

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

Berdasarkan Penelitian yang dilakukan Pinto, M.D.R., dkk (2020) dengan judul **“Rancang Bangun Aplikasi Pemesanan Air Bersih Berbasis Android Dengan Menggunakan Model Prototype”**. Tujuan penelitian ini yaitu membangun sebuah aplikasi pemesanan air bersih agar dapat membantu proses pemesanan air bersih di kabupaten Timor Tengah Selatan (TTS) Provinsi Nusa Tenggara Timur (NTT) menggunakan digital. Aplikasi pemesanan ini dibangun menggunakan bahasa pemrograman Java, XML, SQLite, PHP, Javascript dan database MySQL. Software pengembang aplikasi android menggunakan android studio. Penelitian ini menghasilkan 2 user berbeda yaitu admin dan customer. Setiap fitur pada masing-masing user berhasil untuk menampilkan, memesan produk, menambahkan produk, serta update data produk. Dari Hasil pengujian yang diperoleh dengan menggunakan pengujian alpha dan pengujian beta, sistem aplikasi berhasil mengoperasikan bagian-bagian menu serta memperoleh nilai persentase 93.20% berupa hasil pengujian yang dicoba pada pengguna.

Berdasarkan Penelitian yang dilakukan SETIAWAN, A.T. (2019) dengan judul **“Rancang Bangun Aplikasi Pemesanan Paket Umroh Pada PT.AR.Rosyid Tour Dan Travel Berbasis Android Menggunakan Model**

Prototype". Penelitian Ini Bertujuan untuk membuat aplikasi pemesanan paket umroh berbasis Android di PT. Ar Rosyid Tour dan Travel dengan menggunakan model prototype. Dari hasil pengujian, tidak ditemukan fungsi-fungsi yang tidak sesuai dengan kebutuhan kegiatan proses pemesanan perjalanan paket umroh. Dengan menggunakan aplikasi ini, pengguna dapat melakukan pemesanan paket perjalanan umroh dengan mudah dan nyaman.

Berdasarkan Penelitian yang dilakukan Putri, N. dkk (2019) dengan judul **"Implementasi Metode Prototyping pada Perancangan Aplikasi *Electronic Ticket (E-Ticket)* berbasis Android"**. Untuk memudahkan calon pengguna dalam melakukan pemesanan tiket dan pencarian informasi jadwal keberangkatan bus, maka dibangun aplikasi pemesanan tiket bus berbasis Android. Aplikasi ini dirancang dengan menggunakan MySQL, Adobe Dreamweaver dan Android Studio, Metode penelitian yang digunakan adalah metode prototyping. Sedangkan untuk metode pengujian dengan menggunakan BlackBox Testing. Metode ini digunakan untuk mengetahui apabila ada kesalahan kinerja pada sistem tersebut. Tujuan dibuatnya aplikasi ini adalah untuk memudahkan calon pengguna bus dalam melakukan pencarian bus dan pemesanan tiket agar bisa dilakukan dimana saja dan kapan saja tanpa terikat dengan waktu dan tempat.

II.2. Landasan Teori

Berikut ini adalah landasan teori yang berkaitan dengan penelitian yang diambil dari beberapa jurnal terkait penelitian :

II.2.1. Rancang Bangun

Perancangan merupakan sebuah alur atau proses aplikasi macam-macam teknik prinsip dengan tujuan pendefinisian suatu sistem, perangkat, atau proses detail yang mencakupi untuk memungkinkan pengaktualan fisiknya (Galuh Saputri, 2020).

Perancangan sistem adalah sebuah kegiatan merancang dan menentukan cara mengolah sistem informasi dari hasil analisa sistem sehingga dapat memenuhi kebutuhan dari pengguna termasuk diantaranya perancangan *user interface*, data dan aktivitas proses. (Kholik Hidayatulloh dkk, 2020)

II.2.2. Aplikasi

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut. Aplikasi mempunyai ar yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. Pengeran aplikasi secara umum adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya. Aplikasi

merupakan suatu perangkat computer yang siap pakai bagi user. (Refni Wahyuni dan Yuda Irawan, 2020).

Aplikasi merupakan program yang secara langsung dapat melakukan proses-proses yang digunakan dalam komputer oleh pengguna. Aplikasi merupakan kumpulan dari *file-file* tertentu yang berisi kode program yang menghubungkan antara pengguna dan perangkat keras Komputer. (Sihombing dan Yanris, 2020: 13).

III.2.3. Pemesanan

Pemesanan adalah aktivitas yang selalu dilakukan konsumen sebelum proses membeli. Agar dapat memberikan kepuasan kepada konsumen maka dari itu, tentunya perusahaan harus mempunyai sistem pemesanan yang baik, (Rizky Ahmad Bagja, 2022)

Dalam proses pemesanan atau pengiriman pesanan. Aktivitas-aktivitas yang termasuk didalamnya:

1. Mencari dan melacak pesanan sepanjang keseluruhan siklus pesanan;
2. Berkomunikasi dengan konsumen mengenai keberadaan pesanan dan kapan pengiriman akan berlangsung.

Kesimpulannya, pemesanan adalah satu kegiatan penting dalam setiap perusahaan yang bergerak dibidang perdagangan. Untuk membantu dan mengawasi kegiatan pembelian, maka diperlukan dukungan sistem informasi yang baik, sehingga dapat mengikuti perkembangan perusahaan yang sedang tumbuh. (Rahmasari, 2017:58).

III.2.4. Gear

Roda gigi (*Gear*) adalah salah satu komponen penting dalam dunia mekanika, digunakan untuk meningkatkan atau menurunkan torsi, merubah arah gerak serta mentransmisikan daya dari suatu sistem gerak (Y. Martin dan A. Suwandi, 2020).

Roda gigi merupakan suatu elemen mesin yang berfungsi untuk mentransmisikan daya dan putaran poros sehingga system mekanisme pada mesin dapat bekerja sesuai dengan fungsinya. Ada beberapa yang harus diperhatikan dalam perancangan roda gigi, yaitu seperti dimensi, gaya, dan torsi dan tegangan-tegangan yang bekerja pada roda gigi, karena apabila hal ini tidak dilakukan maka akan menyebabkan roda gigi tidak dapat digunakan dengan baik. Perancangan roda gigi yang tidak dapat beroperasi dengan baik seperti kontak antar gigi yang kasar mengakibatkan gerak antar gigi tidak sempurna sehingga mengakibatkan ketidakseragaman gaya kontak antar satu gigi dengan gigi lainnya. Kerusakan roda gigi dapat disebabkan oleh berbagai faktor termasuk pelumasan yang tidak memadai, kondisi operasi, material dan proses manufaktur yang tidak sesuai dengan spesifikasi (Y. Martin dan A. Suwandi, 2020).

II.2.5. Besi

Besi merupakan logam yang paling banyak terdapat di alam, dalam kehidupan, Besi merupakan logam paling biasa digunakan dari pada logam-logam yang lain sebagai paduan logam. Salahsatunya adalah meningkatnya Besi atau logam yang dijadikan bahan utama dalam penciptaan atau produktivitas suatu benda, seperti Besi beton untuk bangunan, Besi pada kendaraan bermotor dan benda

lainnya yang berbahan baku Besi. Dengan adanya kejadian seperti ini maka limbah Besi akan lahir dan akan menjadi permasalahan baru untuk kelangsungan hidup manusia jika penanganannya tidak tepat. (Maryam Pomalingo dkk, 2020).

II.2.6. Model Prototype

Model ini melakukan pengerjaan suatu sistem secara berurutan atau linear. Metode penelitian ini memiliki 5 langkah yaitu: *Communication*, *Quick Plan*, *Modeling Quick Design*, *Construction of Prototype*, dan *Deployment Delivery and Feedback*. Melalui model prototyping ini pengembang dan pengguna dapat saling berinteraksi selama proses pembuatan sistem. Perubahan terhadap desain dan sistem yang diharapkan bersama dapat dengan mudah diadopsi dan disesuaikan melalui kesepakatan. Secara spontan, pengguna dapat memberikan masukan atas hasil yang diperoleh dan tentu menjadi masukan bagi pengembang sistem. Kemudian pengembang berkewajiban membuat sistem berdasarkan masukan agar diperoleh hasil desain yang baik dan sesuai dengan kebutuhan pengguna. Sistem yang baik akan mengurangi risiko kesalahan dalam menangani transaksi yang jumlahnya banyak dan transaksi yang terjadi berulang-ulang.

Berikut penjelasan mengenai tahapan dalam model *Prototype*, yaitu:

1. Communication

Tahapan ini dilakukan dengan cara melakukan wawancara dan pengumpulan data yang berkaitan dengan kebutuhan.

2. *Quick Plan*

Tahapan ini membuat perencanaan secara cepat dengan cara membuat agenda pertemuan, mendiskusikan materi, membuat langkah perencanaan, hingga pelaksanaan.

3. *Modeling Quick Design*

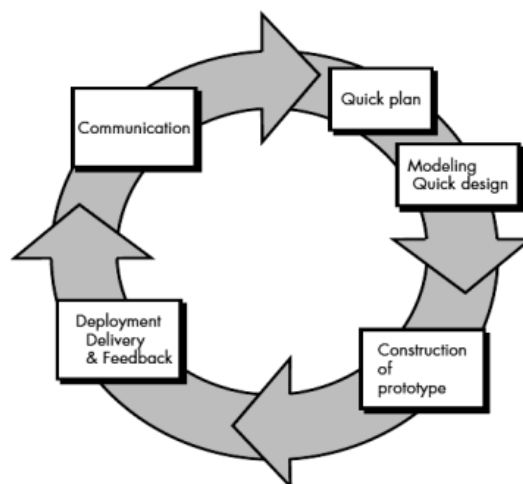
Tahapan ini membuat pemodelan perancangan secara cepat dengan cara mendesain model.

4. *Construction of Prototype*

Tahapan ini membentuk prototipe yang sesuai dengan harapan dan kebutuhan.

5. *Deployment Delivery and Feedback*

Tahapan ini dilakukan dengan cara menyerahkan model kepada pengguna untuk kemudian melakukan pengiriman dan menerima umpan balik. (Saptadi, dkk, 2019).



Gambar II.1. Model Prototype

(Sumber: Saptadi, dkk, 2019)

II.2.7. Android

Android merupakan sebuah perangkat lunak yang terdapat dalam perangkat *mobile* berbasis linux yang mencakup sistem operasi, layanan di tengah dan aplikasi. *Android* menyediakan *platform* terbuka untuk para pengembang untuk membuat aplikasi. *Android* dibeli oleh Google Inc. dari Android Inc. *Android* bukan merupakan bahasa pemrograman, akan tetapi hanya menyediakan lingkungan atau lingkungan kerja yang disebut DVM (*Dalvik Virtual Machine*) yang telah dioptimalkan untuk alat dengan sistem memori yang kecil. Proses android dikembangkan pertamakali dikembangkan oleh OHA (*Open Handset Alliance*), konsorsium dari 34 perusahaan peranti keras, peranti lunak dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia (Ivan Maurits, 2020).

II.2.8. Hypertext Preprocessor (PHP)

PHP merupakan singkatan dari “*Hypertext Preprocessor*”. PHP adalah sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaknya mirip dengan bahasa pemrograman C, Java, ASP dan Perl ditambah beberapa fungsi PHP yang Spesifik dan mudah dimengerti. PHP digunakan untuk membuat tampilan *web* menjadi lebih dinamis, dengan PHP anda bisa menampilkan atau menjalankan beberapa *file* dalam 1 *file* dengan cara di *include* dan *require*. PHP itu sendiri sudah dapat berinteraksi dengan beberapa database walaupun dengan kelengkapan yang berbeda yaitu seperti DBM, MySQL, Oracle (Tiara Rahmasari, 2019)

II.2.9. *Hyper Text Markup Language (HTML)*

HTML ialah kepanjangan dari *Hypertext Markup Language*. Definisi HTML adalah bahasa yang digunakan untuk menulis halaman *web*. fungsi utama HTML ialah memberi perintah pada *browser* untuk melakukan manipulasi tampilan melalui *tag-tag* yang ditulis dalam *html* (Tiara Rahmasari, 2019)

II.2.10. *MySQL*

Pada perkembangannya, *MYSQL* disebut juga *SQL* yang merupakan singkatan dari *Structured Query Language*. *SQL* merupakan bahasa terstruktur yang khusus digunakan untuk mengolah *database*. *SQL* pertama kali didefinisikan oleh *American National Standards Institute (ANSI)* pada tahun 1986. *MYSQL* adalah sebuah sistem manajemen *database* yang *set* bersifat *open source*. *MYSQL* merupakan sistem manajemen *database* yang bersifat *relational*. Artinya, data yang dikelola dalam *database* yang akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan jauh lebih cepat. *MYSQL* dapat digunakan untuk mengelola *database* mulai dari yang kecil sampai dengan yang sangat besar. (Muhammad Saed Novendri, 2019)

II.2.11. *Basis Data (Database)*

Basis data terdiri atas dua kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang. Sedangkan data adalah representasi fakta dunia nyata mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya. (Sugeng dan Hairil, 2018)

II.2.12. *Unified Modelling Language (UML)*

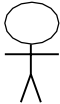
UML yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan system yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. (Andikos, 2019: 39).

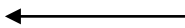
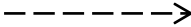
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.1.

Tabel II.1. Simbol *Use Case*

Gambar	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .

	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.

	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang mnghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

(Sumber: Andikos, 2019: 39)

2. Diagram Aktivitas (*Activity Diagram*)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.2.

Tabel II.2. Simbol *Activity Diagram*

Gambar	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka salingberinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

(Sumber: Andikos, 2019: 39)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.3.

Tabel II.3. Simbol *Sequence Diagram*

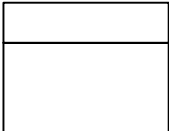
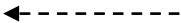
Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek entity, antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antarobjek yang memuat informasi- informasi tentang aktifitas yang terjadi.
	<i>Message</i>	Spesifikasi dari komunikasi antarobjek yang memuat informasi- informasi tentang aktifitas yang terjadi.

(Sumber: Andikos, 2019: 39)

4. *Class Diagram* (Diagram Kelas)

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4

Tabel II.4. *Class Diagram*

Gambar	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan

		mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
_____	<i>Assocation</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber: Andikos, 2019: 39)