

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam melakukan penelitian mengenai “Rancang Bangun Sistem Informasi Manajemen Kepegawaian (SIMPEG) Berbasis Web di PT Sembilan Hakim Nusantara”, penulis melakukan peninjauan terhadap penelitian terdahulu yang pernah dilakukan beberapa tahun terakhir, peneliti mendapatkan lima hasil penelitian terdahulu yang berkaitan dan dapat dijadikan sebagai pembandingan dengan penelitian yang akan dilaksanakan.

Tabel 2.1 Penelitian Terdahulu

Nama Peneliti	Judul	Metode	Hasil Penelitian	Perbedaan Penelitian
(Madeira, 2017)	Rancang Bangun Sistem Informasi Manajemen Kepegawaian Berbasis Web (Studi Kasus : Kecamatan Natarbora Timor-Leste)	<i>Waterfall</i>	Sistem Informasi Manajemen Kepegawaian Berbasis Web	Pada penelitian ini berfokus pada pengolahan data pegawai, pensiun, presensi, dan kenaikan jabatan
(Pakpahan dan Sinulingga, 2020)	Perancangan Aplikasi Manajemen Kepegawaian Berbasis Web di PT. Asian Isuzu Casting	<i>System Life Development Cycle</i>	Sistem Informasi Kepegawaian Berbasis Web	Pada penelitian ini berfokus pada pencatatan kontrak, cuti, izin, dan resign

Tabel 2.2 Penelitian Terdahulu (lanjutan)

Nama Peneliti	Judul	Metode	Hasil Penelitian	Perbedaan Penelitian
(Wibawa dan Julianto, 2016)	Rancang Bangun Sistem Informasi Kepegawaian (Studi Kasus : PT Dekatama Centra)	<i>waterfall</i>	Aplikasi sumber daya manusia atau <i>Human Resource Management</i>	berfokus pada pencatatan daftar kehadiran, lembur, kenaikan gaji, dan surat peringatan
(Apriandy <i>et al.</i> , 2016)	Sistem Informasi Manajemen Kepegawaian Berbasis Web pada PT. Tunas Cahaya Mandiri	<i>Rational Unified Process</i>	Sistem Informasi Manajemen Kepegawaian berbasis web	metode yang digunakan yaitu <i>Rational Unified Process</i> dan rancangan sistem menggunakan <i>Entity Relationship Diagram (ERD)</i>
(Nurmawan dan Mulyati, 2019)	Sistem Informasi Kepegawaian Berbasis Web Pada PT Sumatera Panca Rajo Palembang	<i>Iterative</i>	Sistem informasi kepegawaian berbasis web	menggunakan metode <i>Iterative</i>

2.1.1 Tinjauan Literatur 01

Oleh Madeira (2017) dari Fakultas Ilmu Komputer, Universitas Noratama, Surabaya, dengan judul “Rancang Bangun Sistem Informasi Manajemen Kepegawaian Berbasis Web”, dalam penelitiannya penulis mengangkat masalah tentang penyediaan informasi, penyajian dan penyimpanan data kepegawaian di kecamatan Natarbora masih dilakukan secara manual dengan mencatat data-data tersebut kedalam buku dan disimpan dalam rak. Sehingga saat melakukan pencarian data sering diperoleh data yang ganda ataupun tidak sesuai, sehingga menghambat kinerja pegawai. Tujuan penelitian ini membuat sebuah sistem informasi yang mampu mengolah data pegawai, administrasi personalia, kinerja, dan pensiun atau pemutusan hubungan kerja, hasil dari penelitian ini adalah sistem informasi manajemen kepegawaian berbasis web pada Kecamatan Natarbora Timor-Leste dengan menggunakan metode *waterfall* agar informasi yang dihasilkan lebih akurat, efisien, dan tepat.

2.1.2 Tinjauan Literatur 02

Oleh Pakpahan dan Sinulingga (2020) dari Fakultas Teknologi Informasi, Universitas Advent Indonesia, dengan judul “Perancangan Aplikasi Manajemen Kepegawaian Berbasis Web, dalam penelitian ini penulis mengangkat masalah tentang manajemen data karyawan di PT. AICC masih menggunakan *Microsoft Excel*. Sistem manajemen yang digunakan masih memiliki banyak kekurangan terutama dalam mencari data dan menampilkan data yang sudah disimpan. Untuk menghindari masalah yang ada pihak HRD dianjurkan menggunakan teknologi informasi agar membantu dalam menyimpan data dan data akan disimpan didalam

satu sistem dan dapat dibuka melalui perangkat mana saja yang terhubung dengan internet di PT. AICC. Selain itu, kegunaan yang diharapkan dari penelitian ini yaitu meningkatkan kinerja karyawan dalam melaksanakan tugas, menghemat waktu dalam pencarian data, mempermudah karyawan mengolah data dimana saja dan kapan saja, dan mengurangi kerugian yang tidak terduga karena data yang tidak teratur. Dalam penelitian ini penulis menggunakan metode SDLC (System Life Development Cycle). Hasil dari penelitian ini yaitu Sistem Informasi Kepegawaian pada PT. Asian Isuzu Casting Center yang dirancang dan komponen berjalan dengan baik, lalu sistem ini dapat membantu mempermudah dalam pembuatan laporan data karyawan di PT. AICC, dan dapat membuat laporan data cuti dan data izin yang dapat diakses dengan mudah oleh pengguna.

2.1.3 Tinjauan Literatur 03

Oleh Wibawa dan Julianto (2016) dari Program Studi Sistem Informasi, Universitas Komputer Indonesia, dengan judul “Rancang Bangun Sistem Informasi Kepegawaian”. Dalam penelitian ini peneliti mengangkat masalah pada PT Dekatama Centra pengolahan data SDM sudah terkomputerisasi, tetapi penggunaannya belum optimal seperti pengolahan data absen sidik jari pegawai hanya diconvert ke Microsoft Excel, sehingga adanya kelemahan-kelemahan yang terjadi dalam manajemen SDM, seperti informasi perhitungan lembur, perhitungan pinjaman, perhitungan absensi, prestasi kerja tidak akurat dan tingkat kecepatan akses data yang lambat. Berdasarkan permasalahan tersebut peneliti memberikan solusi yaitu dengan membuat suatu aplikasi sistem manajemen sumber daya yang akan mencatat data dan daftar kehadiran pegawai, waktu masuk dan pulang kerja, data hari libur, pinjaman, lembur, cuti, kenaikan gaji, surat peringatan, data keluar

dan masuk pegawai. Metode yang digunakan pada penelitian ini yaitu *waterfall* dan hasil yang didapatkan pada penelitian ini yaitu aplikasi sumber daya manusia atau *Human Resource Management*.

2.1.4 Tinjauan Literatur 04

Oleh Apriandy dkk. (2016) dari Program Studi Sistem Informasi, STMIK GI MDP, Palembang, dengan judul Sistem Informasi Manajemen Kepegawaian Berbasis Web. Dalam penelitian ini peneliti mengangkat masalah sistem kepegawaian yang ada saat ini membuat pihak manajemen yang kesulitan dalam mengontrol dan memantau kinerja karyawan, waktu untuk merekap data absensi dan data gaji pegawai terlalu lama, keamanan data yang belum terjamin dari kecelakaan maupun tindak kejahatan, dan sistem yang berjalan tidak menghasilkan informasi yang lengkap. Berdasarkan permasalahan tersebut penulis merancang sistem informasi manajemen kepegawaian pada PT. Tunas Cahaya Mandiri agar pihak manajemen dapat memantau, mengevaluasi, dan mengontrol kinerja pegawainya. Dalam penelitian ini peneliti menggunakan metode RUP (*Rational Unified Process*). Hasil dari penelitian ini adalah Sistem Informasi Manajemen Kepegawaian pada PT. Tunas Cahaya Mandiri agar dapat mempermudah pihak manajemen dalam memantau kinerja karyawan secara lebih efektif dan efisien dan membantu pihak staff operasional membuat laporan secara cepat dan akurat.

2.1.5 Tinjauan Literatur 05

Oleh Nurmawan dan Mulyati (2019) dari Program Studi Sistem Informasi, STMIK GI MDP, Palembang, dengan judul Sistem Informasi Kepegawaian Berbasis Web Pada PT Sumatera Panca Rajo Palembang, dalam penelitian ini peneliti mengangkat masalah tentang kegiatan operasional sehari-hari pada PT

Sumatera Panca Rajo Palembang masih menggunakan aplikasi perkantoran biasa dan belum menerapkan sistem kepegawaian, sehingga mengalami kesulitan dalam perekapan data absensi dan lembur karena masih menggunakan kertas yang disimpan didalam folder, sehingga rentan terjadinya kerusakan dan kehilangan berkas, dan untuk perekapan data absensi bagian operasional harus mencari data yang tersimpan dahulu sehingga perusahaan mengalami kesulitan dalam mengevaluasi kinerja pegawainya, metode yang digunakan yaitu *Iterative Model*. Hasil dari penelitian ini adalah sebuah sistem informasi kepegawaian berbasis web pada PT Sumatera Panca Rajo Palembang yang dapat memperoleh informasi mengenai kehadiran, data pegawai, mutasi, pemberhentian pegawai, perekrutan pegawai, dan dapat melakukan penilaian kinerja terhadap kinerja pegawainya.

Perbedaan penelitian dari kelima jurnal diatas adalah tempat objek penelitian dilakukan yaitu PT Sembilan Hakim Nusantara, diantaranya pada jurnal pertama berfokus pada pengolahan data pegawai, pensiun, presensi, dan kenaikan jabatan, sedangkan peneliti berfokus pada data kontrak, pegawai, bonus, gaji, pengeluaran, penempatan tugas, cuti, dan laporan ,pada jurnal kedua berfokus pada pencatatan kontrak, cuti, izin, dan resign, pada jurnal ketiga berfokus pada pencatatan daftar kehadiran, lembur, kenaikan gaji, dan surat peringatan, pada jurnal keempat metode yang digunakan yaitu *Rational Unified Process* dan rancangan sistem menggunakan *Entity Relationship Diagram (ERD)* sedangkan peneliti menggunakan *Unified Modeling Language (UML)* dan menggunakan metode *waterfall*, dan pada jurnal kelima menggunakan metode *Iterative Model*.

2.2 Sistem

Sistem adalah kumpulan yang terdiri dari unsur, komponen, atau variabel yang telah diatur dan dapat berinteraksi satu sama lain serta saling bergantung.

Sistem ini memiliki karakteristik sebagai berikut (Sutabri, 2012):

1. Komponen Sistem (*Components*)

Suatu sistem terdiri dari beberapa komponen yang terhubung untuk membentuk satu kesatuan. Komponen sistem tersebut dapat berupa bentuk subsistem yang memiliki sifat dan fungsi tertentu yang dapat mempengaruhi proses sistem secara keseluruhan.

2. Batasan Sistem (*Boundary*)

Sistem memiliki ruang lingkup yang membedakan antara sistem, keterbatasan sistem ini membuat suatu sistem terlihat sebagai satu kesatuan yang tidak dapat dipisahkan.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan eksternal sistem adalah pengoperasian sistem, yang dipengaruhi oleh segala aspek yang berada di luar batasan sistem, yang memiliki manfaat dan kerugian bagi sistem. Lingkungan eksternal yang menguntungkan harus dipertahankan dan dijaga, karena merupakan energi untuk sistem, sedangkan lingkungan eksternal yang merugikan harus dikendalikan agar sistem lain tidak terganggu.

4. Penghubung Sistem (*Interface*)

Penghubung sistem adalah media yang menghubungkan antara sistem satu dengan subsistem yang lainnya, dengan penghubung ini membuat sumber daya bergerak dari satu subsistem ke subsistem yang lainnya.

5. Masukan Sistem (*Input*)

Masukan sistem adalah energi dalam sistem yang terdapat pemeliharaan (*maintnce input*) dan sinyal (*sinal input*). Sebagai contoh, pada sistem komputer, program merupakan pemeliharaan masukan yang berfungsi mengoperasikan komputer, sedangkan data adalah sinyal masukan yang akan diolah menjadi informasi.

6. Keluaran Sistem (*Output*)

Energi yang sudah dikelola dan dikelompokkan menjadi hasil keluaran yang memiliki fungsi, hasil keluaran ini akan menjadi masukan bagi subsistem yang lain. Contoh sistem informasi keluaran yang dihasilkan adalah informasi.

7. Pengolah Sistem (*Procces*)

Sebelum menjadi keluaran, masukan akan dikelola oleh sistem. Sebagai contoh, sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan, apabila suatu sistem tidak memiliki tujuan, maka pegoperasian sistem menjadi tidak berguna karena sistem dikatakan berhasil saat mengenai sasaran atau tujuan yang telah direncanakan.

2.3 Informasi

Informasi adalah data yang telah diolah maupun dikelompokkan agar digunakan saat proses pengambilan keputusan. Informasi memiliki nilai yang ditentukan oleh dua hal yaitu manfaat dan biaya, informasi dikatakan bernilai apabila manfaat yang diberikan lebih berharga daripada biaya untuk mendapatkannya.

Informasi merupakan proses selanjutnya dari data yang sudah memiliki nilai, informasi dapat dikelompokkan menjadi 3 bagian, yaitu (Sutabri, 2012):

1. Informasi Strategis

Yaitu informasi digunakan guna mengambil keputusan dalam jangka panjang, yang meliputi informasi eksternal, rencana perluasan perusahaan, dan sebagainya.

2. Informasi Taktis

Informasi ini diperlukan dalam pengambilan keputusan jangka menengah, seperti informasi penjualan yang dapat digunakan untuk menyusun rencana penjualan.

3. Informasi Teknis

Informasi ini diperlukan untuk keperluan operasional setiap harinya, seperti informasi, persediaan stok, dan laporan kas harian.

2.4 Sistem Informasi

Sistem informasi adalah suatu sistem pada suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, yang berfungsi sebagai operasi organisasi yang memiliki sifat manajerial dengan aktivitas strategi dari suatu organisasi untuk mempersiapkan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan (Bahar *et al.*, 2021).

2.5 Sistem Informasi Manajemen

Sistem informasi manajemen adalah hubungan sub sistem yang menyeluruh dan teratur secara kasatmata yang mampu bertukar data sehingga menjadi informasi lewat serangkaian proses untuk menambah produktifitas yang sesuai dengan gaya dan sifat manajer atas kriteria mutu yang telah ditentukan (Madeira, 2017).

2.6 Kepegawaian

kepegawaian adalah seluruh aktivitas yang berkaitan dengan kepentingan kepegawaian. Sesuai dengan pengertian tersebut, bagian kepegawaian adalah segala aspek mengenai kedudukan, kewajiban, hak, dan pembinaan pegawai. Sistem informasi pengelolaan data pegawai ini memiliki fungsi untuk mendukung pelaksanaan fungsi dari suatu bagian dalam instansi tersebut untuk menoperasikan kegiatan pegawai dengan maksud mencapai tujuan (Madeira, 2017).

2.7 Sistem Informasi Manajemen Kepegawaian

Sistem informasi manajemen kepegawaian adalah sistem informasi yang dirancang sebagai jalan keluar untuk mengatur berbagai hal dalam pengurusan kepegawaian mulai dari penyimpanan dan pemusatan data secara terkomputerisasi, hingga mengatur berbagai macam laporan yang berhubungan dengan kepegawaian sehingga memudahkan untuk meningkatkan kebutuhan administrasi kepegawaian (Zairen dan Anggit, 2013).

2.8 Gaji

Gaji adalah bentuk pembayaran rutin dari perusahaan kepada karyawan, yang dapat didasarkan pada kontrak kerja. Besaran gaji cenderung naik turun sesuai dengan kebijakan perusahaan. Dasar pemberian gaji sendiri tumbuh dari kebijakan perusahaan dan kebijakan pemerintah terkait penggajian. Kebijakan perusahaan dapat diperoleh dengan membandingkan tingkat gaji perusahaan lain yang mirip dengan perusahaan-perusahaan ini dalam satu bidang (Nilasari, 2016).

2.9 Bonus

Bonus adalah pembayaran tambahan yang tidak termasuk dalam gaji atau upah yang diberikan kepada pekerja, memacu untuk melakukan tugasnya dengan lebih baik dan bertanggung jawab, dengan harapan memberikan keuntungan yang lebih tinggi bagi perusahaan. (Brata dan Whidyanto, 2017).

2.10 Website

Sebuah situs web yang dapat diartikan sebagai kumpulan halaman yang digunakan untuk menampilkan informasi berupa teks, gambar diam atau gerakan, animasi, atau kombinasi dari mereka, baik statis maupun dinamis, membentuk serangkaian bangunan yang saling berhubungan, masing-masing terhubung ke jaringan halaman (Bahar *et al.*, 2021).

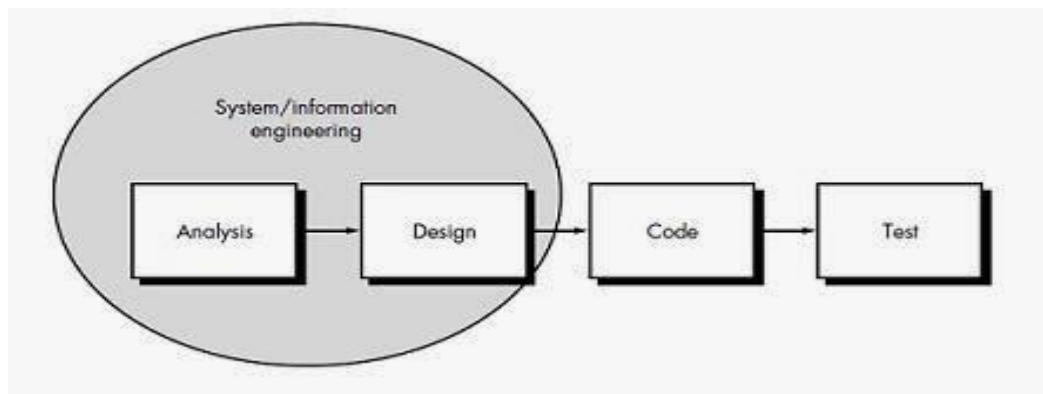
2.11 System Development Life Cycle

System Development Life Cycle atau yang lebih dikenal dengan istilah SDLC adalah metodologi yang paling umum digunakan untuk mengembangkan sistem informasi. SDLC memiliki beberapa fase yang terdiri dari perencanaan, analisis, perancangan, implementasi hingga pemeliharaan. SDLC memiliki konsep yang mendasar dari berbagai jenis model pengembangan perangkat lunak agar membentuk suatu serangkaian kerja untuk perencanaan pembuatan sistem informasi. Model-model SDLC yang umum digunakan antara lain *Waterfall* dan *Protoyping*. Model *waterfall* adalah salah satu model SDLC yang umum digunakan atau biasa disebut juga dengan model konvensional atau *classic life cycle*. SDLC juga merupakan metode yang sering digunakan dalam pengembangan sistem untuk mengenali proses perkembangan analisis dan desain. Fase – fase yang terdapat dalam SDLC yaitu (Susanto dan Andriana, 2016) :

- a. Perencanaan Sistem (*System Planning*)
- b. Analisis Sistem (*System Analysis*)
- c. Perancangan Sistem (*System Design*)
- d. Implementasi Sistem (*System Implementation*)
- e. Pemeliharaan Sistem (*System Maintance*)

2.12 Model Waterfall

Model air terjun (*waterfall*) atau yang biasa disebut sekuensial linier atau siklus hidup klasik yaitu model yang memberikan pendekatan alur hidup perangkat lunak secara terorganisir dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung, berikut adalah gambar model air terjun (A.S and Shalahuddin, 2018):



Gambar 2.1 Ilustrasi Model *Waterfall*
(Sumber : Rosa A.S dan M. Shalahuddin, 2018)

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan yang diperlukan dan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar mudah dimengerti oleh *user*.

2. Desain

Desain perangkat lunak adalah proses perancangan desain perangkat lunak

yang terdiri dari struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini memindahkan kebutuhan perangkat lunak dari tahapan analisis untuk diimplementasikan menjadi program pada tahap selanjutnya.

3. Pembuatan kode program

Desain yang telah dibuat selanjutnya dialihkan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program yang sudah dibuat berdasarkan desain yang telah ditentukan pada tahap desain.

4. Pengujian

Pengujian berfokus pada perangkat lunak untuk membuktikan bahwa semua bagian telah dilakukan pengujian agar mengurangi kesalahan (*error*) dan memastikan hasil yang dikeluarkan sesuai dengan yang dibutuhkan.

5. Pendukung (*support*) atau pemeliharaan (*maintance*)

Tahap pendukung atau pemeliharaan berguna untuk mengantisipasi terjadinya perubahan perangkat lunak yang sudah digunakan oleh user, perubahan bisa terjadi karena adanya kesalahan yang muncul atau perangkat lunak yang harus dikembangkan lebih lanjut sesuai dengan lingkungan baru.

2.13 PHP

PHP adalah sebuah bahasa pemrograman yang perintahnya dilaksanakan *server* dan kemudian hasilnya ditampilkan pada komputer *client*. PHP juga merupakan HTML *embedded*, yaitu sintaks PHP yang dituliskan bersamaan dengan sintaks HTML. Jadi PHP dan HTML adalah sinergi dua bahasa pemrograman yang saling menguatkan. Walaupun sebagian orang berpendapat HTML bukan sebuah Bahasa pemrograman. PHP juga merupakan bahasa pemrograman *open source* yang

bisa didownload gratis. Beberapa kelebihan Bahasa pemrograman PHP sebagai berikut (Haviluddin *et al.*, 2016) :

1. Keamanan

Keamanan sebuah program selain sistem operasi menjadi sangat penting. PHP menyediakan 3 jenis autentikasi *user*, yaitu *http* autentikasi, penggunaan *cookies* dan penggunaan *session*. Selain itu ada beberapa fungsi disediakan seperti *crc32*, *crypt*, *md5*, *base64-decode*, *base64-encode* dan lain-lain.

2. Integritas dengan *database*

PHP mendukung integritas, kecepatan, dan efisiensi akses ke *database* yang kebanyakan menggunakan *database* berjenis *relational* seperti *MySQL*, *PostgreSQL*, *Oracle*, *SQLite*, dan lain-lain.

3. *Cros-Platform*

PHP mendukung berbagai jenis sistem operasi seperti semua *varian Linux*, *Microsoft Windows*, *Mac Os* dan lain-lain.

4. Reliabilitas

PHP merupakan salah satu bahasa pemrograman yang berbasis web. Alasan utama adalah dukungan dokumentasi yang lengkap, aman dan banyak komunitas *helpdesk* untuk membantu para pengembang web sistem yang menggunakan PHP.

5. Harga

PHP berada dalam lisensi GPL (*GNU Public License*). Hal ini berarti bahwa PHP bebas digunakan dan didistribusikan serta gratis. Saat ini juga banyak *hosting* gratis dan *unlimited* mensupport PHP.

6. Kemudahan bermigrasi

Tujuannya adalah memperbaiki kinerja dan menambah fitur-fitur baru. Kelebihan ini karena banyaknya dukungan terhadap PHP sehingga berdampak PHP terus menerus dikembangkan.

2.14 MySQL

MySQL adalah koleksi yang besar dan kompleks dari klien, dan server perangkat lunak untuk menyimpan dan mengambil data. Terkadang seseorang yang baru untuk MySQL atau SQL sepenuhnya kewalahan oleh kedalaman dan jumlah pilihan yang terlibat dengan apa yang tampaknya menjadi tugas *database* sederhana. Sedangkan menurut Raharjo (2011) MySQL merupakan salah satu *Rational Database Management System* (RDMS) yang saat ini sedang populer dikalangan pengembang aplikasi *database*, mulai dari aplikasi dekstop maupun web yang memiliki fungsi untuk mengelola data pada aplikasi tersebut.

MySQL merupakan aplikasi yang memiliki fungsi untuk membuat, menyimpan, mengatur, dan mengelola *database* yang bisa dibuat sendiri (Suehring, 1991).

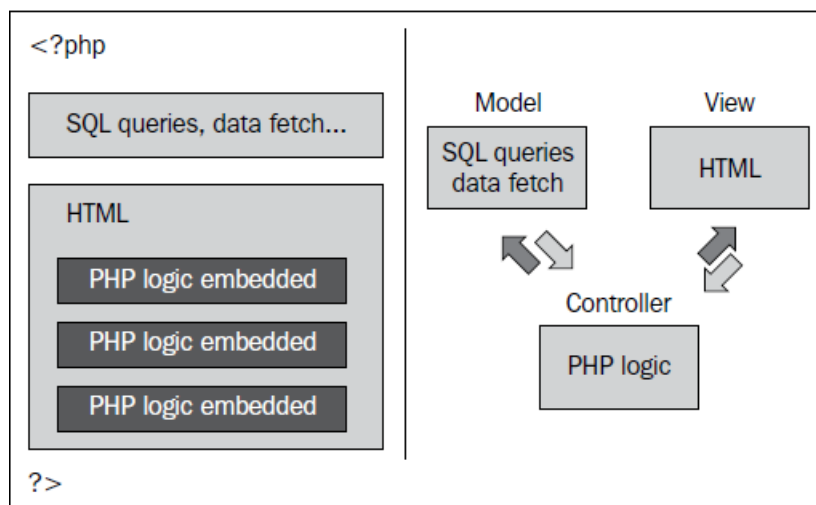
2.15 CodeIgniter

CodeIgniter adalah *framework* PHP *open source* yang kuat dengan *footprint* yang sangat kecil, dibangun untuk *programmer* PHP yang membutuhkan *toolkit* sederhana dan elegan untuk membuat aplikasi web berfitur lengkap. *CodeIgniter* adalah kerangka kerja *Model-View-Controller* (Blanco and Upton, 2009).

- a. *Models* : Model mewakili data aplikasi kita, baik itu dalam *database*, dalam *file* XML atau di mana pun. Juga, interaksi dengan *database* dilakukan di sini. Misalnya, model akan memungkinkan kita untuk mengambil, memodifikasi,

memasukkan, dan menghapus data dari *database* kita. Semua tindakan yang mengharuskan aplikasi kita untuk berbicara dengan *database* kita harus dimasukkan ke dalam model.

- b. *Views* : *File* yang ditempatkan di sini bertanggung jawab untuk menampilkan data kepada pengunjung situs, atau pengguna aplikasi. Tidak ada logika pemrograman, tidak ada *query* penyisipan atau pembaruan yang harus dijalankan di sini, meskipun akses data dapat terjadi dalam *file-file* ini. Di sini hanya untuk menunjukkan hasil dari dua lainnya. Jadi dengan mengambil data dalam model, dan menampilkannya dalam tampilan.
- c. *Controllers* : Ini bertindak sebagai penghubung antara *models* dan *view*, dan logika pemrograman terjadi di sini.



Gambar 2.2 Arsitektur MVC

2.16 Balsamic Mockups

Balsamic mockups merupakan aplikasi *wireframing* yang digunakan untuk membuat sketsa ide produk atau antarmuka web dalam bentuk kerangka, menggambarkan pada tingkat tinggi apa yang akan dilakukannya, bagaimana tampilannya, dan bagaimana fungsinya (Faranello, 2012).

Balsamiq menawarkan hal-hal yang tidak dilakukan oleh yang lain, yaitu sebagai berikut :

- a. Kemudahan penggunaan : dapat membuat *wireframe* beberapa saat setelah membuka aplikasi.
- b. Alat Sederhana : balsamiq dipenuhi dengan alat visual yang mudah dipahami yang akan membantu mentransfer ide-ide ke layar dengan cepat dan jelas.
- c. Portabilitas : *file balsamiq* dapat dengan mudah dibagikan di antara rekan dan klien. Dengan *balsamiq*, dan akan merancang dan berbagi pekerjaan di berbagai sistem operasi dan dengan siapa pun yang Anda pilih melalui email, cetak, atau *online*.
- d. Keterjangkauan : *balsamiq* adalah harga yang bisa dihargai semua orang. Dan itu karena balsamiq benar-benar hanya melakukan satu hal *wireframing*. Tapi, seperti yang akan Segera lihat, itu melakukannya dengan sangat, sangat baik.

2.17 UML

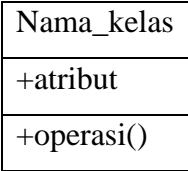


Unified Modeling Language (UML) merupakan *tools* atau alat bantu yang digunakan untuk mendesain suatu sistem yang akan dibangun dengan tersedianya bahasa pemodelan visual yang dapat membantu pengembang sistem untuk membuat cetak biru atas visi dalam bentuk yang baku. UML adalah salah satu alat bantu yang sangat diandalkan di dunia perkembangan sistem berorientasi objek. Hal ini dikarenakan UML dapat menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat *blueprint* atau cetak biru atas visi dalam bentuk baku, mudah dipahami serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkomunikasikan rancangan dengan yang lain (Nurul, 2017).

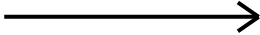

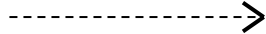

2.17.1 Class Diagram

Diagram kelas menggambarkan struktur sistem berdasarkan pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki atribut dan metode atau operasi. Susunan struktur kelas pada diagram kelas yang baik memiliki jenis-jenis kelas berikut (A.S dan Shalahuddin, 2018) :

1. Kelas main, kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
2. Kelas yang menangani tampilan sistem (*view*), kelas yang mendefinisikan dan mengatur tampilan pemakai.
3. Kelas yang diambil dari pendefinisian use case (*controller*), kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian use case.
4. Kelas yang diambil dari pendefinisian data (model), kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Tabel 2.3 Simbol-simbol *Class Diagram*

Simbol	Deskripsi
<p>Operasi</p> 	Kelas pada struktur sistem.
<p>Antarmuka / interface</p> 	Sama dengan konsep interface dalam pemrograman berorientasi objek.
<p>Asosiasi / association</p> 	Relasi antarkelas dengan makna umum biasanya juga disertai dengan <i>multiplicity</i> .

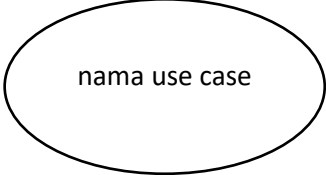
Asosiasi berarah / directed association 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain biasanya juga disertai dengan <i>multiplicity</i> .
Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
Kebergantungan / dependency 	Relasi antarkelas dengan makna ketergantungan antarkelas.
Agregasi / aggregation 	Relasi antarkelas dengan makna semua-bagian (whole-part).

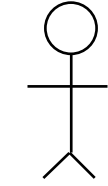

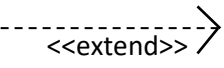

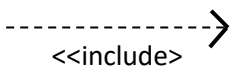
Sumber : (A.S and Shalahuddin, 2018)

2.17.2 Use Case Diagram

Use Case merupakan pemodelan sistem informasi yang akan dibuat untuk mendeskripsikan sebuah intraksi antara satu atau lebih aktor dari sistem yang akan dibuat, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (A.S dan Shalahuddin, 2018).

Tabel 2.4 Simbol-simbol *Use Case* Diagram

Simbol	Deskripsi
<i>Use Case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama <i>use case</i>

<p>Aktor / <i>actor</i></p>  <p>nama aktor</p>	<p>Orang, proses, atau sitem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor</p>
<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antara ator dan <i>use case</i> yang berpartisipasi <i>pada use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>
<p>Ekstensi / <i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan</p>
<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan sepsialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum</p>
<p><i>Include</i></p> 	<p>Reasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan</p>

	fungsinya atau sebagai syarat dijalankan <i>use case</i> ini
--	--

Sumber : (A.S and Shalahuddin, 2018)


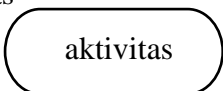
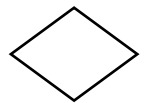
2.17.3 Activity Diagram



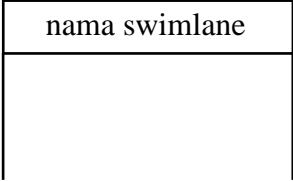
Diagram aktivitas atau *activity diagram* menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis yang ada pada perangkat lunak yang dilakukan oleh sistem bukan aktor (A.S dan Shalahuddin, 2018).

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Urutan aktivitas yang digambarkan dalam rancangan proses bisnis merupakan proses sistem yang didefinisikan.
2. Pengelompokan atau urutan tampilan dari sistem / *user interface* dimana setiap aktivitas memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujiannya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

Tabel 2.5 Simbol-simbol Activity Diagram

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu

Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber : (A.S and Shalahuddin, 2018)

2.18 Class Responsibility Collaboration Card

CRC Card adalah index kartu yang digunakan untuk merekam kelas yang disarankan, hal-hal yang mereka lakukan, tanggung jawab mereka, dan hubungan mereka dengan kolaborasi kelas lain. *CRC card* bergantung pada dua strategi yaitu kerja utama kerja kelompok kerja dan itrasi. Pedoman manajemen proyek harus mencerminkan dan menghormati hal ini. Untuk membuat kartu *CRC* berfungsi, tim analisis dan manajer harus fokus pada item-item penting ini yaitu membangun tim, memasukkan ahli aplikasi, koordinasi dengan metodologi formal, pemilihan proyek percontohan yang cermat, desain dan pengkodean yang cermat (Beck dan Cunningham, 1997)

2.19 Blackbox Testing

Blackbox testing merupakan metode teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak tanpa harus memperhatikan detail perangkat lunak. *Blackbox testing* bekerja dengan memasukkan data pada setiap formnya. *Blackbox testing* memungkinkan

pengembang perangkat lunak untuk membuat kumpulan kondisi input yang akan melatih seluruh syarat fungsional suatu program sehingga dapat mengetahui apakah program tersebut berjalan sesuai keinginan atau tidak.

Keuntungan menggunakan metode *blackbox* adalah penguji tidak harus memiliki pengetahuan lebih tentang bahasa pemrograman tertentu karena pengujian ini dilakukan dari sudut pandang pengguna untuk membantu mengungkapkan inkonsistensi dalam spesifikasi persyaratan. Ada beberapa teknik pengujian dalam menggunakan *blackbox*, salah satunya *Equivalence Partitions*.

Equivalence partitions merupakan sebuah pengujian berdasarkan inputan data pada setiap form yang ada pada aplikasi yang akan diuji, setiap menu inputan akan dilakukan pengujian dan dikelompokkan berdasarkan fungsinya, baik itu hasilnya valid atau tidak valid (Sinulingga *et al.*, 2020).