

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Penelitian Terkait**

Adapun Penelitian terkait ini, peneliti akan membandingkan hasil penelitiannya dengan dua jurnal yaitu :

1. Sri Mulyati dan Miftahur, H. (2018) dengan judul penelitian “Rancang Bangun Sistem Informasi Penyewaan *Wedding organizer* Berbasis Web Dengan Php Dan Mysql Pada Kiki Rias”. Kiki Rias *Wedding organizer* merupakan suatu pelayanan jasa pernikahan yang ada dikota Tangerang yang menyediakan beberapa jenis jasa pernikahan dan paket pernikahan yang pemesanannya masih secara manual. Hal ini menyebabkan sering dijumpai masalah calon pengantin tidak dapat menemui pemilik karena sedang tidak berada di tempat sehingga calon pengantin harus datang langsung ke Kiki Rias *Wedding organizer* untuk memesan atau sekedar mengetahui informasi layanan jasa yang disediakan pada *wedding organizer* tersebut, konfirmasi pembayarannya masih harus datang ketempat. Sistem ini menggunakan pengujian blackbox untuk mengetahui apakah data yang dimasukkan sudah benar atau belum serta bagaimana hasil keluarannya apakah sesuai dengan harapan atau tidak.
2. Dwik, dkk (2018) dengan judul penelitian “Perancangan Sistem Informasi Pemesanan *Wedding organizer* Di Kota Padang”. Menyimpulkan bahwa untuk membangun sistem informasi pemesanan *wedding organizer* yang menyajikan informasi mengenai jasa pernikahan, beserta paket yang dimiliki dan memberikan kemudahan untuk penyedia jasa dalam melakukan promosi paket pernikahan yang

dimiliki, serta memberikan kemudahan kepada pengguna dalam melakukan pemesanan paket pernikahan yang dibutuhkan tanpa harus mendatangi langsung lokasi penyedia jasa.

3. Menurut Ridwan, dkk (2019) dengan judul penelitian “Pengembangan Aplikasi Katalog *Wedding organizer* berbasis Android menggunakan *Restful Web Service* di Ruang Lingkup Kota Malang”. Perancangan basis data dibuat, agar nantinya lebih mudah dalam pengimplementasiannya dan juga agar basis data sesuai dengan kebutuhan sistem dan tidak menjauh dari tujuan utama sistem. Kemudian yang terakhir ialah perancangan antarmuka. Perancangan antarmuka pada sistem katalog *wedding organizer* dan *vendor* ini bertujuan untuk merancang sebuah antarmuka yang efektif untuk sistem dan mudah dioperasikan oleh pengguna. Antarmuka pada sistem ini dirancang agar pengguna dapat merasa nyaman untuk sering menggunakan sistem ini.
4. Menurut Asmara, dkk (2020) dalam penelitian yang berjudul “Perancangan Aplikasi Mobile Penentuan Jalur Tol dan Tarif Menggunakan Metode Breadth First Search” Diharapkan dengan aplikasi ini dapat mempermudah serta membantu pengguna jalan Tol untuk menentukan rute yang ingin ditempuh serta mengetahui jumlah tarif Tol keseluruhan yang akan di keluarkan hingga ke gerbang Tol tujuan. Dalam algoritma *Breadth First Search*, simpul anak yang telah dikunjungi disimpan dalam suatu antrian. Antrian ini digunakan untuk mengacu simpul-simpul yang bertetangga dengannya yang akan dikunjungi kemudian sesuai urutan pengantrian.
5. Menurut Galih dan Fachruddin (2020) dengan judul penelitian “Perancangan Sistem Informasi Pemesanan Jasa Pada Choki Al-Huda Wedding Decoration and

Catering Kota Jambi Berbasis Web”. Choki Al-Huda Wedding Decoration and Catering merupakan salah satu perusahaan *Wedding organizer* (WO) yang bergerak dalam bidang jasa pelayanan paket pernikahan yang hanya berfokus pada decoration dan cateringnya saja. Penelitian ini bertujuan untuk membuat sebuah sistem informasi berbasis *website* yang dimana dapat membantu mempermudah sistem pengolahan data Choki Al-Huda Wedding Decoration and Catering dari setiap permasalahan yang ada. Pembuatan *website* ini juga menjadi sebuah sarana bagi pengunjung dan pelanggan untuk dapat melakukan pemesanan jasa tanpa harus datang langsung ke tempat.

## **II.2. Studi Literatur**

### **II.2.1. *Wedding organizer***

*Wedding organizer* (WO) merupakan ”bisnis jasa yang memberikan pelayanan jasa pengorganisasian segala aktivitas yang berkaitan dengan kebutuhan dalam suatu pesta pernikahan. Bisnis ini muncul karena adanya peluang dalam kehidupan modern yang menginginkan kecepatan, kemudahan, dan kepraktisan untuk mengatasi masalah masyarakat perkotaan” (Sutan Malenggan: 2016)

### **II.2.2. *Vendor***

*Vendor* adalah penjual. Namun, *vendor* memiliki artian yang lebih spesifik yakni pihak ketiga dalam supply chain istilah dalam industri yang menghubungkan produk dari produsen untuk sampai ke tangan customer yang menjual barang kepada perusahaan untuk dijual kembali atau dipergunakan oleh user dari perusahaan tersebut. Dalam hal ini, *vendor* dapat pula di sebut supplier dari produk

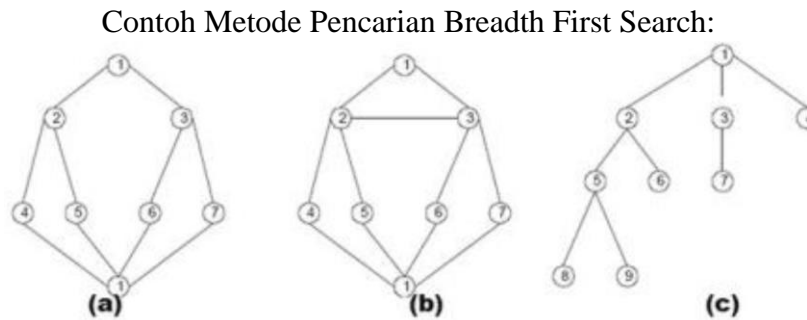
atau jasa. Kegiatan pengadaan adalah usaha untuk mendapatkan barang berupa material dan peralatan dan atau jasa dari pihak luar untuk proyek . Proses bisnis pengadaan barang terdiri dari beberapa aktivitas. Salah satunya adalah pengelolaan *vendor*. Pengelolaan *vendor* adalah aktivitas yang dilakukan oleh perusahaan untuk memastikan *vendor* mampu memenuhi target pengadaan barang sehingga kegiatan bisnis yang berkaitan dengan *vendor* dapat lebih cermat untuk mendukung jalannya kegiatan bisnis perusahaan dan mempererat hubungan bisnis dengan *vendor*. *Vendor* yang baik yaitu *vendor* yang mempunyai standar kriteria yang telah ditetapkan. Penetapan *vendor* merupakan bagian dari aktivitas pengelolaan *vendor* (Ricec dan Shanti : 2018)

### **II.2.3. BFS (BREADTH FIRST SEARCH)**

Breadth First Search adalah salah satu algoritma pencarian dalam pohon yang akan membangkitkan semua anak dari suatu simpul terlebih dahulu (Dicky Andrian : 2017). Untuk memperjelas cara kerja algoritma Breadth First Search beserta antrian yang digunakannya, berikut langkah-langkah algoritma Breadth First Search (Mustika) :

- a. Masukkan simpul ujung (akar) ke dalam antrian
- b. Ambil simpul dari awal antrian, lalu cek apakah simpul merupakan solusi.
- c. Jika simpul merupakan solusi, pencarian selesai dan hasil dikembalikan.
- d. Jika simpul bukan solusi, masukkan seluruh simpul yang bertetangga dengan simpul tersebut (simpul anak) ke dalam antrian.
- e. Jika antrian kosong dan setiap simpul sudah dicek, pencarian selesai dan mengembalikan hasil solusi tidak ditemukan.

f. Ulangi pencarian dari langkah kedua.



Gambar II.1 Contoh Metode Breadth First Search Maka penyelesaiannya adalah:

Gambar (a) : 1, 2, 3, 4, 5, 6, 7, 1

Gambar (b) : 1, 2, 3, 4, 5, 6, 7, 1

Gambar (c) : 1, 2, 3, 4, 5, 6, 7, 8, 9

#### II.2.4. Web

*Website* atau web merupakan suatu dokumen berupa sekumpulan halaman yang berisi berbagai informasi berbentuk digital. Informasi itu bisa berupa teks, gambar, animasi, video atau gabungan dari semuanya yang disediakan melalui internet dan dapat diakses oleh banyak orang di seluruh dunia selama memiliki koneksi internet. Kumpulan dari halaman – halaman situs, yang terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam *World Wide Web* ( WWW ) di dalam Internet. Sebuah halaman web biasanya berupa dokumen yang ditulis dalam format HTML ( *Hyper Text Markup Language* ), yang selalu bisa diakses melalui HTTP, yaitu sebuah protokol yang menyampaikan informasi dari server *website* untuk ditampilkan kepada para pemakai melalui web browser (Bilal Syaid : 2021)

## II.2.5. Bahasa Pemrograman

Bahasa pemrograman yang dipakai untuk membangun aplikasi system informasi adalah HTML dan *PHP*. Kombinasi kedua bahasa pemrograman *web* tersebut akan membuat tampilan halaman *web* menjadi lebih dinamis.

### a. *Hyper Text Markup Language (HTML)*

*Hypertext Markup Language (HTML)* adalah bahasa markah standar untuk dokumen yang dirancang untuk ditampilkan di peramban internet. Ini dapat dibantu oleh teknologi seperti *Cascading Style Sheets (CSS)* dan bahasa scripting seperti *JavaScript* dan *VBScript*. Peramban internet menerima dokumen *HTML* dari server web atau dari penyimpanan lokal dan membuat dokumen menjadi halaman web multimedia.

*HTML* menggambarkan struktur halaman web secara semantik dan isyarat awal yang disertakan untuk penampilan dokumen.

Elemen *HTML* digambarkan oleh *tag*, ditulis menggunakan tanda kurung

sudut. Tag seperti

```
<img />
```

```
<input />
```

```
<p>
```

dan langsung perkenalkan konten kedalam halaman. Tag lain seperti mengelilingi dan memberikan informasi tentang teks dokumen dan mungkin menyertakan tag lain sebagai sub-elemen. Peramban tidak menampilkan tag *HTML*, tetapi menggunakannya untuk menafsirkan konten halaman. *HTML* dapat menyematkan program yang ditulis dalam bahasa scripting seperti *JavaScript*, yang memengaruhi perilaku dan konten halaman web. Dimasukkannya *CSS* mendefinisikan tampilan dan tata letak konten. World **W**ide Web Consortium

(W3C), mantan pengelola HTML dan pemelihara standar CSS saat ini, telah mendorong penggunaan CSS pada HTML presentasi eksplisit sejak 1997. (wikipedia : 2020)

#### ***b. PHP***

PHP atau kependekan dari Hypertext Preprocessor adalah salah satu bahasa pemrograman open source yang sangat cocok atau dikhususkan untuk pengembangan web dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, Java, dan Perl serta mudah untuk dipelajari (Astria Firman, 2016 : 30).

*PHP* merupakan aplikasi *web* yang dinamis, dengan menggunakan perangkat lunak tambahan, perubahan informasi dalam halaman-halaman *web* dapat ditangani melalui perubahan data, bukan melalui perubahan program.

Sebagai implementasinya, aplikasi *web* dinamis dapat dikoneksikan ke basis data. Dengan demikian perubahan informasi dapat dilakukan oleh operator atau yang bertanggung jawab terhadap kemutakhiran data dan tidak menjadi tanggung jawab pemrogram atau *webmaster*.

#### ***II.2.6. Basis Data / Database***

Basis data terdiri atas dua kata, yaitu Basis dan Data. Basis kurang lebih diartikan sebagai markas atau gedung, tempat bersarang atau berkumpul. Data adalah *representasi* fakta dunia nyata yang mewakili suatu obyek seperti manusia, hewan, tumbuhan, benda, kondisi, peristiwa, waktu, tempat dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, suara atau kombinasi dari bentuk-bentuk tersebut. Pada penelitian kali ini penulis menggunakan *MySQL*

sebagai wadah untuk penyimpanan data.

### **II.2.7. PhpMyAdmin**

Menurut Firdaus (2007:15), *Phpmyadmin* adalah suatu program *open source* yang berbasis *web* yang dibuat menggunakan aplikasi *PHP*. Program ini digunakan untuk mengakses *Database MySQL*. Kemudahan dalam pengolahan *Database* dengan *interface web* yang familiar menjadi kelebihan tersendiri *Phpmyadmin*. Sampai saat ini *Phpmyadmin* terus dikembangkan dengan penambahan *skin* (model tampilan), penambahan bahasa, dan pengurangan *bug* (kesalahan dalam program yang mengakibatkan jalannya program menjadi kurang baik).

*Feature* atau fasilitas pada *Phpmyadmin* antara lain :

- a. Membuat dan menghapus *Database*
- b. Membuat, menyalin, menghapus, mengubah nama dan *struktur table*
- c. Menghapus, menambah, dan mengedit *field*
- d. Mengeksekusi *statemen SQL*
- e. Mengekspor data ke format *CSV, XML, dan Latex*
- f. Menangani beberapa atau *multiple server*
- g. Memanajemen *user* dan *privilege* (hak akses) pada *MySQL*
- h. Mengecek *referential integrity*
- i. Membuat *PDF graphics* dan *Database layout*
- j. Mampu mencari data dalam suatu *Database*

### **II.2.8. XAMPP**

Menurut Frangky Rawung (2017;7) XAMPP adalah perangkat lunak bebas

(*free software*), yang mendukung untuk banyak sistem operasi, dan merupakan kompilasi dari beberapa program. Fungsi XAMPP sendiri sebagai server yang berdiri sendiri (*localhost*), yang terdiri dari beberapa program, antara lain *Apache HTTP Server*, *MySQL Database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan *Perl*.

Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP*, dan *Perl*. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan *web server* yang mudah digunakan dan dapat menampilkan halaman web yang dinamis.

Komponen dan fungsi XAMPP dijelaskan pada penjelasan berikut ini :

- 1) ***Htdoc*** adalah folder dimana dapat meletakkan file yang akan dijalankan, seperti *file* PHP, HTML, dan *script* lainnya.
- 2) ***phpMyAdmin*** adalah bagian untuk mengelola *Database MySQL* yang ada di komputer. Untuk membukanya, buka *browser* dan ketik alamat <http://localhost/phpMyAdmin>, halaman *phpMyAdmin* akan muncul.
- 3) ***Control panel***, yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti *stop service* (berhenti) atau *start* (mulai).

### ***II.2.9. UML (Unified Modeling Language)***

UML (Unified Modeling Language) adalah Spesifik standar yang dipergunakan untuk mendokumentasikan, menspesifikasi dan membangun aplikasi. UML merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek”

(OOP). Namun demikian model - model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain: *Usecase Diagram*, *Sequence Diagram*, *Activity Diagram* dan *Class Diagram* (Sri Mulyani 2016 ; 35).

### **1. Tujuan Pemanfaatan UML**

Tujuan dari penggunaan diagram seperti diungkapkan oleh SchullerJ. (2004), "*The purpose of the diagram is to present multiple views of a system; this set of multiple views is called a model*". Berikut tujuan utama dalam desain UML adalah (Haviluddin; 2015:4) :

- a. Menyediakan bagi pengguna (analisis dan desain sistem) suatu bahasa pemodelan *visual* yang ekspresif sehingga mereka dapat mengembangkan dan melakukan pertukaran model data yang bermakna.
- b. Menyediakan mekanisme yang spesialisasi untuk memperluas konsep inti.
- c. Karena merupakan bahasa pemodelan visual dalam proses pembangunannya maka UML bersifat independen terhadap bahasa pemrograman tertentu.
- d. Memberikan dasar formal untuk pemahaman bahasa pemodelan.
- e. Mendorong pertumbuhan pasar terhadap penggunaan alat desain sistem yang berorientasi objek (OO).
- f. Mendukung konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola dan komponen terhadap suatu sistem.
- g. Memiliki integrasi praktik terbaik.

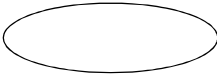

### **2. Struktur Diagram**



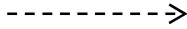
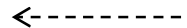
Menggambarkan elemen dari spesifikasi dimulai dengan kelas, obyek, dan hubungan mereka, dan beralih ke dokumen arsitektur logis dari suatu sistem. Struktur diagram dalam UML terdiri atas:

**a. Use Case Diagram**

*Use case* diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case* diagram dapat dilihat pada table II.1 di bawah ini :

**Tabel II.1. Simbol Use Case**

Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i>.</p>
	<p>Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam</p>



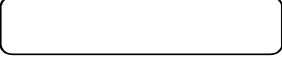
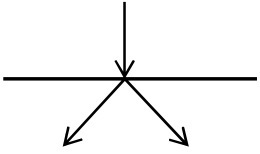
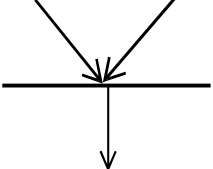
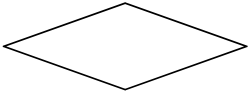

	beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i> , tetapi tidak memiliki control terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>use case</i> lain ( <i>required</i> ) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain,
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain.

(Sumber : Urva ; 2015)

### ***b. 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* merupakan *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*) (Dharwiyanti ; 2003).

**Tabel II.2 Simbol Activity Diagram**

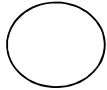
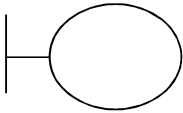


Gambar	Keterangan
	<p><i>Start point</i>, diletakkan pada pojok kiri atas dan merupakan awal aktifitas.</p>
	<p><i>End point</i>, akhir aktifitas.</p>
	<p><i>Activites</i>, menggambarkan suatu proses/kegiatan bisnis.</p>
	<p><i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.</p>
	<p><i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.</p>
	<p><i>Decision Points</i>, menggambarkan pilihan untuk pengambilan keputusan, <i>true</i>, <i>false</i>.</p>
	<p><i>Swimlane</i>, pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.</p>

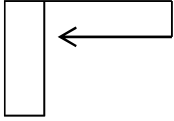


(Sumber : Urva ; 2015)

### c. *Sequence Diagram*

Menggambarkan interaksi antar objek didalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence* diagram digunakan untuk 9 menggambarkan *scenario* atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan output tertentu (Dharwiyanti ; 2003).

**Tabel II.3 Simbol *Sequence Diagram***

Gambar	Keterangan
	<i>EntityClass</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .

	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri</p>
	<p><i>Activation</i>, <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Urva ; 2015)

#### d. *Class Diagram*

Adalah sebuah spesifikasi yang jika diintansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan deain *beorientasi objek*. *Class* menggambarkan keadaan (*atribut/properti*) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (*metode/fungsi*) (Dharwiyanti ; 2003).

**Tabel II.4 Tabel *Multiplicity Class Diagram***

<b><i>Multiplicity</i></b>	<b>Penjelasan</b>
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih

0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Urva ; 2015)