

## BAB II TINJAUAN PUSTAKA

### 2.1 Kajian Pustaka

Berdasarkan hasil penelitian terdahulu, berikut kajian pustaka yang memiliki kesamaan dengan hasil penelitian pada Tabel 2.1:

Tabel 2.1 Penelitian terdahulu

No	Judul	Perbandingan	Kontras	Mengkritik	Mempersatukan	Kesimpulan
1	Implementasi <i>System Informasi Akademik Berbasis Web</i> Menggunakan <i>Framework Laravel</i> [5]	Penelitian ini dengan melakukan perancangan <i>system informasi akademik</i> yang berbasis web dengan menggunakan <i>Framework laravel</i> .	Membahas penggunaan <i>Framework Laravel</i> dalam perancangan <i>system informasi akademik</i> yang berbasis web.	Kekurangannya terletak pada penelitian yang kurang menjelaskan secara rinci mengenai implementasi <i>Framework laravel</i> dan hanya berfokus kepada kelebihannya saja.	Penelitian ini memiliki tujuan untuk melakukan implementasi penggunaan <i>Framework laravel</i> pada web.	Hasil penelitian menunjukkan kelebihan-kelebihan yang didapatkan daripada pengguna <i>Framework laravel</i> .
2	Rancangan Bangunan Aplikasi Monografi Kecamatan Menggunakan Metod	Melakukan requirement planning, dimana pengguna mengidentifikasi tujuan-tujuan aplikasi, serta syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut.	Peng analisis dan pemrogram dapat bekerja dengan dan menunjukkan proses visual desain dan pola kerja	Hanya dilakukan berdasarkan studi kasus pada kelurahan dan kecamatan.	Peneliti melakukan pengembangan <i>system</i> dengan menggunakan metode RAD menghasilkan aplikasi yang memenuhi kebutuhan pengguna.	<i>System informasi monografi</i> yang sedang berjalan di dalam melakukan pengumpulan data dari kelurahan masih menggunakan dokumen kemudian di

No	Judul	Perbandingan	Kontras	Mengkritik	Mempersatukan	Kesimpulan
	e Rapid Application Development (RAD) [6]		terhadap pengguna			input dalam excel, sedangkan untuk menampilkan hasil dari monografi setiap kelurahan di kecamatan menggunakan <i>whiteboard</i> .
3	Aplikasi Point Of sales Menggunakan <i>Framework</i> Laravel [7]	penelitian ini melakukan perancangan aplikasi <i>Point Of Sales</i> .	Menjelaskan proses pengelolaan toko dengan aplikasi <i>Point Of Sales</i> menggunakan perangkat	Hasil pengujian menggunakan black box, tetapi tidak penjelasan secara detail hasil pengujian.	Tujuan penelitian ini adalah pembuatan aplikasi Point Of Sales guna mempermudah proses pengelolaan toko.	Hasil menunjukkan <i>Framework</i> laravel dalam penelitian memberikan keuntungan, black box testing modul berfungsi dengan baik.
4	Penggunaan <i>Framework</i> Laravel Dalam Rancangan Bangunan Modul Back-End Artikel Website Bisnis	Penelitian ini melakukan rancangan bangun model back-end artikel pada website BisnisBisnis.Id	Membahas perancangan model back-end artikel pada website BisnisBisnis.Id	Kekurangan pada penelitian ini adalah tidak disebutnya testing	Penelitian ini memiliki tujuan untuk membangun sebuah <i>system</i> dalam berbagai versi.	Hasil penelitian ini menunjukkan bahwa penggunaan <i>Framework</i> laravel dalam website memudahkan dalam segi penggunaan.

No	Judul	Perbandingan	Kontras	Mengkritik	Mempersatukan	Kesimpulan
	Bisnis. ID [2]					
5	Implementasi <i>Framework</i> Laravel Dalam <i>System</i> Pendaftaran Mahasiswa Baru Politeknik Kota Malang [8]	Penelitian ini mencoba menggunakan <i>Framework</i> laravel pada <i>system</i> pendaftaran mahasiswa baru di Politeknik Kota Malang.	Dalam penelitian ini akan dibangun sebuah platform web yang dapat diakses kapanpun dan dimanapun.	Kekurangan dalam penelitian ini terletak pada hasil dan pembahasan yang hanya memuat data tanpa penjelasan aplikasi yang dibentuk hanya menunjukkan informasi.	Tujuan dari penelitian ini adalah mempermudah aplikasi mengenai pendaftaran mahasiswa baru Politeknik Kota Malang menggunakan aplikasi yang dapat diakses kapanpun dan dimanapun.	Hasil menunjukkan bahwa dengan adanya aplikasi mempermudah mahasiswa baru dan pegawai dalam mengunggah dan mengarsipkan data.
6	Rancangan Aplikasi Helpdesk (A-Desk) dengan Berbasis Web dengan <i>Framework</i> Laravel (studi kasus di PDAM)	Penelitian ini melakukan perancangan aplikasi Helpdesk (A-Desk) berbasis web menggunakan <i>Framework</i> laravel dan teknologi postgresQL serta metode Rapid Application Management Surabaya	Membahas mengenai penggunaan <i>Framework</i> laravel dan teknologi PostgreSQL serta metode Rapid Application Management.	Pengujian menggunakan black box testing tidak menjelaskan secara rinci mengapa kepuasan serta persetujuan.	Tujuan penelitian ini adalah untuk memberikan penyelesaian masalah terhadap penginputan data manual di PDAM Surya Sembada Kota Surabaya	Hasil dari penelitian ini akan menunjukkan kepuasan serta persetujuan penggunaan aplikasi helpdesk di PDAM Surya Sembada Kota Surabaya.

No	Judul	Perbandingan	Kontras	Mengkritik	Mempersatukan	Kesimpulan
	Surya Sembada Kota)[9]					
7	<i>System Informasi Reservasi Fasilitas Universitas Negeri Padang Berbasis Framework laravel</i> .[10]	Penelitian ini mencoba merancang sistem informasi reservasi fasilitas Berbasis <i>Framework laravel 5</i> , serta javascript, <i>MySQL</i> sebagai DBMS.	Membahas Mengenai <i>System informasi</i> yang Dirancang Menggunakan teknik MVC, Universitas, Melainkan hanya secara umum.	Untuk contex <i>diagram</i> , use case <i>diagram</i> , serta entity relationship <i>diagram</i> tidak dijelaskan dengan PHP	Bertujuan untuk meminimalisir pencatatan data manual demi meningkatkan orientasi online, secara rinci.	Berdasarkan hasil uji coba <i>system</i> menunjukkan bahwa <i>system</i> informasi reservasi yang dirancang menggunakan PHP dengan <i>Framework laravel</i> serta teknik MVC menunjukkan keberhasilan dalam setiap uji yang dilakukan.

No	Judul	Perbandingan	Kontras	Mengkritik	Mempersatukan	Kesimpulan
8	Rancangan Bangunan Aplikasi Marketplace Penyedia Jasa Les Private di Kota Pontianak Berbasis Web [11]	Penelitian ini menjelaskan proses pembuatan aplikasi marketplace dengan berbasis web.	Dimana menggunakan <i>Framework</i> laravel yang digabungkan dengan media sosial terhadap OpenGraph API dan WebPush Menggunakan OneSignal API.	Hasil dari penelitian kurang menjelaskan pertanyaan mengenai <i>Framework</i> laravel.	Tujuan dari penelitian ini ialah untuk mempermudah pencari dan penyedia jasa, dimana <i>Framework</i> laravel memainkan peran dalam hal tersebut.	Hasil menunjukkan bahwa <i>Framework</i> laravel yang diintegrasikan dengan media sosial, berhasil bekerja dengan maksimal. Programmer merasa <i>Framework</i> laravel mempermudah perancangan aplikasi.
9	Pembangunan Aplikasi Pelaporan Kecelakaan Lalu Lintas dengan Web Menggunakan <i>Framework</i> Laravel. [12]	Penelitian ini mencoba merancang sebuah aplikasi pelaporan kecelakaan lalu lintas dengan menggunakan <i>Framework</i> laravel dan GPS.	Membahas mengenai bagaimana aplikasi bekerja antara pelapor dan admin secara maksimal.	Kekurangan terletak pada masih adanya kesalahan terkait akurasi data yang masuk sehingga admin perlu menghapus secara manual melalui website.	Tujuan dari penelitian ini ialah menciptakan sebuah aplikasi yang dapat mempermudah pelaporan kecelakaan lalu lintas.	Hasil menunjukkan bahwa penggunaan <i>Framework</i> laravel dibantu dengan GPS dapat mempermudah aplikasi, template blade serta grafik dalam package lavachart membantu meningkatkan

No	Judul	Perbandingan	Kontras	Mengkritik	Mempersatukan	Kesimpulan
						kualitas dari segi kegunaan dari aplikasi.
10	Rancang Bangun Repository Publikasi Ilmiah Dosen Dengan Web Menggunakan Framework Laravel [13]	Merancang sebuah repository publikasi dimana berkas tersebut dapat dikumpulkan melalui website sehingga tidak secara manual.	Membahas perancangan repository dengan menggunakan metode waterfall yang dapat memberikan kemudahan bagi penggunaanya	Penelitian ini, metode yang digunakan tidak dijelaskan secara rinci khususnya pada kekurangan metode yang digunakan dengan metode lain yang hampir serupa.	Tujuan dari penelitian ini ialah memberikan kemudahan bagi dosen untuk mengunggah artikel ilmiah, serta kemudian repository tersebut dapat digunakan secara luas dalam lingkup universitas khususnya dalam menunjang akreditasi jurusan.	Hasil daripada penelitian ini berdasarkan pengujian yang dilakukan menunjukkan bahwa perancangan bangun repository publikasi memberikan manfaat, kemudahan, dan kenyamanan bagi penggunaanya.

Berdasarkan penelitian yang sudah dilakukan sebelumnya, pada penelitian yang diajukan memilih menggunakan *Framework* Laravel dan Menggunakan Metode Rapid Application Development (RAD) dikarenakan dengan menggunakan *Framework* laravel dan metode RAD tersebut memiliki tingkat akurasi yang tinggi.

## 2.2 Dasar Teori

### 2.2.1 Rancang Bangun

Definisi perancangan secara umum adalah suatu proses dimana memiliki tujuan untuk menganalisis, menilai, serta memperbaiki atau menyusun suatu *system*, baik berupa *system* fisik maupun non-fisik yang optimal untuk dimasa yang akan datang dengan memanfaatkan berbagai informasi yang ada [14].

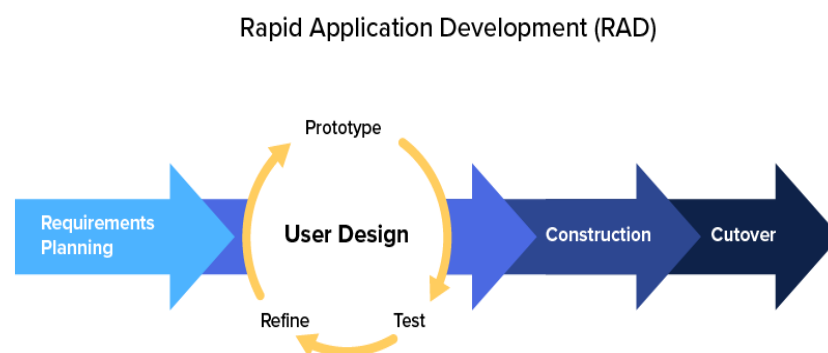
Dalam metode teknik, perancangan suatu alat merupakan salah satunya oleh karena demikian langkah-langkah dalam pembuatan perancangan akan mengikuti prosedur teknik. Seorang penelitian bernama Morris Asimov menjelaskan bahwa perencanaan merupakan suatu aktivitas yang memiliki maksud menuju kepada pemenuhan kebutuhan manusia, terutama yang kemudian dapat menjadi salah satu bagian dari peradaban umat manusia [14].

Definisi tersebut setidaknya menjelaskan bahwa ada tiga hal yang perlu diperhatikan dalam perancangan, yaitu:

1. Aktivitas yang dimiliki maksud tertentu
2. Pemenuhan kebutuhan manusia
3. Pertimbangan teknologi mempengaruhinya

### 2.2.2 Rapid Application Development (RAD)

Pendekatan yang mengarah pada objek terhadap pengembangan dan perangkat-perangkat lunak. Tujuan RAD guna menghemat waktu yang mana biasanya diperlukan dalam siklus hidup pengembangan *system* tradisional antara perancangan serta penerapan suatu *system* informasi. Setidaknya ada tiga fase di dalam RAD yang man mengikutsertakan antara penganalisis dan pengguna dalam tahapan penilaian, perancangan, dan penerapan



Sumber Gambar: [15]

Langkah Langkah Rapid Application Development sebagai berikut:

a. Requirements Planning

Requirements Planning adalah proses perencanaan yang berhubungan dengan kebutuhan *system* yang akan dibangun, proses ini juga sangat erat hubungannya dengan proses pengumpulan data, proses requirement dilakukan dengan metode diskusi, sehingga kebutuhan pengguna lebih jelas dan arah dari *system* lebih tertuju pada hasil yang diharapkan.

b. Desain Pengguna (User Design)

Tahap membuat rancangan yang akan diusulkan agar sesuai dengan kebutuhan, berjalan sesuai rencana dan diharapkan dapat mengatasi masalah yang sedang terjadi. Pada penelitian ini, desain *system* yang digambarkan menggunakan Tools Unified Modeling Language (UML).

c. Construction

Tahap ini adalah tahap memulai membuat *system* yang sudah direncanakan. Memulai menyusun suatu kode program atau biasa disebut coding, untuk merubah desain *system* yang telah dibuat menjadi sebuah aplikasi yang telah direncanakan agar dapat digunakan.

d. Testing

Pengujian *system* yang dibangun menggunakan metode pengujian *Blackbox*, dimana semua kebutuhan utama dalam *system* diuji secara fungsionalitasnya [15].

### 2.2.3 Black Box Testing

Pengujian *system* dapat dilakukan dengan beberapa cara, salah satu diantaranya yaitu *black box testing*. Metode black box testing digunakan untuk menguji fungsi operasional suatu *system* [16].

Salah satu jenis pengujian *Blackbox* testing adalah dengan menggunakan teknik *Equivalence Class Partitioning* yang peneliti gunakan untuk menguji masukan serta membagi masukan kedalam



kelompok kelompok berdasarkan fungsinya. Sehingga didapatkan sebuah *test case* yang akurat [17].

#### 2.2.4 Aplikasi

Secara umum aplikasi didefinisikan sebagai alat terapan dimana memiliki fungsi secara khusus dan terpadu sesuai kemampuan yang tertanam, aplikasi diartikan sebagai perangkat komputer yang siap digunakan oleh pengguna [18]. Aplikasi merupakan suatu subkelas dari suatu perangkat lunak komputer yang memanfaatkan kemampuan secara langsung untuk melakukan tugas yang pengguna inginkan.

#### 2.2.5 Visual Studio Code

*Visual Studio code* adalah salah satu software yang sifatnya sangat ringan, tetapi kuat *editor* kode yang mana sumbernya berjalan dari desktop. Muncul dengan *built-in* dukungan untuk *JavaScript*, naskah dan *Node.js* dan memiliki *array* beragam ekstensi yang tersedia untuk bahasa lain, termasuk C++, C #, Python, dan PHP. Dalam hal ini didasarkan pada sekitaran Github ini Elektron, yang merupakan versi cross-platform dari Atom komponen kode-Editing, berdasarkan JavaScript dan *HTML 5*. Editor tersebut adalah fitur yang lumayan lengkap lingkungan pengembangan terpadu (IDE) dibuat untuk pengembangan yang bekerja dengan teknologi cloud yang terbuka Microsoft, Visual studio Cod menggunakan opensource NET perkakas guna memberikan dukungan untuk ASP.NET C # kode, membuat alat pengembangan Omnisharp NET serta Compiler Roslyn. Antar muka yang mudah untuk bekerja, sebab berdasarkan pada gaya explorer umum, dengan panel di sebelah kiri, yang mengarahkan semua file dan folder anda mempunyai akses ke panel editor di sebelah kanan, yang menunjukkan isi dari file yang telah dibuka. Dalam hal ini, tentunya editor sudah dikembangkan dengan sangat baik, dengan *intellisense* dan *autocomplete* yang bekerja dengan baik untuk JSON, CSS, *HTML* {kurang}, dan Node JS.

*Visual Studio Code* dirancang untuk bekerja dengan menggunakan alat yang ada, dan Microsoft menyediakan tempat dokumentasi untuk membantu pengembangan bersama, dengan bantuan untuk bekerja dengan ASP.NET 5, Node.js, dan Microsoft naskah, serta alat-alat yang dapat digunakan untuk membantu membangun serta mengelola aplikasi Node.js. *Visual Studio Code* memang ditargetkan pada pengembangan JavaScript yang menginginkan alat pengembangannya lengkap untuk scripting server-side mereka dan ingin usaha dari Node.js untuk kerangka berbasis NET. *Visual Studio Code* adalah belum solid, lintas platform code Editor ringan, yang dapat dipakai oleh siapa saja guna membangun aplikasi untuk web [19].

#### **2.2.6 Xampp Server**

Dalam pembuatan sebuah website tentu saja setiap programmer menggunakan web server untuk mengkoneksikan file- file website ke basis data. Beberapa diantaranya web server yang sering digunakan adalah Apache, Web Server, dan sebagainya. Di dalam penelitian ini, peneliti menggunakan Xampp server dalam membangun web tersebut.

Xampp merupakan salah satu paket instalasi Apache, PHP, dan MySQL. Dengan menggunakan aplikasi ini dapat secara langsung melakukan instalasi Apache, PHP, dan MySQL sekaligus Aplikasi XAMPP yang dapat diperoleh dengan melakukan download [19].

#### **2.2.7 Framework Laravel**

*Framework* laravel didefinisikan sebagai sebuah struktur konsep dasar yang digunakan untuk memecahkan atau menangani suatu permasalahan yang kompleks [14]. Secara singkat, *Framework* dapat dikatakan sebagai sebuah wadah ataupun sebuah kerangka dari sebuah website yang akan dibuat. Penggunaan kerangka tersebut mempersingkat waktu yang habis terbuang dalam pembuatan website serta mempermudah perbaikan jika kemudian ditemukan ada kesalahan [1].

### 2.2.8 MySQL

Secara sederhananya, database bisa kita sebut sebagai gudang data. Secara teori, database adalah suatu kumpulan data yang bersifat kompleks, data-data tersebut disusun menjadi beberapa kelompok dengan tipe data yang sejenis yang disebut table/entity), di mana datanya dapat saling berhubungan satu sama lain atau dapat berdiri sendiri, sehingga memudahkan pengaksesan. MySQL adalah bentuk database yang di awalnya hanya berjalan terhadap *system* Unix dan Linux. Namun seiring waktu dan banyaknya peminat yang menggunakan database ini, MySQL akhirnya merilis versi yang mudah diinstal pada semua platform, termasuk Windows. Lisensi dari MySQL adalah freeware. Yang mana kita dapat mendownload dan menggunakannya tanpa perlu mengeluarkan biaya. Walaupun kita menjual produk dengan mencantumkan software MySQL, kita tidak akan melanggar hak cipta. Mungkin kalau bagi kita yang baru dengan MySQL akan merasa kebingungan dengan dua kata "SQL" dan "MySQL". Biasanya pertanyaan yang muncul adalah, apakah SQL itu dan apa bedanya dengan MySQL? SQL berasal dari kata "Structured Query Language". SQL adalah salah satu bahasa permintaan yang sifatnya terstruktur serta melekat pada satu database atau SDBD tertentu, sedangkan MySQL merupakan databasenya. Dapat dikatakan bahwasannya, MySQL merupakan SDBD-nya dan SQL adalah sebuah perintah atau bahasa yang melekat di dalam SDBD tersebut. Struktur dasar dari ekspresi SQL terdapat tiga klausa, yaitu: select, from dan where.

a. Select

Operasi ini digunakan untuk mendaftar semua atribut yang diinginkan sebagai hasil suatu

b. From

Operasi ini mencatat semua relasi yang discan dalam evaluasi suatu query.

c. Where

Operasi ini terdiri dari sebuah predikat yang menyangkut atribut atribut dari relasi yang muncul dalam klausa from [20].

### 2.2.9 E-T10

E-T10 atau T10 Merupakan sebuah Surat Izin Kunjungan Tahanan. Surat Izin Kunjungan Tahanan merupakan sebuah persuratan dimana diperlukan oleh seseorang yang ingin melakukan kunjungan ke tahanan. Surat ini hanya dikeluarkan oleh Kejaksaan Negeri. Surat tersebut dikeluarkan apabila pengunjung telah melakukan pengajuan di Kantor Kejaksaan setempat untuk melakukan pengunjungan di Rutan setempat [5].

### 2.2.10 Website

Website adalah suatu bentuk media publikasi berbentuk elektronik yang tersusun dari halaman – halaman web (web page) yang berhubungan satu dengan yang lainnya menggunakan link yang diletakkan pada suatu teks atau image. Di mana website ini dibuat pertama kalinya oleh Tim Berners Lee pada tahun 1990. Website ini menggunakan bahasa *HTML* dan memanfaatkan protokol komunikasi *HTTP* yang terletak pada application layer terhadap referensi layer OSI. Kemudian halaman websitenya diakses dengan menggunakan aplikasi yang disebut internet browser [21].

### 2.2.11 PHP

Jika diartikan, PHP memiliki beberapa pandangan dalam mengartikannya, namun kurang lebihnya PHP kita ambil arti sebagai PHP. Yang mana merupakan bahasa yang dapat berjalan pada server dan hasilnya bisa ditampilkan pada client. PHP adalah produk opensource yang digunakan gratis tanpa harus mengeluarkan biaya untuk menggunakannya. Interpreter PHP dalam mengeksekusi kode PHP pada

sisi server (server side), sedangkan tanpa adanya interpreter PHP, maka semua skrip dan aplikasi PHP yang dibuat tidak akan bisa dijalankan [21].

#### 2.2.12 Database

*System* basis data adalah *system* yang terkomputerisasi yang mempunyai tujuan memelihara data yang sudah diolah dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat [22].

#### 2.2.13 Unified Modeling Language (UML)






Unified modeling language (UML) muncul disebabkan adanya kebutuhan pemodelan visual untuk menspesifikasikan, membangun, menggambarkan, dan dokumentasi dari perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi terhadap *system* dengan menggunakan *diagram* dan teks-teks pendukung UML yang hanya berfungsi melakukan pemodelan saja. Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan *system* berorientasi objek dan juga merupakan alat untuk mendukung pengembangan *system*. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek. Adapun *diagram* yang sering digunakan adalah:





1. *Use Case Diagram Use Case* atau *diagram Use Case* merupakan bentuk pemodelan untuk kelakuan (behavior) *system* informasi yang akan dibuat. *Use Case* mengartikan sebuah interaksi antara satu atau lebih aktor dengan informasi yang akan dibuat. Secara kasarnya, *Use Case* dibuat untuk mengetahui fungsi apa saja yang ada di dalam sebuah *system* informasi tersebut dan siapa saja yang berhak

menggunakan fungsi-fungsi itu. Penamaan pada *Use Case* mempunyai syarat yaitu nama harus dibuat sesimpel mungkin dan mudah dipahami. Terdapat dua hal utama pada *Use Case* yaitu pendefinisian apa yang disebut aktor dan *Use Case*.

- a. Aktor merupakan orang, proses, atau *system* lain yang dapat berinteraksi dengan *system* informasi yang akan dibuat di luar *system* informasi yang akan dibuat itu sendiri, jadi simbol dari aktor adalah orang, tapi aktor belum tentu merupakan orang.
- b. *Use Case* merupakan fungsionalitas yang tersedia di *system* sebagai unit-unit yang bertukar pesan antar unit atau aktor.
- c.



Tabel 2.2 Simbol *Use Case Diagram* [23]




No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang dimainkan pengguna ketika berinteraksi Dengan <i>Use Case</i> .
2		<i>Dependency</i>	Bentuk hubungan dimana perubahan dapat terjadi pada suatu elemen yang sifatnya mandiri ( <i>independent</i> ) dapat mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri ( <i>independent</i> ).
3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek Yang terdapat di atasnya objek induk ( <i>ancestor</i> ).
4		<i>Include</i>	Menspesifikasikan <i>Use Case</i> sebagai sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan <i>Use Case</i> targetnya memperluas perilaku dari <i>Use Case</i> Pada sumber suatu titik yang diberikan.

No	Gambar	Nama	Keterangan
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket menunjukkan secara terbatas.
8		<i>Collaboration</i>	Tindakan antara aturan-aturan dan elemen lainnya yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
9		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan <i>system</i> yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

2. *Activity Diagram Diagram* aktivitas atau *Activity diagram* menggambarkan workflow (aliran kerja) atau aktifitas dari sebuah *system* atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa *diagram* aktivitas menggambarkan aktivitas *system* bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh *system*. *Diagram* aktivitas [22].

Tabel 2.3 Simbol *Activity Diagram* [23]


No	Gambar	Nama	Keterangan
1		<i>Activity</i>	Menunjukkan bagaimana masing-masing kelas antarmuka yang saling berinteraksi satu sama lainnya
2		<i>Action</i>	<i>State</i> dari <i>system</i> yang menunjukkan eksekusi dari suatu aksi tersebut

No	Gambar	Nama	Keterangan
3		<i>Initial Node</i>	Bagaimana objek itu dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan lalu dihancurkan
5		<i>Extend</i>	Satu aliran berubah menjadi beberapa aliran pada tahap tertentu

3. *Sequence Diagram* adalah salah satu *diagram* yang menjelaskan interaksi objek dan menunjukkan (memberi petunjuk) komunikasi diantara objek-objek tersebut.

*Sequence Diagram* digunakan untuk menjelaskan suatu perilaku pada sebuah skenario dan menggambarkan bagaimana entitas dan *system* berinteraksi, termasuk didalamnya pesan yang dipakai pada saat interaksi. Keseluruhan pesan dapat digambarkan dalam urutan pada eksekusi.

Tabel 2.4 Simbol *Sequence Diagram*[23]







No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Objek <i>entity</i> , bentuk antarmuka yang saling berinteraksi
2		<i>Message</i>	Spesifikasi dari sebuah komunikasi antara objek yang memuat informasi-informasi mengenai aktivitas yang terjadi.
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek tentang informasi-informasi mengenai aktivitas yang terjadi



4. *Class Diagram* adalah Salah satu jenis *diagram* berbentuk struktur pada model UML. *Diagram* ini menggambarkan struktur, atribut, kelas, hubungan dan metode dengan sangat jelas dari setiap objeknya.

*Class Diagram* atau *diagram* kelas juga merupakan bagian terpenting dari UML (*unified modeling language*) [23].

Tabel 2.5 Simbol *Class Diagram* [23]

No	Gambar	Nama	Keterangan
1		<i>Association</i>	Apa yang dapat menghubungkan antara objek satu dengan objek lainnya.
2		<i>Nary Association</i>	Upaya menghindari asosiasi dengan lebih dari dua objek.
3		<i>Class</i>	Himpunan objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan oleh <i>system</i> yang dapat menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri