

## **BAB II**

### **LANDASAN TEORI**

#### **A. Dasar-Dasar Teori**

##### **1. Sistem Informasi**

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi organisasi yang bersifat manajerial dalam kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan (Kristanto, 2018).

##### **2. Penggalangan Dana**

Menurut Mulyani & Kurniadi (2020) penggalangan dana adalah suatu kegiatan sosial untuk membantu sesama yang kemudian disalurkan untuk membantu fakir miskin di jalanan dan anak yatim.

##### **3. Metode *Crowdfunding***

*Crowdfunding* merupakan skema pendanaan yang mengumpulkan dana dari masyarakat besar (*crowd*) berbasis internet. Jumlah pendanaan yang dikenakan pada masyarakat seringkali dalam nominal yang relatif kecil. Namun karena jumlah masyarakat yang berpartisipasi cukup besar, maka dana yang dikumpulkan dapat menjadi sangat besar.

*crowdfunding* sebagai sebuah metode untuk menghubungkan antara entrepreneur yang menginginkan peningkatan modal dan investor baru yang memiliki sumber daya modal dan ingin berinvestasi dalam jumlah yang kecil melalui entitas intermediari berbasis internet (Vulkan, 2019).

*Crowdfunding* dapat dibagi menjadi empat kategori sesuai basis pendanaan yang ditawarkan oleh *platform crowdfunding*. Keempat kategori tersebut adalah donasi, *pre-selling*, *lending*, dan *equity crowdfunding* (Vulkan, 2019).

Terdapat tiga pihak utama yang menjadi pemegang kepentingan (*stakeholders*) yaitu entrepreneur, *platform crowdfunding*, dan investor (*backers*). Para pemegang kepentingan utama tersebut memiliki peran dan kepentingan

masing-masing. Alur pertama dimulai dari entrepreneur (bisnis atau *startups*) mengajukan ide, permintaan pendanaan melalui *platform crowdfunding* lalu menjanjikan *return* pada investor. *Backers* (investor) akan melihat peluang investasi yang ditawarkan oleh entrepreneur lalu memberikan komitmennya untuk mendanai ataupun memberi saran. *Platform crowdfunding* berperan sebagai lembaga perantara yang mempertemukan antara investor dan *backers*. Berikut ini merupakan penjelasan mengenai tipe *crowdfunding* berdasarkan Massolution yang telah dipublikasikan kedalam laporan industri tahun 2013 :

1. *Equity-based crowdfunding*

Donatur sebagai penyandang dana mengharapkan kompensasi dalam bentuk ekuitas atau pendapatan atau pengaturan saham dari hasil proyek penggalangan dana tersebut.

2. *Lending-based crowdfunding*

Donatur sebagai penyandang dana menerima kompensasi secara berkala (bunga) dan mengharapkan pembayaran kembali dari dana yang telah diberikan setelah proyek berhasil.

3. *Reward-based crowdfunding*

Donatur sebagai penyandang dana memberikan uang untuk mendapatkan keuntungan atau kompensasi selain uang.

4. *Donation-based crowdfunding*

Donatur sebagai penyandang dana tidak mengharapkan kompensasi dari pemilik proyek.

#### 4. **Android**

Android adalah sistem operasi untuk telpon seluler berbasis Linux sebagai karnelnya. Android menyediakan *platform* terbuka (*open source*) bagi *developer* untuk menciptakan aplikasinya sendiri. Android Inc. adalah pendatang baru yang bergerak dalam pembuatan piranti lunak untuk ponsel. Android Inc didirikan oleh (Jumri, 2014)

Android adalah *mobile Operating System* (OS) yang dikembangkan oleh Google. OS Android berbasis pada OS Linux Kernel. Android bersifat open source, artinya pengembang bisa memodifikasi dan menyesuaikan OS untuk setiap ponsel. Android menyediakan *platform* terbuka bagi para pengembang

untuk menciptakan aplikasi mereka sendiri untuk di gunakan oleh bermacam-macam kegunaan. Android beberapa kali melakukan pembaruan versinya, kebanyakan nama di setiap versinya adalah nama makanan.



**Gambar 2. 1 Logo Android**

Android memiliki empat karakteristik sebagai berikut :

1. Terbuka

Android dibangun untuk benar-benar terbuka sehingga sebuah aplikasi dapat memanggil salah satu fungsi inti ponsel seperti membuat panggilan, mengirim pesan teks, menggunakan kamera dan lain-lain. Android merupakan sebuah mesin virtual yang dirancang khusus untuk mengoptimalkan sumber daya memori dan perangkat keras yang terdapat di dalam perangkat. Android merupakan *open source*, dapat secara bebas diperluas untuk memasukkan teknologi baru yang lebih maju pada saat teknologi tersebut. *Platform* ini akan terus berkembang untuk membangun aplikasi *mobile* yang inovatif.

2. Semua aplikasi dibuat sama

Android tidak memberikan perbedaan terhadap aplikasi utama dari telepon dan aplikasi pihak ketiga (*third-party application*). Semua aplikasi dapat dibangun untuk memiliki akses yang sama terhadap kemampuan sebuah telepon dalam menyediakan layanan dan aplikasi yang luas terhadap para pengguna.

### 3. Memecahkan hambatan pada aplikasi

Android memecah hambatan untuk membangun aplikasi yang baru dan inovatif. Misalnya, pengembang dapat menggabungkan informasi yang diperoleh dari web dengan data pada ponsel seseorang seperti kontak pengguna, kalender atau lokasi geografis.

### 4. Pengembangan aplikasi yang cepat dan mudah

Android menyediakan akses yang sangat luas kepada pengguna untuk menggunakan aplikasi yang semakin baik. Android memiliki sekumpulan tools yang dapat digunakan sehingga membantu para pengembang dalam meningkatkan produktivitas pada saat membangun aplikasi yang dibuat.

Versi Android diawali dengan dirilisnya Android beta pada bulan November 2007. Komersial pertama, Android 1.0, dirilis pada September 2008. Sejak April 2009, versi Android dikembangkan dengan nama kode yang dinamai berdasarkan makanan pencuci mulut dan makanan manis. Masing-masing versi dirilis sesuai urutan alfabet, yaitu :

1. Cupcake (1.5)
2. Donut (1.6)
3. Eclair (2.0–2.1)
4. Froyo (2.2–2.2.3)
5. Gingerbread (2.3–2.3.7)
6. Honeycomb (3.0–3.2.6)
7. Ice Cream Sandwich (4.0–4.0.4)
8. Jelly Bean (4.1–4.3)
9. KitKat (4.4+)
10. Lollipop (5.0-5.1)
11. Marshmallow (6.0)
12. Nougat (7.0)
13. Oreo (8.0)
14. Pie (9.0)
15. Android 10.

Berdasarkan pembahasan mengenai android dan definisi pengertian Android menurut para ahli di atas, dapat kita simpulkan bahwa **Android** adalah nama sistem operasi yang digunakan pada banyak smartphone dan tablet. Dimiliki dan dikelola oleh Google. Diantara masing-masing versi android penulis akan menggunakan versi Ice Cream Sandwich (4.0), karena pada versi ini lebih ringan untuk digunakan pada komputer .

Dirilis pada tanggal 19 Oktober 2011, versi ini memiliki fitur terbaru seperti terintegrasinya kontak jaringan sosial, pembukaan kunci dengan fitur pengenalan wajah penggunanya, mencari email secara *offline*, jaringan data control dan pemantauan penggunaan, fitur fotografi, dan NFC yang digunakan untuk berbagi informasi.



**Gambar 2. 2 Logo Android 4.0 Ice Cream Sandwich**

## **5. Native Andorid**

Menueut Sheldon and Moes (2015) Native Android adalah aplikasi *mobile* khusus yang hanya dapat digunakan di perangkat tertentu baik android maupun IOS, dikarenakan aplikasi native android bekerja pada sistem yang berbedasa serta bahasa pemograman juga berbeda.

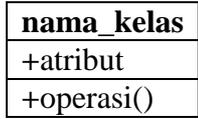
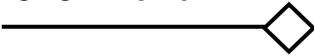
## **6. Perancangan Sistem UML (*Unified Modeling Language*)**

UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Rosa & Shalahudin, 2018)

## 1. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Simbol-simbol yang ada pada diagram kelas pada **Tabel 2.1** di bawah ini:

**Tabel 2. 1** Simbol *Class Diagram*

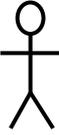
Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka/ <i>Interface</i>  <b>nama_interface</b>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>asociation</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan/ <i>dependecy</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi/ <i>agregation</i> 	Relasi antar kelas dengan makna semua bagian ( <i>whole-part</i> )

## 2. *Use Case Diagram*

Menurut Rosa dan Shalahuddin (2018) *Use case diagram* atau diagram *use case* merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih

aktor dengan sistem informasi yang akan dibuat. Simbol-simbol yang ada pada diagram *use case* dapat dilihat pada **Gambar 2.2** di bawah ini

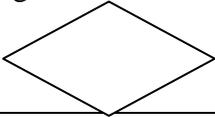
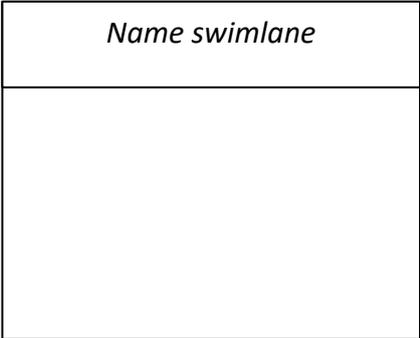
**Tabel 2. 2** Simbol diagram *use case*

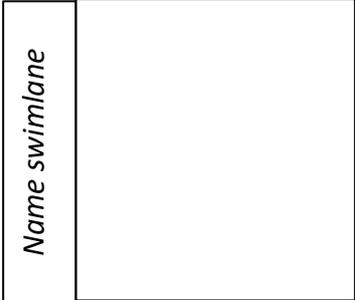
Simbol	Deskripsi
<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i></p>
<p>Aktor/<i>actor</i></p> 	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase</p>
<p>Asosiasi/<i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>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 sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan</p>
<p>&lt;&lt;<i>extend</i>&gt;&gt;</p> 	<p>dapat berdiri sendiri walau 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</p>
<p>Generalisasi/<i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>
<p>Menggunakan/<i>Include/uses</i> &lt;&lt;<i>include</i>&gt;&gt;</p> 	<p>Relasi <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 fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p>

### 3. Activity Diagram

Diagram Aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak, diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor yaitu aktivitas yang dapat dilakukan oleh sistem. simbol-simbol yang ada pada *sequence* digram dapat dilihat pada **Tabel 2.3** di bawah ini:

**Tabel 2. 3** Simbol *Activity Diagram*

Simbol	Deskripsi
Status Awal 	Status awal aktivitas sistem sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan oleh 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> 	Swimlane adalah elemen visual yang digunakan dalam <b>diagram</b> alir proses yang menggambarkan siapa bekerja pada subset tertentu dari sebuah proses. <b>Swimlane</b> tersebut diatur baik horisontal maupun vertikal dan digunakan untuk mengelompokkan sub-proses sesuai dengan tanggung jawab mereka.

Simbol	Deskripsi
<p style="text-align: center;">Atau</p> 	

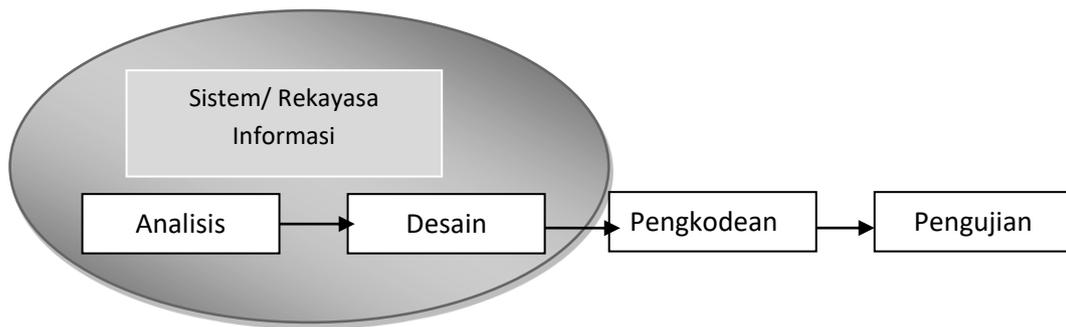
### B. Pengertian *MySQL*

Gilmore (2015) *MySQL* adalah sebuah relasional *database server* yang menawarkan berbagai macam mekanisme dalam pengolahan data yang dikenal dengan *storage engines*.

Sheldon and Moes (2015) *MySQL* sama seperti sistem *DBMS lainnya*, seperti *Oracle*, *DB2*, dan *SQL Server*. Memungkinkan untuk mengakses, memanipulasi, melindungi, dan memelihara metadata yang diperlukan untuk mendefinisikan data yang disimpan.

### C. Merode Pengembangan *Waterfall*

Menurut Rosa & Shalahudin (2018) “*Waterfall* merupakan metodologi pengembangan sistem yang menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut di mulai dari analisis, desain, pengodean, pengujian dan tahap pendukung. *Waterfall* merupakan metodologi klasik yang digunakan untuk mengembangkan, memelihara, dan menggunakan sistem informasi (Simatupang, Yanris and Sugiyarti, 2020). Berdasarkan pengertian menurut para ahli maka, dapat disimpulkan *waterfall* adalah metodologi klasik yang digunakan untuk menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut di mulai dari analisis, desain, pengodean, pengujian dan tahap pendukung. Siklus pengembangan sistem menggunakan model *waterfall* sebagai berikut :



**Gambar 2. 3 Ilustrasi model *Waterfall***

**Sumber :** (Rosa and Shalahudin, 2018)

Menurut (Rosa and Shalahudin, 2018) terdapat tahapan dalam proses *waterfall* yaitu:

1. Analisis Kebutuhan Perangkat Lunak, yaitu proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak seperti apa yang dibutuhkan *user*.
2. Desain Perangkat Lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean.
3. Pengkodean adalah proses implemetasi pembuatan sistem dengan bahasa pemograman yang digunakan yang akan dilakukan pada pembuatan program perangkat lunak
4. Pengujian adalah tahapan uji coba aplikasi yang dibangun untuk menentukan layak atau tidak nya aplikasi digunakan.

### 1. Kelemahan *Waterfall*

Menurut Rosa & Shalahudin (2018) model air terjun dapat dilakukan sesuai alurnya karena sebab berikut:

- a. Perubahan spesifikasi perangkat lunak terjadi di tengah alur pengembangan.
- b. Sangat sulit bagi pelanggan untuk mendefinisikan semua spesifikasi di awal alur pengembangan.
- c. Pelanggan tidak mungkin bersabar mengakomodasi perubahan yang diperlukan di akhir alur pengembangan.

## 2. **Kelebihan *Waterfall***

Menurut Rosa & Shalahudin (2018) “Hal positif dari model air terjun adalah struktur tahap pengembangan sistem jelas, dokumentasi dihasilkan di setiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap).”

## D. **Pengujian *ISO 25010***

ISO/IEC 25010 merupakan model kualitas sistem dan perangkat lunak yang menggantikan ISO/IEC 9126 tentang software engineering (Iqbal, 2019). Product quality ini juga digunakan untuk tiga model kualitas yang berbeda untuk produk perangkat lunak antara lain:

1. Kualitas dalam model penggunaan
2. Model kualitas produk
3. Data model kualitas

Adapun dimensi yang pertama terdapat beberapa faktor elemen diantaranya :

- 1) *Functionality* (Fungsionalitas). Kemampuan perangkat lunak untuk Merupakan tingkatan dimana perangkat lunak dapat menyediakan fungsionalitas yang dibutuhkan ketika perangkat lunak digunakan pada kondisi spesifik tertentu dalam hal ini perangkat lunak dapat memenuhi kelayakan dari sebuah fungsi untuk melakukan pekerjaan yang spesifik bagi pengguna dan dapat memberikan hasil yang tepat dan ketelitian terhadap tingkat kebutuhan pengguna.
- 2) *Reliability* Merupakan tingkatan dimana perangkat lunak dapat bertahan pada tingkatan tertentu ketika digunakan oleh pengguna pada kondisi yang spesifik dalam hal ini perangkat lunak dapat beroperasi dan siap ketika dibutuhkan untuk digunakan dan juga dapat bertahan pada tingkat kemampuan tertentu terhadap kegagalan, kesalahan serta perangkat lunak kembali pada tingkat tertentu dalam mengembalikan pengembalian data yang disebabkan kegagalan atau kesalahan pada perangkat lunak.

- 3) *Performance efficiency* Merupakan tingkatan dimana perangkat lunak dapat memberikan kinerja terhadap sejumlah sumber daya yang digunakan pada kondisi tertentu dalam hal ini *performance efficiency* dapat memberikan reaksi dan waktu yang dibutuhkan ketika melakukan aksi dari sebuah fungsi dan perangkat lunak dapat menggunakan sejumlah sumber daya ketika melakukan aksi dari sebuah fungsi.
- 4) *Usability* Perangkat lunak dapat dimengerti, dipelajari, digunakan, dan menarik pengguna ketika digunakan dalam hal ini perangkat lunak mudah dipelajari oleh pengguna, perangkat lunak dapat digunakan dan dioperasikan oleh pengguna.
- 5) *Security* Merupakan perlindungan terhadap perangkat lunak dari berbagai ancaman atau keganjalan dalam hal ini perangkat lunak memiliki perlindungan terhadap data atau informasi dari pengguna dan merupakan dari kelengkapan, ketepatan dari sejumlah *asset* yang telah dijaga sehingga aksi atau tindakan yang dilakukan telah terbukti dan hal tersebut tidak dapat ditolak.
- 6) *Compability* Faktor ini merupakan kemampuan dari dua atau lebih komponen perangkat lunak dapat melakukan pertukaran informasi dan melakukan fungsi yang dibutuhkan ketika digunakan pada *hardware* atau lingkungan perangkat lunak yang sama.
- 7) *Maintainability* Merupakan tingkat dimana sebuah perangkat lunak dapat dimodifikasi. Dalam hal ini modifikasi adalah perbaikan, perubahan atau penyesuaian perangkat lunak untuk dapat berubah pada lingkungan, kebutuhan dan fungsionalitas yang spesifik. Selain itu perangkat lunak dapat dianalisis untuk mengetahui apa yang menyebabkan kegagalan pada perangkat lunak untuk mengidentifikasi bagian yang dapat dimodifikasi.
- 8) *Transferability*. Merupakan kemudahan dimana sistem atau komponen dapat berpindah dari lingkungan satu ke lingkungan yang lain dalam hal ini perangkat lunak dapat beradaptasi dengan cepat pada spesifikasi lingkungan yang berbeda tanpa menerapkan aksi atau cara lain dari pada memberikan tujuan tertentu terhadap perangkat lunak yang telah ada.

### E. Skala Pengukuran

Skala pengukuran yang digunakan adalah skala Likert, skala yang didasarkan pada penjumlahan sikap responden dalam merespon pernyataan berkaitan indikator-indikator suatu konsep atau variable yang sedang diukur (Sugiyono, 2019). Skala Likert umumnya menggunakan lima titik dengan label netral pada posisi tengah (ketiga). Skala Likert dapat dilihat pada **Tabel 2.4** di bawah ini.

**Tabel 2. 4** Skala Likert

Jawaban	Skor
Sangat Setuju	5
Setuju	4
Netral	3
Tidak Setuju	2
Sangat Tidak Setuju	1

**Sumber :** (Sugiyono, 2019)

Hasil penilaian responden akan dihitung *persentase* kelayakannya dengan menggunakan perhitungan, dapat dilihat di bawah ini

$$\text{Persentase} = \frac{\text{Skor Aktual (f)}}{\text{Skor Ideal (n)}} \times 100\% \quad \dots\dots\dots(1)$$

Persentase kelayakan yang diperoleh kemudian dibandingkan dengan Tabel konversi yang berpedoman pada acuan konversi nilai, dapat dilihat pada **Tabel 2.5** di bawah ini.

**Tabel 2. 5** Skala Konversi Nilai

Persentase Pencapaian (%)	Interpretasi
$90 \leq x$	Sangat Baik
$80 \leq x < 90$	Baik
$70 \leq x < 80$	Cukup
$60 \leq x < 70$	Kurang
$X < 60$	Sangat Kurang

**Sumber :** (Sugiyono, 2019)

Keterangan: x = persentase hasil pengujian.

## F. Penelitian Terdahulu

Berikut adalah beberapa penelitian yang diperlukan tinjauan pustaka yang diambil dari beberapa jurnal penelitian yang berkaitan dengan judul penelitian ini dan pokok bahasan, berikut ini merupakan tinjauan pustaka yang diambil yaitu :

**Tabel 2. 6** Penelitian Terdahulu

No	Nama (Tahun)	Judul	Masalah	Metode	Hasil
1	Sari et al (2019)	Aplikasi Penggalan Dana Komunitas Jumat Jaman Now Berbasis Web Menggunakan Framewor k Laravel	Donatur mengalami kesulitan untuk melakukan donasi karena harus menghubungi relawan atau founder JJN langsung untuk berdonasi. Relawan	metode pengembangan PIECES. Menggunakan Rapid Application Development(RAD)	Hasil penelitian ini yaitu sistem yang dibangun dapat dengan mudah melakukan donasi sehingga tidak perlu lagi menunggu donasi yang dilakukan relawan yang langsung turun kejalanan. Begitupun sebaliknya pada sisi penggalang dana, mereka dapat dengan mudah mengumpulkan donasi masyarakat.
2	Pratama & Sokhivah	Penggalan Dana melalui	Berdasarkan data pasien terdapat 30%	Kualitatif	Hasil penelitian ini yaitu dapat

No	Nama (Tahun)	Judul	Masalah	Metode	Hasil
	(2020)	Media Sosial untuk Pasien Diabetes di Rumah Sehat Ohisama Muhamma diyah Pontianak Utara	pasien dari kalangan dhuafa yang mendapatkan pelayanan Kesehatan dengan harga yang terjangkau sesuai dengan kemampuannya.		memberikan kontribusi riil bagi pasien luka diabetes yang tidak mampu dan meningkatkan pelayanan kesehatan di Rumah sehat
3	Fitriani & Sholihat (2020)	Pengembangan Sistem Informasi Geografis Penggalan Dana dan Donasi Berbasis Web	Antara lembaga sosial yang ada, dengan orang atau keluarga yang membutuhkan, lebih banyak orang atau keluarga yang membutuhkan, terlihat dari data yang ada di Badan Pusat Statistik Provinsi Jawa Barat.	<i>Rational Unified Process</i>	Hasil pengembangan sistem informasi geografis penggalan dana dan donasi berbasis web mampu menyajikan informasi penerima bantuan yaitu lokasi penerima bantuan, dan detail kebutuhan penerima bantuan, yang dipisahkan menjadi tiga kategori penerima bantuan yaitu orang membutuhkan

No	Nama (Tahun)	Judul	Masalah	Metode	Hasil
4	Romansyah et al (2019)	Rancang Bangun Sistem Informasi Penggalangan Dana Sosial	Masih banyak panti sosial yang tidak mempunyai website sebagai	<i>Waterfall</i>	<p>n, keluarga membutuhkan, dan panti asuhan yang dibedakan dengan warna icon lokasi, selanjutnya data detail kebutuhan penerima bantuan dikelola hingga dapat memisahkan penerima bantuan mana yang harus ditampilkan, dan penerima bantuan mana yang tidak boleh ditampilkan, kemudian laporan penyaluran donasi dibuat dan ditujukan langsung kepada donatur.</p> <p>Hasil yang diperoleh dari aplikasi ini berupa sistem informasi penggalangan dana berbasis</p>

No	Nama (Tahun)	Judul	Masalah	Metode	Hasil
		Panti Asuhan Berbasis Web	sarana penggalangan dana secara online		Web
5	Mulyani & Kurniadi (2020)	Rancang Bangun Sistem Informasi Penggala gan Dana Sosial Panti Asuhan Berbasis Web.	Belum adanya sistem penggalangan dana yang dilakukan	SDLC	Hasil yang diperoleh dari aplikasi ini berupa sistem informasi penggalangan dana berbasis Web