

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Penelitian Terdahulu**

Adanya perbandingan terhadap sumber referensi terdahulu dalam pengembangan “Rancang Bangun Aplikasi Pemesanan Vendor Pernikahan Berbasis Android di Kota Probolinggo” maka diperlukan adanya suatu acuan dalam tahap penelitian ini yang telah dirancang sebelumnya. Berikut merupakan hasil penelitian yang dijadikan referensi dalam membangun yaitu :

Penelitian yang sudah dilakukan peneliti Haris dan Ridwan (2020) yang berjudul “Perancangan Aplikasi Wedding Organizer Berbasis Android”. Adapun tujuannya dari penelitian ini yaitu membuat aplikasi wedding organizer berbasis android dengan tujuan memudahkan customer dalam memesan paket pernikahan menggunakan perangkat android. Pada proses perancangan sistem terdapat metode didalam penelitian ini, yaitu metode Rational Unified Process dimana metode ini mempunyai 4 tahapan diantaranya inception, elaboration, construction, dan transition. Hasil pembuatan penelitian yaitu membantu customer dengan wedding organizer untuk mempermudah memesan jasa didalam perangkat android, proses ini dilakukan agar menjadi lebih mudah.

Penelitian yang dilakukan oleh Ainun dan Nikmasari (2021) dengan judul “Sistem Informasi Wedding Organizer Berbasis Android”. Tujuan adanya penelitian adalah memberikan informasi terkait wedding organizer kepada calon pengantin dengan tepat dan mempermudah pihak *wedding organizer*

menawarkan jasanya secara efektif. Adapun metode yang telah digunakan dalam penelitian ini yaitu, metode prototype dimana metode ini memiliki 4 tahapan diantaranya, communication, quick plan and modeling quick design, construction of prototype dan development delivery and feedback. Adapun hasil dari pembuatan penelitian adalah dapat memberikan sistem informasi wedding organizer dengan tepat dan bisa menambahkan fitur penambahan paket pernikahan yang bisa dilakukan kapan saja dan menambahkan fitur pemesanan dengan cepat.

Penelitian yang sudah dilakukan Rima dan Afriyudi (2019) yang berjudul “Aplikasi Pemesanan Paket Wedding Organizer Berbasis Android”. Adapun tujuan dibuatnya aplikasi yang sudah dibuat yaitu membangun aplikasi wedding organizer berbasis android dengan tujuan mempermudah customer untuk memesan paket nikah dengan menggunakan bahasa pemrograman android dan PHP. Adapun metode yang sudah ada didalam penelitian menggunakan metode pengumpulan data yaitu dengan, studi literature, interview, serta observasi. Dimana dengan metode studi literature, interview, serta observasi bisa sangat membantu peneliti dalam mendukung pengumpulan data pada penyusunan aplikasi tersebut. Hasil dari pembuatan penelitian ini adalah memberikan kemudahan kepada calon pengantin terkait konsep pernikahan dan pemesanan paket acara pernikahan sesuai keinginan calon pengantin adanya sistem android ini sangat membantu prosesnya bisa lebih cepat dan efisien tanpa harus ke tempat atau lokasi *wedding organizer* dan juga sangat memudahkan bagi pengguna jasa *wedding organizer* dalam menawarkan jasanya.

Tabel 2.1 Perbandingan Penelitian Terdahulu

NO	PENELITI	JUDUL	Fitur pada Sistem			
			MUA	Fotografer	Video grafer	Deko rasi
1	Rima Ulfa, Afriyudi (2019)	Aplikasi Pemesanan Paket Wedding Organizer Berbasis Android	√	√	√	X
2	Haris Ginanjar, Ridwan Setiawan (2020)	Perancangan Aplikasi Wedding Organizer Berbasis Android	√	√	√	X
3	Ainun Ishak, Nikmasari Pakaya (2021)	Sistem Informasi Wedding Organizer Berbasis Android	√	X	X	√
4	Peneliti (2022)	Rancang Bangun Aplikasi Pemesanan Vendor Pernikahan Berbasis Android di Kota Probolinggo	√	√	√	√

(Sumber : data diolah oleh penulis)

## 2.2 Landasan Teori

### 2.2.1 Aplikasi

Aplikasi adalah sistem yang ada di perangkat lunak yang meminimalkan kemampuan dari komputer baik langsung untuk menjalankan perintah yang diperintahkan. Aplikasi adalah suatu perintah yang dapat dijalankan sesuai dengan penggunaannya yang kemudian akan diproses dengan perangkat keras. Aplikasi

adalah suatu file yang didalamnya terdapat perintah atau kode program yang terdapat koneksi antar pengguna serta perangkat keras. (Putranto,2018).

Aplikasi juga dikenal dengan perangkat lunak, dimana kode perintah yang terdapat pada komputer yang perintahnya bisa dirubah dengan sangat mudah. Aplikasi biasanya dipergunakan untuk mengatur perangkat keras atau yang biasa dinamakan *device driver*. Secara garis besar aplikasi dipilah menjadi 3 tingkatan yang terdapat pada program *software* (application program misal : Microsoft Office), tingkat sistem operasi (*operating system* misal : Microsoft Windows), dan tingkat bahasa pemrograman (misal Java). (Putranto,2018).

Terdapat aplikasi yang sudah digabungkan untuk menjadi sebuah paket aplikasi yang dinamakan sebagai suite aplikasi (*application suite*). Contoh beberapa aplikasi yaitu Microsoft Office dan OpenOffice.org yaitu dapat menyatukan aplikasi pemilah kata, lembar pekerjaan, serta aplikasi lainnya. Aplikasi didalam sebuah paket terdapat fitur antar muka dari pengguna dengan mempunyai kesamaan sehingga lebih memudahkan pengguna untuk menjalankan aplikasi. Biasanya pengguna mempunyai keinginan untuk mengenal pengguna satu dengan yang lain sehingga bisa menguntungkan. Contoh, sebuah project dapat dihapus dalam sebuah dokumen pengelolaan kata maupun dikerjakan didalam aplikasi project yang tidak sama.

Adapun aplikasi dan perangkat lunak bisa dikategorikan menjadi beberapa kelas yaitu :

1. Perangkat lunak infrastruktur perusahaan
2. Perangkat lunak informasi kerja

3. Perangkat lunak media dan hiburan
4. Perangkat lunak perusahaan (enterprise)
5. Perangkat lunak pendidikan
6. Perangkat lunak pengembangan media
7. Perangkat lunak rekayasa produk

### **2.2.2 Android**

Android merupakan suatu sistem operasi berbasis linux yang dipergunakan untuk software yang bergerak layar sentuh misalnya smartphone serta komputer tablet. Android juga terdapat keunggulan jika dibanding dengan software smarthphone, keunggulannya adalah open souce code dimana dapat dengan memudahkan dalam perancangan aplikasinya. (Haris dan Ridwan, 2020).

### **2.2.3 Java**

Bahasa pemograman pada java merupakan bahasa pemograman yang digunakan didalam macam komputer, termasuk smartphone. Bahasa java pertama dikemukakan oleh Sun Microsystem ditahu 1995. Bahasa java adalah sebuah teknologi perangkat lunak yang masuk multi platform. Bahasa java merupakan platform dengan virtual machine yang digunakan untuk menjalankan perintah program. Didalam penelitian ini digunakan bahasa java sebagai perintah kode program yang terdapat pada aplikasi yang dibuat. Peneliti menggunakan bahasa pemograman java pada penelitian ini, dikarenakan bahasa pemograman java adalah salah satu pemograman yang mudah untuk dipahami.

#### 2.2.4 DBMS (*Database Management System*)

DBMS adalah sebuah sistem untuk penyimpanan suatu data yang biasa dikenal dengan sebutan database, dengan fungsi untuk menyimpan data yang keterkaitan dan mempunyai hubungan antar yang lain, contohnya yaitu *Postgree*, *Oracle*, dan banyak lainnya (R. Y. Endra, 2016). Untuk mengakses DBMS, digunakan bahasa SQL (*Structured Query Language*). Kode *database* dikenal dengan banyak macam perintah yang disingkat dan terdaoat kesimpulan perintah itu dikelola *database*. Perintah ini dikelola oleh pengguna, kode perintah yang sering dipergunakan yaitu ada 2 perintah :

1. DDL

DDL (*Data Definition Language*), adalah bahasa biasa digunakan untuk menjelaskan gambar rancangan suatu *database* secara detail. DDL (*Data Definition Language*) digunakan untuk merancang suatu table atau merubah table serta pembuatan indeks.

2. DML

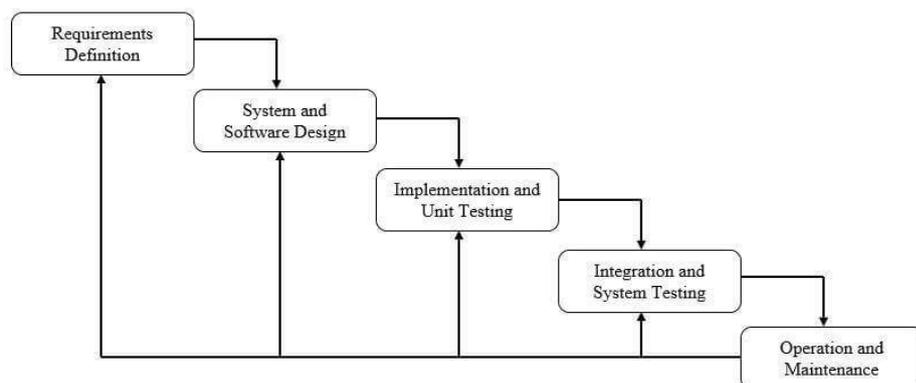
DML (*Data Manipulation Language*), adalah digunakan untuk memanipulasi suatu data untuk pengambilan data di dalam *database*, seperti penambahan data yang baru dimasukkan di dalam *database*, melakukan hapus data di dalam *database*, serta melakukan proses hapus di dalam *database*. Didalam penelitian ini sangat dibutuhkan DML (*Data Manipulation Language*), dimana dalam penelitian ini fungsi tambah, hapus, edit sangat dibutuhkan. Dan DML (*Data Manipulation Language*) memudahkan dalam pengambilan *database*.

### 2.2.5 Waterfall Model

Metode pengembangan sistem yang digunakan merupakan suatu permodelan dari pengembangan suatu perangkat lunak. Metode *waterfall* merupakan pemrosesan pengembangan suatu perangkat, didalam pemrosesan tersebut dilihat untuk mengalir ke bawah (seperti model air terjun) dan melewati tahap perencanaan, pemodelan, implementasi, serta pengujian. Hubungan dari tahapan tersebut muncul dikarenakan keluaran dari proses didalam metode ini adalah masukan untuk proses setelahnya. Untuk ini ketidak sempurnaan didalam hasil proses sebelumnya merupakan awalan ketidak sempurnaan proses setelahnya

Model ini digunakan dalam penelitian ini yaitu memiliki kemudahan dalam rekayasa yang dapat membuat proses kualitas perangkat tetap baik. Model *waterfall* merupakan model pengembangan yang terstruktur dan terawasi.

Gambar 2.1 Contoh *Waterfall Model*



(Sumber : Nandini dan Nelsi,2021)

Penjelasan dari masing-masing tahapan Model Waterfall yaitu:

1. Analisis Kebutuhan

Pada tahapan ini memiliki fungsi sebagai pembantu untuk mengerti keinginan dari sistem yang dibuat jadi dapat diteruskan sebagai sistem yang bisa untuk memenuhi kebutuhannya.

## 2. Perancangan

Pada tahap perancangan sangat dibutuhkan untuk memperluas sistem informasi yang didapat. Pada tahap ini memiliki beberapa proses data, yaitu perancangan antarmuka, arsitektural, serta perancangan komponen data.

## 3. Pemograman

Pada tahapan ini disebut tahapan implementasi sistem perangkat lunak atau sering disebut *coding*. Didalam tahapan ini ada proses implementasi hasil proses didalam tiap baris bahasa proses yang sudah dimengerti oleh sistem.

## 4. Pengujian

Pada tahapan pengujian adalah suatu proses yang dijalankan dan evaluasi pada sistem baik dalam cara manual ataupun dengan cara otomatis. Dalam tahapan ini berfungsi sebagai penguji untuk sistem untuk menentukan hasil yang sudah dirancang sebelumnya dan apakah sudah sesuai dengan hasilnya.

Kelebihan Model Waterfall :

1. Permodelan ini adalah sebuah model yang sudah dikembangkan dengan sangat lama

2. Permodelan ini sudah sangat cocok digunakan sebagai sistem yang memiliki sifat
3. Pada proses pengerjaan model ini sistem sudah tertata dengan sangat baik dan sangat mudah

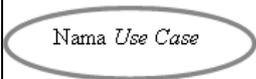
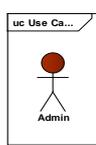
Adapun Kekurangan Model Waterfall :

1. Permodelan ini harus digambarkan dengan sangat jelas
2. Susunan pada proses permodelan ini tidak diperbolehkan untuk diganti
3. Pada permodelan ini masih sangat sulit untuk mendeteksi jika ada pergantian pada proses permodelan
4. Permodelan ini masih tidak mudah untuk dipahami

### **2.2.6 Usecase Diagram**

*Usecase diagram* merupakan satu dari jenis diagram UML (*Unified Modelling Language*) yang menggambarkan hubungan interaksi antara sistem dan actor. Usecase dapat mendeskripsikan interaksi antara pengguna sistem dengan sistemnya. Fungsi dari usecase ini salah satunya untuk memperlihatkan proses aktivitas secara urut dalam sistem. Didalam penelitian ini usecase diagram sangat diperlukan, dimana mengingat fungsi usecase diagram adalah untuk menggambarkan hubungan interaksi antara sistem dengan actor. Pada *usecase diagram* ini nantinya akan dijelaskan bagaimana alur hubungan antara actor siapa saja yang akan menggunakan aplikasi tersebut dengan sistem yang ada pada aplikasi. *Usecase diagram* juga mempermudah bagi pengguna untuk menjelaskan terkait alur-alur pada penelitian ini.

Tabel 2.2 Simbol Usecase Diagram

Simbol	Nama	Keterangan
	<i>Use case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. Biasanya dinyatakan dengan menggunakan kata kerja awal di awal frase nama <i>use case</i> .
	Aktor	Orang, proses atau sistem lain yang berinteraksi dengan sistem yang akan dibuat.
	Asosiasi	Komunikasi antar aktor atau <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
	<< <i>Extend</i> >>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

Tabel 2.2 Lanjutan

Simbol	Nama	Keterangan
	<<Include>>	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
	Generalisasi	Hubungan generalisasi antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.

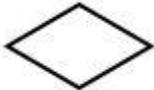
Sumber : (Haris dan Ridwan,2020)

Gambar diatas merupakan simbol-simbol *usecase* diagram. Dimana simbol *usecase* diagram diatas menjelaskan fungsi setiap simbol yang ada di *usecase* diagram.

### 2.2.7 Activity Diagram

Activity Diagram atau Diagram aktivitas adalah bentuk visual dari alur kerja yang berisi aktivitas dan tindakan, yang juga dapat berisi pilihan, atau pengulangan. Dalam Unified Modeling Language(UML), diagram aktivitas dibuat untuk menjelaskan aktivitas komputer maupun alur aktivitas dalam organisasi. Selain itu diagram aktivitas juga menggambarkan alur kontrol secara garis besar.

Tabel 2.3 Simbol Activity Diagram

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Swimlane adalah pemisah pada tiap aktivitas

(Sumber : Nandini dan Nelsi,2021)

### **2.2.8 Sequence Diagram**

*Sequence* diagram digunakan untuk menggambarkan interaksi antar objek di dalam dan di sekitar sistem yang berupa message yang digambarkan terhadap waktu. *Sequence* diagram terdiri antara dimensi vertical (waktu) dan dimensi horizontal (objek-objek yang terkait).

### **2.2.9 Class Diagram**

*Class* diagram adalah visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak. *Class* diagram juga dapat memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas didalam model desain (*logical view*) dari suatu sistem. Selama proses *desain*, *class* diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

### **2.2.10 Deployment Diagram**

*Deployment* diagram adalah salah satu model diagram dalam UML untuk mengarahkan artefak dalam node. *Deployment* diagram digunakan untuk memvisualisasikan hubungan antara *software* dan *hardware*.

### **2.2.11 Desain Interface**

*Desain interface* adalah proses yang digunakan untuk membuat tampilan dalam perangkat lunak. Tujuan adanya desain interface agar pengguna bisa

mengeksplor sistem dengan mudah. *Desain interface* yang baik yaitu desain yang tidak membingungkan pengguna ketika memakai aplikasi tersebut.