

## BAB II TINJAUAN PUSTAKA

### 2.1. Tinjauan Pustaka

Penggunaan hasil penelitan oleh peneliti lain dapat membantu penelitian yang sedang diteliti oleh penulis. Kesimpulan dari penelitan yang diteliti oleh penulis dapat dilihat dibawah ini.

1. Dalam penelitian yang diteliti oleh Mulyati, Hidayat, & Lestari (2015), peneliti menemukan masalah bahwa pengumpulan informasi mengenai buku yang terpinjam oleh anggota dan buku yang masih tersedia pada perpustakaan tidak dapat terlaksana dengan cepat dan tepat. Hasil penelitian tersebut adalah perancangan sistem berbasis web *online* yang mencatat transaksi peminjaman buku. Kesimpulan dari penelitian adalah sistem berbasis web online untuk mencatat transaksi peminjaman buku dapat memudahkan pihak perpus dalam mendapatkan informasi seperti data anggota perpustakaan, data buku, laporan transaksi dan pengembalian.
2. Dalam penelitian yang diteliti oleh Rhouf, Wardhani, & Mustain (2017), peneliti menemukan masalah bahwa pengolahan data administrasi seperti data penjualan dan pembelian ATK pada perusahaan masih tetap dilakukan dengan manual. Hasil penelitian tersebut adalah perancangan sebuah aplikasi penjualan ATK berbasis web *mobile JQuery*. Kesimpulan dari penelitian adalah aplikasi penjualan berbasis web dapat meningkatkan produktifitas dan pendapat.
3. Dalam penelitian yang diteliti oleh Brianorman & Octariadi (2017), peneliti menemukan masalah bahwa timbulnya kesulitan pada bagian perlengkapan dalam

melakukan validasi karena pada bagian kelengkapan terjadi kesulitan dalam mendokumentasikan data yang masuk. Hasil penelitian tersebut adalah perancangan sistem penggunaan ruangan berbasis web yang memiliki fitur untuk mencatat data ruangan, data peminjaman dan juga penyusunan laporan. Kesimpulan dari penelitian adalah sistem berbasis web ini dapat membantu pengolahan data peminjaman ruangan seperti, mencatat data peminjaman dan memberi informasi peminjaman seperti status permintaannya kepada pihak yang meminjam ruangan secara *online*.

4. Dalam penelitian yang diteliti oleh Maulana, Sadikin, & Izzuddin (2018), peneliti menemukan masalah bahwa pengelolaan data inventaris selama ini masih dilakukan secara manual. Hasil penelitian tersebut adalah perancangan sistem inventaris yang bersifat *web-based*. Kesimpulan dari penelitian adalah dengan menggunakan sistem inventaris, sistem tersebut dapat membantu meringankan dan mempercepat proses pengadaan barang dari program Daftar Isian Pelaksanaan Anggaran (DIPA) serta dapat memajemen *stock opname*, peminjaman dan pengembalian aset, *maintenance* aset, dan proses permintaan *consumable*.

5. Dalam penelitian yang diteliti oleh Ramadhan & Andah (2018), peneliti menemukan masalah bahwa tidak adanya pendataan aset yang dimiliki, proses peminjaman yang tidak praktis dan barang peminjaman lupa disiapkan. Hasil penelitian tersebut adalah dengan merancang sebuah sistem peminjaman aset berbasis web. Kesimpulan dari penelitian adalah dengan menggunakan peminjaman aset, sistem dapat mempermudah pendataan peminjaman aset dan sistem dapat mencetak laporan permintaan supaya petugas bisa mendapat informasi akurat mengenai barang yang harus disiapkan.

6. Dalam penelitian yang diteliti oleh Yuniva & Syafi'i (2018), peneliti menemukan masalah bahwa proses pendaftaran siswa masih dilakukan secara manual, dimana formulir pendaftaran siswa dan penyusunan laporannya masih menggunakan buku manual untuk menyimpan data-data siswa. Hasil penelitian adalah dengan merancang sebuah sistem berbasis web untuk menerima siswa baru. Kesimpulan penelitian adalah dengan adanya sistem berbasis web tersebut, pihak sekolah dapat mendapatkan kemudahan dalam menerima siswa baru beserta pembuatan laporan siswa.

Tabel 2.1 Tinjauan Pustaka

No.	Nama Peneliti	Tahun	Judul	Kesimpulan
1	Sri Mulyati, dkk	2015	Rancang Bangun Sistem Informasi Transaksi Peminjaman Buku Berbasis Web Online Pada Perpustakaan Universitas Muhammadiyah Tangerang	Penggunaan metode waterfall dapat membantu penulis dalam merancang sebuah sistem berbasis web.
2	Ach Abdul Rhouf, dkk	2017	Aplikasi Penjualan Alat Tulis Kantor (ATK) berbasis Web Mobile JQuery Pada Estu Gresik	JQuery memberi kemudahan dalam merancang sebuah website.
3	Yulrio Brianorma n dan Barry Ceasar Octariadi	2017	Perancangan Sistem Pengelolaan Ruang Berbasis Web	Validasi pada sistem berbasis web yang benar dapat mengurangi duplikasi data yang sama.
4	Alief Maulana, dkk	2018	Implementasi Sistem Informasi Manajemen Inventaris Berbasis Web Di Pusat	Alur kerja sistem berbasis web yang teratur dapat meningkatkan kecepatan pekerjaan.

			Teknologi Informasi Dan Komunikasi – BPPT	
5	Muhamad Ihpaz Ramadhan dan Bullion Dragon	2018	Implementasi Sistem Informasi Peminjaman Aset Kampus Berbasis Web Pada Direktorat Administrasi Umum Universitas Budi Luhur Dengan Metodologi Berorientasi Obyek	Pendataan peminjaman diperlukan untuk menghasilkan laporan peminjaman yang akurat.
6	Ika Yuniva dan Ahmad Syafi’I	2018	Pendekatan Model Waterfall Dalam Perancangan Web Sistem Informasi Penerimaan Siswa Baru Menggunakan Framework Bootstrap	Tampilan web yang bagus dapat mempermudah <i>user</i> dalam mendapatkan informasi dengan cepat.

## 2.2. Landasan Teori

### 2.2.1. Metode Pengembangan SDLC Waterfall

*Software Development Life Cycle* atau disingkat sebagai SDLC adalah sebuah rangkaian tugas yang berjalan menurut tahap-tahap penerapan sebuah aplikasi ataupun sebuah sistem komputer (Fatmawatie, Adittyta, & Sardi, 2016).

Metode SDLC yang digunakan oleh penulis adalah metode waterfall. Metode tersebut dibagikan menjadi 6 tahap, yakni:

1. *Requirement Analysis*, pada tahap ini syarat-syarat, fitur dan fungsionalitas sistem akan di pelajari.
2. *System Design*, pada tahap ini perancangan tampilan sistem secara garis besar akan dimulai. Tahap ini juga memutuskan spesifikasi minimal perangkat keras beserta perangkat lunak yang diperlukan.

3. *Implementation*, pada tahap ini perancangan sistem akan dimulai dengan menggunakan hasil desain sistem pada tahap sebelumnya sebagai basis perancangan.
4. *Testing*, pada tahap ini pengujian sistem yang telah dirancang akan dimulai. Ketika ada fitur sistem yang masih berbeda dengan hasil perencanaan sebelumnya, maka sistem tersebut perlu merancang kembali sistem tersebut pada tahap perancangan sistem.
5. *Deployment*, pada tahap ini sistem yang sudah dirancang dan telah melewati hasil pengujian sistem akan di implementasi ke sisi klien.
6. *Maintenance*, pada tahap ini juga digunakan untuk menambah fitur atau fungsionalitas sistem.

### 2.2.2. Sistem Informasi

Sistem informasi merupakan himpunan ataupun grup dari komponen atau elemen yang bergantung pada satu sama lainnya demi mewujudkan sebuah tujuan yang telah ditentukan. Sebuah sistem dibangun dengan bagian yang saling terkait dan berkerja bersama demi mencapai sasam dan maksud. Maka dari itu, sebuah sistem tidak disusun oleh unsur-unsur secara berantakan, tetapi unsur-unsur tersebut disusun oleh secara teratur dan saling melengkapi untuk mencapai tujuannya tersebut (Susanti, 2016).

### 2.2.3. Browser

Browser merupakan sebuah *software* komputer maupun smartphoneyang memiliki fungsi untuk mengakses informasi-informasi pada halaman web.



Browser merupakan *software* yang digunakan untuk melakukan interaksi beserta memperlihatkan dokumen-dokumen yang telah disediakan oleh sebuah situs web (Hastanti, Purnama, & Wardati, 2015).

#### 2.2.4. Website

*Website* ataupun juga dikenal sebagai situs web atau web adalah sebuah kumpulan halaman-halaman yang saling bersangkutan dan bisa diakses dengan mengunjungi halaman beranda dari situs web tersebut dengan memaai *browser*.

Website dapat didefinisi sebagai gabungan dari berbagai tampilan halaman yang berfugis untuk menampilkan informasi-informasi seperti tulisan, gambaran, animasi bergerak maupun suara yang membentuk satu rangkai bangunan yang bersangkutan satu sama lainnya (Manuho, Rindengan, & Sinsuw, 2018).

Untuk menjalankan sebuah website, diperlukan berbagai komponen khusus seperti bahasa markah, server dan lain-lainnya yang dapat dilihat sebagai berikut:

##### 1. HTML5

*Hyper Text Markup Language* atau lebih dikenal sebagai HTML adalah bahasa markah yang berguna untuk merancang situs web yang relatif mudah untuk dipelajari. Pengembangan HTML5 bertujuan untuk meningkatkan teknologi HTML supaya dapat lebih mudah dibaca manusia, mendukung teknologi multimedia terbaru, dan juga mudah dimengerti oleh mesin (Mubarak & Hermawaty, 2017).

HTML merupakan bahasa markup yang berisi *file* teks murni yang berfungsi untuk merancang sebuah *website*. HTML menampilkan informasi di browser web dan dalam format *hypertext* sederhana yang disimpan dalam format ASCII sehingga

menghasilkan wujud tampilan terintegrasi. HTML5 merupakan perkembangan/versi dari HTML yang terbaru.

Fitur-fitur yang diperkenalkan HTML5 antara lain elemen semantik baru, yakni pada elemen *header*, *footer*, dan elemen *section*. Selain itu, fitur *WebSocket* sebagai teknologi komunikasi aplikasi web dua arah. Fitur Audio dan Video secara *embedded* tanpa harus menggunakan *plugin* tambahan. Fitur *Canvas* yang dapat digunakan untuk menggambar grafik dua dimensi. Fitur *Forms 2.0* dengan pengenalan atribut baru yaitu tag `<input>`. Fitur *Geolocation* yang memungkinkan pengunjung untuk membagikan data lokasi dimana mereka berada, dan masih banyak fitur lainnya seperti fitur *Persistent Local Storage*, *Drag and Drop*, *Server-Sent Event (SSE)*, dll (Mangkey, Rindengan, & Tulenan, 2016).

## 2. CSS3

*Cascading Style Sheet* atau lebih dikenal sebagai CSS merupakan aturan yang mengatur komponen-komponen seperti ukuran gambar, warna tulisan, ukuran border pada tabel dan lain-lainnya pada sebuah *website* sehingga *website* tersebut akan terlihat lebih seragam dan terstruktur.

CSS3 merupakan penambahan dari CSS2 sehingga pada saat pemindahan CSS2 ke CSS3, perubahan apapun tidak diperlukan pada *website* dikarenakan CSS3 mendukung seutuhnya CSS2. CSS3 merupakan versi CSS terbaru (Mangkey et al., 2016).

CSS3 dikembangkan oleh W3C dan pada saat ini merupakan bahasa umum dalam perancangan web. CSS berfungsi sebagai pembantu dan pelengkap html yang

memiliki peran penataan kerangka dari web yang dirancang. CSS bersifat *multi-platform*, ini mengartikan bahwa CSS dapat berjalan pada berbagai macam web browser dan sistem operasi (Chulkamdi & Purnomo, 2016).

### 3. Javascript

Javascript merupakan bahasa pemrograman pada situs web yang dipakai untuk merancang dan mengendalikan situs web dinamis seperti animasi, foto slideshow, bentuk interaktif dan lain-lainnya pada sisi klien. Javascript menyediakan *library* yang dapat digunakan untuk mempermudah pemrograman, seperti sheetJS, QR Lib, dll.

Secara fungsional, Javascript diguna untuk memberi objek yang telah dibenamkan dapat mengakses script. Contohnya adalah pada saat pemakaian javascript untuk membuka halaman baru dengan fitur pop up, fitur validasi data pada sebuah form sebelum data dikirim ke *server*, mengubah gambar kursor mouse pada saat melewati objek tertentu pada situs web dan lain-lainnya (Chulkamdi & Purnomo, 2016).

### 4. JQuery

JQuery merupakan pustaka atau kumpulan-kumpulan dari kode javascript yang telah dirancang untuk mempermudah pengontrolan situs web. JQuery dapat diartikan sebagai *framework*, yakni library dari Javascript yang menekankan pada interaksi HTML dan Javascript. Kelebihan dari JQuery adalah pemudahan dan penyederhanaan dalam penulisan kode program untuk menghasilkan fungsi dan interaksi pada program.

JQuery adalah pustaka atau kumpulan-kumpulan dari kode-kode javascript yang sudah siap digunakan (Pratama & Darwati, 2017).



JQuery bersifat *open-source* dan memiliki ukuran yang sangat kecil, berkisaran hanya 20-an kb dan terdiri dari satu file. Walaupun memiliki ukuran yang minim,

JQuery sangat handal dalam membantu proses pemograman situs web. Untuk mendapatkan fungsi dan fitur JQuery yang lebih banyak, *developer* dapat menambahkan dan men-*download* fungsionalitas lebih. Saat ini terdapat ribuan file

JQuery gratis yang bisa diunduh dari internet (Chulkamdi & Purnomo, 2016).

## 5. Bootstrap

Bootstrap merupakan framework yang bersifat *open-source* terpopuler dalam perancangan tampilan website yang bersifat responsif. Bootstrap berisi kode-kode yang telah lengkap dengan desian-desain seperti tombol, bentuk, tipografi dan lain-lainnya dalam bentuk file HTML dan CSS, sehingga perancangan dan pengembangan situs web menjadi lebih mudah dan ringan.

Bootstrap merupakan sebuah *platform* yang dibangun oleh tim twitter. Bootstrap bertampil untuk pertama kalinya pada ajang hackweek dan sekarang penyempurnaan bootstrap sudah mulai. Untuk menggunakan *platform* bootstrap ini hanya memerlukan koding CSS dan javascript yang sedikit tetapi tetap bisa menghasilkan sebuah situs web yang dapat mengikuti perkembangan browser. Penggunaan bootstrap juga membuat tampilan website menjadi nyaman, cepat dan responsif (Saras, 2018).

## 6. Hypertext Preprocessor (PHP)

*Hypertext Preprocessor* atau lebih dikenal sebagai PHP adalah bahasa pemrograman pada bagian *server* situs web yang banyak dipakai dalam mengembang sebuah sistem berbasis web.

Kode (*Script*) PHP bersifat *server-side* yang dirancang eksklusif guna merancang sebuah situs web. Pada situs web tersebut, dapat ditenamkan (*embedded*) kode-kode PHP. Kode tersebut tidak dieksekusi pada sisi komputer klien tetapi kode tersebut akan dieksekusi pada sisi server. Hasil yang ditampilkan setelah eksekusi kode PHP pada sisi server adalah kode HTML (Ilmi, Said, & Hadi, 2016).

Selain karakteristik PHP yang bersifat *server-side*, PHP merupakan bahasa script yang bersifat *open source*. Kelebihan PHP yang lain yaitu mempunyai banyak referensi untuk dipelajari sehingga sangat mudah untuk dipahami bagi developer. PHP mendukung *operating system* Windows, Unix, Linux, dan juga dapat berjalan pada *console* secara *runtime* (Susanti, 2016).

## 7. XAMPP

XAMPP merupakan paket instalasi yang berisi 4 aplikasi yaitu MySQL, Apache, Perl dan PHP. Developer dapat membangun sebuah server pada sisi local untuk perancangan aplikasi berbasis web mereka dengan menggunakan yang namanya *localhost*. XAMPP adalah perangkat lunak (*software*) *open source* yang berfungsi sebagai web server lokal. XAMPP mendukung banyak sistem operasi, seperti sistem operasi Linux, Windows, Machintosh dan juga sistem operasi lainnya (Manuho et al., 2018).

### 2.2.5. QR Code

*QR code* adalah sebuah gambar yang mengisi matriks yang bersifat dua dimensi yang dikembangkan dari kode batang (*barcode*) dan memiliki kapabilitas untuk menyimpan data (Adiguna & Gunawan, 2016).

*QR Code* mempunyai kapasitas dalam data pengkodean yang tinggi dikarenakan semua jenis data seperti angka, abjad, simbol, kode biner beserta tulisan-tulisan seperti *kanji*, *kana* dan lain-lainnya dapat disimpan oleh *QR Code*. Penampungan data pada *QR Code* ditampung secara horizontal dan vertical menyebabkan tampilan pada *QR Code* lebih kecil dibandingkan dengan tampilan pada *barcode*. *QR Code* juga mampu memperbaiki kesalahan sebanyak 30%, hal ini menyebabkan *QR Code* lebih tahan pada kerusakan. Dikarenakan itu, pada saat sebagian simbol pada sebuah *QR Code* terjadi kerusakan, data yang disimpan pada *QR Code* tersebut tetap bisa terbaca.

### 2.2.6. Database

Database merupakan sebuah kumpulan-kumpulan informasi yang telah disusun sedemikian rupa dengan ketentuan dan aturan, sehingga dapat dikontrol dengan sebuah program atau aplikasi untuk mendapatkan informasi pada database tersebut.

Database merupakan sebuah bagian penting dalam sistem informasi dikarenakan database adalah tempat penyimpanan data dimana data tersebut akan diolah selanjutnya menjadi informasi yang akurat (Majid & Subandi, 2018).

*My Structured Query Language* atau lebih dikenal dengan nama MySQL merupakan sejenis basis data pada server yang digunakan untuk merancang sebuah

aplikasi yang membutuhkan penggunaan *database*. MySQL merupakan turunan dari konsep dalam mengolah *database* seperti penambahan, penghapusan dan penyeleksian data yang dikenal sebagai SQL (*Structured Query Language*) (Fatmawatie et al., 2016).

### 2.2.7. *Unified Modeling Language (UML)*

UML ataupun juga dikenal sebagai *Unified Modeling Language* adalah sebuah bahasa dalam pemodelan standar orientasi objek yang dipakai untuk memodel dan merancang sebuah sistem. UML adalah dasar dalam perancangan dan visualisasi sebuah sistem informasi.

Tujuan dalam pengaplikasian UML selain menjadi sebagai bentuk dasar perancangan sebuah *software*, pengaplikasian UML juga bertujuan untuk merincikan sistem untuk analisa kebutuhan sistem dan sebagai penghubung software dengan proses bisnis (Sonatha, Azmi, Suryani, & Sari, 2017).

Untuk merancang sebuah sistem, UML memiliki beberapa *diagram* yang digunakan, yakni:

#### 1. Use Case Diagram

*Use case diagram* merupakan sebuah gambaran model yang digunakan untuk menggambar fitur-fitur ataupun fungsionalitas yang diinginkan dalam sebuah sistem.


Gambaran *use case* mewakili interaksi dari seorang atau sekelompok aktor dengan sistem (Rori, Sentinuwo, & Karouw, 2016).

Simbol yang dipakai untuk merancang sebuah *use case* dapat ditinjau pada tabel 2.2 dibawah ini.

Tabel 2.2 *Use Case Diagram*

Deskripsi	Simbol
<p><b>Actor</b></p> <p><i>Actor</i> adalah seorang atau sekelompok orang yang telah diberi sebuah peran atau lebih dalam berkomunikasi dengan sistem.</p>	 <p>Actor</p>
<p><b>Use Case</b></p> <p><i>Use case</i> menggambarkan fitur sistem yang dapat diakses oleh actor.</p>	 <p>Use Case</p>
<p><b>Association Relationship</b></p> <p><i>Association relationship</i> menggambarkan hubungan antar <i>actor</i> dengan <i>use case</i>.</p>	
<p><b>Include Relationship</b></p> <p><i>Include relationship</i> menggambarkan use case yang membutuhkan proses pada use case lain.</p>	<p>&lt;&lt;include&gt;&gt;</p> 
<p><b>Extend Relationship</b></p> <p><i>Extend relationship</i> menggambarkan lanjutan dari <i>use case</i> sebelumnya ketika memenuhi syarat tertentu.</p>	<p>&lt;&lt;extend&gt;&gt;</p> 



<p><b>Boundary</b></p> <p><i>Boundary</i> menentukan batasnya dari sebuah sistem.</p>	
---	--

## 2. Entity Relationship Diagram (ERD)

*Entity relationship diagram* atau lebih dikenal sebagai ERD adalah model yang menggambarkan relasi data dengan data lain dalam sebuah *database* yang didasari pada objek data yang bersangkutan dengan data lainnya. Kegunaan ERD adalah untuk menginterpretasikan hubungan/relasi antar data dalam *database* kepada pengguna (Fatmawatie et al., 2016).

## 3. Flowchart





*Flowchart* adalah langkah-langkah pemecahan masalah dengan mewakili simbol-simbol standar tertentu yang mudah dipahami dan mudah digunakan.

*Flowchart* bertujuan untuk menggambarkan sebuah tahap penyelesaian masalah secara teratur, rapi dan sederhana dengan menggunakan simbol-simbol standar yang dimengerti oleh *programmer* (Syamsiah, 2019).

Simbol yang dipakai untuk merancang sebuah *flowchart* dapat ditinjau pada tabel 2.3 dibawah ini.

Tabel 2.3 *Flowchart*

Deskripsi	Simbol
<b><i>Terminator</i></b>	

<p><i>Terminator</i> menggambarkan mulai dan akhirnya <i>flowchart</i>.</p>	
<p><b>Process</b></p> <p><i>Process</i> menggambarkan sebuah proses, aktivitas ataupun fitur yang dilakukan oleh sistem.</p>	
<p><b>Flowline</b></p> <p><i>Flowline</i> menggambarkan arah atau alur kerja sebuah proses.</p>	
<p><b>Input/Output(I/O) Data</b></p> <p><i>I/O Data</i> menggambarkan proses input output sebuah sistem.</p>	
<p><b>Decision</b></p> <p><i>Decision</i> menggambarkan proses terjadinya keputusan berdasarkan kondisi yang telah ditentukan.</p>	




#### 4. Sequence Diagram

*Sequence diagram* adalah gambaran kegiatan dari objek-objek yang dimiliki use case dengan menguraikan pengiriman dan penerimaan pesan antara objek. Oleh sebab itu, penggambaran *sequence diagram* perlu diketahui objek-objek yang berhubungan dan metode kelas yang dimiliki dijadikan objek (Haryana, Rochman, & Setyaningsih, 2017).

Simbol yang dipakai untuk merancang sebuah *sequence diagram* dapat ditinjau pada tabel 2.4 dibawah ini.

Tabel 2.4 Sequence Diagram

Deskripsi	Simbol
<p><b>Actor</b></p> <p><i>Actor</i> adalah seorang atau sekelompok orang yang memiliki satu peran atau lebih pada saat berinteraksi dengan sistem.</p>	
<p><b>Lifeline</b></p> <p><i>Lifeline</i> menggambarkan sebuah individu atau peserta dalam berinteraksi.</p>	
<p><b>Active</b></p> <p><i>Active</i> menggambarkan suatu elemen pada saat menjalankan sebuah proses.</p>	
<p><b>Call Message</b></p> <p><i>Call message</i> menggambarkan komunikasi antar <i>lifeline</i> dimana sebuah permintaan dikirim kepada <i>lifeline</i> lain.</p>	
<p><b>Return Message</b></p>	


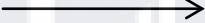


<p><i>Return message</i> menggambarkan komunikasi antar lifeline dimana hasil pembalasan dari permintaan dikirim ke lifeline sebelumnya.</p>	
<p><i>Self Message</i></p> <p><i>Self Message</i> menggambarkan komunikasi dengan lifeline sendiri.</p>	
<p><i>Alternative</i></p> <p><i>Alternative</i> menggambarkan proses pada sequence diagram yang digunakan ketika suatu keputusan perlu dilakukan.</p>	

## 5. Activity Diagram

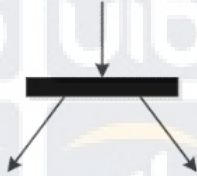


*Activity diagram* merupakan sebuah gambaran dari alur kerja fungsionalitas atau fitur sebuah sistem. Pada saat pemodelan, penggunaan *activity diagram* adalah untuk menggambarkan alur kerja sistem maupun alur kerja suatu kejadian (Sudiarjo, Dzulhaq, & Gusti, 2016).

Simbol yang dipakai untuk merancang sebuah *activity diagram* dapat ditinjau pada tabel 2.5 dibawah ini.

Tabel 2.5 *Activity Diagram*

Deskripsi	Simbol
<p><b>Activity</b></p> <p><i>Activity</i> menggambarkan aktivitas atau kejadian pada <i>activity diagram</i>.</p>	
<p><b>Association</b></p> <p><i>Association</i> digunakan untuk menghubungkan aktivitas sebelumnya dengan aktivitas selanjutnya.</p>	
<p><b>Initial State</b></p> <p><i>Initial state</i> menggambarkan awal mulainya aktivitas pada <i>activity diagram</i>.</p>	
<p><b>Final State</b></p> <p><i>Final state</i> menggambarkan akhirnya sebuah aktivitas pada <i>activity diagram</i>.</p>	



<p><b><i>Synchronization Fork</i></b></p> <p><i>Synchronization fork</i> memisah aktivitas menjadi beberapa alur.</p>	
<p><b><i>Synchronization Join</i></b></p> <p><i>Synchronization join</i> menggabungkan kembali aktivitas yang terpisah.</p>	
<p><b><i>Decision</i></b></p> <p><i>Decision</i> menggambarkan proses terjadinya keputusan berdasarkan kondisi yang telah ditentukan.</p>	
<p><b><i>Swimlane</i></b></p> <p><i>Swimlane</i> menggambarkan batas dimana alur aktivitas digambarkan.</p>	