

## BAB II LANDASAN TEORI

### 2.1. Tinjauan Studi

Tinjauan studi dilakukan terhadap beberapa penelitian sebelumnya sebagai bahan referensi, Penelitian tersebut antaranya:

Euis Nurpatonah (2015), melakukan studi "*Sistem Informasi Pemesanan Wedding Organizer Berbasis Web pada Java Exist Management*". Masalah penelitian yang diajukan yaitu pemesanan paket pernikahan di Java Exist Management masih dilakukan secara manual. Klien harus datang ke kantor langsung untuk melakukan pemesanan paket yang disebabkan keterbatasan waktu. Metode yang digunakan adalah pengembangan sistem informasi berbasis *WEB* dengan menggunakan bahasa pemrograman *PHP* dan database *MySQL* yang bertujuan untuk memberikan segala informasi mengenai paket-paket pernikahan yang ditawarkan dari Wedding Organizer Java Exist Management [3].

Rony Adi Nugroho (2015), melakukan studi "*Sistem Informasi Pemesanan Jasa Fotografi Berbasis Web pada Karma Kreatif Semarang*". Karma Kreatif Semarang adalah perusahaan jasa yang bergerak dibidang jasa fotografi. Pada perusahaan masih memiliki beberapa permasalahan, salah satu permasalahannya adalah masih manualnya proses promosi pada Karma Kreatif Semarang, dengan menggunakan brosur dan kartu nama yang disebar dengan hanya dilengkapi dengan informasi alamat dan jasa Karma Kreatif Semarang, sehingga pelanggan harus mendatangi *workshop office* untuk informasi lebih lengkap. Pemilik memiliki beberapa kendala dalam menggunakan brosur dan kartu nama yaitu membutuhkan biaya yang cukup besar untuk membayar percetakan, dan memberikan upah untuk orang yang menyebarkan brosur. Dengan permasalahan tersebut maka perlu dibuat website pada Karma Kreatif Semarang untuk mempermudah proses promosinya. Pada proses pembuatan website, terlebih dahulu menganalisis

system yang ada di perusahaan, membuat desain system, desain struktur menu web dan desain interface dari web tersebut, dan setelah itu ditentukan pemakaian software. Pada aplikasi ini *software* yang digunakan adalah *Adobe Dreamware CS6* untuk pembuatan desain *interface*, MySQL sebagai database untuk menyimpan data dari situs dan skrip PHP sebagai Bahasa pemrogramannya [6].

Gina Diah Permatasari (2017), melakukan studi “*Sistem Informasi Pemesanan Jasa Fotografi Berbasis Web di Fournalight Photography*”. Fournalight Photography merupakan perusahaan yang bergerak di bidang fotografi, maka dalam mengikuti perkembangan jaman Fournalight Photography harus memanfaatkan teknologi informasi, agar mampu bersaing dengan perusahaan lainnya, akan tetapi sistem yang berjalan di Fournalight Photography meliputi proses promosi, pemesanan jasa, penjadwalan fotografer, dan penggajian karyawan dirasa perusahaan kurang maksimal. Berdasarkan permasalahan diatas maka perlu dibuat suatu perancangan sistem informasi pemesanan jasa fotografi yang terkomputerisasi. Diharapkan kegiatan pemesanan jasa fotografi seperti promosi, pemesanan jasa, manajemen proyek dan penggajian dapat berjalan lebih baik. [7].

## **2.2. Tinjauan Pustaka**

### **2.2.1. Sistem Informasi**

Sistem Informasi adalah sistem yang menghasilkan informasi yang dapat berguna. System informasi adalah suatu rangkaian informasi yang didalamnya terdapat bagian-bagian yang berhubungan dan betegantungan satu sama lain, mulai dari bagian-bagian besar ke bagian yang terkecil, yaitu mulai dari sub, sub-sub, sub-sub-sub, dan seterusnya sampai yang terkecil [3].

### 2.2.2. CodeIgniter

*CodeIgniter* adalah *framework* pengembangan aplikasi dengan menggunakan *PHP*, suatu kerangka pembuatan dengan menggunakan *PHP*, pengembang dapat langsung menghasilkan program dengan cepat dan efisien, dengan mengikuti kerangka kerja yang telah disiapkan oleh *framework CodeIgniter* ini. *Framework CodeIgniter* merupakan *framework* yang memiliki dokumentasi yang jelas dan lengkap, yang memudahkan pengembang untuk mempelajari dengan mudah. Pendekatan *CodeIgniter* sangatlah mudah, dari membuat sekedar tulisan sampai dengan yang kompleks. *CodeIgniter* cukup dengan satu file dan satu prosedur atau method. Di dalam *CodeIgniter* terdapat metode, yaitu *MVC*. *MVC* adalah sebuah pola rancangan yang memungkinkan para pengembang memisahkan kode program mereka ke dalam tiga kelompok, yaitu *Model*, *View*, dan *Controller*.

Fitur dari *CodeIgniter* adalah sistem berbasis *MVC*, benar-benar *framework* yang ringan, memiliki fitur *class database* yang mendukung beberapa *platform*, dukungan *database* dengan *active record*, *form* dan validasi data, keamanan dan *filtering XSS*, manajemen sesi, *class* untuk pengiriman email yang mendukung lampiran, email dengan HTML/Teks, menggunakan berbagai protocol, librari manipulasi *image*, mendukung GD, *ImageMagick*, NetPBM, *class* untuk *upload file*, *class* FTP, lokalisasi, paginasi, enkripsi data, *benchmarking*, *full page caching*, *logging error*, *profiling* aplikasi, *class* calendar, *class* user agent, *class* encodingzip, *class* engine template, *class* trackback, librari XML-RPC, *class* unit pengujian, URL yang bersahabat dengan *search engine*, rotasi URI yang fleksibel, mendukung *hook* dan ekstensi *class*, dan librari fungsi “*helper*” yang banyak [10].

### 2.2.3. PHP

Secara umum PHP dikenal sebagai bahasa pemrograman *script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen HTML yang dibuat dengan *text editor* atau *editor* HTML. PHP juga dikenal sebagai bahasa pemrograman *server side*. [10].

### 2.2.4. Aplikasi

Aplikasi adalah suatu kelompok file (form, class, report) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait [10].

Aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya disandingkan dengan perangkat lunak system yang mengintegrasikan beberapa kemampuan komputer, tetapi tidak langsung menerapkan kemampuan tersebut untuk mengerjakan tugas yang menguntungkan pengguna. Contoh aplikasi adalah pengolah kata, lembar kerja, dan pemutar media. Beberapa aplikasi yang digabung bersama menjadi suatu paket kadang disebut sebagai suatu paket aplikasi. Contohnya adalah *Microsoft Office* yang menggabungkan suatu paket aplikasi pengolah kata, lembar kerja, serta beberapa 20 aplikasi lainnya. Contoh-contoh aplikasi yang lain adalah program pemroses kata dan web browser. Aplikasi akan menggunakan sistem operasi computer dan aplikasi yang lainnya yang mendukung [11]

### 2.2.5. Sublime Text 3

*Sublime text* adalah *text editor* berbasis *python*, sebuah *text editor* yang elegan, kaya fitur, cross, platform, mudah dan simpel yang cukup terkenal di kalangan *developer* (pengembang), penulis dan desainer. Para *programmer* biasanya menggunakan *sublime text* untuk menyunting *source code* web yang sedang ia kerjakan. Sampai saat ini *sublime text* sudah

mencapai versi 3. *Sublime text* mempunyai beberapa keunggulan yang dapat membantu pengguna dalam membuat sebuah *web development*, yaitu *Multiple Selection, Command Pallete, Distraction Free Mode, Find In Project, Plugin API Switch, Drag and Drop, Split Editing, Multi Platform*. [8].


### 2.2.6. UML



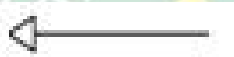


*UML* singkatan dari *Unified modeling Language* yang berarti bahasa pemodelan standart. Ketika membuat model dengan konsep UML ada aturan-aturan yang harus diikuti. Bagaimana elemen pada model-model yang dibuat berhubungan satu dengan yang lainnya harus mengikuti standart yang ada. UML bukan hanya sekedar diagram, tetapi juga menceritakan konteksnya [9].



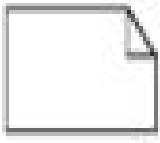
#### 1) Diagram Use Case

Menjelaskan manfaat system jika dilihat menurut pandangan orang yang diluar system (actor). Diagram ini menunjukkan fungsionalitas atau system atau kelas dan bagaimana system berinteraksi dengan dunia luar.

**Tabel 2. 1 Use Case Diagram**

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang dimainkan ketika berinteraksi dengan use case.
2		Dependency	Hubungan dimana terjadi pada suatu elemen mandiri akan


No	Gambar	Nama	Keterangan
			mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
3		Generalization	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk
4		Extend	Menspesifikasikan bahwa use case target memperluas perilaku yang dari usecase sumber pada titik yang diberikan.
5		Include	Menspesifikasikan bahwa Use case sumber secara Eksplisit
6		Assosiation	Apa yang dihubungkan antara objek satu dan objek lainnya
7		Sistem	Menspesifikasikan paket yang menampilkan system secara terbatas

No	Gambar	Nama	Keterangan
8		Use case	Deskripsi dari urutan-urutan aksi yang ditampilkan system yang menghasilkan suatu hasil yang terstruktur bagi actor
9		Collaboration	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah elemennya
10		Note	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

## 2) Activity Diagram

*Activity Diagram* merupakan diagram yang bersifat dinamis, Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek [9].


Tabel 2. 2 Simbol Activity Diagram

SIMBOL	KETERANGAN
	<b>Action state</b> , berfungsi menggambarkan keadaan elemen dalam suatu aktivitas
	<b>State</b> , berfungsi untuk menggunakan kondisi suatu elemen
	<b>Flow dan Control</b> , sebagai penghubung aliran aktivitas dari elemen satu ke elemen lain
	<b>Initial State</b> , menunjukkan titik awal dari suatu elemen
	<b>Final State</b> , menunjukkan titik akhir dari suatu elemen


### 3) Sequence Diagram

Sequence diagram adalah sebuah diagram yang menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dalam *use case*. *Sequence* diagram memperlihatkan tahap demi tahap apa yang sebenarnya terjadi untuk menghasilkan sesuatu di dalam *usecase*. Diagram *sequence* sebaiknya digunakan diawal tahap desain atau analisis karena kesederhanaan dan mudah untuk dimengeti.

Tabel 2. 3 Simbol-simbol Sequence diagram

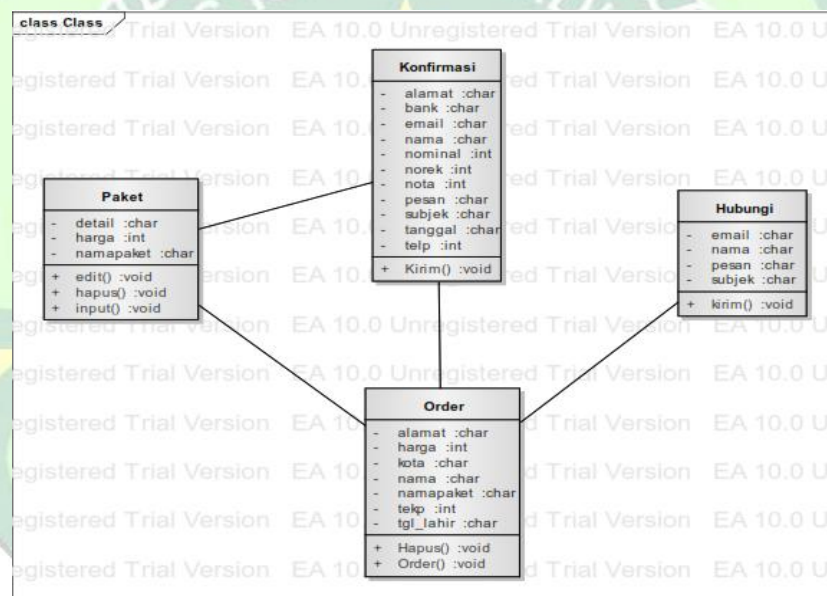
No	Simbol	Nama	Deskripsi
1		<i>Objek</i>	Menggambarkan objek yang mengirim dan menerima pesan



2		<i>Message</i>	Aliran pesan yang dikirim oleh objek
---	---	----------------	--------------------------------------

#### 4) Diagram Class

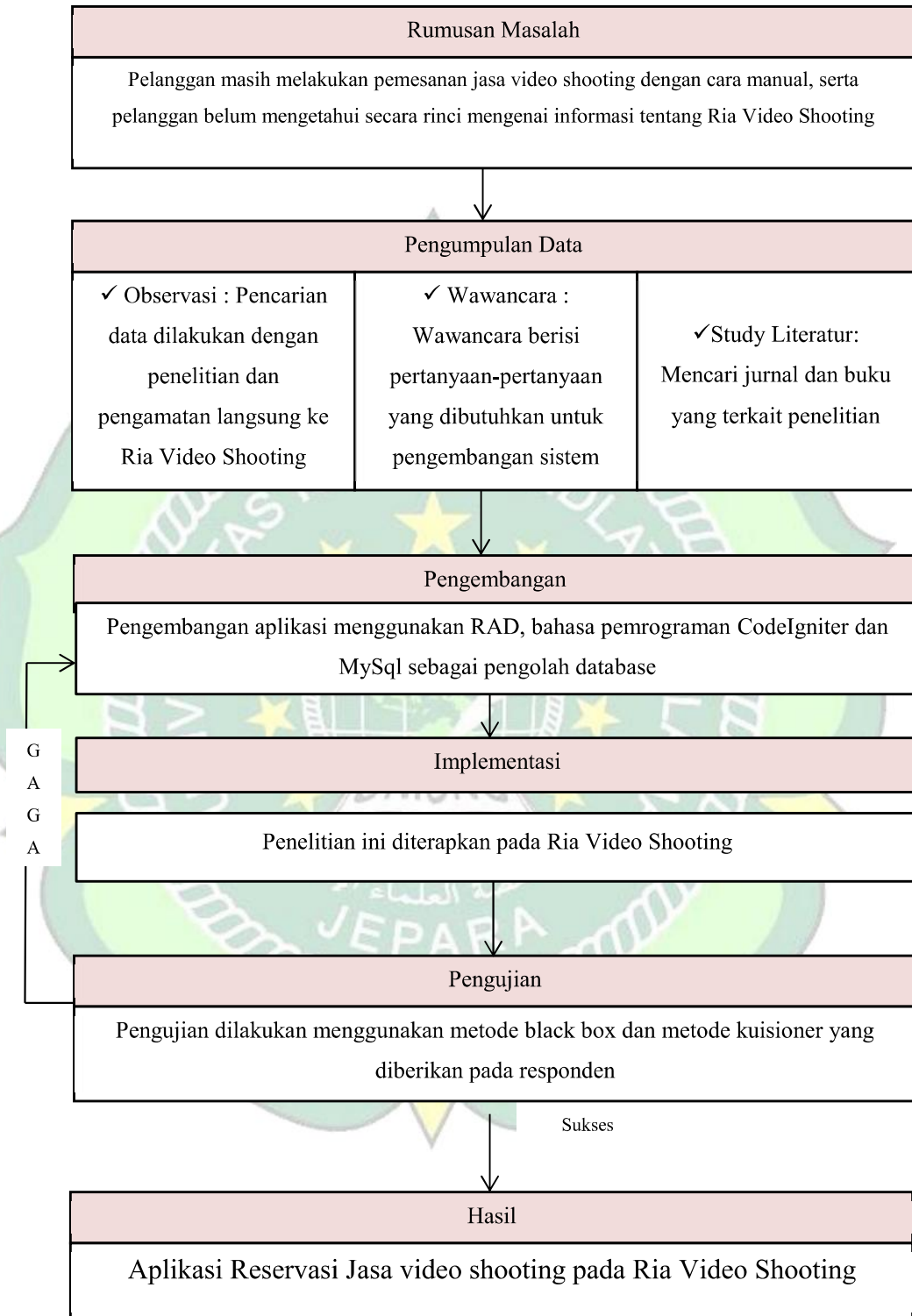
Dari gambar *Diagram Class* dibawah ini dapat dijelaskan bahwa *Class Diagram* mendeskripsikan jenis-jenis objek dalam system yang saling berelasi, dan nantinya digunakan untuk merancang database system. Class yang dibentuk yaitu Hubungi, Konfirmasi, Paket, dan Order [3].



Sumber: Sistem Informasi Pemesanan Wedding Organizer Berbasis Web Pada Java Exist Management, 2015

**Gambar 2. 1 Diagram Class**

## 5) Kerangka pemikiran



Gambar 2. 2 Kerangka Pemikiran