

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Beberapa penelitian yang relevan diantaranya adalah penelitian dengan judul Impementasi Sistem Informasi Rekapitulasi Absensi Perkuliahan Mahasiswa pada STIKOM Dinamika Bangsa yang dilakukan oleh Merrylinteri Istoningtyas pada tahun 2021. Absensi digunakan sebagai tanda kehadiran seseorang dalam suatu kegiatan, seperti absensi kehadiran mahasiswa saat perkuliahan. Mahasiswa wajib melakukan absensi dengan cara menandatangani kertas daftar hadir pada setiap pertemuan perkuliahan. Setiap akhir semester absensi tersebut akan direkap oleh bagian staff prodi untuk mengetahui jumlah kehadiran mahasiswa. Hasil rekapan absensi ini digunakan sebagai syarat mahasiswa dapat mengikuti Ujian Akhir Semester (UAS) dengan minimal kehadiran 75% dari total pertemuan untuk setiap mata kuliah. Perekapan data absensi perkuliahan mahasiswa diinputkan menggunakan aplikasi pengolah angka yang mnimbulkan beberapa masalah. Seperti, pihak BAAK memerlukan waktu lama untuk memberikan pengumuman nama mahasiswa yang tidak dapat mengikuti Ujian Akhir Semester (UAS), menggunakan banya kertas untuk membuat pengumuman mahasiswa yang tidak dapat mengikuti Ujian Akhir Semester (UAS), dan mahasiswa akan terlambat dalam mendapatkan informasi tentang ruang Ujian Akhir Semester. Berdasarkan permasalahan tersebut, diperlukan sistem yang dapat memudahkan pihak BAAK untuk mengolah data absensi perkuliahan mahasiswa dengan cepat dan membantu penyebaran informasi hasil rekapan absensi, jadwal dan ruang pelaksanaan UAS kepada mahasiswa. Sistem yang dirancang ini dibuat dengan menggunakan bahasa pemrograman PHP dan database MySQL [7].

Penelitian selanjutnya berjudul Rancang Bangun Aplikasi Sistem Informasi Absensi Karyawan Online yang dilakukan oleh Rully Roosdianto, dkk pada tahun 2021. CV Cahaya Toner merupakan perusahaan swasta yang bergerak di bidang retail toner dan printer. Saat ini CV Cahaya Toner dalam menangani data karyawan terutama data absensi masih menggunakan cara konvensional, yaitu dengan dicatat pada buku absensi dan membubuhkan tanda tangan. Hal ini tentu saja memiliki

dampak terhadap efektifitas pendataan, efisiensi pendataan, pencarian data, dan perhitungan rekap absensi yang membutuhkan waktu lebih lama, sehingga resiko kesalahan data absensi semakin besar. Oleh karena itu, dibutuhkan aplikasi untuk mempermudah pengolahan informasi absensi karyawan bagi bagian kepegawaian dan manager dapat mengolah data karyawan. Sistem yang dibangun dengan memanfaatkan website menjadikan sistem ini dapat diakses dimanapun dan kapanpun. Sistem ini dibuat dengan *framework codeigniter*, konsep pemodelan sistem dengan *Unified Modeling Language (UML)* dan mempresentatikan *user interface* dan desain database dengan ERD dan LRS. Metode pengembangan sistem yang digunakan adalah metode *waterfall*, sedangkan metode pengujian sistem yang digunakan adalah *blackbox testing* [8].

Penelitian selanjutnya berjudul Perancangan dan Implementasi Sistem Informasi Absensi Berbasis Web Menggunakan Metode Agile Software Development oleh Tata Ayunita Pertiwi, dkk pada tahun 2023. Dinas Perdagangan Koperasi dan UMK merupakan urusan pemerintah yang bergerak di bidang koperasi dan usaha kecil menengah, perdagangan, dan perindustrian. Disperdagkopumk menerapkan pencatatan absensi kehadiran untuk para pegawainya saat ini menggunakan kertas dan tanda tangan para pegawai untuk pencatatannya. Berdasarkan permasalahan tersebut, sistem yang dirancang diharapkan dapat membantu mempersingkat waktu, mempermudah pekerjaan dan membangun perkembangan sesuai tujuan perusahaan. Sistem ini dibuat dengan metode *agile software development* dan menggunakan metode pengujian *blackbox testing* [9].

Penelitian selanjutnya berjudul Sistem Informasi Rekapitulasi Absensi Dosen Berbasis Web di Politeknik Piksi Input Serang oleh Riki Aldi Pari, dkk pada tahun 2023. Absensi kehadiran dosen merupakan kegiatan yang harus dilakukan sebelum dan sesudah proses pembelajaran baik secara online maupun offline. Laporan absensi yang dihasilkan dapat digunakan untuk bahan rekapitulasi dan evaluasi dosen oleh pihak akademik. Proses absensi dosen di Politeknik Piksi Input Serang dilakukan dengan cara mengisi agenda pembelajaran pada website resmi Politeknik Piksi Input Serang. Dengan mengisi agenda pembelajaran tersebut, dosen akan dianggap telah melakukan absensi. Pihak akademik akan merekap absensi dosen secara manual yang memungkinkan terjadi kesalahan menghitung data absensi yang nantinya akan berpengaruh kepada penghasilan dosen tersebut. Berdasarkan permasalahan tersebut,

dibutuhkan sistem yang memudahkan pihak akademik dalam pencatatan dan rekapitulasi data absensi dosen secara otomatis sehingga dapat dijadikan sebagai dasar untuk perhitungan gaji. Sistem ini dibangun berbasis *website* dengan bahasa pemrograman PHP, framework codeigniter, basis data MySQL, dan server apache [10].

Pada penelitian ini, peneliti bermaksud untuk membangun Sistem Informasi Rekapitulasi Kehadiran Pegawai Pada Politeknik Negeri Cilacap yang menangani proses rekapitulasi kehadiran pegawai. Proses presensi di Politeknik Negeri Cilacap sudah menggunakan mesin fingerprint yang terhubung dengan mesin *server internal*. Namun, dalam proses perekapannya masih menggunakan cara konvensional yang memungkinkan terdapat beberapa kendala seperti membutuhkan waktu yang lebih lama dan rentan akan *human error*. Oleh karena itu, penelitian ini dibangun untuk mempermudah bagian absensi untuk melakukan pengelolaan data kehadiran pegawai hingga menjadi laporan rekapitulasi kehadiran pegawai yang nantinya akan dibutuhkan oleh pegawai dan kepala sub bagian umum. Sistem ini akan dibangun dengan berbasis *website* menggunakan *framework* laravel dan metode pengembangan sistem yang digunakan adalah metode *prototype*. Metode pengujian sistem yang digunakan adalah *blackbox testing*.

Tabel 2. 1 Tabel Perbandingan

| Penulis, Tahun dan Judul | Tujuan | Metode | Hasil yang dicapai |
|--|--|---|--|
| Merrylinteri Istoningtyas (2021) “Implementasi Sistem Informasi Rekapitulasi Absensi Perkuliahan Mahasiswa Pada STIKOM Dinamika Bangsa” | Memper memudahkan pihak BAAK dan mahasiswa untuk mengetahui rekapitulasi absensi perkuliahan, memberikan informasi nama-nama mahasiswa yang tidak bisa mengikuti ujian akhir, dan mempermudah pembuatan absensi ujian tugas akhir. | <ul style="list-style-type: none"> - Penelitian ini menggunakan metode <i>waterfall</i>. - Menggunakan bahasa pemrograman PHP - Basis data MySQL | Menghasilkan aplikasi yang dapat membantu pihak akademik untuk mengolah data rekapan absensi mahasiswa secara optimal. Dengan sistem ini, mahasiswa dapat melihat sendiri hasil rekapan mahasiswa, jadwal, dan ruang UAS masing-masing tanpa harus datang ke kampus. |
| Rully Roosdianto, dkk (2021) “Rancang Bangun Aplikasi Sistem | Merancang sistem informasi absensi karyawan agar menghasilkan sistem informasi yang efektif dan efisien serta | <ul style="list-style-type: none"> - Metode pengembangan sistem yang digunakan adalah metode <i>waterfall</i>. - Konsep pemodelan sistem dengan UML. | Menghasilkan sistem absensi berbasis <i>website</i> pada CV. Cahaya Toner yang dapat mempermudah pengolahan informasi |

| Penulis, Tahun dan Judul | Tujuan | Metode | Hasil yang dicapai |
|---|--|---|---|
| Informasi Absensi Karyawan <i>Online</i> " | mempermudah dalam pengolahan data rekap abseni karyawan. | <ul style="list-style-type: none"> - Menggunakan <i>framework CodeIgniter</i>. - Metode pengujian menggunakan <i>blackbox testing</i>. | absensi karyawan bagi bagian kepegawaian. |
| Tata Ayunita Pertiwi, dkk (2023) "Perancangan dan Implementasi Sistem Informasi Absensi Berbasis <i>Web</i> Menggunakan Metode <i>Agile Software Development</i> " | Membantu untuk mempersingkat waktu, mempermudah pekerjaan dan membangun perkembangan sesuai tujuan perusahaan. | <ul style="list-style-type: none"> - Dalam perancangan dan pengembangan website menggunakan metode <i>Agile Software Development</i> untuk membantu developer bekerja dengan cepat dan menyesuaikan dengan kebutuhan <i>user</i> terhadap sistem. - Menggunakan metode pengujian Black Box Testing. | Mempermudah jalannya absensi pegawai di Disperdagkopumk Kab. Kampar tanpa adanya kecurangan dan kesalahan saat melakukan absen bagi pegawai dan mengurangi biaya operasional. |

| Penulis, Tahun dan Judul | Tujuan | Metode | Hasil yang dicapai |
|---|---|--|---|
| <p>Riki Aldi Pari, Panca Indah Lestari, Wangga Nugrahtama (2023)</p> <p>“Sistem Informasi Rekapitulasi Absensi Dosen Berbasis Web Di Politeknik Piksi Input Serang”</p> | <p>Merancang sebuah perancangan aplikasi absensi dosen berbasis <i>website</i> untuk membantu merekap data absensi dosen.</p> | <ul style="list-style-type: none"> - Metode penulisan yang digunakan adalah pendekatan deskriptif kualitatif yang bertujuan untuk mendeskripsikan hasil dan objek yang diperoleh. - Metode pengembangan sistem menggunakan metode <i>waterfall</i>. - Metode pengujian dengan pendekatan <i>blackbox testing</i>. - Bahasa pemrograman PHP - Menggunakan framework CodeIgniter. | <p>Menghasilkan sistem yang memudahkan pihak akademik dalam pencatatan dan rekapitulasi data absensi dosen secara otomatis dan dapat dijadikan sebagai dasar untuk pembayaran honorer atau gaji sesuai dengan data absensi yang tercatat dan tersimpan di dalam sistem.</p> |

| Penulis, Tahun dan Judul | Tujuan | Metode | Hasil yang dicapai |
|---|---|---|--|
| <p>Wulan Aulia Putri (2023)</p> <p>“Sistem Informasi Rekapitulasi Kehadiran Pegawai Pada Politeknik Negeri Cilacap”</p> | <p>Membantu bagian absensi di Bagian Administrasi Umum dan Kepegawaian dalam proses rekapitulasi kehadiran pegawai Politeknik Negeri Cilacap.</p> | <ul style="list-style-type: none"> - Metode penelitian yang digunakan pada penelitian ini adalah metode <i>prototype</i>. - Metode pengujian sistem menggunakan <i>black box testing</i>. - Menggunakan <i>framework Laravel</i> | <p>Penelitian ini menghasilkan sistem informasi rekapitulasi kehadiran pegawai Politeknik Negeri Cilacap yang dapat mempermudah bagian absensi dalam melakukan proses rekapitulasi kehadiran pegawai dan mempermudah pegawai untuk mendapatkan informasi mengenai hasil rekapitulasi kehadirannya masing-masing.</p> |

2.2 Landasan Teori

Penelitian ini diperlukan adanya teori-teori yang mendasar untuk menunjang proses penelitian ini. Teori-teori yang digunakan dalam penelitian ini sebagai berikut :

2.2.1 Sistem Informasi

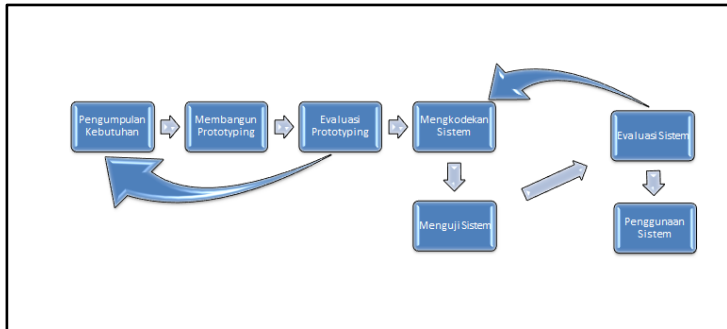
Sistem informasi merupakan sarana yang bertujuan untuk mengolah data menjadi sebuah informasi untuk kebutuhan manajemen dalam mengambil keputusan untuk menjalankan operasional sebuah organisasi. Sistem informasi juga dapat diartikan sebagai sebuah media untuk menyebarkan informasi kepada pengguna dengan cepat dan tepat [11].

2.2.2 Rekapitulasi

Rekapitulasi merupakan hasil akhir dari suatu perhitungan (kalkulasi) atau gabungan perhitungan yang berisi angka-angka yang disajikan dalam bentuk kolom-kolom [12]. Sedangkan pendapat lain mengatakan bahwa rekapitulasi merupakan suatu tindakan yang bertujuan untuk meringkas suatu data dan fakta menjadi lebih baik sehingga dapat menggambarkan hasil dari kegiatan dokumentasi [13].

2.2.3 Metode Prototype

Metode *prototype* merupakan salah satu metode pengemangan perangkat lunak yang banyak digunakan. Dengan menggunakan metode *prototype*, pengembang dan klien dapat saling berinteraksi selama proses pembuatan sistem sehingga dapat mengetahui kebutuhan pengguna secara terperinci. Karena terkadang sering terjadi klien yang hanya mendefinisikan secara umum apa yang diinginkan tanpa menyebutkan detail output, pemrosesan, dan data-data yang dibutuhkan. Sebaliknya, dari sisi pengembang terkadang kurang memperhatikan efisiensi algoritma, kemampuan sistem operasi, dan interface yang menghubungkan pengguna dan sistem [14].



Gambar 2. 1 Metode Prototyping (Pressman, 2012)

Berikut adalah tahap-tahap dalam metode *prototyping* [14]:

1. Pengumpulan kebutuhan
Tahapan pertama yang dilakukan dalam penelitian ini adalah pengumpulan kebutuhan sistem. Pada tahap ini dilakukan identifikasi kebutuhan sistem dan garis besar sistem yang akan dibangun.
2. Membangun *prototyping*
Tahapan selanjutnya adalah membangun *prototyping* yang berfokus pada pembuatan desain sementara yang berpusat pada penyajian kepada pelanggan, misalnya dengan membuat format *input* dan *output*.
3. Evaluasi *prototyping*
Pada tahap ini, dilakukan evaluasi terhadap *prototyping* yang sudah dibangun untuk memastikan sistem yang dirancang sudah sesuai dengan kebutuhan dan harapan pelanggan. Jika sudah sesuai akan dilanjutkan ke tahap pengkodean (*coding*), namun jika belum sesuai maka akan diperbaiki dengan mengulang kembali tahap 1, tahap 2, dan tahap 3.
4. Mengkodekan sistem
Tahap ini dilakukan jika *prototyping* yang dibuat sudah disetujui, yang dilanjutkan dengan pengkodean sistem menggunakan bahasa pemrograman yang sesuai
5. Menguji sistem
Tahap ini dilakukan untuk menguji sistem yang telah selesai dibuat. Sistem yang telah selesai dibuat kemudian akan dilakukan pengujian terlebih dahulu untuk menentukan apakah sistem sudah dapat

berfungsi dengan baik sesuai yang diharapkan atau tidak. Pengujian yang dilakukan pada sistem ini yaitu menggunakan metode *black-box testing*.

6. Evaluasi sistem
Tahap ini dilakukan untuk mengetahui apakah sistem yang dibangun telah sesuai dengan yang diharapkan. Apabila telah dinyatakan sesuai maka dapat dilakukan tahap selanjutnya, tetapi jika tidak maka perlu kembali ke tahap 4 dan tahap 5.
7. Menggunakan sistem
Sistem yang dibangun sudah siap digunakan oleh pelanggan setelah tahap evaluasi berhasil dan diterima oleh pelanggan.

2.2.4 Framework Laravel

Laravel adalah kerangka kerja (*framework*) berbasis PHP yang *open-source* dan dibangun dengan konsep MVC (*Model-View-Controller*) yang digunakan untuk membangun aplikasi *website* [15]. *Laravel* merupakan pengembangan *website* berbasis MVP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, serta untuk meningkatkan pengalaman bekerja dengan aplikasi yang menyediakan sintaks yang ekspresif, jelas, dan menghemat waktu [16].

Selain itu, *Laravel* juga mempermudah proses pengembangan *website* dengan beberapa fitur unggulan, diantaranya adalah *template engine*, *routing*, dan *modularity*. *Responsive user interface* merupakan fitur pada halaman *website* yang mampu menyesuaikan tampilan dengan perangkat yang digunakan pengguna. Dengan adanya fitur ini dapat meningkatkan kenyamanan pengguna yang menggunakan perangkat komputer maupun perangkat mobile yang tampilannya lebih kecil [17].

2.2.5 Pengujian (Testing)

Pengujian merupakan salah satu tahapan yang harus dilalui untuk menghasilkan sebuah sistem yang berkualitas. Melalui pengujian dapat menemukan kekurangan dan kesalahan yang muncul di dalam sistem yang telah dibuat sehingga dapat segera memperbaikinya. Tujuannya adalah menyesuaikan fungsi-fungsi dengan spesifikasi yang diinginkan dan agar sistem dapat dikatakan layak untuk digunakan [18]. Pengujian yang dilakukan dalam penelitian ini adalah pengujian *black-box*.

Pengujian *black-box* adalah pengujian yang berfokus pada tampilan antarmuka (*interface*), fungsi dan kesesuaian pada alur yang dibutuhkan

oleh pengguna. Berbeda dengan pengujian *white-box* yang menguji kode program, pengujian *black-box* tidak menguji bahkan melihat kode program. Pengujian *black-box* dilakukan dengan beberapa tahapan sebagai berikut [6]:

1. Membuat *test case* untuk pengujian fungsi yang terdapat di dalam sistem
2. Membuat *test case* untuk pengujian kesesuaian alur dari kerja suatu fungsi pada program cocok dengan kebutuhan dan permintaan dari pengguna
3. Mencari *bugs/error* berdasarkan tampilan antarmuka (*interface*) pada sistem

Dalam melakukan pengujian harus memilih teknik yang tepat. Teknik yang tepat yaitu teknik yang dapat menemukan kesalahan yang belum terdeteksi sehingga dapat meningkatkan kualitas perangkat lunak (*software*) [6]. Pengujian *black-box* dapat menemukan klasifikasi kesalahan dalam beberapa kategori sebagai berikut [18].

1. Fungsi yang salah atau hilang
2. Kesalahan tampilan antarmuka (*interface*)
3. Kesalahan struktur data atau akses *database eksternal*
4. Kesalahan performa atau kinerja
5. Kesalahan inisialisasi ataupun terminal

Adapula keuntungan penggunaan metode pengujian *black-box* adalah [19]:

1. Penguji tidak perlu memiliki atau menguasai pengetahuan tentang bahasa pemrograman tertentu;
2. Pengujian dilakukan dari sudut pandang pengguna sistem;
3. *Programmer* dan penguji keduanya saling bergantung satu sama lain.

2.2.6 MySql

MySQL (*My Structured Query Language*) merupakan sebuah perangkat lunak yang terdapat di dalam sistem basis data SQL atau sering disebut dengan *Database Management System* (DBMS) yang *multithread* dan *multi-user* [20]. DBMS ini sifatnya *open source* sehingga dapat digunakan untuk aplikasi *multi-user* atau banyak pengguna [21].

MySQL termasuk ke dalam jenis RDBMS (*Relational Database Management System*) *server*. RDBMS adalah program yang memungkinkan pengguna database untuk membuat, mengelola, dan menggunakan data dengan model *relational* [22]. *Relational* artinya adalah

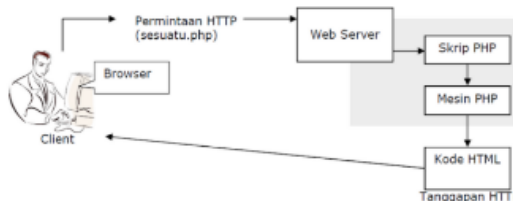
data yang dikelola dalam database yang akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan jauh lebih cepat. MySQL dapat digunakan untuk mengelola database mulai dari yang kecil sampai yang sangat besar [21].

2.2.7 HTML

Hypertext Markup Language (HTML) adalah suatu bahasa komputer yang digunakan untuk membuat dokumen *hypertext* yang portabel dan satu *platform* ke yang lainnya. Dokumen HTML adalah dokumen SGML dengan semantik umum yang sesuai dalam penyajian informasi dengan jangkauan luas aplikasi. HTML sudah menjadi bahasa penghubung *WWW publishing* [23].

2.2.8 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa berbentuk skrip yang ditempatkan dan diproses di dalam server dan hasilnya dikirimkan ke *user* dalam bentuk halaman web yang diakses menggunakan *browser*. PHP membuat *website* menjadi dinamis untuk membuat tampilan website sesuai permintaan dan apabila terjadi perubahan konten bisa dilakukan dengan mudah karena data tersimpan di *database* [24].



Gambar 2. 2 Cara Kerja PHP

Cara kerja dari PHP dimulai dengan *client* mengirimkan permintaan dari suatu halaman *website* oleh *browser*. Berdasarkan URL (*Uniform Resource Locator*) atau dikenal dengan alamat internet, *browser* mendapat alamat dari web server, mengidentifikasi alamat yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh web server. Selanjutnya web server akan mengirimkan informasi ke mesin php dan mesin inilah yang bertugas untuk memproses dan memberikan hasilnya berupa kode html ke web server, selanjutnya web server menyampaikan kepada *client* [25].



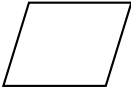
2.2.9 Website



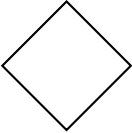
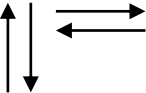

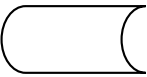
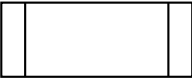

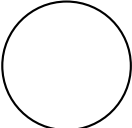
Website adalah sekumpulan dokumen berupa halaman web yang berisi teks dalam format *Hyper Text Markup Language* (HTML). Website disimpan di *server hosting* yang dapat diakses menggunakan browser dengan jaringan internet melalui alamat internet berupa *Uniform Resource Locator* (URL). *Website* terdiri dari dua jenis yaitu website statis (*static website*) dan website dinamis (*dynamic website*). Website statis merupakan website yang dimana informasi pada halaman harus diubah secara manual dengan cara merubah *source code*. Sedangkan *website* dinamis merupakan website yang dapat diperbarui secara berkala, karena informasi website tersimpan pada database [24].

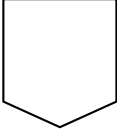
2.2.10 Flowchart Sistem

Flowchart atau diagram alir adalah suatu jenis diagram yang merepresentasikan algoritma atau langkah-langkah suatu proses yang sistematis dalam sebuah sistem. *Flowchart* digunakan oleh seorang analis sistem sebagai bukti dokumentasi untuk menjelaskan gambaran logis sebuah sistem yang akan dibangun kepada programmer. Dengan begitu, *flowchart* dapat membantu untuk memberikan solusi terhadap masalah yang terjadi dalam membangun sebuah sistem. Pada dasarnya, *flowchart* digambarkan dengan menggunakan simbol-simbol yang mewakili suatu proses tertentu [26]. Berikut adalah simbol-simbol pada *flowchart* seperti pada Tabel 2.2.

Tabel 2. 2 Simbol-simbol *flowchart* sistem

| No | Simbol | Nama | Keterangan |
|----|---|-----------------------|---|
| 1. |  | <i>Terminal</i> | Digunakan untuk memulai atau mengakhiri program |
| 2. |  | <i>Processing</i> | Digunakan untuk menunjukkan pengolahan yang akan dilakukan dalam komputer |
| 3. |  | <i>Input / Output</i> | Digunakan untuk menyatakan input dan output tanpa melihat jenisnya |

| No | Simbol | Nama | Keterangan |
|-----|---|---------------------------|--|
| 4. |  | <i>Manual Operation</i> | Digunakan untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer |
| 5. |  | <i>Manual Input</i> | Digunakan untuk menginputkan data secara manual dengan keyboard |
| 6. |  | <i>Decision</i> | Digunakan untuk memilih proses yang akan dilakukan berdasarkan kondisi tertentu |
| 7. |  | <i>Flow</i> | Berfungsi untuk menghubungkan simbol yang satu dengan yang lainnya, menyatakan arus suatu proses |
| 8. |  | <i>Document</i> | Digunakan untuk menyatakan masukan dan keluaran yang berasal dari dokumen |
| 9. |  | <i>Disk Storage</i> | Digunakan untuk menyatakan masukan dan keluaran yang berasal dari disk atau disimpan di disk |
| 10. |  | <i>Predefined Process</i> | Digunakan untuk mempersiapkan penyimpanan yang sedang atau akan digunakan dengan memberikan harga awal |
| 11. |  | <i>Display</i> | Digunakan untuk menyatakan keluaran melalui layar monitor |
| 12. |  | <i>Connector</i> | Digunakan untuk menyatakan sambungan dari proses yang satu ke proses berikutnya di halaman yang sama |

| No | Simbol | Nama | Keterangan |
|-----|---|--------------------------|---|
| 13. |  | <i>Offline Connector</i> | Digunakan untuk menyatakan sambungan dari proses yang satu ke proses berikutnya di halaman yang berbeda |


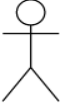

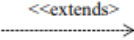
2.2.11 Unified Modeling Language (UML)

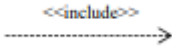
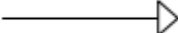

Unified Modeling Language (UML) adalah suatu bahasa standar yang digunakan untuk menggambar, merancang, dan mendokumentasikan model dalam pemrograman berorientasi objek [27].

1. Use Case Diagram

Use case diagram merupakan perancangan untuk kelakuan sistem informasi yang akan dibangun. *Use case diagram* menggambarkan interaksi antar aktor dengan sistem informasi yang akan dibangun. Diagram ini digunakan untuk mengetahui fungsi yang ada pada sistem informasi tersebut dan pengguna yang berhak untuk mengakses fungsi-fungsi tersebut. Berikut ini simbol-simbol use case diagram seperti pada Tabel 2.3 [28].

Tabel 2. 3 Simbol-Simbol Use Case Diagram


| 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 actor. |
| 2. |  | <i>Actor</i> | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case. |
| 3. |  | <i>Association</i> | Menghubungkan antara objek satu dengan objek lainnya. |
| 4. |  | <i>Extend</i> | Menspesifikasikan bahwa use case target |






| No | Simbol | Nama | Keterangan |
|----|---|-----------------------|---|
| | | | memperluas perilaku dari use case sumber pada suatu titik yang diberikan. |
| 5. |  | <i>Include</i> | Menspesifikasikan bahwa use case sumber secara eksplisit. |
| 6. |  | <i>Generalization</i> | Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor). |
| 7. |  | <i>System</i> | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |

2. Sequence Diagram

Sequence diagram merupakan diagram yang digunakan untuk mendeskripsikan interaksi antar objek pada sebuah sistem secara urut dan terperinci. Untuk menggambarkan *sequence diagram* harus mengetahui objek-objek yang terlibat dan metode yang digunakan dalam *use case diagram*. Saat membuat *sequence diagram* perlu untuk melihat skenario yang ada pada *use case diagram*. Berikut adalah simbol-simbol yang ada pada *sequence diagram*, sesuai pada Tabel 2.4 [28].

Tabel 2. 4 Simbol-Simbol Sequence Diagram


| No | Simbol | Nama | Keterangan |
|----|---|--------------|---|
| 1. |  | <i>Actor</i> | Menggambarkan user atau pengguna yang berinteraksi dengan sistem. |

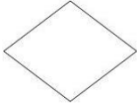



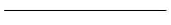
| No | Simbol | Nama | Keterangan |
|----|---|---|--|
| 2. |  | <i>Entity Class</i> | Menggambarkan hubungan yang akan dilakukan. |
| 3. |  | <i>Boundary Class</i> | Menggambarkan sebuah gambaran dari form. |
| 4. |  | <i>Control Class</i> | Digunakan sebagai penghubung antara boundary dengan tabel. |
| 5. |  | <i>A Focus of Control & A Life Line</i> | Menggambarkan tempat mulai dan berakhirnya message. |
| 6. |  | <i>Message</i> | Menggambarkan pengiriman pesan. |

2.2.12 Entity Relationship Diagram (ERD)

ERD (*Entity Relationship Diagram*) merupakan sebuah teknik yang digunakan untuk memodelkan kebutuhan data, biasanya digunakan dalam kegiatan pengembangan sistem pada tahap analisis. ERD digunakan sebagai representasi grafis dari logika database dengan menyertakan deskripsi detail mengenai seluruh entitas (*entity*), hubungan (*relationship*), dan batasan (*constraint*) [29]. Simbol-simbol yang digunakan dalam ERD dapat dilihat pada Tabel 2.5 [30].

Tabel 2. 5 Simbol-simbol ERD

| No. | Simbol | Nama | Keterangan |
|-----|---|---------|--|
| 1. |  | Entitas | Entitas merupakan data inti yang akan disimpan, bakal tabel pada basis data, benda yang memiliki data dan harus disimpan datanya |

| No. | Simbol | Nama | Keterangan |
|-----|---|----------------------|--|
| | | | agar dapat diakses oleh aplikasi komputer. |
| 2. |  | Relasi | Relasi yang menghubungkan antar entitas. Biasanya diawali dengan kata kerja. |
| 3. |  | Atribut | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas |
| 4. |  | Atribut Kunci Primer | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan. |
| 5. |  | Atribut Multinilai | <i>Field</i> yang butuh disimpan dalam entitas yang dapat memiliki nilai lebih dari satu. |
| 6. |  | Alur | Alur memiliki fungsi untuk menghubungkan atribut dengan entitas dan entitas dengan relasi. |

2.2.13 Basis Data

Basis data adalah kumpulan data terstruktur yang disimpan secara terorganisir dalam sebuah sistem komputer. Basis data mengacu pada kumpulan informasi terkait yang disimpan pada format yang dapat diakses, dikelola, dan diperbarui secara efisien. Basis data digunakan untuk menyimpan dan mengelola data yang diperlukan oleh suatu organisasi.

Sistem basis data adalah perangkat lunak yang digunakan untuk menyimpan, mengelola, dan mengakses data secara efisien. Sistem basis data bertanggungjawab atas manajemen data yang terorganisir dalam

struktur yang terdefinisi dengan baik. Sistem ini memungkinkan pengguna untuk menyimpan, mengambil, memperbarui, dan menghapus data dengan mudah untuk proses pengambilan keputusan[31].

Dalam memasukan dan mengambil suatu data dari dan ke dalam media penyimpanan data membutuhkan bantuan perangkat lunak yang disebut dengan sistem manajemen basis data (DBMS). Database Management System (DBMS) adalah perangkat lunak yang digunakan untuk mengelola dan mengatur basis data. DBMS memungkinkan pengguna untuk membuat, mengatur, dan mengelola basis data serta mengakses dan memanipulasinya. DBMS merupakan perantara bagi pengguna dan database, untuk dapat melakukan interaksi diperlukan bahasa pemrograman tertentu [32]. Bahasa pemrograman tersebut terdiri dari dua macam yaitu sebagai berikut :

- a. Data Definition Language (DDL) dapat digunakan oleh pengguna untuk membuat tabel baru, mengubah tabel, dan memuat indeks.
- b. Data Manipulation Language (DML) dapat digunakan oleh pengguna untuk memanipulasi dan pengambilan data dari database, seperti contohnya menambahkan data, menghapus data, dan mengubah data.