



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

Teori Umum terdiri dari 4 bagian yaitu **2.1.1 Pengertian Perangkat Lunak, 2.1.2 Pengertian Komputer, 2.1.3 Pengertian Data, dan 2.1.4 Pengertian Basis Data (Database)**. Berikut penjelasannya,

2.1.1. Pengertian Perangkat Lunak

Menurut Siallagan (2017:3) menyatakan bahwa, “*Software* merupakan program-program komputer yang berguna untuk menjalankan atau mengoperasikan suatu pekerjaan sesuai dengan yang dikehendaki”. Adapun Mulyono (2017:97) mengemukakan bahwa, “*Software* adalah rangkaian instruksi elektronik yang memerintahkan komputer untuk melakukan tugas tertentu sesuai dengan perintah yang diberikan oleh seorang pengguna komputer”. Sedangkan menurut Ladjamudin (2017:20), “*Software* merupakan kumpulan dari perintah/fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu”. Dari beberapa pendapat dari para ahli seperti diatas, maka dapat disimpulkan bahwa definisi Perangkat Lunak atau *Software* merupakan instruksi yang tersusun sedemikian rupa dari sebuah program komputer yang berfungsi mengeksekusi berbagai bentuk masalah tertentu dengan metode tertentu.

2.1.2. Pengertian Komputer

Puspitosari (2017:1) berpendapat bahwa, “Komputer dapat didefinisikan sebagai sekumpulan alat elektronik yang saling terkoordinasi satu sama lain sehingga dapat menerima data, kemudian mengolah data, dan pada akhirnya akan menghasilkan suatu keluaran yang berupa informasi (*Input-Proses-Output*)”. Sama halnya yang dikemukakan oleh Mulyono (2020:1) bahwasanya, “Komputer adalah



seperangkat alat elektronik yang terdiri atas peralatan *input*, dan peralatan *output* yang memberikan informasi, serta bekerja secara otomatis”. Senada dengan Siallagan (2019:1), “Komputer adalah sebagai sekumpulan alat elektronik yang saling bekerja sama, dapat menerima data (*input*), mengolah data (*process*), memberikan informasi (*output*), dan terkoordinasi di bawah kontrol program yang tersimpan dalam memorinya”. Berdasarkan uraian diatas, dapat diambil benang merahnya bahwa definisi dari komputer adalah sebuah alat elektronik yang difungsikan guna memproses data digital dan informasi yang bermanfaat bagi *user* yang diharapkan dapat membantu terhadap pekerjaan manusia dengan terkontrol dan terhitung.

2.1.3. Pengertian Data

Menurut Ladjamudin (2017:9) disebutkan bahwa, “Data adalah kenyataan yang menggambarkan kejadian-kejadian dan kesatuan nyata”. Seirama dengan hal diatas, Tata Sutabri (dikutip Rusdiana dan Irfan 2014:68) menyatakan bahwa, “Data adalah bahan mentah yang diproses untuk menyajikan informasi”. Asropudin (2018:22) mengemukakan bahwa, “Data adalah kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti. Data dapat diolah sehingga menghasilkan informasi.” Maka dari uraian diatas dapat diambil kesimpulan bahwa Data adalah sekumpulan dari fakta-fakta yang diklasifikasikan sehingga nantinya akan membentuk sebuah informasi dengan teknik pengolahan tertentu.

2.1.4. Pengertian Basis Data (*Database*)

Rusdiana dan Irfan (2014:302) menyatakan bahwa, “*Database* adalah kumpulan informasi yang disimpan dalam komputer secara sistematis untuk memperoleh informasi dari basis data”. Senada dengan yang dikemukakan oleh Rusdiana dan Irfan (2019:304) bahwa, “Basis data adalah kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, dan dengan *software* untuk melakukan manipulasi untuk kegunaan tertentu”. Berdasarkan pendapat di atas,

dapat disimpulkan bahwa Basis Data merupakan kumpulan data terstruktur, terhubung dan tersimpan dalam sebuah media penyimpanan di dalam komputer.

2.2. Teori Khusus

Teori Khusus terdiri dari 4 bagian yaitu **2.2.1 Unified Modeling Language**, **2.2.2 Kategori dan Jenis UML (*Unified Modeling Language*)**, **2.2.3 Simbol - Simbol UML (*Unified Modeling Language*)** dan **2.2.4 Metode Pengembangan Sistem**. Berikut penjelasannya,

2.2.1. Unified Modeling Language

Braun, dkk. (Haviluddin, 2011:1) “*Unified Modelling Language (UML)* adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual”.

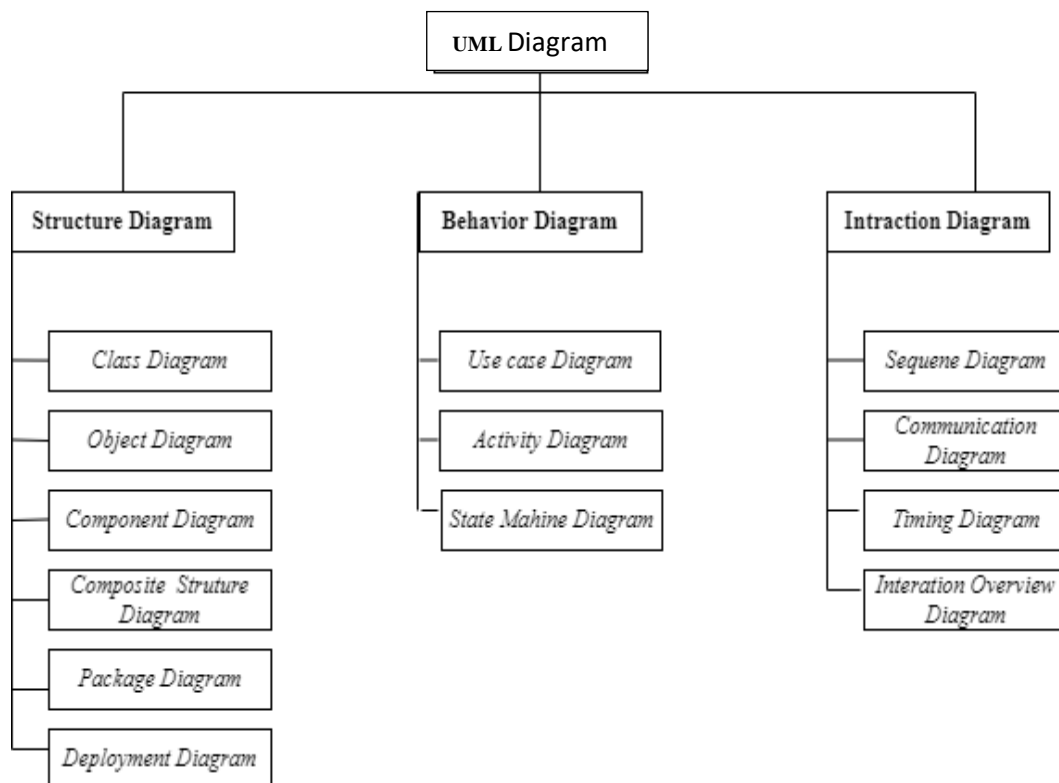


Gambar 2.1 Logo UML (*Unified Modelling Language*)

Sumber: mahesa.id

2.2.2. Kategori dan Jenis UML (*Unified Modeling Language*)

Haviluddin dalam jurnal penelitiannya yang berjudul Memahami Penggunaan UML (*Unified Modelling Language*), menyebutkan bahwa pada UML versi 2 terdiri atas tiga kategori dan memiliki 13 jenis diagram, yaitu :



Gambar 2.2 UML Diagram

a. Struktur Diagram

1) Class Diagram

Class diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas.

b. Behavior Diagram

2) Use Case Diagram

Diagram yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor.

3) Activity Diagram

Menggambarkan aktifitas-aktifitas, objek, state, transisi state dan event. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas.

c. Interaction Diagram

4) Sequence Diagram

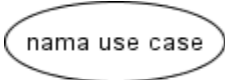
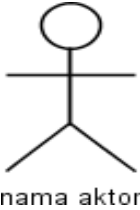
Diagram yang menjelaskan interaksi objek yang disusun berdasarkan urutan waktu.

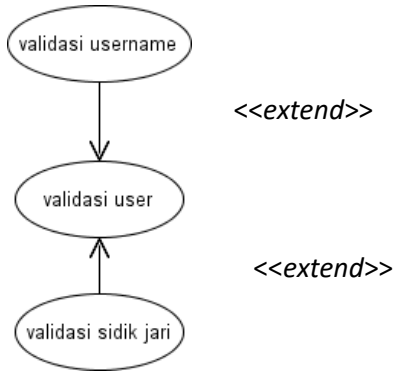
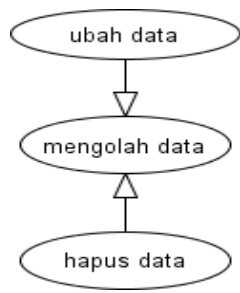
2.2.3. Simbol-Simbol UML (*Unified Modeling Language*)

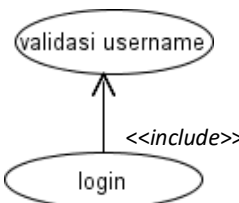
UML memiliki komponen yang akan digunakan untuk menggambarkan analisa dan desain diagram.

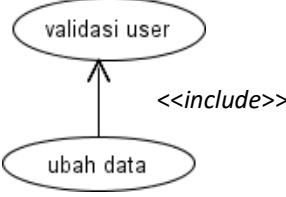
2.2.3.1 Use Case Diagram

Tabel 2.1 *Simbol-simbol Use Case Diagram*

No	Simbol	Deskripsi
1.		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal-awal frase nama <i>use case</i> .
2.		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor.
3.	<i>Asosiasi / Association</i>	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor.

4.	Ekstensi / <i>Extend</i> <code><<extend>></code>→	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang di tambahkan 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 yang sama dengan <i>use case</i> yang ditambahkan, misalnya</p>  <p>Biasanya, <i>use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
5.	Generalisasi/ <i>Generalization</i> ————→	<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> 


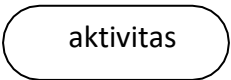

		<p><u>misalnya:</u></p> <p>arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum).</p>
6.	<p>Menggunakan/ <i>Include / Uses</i></p> <p>.....→</p> <p><<include>></p> <p><<uses>></p> <p>→</p>	<p>relasi tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>ada dua sudut pandang yang cukup besar mengenai include di <i>use case</i>:</p> <ul style="list-style-type: none">● <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu di panggil saat <i>use case</i> tambahan dijalankan, misalnya pada kasus berikut:  <pre>graph BT; login([login]) -- <<include>> --> validasi_username([validasi username]);</pre> <ul style="list-style-type: none">● <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang di tambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, misal pada kasus berikut :


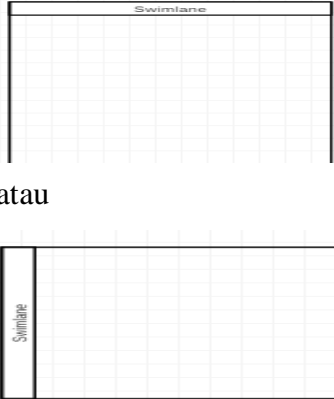
		 <p>kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>
--	--	---

Sumber: Sukamto RA dan Shalahuddin (2013:156-158)

2.2.3.2. Activity Diagram

Tabel 2.2 Simbol-Simbol Activity Diagram

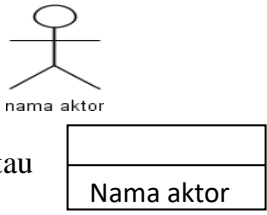

No	Simbol	Deskripsi
1.	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3.	Percabangan / <i>decision</i>	Asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu

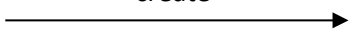
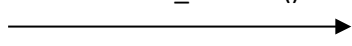
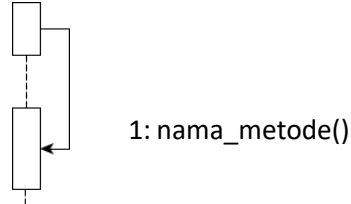
5.	<p>Status akhir</p> 	<p>Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir</p>
6.	<p>Swimlane</p>  <p>atau</p>	<p>Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi</p>

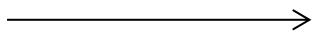
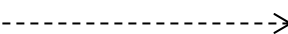
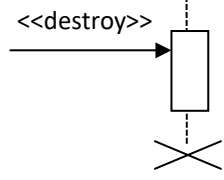
Sumber: Sukanto RA dan Shalahuddin (2013:162-163)

2.2.3.3. Sequence Diagram

Tabel 2.3 Simbol-simbol pada Sequence Diagram

No	Simbol	Deskripsi
1.	<p>Actor</p>  <p>nama aktor</p> <p>atau</p> <p>Nama aktor</p> <p>tanpa waktu aktif</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 nama aktor</p>
2.	<p>Garis hidup / <i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek.</p>

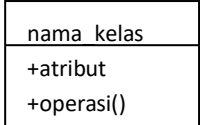
3.	<p>Objek</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> Nama objek : nama kelas </div>	<p>Menyatakan objek yang berinteraksi pesan.</p>
4.	<p>Waktu Aktif</p> <div style="border: 1px solid black; width: 30px; height: 70px; margin: 10px auto;"></div>	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya</p> <pre> sequenceDiagram participant Actor Actor->>Object: 1: login() activate Object Object->>Object: 2: cekStatusLogin() Object->>Object: 3: open() deactivate Object </pre> <p>maka cekStatusLogin () dan open() dilakukan di dalam metode login() aktor tidak memiliki waktu aktif</p>
5.	<p>Pesan tipe create</p> <div style="text-align: center; margin: 10px auto;"> <<create>>  </div>	<p>menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
6.	<p>Pesan tipe call</p> <div style="text-align: center; margin: 10px auto;"> 1: nama_metode()  </div>	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,</p>  <pre> sequenceDiagram participant Object Object-->>Object: 1: nama_metode() </pre> <p>arah panah mengarah pada objek yang memiliki operasi/metode, karena ini</p>

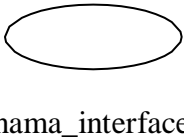

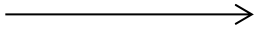
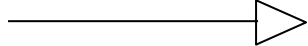


		memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi
7.	Pesan tipe send 1: masukan 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
8.	Pesan tipe return 1: keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
9.	Pesan tipe destroy <<destroy>> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i>

Sumber: Sukamto RA dan Shalahuddin (2013:165-167)

2.2.3.4. Class Diagram

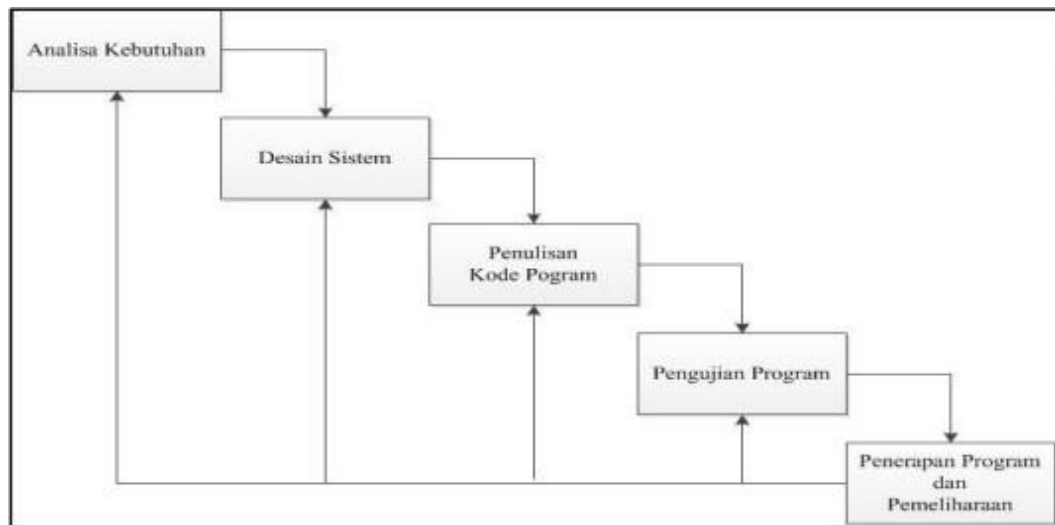
Tabel 2.4 Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
1.	kelas 	Kelas pada struktur sistem

2	antarmuka / interface 	Sama dengan konsep interface dalam pemrograman berorientasi objek
3.	asosiasi / association 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i>
4.	asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
5.	generalisasi 	Relasi antarkelas dengan makna generalisasi – spesialisasi (umum khusus)
6.	kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antar kelas
7.	agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)

Sumber: Sukanto RA dan Shalahuddin (2013:146-147)

2.2.4. Metode Pengembangan Sistem



Gambar 2.3 Ilustrasi Model *Waterfall*

Dalle, dkk (2020:398) menjelaskan tentang metode *waterfall* menyediakan pendekatan alur hidup software secara *sequential* atau terurut dimulai dari analisa, desain, pengkodean, uji coba, dan tahap pendukung. Jika langkah satu belum dikerjakan maka tidak akan bisa melakukan pengerjaan langkah 2, 3, dan seterusnya.

1. Analisa

Sebuah analisis persyaratan sistem yang dilakukan dengan cara wawancara atau literatur belajar. Menghasilkan data yang berkaitan dengan kebutuhan klien dalam proses pembuatan sistem atau dapat dikatakan dokumen persyaratan pengguna, yang merupakan tujuan dari sistem analisis untuk menerjemahkannya ke dalam bahasa pemrograman.

2. Desain

Proses ini menerjemahkan persyaratan ke dalam desain perangkat lunak yang dapat diperkirakan sebelum pengkodean dibuat. Berfokus pada: struktur data, arsitektur perangkat lunak, representasi antarmuka, dan detail prosedural. Menghasilkan sebuah dokumen yang disebut persyaratan perangkat lunak,

yang merupakan apa yang programmer gunakan dalam kegiatan manufaktur sistemnya.

3. Coding dan Testing

Terjemahkan desain ke dalam bahasa pemrograman sehingga dapat dibaca oleh komputer. Setelah coding tahap berikutnya, pengujian. Pengujian dalam hal ini memiliki manfaat menemukan kesalahan dengan sistem dan kemudian memperbaikinya.

4. Penerapan

Step ini bisa dikatakan akhir dalam pembuatan sebuah sistem. Sesudah melakukan analisa, design dan pengkodean maka sistem yang sudah jadi akan dipakai oleh user.

5. Pemeliharaan

Software yang sudah disampaikan kepada client pasti mengalami revisi. Revisi tersebut bisa karena mengalami kesalahan karena software harus menyesuaikan dengan daerah nya (*peripheral* atau sistem operasi baru) baru, atau karena client membutuhkan perkembangan fungsional.

2.3. Teori Judul

Teori Judul terdiri dari 4 bagian yaitu **2.3.1 Pengertian Aplikasi, 2.3.2 Pengertian Pendaftaran/Reservasi, 2.3.3 Pengertian Umroh, dan 2.3.4 Pengertian Web.** Berikut penjelasannya,

2.3.1. Pengertian Aplikasi

Sujatmiko (2019:23) menyatakan bahwa, “*Application* merupakan program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya *Ms. Office*”. Senada dengan pendapat Asropudin (2019:6) yaitu, “Aplikasi (*Application*) adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Ms-Word, Ms-Excel*”. Sedangkan menurut Mulyono (2020:109) mengemukakan bahwa, “Perangkat lunak yang khusus ditambahkan dalam sistem

operasi yang ada untuk melakukan pekerjaan khusus sesuai dengan kebutuhan pengguna komputernya disebut software aplikasi”. Berdasarkan pendapat di atas, maka dapat disimpulkan bahwa pengertian Aplikasi adalah program komputer yang berfungsi untuk membantu *user* mengerjakan dan menyelesaikan tugas-tugas tertentu dengan metode pemecahan tertentu.

2.3.2. Pengertian Pendaftaran/Reservasi

Suartana menyatakan bahwa, “*Booking* atau *Reservasi* adalah kata yang berasal dari bahasa Inggris yaitu “*toreserve*” yang berarti menyediakan atau mempersiapkan tempat atau fasilitas sebelumnya”

2.3.3. Pengertian Umroh

Umrah adalah salah satu kegiatan ibadah dalam agama Islam. Hampir mirip dengan ibadah haji, ibadah ini dilaksanakan dengan cara melakukan beberapa ritual ibadah di kota suci Makkah, khususnya di Masjidil Haram. Umrah dapat dilaksanakan sewaktu-waktu (setiap hari, setiap bulan, setiap tahun) dan hanya di Makkah.

2.3.4. Pengertian Web

Menurut pendapat Abdulloh (2019:1) menyatakan bahwa, “*Website* atau disingkat *Web* dapat diartikan sekumpulan halaman yang terdiri atas beberapa laman yang berisi informasi dalam bentuk data digital, baik berupa teks, gambar video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet”.

2.4. Teori Program

Teori Program terdiri dari 4 bagian yaitu **2.4.1 Basis Data (*Database*)**, **2.4.2 Web Editor**, **2.4.3 Bahasa Pemrograman**, dan **2.4.4 *Webserver***. Berikut penjelasannya,

2.4.1. Basis Data (*Database*)

Yakub (2022:55) menyatakan bahwa, “Basis data adalah koleksi dari data - data yang terorganisasi dengan cara sedemikian rupa sehingga data tersebut mudah

disimpan dan dimanipulasi”. Senada dengan yang dikemukakan oleh Indrajani (2019:70) bahwa, “Basis data adalah sebuah kumpulan data yang saling berhubungan secara logis dan merupakan sebuah penjelasan dari data tersebut, yang didesain untuk menemukan data yang dibutuhkan oleh sebuah organisasi. Sedangkan menurut Sujatmiko (2019:40) menyatakan bahwa, “Basis data (*database*) adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut”. Dari beberapa definisi di atas dapat disimpulkan bahwa basis data (*database*) merupakan wadah yang digunakan untuk menyimpan data-data agar lebih terorganisi serta terstruktur antara satu sama lain.

Dalam membangun aplikasi ini, penulis menggunakan sistem *database* MySQL sebagai tempat untuk menyimpan dan mengolah data aktivasi dan pembangunan.

2.4.1.1. Pengertian MySQL

Menurut Purbo (2020:1) menyatakan bahwa, ”MySQL merupakan salahsatu database relasional yang mendukung pemakaian SQL dan dirancang untuk penggunaan aplikasi dengan arsitektur client/server”. Kadir (2018:2), “MySQL (baca: mai-se-kyu-el) merupakan *software* yang tergolong sebagai DBMS (*Database Management System*) yang bersifat *Open Source*”. Winarno, dkk (2019:102) mengemukakan bahwa, “MySQL merupakan tipe data relasional yang artinya MySQL menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan”. Dari beberapa pendapat di atas dapat disimpulkan bahwa *MySQL* adalah sebuah aplikasi untuk membuat database yang siapa saja boleh menggunakannya (*open source*).

2.4.2 Web Editor

Penulis menggunakan Adobe Dreamwaver sebagai web editor dalam membangun Sistem Informasi Jadwal Booking Gedung Serbaguna (GSG) sebagai tempat pengkodean yang akan menghubungkan database agar terhubung dengan

program yang akan dibuat. Penulis menggunakan Adobe Dreamwaver karena Adobe Dreamwaver merupakan web editor yang memiliki banyak keunggulan dan juga fitur-fitur yang terdapat didalamnya mudah digunakan oleh penulis.

2.4.2.1. Adobe Dreamwaver

Madcoms (2019:10) mengemukakan bahwa, “*Dreamweaver* merupakan perangkat lunak yang ditujukan untuk membuat situs web”. Selain itu, Sadeli (2019:12-13) juga menyebutkan bahwa, “*Dreamweaver* merupakan suatu perangkat lunak web editor keluaran Adobe System yang digunakan untuk membangun dan mendesain suatu website dengan fitur-fitur yang menarik dan kemudahan dalam penggunaannya”. Hal yang senada juga dikemukakan oleh Madcoms (2022:2) bahwa, “*Dreamweaver* adalah sebuah HTML editor professional untuk mendesain web secara visual dan mengelola situs atau halaman web”.

2.4.3. Bahasa Pemrograman

Bahasa program adalah sekumpulan instruksi yang diberikan kepada komputer untuk melaksanakan tugas-tugas tertentu dalam menyelesaikan suatu permasalahan agar dapat mengolah data sesuai dengan langkah-langkah penyelesaian yang telah ditentukan oleh programmer.

2.4.3.1. HTML (*HyperText Markup Language*)

Winarno, dkk (2019:1) berpendapat bahwa, *HyperText Markup Language* (HTML) adalah sebuah bahasa menampilkan konten di Web. Selain itu, menurut Sunarfrihantono (2019:2) “HTML (*Hypertext Markup Language*) merupakan salah satu format yang digunakan dalam pembuatan dokumen dan aplikasi yang berjalan di halaman web”.

2.4.3.2 PHP (*Hypertext Preprocessor*)

Badiyanto (2019:32) menyebutkan bahwa, “PHP: *Hypertext Preprocessor* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML/PHP



banyak dipakai untuk membuat situs web dinamis”. Selain itu, Madcoms (2022:206) juga menjelaskan bahwa, “PHP (*Hypertext Preprocessor*) adalah salah satu bahasa pemrograman yang berjalan dalam sebuah webserver dan berfungsi sebagai pengolah data pada sebuah server”.

2.4.4. Webserver

Untuk bisa mengakses aplikasi web yang akan dibuat oleh penulis, maka diperlukan yang namanya *webserver*. Penulis menggunakan *Xampp* sebagai *webserver* dalam membangun Sistem Informasi Jadwal Booking.

2.4.4.1. Pengertian XAMPP

“*XAMPP* menyebutkan salah satu paket software web server yang terdiri dari *Apache*, *MySQL*, *PHP* dan *phpMyAdmin*” (Madcoms, 2022:1). Hal yang senada juga dikemukakan oleh Nugroho (2021:1) bahwa, “*Xampp* adalah paket program web lengkap yang dapat anda pakai untuk belajar pemrograman web, khususnya *PHP* dan *MySQL*, paket ini dapat didownload secara gratis dan legal”.