

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Literatur

Tabel 2. 1 Penelitian Terdahulu

No	Judul	Persamaan	Perbedaan	Kritik	Sintesis	Kesimpulan
1.	Sosialisasi Pengembangan <i>Prototype Company Profile</i> Berbasis <i>Website</i> UMKM Pengusaha Gerabah dan Keramik Pakerjurang Melikan Bayat – Wedi Klaten[11].	Penelitian ini mengangkat topik pengembangan <i>website</i> dengan objek penelitian yaitu UMKM Pengusaha Gerabah dan Keramik di Pakerjurang Melikan Bayat Wedi Klaten.	Penelitian ini menggunakan metode pengembangan <i>waterfall</i> dan <i>prototype</i> .	Penjelasan mengenai alur penelitian yang tidak detail.	Topik yang dapat dilakukan oleh penelitian selanjutnya dengan penelitian ini sebagai referensi yaitu penelitian pembuatan sistem <i>profile company</i> .	Hasil dari penelitian ini berupa <i>website profile company</i> yang di dalamnya terdapat fitur pembelian gerabah dan keramik yang dijual untuk mempermudah dalam berbelanja dan meningkatkan layanan bagi pelanggan

No	Judul	Persamaan	Perbedaan	Kritik	Sintesis	Kesimpulan
2.	Perancangan Sistem Informasi Konsultasi Akademik Berbasis Website[12].	Penelitian ini mengangkat topik pengembangan <i>website</i> dengan objek penelitian yaitu Lembaga Pendidikan Tinggi Universitas Victory Sorong.	Penelitian ini menggunakan metode pengembangan <i>waterfall</i> .	Pada penelitian ini tidak disertakan desain sistem seperti <i>activity diagram</i> , <i>class diagram</i> , dan <i>Sequence Diagram</i> .	Topik yang dapat dilakukan oleh penelitian selanjutnya dengan penelitian ini sebagai referensi yaitu pembuatan sistem informais pada lembaga pendidikan	Hasil dari penelitian ini berupa <i>website</i> sistem konsultasi akademik pada Universitas Victory Sorong sehingga proses konsultasi antara dosen PA dan mahasiswa lebih efisien

No	Judul	Persamaan	Perbedaan	Kritik	Sintesis	Kesimpulan
3.	Penerapan Metode Prototype dalam Perancangan Aplikasi SIPINJAM Berbasis <i>Website</i> pada Credit Union Canaga Antutn[13].	Penelitian ini mengangkat topik pengembangan <i>website</i> dengan objek Credit Union Canaga Antutn.	Penelitian ini menggunakan metode pengembangan <i>Prototype</i> .	Pada penelitian ini tidak dilakukan adanya pengujian sistem dan uji pengguna.	Topik yang dapat dilakukan oleh penelitian selanjutnya dengan penelitian ini sebagai referensi yaitu penelitian pembuatan sistem badan usaha peminjaman dana.	Hasil dari penelitian ini berupa <i>website</i> yang memiliki fungsi untuk pengolahan data simpan pinjam seperti mengelola data anggota, simpanan, pinjaman, angsuran dan lain sebagainya.

No	Judul	Persamaan	Perbedaan	Kritik	Sintesis	Kesimpulan
4.	Perancangan <i>Website E-Commerce</i> sebagai Media Penjualan Miniatur Bus[14].	Penelitian ini mengangkat topik pengembangan <i>website</i> .	Penelitian ini menggunakan metode pengembangan <i>waterfall</i> .	Pada penelitian ini tidak dilakukan adanya pengujian sistem dan uji pengguna.	Topik yang dapat dilakukan oleh penelitian selanjutnya dengan penelitian ini sebagai referensi yaitu penelitian pembuatan <i>e-commerce</i> berbasis <i>website</i> .	Hasil dari penelitian ini yaitu <i>e-commerce</i> berbasis <i>website</i> yang diharapkan mempermudah dalam melakukan promosi dan meningkatkan penjualan produk.

No	Judul	Persamaan	Perbedaan	Kritik	Sintesis	Kesimpulan
5.	Perancangan Sistem dan Desain Undangan Digital Menggunakan Metode Waterfall Berbasis <i>Website</i> (Studi Kasus: Asco Jaya)[15].	Penelitian ini mengangkat topik pengembangan <i>website</i> dengan objek Perusahaan PT.Asco jaya.	Penelitian ini menggunakan metode pengembangan <i>Waterfall</i> .	Pada penelitian ini tidak disertakan desain sistem seperti <i>activity diagram</i> , <i>class diagram</i> , dan <i>Sequence Diagram</i> .	Topik yang dapat dilakukan oleh penelitian selanjutnya dengan penelitian ini sebagai referensi yaitu penelitian pembuatan <i>e-commerce</i> berbasis <i>website</i> .	Hasil penelitian ini berupa <i>website</i> pemesanan undangan digital menggunakan metode pengembangan <i>waterfall</i> .

No	Judul	Persamaan	Perbedaan	Kritik	Sintesis	Kesimpulan
6.	<i>Web Development Using WISDM and RAD</i> [16].	Penelitian ini mengangkat topik pengembangan <i>website</i> usaha Universitas Bengkulu (SysIO) dan Batik Museum (SysBatik).	Penelitian ini bertujuan untuk membuat sebuah <i>website</i> dengan menggunakan metode WISDM dan RAD	Pada judul penelitian tidak lengkap, subjek dari penelitian tidak dicantumkan.	Topik yang dapat dilakukan oleh penelitian selanjutnya menggunakan penelitian ini sebagai referensi yaitu pembuatan sistem dengan menggunakan metode WISDM dan RAD.	Hasil penelitian ini yaitu <i>website</i> yang memanfaatkan metode pengembangan WISDM dan RAD.

No	Judul	Persamaan	Perbedaan	Kritik	Sintesis	Kesimpulan
7.	<i>Website Development for Principals Induction Program</i> [17].	Penelitian ini mengangkat topik pengembangan <i>website</i> pengukuran kepemimpinan.	Tujuan penelitian ini yaitu mengukur tingkat pembelajaran kepemimpinan, dan memajukan sekolah.	Penjelasan mengenai pembuatan <i>website</i> tidak detail.	Topik yang dapat dilakukan oleh penelitian selanjutnya dengan penelitian ini sebagai referensi yaitu penelitian pembuatan <i>website</i> pengukuran tingkat kepemimpinan	Hasil dari penelitian ini berupa <i>website</i> dengan fungsi yang telah berjalan guna melakukan pengukuran tingkat kepemimpinan pada kepala sekolah.

No	Judul	Persamaan	Perbedaan	Kritik	Sintesis	Kesimpulan
8.	<i>Information System Development Based-on ERP and RAD Methods: Application for Activities Information Broadcasting</i> [18].	Penelitian ini mengangkat topik pengembangan <i>website</i> sistem informasi	Penelitian ini menggunakan metode pengembangan ERP dan RAD	Proses pengujian sistem tidak dicantumkan.	Topik yang dapat dilakukan oleh penelitian selanjutnya menggunakan penelitian ini sebagai referensi yaitu pembuatan sistem informasi.	Hasil penelitian ini yaitu aplikasi yang digunakan menggunakan <i>mobile</i> dan <i>website</i> untuk menyampaikan informasi secara <i>real-time</i> kepada pengguna

Berdasarkan hasil tinjauan pustaka, disimpulkan bahwa penelitian diatas membahas mengenai pembuatan *website* dengan menggunakan berbagai metode pengembangan dan *tools*. Kebaruan penelitian yang akan dilakukan yaitu pengembangan *website* menggunakan metode pengembangan SCRUM dengan tujuan dilakukan penelitian ini sebagai wadah dalam penyebaran informasi mengenai Panti Asuhan Dharmoyuwono, dan membantu untuk melakukan promosi usaha yang dimiliki oleh Panti Asuhan Dharmoyuwono.

2.2 Landasan Teori

2.2.1 Rancang Bangun

Rancang bangun merupakan proses pembuatan sistem untuk menghasilkan sebuah sistem baru, baik untuk mengganti sistem yang sudah ada atau sedang berjalan baik secara keseluruhan ataupun sebagian [19]. Rancang bangun merupakan tahapan untuk membuat gambaran yang belum pernah dirancang sebelumnya, yang kemudian diimplementasikan menjadi sebuah sketsa atau gambaran yang memiliki kegunaan tertentu. Rancang bangun merupakan kegiatan dalam melakukan penggambaran, perencanaan, dan pembuatan dari beberapa elemen yang disatukan dan memiliki nilai fungsi [20].

2.2.2 Website

Website merupakan gabungan dari beberapa halaman yang menampilkan data informasi yang berupa video, suara, gambar, teks, ataupun gabungan yang bersifat statis ataupun dinamis yang saling terhubung untuk membentuk suatu halaman yang saling terkait [21]. *Website* digolongkan menjadi 3 yaitu *website* statis dimana *website* tersebut memiliki halaman yang tidak berubah sehingga untuk melakukan perubahan harus dilakukan dengan pengetikan kode pada struktur *website* tersebut. *website* dinamis yaitu disediakan halaman khusus yang digunakan untuk melakukan perubahan pada konten *website*. Sedangkan *website* interaktif yaitu *user* dapat berinteraksi dengan *user* lainnya [22].

2.2.3 HTML dan CSS

HTML adalah bahasa pemrograman untuk membuat sebuah *website* dengan menggunakan tag <...> guna menyatakan kode yang akan diproses oleh broser dalam menampilkan halaman *website*. Penggunaan HTML merupakan dasar dalam pembangunan sebuah *website* yang lebih tersusun sebelum masuk ke tahap perancangan desain dan fungsionalitas *website*[23]. Penggunaan HTML dikolaborasikan bersama CSS untuk membuat tampilan *website* sesuai dengan desain yang telah dirancang[24].

CSS adalah bahasa pemrograman untuk mengatur tampilan desain dari *website* yang akan dibangun. Penggunaan CSS bertujuan untuk menghasilkan *website* yang menarik yang dikolaborasikan dengan HTML dalam membuat kerangka *website* . Tampilan desain yang akan digunakan seperti *font*, *outline*, *background*, dan warna dilakukan menggunakan CSS untuk membuat *website* yang lebih menarik[25].

2.2.4 PHP

PHP adalah bahasa pemrograman script server-side untuk mengembangkan *website* yang dimasukkan kedalam HTML dan dijalankan melalui server. PHP merupakan singkatan dari *Hypertext Processor*. PHP dapat digunakan secara gratis dan bersifat *open source*. PHP diciptakan pertama kali oleh Rasmus Lerdift pada tahun 1994 [26].

2.2.5 MySQL

MySQL merupakan sistem manajemen basis data berbasis relasional. Penggunaan MySQL ini bertujuan untuk menyimpan dan mengelola data yang ada dalam bentuk tabel, dimana relasi antar tabel tersebut terjaga. MySQL menggunakan bahasa *Structured Query Language* (SQL) guna memanipulasi, mengakses, dan mengelola data yang ada di dalam *database*[27].

2.2.6 Database

Database merupakan kumpulan dari data yang saling terhubung secara logika dan dekripsi yang dirancang guna memenuhi kebutuhan

informasi. Penyimpanan data secara independen dan fleksibel menjadi keunggulan dalam penggunaannya. *Database* didefinisikan terpisah dari program yang sedang dikembangkan sehingga tidak akan berdampak pada program yang sedang dikembangkan tersebut[28].

2.2.7 Figma

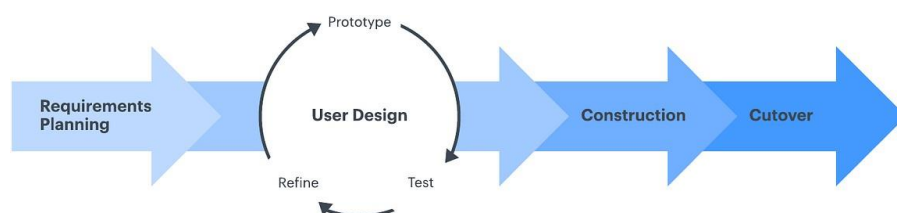
Figma merupakan sebuah aplikasi yang digunakan untuk membuat tampilan *User Interface* (UI) maupun *User Experience* (UX) pada *website* maupun aplikasi. Figma dapat digunakan sebagai alat kolaborasi antar desainer dalam membuat desain sehingga dapat menghemat waktu pengerjaan. Aplikasi figma berbasis *cloud* dan digunakan sebagai alat *prototyping* yang digunakan untuk proyek digital[29].

2.2.8 Laravel

Laravel adalah framework yang digunakan dalam perancangan *website*[30]. Laravel merupakan framework yang dapat digunakan untuk membantu memaksimalkan penggunaan PHP dalam perancangan *website* dengan mengikuti arsitektur *Model View Controller* (MVC). MVC merupakan pendekatan *software* yang memisahkan aplikasi berdasarkan komponen dari aplikasi tersebut seperti *controller*, manipulasi data, dan *user interface*[31].

2.2.9 Rapid Application Development (RAD)

Metode *Rapid Application Development* (RAD) merujuk pada pengembangan dengan waktu pengembangan yang lebih cepat dan hasil yang lebih baik dalam hal kualitas. *Rapid Application Development* (RAD) adalah sebuah metode pengembangan yang terkenal dengan kecepatan dan efektivitasnya. *Rapid Application Development* (RAD) ini pertama kali diperkenalkan oleh James Martin pada tahun 1991[32].



Gambar 2. 1 Siklus *Rapid Application Development* (RAD)[32]

Berikut tahapan dalam pengembangan sistem menggunakan metode RAD[32].

a. *Requirements Planning* (Perencanaan Kebutuhan)

Dalam tahapan ini dilakukan pengidentifikasian masalah dan pengumpulan data yang akan digunakan dalam penelitian ini untuk mengetahui tujuan dari sistem. Setelah itu, dilakukan analisis untuk mengetahui kebutuhan yang diperlukan dalam pembuatan sistem.

b. *User Design* (Desain Pengguna)

Dalam tahapan ini, dilakukan proses perancangan desain yang melibatkan pengguna untuk memastikan bahwa sistem yang dibuat sesuai dengan kebutuhan. Pada tahap ini mencakup proses iterasi dengan menggunakan *prototype* untuk memperkuat fungsi dan desain dari sistem yang akan dibuat. Proses ini dilakukan secara berulang sehingga memungkinkan penyesuaian dan perbaikan yang terus menerus mendapatkan respon yang cepat dalam menghadapi perubahan kebutuhan selama pengembangan sehingga dapat mencapai hasil yang cepat dan fleksibel.

c. *Construction* (Konstruksi)

Pada tahap *construction*, akan dilakukan pengetikan kode program untuk mengimplementasikan hasil yang telah ditetapkan sebelumnya. Selama tahap ini, *programmer* bertanggungjawab dalam memastikan sistem yang dibangun cepat dan efektif.

d. *Cutover* (Pemotongan)

Tahap *cutover* dilakukan dengan pengujian dan peluncuran sistem ke dalam lingkungan organisasi. *Programmer* memastikan bahwa sistem berjalan dengan baik dan memenuhi kebutuhan pengguna. Pada tahap ini, pengguna memiliki kesempatan untuk memberikan *feedback* dan menyetujui sistem yang telah dikembangkan.

2.2.10 *System Usability Scale* (SUS)

System Usability Scale (SUS) adalah metode pengujian *software* yang

banyak digunakan. SUS banyak digunakan karena kemudahan dalam pengujian dan hasil dari pengujian ini dapat digunakan sebagai bahan pertimbangan [33]. SUS diperkenalkan oleh John Brooke pada 1986 dengan skala *usability* yang efektif, populer, murah, dan handal. 10 instrumen pertanyaan SUS adalah

Tabel 2. 2 Instrumen Pertanyaan SUS

No	Pertanyaan	Skor
1.	Saya pikir saya akan menggunakan aplikasi ini	1-5
2.	Saya merasa <i>website</i> ini rumit untuk digunakan	1-5
3.	Saya pikir aplikasi ini mudah untuk digunakan	1-5
4.	Saya pikir saya perlu bantuan orang teknis dalam menggunakan sistem ini	1-5
5.	Saya menemukan berbagai fungsi di aplikasi ini terintegrasi dengan baik	1-5
6.	Saya pikir ini terlalu banyak ketidak konsistenan dalam sistem ini	1-5
7.	Saya akan membayangkan bahwa kebanyakan orang akan belajar dengan mudah dalam mempelajari aplikasi ini	1-5
8.	Saya menemukan aplikasi ini sangat tidak praktis	1-5
9.	Saya merasa sangat percaya diri dalam menggunakan aplikasi ini	1-5
10.	Saya perlu banyak belajar sebelum menggunakan aplikasi ini	1-5

Pada Tabel 2.2, terdapat skor 1-5 yang akan dijawab oleh responden berdasarkan seberapa banyak tingkat kesetujuan responden terhadap sistem yang diuji. Skor 1 yang berarti sangat tidak setuju, dan skor 5 yang berarti sangat setuju. Masing-masing jawaban memiliki skor penilaian seperti pada Tabel 2.3 dibawah ini

Tabel 2. 3 Skala Penilaian Skor

Jawaban	Skor
Sangat Tidak Setuju (STS)	1
Tidak Setuju (TS)	2
Ragu-Ragu (RG)	3
Setuju (S)	4
Sangat Setuju (SS)	5

Setelah data kuisioner terkumpul, maka dilakukan konversi tanggapan responden dengan perhitungan :

1. Pernyataan ganjil, yaitu 1, 3, 5, 7, dan 9. Hasil skor dari responden akan dikurangi 1

$$\text{skor SUS ganjil} = \sum Px - 1 \quad (2.1)$$

Px = jumlah pertanyaan ganjil.

2. Pernyataan genap, yaitu 2, 4, 6, 8, dan 10. Hasil skor dari responden akan digunakan untuk mengurangi 5.

$$\text{skor SUS genap} = \sum 5 - Pn \quad (2.2)$$

Pn = jumlah dari pertanyaan genap

3. Hasil dari konversi yang telah dilakukan kemudian dijumlahkan bagi setiap responden, dan selanjutnya hasil tersebut dikali 2,5 untuk mendapat nilai dengan rentang 0 – 100.

$$(\sum \text{skor ganjil} - \sum \text{skor genap}) \times 2,5 \quad (2.3)$$

4. Setelah mendapatkan skor, selanjutnya menghitung skor rata rata dengan menjumlahkan hasil skor dan dibagi dengan jumlah responden. Rumus dari perhitungan ini yaitu

$$\bar{x} = \frac{\sum x}{n} \quad (2.4)$$

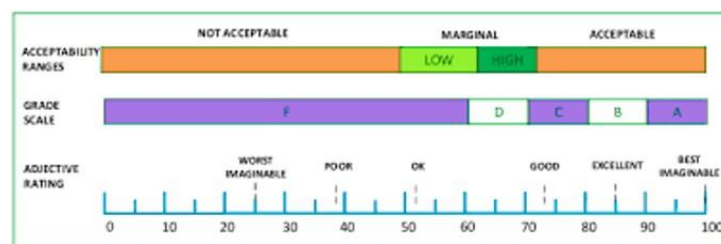
\bar{x} = skor rata-rata, $\sum x$ = jumlah skor SUS dan n = jumlah responden.

Penentuan *grade* penilaian dilakukan dengan dua cara. Penentuan pertama dilakukan dengan melihat sisi penerimaan pengguna dikategorikan menjadi *acceptable*, *not acceptable*, dan *marginal*. Pada

sisi *grade* skala dibagi menjadi 6 skala yaitu A, B, C, D, E, dan F. pada sisi adjektif rating terbagi menjadi *best imaginable*, *excellent*, *good*, *ok*, *poor*, dan *worst imaginable*. Penentuan kedua berdasarkan sisi *percentile range* (SUS skor) yang terdiri dari A, B, C, D, dan E. Penentuan hasil berdasarkan perhitungan pengguna berdasarkan SUS *Score Percentile Rank* yang dapat dilihat pada Tabel 2.4.

Tabel 2. 4 *SUS Score Percentile Rank*

Grade	Keterangan
A	Skor $\geq 80,3$
B	Skor ≥ 74 dan $< 80,3$
C	Skor ≥ 68 dan < 74
D	Skor ≥ 51 dan < 68
E	Skor lebih < 51



Gambar 2. 2 Hasil Penilaian Menggunakan SUS[31]

2.2.11 *Black Box Testing*

Black box testing adalah metode pengujian fungsionalitas sistem guna mengecek kesalahan fungsi yang terdapat pada sistem yang telah dibuat. Pengujian ini berfokus pada *input* dan *output* sistem berdasarkan persyaratan dan spesifikasi sistem yang telah dibuat. Dalam pengujian ini sistem diuji tanpa melihat struktur kode, detail implementasi, dan jalur internal.

Berikut langkah-langkah yang umum diikuti dalam melakukan pengujian sistem menggunakan *black box testing*[34] :

1. Pemeriksaan berdasarkan persyaratan dan spesifikasi sistem.
2. Penguji menginputkan *input* yang *valid* (skenario pengujian positif) untuk menguji apakah sistem memproses *inputan* dengan benar.
3. Kemudian dilakukan penginputan yang tidak *valid* (skenario pengujian negatif) untuk memverifikasi bahwa sistem dapat mendeteksi kesalahan.

4. Penguji menentukan *output* yang diharapkan.
5. Penguji *software* membuat pengujian yang telah dipilih.
6. Pengujian dilakukan.
7. Penguji *hardware* membandingkan *output* yang diharapkan dengan output aktual.
8. Dilakukan pencatatan jika terdapat *bug*, dan akan dilakukan pengujian ulang

2.2.12 Slovin Metode

Metode slovin merupakan metode untuk menentukan jumlah sampel yang akan digunakan digunakan. Berikut rumus metode slovin[35].

$$n = \frac{N}{1 + N(e)^2} \quad (2.5)$$

Dimana n = ukuran sampel, N = ukuran populasi, dan e = tingkat error


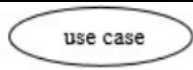
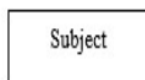

2.2.13 Unified Modelling Language (UML)

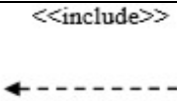
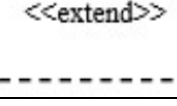


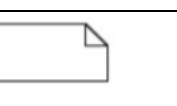
Unified Modelling Language (UML) merupakan notasi yang didukung oleh meta model tunggal guna membantu dalam pendeskripsian dan desain sistem, khususnya yang menggunakan pemrograman berorientasi objek. UML dideskripsikan dalam beberapa diagram yaitu[36] :

a. Use Case Diagram

Use case diagram berfokus pada fungsionalitas aktor yang ada pada sistem. Berikut simbol use case diagram pada tabel 2.5

Tabel 2. 5 Notasi Use Case Diagram

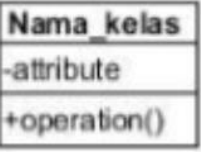




Notasi	Nama Elemen	Fungsi
 Actor/Role	Aktor	Berinteraksi dengan specialization atau superclass association ditempatkan di luar subject boundary.
 use case	Use case	Mewakili bagian fungsionalitas sistem dalam system boundary.
 Subject	Subject boundary	Menyatakan lingkup subjek.
	Association relationship	Menghubungkan aktor untuk melakukan interaksi dengan


Notasi	Nama Elemen	Fungsi
		use case.
	Include relationship	Menunjukkan inclusion fungsionalitas use case dengan use case lainnya.
	Extend relationship	Menunjukkan extension dari use case untuk menambah optional behavior.
	Generalization relationship	Menunjukkan generalisasi dari use case khusus ke umum.
	Collaboration	Interaksi aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya.
	Note	Elemen eksis saat aplikasi dijalankan dan sumber daya komputasi.

b. *Class Diagram*

Class diagram merupakan spesifikasi yang menghasilkan objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class diagram menggambarkan atribut yang ada pada sistem. Notasi pada class diagram seperti pada tabel 2.6

Tabel 2. 6 Notasi *Class Diagram*








Notasi	Nama Elemen	Fungsi
	Kelas	Struktur dari sistem
	Antarmuka	Sama dengan konsep interface dalam pemrograman berorientasi objek
	Asosiasi	Relasi antarmuka, biasanya disertai dengan multiplicity
	Asosiasi berarah	Makna kelas yang satu digunakan oleh kelas lain
	Generalisasi	Relasi antar kelas dengan makna generalisasi-spesialisasi

Notasi	Nama Elemen	Fungsi
	Agregasi	Relasi antar kelas dengan makna semua bagian

c. *Sequence Diagram*

Sequence Diagram menggambarkan interaksi antar objek yang ada pada sistem. Kegunaan dari *Sequence Diagram* ini yaitu menunjukkan pesan yang dikirim antar onjek dan interaksi antar objek yang terjadi pada sistem dalam melakukan eksekusi. Notasi pada *Sequence Diagram* seperti pada tabel 2.7

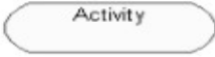



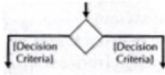


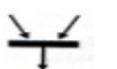
Tabel 2. 7 Notasi *Sequence Diagram*

Notasi	Nama Elemen	Fungsi
	Entity Class	Kumpulan kelas berupa netitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data
	Boundary Class	Kumpulan kelas menjadi interaksi antar aktor dengan sistem
	Control Class	Suatu objek yang yang berisi logika aplikasi yang tidak memiliki tanggungjawab pada entitas
	Message	Simbol mengirim pesan antar class
	Recursive	Menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri
	Activation	Mewakili eksekusi operasi dari objek
	Lifeline	Garis terputus dengan objek sepanjang lifeline terdapat activation

d. *Activity Diagram*

Activity diagram menunjukkan rangkaian aliran dari aktivitas dan interaksi beberapa use case. Diagram ini menggambarkan workflow dari aktifitas satu ke aktifitas lainnya. Pembuatan activity diagram ini membantu dalam memahami keseluruhan dari proses yang ada pada sistem notasi pada activity diagram seperti pada tabel 2.8

Tabel 2. 8 Notasi *Activity Diagram*

Notasi	Nama Elemen	Fungsi
	Activity	Merepresentasikan sekumpulan aktivitas
	Control flow	Menunjukkan rangkaian dari suatu eksekusi
	Initial node	Pertanda awal suatu aktivitas
	Final activity node	Pertanda akhir suatu aktivitas
	Decision node	Memastikan bahwa alur objek hanya bergerak dalam satu jalur
	Merge node	Mengembalikan berbagai decision path menjadi satu
	Fork node	Membagi perilaku menjadi aktivitas yang berjalan bersama
	Join node	Menyatukan aktivitas yang berjalan secara bersamaan