan kualitas perangkat lunak. Dengan menggunakan *Laravel*, *website* yang dihasilkan akan lebih dinamis. Kehadiran *framework Laravel* menjadikan bahasa pemrograman *PHP* menjadi lebih *powerful*. [33]

### 2.2.7 Basis Data (*Database*)

Basis data atau *database* adalah suatu aplikasi yang menyimpan sekumpulan data. *Database* yaitu kumpulan file-file yang berhubungan satu dengan yang lainnya, diatur sedemikian rupa sehingga dapat digunakan oleh beberapa program aplikasi database [34]. Basis data memungkinkan tempat penyimpanan data yang besar dan dapat digunakan secara bersamaan oleh banyak departemen dan pengguna. Database mewakili entitas, atribut, dan hubungan logis antara entitas. Basis data terdiri dari kumpulan data yang terorganisir, relasi antar data, dan objektifnya. Objektif utama adalah kecepatan dan kemudahan berinteraksi dengan data yang dikelola atau diolah. Selain itu terdapat pengertian bahwa basis data adalah sekumpulan data persisten yang digunakan oleh aplikasi sistem dari perusahaan.

#### a. *DBMS (Database Management System)*

*DBMS (Data Base Management System)* adalah koleksi terpadu dari aplikasi program (*system software*) yang dapat digunakan dalam melakukan definisi membuat melakukan akses dan memelihara *database*. *DBMS* menyediakan berbagai fasilitas yaitu [35]:

1. *DDL (Data Definition Language)* merupakan perintah-perintah yang biasa digunakan administrator database untuk mendefinisikan skema dan subskema database. Perintah yang termasuk di dalamnya yaitu :
  - a. *CREATE* : Digunakan untuk membuat termasuk diantaranya membuat database dan tabel baru.
  - b. *ALTER* : Digunakan untuk mengubah struktur tabel yang ada.
  - c. *DROP* : Digunakan untuk menghapus database dan tabel.

2. **DML (*Data Manipulation Language*)** merupakan perintah-perintah yang memungkinkan pengguna melakukan akses dan manipulasi data sebagaimana yang telah diorganisasikan sebelumnya dalam model data yang tepat, DML digunakan untuk memanipulasi *database* yang telah didefinisikan dengan DDL. Perintah yang termasuk DML adalah sebagai berikut :
- a. ***INSERT*** : Digunakan untuk menyisipkan atau memasukan data dalam tabel.
  - b. ***SELECT*** : Digunakan untuk mengambil data atau menampilkan data dari suatu tabel atau beberapa tabel.
  - c. ***UPDATE*** : Digunakan untuk memperbarui data lama menjadi data terkini.
  - d. ***DELETE*** : Digunakan untuk menghapus data dari tabel.

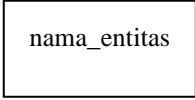
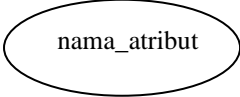
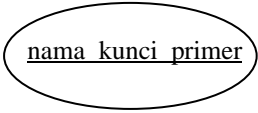
**b. *SQL (Structured Query Language)***

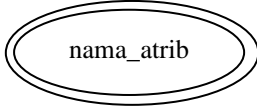
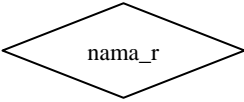
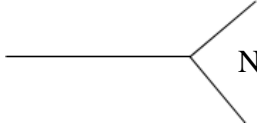
*SQL (Structured Query Language)* adalah bahasa yang digunakan untuk mengelola data pada RDBMS. Sifat Bahasa SQL adalah Bahasa tingkat tinggi (*high level*) [36]. MySQL merupakan basis data yang dikembangkan dari bahasa SQL (*structure query language*). MySQL dapat dikatakan sebagai *relational database management system* (RDBMS), yaitu hubungan antar table yang berisi data-data pada suatu database.

**c. *ERD (Entity Relationship Diagram)***

*ERD (Entity Relationship Diagram)* merupakan *tools* yang digunakan untuk memodelkan struktur data dengan menggambarkan entitas dan hubungan antar entitas (*relationship*) secara abstrak (*konseptual*). ERD berfungsi untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol [37]. Berikut tabel 2.6 adalah daftar simbol dalam ERD :

Tabel 2.6 Simbol-Simbol ERD [24]

No	Simbo	Keterangan
1.	Entitas / <i>entity</i>  	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanay lebih ke kata benda dan belum merupakan nama tabel.
2.	Atribut  	Filed atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer  	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom; asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).

4.	Atribut multi nilai / multivalue 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih.
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
6.	Asosiasi / <i>Association</i> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> yang menghubungkan entitas A dan entitas B.

### **2.2.8 Cuti**

Berdasarkan Kementerian Pendidikan dan Kebudayaan Republik Indonesia, landasan hukum yang mengatur tentang cuti adalah Pasal 5 UU nomor 8 tahun 1974. Cuti terdiri dari cuti tahunan, cuti besar, cuti sakit, cuti bersalin, cuti karena alasan penting, cuti diluar tanggungan institusi terkait [3].