

## **BAB II**

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

#### **2.1. Tinjauan Pustaka**

Penelitian mengenai pembuatan atau pembangunan sebuah perangkat lunak sistem informasi berbasis web yang bertujuan untuk mendigitalisasi suatu pengelolaan data yang masih manual atau konvensional telah banyak dilakukan. Hal tersebut menunjukkan dengan menerapkan teknologi berupa pembangunan sebuah sistem informasi dengan basis web dalam upaya untuk mendigitalisasikan suatu proses yang masih manual membawa manfaat berupa kinerja yang lebih lancar, mudah, serta lebih efektif dan efisien, dimana hal tersebut juga akan memberikan dampak positif baik bagi individu maupun sebuah organisasi atau perusahaan. Berikut adalah beberapa penelitian sebelumnya yang penulis gunakan sebagai acuan literasi dalam pelaksanaan penelitian penulis.

Penelitian pertama dilakukan oleh Fahrival dan rekannya pada tahun 2018, dengan judul “Perancangan Sistem Inventory Barang Pada UD. Minang Dewi Berbasis Website”. Penelitian ini membahas mengenai pembangunan sebuah sistem informasi inventory barang berbasis web menggunakan metode pembangunan perangkat lunak dengan pendekatan yang sistematis dan sekuensial, yaitu metode air terjun (*waterfall*). Belum adanya sistem informasi dengan basis data untuk mengelola data barang di sebuah gudang menjadi latar belakang dalam penelitian ini. Berangkat dari latar belakang tersebutlah dibuat sebuah sistem informasi berbasis web pada penelitian ini agar dapat mempermudah karyawan mengetahui data stok barang yang ada, mengontrol data keluar dan masuk barang serta menghasilkan keluaran laporan data keluar dan masuk barang. Metode pengumpulan data yang diterapkan pada penelitian ini berupa wawancara, observasi dan studi pustaka. Teknologi yang digunakan dalam pembuatan sistem informasi pada penelitian ini diantaranya adalah *Unified Modelling Language* (UML) untuk proses perancangan dan desain serta bahasa pemrograman PHP dan basis data MySQL untuk proses konstruksi sistem. Hasil sistem informasi-

-inventory yang dibuat memberikan dampak positif berupa proses pengelolaan data persediaan barang menjadi lebih cepat, akurat, dan mudah sehingga lebih baik daripada sistem sebelumnya. Pengelolaan data juga lebih memudahkan karyawan bagian admin sehingga berefek pada meningkatnya kinerja organisasi atau perusahaan. Pemilihan teknologi sistem informasi yang berbasis web juga menjadikan sistem lebih mudah dipahami dan dipelajari oleh pengguna pada penelitian ini [11].

Kemudian penelitian kedua dengan judul “Aplikasi Pengelolaan Data Barang pada Bagian Perlengkapan (Studi Kasus Balai Veteriner Banjarbaru)”, dilakukan oleh Eka Chandra Kirana pada tahun 2018. Sama seperti penelitian sebelumnya, penelitian ini juga membahas tentang pembangunan sebuah sistem informasi pengelolaan data barang, hanya pada penelitian ini studi kasus bertempat di bagian perlengkapan pada Unit Pelaksana Teknis (UPT) Direktorat Jenderal (Dirjen) Peternakan dan Kesehatan Hewan Balai Veteriner Banjarbaru. Latar belakang dan tujuan pembuatan aplikasi sistem informasi pada penelitian ini adalah untuk membantu pengelolaan data barang yang ada di gudang agar dapat menghasilkan keluaran laporan data yang lebih efektif dan efisien dari sistem yang digunakan sebelumnya yang masih menggunakan cara manual. Metode pada penelitian ini mengacu pada model *System Development Life Cycle* (SDLC) untuk proses pembangunan sistem perangkat lunak atau aplikasinya. Metode tersebut meliputi tahap analisis kebutuhan sistem, perancangan sistem dengan menggunakan diagram konteks, diagram *use case*, dan rancangan arsitektur sistem, lalu kemudian berlanjut ke tahap konstruksi sistem, implementasi dan terakhir uji coba sistem. Proses pengujian sistem pada penelitian ini menggunakan metode *User Acceptance Test* (UAT) melalui penyebaran kuisioner dan memperoleh hasil yang menyebutkan bahwa aplikasi dapat mengatasi permasalahan yang ada pada sistem sebelumnya [12].

Penelitian ketiga berjudul “Model Sistem Informasi Pemesanan dan Pengelolaan Data pada Usaha Sablon ‘Al-Qaws’ Berbasis Web” dilakukan oleh Dinor dan rekannya pada tahun 2019. Penelitian ini membahas mengenai pembangunan sebuah sistem informasi berbasis web yang bukan hanya untuk

pengelolaan data namun juga sekaligus untuk pemesanan pada studi kasus usaha sablon “Al-Qaws”. Sistem informasi berbasis web pada penelitian ini dibangun dengan latar belakang belum adanya sistem informasi tersendiri untuk melakukan ekspansi penjualan yang lebih luas sekaligus agar dapat mempermudah toko dalam pengelolaan data dan pembuatan laporan. Tahapan penelitian pada penelitian ini belum mengacu pada suatu metode tertentu, hanya diawali dengan tahap pengumpulan data, lalu berlanjut ke tahap perancangan dengan membuat rancangan basis data, diagram konteks, diagram *use case*, serta rancangan arsitektur sistem, dan dilanjutkan dengan tahap konstruksi sistem. Di akhir penelitian ini dilakukan proses pengujian menggunakan *User Acceptance Test* sama seperti pada penelitian sebelumnya. Berdasarkan pengujian tersebut didapatkan hasil uji berupa sistem informasi berhasil dibangun untuk menjawab permasalahan dan tujuan penelitian yang telah didefinisikan [13].

Berlanjut ke penelitian keempat dengan judul “Pembangunan Sistem Informasi Penjualan Aksesoris-*Hardware* pada CV Bali *Source Technology Computer*” yang dilakukan Fitri beserta rekannya pada tahun 2019. Penelitian membahas seputar pembangunan sistem informasi penjualan aksesoris dan perangkat keras (*hardware*) pada sebuah CV bernama Bali *Source Technology Computer*. Pembangunan dilakukan dengan tujuan untuk dapat mempermudah pemrosesan dan pengelolaan data penjualan barang aksesoris dan perangkat keras pada toko tersebut, sehingga dapat memberikan efisiensi waktu dan meningkatkan kinerja karyawan dalam bekerja, serta dapat meminimalisir permasalahan yang timbul ketika menggunakan sistem sebelumnya yang masih manual. Metode pembangunan perangkat lunak sistem informasi pada penelitian ini mengacu pada metode *waterfall* yang diawali dengan tahap perencanaan dan pengumpulan data dengan teknik wawancara, observasi dan analisis. Kemudian berlanjut ke analisis dan desain sistem dengan memodelkannya menggunakan *flowchart*, serta tahap perancangan menggunakan *Data Flow Diagram* (DFD), *Entity Relationship Diagram* (ERD), dan juga kamus data. Setelah pembangunan sistem selesai, dilakukan proses pengujian dengan metode *black box testing* dengan responden pengujian pertama adalah sang peneliti selaku pengembang perangkat lunak serta

bersama pemilik dan pegawai CV *Bali Source Technology Computer* sebagai responden pada pengujian kedua. Kesimpulan yang didapat pada penelitian ini menyatakan pembangunan perangkat lunak sistem informasi yang dibuat berjalan sesuai dengan yang diharapkan dan mampu memberikan kemudahan pada perusahaan dalam proses pengelolaan data dan pembuatan laporan serta dapat memperlancar perusahaan dalam proses transaksi sehingga dapat menghasilkan manfaat dari segi efisiensi waktu [14].

Penelitian kelima dengan judul “Metode *Waterfall* pada Sistem Informasi Pengolahan Data Penjualan dan Pembelian Barang” dilakukan oleh Rohmat Rifai dan Mely Mailasari pada tahun 2020. Penelitian membahas mengenai pembangunan sistem informasi pengolahan data penjualan dan pembelian barang di toko “Fang Fifi Cell” yang terletak di Kota Bekasi. Sistem informasi dibangun untuk mengatasi permasalahan pengolahan data penjualan dan pembelian barang mulai dari data barang, transaksi, hingga pembuatan laporan yang masih sangat manual berupa tulis tangan sehingga menimbulkan sering terjadinya kesalahan pencatatan dan sulitnya proses pencarian data. Perangkat lunak sistem informasi dibangun dengan mengacu pada metode pembangunan perangkat lunak yang menggunakan model terstruktur yaitu *waterfall*. Tahapan pada metode ini diawali dengan pengumpulan data menggunakan teknik observasi, wawancara dan studi pustaka, lalu dilanjutkan proses perancangan basis data dengan ERD, *Logical Record Structure* (LRS), dan rancangan struktur tampilan sistem, kemudian tahap implementasi atau pembangunan sistem dengan bahasa pemrograman PHP dan basis data MySQL, serta diakhiri dengan tahap pengujian menggunakan metode *black box testing*. Penelitian menghasilkan kesimpulan bahwa sistem yang dibangun dapat mempermudah karyawan di toko Fang Fifi Cell dalam pengelolaan datanya sehingga dapat mengurangi kesalahan – kesalahan ketika masih menggunakan sistem konvensional. Sistem juga memberikan manfaat berupa berkurangnya duplikasi data dan meningkatnya fleksibilitas dan keamanan data karena dapat disimpan dalam jumlah yang lebih banyak dan terproteksi dengan *password* [15].

Penelitian terakhir adalah penelitian yang dilakukan Puspita dan rekannya pada tahun 2021, dengan judul “Sistem Persediaan untuk Mendukung Optimalisasi Bisnis Produk Parfume”. Penelitian ini mengangkat permasalahan mengenai sistem pengelolaan data barang pada perusahaan “*Trends Parfume*” yang masih menggunakan sistem konvensional atau manual, mulai dari pengolahan hingga penyajian datanya. Selain itu terdapat juga masalah mengenai ketidak-sinkronan data antara data pada gudang utama dengan gudang *client*. Berdasarkan permasalahan tersebut, pada penelitian ini dilakukan pembangunan sistem informasi inventory barang yang terkomputerisasi sehingga pengelolaan data menjadi lebih optimal dan dapat memberikan keluaran informasi yang bisa menjadi bahan pertimbangan dalam pengambilan keputusan. Sistem informasi tersebut dibangun berbasis web dengan teknologi bahasa pemrograman PHP dan basis data MySQL yang sudah terintegrasi dalam perangkat lunak XAMPP serta perangkat lunak penjelajah web Google Chrome. Metode pembangunan perangkat lunak yang digunakan pada penelitian ini adalah model *Waterfall*, sedangkan metode pengujian perangkat lunak yang diterapkan menggunakan model *User Acceptance Test*. Berdasarkan hasil pengujian menggunakan model tersebut didapatkan nilai kelayakan sistem sebesar 79,68% yang menunjukkan bahwa sistem yang dibangun berjalan dengan cukup baik dari segi fungsionalitasnya dan sudah sesuai dengan kebutuhan yang didefinisikan pengguna untuk dapat menjawab permasalahan yang ada [16].

Berdasarkan penjelasan penelitian – penelitian yang telah dijabarkan sebelumnya, dapat diringkas ke dalam tabel *State of The Art* dibawah ini:

Tabel 2.1. Tabel *State of The Art* tinjauan pustaka penelitian sebelumnya

| No. | Judul Penelitian  | <i>Comparing</i>   | <i>Contrasting</i>  | <i>Criticize</i>   | <i>Synthesize</i>  | <i>Summarize</i>  |
|-----|---|--|---|--|--|---|
| 1.  | Perancangan Sistem Inventory Barang Pada UD. Minang Dewi Berbasis Website | Membahas mengenai pembangunan sebuah perangkat lunak sistem informasi berbasis web untuk UD. Minang Dewi yang belum memiliki sistem informasi dengan basis data untuk dapat mengelola data barang di gudang. | Metode pembangunan perangkat lunak yang digunakan sebagai acuan dalam proses pembangunan perangkat lunak sistem informasi pada penelitian ini adalah model air terjun ( <i>waterfall</i> ). | Metode model <i>Waterfall</i> yang digunakan sudah terlalu berumur untuk diterapkan di zaman yang sudah modern dimana terdapat berbagai model metode yang lebih baru yang dapat digunakan. Sistem dapat dikembangkan lagi dari berbagai aspek untuk menghadapi pertumbuhan data yang semakin besar serta dapat dilakukan kegiatan pemeliharaan sistem yang rutin untuk menjaga keberlangsungan sistem. | Penelitian membahas tentang pembangunan sistem informasi berbasis web untuk pengelolaan data barang di gudang UD. Minang Dewi yang lebih baik. Pembangunan menggunakan metode <i>waterfall</i> , perancangan dengan UML dan konstruksi sistem dengan bahasa pemrograman PHP dan basis data MySQL. Hasil akhir penelitian menyatakan sistem informasi berhasil dibuat dan dapat menjawab permasalahan penelitian. | Hasil Sistem informasi berbasis web yang dibuat memberikan manfaat kecepatan, keakuratan dan kemudahan bagi karyawan khususnya bagian admin dalam pengelolaan data barang dibandingkan dengan sistem sebelumnya sehingga menghasilkan peningkatan kinerja organisasi perusahaan. Sistem informasi yang berbasis web menjadikan sistem mudah dipahami dan dipelajari pengguna. |

| No. | Judul Penelitian   | Comparing   | Contrasting  | Criticize   | Synthesize   | Summarize  |
|-----|--|---|--|---|--|--|
| 2.  | Aplikasi Pengelolaan Data Barang pada Bagian Perlengkapan (Studi Kasus Balai Veteriner Banjarbaru) | Penelitian dengan latar belakang pengelolaan data barang yang masih manual sehingga perlu dibuat sebuah aplikasi sistem informasi pengelolaan data barang agar dapat menghasilkan keluaran laporan data dengan lebih efektif dan efisien dibanding sistem lama. | Studi kasus penelitian bertempat di bagian perlengkapan UPT Dirjen Peternakan dan Kesehatan Hewan Balai Veteriner Banjarbaru. Metode Model SDLC klasik menjadi pilihan dalam proses pembangunan perangkat lunak pada penelitian ini. | Aplikasi sistem informasi yang telah dibangun dapat dikembangkan lebih lanjut lagi dari sisi desain dan penambahan fitur dengan menggunakan model metode pembangunan perangkat lunak yang lebih modern.               | Penelitian berupa pembangunan aplikasi sistem informasi dimulai dengan tahap analisis kebutuhan, lalu perancangan, konstruksi, implementasi dan tahap pengujian menggunakan metode UAT melalui penyebaran kuisioner.                           | Penelitian menghasilkan kesimpulan bahwa aplikasi sistem informasi pengelolaan data barang yang dibuat telah mampu menjawab permasalahan yang telah didefinisikan dengan berdasarkan analisa pada hasil kuisioner yang disebar menggunakan metode UAT. |
| 3.  | Model Sistem Informasi Pemesanan dan Pengelolaan Data pada Usaha Sablon 'Al-Qaws' Berbasis Web     | Belum adanya sistem informasi tersendiri, serta untuk mempermudah pengelolaan data dan pembuatan laporan menjadi latar belakang dan tujuan dalam pembangunan sebuah sistem informasi berbasis web pada penelitian yang berstudi kasus di usaha                  | Tahapan pembangunan perangkat lunak pada penelitian ini belum mengacu pada suatu model metode tertentu yang sistematis dan terstruktur, hanya diawali dengan tahap pengumpulan data, lalu tahap perancangan dan kemudian tahap       | Sistem informasi dapat dikembangkan lagi dengan mengacu pada suatu model metode pembangunan perangkat lunak tertentu yang sistematis dan terstruktur agar dapat memberikan hasil pengujian melalui kuisioner terhadap | Proses penelitian pembangunan sistem informasi dimulai dengan tahap pengumpulan data, lalu tahap perancangan dengan membuat rancangan basis data, diagram konteks dan <i>use case</i> serta rancangan arsitektur sistem, dan dilanjutkan tahap | Berdasarkan proses pembangunan sistem yang telah dilalui pada penelitian ini, ditarik kesimpulan bahwa sistem informasi yang dibangun berhasil menjawab tujuan penelitian yang didefinisikan walaupun masih terdapat beberapa kekurangan dari          |

| No. | Judul Penelitian  | Comparing  | Contrasting  | Criticize  | Synthesize  | Summarize  |
|-----|---|--|--|--|---|--|
|     |   | toko sablon “Al Qaws ini.  | konstruksi serta pengujian.  | pengguna yang lebih baik lagi.   | konstruksi serta pengujian dengan penyebaran kuisioner mengacu pada metode UAT.   | pendapat pengguna melalui kuisioner pengujian dengan metode UAT.   |
| 4.  | Pembangunan Sistem Informasi Penjualan Aksesoris-Hardware pada CV Bali<br><i>Source Technology Computer</i> | Penelitian mengangkat topik pembangunan sistem informasi penjualan dengan studi kasus pada toko aksesoris dan perangkat keras CV Bali<br><i>Source Technology Computer</i> yang bertujuan untuk mempermudah pemrosesan dan pengelolaan data agar dapat memberikan efisiensi dari segi waktu dan juga meminimalkan masalah yang timbul dibandingkan dengan sistem lama yang manual. | Proses pembangunan perangkat lunak sistem informasi pada penelitian ini mengacu pada metode model air terjun atau <i>waterfall</i> . | Model metode pembangunan perangkat lunak yang digunakan dalam penelitian ini masih menggunakan acuan model metode yang sudah cukup usang dan terkesan kaku yaitu metode air terjun atau <i>waterfall</i> . | Pembangunan sistem informasi dengan acuan metode <i>waterfall</i> pada penelitian ini diawali tahap pengumpulan data, lalu tahap analisis dan desain sistem dengan pemodelan bentuk <i>flowchart</i> , serta perancangan menggunakan DFD, ERD dan kamus data. Tahapan diakhiri dengan proses pengujian menggunakan metode <i>blackbox testing</i> oleh sang peneliti dan juga pemilik serta pegawai toko sebagai responden. | Penelitian berkesimpulan pembangunan perangkat lunak sistem informasi yang telah dilakukan berhasil dibuat sesuai dengan harapan dan mampu memberikan dampak positif bagi perusahaan studi kasus penelitian dalam pengelolaan datanya sehingga dapat meningkatkan kinerja karyawan dalam bekerja karena manfaat dari segi efisiensi waktu dibandingkan dengan sistem sebelumnya. |

| No. | Judul Penelitian   | Comparing   | Contrasting  | Criticize  | Synthesize   | Summarize   |
|-----|--|---|--|--|--|---|
| 5.  | Metode <i>Waterfall</i> pada Sistem Informasi Pengolahan Data Penjualan dan Pembelian Barang | Meneliti tentang pembangunan perangkat lunak sistem informasi pengolahan data penjualan dan pembelian barang di studi kasus sebuah toko bernama “Fang Fifi Cell”. Sistem lama yang sangat manual berupa tulis tangan penyebab sering terjadinya salah catat dan sulitnya pencarian data menjadi latar belakang permasalahan utama dibangunnya sistem informasi pada penelitian ini. | Model metode yang digunakan dalam proses pembangunan perangkat lunak sistem informasi pada penelitian ini adalah model yang terstruktur dan sistematis yaitu metode air terjun atau <i>waterfall</i> , dengan proses tahapan perancangan basis data menggunakan ERD dan LRS serta pendefinisian kode nama data yang lebih ditonjolkan pada penelitian ini. | Model metode pembangunan perangkat lunak <i>waterfall</i> yang digunakan dalam penelitian ini termasuk metode yang sudah cukup usang dan terkesan kaku untuk digunakan di zaman yang serba dinamis dan modern saat ini. Perangkat lunak sistem informasi yang dibangun juga dapat dilakukan pengembangan dan pembaruan lebih jauh lagi untuk dapat meningkatkan kualitas sistem serta memberikan kemudahan bagi pengguna dalam pemakaian sistem. | Dengan mengacu pada metode <i>waterfall</i> , pembangunan sistem pada penelitian ini diawali dengan tahap pengumpulan data, lalu berlanjut ke tahap perancangan mulai dari rancangan basis data dengan ERD, LRS dan pendefinisian kode nama data, serta rancangan struktur tampilan sistem, kemudian dilanjutkan tahap konstruksi dengan bahasa pemrograman PHP dan basis data MySQL, serta terakhir tahap pengujian menggunakan metode <i>black box testing</i> . | Sistem informasi yang dibuat pada penelitian ini memberikan kesimpulan penelitian bahwa dengan adanya sistem informasi yang dibangun tersebut berhasil menjawab permasalahan penelitian untuk dapat mempermudah karyawan toko dalam mengelola data sehingga dapat mengurangi berbagai kesalahan yang timbul saat menggunakan sistem lama yang sangat manual. Sistem yang dibangun juga dapat mengurangi duplikasi data, meningkatkan fleksibilitas jumlah penyimpanan data dan juga keamanan data dengan proteksi <i>password</i> . |

| No. | Judul Penelitian   | Comparing  | Contrasting   | Criticize  | Synthesize  | Summarize   |
|-----|--|--|---|--|---|---|
| 6.  | Sistem Persediaan untuk Mendukung Optimalisasi Bisnis Produk Parfume | Menyajikan penelitian dengan pembahasan mengenai pembangunan suatu sistem persediaan pada studi kasus perusahaan “Trends Parfume” berupa sebuah perangkat lunak jenis sistem informasi inventory barang berbasis web untuk mengatasi permasalahan sistem pengelolaan data barang yang masih manual mulai dari pengolahan hingga penyajian data agar dapat memberikan keluaran informasi yang lebih optimal dan juga data yang sinkron satu sama lain dengan bantuan komputerisasi. | Metode pembangunan perangkat lunak dengan jenis model sekuensial linier atau sering disebut juga sebagai model metode <i>waterfall</i> menjadi pilihan dalam proses pembangunan perangkat lunak sistem informasi pada penelitian ini. | Produk keluaran dari penelitian ini yaitu sebuah perangkat lunak sistem informasi inventory barang yang telah dibangun dapat dikembangkan lebih jauh lagi untuk memperbaiki kesalahan atau error di beberapa bagian fitur berdasarkan umpan balik yang telah diberikan oleh pengguna saat melakukan uji coba sistem, dengan menggunakan model metode pembangunan perangkat lunak yang lebih terbaru. | Sistem informasi inventory barang yang dibangun memakai metode <i>waterfall</i> pada penelitian ini terdiri atas tahapan awal berupa analisis dan kemudian tahap implementasi dan pengujian dari sistem informasi yang akan dibangun dimana pada penelitian ini teknologi yang digunakan meliputi model pengujian jenis UAT, lalu bahasa pemrograman PHP dan basis data MySQL yang terintegrasi dalam perangkat lunak XAMPP, serta perangkat lunak penjelajah web jenis Google Chrome karena sistem yang dibangun berbasis web. | Hasil akhir kesimpulan pada penelitian pembangunan perangkat lunak sistem informasi inventory barang ini menyebutkan bahwa sistem informasi yang dibangun telah berhasil dibuat sesuai dengan kebutuhan yang didefinisikan untuk menjawab permasalahan penelitian berdasarkan data hasil nilai kelayakan sistem menggunakan model pengujian UAT sebesar 79,68%. |

## **2.2. Landasan Teori**

Berikut adalah beberapa dasar teori yang penulis gunakan sebagai acuan literasi dalam pelaksanaan penelitian ini.

### **2.2.1. Digitalisasi**

Digitalisasi adalah proses peralihan bentuk media dari bentuk tercetak, audio, maupun video menjadi bentuk digital. Hal ini melibatkan pengubahan informasi menjadi format yang dapat dibaca oleh komputer, yang biasanya disusun dalam bentuk bit. Dalam konteks pengelolaan dokumen, digitalisasi melibatkan konversi dokumen tercetak menjadi dokumen elektronik atau digital yang dapat dibaca oleh komputer. Kemajuan teknologi dan ketersediaan data digital telah memungkinkan proses penciptaan, transfer, penyimpanan, dan analisis data digital yang lebih efisien. Dalam era digitalisasi, individu maupun organisasi dapat memanfaatkan berbagai fitur dan manfaat yang ditawarkan oleh digitalisasi seperti misalnya pembuatan arsip dokumen digital, kemudahan fotokopi digital, dan koleksi perpustakaan digital. Peralatan seperti komputer, scanner, dan perangkat lunak pendukung lainnya menjadi komponen penting dalam melaksanakan proses digitalisasi ini. Secara keseluruhan, digitalisasi memiliki potensi besar untuk membentuk dan mempengaruhi dunia kontemporer yang dijalani saat ini [5].

### **2.2.2. Sistem Informasi Berbasis Web**

Sistem informasi memiliki pengertian sebagai suatu kumpulan perangkat keras (*hardware*) dan perangkat lunak (*software*) yang didesain menjadi satu kesatuan dengan tujuan untuk mengubah data mentah menjadi suatu bentuk informasi yang memiliki nilai dan kegunaan, dimana kegunaan atau manfaat tersebut diantaranya untuk mendukung dalam pengambilan keputusan, dan untuk memenuhi berbagai kebutuhan pengguna lainnya [17], [18]. Sistem informasi melibatkan pengorganisasian cara-cara dalam mengumpulkan, memasukkan, mengolah, dan menyimpan data, serta mengelola, mengendalikan, dan melaporkan informasi dengan cara yang memungkinkan organisasi mencapai tujuan yang telah ditetapkan. Dalam lingkup perusahaan atau organisasi, sistem informasi terus

mengalami perkembangan. Jika pada masa lalu sistem informasi masih bersifat konvensional dengan pencatatan dan berkas-berkas fisik, kini semuanya telah mulai terkomputerisasi. Proses bisnis yang terjadi dalam perusahaan dibantu dan dilakukan menggunakan komputer. Kemajuan dalam sistem informasi mendorong setiap perusahaan dan organisasi untuk berlomba-lomba menciptakan sistem informasi yang kompleks dan memberikan manfaat yang signifikan bagi proses bisnis mereka [19].

Berdasarkan *literature review* yang dilakukan pada penelitian [19], platform yang dominan digunakan dalam pengembangan suatu sistem informasi adalah platform berbasis web. Web sendiri merupakan kependekan kata dari “*World Wide Web*”, yang memiliki pengertian sebagai sebuah jaringan global yang saling terhubung melalui jaringan internet. Web mencakup berbagai macam jenis informasi, mulai dari teks, gambar, suara, video dan lainnya [20]–[24]. Informasi – informasi dengan format penulisan dokumen HTML tersebut tersimpan dalam sebuah perangkat bernama server web [24], [25]. Proses pengiriman, penerimaan dan permintaan informasi web dengan pengguna dilakukan melalui sebuah protokol dalam jaringan internet yang disebut HTTP (*HyperText Transfer Protocol*) [23], [24], [26].

Berdasarkan dua pengertian tersebut, dapat disimpulkan pengertian dari sistem informasi berbasis web adalah sebuah program atau perangkat lunak (*software*) yang dioperasikan melalui perpaduan perangkat keras dan perangkat lunak yang disebut sebagai peladen web (*web server*), merespons permintaan konten laman web yang bersifat statis dan atau dinamis berupa berbagai jenis informasi dengan format dokumen HTML melalui sebuah format pertukaran data dalam jaringan internet yang disebut HTTP untuk dapat mendukung kegiatan atau proses bisnis dalam suatu organisasi atau perusahaan dalam mencapai suatu tujuan [27], [28]. Adapun proses interaksi pengguna dengan sistem informasi berbasis web terbagi atas tiga tahapan, yakni permintaan, pemrosesan, dan tanggapan [24], [29]. Dengan model pengoperasian perangkat lunak sistem informasi yang dilakukan melalui peladen web, ketika terdapat pembaruan pada sistem, semua pengguna

dapat langsung menikmatinya tanpa harus memperbarui secara manual sistem perangkat lunaknya [20], [30].

Perangkat lunak sistem informasi dengan basis web bersifat fleksibel, dapat diakses pengguna kapan saja dan dimana saja asalkan terdapat jaringan internet. Pengguna dapat mengakses sistem informasi melalui penjelajah web (*web browser*) pada berbagai perangkat seperti laptop, tablet, atau ponsel pintar (*smartphone*) dan tidak terbatas kepada suatu platform sehingga konten yang disajikan dapat lebih konsisten, hanya bergantung pada jenis penjelajah web (*web browser*) pada suatu platform yang digunakan untuk mengaksesnya [30]. Dengan berdasarkan kemudahan akses tersebut, penggunaan sistem informasi yang berbasis web juga memberikan manfaat untuk dapat mengurangi perangkat lunak yang diunduh (*di-download*) dan dipasang (*di-install*) pada suatu platform [31]. Selain itu, data – data yang diperoleh, diolah, diproses dan disimpan dalam sistem informasi yang berbasis web disimpan pada jarak jauh sehingga memungkinkan pengguna untuk mengakses data yang sama dari beberapa perangkat berbeda, daripada harus bertukar-kirim fail atau dokumen antar perangkat berbeda [30].

Teknologi yang umum digunakan dalam membangun sistem informasi berbasis web diantaranya adalah bahasa pemrograman PHP, Python, Javascript, dan masih banyak lainnya. Bahasa pemrograman atau bisa disebut juga sebagai sebuah *script* tersebut dapat terdiri atas beberapa fail untuk dapat membangun satu kesatuan perangkat lunak sistem informasi berbasis web yang biasanya bersifat dinamis dan terhubung dengan basis data atau konten dinamis lainnya [27].

#### **2.2.2.1. Bahasa Pemrograman Web PHP**

PHP adalah salah satu bahasa pemrograman dengan lisensi *Open-Source*, sehingga dapat digunakan dan didistribusikan secara bebas oleh pengguna dengan menyertakan atribusi-nya [32]. PHP diciptakan oleh seorang pemrogram Unix dan Perl dengan nama Rasmus Lerdoft [33]. Pada awal mulanya PHP hanya berupa kumpulan dari beberapa skrip (*script*) dan dibuat sebagai personal project, lalu disempurnakan oleh grup *six of developers*, kemudian seiring berjalannya waktu

ditambahkan fitur pemrograman berorientasi objek, dan selanjutnya dilahirkan kembali dengan nama PHP, akronim dari *PHP Hypertext Preprocessor* [34], [35].

Penggunaan dari bahasa pemrograman PHP ini sebagian besar untuk membangun atau membuat sebuah perangkat lunak yang berbasis web, khususnya sistem informasi berbasis web yang bersifat dinamis dengan performa yang cukup cepat [22], [36]. Hal ini dikarenakan PHP adalah sebuah bahasa pemrograman berbasis interpreter dengan sifat *server-side embedded script language*, yang artinya sintaks – sintaks PHP disisipkan ke dalam halaman web dengan format dokumen HTML, namun sintaks tersebut dieksekusi untuk diterjemahkan menjadi bahasa mesin di sisi peladen web (*web server*) secara langsung pada saat baris kode itu dijalankan, yaitu ketika terdapat permintaan dari pengguna melalui perangkat lunak penjelajah web dan kemudian hasilnya dikirimkan kembali kepada pengguna dalam bentuk format dokumen HTML yang ditampilkan melalui perangkat lunak penjelajah web juga [37], [38]. Adapun sintaks – sintaks dari PHP beberapa diantaranya memiliki kemiripan dengan bahasa pemrograman lain seperti bahasa pemrograman C, Java, Asc, dan Perl [39]. Diluar dari beberapa hal yang telah dijelaskan, bahasa pemrograman PHP memiliki beberapa kelebihan lain yang membuatnya menjadi salah satu bahasa pemrograman yang banyak dipakai khususnya dalam pembangunan perangkat lunak sistem informasi berbasis web, diantaranya adalah [40]:

1. PHP merupakan bahasa pemrograman yang sangat sederhana bagi seorang pemula, sehingga mudah untuk dibuat dan juga menawarkan banyak fitur lanjutan bagi seorang pemrogram profesional [30].
2. Kode sintaks PHP dieksekusi di sisi peladen web (*web server*) dan dikirimkan ke pengguna dalam bentuk format dokumen HTML, sehingga menghasilkan kecepatan akses yang tinggi dikarenakan pengguna tidak perlu memproses sintaks yang diperlukan, serta juga kode sintaks PHP menjadi aman karena hanya dari sisi pemrogram melalui peladen web (*web server*) saja yang dapat melihatnya [41].
3. PHP memiliki fitur yang dapat mempermudah melakukan koneksi ke berbagai jenis basis data (*database*) seperti MySQL, ORACLE,

Sysbase, dan lainnya sehingga memungkinkan untuk dapat mengelola, memproses dan menyajikan konten web yang dinamis dengan lebih mudah[34].

4. PHP dapat berjalan pada platform atau sistem operasi perangkat yang berbeda – beda seperti UNIX, Windows XP, Windows 97, Windows NT dan Macintosh.
5. PHP juga dapat bekerja dengan baik di banyak perangkat lunak peladen web (*web server*) seperti Microsoft Personal Web Server, Apache, Xitami, dan sebagainya [21].

#### **2.2.2.2. Kerangka Kerja Pemrograman Web berbasis PHP CodeIgniter**

Kerangka kerja atau sering disebut juga sebagai *Framework* memiliki pengertian sebagai sebagai suatu kumpulan dari beberapa pustaka (Kelas) yang memiliki sifat inheritance atau bisa diturunkan atau diwariskan, atau bisa juga langsung dipakai dalam pengembangan suatu aplikasi atau sistem perangkat lunak. Penggunaan kerangka kerja memiliki beberapa kelebihan daripada menggunakan metode pembangunan yang konvensional, yaitu [39]:

1. Mempermudah dalam memahami mekanisme kerja dari sebuah perangkat lunak, sehingga dapat mempercepat pembangunan perangkat lunak, baik secara mandiri maupun tim.
2. Memangkas waktu pengerjaan sehingga menjadi lebih efektif dan efisien, karena dengan menggunakan kerangka kerja, para pembangun perangkat lunak sudah memiliki sebuah patokan dalam menyelesaikan suatu modul sehingga tidak perlu membuat ulang Kelas (*Class*) untuk suatu kasus yang relatif sama [39].

Dalam dunia pembangunan perangkat lunak, permintaan untuk pembangunan sebuah perangkat lunak berbasis web semakin meningkat sehingga menghasilkan permintaan yang tinggi akan efisiensi, keandalan, pemeliharaan dan skalabilitas. Hal tersebut menunjukkan perlunya penggunaan suatu kerangka kerja dalam membangun perangkat lunak. Untuk kasus pembangunan sebuah perangkat lunak berbasis web khususnya sistem informasi berbasis web, salah satu teknologi

bahasa pemrograman yang populer dan banyak digunakan adalah bahasa pemrograman PHP, sehingga suatu kerangka kerja dengan bahasa pemrograman PHP menjadi salah satu alat yang cukup penting dalam pembangunan perangkat lunak sistem informasi berbasis web [40].

Kerangka kerja PHP dapat mempercepat proses pembangunan sistem informasi berbasis web namun dengan tetap memperhatikan standar penulisan kode yang baik, hal ini dikarenakan di dalam suatu kerangka kerja PHP sudah terdapat berbagai set API (*Application Programming Interfaces*), pustaka (*libraries*), dan juga berbagai ekstensi (*extensions*) yang dapat membantu pembangun (*developer*) sistem perangkat lunak menjadi lebih produktif dengan mengurangi kesalahan pengulangan kode dalam proses pembangunan sistem. Selain itu hasil sistem informasi berbasis web yang dibuat dengan kerangka kerja PHP menjadi lebih stabil, dan aman karena keandalan dan kemudahan pemeliharannya [40].

Model kerangka kerja PHP berbasiskan konsep *Model*, *View*, dan *Controller* atau sering disebut juga konsep MVC. Konsep model tersebut merupakan model yang efektif dan terbukti untuk menghasilkan perangkat lunak yang modular dan terorganisasi dengan baik. Konsep model MVC membagi perangkat lunak menjadi beberapa lapisan yang dapat dianalisis dan diimplementasikan secara terpisah, sehingga dapat mengurangi kompleksitas rancangan arsitektur perangkat lunak dan meningkatkan fleksibilitas dan pemakaian ulang kode [40].

Hingga saat ini, terdapat berbagai macam kerangka kerja PHP, salah satunya adalah kerangka kerja (*framework*) CodeIgniter. CodeIgniter didirikan pertama kali pada 28 Februari 2006 oleh Rick Ellis, pendiri dan CEO dari *EllisLab.com*, perusahaan dibalik pengembangan CodeIgniter di masa awal. CodeIgniter kemudian dikembangkan lebih lanjut oleh berbagai komunitas dan disebarkan ke seluruh dunia dengan lisensi bebas dan bersifat *open source* [42]. CodeIgniter memiliki logo api menyala yang menandakan CodeIgniter sebagai salah satu kerangka kerja bahasa pemrograman PHP dengan cepat “membakar” semangat para pengembang web (*web developer*) untuk membangun dan mengembangkan web khususnya yang bersifat dinamis dengan cepat dan mudah.

Sama halnya seperti berbagai kerangka kerja bahasa pemrograman web PHP lainnya, CodeIgniter juga menerapkan konsep MVC (*Model, View, Controller*) dalam proses pembangunan perangkat lunak berbasis webnya, sehingga dapat mempercepat dan mempermudah pengembang (*Developer*) ketika membangun suatu perangkat lunak berbasis web. Disamping keunggulannya dalam hal ringan dan cepat, CodeIgniter juga menawarkan dokumentasi yang sangat lengkap beserta contoh implementasi kodenya. Keberadaan dokumentasi yang komprehensif ini menjadi salah satu alasan kuat mengapa banyak orang memilih kerangka kerja CodeIgniter. Dengan segala kelebihan yang dimilikinya, CodeIgniter mendapat pujian dari sang pencipta bahasa pemrograman PHP, Rasmus Lerdorf di acara *frOSCon* pada Agustus 2008 dengan menyatakan alasan ia menyukai CodeIgniter adalah karena “*it is faster, lighter and the least like a framework*”. Selain itu terdapat berbagai kelebihan lain yang menjadikan CodeIgniter cukup populer di kalangan pengembang perangkat lunak berbasis web yang dituangkan dalam bentuk poin sebagai berikut [43]:

1. Ukurannya yang ringkas. Fail unduhan CodeIgniter hanya sekitar 2MB, termasuk dokumentasi yang sangat lengkap didalamnya.
2. Kompatibilitas dengan berbagai layanan *hosting*. CodeIgniter dapat berjalan dengan baik pada hampir semua platform *hosting*. Selain itu, CodeIgniter juga mendukung berbagai jenis basis data (*database*) yang umum digunakan seperti misalnya MySQL.
3. Tidak adanya aturan pengkodean (*coding*) yang ketat. Pengguna bebas memilih apakah ingin menggunakan *Controller* tanpa *View*, atau bahkan tidak menggunakan *Model*. Meskipun demikian, disarankan untuk menggunakan ketiga komponen tersebut untuk kebijakan yang lebih bijaksana.
4. Performa yang baik. CodeIgniter memiliki kecepatan yang tinggi dan bahkan dapat dikatakan sebagai salah satu kerangka kerja paling cepat yang ada saat ini.
5. Kemudahan dalam mengintegrasikan dengan pustaka (*library*) eksternal. CodeIgniter telah menyediakan dukungan yang baik untuk

pengembangan dengan berbagai pustaka yang tersedia saat ini. Hal ini mempermudah integrasi dengan pustaka - pustaka yang ingin digunakan dalam pembangunan perangkat lunak.

### 2.2.2.3. Basis Data Relasional MySQL

Basis data (*Database*) terdiri dari dua kata, basis dan data yang dapat didefinisikan sebagai himpunan kelompok data dan deskripsinya yang saling berhubungan secara logis dan diorganisasi sedemikian rupa agar dapat dimanfaatkan kembali bersama – sama untuk menjawab permintaan kebutuhan atas informasi data tersebut dengan cepat dan mudah [29], [44]. Tabel, baris, dan kolom adalah beberapa istilah yang dapat dijumpai dalam terminologi basis data. Tabel sendiri merupakan bentuk dua dimensi yang berisikan suatu kelompok data dengan jenis yang sama. Satu buah tabel dapat berisikan sejumlah kolom yang dapat disebut sebagai *field* dan juga sejumlah baris yang bisa disebut juga dengan *record* atau *tuple* [34]. Tujuan dari adanya basis data pada umumnya adalah untuk memberikan kemudahan dan kecepatan pada saat proses manipulasi data yang tersimpan seperti penambahan, perubahan, penghapusan, pengambilan dan pencarian data. Selain itu, salah satu keunggulan dari penggunaan basis data adalah dapat mengelola data berdasarkan fungsi dan jenisnya secara tepat dan teratur [33], [45].

Adapun salah satu model basis data yang banyak digunakan untuk menyimpan data berjumlah besar dan saling berhubungan adalah dengan menggunakan model basis data relasional. Model basis data relasional merupakan suatu cara untuk merepresentasikan model data dalam perancangan basis data dimana model dari basis data relasional didasarkan pada *record* [45]. Dalam model ini, semua data tersimpan dalam bentuk tabel – tabel yang saling berhubungan, dimana hubungan antar tabel tersebut diwakili dengan menggunakan nilai kunci yang saling terkait antar tabel. Nilai kunci pada setiap tabel tersebut adalah kunci primer (*primary key*), yang memiliki pengertian sebagai suatu nilai dengan sifat unik (tidak ada nilai yang sama) yang membedakan satu baris dengan baris lainnya dalam sebuah tabel [34].

Salah satu bahasa relasional dalam dunia basis data adalah SQL (*Structured Query Language*). SQL bukan sebuah perangkat lunak atau aplikasi basis data, tetapi lebih kepada suatu bahasa yang digunakan untuk mengajukan kueri kedalam basis data seperti misalnya kueri membuat atau memasukkan, memilih, menampilkan, mengubah, dan menghapus data. Sistem manajemen basis data relasional (*Relational Database Management System / RDBMS*) yang berbasis SQL diantaranya adalah PostgreSQL dan MySQL. MySQL menggunakan model koneksi dengan protokol TCP-IP, membuat akses ke peladen basis data (*database server*) lebih cepat jika dibandingkan dengan menggunakan *mapping drive* [34].

MySQL merupakan salah satu RDBMS yang banyak diminati oleh para pengembang perangkat lunak, baik itu perangkat lunak desktop maupun perangkat lunak berbasis web untuk menyimpan, mengatur, dan mengelola data pada perangkat lunak tersebut [46]. MySQL dapat diakses secara gratis dan *Open Source*, dimana pada awalnya hanya diperuntukkan untuk sistem operasi berbasis Linux atau Unix, namun seiring dengan berkembangnya zaman kini bisa juga digunakan pada berbagai macam sistem operasi [33]. MySQL mulai populer sejak tahun 1990 di waktu perangkat lunak berbasis web mulai banyak digunakan. Salah satu dari beberapa faktor yang menjadikan MySQL banyak digunakan adalah dikarenakan tidak adanya alternatif RDBMS lain yang simpel, cepat, stabil, dan juga memiliki banyak fitur. Dalam hal penginstalan dan penggunaan, MySQL terbukti mudah dan juga andal serta dapat dihubungkan dengan berbagai jenis bahasa pemrograman [30], [46]. Selain itu, dalam hal penyimpanan data MySQL tergolong dalam basis data yang fleksibel yaitu dapat menyimpan data dengan jumlah yang banyak maupun sedikit. [41].

### **2.2.3. Pemodelan Sistem Perangkat Lunak**

Pemodelan sendiri memiliki pengertian sebagai sebuah proses perancangan suatu sistem sebelum dilakukannya proses pembuatan dari sistem itu sendiri. Tujuan dari dilakukannya pemodelan adalah agar dapat memahami sistem yang akan dibangun secara menyeluruh sehingga dapat menghasilkan sistem yang

berhasil menjawab kebutuhan – kebutuhan dari pengguna yang ditargetkan, dan semakin kompleks suatu sistem yang dibangun maka semakin penting pula adanya sebuah pemodelan [47].

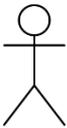
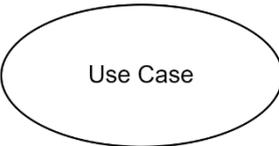
### **2.2.3.1. Pemodelan Fungsionalitas *Use Case Diagram***

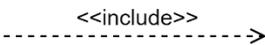
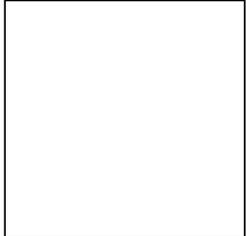
Diagram *use case* adalah salah satu dari diagram UML (*Unified Modelling Diagram*) yang merupakan sebuah metode pemodelan secara visual yang dapat digunakan untuk merancang dan atau membuat suatu sistem perangkat lunak [48]. Diagram *use case* termasuk dalam jenis diagram perilaku (*Behavior Diagram*) yang didalamnya disajikan interaksi – interaksi antara aktor dan sistem melalui *use case* yang terhubung atau ber-relasi dengan sang aktor [35], [49], [50]. Aktor dan *use case* menjadi 2 unsur inti dalam suatu diagram *use case*, serta ditambah dengan relasi yang menghubungkan antara keduanya bersama – sama menjadi satu kesatuan diagram *use case* [50], [51]. Hal ini menjadikan diagram *use case* dapat disebut juga sebagai suatu gambaran visual yang berisikan himpunan aktor dan *use case* yang saling terlibat dalam suatu sistem [52]. *Use case* sendiri merupakan kebutuhan – kebutuhan yang harus ada pada suatu sistem sesuai dengan kebutuhan pengguna, sehingga melalui diagram *use case* dapat digambarkan secara visual dengan cukup luas akan kebutuhan – kebutuhan dari suatu sistem yang dibutuhkan oleh pengguna. Kebutuhan tersebut dapat berupa suatu fungsionalitas atau fungsi – fungsi yang harus ada pada suatu sistem untuk menyelesaikan kebutuhan dari sang pengguna [36], [53]. Namun dalam hal ini, yang harus diperhatikan adalah “apa” yang dilakukan oleh sistem, bukan “bagaimana” sistem melakukannya [47], [48].

Diagram *use case* ini sangat berguna bagi seorang pengembang perangkat lunak untuk berkomunikasi dengan para pengguna agar dapat ikut berpartisipasi dalam memodelkan kebutuhan yang dibutuhkan pada sistem perangkat lunak yang akan dibuat [47], [54]. Diagram *use case* juga dapat digunakan bersama pengguna untuk merancang sebuah *test case* terhadap keseluruhan fungsi yang ada dalam sistem yang akan dibuat [47].

Notasi – notasi yang umum dan sering digunakan pada suatu diagram *use case* beserta penjelasan lebih lengkapnya dapat dilihat pada tabel 2.2 berikut ini.

Tabel 2.2. Notasi Diagram *Use Case*

| Notasi  | Nama   | Penjelasan  |
|---|--|---|
|  <p data-bbox="427 533 494 560">Actor</p>        | Aktor  | <p data-bbox="959 383 1356 750">Merupakan pengguna dari sistem yang berinteraksi dengan sistem. Dapat disebut juga sebagai apapun yang terdapat diluar lingkup sistem, namun berinteraksi dengan sistem tersebut memicu sebuah aksi dan menghasilkan reaksi atau respon. Satu aktor dapat berasosiasi atau berhubungan dengan banyak <i>use case</i>.</p> <p data-bbox="959 757 1197 788">Aktor dapat berupa:</p> <ul data-bbox="959 795 1356 1124" style="list-style-type: none"> <li>- Pengguna manusia</li> <li>- Waktu pemicu interaksi atau proses pada sistem</li> <li>- Sistem eksternal yang berinteraksi ketika pembangunan sistem, seperti layanan basis data eksternal atau layanan lainnya.</li> <li>- Perangkat eksternal, seperti printer, ponsel.</li> </ul> |
|  <p data-bbox="406 1209 510 1236">Use Case</p> | <i>Use Case</i>  | Sebuah interaksi yang dilakukan atau dijalankan antara aktor dan sistem. Merepresentasikan apa yang sistem lakukan berupa sebuah fitur atau fungsionalitas sebagai respon atas interaksi dengan aktor, bukan bagaimana sistem melakukannya.   |
|    | Asosiasi ( <i>Association</i> ) aktor dengan <i>use case</i>   | Menunjukkan relasi atau hubungan antara aktor dengan <i>use case</i> . Merepresentasikan komunikasi atau interaksi antara aktor dengan sistem melalui <i>use case</i> yang terhubung.   |
|    | <p data-bbox="670 1624 893 1758">Relasi aktor dengan aktor atau <i>use case</i> dengan <i>use case</i></p> <p data-bbox="678 1765 885 1832">Generalisasi (<i>Generalization</i>)</p> | <p data-bbox="959 1624 1356 1792">Menunjukkan relasi kepada <i>use case</i> yang lebih umum dalam relasi <i>use case</i> dengan <i>use case</i>. Arah panah menunjuk ke <i>use case</i> yang lebih umum.</p> <p data-bbox="959 1798 1356 1948">Menunjukkan relasi kepada aktor yang lebih umum dalam relasi aktor dengan aktor. Arah panah menunjuk ke aktor yang lebih umum.</p>   |

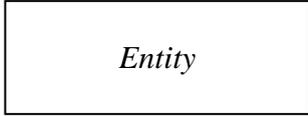
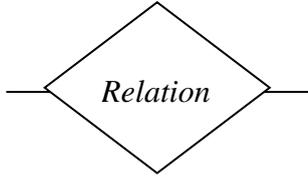
| Notasi   | Nama  | Penjelasan  |
|--|---|---|
|   | Relasi<br><i>use case</i> dengan <i>use case</i><br>Inklusi ( <i>Inclusion</i> )  | Menunjukkan relasi pengikutsertaan suatu <i>use case</i> ketika <i>use case</i> asal dijalankan. Arah panah menunjuk ke <i>use case</i> yang diikuti-sertakan. Satu <i>use case</i> dapat diikuti-sertakan oleh lebih dari satu <i>use case</i> untuk menghindari duplikasi fungsionalitas. |
|   | Relasi<br><i>use case</i> dengan <i>use case</i><br>Ekstensi ( <i>Extension</i> ) | Menunjukkan relasi penjabaran khusus dari suatu <i>use case</i> , dapat berupa fungsionalitas, fitur atau opsi tambahan yang lebih spesifik. Arah panah menunjuk ke <i>use case</i> asal yang dijabarkan.   |
|  | Batasan Sistem<br>( <i>System Boundary</i> )                                      | Menunjukkan batasan sistem, memisahkan antara aktor dan <i>use case</i> . Aktor berada diluar batasan sistem, sedangkan <i>use case</i> berada di dalam sistem.   |

Sumber: B. Unhelkar, 2018

### 2.2.3.2. Pemodelan Basis Data *Entity Relationship Diagram* (ERD)

*Entity Relationship Diagram* (ERD) adalah salah satu jenis konsep pemodelan data yang umum digunakan untuk merancang suatu basis data (*database*) [55]. Rancangan berupa hubungan antara data yang ada digambarkan melalui sebuah diagram yang terdiri atas beberapa notasi grafis. Dengan rancangan tersebut, ERD dapat berfungsi sebagai alat bantu bagi para pengembang (*developer*) perangkat lunak dalam pembuatan suatu basis data untuk memberikan gambaran bagaimana suatu basis data akan beroperasi nantinya. Komponen utama dalam suatu ERD terdiri atas himpunan tiga elemen inti, yakni meliputi himpunan entitas, himpunan atribut, dan juga himpunan relasi antar entitas [55], [56]. Penjelasan lebih lanjut tentang tiga elemen inti beserta notasi grafisnya dapat dilihat pada tabel 2.3. berikut:

Tabel 2.3. Notasi Diagram Relasi Entitas (*Entity Relationship Diagram, ERD*)

| Notasi   | Nama                         | Penjelasan   |
|--|------------------------------|--|
|   | Entitas ( <i>Entity</i> )    | Merupakan objek yang menjadi fokus inti dalam suatu basis data. Beberapa yang dapat disebut sebagai entitas meliputi manusia, tempat, benda atau keadaan yang terkait dengan data yang diperlukan.   |
|   | Atribut ( <i>Attribute</i> ) | Merupakan data yang terkait dengan entitas. Setiap entitas harus memiliki atribut yang menjadi kunci utama ( <i>primary key</i> ) sebagai identitas uniknya, ditambah dengan atribut deskriptif lainnya. Atribut dapat disimpan dalam tabel entitas atau dapat dipisahkan. |
|  | Relasi ( <i>Relation</i> )   | Merupakan simbol yang menggambarkan hubungan antara dua atau lebih entitas.  |

Sumber: A. Khoulah dkk, 2022

Untuk elemen relasi yang terdapat dalam suatu ERD, terdapat beberapa macam jenis relasi yang dapat diterapkan dalam suatu diagram relasi entitas (ERD).

Berikut adalah beberapa diantaranya [56]:

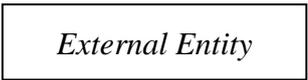
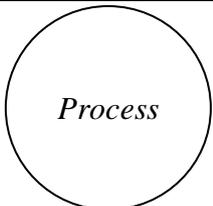
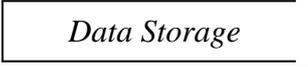
1. Relasi satu ke satu (*One to One*), satu anggota entitas dapat berelasi dengan satu anggota entitas lain.
2. Relasi satu ke beberapa (*One to Many*), Satu anggota entitas dapat berelasi dengan beberapa anggota entitas lain.
3. Relasi beberapa ke satu (*Many to One*), beberapa anggota entitas dapat berelasi dengan satu anggota entitas lain.
4. Relasi beberapa ke beberapa (*Many to Many*), Beberapa anggota entitas dapat berelasi dengan beberapa anggota entitas lain.

Menurut penelitian [56], suatu desain ERD dapat dikatakan baik dan benar apabila yang pertama, semua entitas pada basis data saling terhubung oleh relasi, lalu tiap entitas memiliki atribut, dan terakhir tiap entitas memiliki kunci utama (*primary key*) dan atribut deskriptif.

### 2.2.3.3. Pemodelan Aliran Data Sistem *Data Flow Diagram* (DFD)

*Data Flow Diagram* (DFD) adalah salah satu alat pemodelan yang biasa digunakan untuk menggambarkan interaksi suatu sistem dengan lingkungannya, khususnya dalam hal aliran data yang masuk dan keluar dari sistem tersebut ketika interaksi terjadi. Dengan menggunakan DFD juga bisa didapatkan hasil analisis kebutuhan pengguna dengan fokus pada struktur dan proses kerja sistem untuk mengembangkan sistem yang akan dibangun. DFD memiliki beberapa model yang biasa digunakan dalam proses pembangunan suatu sistem, beberapa diantaranya adalah model *Gane Sarson*, *Yourdon De Marco*, *Yourdon and Coad*, dan *Structured System Analysis and Design Methodology* (SSADM). Elemen – elemen penting yang terdapat dalam suatu diagram alir data atau DFD diantaranya adalah entitas eksternal (*external entity*), proses (*process*), aliran data (*data flow*), dan penyimpanan data (*data storage*). Notasi - notasi dari elemen - elemen penting DFD tersebut dapat dilihat pada tabel 2.4. di bawah ini [55].

Tabel 2.4. Notasi Diagram Alir Data (*Data Flow Diagram*, DFD)

| Notasi  | Nama   | Penjelasan  |
|---|--|---|
|  | Entitas Eksternal ( <i>External Entity</i> ) | Menunjukkan entitas / objek diluar sistem yang berinteraksi dengan sistem |
|  | Proses ( <i>Process</i> )                    | Menunjukkan proses yang terjadi pada sistem                               |
|  | Penyimpanan Data ( <i>Data Storage</i> )     | Menunjukkan objek tempat penyimpanan data pada sistem                     |
|  | Alir Data ( <i>Data Flow</i> )               | Menunjukkan arah aliran data yang terjadi pada sistem                     |

Sumber: R. Rizki dkk, 2022

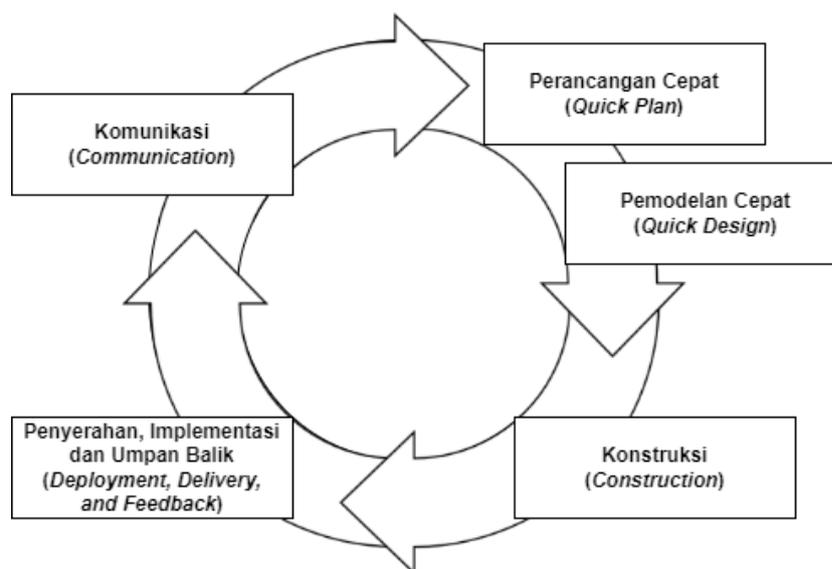
### 2.2.4. Metode Pembangunan Perangkat Lunak Prototipe

Metode prototipe adalah salah satu metode untuk membangun suatu sistem perangkat lunak dengan model pendekatan yang cepat dan bertahap agar dapat selalu berkomunikasi dengan pengguna untuk mendapatkan umpan balik atas proses pembangunan sistem yang sedang berjalan [10]. Metode ini merupakan

suatu evolusi dalam dunia pembangunan sistem perangkat lunak karena telah merevolusi metode pembangunan terdahulunya yang menggunakan sistem sekuensial atau sering dikenal dengan metode air terjun (*waterfall*) [57]. Salah satu proses yang ada dalam metode prototipe ini adalah *prototyping* yang memiliki pengertian sebagai proses pembuatan sebuah prototipe dari sistem perangkat lunak yang akan dibangun secara cepat untuk segera memperoleh umpan balik dari pengguna agar prototipe dapat segera diperbaiki dan disempurnakan berdasarkan umpan balik kebutuhan pengguna secara cepat juga, hal tersebut menjadi inti pemikiran dalam metode ini [58], [59]. Proses ini berjalan terus menerus secara iteratif hingga tercapai kesepahaman antara pengguna dengan sistem yang akan dibangun terhadap kebutuhan - kebutuhan yang diperlukan [23], [57]. Sebuah prototipe yang dibuat dalam metode prototipe ini akan memberikan gambaran atau pemodelan secara struktur, fungsional dan juga operasional dari sistem perangkat lunak yang akan dibangun. Dengan menggunakan prototipe yang dibuat dalam metode ini akan dapat memudahkan para pengembang sistem perangkat lunak dalam membangun suatu sistem perangkat lunak secara cepat dan tepat memenuhi kebutuhan pengguna, karena adanya informasi yang terus terkumpul dari pengguna melalui proses interaksi menggunakan prototipe tersebut [59], [60].

Metode prototipe ini dapat diaplikasikan dalam pembangunan sistem perangkat lunak dengan skala kecil hingga skala besar dengan tujuan untuk membangun sebuah sistem dengan lancar, terorganisir serta dapat selesai sesuai dengan tenggat waktu yang ditetapkan [59]. Metode prototipe juga sangat tepat untuk pembangunan sistem perangkat lunak yang sifatnya kustomisasi, yaitu seperti sistem yang dibangun berdasarkan keinginan, kebutuhan atau kondisi tertentu dan atau bisa juga untuk pembangunan suatu sistem dengan implementasi suatu metode atau algoritma tertentu didalamnya [9]. Peran serta pengguna sistem perangkat lunak yang akan dibangun secara konsisten menjadi kunci dalam metode ini agar dapat menguntungkan berbagai pihak yang terlibat mulai dari para pimpinan, para pengguna itu sendiri dan juga para pengembang sistem perangkat lunak [59]. Selain itu, terdapat satu hal penting lagi yang juga menjadi kunci keberhasilan dalam penggunaan metode ini, yaitu untuk dapat menetapkan kesepahaman di awal antara

pengembang dan pengguna mengenai tujuan dibangunnya sebuah prototipe yang ditujukan untuk dapat menggali kebutuhan – kebutuhan yang harus ada pada sistem yang akan dibangun [58]. Kemudian untuk tahapan - tahapan yang ada di dalam metode prototipe ini terdapat 5 tahapan seperti yang dapat dilihat pada gambar dibawah, dengan penjelasan lebih lengkapnya dapat dilihat pada poin penjelasan dibawahnya.



Sumber: M. Prabowo, 2020

Gambar 2.1. Tahapan Metode Prototipe

#### 1. Komunikasi (*Communication*)

Tahap pertama dalam metode prototipe adalah tahap komunikasi yang merupakan langkah awal dalam metode ini sebelum masuk ke tahapan yang lebih bersifat teknis [59]. Sesuai dengan namanya, pada tahap ini para pengembang sistem melakukan komunikasi dengan pengguna atau pemangku kepentingan terkait mengenai kebutuhan dan persyaratan apa saja yang diperlukan dalam sistem yang akan dibangun seperti layanan – layanan (*services*), data – data, ataupun spesifikasi – spesifikasi [61], [62]. Dalam tahap ini juga dilakukan identifikasi permasalahan – permasalahan yang akan diselesaikan melalui sistem yang dibuat [41]. Dalam pengumpulan data mengenai kebutuhan dan berbagai hal yang diperlukan pada tahap ini dapat menggunakan beberapa teknik pengumpulan data seperti diantaranya adalah observasi terhadap sistem yang sebelumnya atau yang tengah berjalan, wawancara secara langsung bersama pengguna dan pemangku

kepentingan terkait, dan atau studi literatur terhadap beberapa teori yang akan digunakan dalam proses pelaksanaan metode ini [60], [61].

## 2. Perancangan Cepat (*Quick Plan*)

Tahap kedua dari metode pembangunan prototipe, berdasarkan hasil persyaratan dan kebutuhan yang didapat pada tahap sebelumnya, pada tahap ini dilakukan perancangan awal terhadap sistem yang akan dibangun berupa sebuah rancangan awal antarmuka atau tampilan (*Interface*) sesuai dengan kebutuhan yang telah didefinisikan [60]. Rancangan awal antarmuka sistem yang dibuat hanya poin pentingnya saja, menitikberatkan kepada hal – hal pada sistem yang berinteraksi atau terlihat oleh pengguna saja [59]. Hal ini dikarenakan inti pemikiran metode prototipe yang dilakukan secara cepat sehingga pembuatan rancangan awal ini juga dilakukan secara cepat [23], [62]. Rancangan awal antarmuka sistem dengan berisikan poin – poin penting saja ini dapat disebut juga sebagai sebuah *wireframe* yang pembuatannya dapat menggunakan sebuah perangkat lunak atau aplikasi seperti contohnya balsamiq *wireframes* [63].

## 3. Pemodelan Cepat (*Quick Design*)

Berlanjut ke tahap ketiga dari metode prototipe yakni tahap pemodelan cepat, sama seperti tahap lainnya pada tahap ini juga menggunakan hasil data yang didapat dari tahap - tahap sebelumnya seperti masalah – masalah yang teridentifikasi, kebutuhan – kebutuhan dan atau persyaratan yang diperlukan serta informasi lain yang dapat berguna dalam membuat pemodelan yang menjelaskan rancangan dari sistem yang akan dibangun. Pemodelan atau perancangan pada tahap ini dibuat dengan cukup detail dengan patokan utama memenuhi seluruh kebutuhan pengguna, mulai dari awal data atau masukan yang dimasukkan pengguna, kemudian proses yang berjalan pada sistem hingga keluaran yang dihasilkan oleh sistem [59], [60]. Adapun untuk alat yang digunakan dalam pemodelan pada tahap ini dapat menggunakan berbagai macam jenis alat pemodelan, seperti misalnya untuk pemodelan fungsionalitas dengan menggunakan diagram *use case* dari diagram UML, lalu untuk pemodelan basis data menggunakan diagram relasi

entitas (*Entity Relationship Diagram*, ERD), serta diagram aliran data (*Data Flow Diagram*, DFD) untuk memodelkan aliran data sistem [36], [53], [55].

#### 4. Konstruksi (*Construction*)

Tahap selanjutnya adalah tahap konstruksi. Pada tahap ini, rancangan awal antarmuka sistem yang telah dibuat pada tahap sebelumnya disempurnakan lebih lanjut lagi menjadi sebuah prototipe sistem dengan tetap mengacu pada poin penting yang akan terlihat dari sisi user agar sesuai dengan inti pemikiran metode prototipe ini yaitu untuk melakukannya dengan cepat [62]. Perbedaan prototipe sistem yang dihasilkan pada tahap ini dibandingkan dengan rancangan awal antarmuka sistem yang berupa *wireframe* adalah pada tahap ini sudah dibuat hampir mendekati tampilan yang ada pada sistem nantinya namun dengan desain dan warna yang masih simpel dan hitam putih [63]. Selain itu, prototipe sistem juga dibuat agar dapat lebih menggambarkan aksi dan reaksi pada saat interaksi bersama pengguna ketika proses evaluasi prototipe, yakni dengan menambahkan fitur aksi dan reaksi pada setiap fungsi yang ada dalam *wireframe* yang telah dibuat sebelumnya. Tipe prototipe sistem yang dibuat tersebut dapat disebut juga sebagai sebuah prototipe dengan *fidelity* rendah (*Low-Fidelity Prototype*, *Lo-Fi Prototype*), yang dibuat untuk tujuan mendapatkan evaluasi lebih mendalam dari pengguna seperti mencari kekurangan dan kelebihan, penambahan atau penghapusan sesuatu, serta komentar dan juga saran [23], [64].

Kemudian setelah prototipe sistem atau *Lo-Fi Prototype* yang dibuat telah mencapai iterasi akhir yang ditandai dengan adanya kesepakatan bersama pengguna dan sudah tidak terdapat lagi perubahan, komentar ataupun saran yang perlu dilakukan terhadap prototipe sistem tersebut, maka akan dilakukan proses pembuatan sistem perangkat lunak atau bisa disebut juga dengan proses pengkodean (*Coding*). Proses pengkodean dilakukan sama seperti tahap – tahap lainnya dengan mengacu pada informasi – informasi, hasil rancangan, data dan kebutuhan yang telah didefinisikan pada tahap sebelumnya. Setelah proses pengkodean selesai dilakukan dan menghasilkan sebuah sistem perangkat lunak yang lengkap dan fungsional, selanjutnya segera dilakukan pengujian agar dapat

mengurangi kesalahan – kesalahan yang dapat terjadi dalam proses pengkodean [41], [59].

5. Penyerahan, Implementasi dan Umpan Balik (*Delivery, Deployment and Feedback*)

Setelah tahap konstruksi selesai dilakukan, tahapan kemudian berlanjut ke tahap terakhir dari metode prototipe yaitu tahap penyerahan, implementasi, dan umpan balik. Dalam tahap akhir ini, produk hasil dari tahap sebelumnya yaitu tahap konstruksi, baik berupa prototipe maupun sistem perangkat lunak lengkap yang telah selesai dibangun diserahkan ke pengguna untuk dilakukan proses pengujian menggunakan berbagai jenis metode yang ada seperti misalnya *Black Box Testing* [61]. Pengguna akan mencoba prototipe atau sistem yang telah dibangun dan memastikan apakah seluruh fungsi yang ada telah memenuhi kebutuhan – kebutuhan yang telah didefinisikan sebelumnya, jika terdapat hal yang tidak sesuai maka pengguna akan memberikan umpan balik (*feedback*) kepada pengembang (*developer*) untuk dilakukan evaluasi dan perbaikan [59], [60]. Hal tersebut dilakukan hingga pengguna sepakat bahwa seluruh kebutuhan telah terpenuhi sehingga sebuah produk akhir sistem perangkat lunak yang telah teruji, lengkap dan fungsional dapat tercipta, yang kemudian dilanjutkan dengan proses implementasi sistem bersama pengguna dan pihak terkait dan disertai proses pemeliharaan secara berkala agar dapat meminimalisir kegagalan dan juga *downtime* terhadap sistem yang telah dibuat [21], [23].

Selanjutnya berikut adalah beberapa keunggulan metode prototipe sehingga menjadi pilihan metode yang penulis gunakan dalam membangun sistem perangkat lunak pada penelitian ini [58], [59], [65]:

- a. Dapat diterapkan dalam pembangunan sistem dengan skala kecil maupun besar.
- b. Mampu mengumpulkan informasi kebutuhan dan persyaratan sistem yang akan dibangun dari pengguna secara konkret, tidak secara abstrak.

- c. Melibatkan calon pengguna sistem perangkat lunak yang akan dibangun secara aktif dalam pembuatan prototipe sistem untuk proses analisa dan perancangan.
- d. Komunikasi antara pengguna dengan pengembang sistem terjalin dengan baik sehingga dapat memberikan keuntungan di kedua pihak.
- e. Masukan atau umpan balik yang diberikan pengguna sesuai dengan keinginannya.
- f. Setiap proses evaluasi dan perbaikan dari prototipe maupun sistem perangkat lunak menggunakan acuan masukan atau umpan balik secara langsung dari pengguna yang akan menggunakan sistem sehingga sistem yang akan dibangun dapat dibuat secara lebih reliabel.
- g. Memberikan keuntungan bagi pengembang sistem dalam proses perancangan karena dapat menggunakan rancangan sebelumnya yang lebih disempurnakan.
- h. Pelaksanaan proses implementasi atau penerapan bisa lebih mudah karena pengguna ikut berperan aktif dalam pembangunan sistem sehingga telah mengetahui seperti apa sistem yang dibangun.
- i. Dapat menghemat waktu dan biaya dalam proses pembangunan sistem.

### 2.2.5. Metode Pengujian Perangkat Lunak *Black Box*

Metode pengujian kotak hitam atau sering disebut sebagai *Black Box Testing* merupakan salah satu metode pengujian perangkat lunak yang menggunakan pendekatan jenis *Alpha Testing*. Tujuan utama dari metode pengujian *Black Box* adalah untuk menguji fungsionalitas suatu sistem perangkat lunak yang telah dibuat, seperti yang dapat dilihat pada gambar cuplikan angket pengujian metode kotak hitam di bawah.

Tabel 2. Tabel rencana pengujian

| No | Akses | Kelas Uji           | Butir Uji                              | Jenis Pengujian | Status   |
|----|-------|---------------------|--|-----------------|----------|
| 1  | Admin | <i>Login</i>        | Verifikasi data                        | <i>Blackbox</i> | Berhasil |
| 2  | Admin | Master data Petugas | Tambah data<br>Ubah data<br>Hapus data | <i>Blackbox</i> | Berhasil |

Sumber: R. Wahyudi dan K. Rhinaldi, 2018

Gambar 2.2. Cuplikan Contoh Pengujian Perangkat Lunak Metode *Black Box*

Dalam metode ini pengujian dilakukan melalui berbagai skenario pengecekan masukan (*input*) dan keluaran (*output*) di seluruh fungsionalitas sistem perangkat lunak, apakah sudah sesuai dengan kebutuhan fungsionalitas yang telah didefinisikan oleh pengguna atau belum [66]. Proses pengujian dilakukan oleh sang pengembang (*developer*) sistem perangkat lunak itu sendiri atau orang – orang yang memang ahli di bidang seputar perangkat lunaknya, setelah perangkat lunak tersebut selesai dibuat. Dengan melalui proses pengujian menggunakan metode *Black Box* ini dapat bermanfaat untuk mengidentifikasi dan menghilangkan lebih dini berbagai masalah (*bug*) sebelum perangkat lunak diserahkan atau diluncurkan dan digunakan langsung oleh pengguna akhir (*end-user*) [8], [67], [68].