

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Teori Dasar**

##### **2.1.1. Perancangan**

Menurut Soetam Rizky dalam (Nugroho et al., 2015) Perancangan merupakan sebuah proses yang mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta di dalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya. Sedangkan menurut (Firdaus & Rahmawati, 2018) Perancangan merupakan kemampuan untuk membuat beberapa alternatif pemecahan masalah yang mencakup dalam perancangan logis dan fisik.

Berdasarkan beberapa defenisi diatas, dapat disimpulkan bahwa perancangan merupakan tahap awal yang dilakukan dalam menentukan rancangan dalam membangun sebuah projek yang akan dikerjakan dan memecahkan masalah yang akan dihadapi dalam proses pengerjaannya.

##### **2.1.2. Aplikasi**

Secara umum aplikasi merupakan perangkat lunak yang diprogram kedalam satu tujuan aplikasi tersebut yang secara langsung dapat digunakan sesuai dalam keinginan penggunanya. Aplikasi sebagai program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan pembuatan aplikasi tersebut (Wahyuni & Irawan, 2020).

Menurut Dendi Permadi dan Ahmad Rafi dalam (Baker et al., 2020) kepuasan para pengguna dapat ditingkatkan melalui sesuatu hal yang baru dalam aplikasi. Keinginan pengguna terhadap aplikasi diyakini bahwa dapat memicu keingintahuan serta memberikan pengalaman-pengalaman baru kepada pengguna. Dari hal tersebut dapat disimpulkan dengan banyaknya aplikasi-aplikasi baru dapat meningkatkan keingintahuan lebih dari para pengguna serta lebih terlibat dalam mempelajari penggunaan aplikasi.

### **2.1.3. Android**

Android merupakan sistem operasi terhadap perangkat seluler yang berbasis linux yang menyertakan sistem operasi, *middleware*, dan aplikasi. Android menyediakan platform terbuka bagi pengembang untuk pembuatan aplikasinya dalam tujuan memberikan kebebasan terhadap orang yang ingin mengembangkan aplikasinya tersebut diperangkat yang berbeda guna mempermudah pengguna android, serta berpindah dari host ke host lain dalam kinerja tertentu secara efisien (Manikanthan et al., 2020).

Menurut (Irsa et al., 2015) android sebagai platform pertama yang bersifat terbuka dan komprehensif untuk perangkat *mobile*, sederhananya, Android merupakan kombinasi dari komponen-komponen sebagai berikut :

1. Sistem operasi yang bersifat *open source* untuk perangkat *mobile*.
2. Sebuah *open source* platform untuk menciptakan aplikasi *mobile*.
3. Perangkat, telepon seluler khususnya yang menjalankan sistem operasi android dan aplikasi yang dibuat.

Android selalu berkembang sejalan dengan era teknologi sekarang, android terbagi ke beberapa versi beserta tahunnya berikut versi android ada di tabel 2.1.

**Tabel 2.1** Versi Android

| No  | Versi              | Pada Tahun        |
|-----|--------------------|-------------------|
| 1.  | Alpa               | 23 September 2008 |
| 2.  | Beta               | 09 Februari 2009  |
| 3.  | Cupcake            | 27 April 2009     |
| 4.  | Donut              | 15 September 2009 |
| 5.  | Éclair             | 26 Oktober 2009   |
| 6.  | Froyo              | 20 Mei 2010       |
| 7.  | Gingerbread        | 06 Desember 2010  |
| 8.  | Honeycomb          | 22 Februari 2011  |
| 9.  | Ice Cream Sandwich | 18 Oktober 2011   |
| 10. | Jelly Bean         | 09 Juli 2012      |
| 11. | Kit-kat            | 31 Oktober 2013   |
| 12. | Lolipop            | 12 November 2014  |
| 13. | Marshmallow        | 5 Oktober 2015    |
| 14. | Nougat             | 22 Agustus 2016   |
| 15. | Oreo               | 21 Maret 2017     |
| 16. | Pie                | 06 Agustus 2018   |
| 17. | Q                  | 03 September 2019 |

Sumber: (Sadewo et al., 2017)

## **2.2. Teori Khusus**

### **2.2.1. E-Music**

E-Music adalah suatu aplikasi penyewaan alat musik atau sound sistem yang berbasis android. Aplikasi E-music bertujuan untuk memudahkan calon pemesan alat musik dalam melakukan pemesanan kepada penyedia jasa sewa alat musik serta memudahkan para calon penyewa dalam melakukan pembayaran, yang dapat dilakukan secara online atau secara langsung didalam aplikasi.

### **2.2.2. Rapid Application Development (RAD)**

*Rapid Application Development* (RAD) merupakan suatu metode pengembangan perangkat lunak berorientasi objek (Object Oriented Approach) untuk pengembangan sistem. Cara tersebut bertujuan untuk mempersingkat dan mempercepat waktu dalam merencanakan, merancang dan mengimplementasikan sistem dibandingkan dengan metode lain (Putra & Fauziah, 2018). Ada beberapa langkah metode RAD menurut (Handayani et al., 2019) antara lain :

#### 1. *Requirements Planning* (Perencanaan Kegiatan)

Tahap pertama, *requirements planning* bertujuan untuk menentukan kebutuhan dan kendala, pengguna dan analisis bekerja sama dalam membangun objektivitas sistem dengan mengumpulkan data.

#### 2. *RAD Design Workshop* (Desain Pemodelan)

Tahap *RAD Design Workshop* bertujuan merancang dan mendesain semua aktivitas gambaran pola aplikasi sistem kerja. Analisis dan pemrogram komputer dapat bekerja dalam membuat, membangun dan mengekspresikan desain visual dan mekanisme kerja untuk pengguna.

### 3. *Implementation* (Implementasi)

Tahap terakhir, *Implementation* bertujuan dalam mengimplementasikan metode dan program yang sesuai dengan persyaratan sistem. Aktivitas dilakukan melalui sistem build yang sesuai dengan model yang dibangun. Sehingga sistem dapat di uji coba serta memperkenalkan kepada pengguna.

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

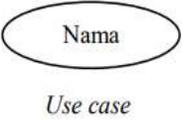
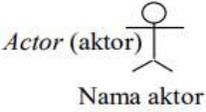
*Unified Modeling Language* atau UML merupakan salah satu dari pada pemodelan visual yang digunakan dalam bentuk dan membuat *software* yang berorientasikan objek. UML merupakan standar penulisan atau sejenis *blue print* yang merangkumi proses bisnis, dalam penulisan kelas dalam bahasa spesifik (Prihandoyo, 2018).

Sedangkan menurut (Munawar, 2018) *Unified Modeling Language* merupakan salah satu alat bantu bahasa sistem yang berorientasi objek yang tersebar luas di dunia yang menyediakan bahasa visual pemodelan untuk menentukan desain serta menggambarkan arsitektur interior pemrograman dalam bentuk objek. Terdapat beberapa diagram UML adalah sebagai berikut :

#### 1. *Use Case Diagram*

*Use case diagram* atau *diagram use case* sebagai *diagram* model perilaku sistem yang akan dirancang dengan mendeskripsikan interaksi antar satu atau bahkan aktor yang akan menggunakan sistem (Hutabri & Putri, 2019).

Tabel 2.2 Use Case Diagram

| Simbol  | Keterangan   |
|---|--|
|    | Menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor.   |
|    | Sebagai aktor atau orang, sistem atau proses yang berperan dan memiliki hubungan dengan sistem informasi yang dibuat.  |
| <p data-bbox="391 842 651 873"><i>Association (asosiasi)</i></p>   | Sebagai interaksi aktor dengan <i>use case</i> atau hubungan komunikasi antara aktor dan <i>use case</i> .   |
| <p data-bbox="435 1020 626 1052"><i>Extend (ekstensi)</i></p> <p data-bbox="456 1083 605 1115">&lt;&lt;extend&gt;&gt;</p>    | Sebagai relasi tambahan <i>use case</i> dengan <i>use case</i> lain. Tanpa digunakan <i>use case</i> , <i>use case</i> bisa berdiri. Petunjuk arah dari panah menuju <i>use case</i> yang ditambahkan.                       |
| <p data-bbox="440 1293 621 1325"><i>Generalization</i></p> <p data-bbox="444 1356 617 1388">(Generalisasi)</p>   | Sebagai hubungan umum–khusus (generalisasi dan spesialisasi) untuk dua <i>use case</i> . Satu <i>use case</i> memiliki fungsi yang lebih umum, dari pada <i>use case</i> lainnya. Arah panah sebagai <i>use case</i> umum.   |
| <p data-bbox="423 1535 634 1566"><i>Uses atau Include</i></p> <p data-bbox="435 1587 623 1619">(Menggunakan)</p> <p data-bbox="440 1640 613 1671">&lt;&lt;include&gt;&gt;</p>  <p data-bbox="456 1692 597 1724">&lt;&lt;uses&gt;&gt;</p>  | Sebagai hubungan dua <i>use case</i> , <i>use case</i> ditambahkan dengan <i>use case</i> tambahan. <i>Use case</i> yang ditambahkan membutuhkan <i>use case</i> selain menjalankan kegunaan fungsi syarat <i>use case</i> . |

Sumber: (Hutabri & Putri, 2019).

## 2. *Activity Diagram*

*Activity Diagram* atau diagram aktivitas mendeskripsikan jalur kerja (*workflow*) atau aktivitas sistem, proses bisnis atau menu yang sudah ada disistem atau perangkat lunak.

**Tabel 2.3** *Activity Diagram*

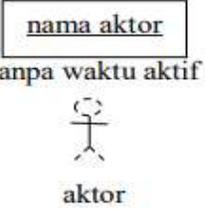
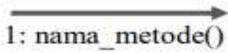
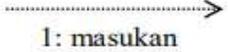
| Simbol   | Keterangan   |
|--|--|
| Status awal/akhir<br>               | Sebagai status awa atau akhir keadaan dari sistem, setiap diagram aktivitas memiliki satu status awal. |
| Aktivitas<br>                       | Sebagai kegiatan yang dilakukan oleh sistem, berawal dengan kata kerja.                                |
| <i>Decision</i> (Percabangan)<br> | Sebagai hubungan percabangan dalam keputusan aktivitas yang memiliki lebih dari satu.                  |
| <i>Join</i> (Penggabungan)<br>    | Merupakan sebagai hubungan penggabungan jika lebih dari satu aktivitas akan menjadi satu.              |
| <i>Swimlane</i><br>               | Merupakan sebagai pemisah organisasi bisnis, dan memiliki tanggung jawab dalam aktivitas yang terjadi. |

Sumber: (Hutabri & Putri, 2019).

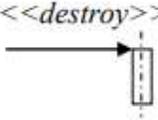
## 3. *Sequence Diagram*

*Sequence Diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

Tabel 2.4 *Sequence Diagram*

| Simbol   | Keterangan   |
|--|--|
|                                       | <p>Sebagai orang atau sistem lain atau proses diluar sistem yang dibuat, yang berkaitan dengan sistem yang dibuat. Aktor belum pasti orang, walaupun simbol bergambarkan orang.</p>          |
| <p><i>Lifeline</i> (Garis hidup)</p>  | <p>Merupakan garis titik-titik yang terhubung dengan objek yang menjelaskan kehidupan objek.</p>   |
|                                       | <p>Merupakan sebagai objek yang melakukan interaksi pesan.</p>   |
| <p>Waktu aktif</p>                  | <p>Merupakan simbol menyatakan objek dalam keadaan aktif dan keadaan berinteraksi, dan semua yang berkaitan dengan waktu aktif. Merupakan tingkatan yang dilakukan selama keadaan aktif.</p> |
| <p>Pesan tipe <i>create</i></p>     | <p>Merupakan simbol menyatakan satu objek membuat objek lain.</p>  |
| <p>Pesan tipe call</p>              | <p>Merupakan pernyataan objek tunggal metode panggilan atau operasi objek lain atau diri sendiri.</p>  |
| <p>Pesan tipe send</p>              | <p>Merupakan sebagai pernyataan pengiriman objek informasi atau input data ke objek lain.</p>  |

Tabel 2.4 Lanjutan

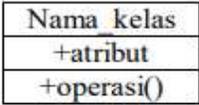
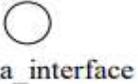
|   |   |
|---|---|
| <p>Pesan tipe return</p> <p>I: keluaran</p>  | <p>Merupakan pernyataan objek sedang menjalankan metode atau operasi yang membuahkan hasil kembali atau keluaran ke objek tertentu.</p> |
| <p>Pesan tipe <i>destroy</i></p>             | <p>Merupakan pernyataan objek lain mengakhiri hidup objek dari objek lain, jika ada <i>create</i> lebih baik ada <i>destroy</i>.</p>    |

Sumber: (Hutabri & Putri, 2019).

#### 4. *Class Diagram*

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, dengan mendeskripsikan struktur sistem dari sisi pendefinisian kelas yang dibuat. Kelas memiliki variabel kelas (atribut) dan fungsi kelas (metode atau operasi).

Tabel 2.5 *Class Diagram*

| Simbol  | Keterangan   |
|---|--|
| <p><i>Class</i> (Kelas)</p>          | <p>Merupakan kelas yang sudah ada dalam struktur sistem. Dengan atribut dan operasi di dalam kelas.</p>                |
| <p><i>Interface</i> (antarmuka)</p>  | <p>Merupakan kesamaan dengan kelas, perbedaannya memiliki metode yang di deklarasikan tanpa atribut dan tanpa isi.</p> |

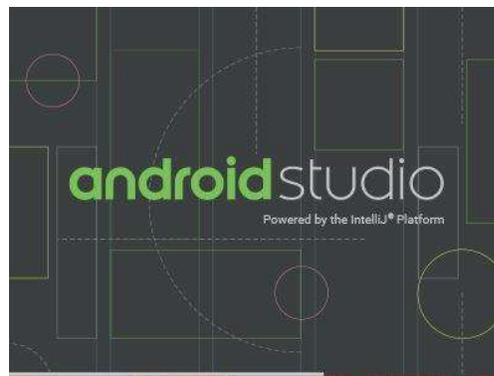
Tabel 2.5 Lanjutan

|   |   |
|---|---|
| <i>Association (Asosiasi)</i><br>                  | Merupakan hubungan antar kelas (umum), biasanya dilengkapi secara <i>multiplicity</i> .   |
| <i>Directed association (Asosiasi berarah)</i><br> | Merupakan hubungan antara satu kelas yang berarti satu kelas digunakan oleh kelas lain. Biasanya dilengkapi <i>multiplicity</i> . |
| <i>Generalisasi</i><br>                            | Merupakan hubungan antara kelas yang berarti generalisasi-spesialialisasi (umum ke khusus).                                       |
| <i>Dependency (Kebergantungan)</i><br>            | Merupakan hubungan mengandalkan atau ketergantungan antar kelas ( <i>dependency</i> ).  |
| <i>Aggregation (Agregasi)</i><br>                | Merupakan hubungan antara kelas mewakili semua bagian ( <i>whole-part</i> ).  |

Sumber: (Hutabri & Putri, 2019).

## 2.2.4. Software Pendukung

### 2.2.4.1. Android Studio



Gambar 2.1 Android Studio

Android Studio merupakan sebuah IDE (*Integrated Development Environment*) untuk pengembangan aplikasi android, aplikasi ini diperkenalkan oleh Google pada 16 Mei 2013 dan tersedia secara gratis dengan lisensi Apache 2.0, Android studio ini menggantikan software pengembangan android sebelumnya yaitu *Eclipse* (Maiyana, 2018).

Menurut (Andi, 2015) pengembangan android studio berdasarkan *IntelliJ IDEA* mirip dengan *Eclipse Plugin ADT (Android Development Tools)*. Android studio memiliki beberapa fitur yakni :

1. Proyek berbasis pada Gradle Build
2. Perbaikan bug yang cepat.
3. Alat baru yang disebut “Lint” diizinkan pemantauan kecepatan, kegunaan dan kompatibilitas dengan cepat.
4. Mendukung Proguard dan penandatanganan aplikasi untuk keamanan.
5. Mendapatkan aplikasi GUI android lebih mudah.
6. Didukung oleh Google Cloud Platform untuk masing-masing pengembangan aplikasi.

#### **2.2.4.2. Android SDK (Software Development Kit)**

Android SDK sebagai alat API (*Application Programming Interface*) harus dimulai pengembangan aplikasi di platform android menggunakan bahasa pemrograman Java. Di android SDK ini terdiri dari debugger dan perpustakaan simulator telepon, otentikasi kode, contoh dan tutorial. SDK memungkinkan pengembang membuat aplikasi untuk platform android SDK, android tutupi

proyek sampel dengan kode sumber daya dan alat pengembangan, emulator dan perpustakaan diperlukan dalam membangun aplikasi dalam penampilan android. Aplikasi tertulis di bahasa pemrograman Java berjalan di perangkat lunak, mesin virtual desainer dibuat khusus untuk penggunaan yang disematkan ini berjalan di *kernel Linux* (Maiyana, 2018).

#### 2.2.4.3. JDK (Java Development Kit)

JDK merupakan kumpulan program yang bisa digunakan untuk mengembangkan perangkat lunak ini berbasis Java. Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lainnya sebagai sumbernya *compiler Java, compiler, debugger*, pengembangan perpustakaan, dan sebagainya. Sedangkan JRE merupakan sebuah penerapan implementasi *Java Virtual Machine* yang sudah digunakan dalam menjalankan program Java (Andi, 2015).

#### 2.2.4.4. Nox Player



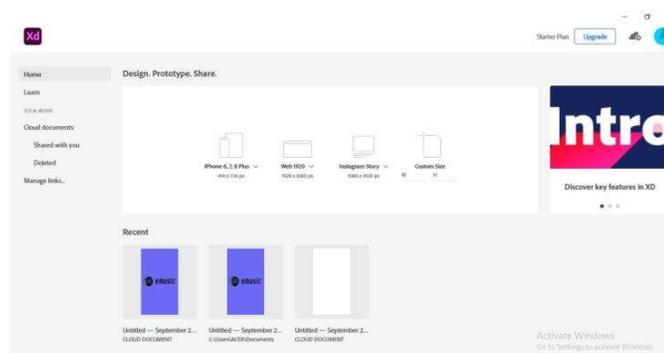
**Gambar 2.2** Nox Player

Nox Player adalah sejenis emulator android yang disukai semua orang, yang memiliki begitu banyak fitur menarik di dalamnya. Nox player yang fungsinya untuk menjalankan aplikasi dalam proyek. Jika terdapat masalah dalam emulator android lain, solusinya anda mencoba emulator ini tanpa ada gangguan. Beberapa kelebihan Nox Player sebagai berikut :

1. Nox Emulator lebih cepat dan tidak mudah *freeze*.
2. Dapat disetting menggunakan *High Performance Mode*.
3. Aplikasi Nox didesain sesuai dengan android yang compatible.
4. Sudah tersedia file manager, facebook lite, kamera dan lainnya didalam.

#### 2.2.4.5. Adobe XD

Menurut (Rahman et al., 2020) *Adobe XD* atau disebut juga *Adobe Experience Design CC* sebagai salah satu aplikasinya untuk membuat desain yang berfokus pada pengalaman tingkat lanjut dan pengguna yang dikembangkan dan diterbitkan oleh *Adobe System*, *Adobe XD* mendukung desain *vector* dan *wireframing*, dan membuat prototype interaktif sederhana. Program ini lebih menekankan pada desain dan mengembangkan produk digital interaktif yang lebih praktis dan ringkas.



**Gambar 2.3** Adobe XD

Terdapat empat fitur utama yang ada di *Adobe XD* yaitu :

1. *Streamlined Design* (beragam tools untuk menggandakan atau memadukan desain dengan mudah).
2. *Interactive Prototypes* (membuat purwarupa desain saling terhubung dari satu halaman ke halaman berikutnya dan kembali lagi).
3. *Live Preview* untuk membuat contoh tampilan hasil akhir dengan mudah.
4. *Easy Sharing Adobe XD* sebagai solusi desain UX/UI untuk desainer dalam merancang dan membuat *prototype*, dan berbagi pengalaman pengguna yang menarik.

#### 2.2.4.6. Adobe Photoshop

Adobe photoshop adalah perangkat lunak (*software*) editor citra dalam menggunakan pengeditan gambar atau foto dari beberapa variasi fasilitas yang disediakan *Adobe Photoshop*. *Adobe photoshop* yang digunakan penulis yaitu versi CS2, karena memiliki fitur instan yang lebih baik dan hampir sama dengan versi yang lain dan lebih menyesuaikan dengan kapasitas prosesor yang ada kadarnya sehingga pengguna dapat berkreasi secara maksimal.



**Gambar 2.4** Adobe Photoshop

#### 2.2.4.7. Firebase

Secara umum *firebase* diartikan sebagai penyimpanan data, *firebase* merupakan platform BaaS (Backend sebagai layanan), *firebase* memiliki beberapa fitur, diantaranya adalah realtime data base yang disimpan secara cloud (Shashidhar et al., 2020), layanan ini menggunakan *Application Program Interface* (API), data disimpan sebagai JSON dan disinkronkan secara realtime ke setiap klien yang terhubung, apabila ada perubahan pada data yang tersimpan, maka setiap pengguna yang terhubung akan menerima pembaruan data secara otomatis.



**Gambar 2.5** Firebase

Menurut (Sudiarta et al., 2018) *firebase* memiliki SDK *real-time* dan memprioritas aplikasi seluler dan mendukung penyimpanan data lokal secara offline. Tujuan utama dari realtime database *firebase* adalah kinerja waktu akses data. Realtime database mengoptimalkan waktu akses, sehingga akses data berada dikisaran mikrodetik bahkan nanodetik, sehingga biaya akses data dapat diminimalkan.

### 2.3. Penelitian Terdahulu

Penelitian terdahulu yang sebelumnya berhubungan dengan aplikasi E-Music sebagai aplikasi pemesanan alat musik, berikut penelitian terdahulu dalam tabel 2.6.

**Tabel 2.6** Penelitian Terdahulu

| <b>Nama Author</b>     | <b>Judul Penelitian</b>  | <b>Masalah Penelitian</b>   | <b>Metode dan Hasil Penelitian</b>  |
|------------------------|--|---|---|
| (Maulana et al., 2018) | Sistem Infomasi Penyewaan Dan Penjualan Pada Darmo Music Studio. JURTI, Vol.2 (1) Juni 2018, 72-81, ISSN: 2579-8790. | Masalah dalam penelitian ini adalah sistem penyewaan, pemesanan, dan penjualan pada darmo music studio hanya dicatat pada sebuah buku yang mengakibatkan sistem belum mampu menyediakan informasi yang cepat. | Metode yang digunakan dalam penelitian ini adalah metode waterfall, yang dirancang dengan menggunakan diagram alur FOD (Flow Of Document) dan Diagram Konteks (Context Diagram). Hasil dalam penelitian ini adalah membangun sistem informasi |

Tabel 2.6 Lanjutan

|                     |   |  |   |
|---------------------|---|--|---|
|                     |   |  | berbasis komputer<br>guna<br>mempermudah dan<br>meningkatkan<br>kinerja darmo<br>music studio dalam<br>proses penyewaan,<br>pemesanan dan<br>penjualan.                                     |
| (Maryanti,<br>2019) | Perancangan<br>Sistem<br>Penyewaan<br>Alat-alat Pesta<br>Pada CV.<br>Widodo Group<br>Berbasis<br>Website.<br>ONESISMIK,<br>Vol. 2 (2) Juni<br>2019, 49-59,<br>ISSN: 2657- | Masalah dalam<br>penelitian ini adalah<br>sistem penyewaan<br>alat-alat pesta<br>dilakukan dengan<br>menghubungi<br>pemilik lewat telpon<br>atau datang di<br>tempat pemilik,<br>sehingga mengalami<br>kendala, kesulitan<br>serta keterlambatan | Dengan metode<br>perancangan<br>menggunakan <i>use<br/>case</i> dan <i>class<br/>diagram</i> dengan<br>menghasilkan<br>aplikasi informasi<br>penyewaan alat-<br>alat pesta berbasis<br>web. |

Tabel 2.6 Lanjutan

|                          |  |  |  |
|--------------------------|--|--|--|
|                          | 0750.  | dalam proses penyewaan alat-alat pesta, sehingga sering terjadi kesalahan dan kurang tepat dalam penyajian laporan dan sulit mengetahui data pelanggan yang diperoleh perusahaan secara cepat. |  |
| (Barmansya et al., 2016) | Sistem Informasi Persewaan Peralatan Pesta Menggunakan SMS Gateway Berbasis Web Pada Persewaan AR Music Dengan | Masalah dalam penelitian ini adalah pelayanan penyewaan alat-alat pesta serta kasus yang sering terjadi seperti kehilangan dan kerusakan data sehingga menghambat kinerja                      | Metode yang digunakan dalam penelitian adalah metode waterfall dengan tahap analisis, desain, pengkodean, pengujian dan pemeliharaan. Hasil dari |

Tabel 2.6 Lanjutan

|                              |   |   |   |
|------------------------------|---|---|---|
|                              | Tambahan Fitur Help Desk. Prosiding SENTIA 2016. Vol 8, 122-127. ISSN: 2085-2347.                             | pegawai pada proses pencarian data dalam mengetahui barang apa saja yang masih terjadi untuk disewakan, serta pendataan kondisi pengembalian barang yang tidak sama dengan barang yang telah dikirim. | penelitian ini adalah sistem aplikasi informasi penyewaan peralatan pesta menggunakan SMS Gateway berbasis web pada persewaan AR Music dengan tambahan fitur Help Desk. |
| (Pradana & Kusniwardi, 2020) | Rancang Bangun Sistem Informasi Studio Rental Rekaman GZ Studio Musik Berbasis Web. RAINSTEK, Vol 2 (2) 2020, | Masalah dalam penelitian adalah proses pemesanan hanya dapat dilakukan dengan datang ditempat secara langsung dan serta membayar uang muka agar   | Metode yang dilakukan dalam penelitian ini adalah model Software Development Life Cycles (SDLC). Dan menghasilkan sistem informasi                                      |

Tabel 2.6 Lanjutan

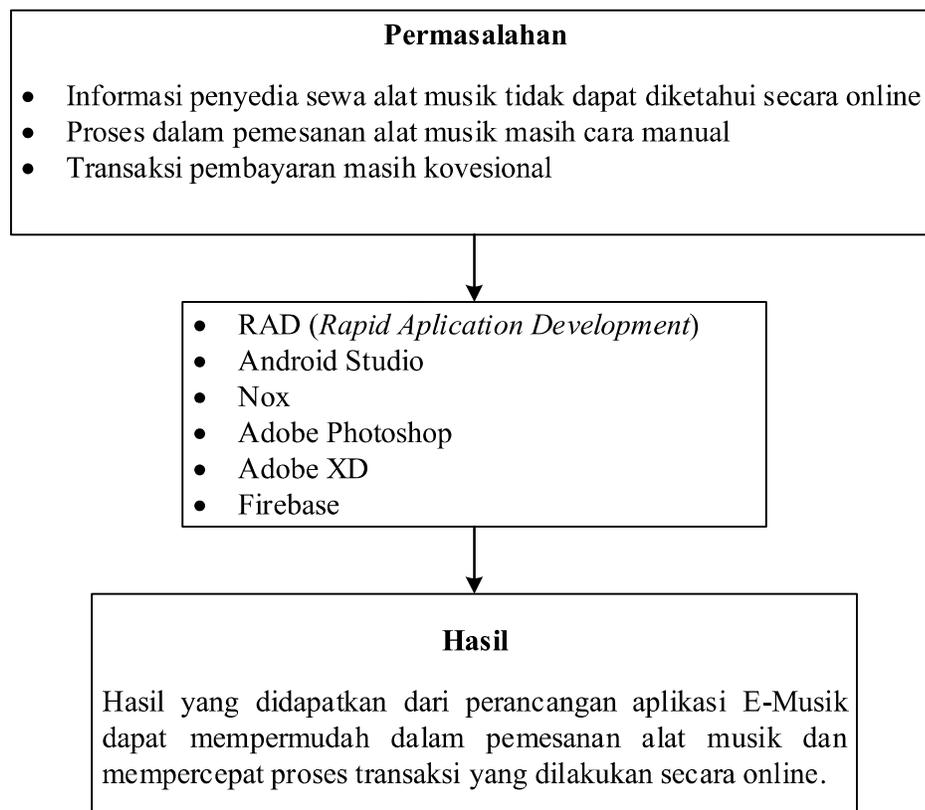
|                            |  |   |  |
|----------------------------|--|---|--|
|                            | 121-128, E-<br>ISSN: 2721-<br>6209.  | waktu yang<br>ditentukan dapat<br>dikonfirmasi, serta<br>penjadwalan sering<br>bentrok terjadi<br>diakibatkan tidak<br>adanya data-data<br>pelanggan.   | studio rental<br>rekaman GZ Studio<br>musik berbasis web   |
| (Ramdhani et<br>al., 2018) | Sistem<br>Informasi<br>Penyewaan<br>Peralatan <i>Event<br/>Organizer</i><br>Berbasis Web<br>Pada PT.<br>Adecon Jakarta.<br>JURNAL<br>ABDIMAS BSI,<br>Vol 1 (3)<br>Agustus 2018,<br>390-397, E-<br>ISSN:2614- | Masalah dalam<br>penelitian ini adalah<br>proses penyewaan,<br>pengelolaan data,<br>pemesanan peralatan<br>serta informasi<br>ketersediaan<br>peralatan yang akan<br>disewakan dan<br>proses transaksi<br>pembayaran hanya<br>bisa dilakukan<br>secara konvensional<br>dimana hanya | Dalam penelitian<br>ini digunakan<br>metode waterfall<br>yang terbagi dalam<br>tiga tahap yaitu,<br>Analisa kebutuhan,<br>Perancangan sistem<br>dan perangkat<br>lunak,<br>Implementasi dan<br>pengujian unit.<br>Dengan hasil<br>membuat website<br>pemesanan |

Tabel 2.6 Lanjutan

|  |       |                                      |  |
|--|-------|--------------------------------------|--|
|  | 6711. | menggunakan catatan buku dan kertas. | penyewaan alat <i>event organizer</i> pada PT. Adecon. |
|--|-------|--------------------------------------|--|

#### 2.4. Kerangka Pemikiran

Kerangka pemikiran dari penelitian ini seperti pada gambar 2.6.



**Gambar 2.6** Kerangka Pemikiran

Sumber: (Penelitian 2020)