

## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Penelitian Terkait

Menurut penelitian yang dilakukan oleh Andi Abdilah , dkk. pada tahun 2021 yang berjudul " Perancangan Sistem Informasi Cuti Berbasis Web Pada Koperasi Syariah Benteng Mikro Indonesia " di Koperasi Syariah Benteng Mikro Indonesia. Penelitian ini menganalisis proses pengajuan cuti dimana proses pengajuan cuti masih secara manual menggunakan *form* kertas, kekurangan dari sistem yang berjalan ini membuat proses pengajuan cuti menjadi kurang efisien.

Dengan permasalahan tersebut, Andi Abdilah , dkk. Merancang sebuah Sistem Informasi Pengajuan Cuti Berbasis Web. Dengan berbasis web, sehingga proses pengolahan cuti dapat dilakukan oleh karyawan maupun atasan secara online yang lebih efektif dan efisien.

Untuk metode penelitian, data dianalisis dengan *SWOT (Strength, Weakness, Opportunity, Threat)* guna mengetahui kelemahan dan kelebihan perusahaan. Metode perancangan yang digunakan yaitu *UML (Unified Modeling Language)* untuk menggambarkan rancangan sistem dan *PHP (Hypertext Preprocessor)* sebagai bahasa pemrograman dengan *database MySQL*. Untuk pengujian sistemnya menggunakan *blackbox testing* terhadap fungsional. Perancangan sistem informasi pengajuan cuti berbasis web oleh Andi Abdilah, dan kawan-kawan dapat meminimalisir kekurangan sistem

yang berjalan saat ini, dan dapat mempermudah para karyawan melakukan proses cuti[2].

## 2.2. Landasan Teori

### 2.2.1. BLUD ( Badan Layanan Umum Daerah)



Gambar 2.1 Ilustrasi Pegawai

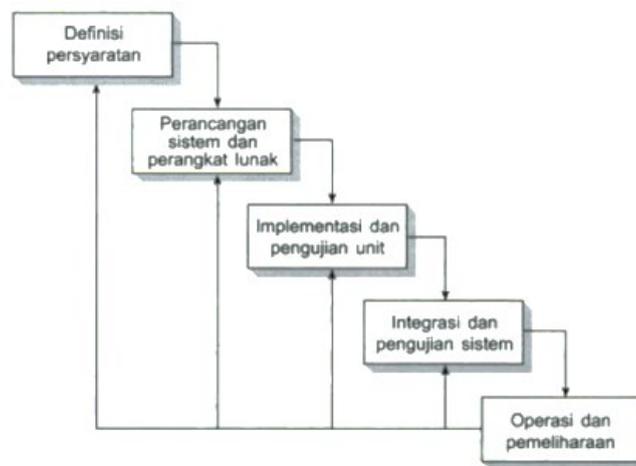
Badan Layanan Umum Daerah yang selanjutnya disingkat BLUD adalah Satuan Kerja Perangkat Daerah atau Unit Kerja pada Satuan Kerja Perangkat Daerah di lingkungan pemerintah daerah yang dibentuk untuk memberikan pelayanan kepada masyarakat berupa penyediaan barang dan/ atau jasa yang dijual tanpa mengutamakan mencari keuntungan, dan dalam melakukan kegiatannya didasarkan pada prinsip efisiensi dan produktivitas[3].

### 2.2.2. Metode *Waterfall*

Menurut Sanubari (2020) Metode *Waterfall* adalah suatu metode pengembangan *software* dengan cara pendekatan sistematis secara berurutan[4].

Pendekatan ini menganggap bahwa setiap tahapan dalam pengembangan perangkat lunak harus diselesaikan sebelum memasuki tahapan berikutnya, dan setiap tahapan menghasilkan

dokumen atau artefak yang menjadi dasar untuk tahapan selanjutnya. Metode *waterfall* mengandalkan perencanaan yang matang di awal proyek dan memiliki sifat yang sangat berurutan, dimana tahapan berikutnya hanya dimulai setelah tahapan sebelumnya selesai sepenuhnya.



Gambar 2.2 Metode *Waterfall*

### 2.2.3. *Web Server*



## Web Server

Gambar 2.3 logo *Web Server*

Menurut Abdulloh (2018:4) “*Web server* merupakan perangkat lunak yang berfungsi untuk menerima permintaan (*request*) melalui protocol *HTTP* atau *HTTPS* dari *client* kemudian mengirimkan kembali dalam bentuk halaman-halaman *web*”[5].

#### 2.2.4. *HTML (HyperText Markup Language)*



Gambar 2.4 logo *HTML*

Menurut Abdulloh (2018:7) “*HTML* merupakan singkatan dari *HyperText Markup Language* yaitu bahasa standar web yang dikelola penggunaannya oleh *W3C (World Wide Web Consortium)* berupa *tag-tag* yang menyusun setiap elemen dari website. membuat halaman *website* yang dapat diakses atau ditampilkan menggunakan *Web Browser*”[5].

#### 2.2.5. *CSS (Cascading Style Sheets)*



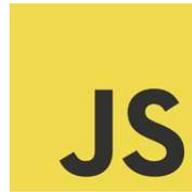
Gambar 2.5 logo *CSS*

Menurut Yudhanto & Prasetyo (2019:6) “*CSS* adalah *Cascading Stylesheet*, yaitu bahasa yang digunakan untuk *HTML* agar menjadi lebih bagus dan efektif dalam tampilan.”[6].

Dengan *CSS*, pengguna dapat membuat halaman web yang lebih menarik, konsisten, dan responsif untuk berbagai jenis perangkat. *CSS* bekerja dengan cara mendefinisikan aturan gaya (*style*

*rules*) yang mengontrol bagaimana setiap elemen *HTML* akan ditampilkan di browser. Ini memungkinkan pemisahan antara struktur (*HTML*) dan presentasi (*CSS*) dalam pengembangan web, sehingga memudahkan perubahan dan pemeliharaan halaman web.

### 2.2.6. *Javascript*



Gambar 2.6 logo *Javascript*

Bahasa pemrograman yang bersifat *client side* yang pemrosesnya dilakukan oleh *client*, sering digunakan pada *web browser* untuk menciptakan halaman web yang menarik.

Menurut Abdulloh (2018:193) “*JavaScript* merupakan bahasa pemrograman *web* yang pemrosesnya dilakukan di sisi *client*”[5].

### 2.2.7. *JQuery*



Gambar 2.7 logo *JQuery*

Menurut Abdulloh (2018:233) “*JQuery* merupakan salah satu dari sekian banyak *JavaScript library*, yaitu kumpulan fungsi *JavaScript* yang siap pakai, sehingga mempermudah dan mempercepat dalam membuat kode *JavaScript*”[5].

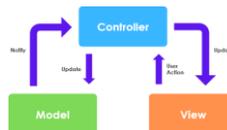
### 2.2.8. PHP (*Hypertext Preprocessor*)



Gambar 2.8 logo *PHP*

Menurut Yudhanto & Prasetyo (2019:9) “*PHP* atau *Hypertext Preprocessor* adalah Bahasa pemrograman *script server side* yang sengaja dirancang lebih cenderung untuk membuat dan mengembangkan web”[6].

### 2.2.9. Konsep *MVC (Model View Controller)*



Gambar 2. 9 konsep *MVC*

Menurut Yudhanto & Prasetyo (2019:8) “*Model View Controller* atau *MVC* adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (*Model*) dari tampilan (*View*) dan cara bagaimana memprosesnya (*Controller*)”[6].

#### 1. *Model*

*Model* mewakili struktur data. Biasanya *model* berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data, seperti memasukkan data ke basis data, pembaharuan data, dan lain-lain.

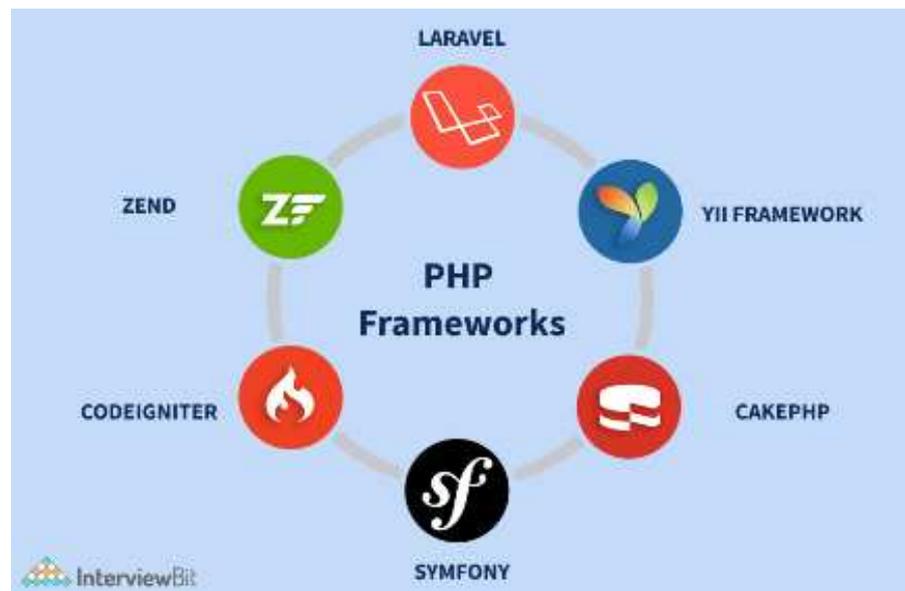
## 2. *View*

*View* adalah bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman *web*.

## 3. *Controller*

*Controller* merupakan bagian yang menjembatani *model* dan *view*. *Controller* berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman *web*.

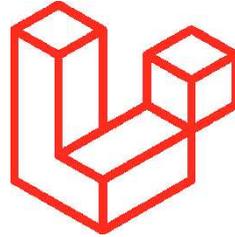
### 2.2.10. *Framework*



Gambar 2.10 *Framework PHP*

Menurut Yudhanto & Prasetyo (2019:10) “*Framework* adalah komponen pemrograman yang siap digunakan ulang kapan saja sehingga *programmer* tidak harus membuat *script* yang sama untuk tugas yang sama”[6].

### 2.2.11. *Laravel*



Gambar 2.11 logo *Laravel*

*Laravel* mempunyai keunggulan yang tidak dimiliki oleh *PHP* Framework lain. Menurut Zanin & Wernke (2019) dalam artikelnya yang berjudul “*A Comparative study of PHP frameworks performance*”, *laravel* mempunyai keunggulan dibandingkan dengan *Symfony* dan *CodeIgniter*. *Laravel* mempunyai permintaan per detik (*request per second*) tertinggi dibandingkan *Symfony* dan *CodeIgniter*. *Laravel* juga mempunyai penggunaan memori (*memory usage*) terendah dibandingkan *Symfony* dan *CodeIgniter*[7].

Selain dua hal itu, *laravel* juga unggul pada waktu respon (*response time*). *Laravel* mencatat *response time* terendah dibandingkan dengan *Symfony* dan *CodeIgniter*. Namun, *laravel* mempunyai kekurangan dibandingkan kedua *framework* lain, yaitu dalam hal jumlah *file* (*numbers of file*).

### 2.2.12. *Bootstrap*



Gambar 2.12 logo *Bootstrap*

Menurut Abdulloh (2018:261) “*Bootstrap* merupakan salah satu *Framework CSS* paling populer dari sekian banyak *framework CSS* yang ada”[5].

Dengan *Bootstrap*, pengembang dapat membangun desain yang konsisten dan responsif tanpa harus menulis kode *CSS* atau *JavaScript* secara manual. *Bootstrap* juga dilengkapi dengan dukungan untuk berbagai perangkat, mulai dari *desktop* hingga perangkat *mobile*, sehingga memastikan bahwa desain *web* dapat tampil dengan baik di berbagai layar dan ukuran *viewport*.

### 2.2.13. *AdminLTE*



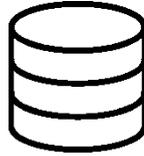
Gambar 2.13 *AdminLTE*

*AdminLTE* adalah salah satu *template* yang sering digunakan oleh *web developer* sebagai *template backend* pada proyek yang sering dikerjakan. Jadi *adminLTE* ini adalah sebuah *dashboard Administrator* dibuat menggunakan *bootstrap* yang merupakan *framework css* yang paling banyak digunakan.

### 2.2.14. **Basis Data**

Basis data kumpulan data yang tersimpan didalam komputer yang digunakan suatu program untuk memperoleh informasi dari

basis data tersebut.



Gambar 2.14 Simbol Basis Data

Ada beberapa jenis basis data, seperti basis data relasional (*SQL*), basis data berorientasi objek (*NoSQL*), dan basis data terdistribusi. Masing-masing jenis basis data memiliki karakteristik dan kegunaan yang berbeda, tergantung pada kebutuhan dan kompleksitas aplikasi yang akan dikembangkan.

#### 2.2.15. *MySQL*



Gambar 2.15 logo *MySQL*

Menurut Nugroho (2019:133) “*MySQL* merupakan *database* yang paling digemari dikalangan *Programmer Web*, dengan alasan bahwa program ini merupakan *database* yang sangat kuat dan cukup stabil untuk digunakan sebagai media penyimpanan data”[8].

#### 2.2.16. *Phpmyadmin*



Gambar 2.16 logo *phpMyAdmin*

*PhpMyAdmin* merupakan aplikasi *open source* yang bertujuan untuk mempermudah mengakses *MySQL* yang dapat membuat *database*, Tabel, *insert*, menghapus, dan memperbaharui data. Menurut Bennetch (2021) *PhpMiyAdmin* adalah perangkat lunak *open source* yang ditulis dalam bahasa *PHP* yang menjalankan administrasi *MySQL* melalui web dan mendukung berbagai operasi di *MySQL* dan *MariaDB*. Operasi yang sering digunakan seperti mengelola *database*, tabel, kolom, relasi, indeks, pengguna, dan izin[9].

#### 2.2.17. *Laragon*



Gambar 2.17 logo Laragon

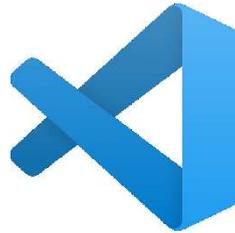
*Laragon* adalah lingkungan pengembangan universal yang *portabel*, terisolasi, cepat & kuat untuk *PHP*, *Node.js*, *Python*, *Java*, *Go*, *Ruby*. Ini cepat, ringan, mudah digunakan dan mudah diperluas.

*Laragon* sangat bagus untuk membangun dan mengelola aplikasi *web* modern. Ini berfokus pada kinerja, dirancang berdasarkan stabilitas, kesederhanaan, fleksibilitas, dan kebebasan[10].

#### 2.2.18. *Visual Studio Code*

*Visual Studio Code* adalah editor kode sumber yang ringan namun kuat yang berjalan di *desktop* pengguna dan tersedia untuk

*Windows, macOS* dan *Linux*. Muncul dengan dukungan bawaan untuk *JavaScript, TypeScript* dan *Node.js* dan memiliki ekosistem ekstensi yang kaya untuk bahasa dan *runtime* lain (seperti *C++, C#, Java, Python, PHP, Go, .NET*)[11].



Gambar 2.18 logo *Visual Studio Code*

#### 2.2.19. *Git*



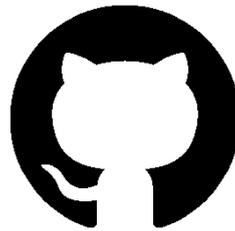
Gambar 2.19 logo *Git*

Salah satu *version control system* populer yang digunakan para *developer* untuk mengembangkan *software* secara bersama-sama. Dengan adanya *GIT* dapat melakukan *programming* dengan kolaborasi di dunia nyata dan di mana lokasinya yang harus sudah terhubung dengan *internet*. Pemanfaat *GIT* ini terdistribusi bukan tersentralisasi. *GIT* dibuat oleh Linus Torvalds[12].

#### 2.2.20. *Github*

*Github* adalah situs *web* dan layanan cloud yang bisa membantu para pengguna untuk menyimpan, mengelola dan mengembangkan.

Di dalam *Github* bisa mengupload *file*, membuat *file* yang mana *file*-nya bisa dikelola, dengan *version control system* punya *github*[13].



Gambar 2.20 logo *Github*

### 2.2.21. *Composer*



Gambar 2.21 *Composer*

Menurut Yosef Murya (2016) *Composer* adalah *dependency manager for PHP*, digunakan untuk mempermudah dalam instalasi aplikasi web berbasis *PHP*[14].

### 2.2.22. *Web Browser*



Gambar 2.22 *Web Browser*

Menurut Mariko (2019), *web browser* terdiri atas dua kata berbahasa Inggris, yaitu "*web*" dan "*browser*." Kata "*web*" merupakan singkatan dari *website*, yang berarti halaman situs yang ada di jaringan

internet, sedangkan "*browser*" dapat diartikan sebagai media penjelajah. Dari kedua arti kata tersebut, dapat disimpulkan bahwa *web browser* merupakan alat penjelajah halaman situs *website*[15].