

## **BAB III**

### **ANALISIS DAN DESAIN SISTEM**

#### **III.1. Analisis Masalah**

Masalah utama yang didapat dari penelitian ini adalah bagaimana membuat sistem aplikasi pencarian dan penyewaan *wedding organizer* berbasis web dengan menerapkan metode *Breadth First Search* (BFS) untuk dijadikan inspirasi maupun pilihan bagi para calon pengantin, serta dapat digunakan sebagai media promosi untuk para vendor, dikarenakan mininya informasi yang didapat para calon pengantin mengenai *wedding organizer* baik lokasi maupun harga yang sesuai *budget* para calon. Oleh sebab itu maka peneliti ingin membuat sebuah sistem aplikasi pencarian dan penyewaan *wedding organizer* berbasis web guna untuk mempermudah para calon pengantin maupun para vendor, metode konvensional yang kurang efektif bagi para calon pengantin dalam mencari *wedding organizer* yang sesuai *budget* calon pengantin.

#### **III.2. Analisa Kebutuhan**

Untuk membuat sistem aplikasi pencarian dan penyewaan *wedding organizer* berbasis web maka dibutuhkan *hardware* dan *software* agar aplikasi ini dapat berjalan dengan lancar, maka perangkat lunak dan perangkat keras tersebut sebagai pendukung jalannya sistem yang akan dijalankan dengan baik dalam menggunakan aplikasi ini :

### **III.2.1. Perangkat Keras**

Adapun kebutuhan perangkat keras untuk menciptakan sistem aplikasi pencarian dan penyewaan *wedding organizer* berbasis web tersebut terdiri dari : Spesifikasi komputer yang digunakan minimal sebagai berikut :

- a. Laptop/ Computer
  - *Intel processor 2core*
  - *Memory 4 GB RAM*
  - *Hardisk 500 GB*
  - *Sistem Operasi Windows 10*

### **III.2.2. Perangkat Lunak**

- a. Macromedia Dreamweaver 8
- b. XAMPP

### **III.3. Desain Sistem**

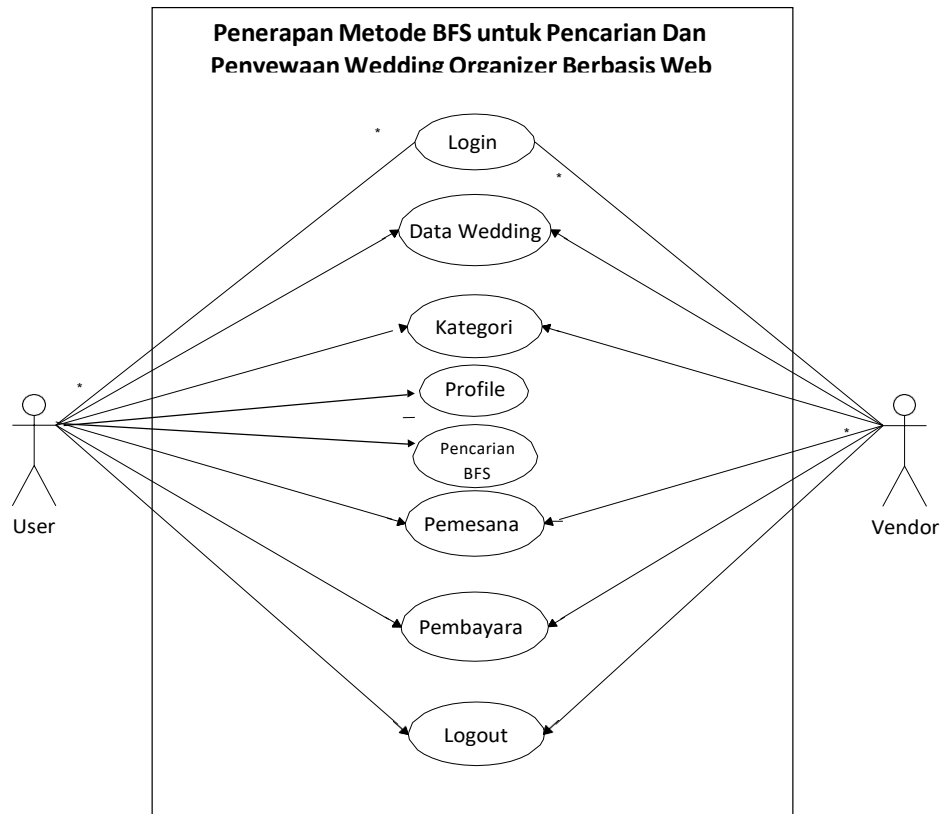
Desain sistem merupakan perancangan *website* menggunakan metode BFS dengan perancangan aplikasi ini terdiri dari tahap sebagai berikut:

1. Perancangan *Usecase Diagram*
2. Perancangan *Sequence Diagram*
3. Perancangan *Activity Diagram*

#### ***III.3.1. Use Case Diagram***

*Use case Diagram* adalah abstraksi dari interaksi antara sistem dan aktor. Use case bekerja dengan cara mendeskripsikan tipe interaksi antara user sebuah

sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Berikut use case diagram penerapan metode BFS untuk pencarian dan penyewaan wedding organizer berbasis web. dapat dilihat pada Gambar III.1. sebagai berikut:

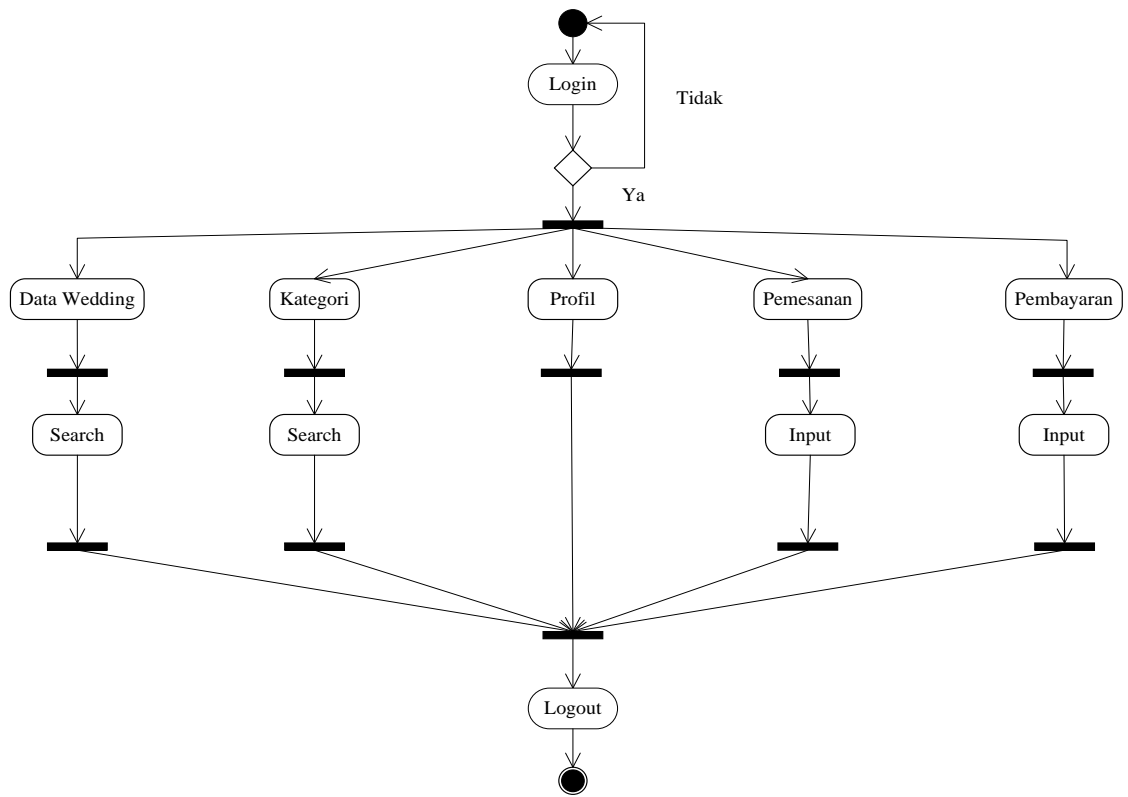


**Gambar III.1. Use Case Diagram Pencarian**

### ***III.3.2. Activity Diagram***

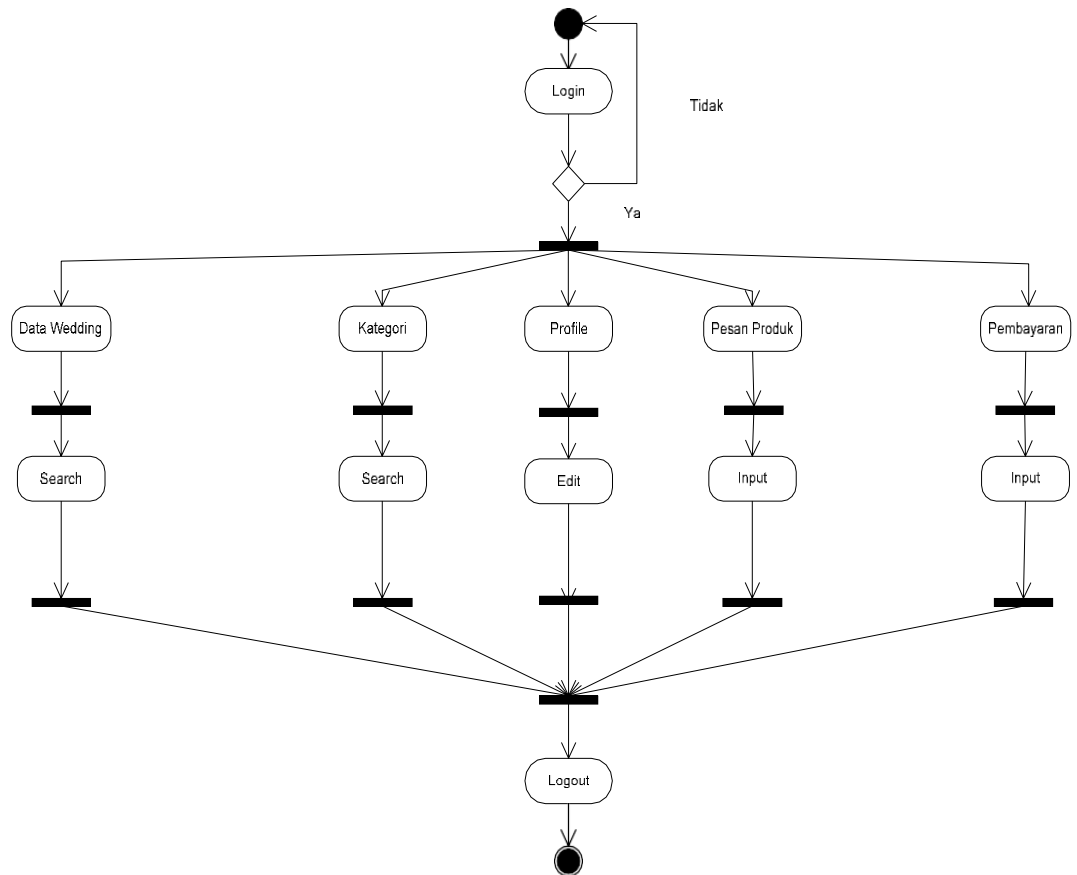
*Activity Diagram* menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *Use case* atau interaksi.

### III.3.2.1. Activity Diagram User



Gambar III.2 Activity Diagram User

### III.3.2.2. Activity Diagram Vendor

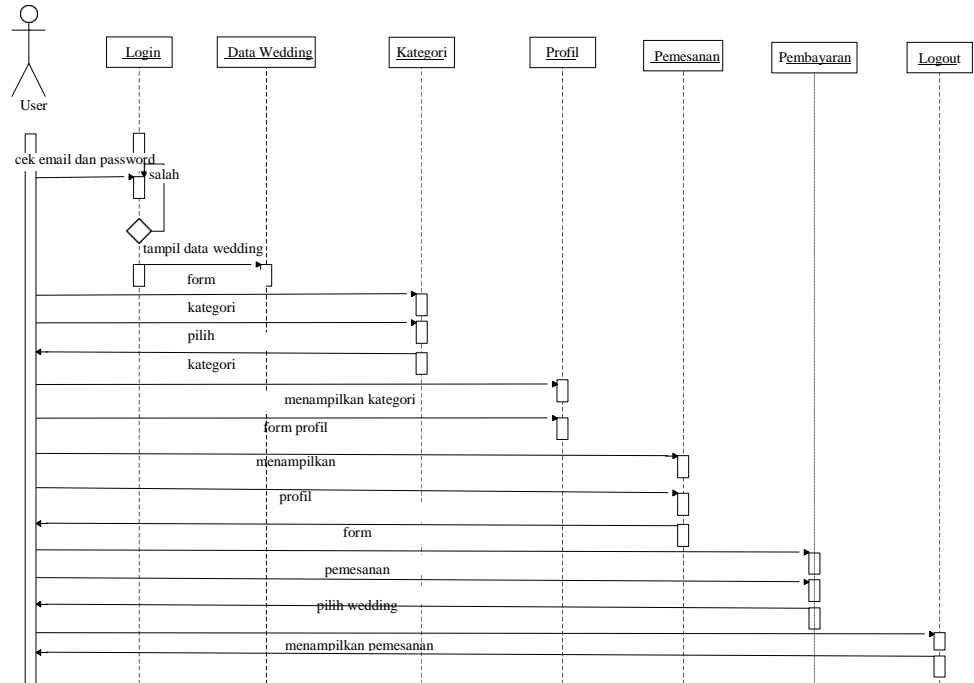


**Gambar III.3.Activity Diagram Vendor**

### III.3.3. Sequence Diagram

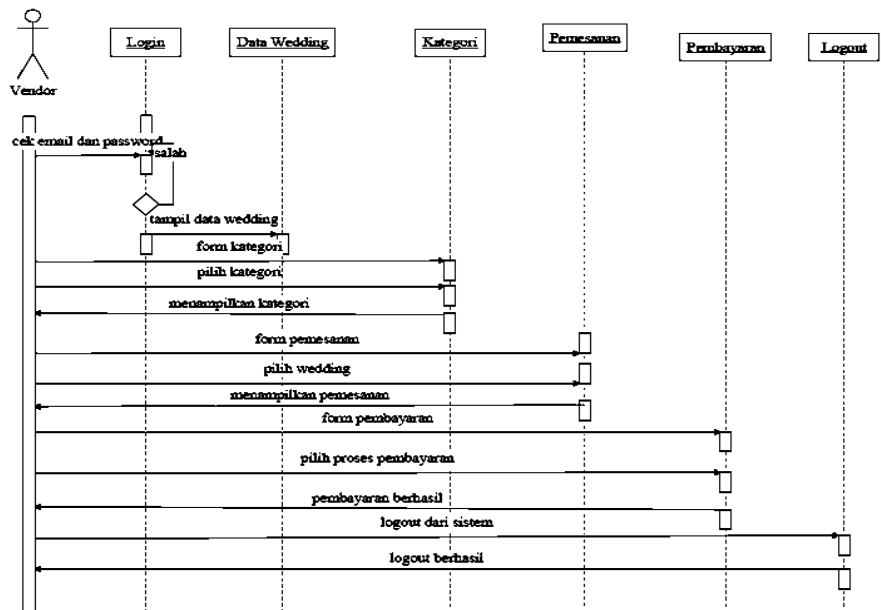
*Sequence Diagram* adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu.

### III.3.3.1. Sequence Diagram User



Gambar III.4. Sequence Diagram User

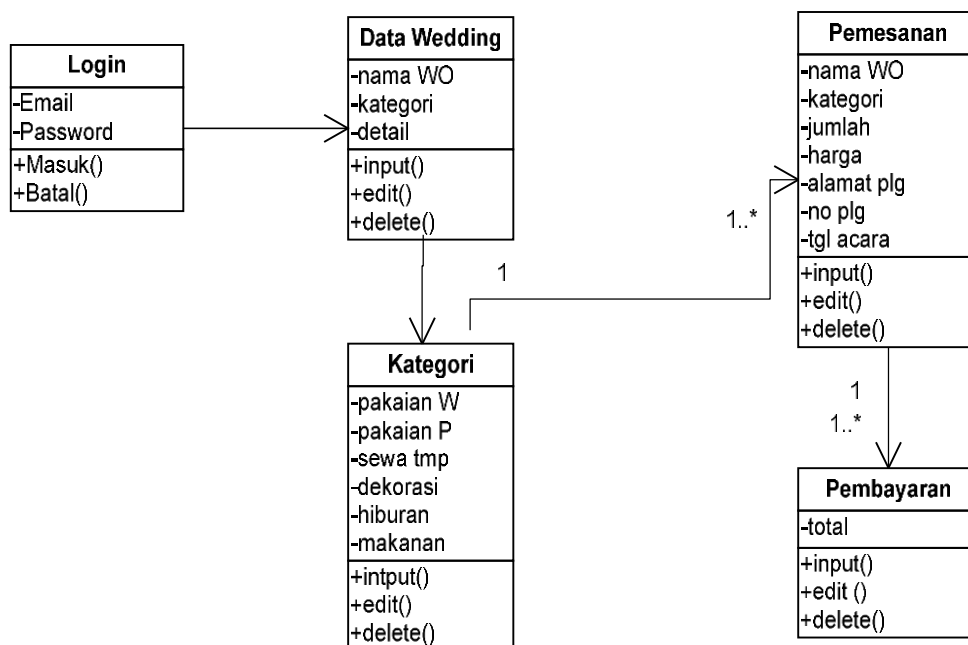
### III.3.3.2. Sequence Diagram Vendor



Gambar III.5. Sequence Diagram Vendor

### III.3.4. Class Diagram

*Class Diagram* digunakan untuk menggambarkan perbedaan yang mendasar antara *class-class*, hubungan antar-*class*, di mana subsistem *class* tersebut. Pada *class diagram* terdapat nama *class*, *attributes*, *operations*, serta *association* (hubungan antar-*class*). Adapun bentuk *class diagram* yang penulis rancang dapat dilihat sebagai berikut :



**Gambar III.6. Class Diagram**

### III.4.1. Normalisasi

Normalisasi adalah proses pembentukan struktur basis data sehingga sebagian besar *ambiguity* bisa dihilangkan. Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data relasional yang mengelompokkan atribut dari suatu tabel sehingga membentuk struktur tabel yang normal. Adapun kriteria

tabel dikatakan normal adalah ketika tidak ada kerangkapan data (redundansi data).

Tujuan dari normalisasi adalah untuk :

1. Untuk menghilangkan kerangkapan data sehingga meminimumkan pemakaian *storage* yang dipakai oleh *base relations* (file).
2. Untuk mengurangi kompleksitas.
3. Untuk mempermudah pemodifikasian data.

Adapun aturan dalam normalisasi adalah suatu tabel dikatakan baik (efisien) atau normal jika memenuhi 3 kriteria sebagai berikut

1. Jika ada dekomposisi (penguraian) tabel, maka dekomposisinya harus dijamin aman (*Lossless-Join Decomposition*). Artinya, setelah tabel tersebut diuraikan/didekomposisi menjadi tabel-tabel baru, tabel-tabel baru tersebut bisa menghasilkan tabel semula dengan sama persis.
2. Terpeliharanya ketergantungan fungsional pada saat perubahan data (*Dependency Preservation*).
3. Tidak melanggar *Boyce-Codd Normal Form* (BCNF).

A. Bentuk Normal Pertama / 1NF, memiliki aturan sebagai berikut :

1. Tidak adanya atribut *multi-value*, atribut komposit atau kombinasinya.
2. Mendefinisikan atribut kunci.
3. Setiap atribut dalam tabel tersebut harus bernilai *atomic* (tidak dapat dibagi lagi).

### **Tabel III.2. Bentuk 1NF**


Email	Alamat	No_hp
xxx	xxx	xxx

B. Bentuk Normal Kedua / 2NF, memiliki aturan sebagai berikut :

1. Sudah memenuhi dalam bentuk normal kesatu (1NF).
2. Semua atribut bukan kunci hanya boleh bergantung (*functional dependency*) pada atribut kunci.
3. Jika ada ketergantungan parsial maka atribut tersebut harus dipisah pada tabel yang lain.
4. Perlu ada tabel penghubung ataupun kehadiran *foreign key* bagi atribut - atribut yang telah dipisah tadi.

**Tabel III.3. Bentuk 2NF**

Email	Alamat	No_hp
xxx	xxx	xxx



Email	Alamat	No_hp	Nama_wo	Kategori_wo	Harga
xxx	xxx	xxx	xxx	xxx	xxx

C. Bentuk Normal Ketiga / 3NF, memiliki aturan sebagai berikut :

1. Sudah memenuhi dalam bentuk normal kedua (2NF).

2. Tidak ada ketergantungan transitif (dimana atribut bukan kunci tergantung pada atribut bukan kunci lainnya).

**Tabel III.4. Bentuk 3NF**

Email	Alamat	No_hp
xxx	xxx	xxx



Email	Alamat	No_hp	Nama_wo	Kategori_wo	Harga
xxx	xxx	xxx	xxx	xxx	xxx



Email	Alamat	No_hp	Jumlah	Total	Tgl_acara
xxx	xxx	xxx	xxx	xxx	xxx

### III.4.2. Desain Tabel

Selanjutnya yang dikerjakan yaitu merancang struktur tabel pada basis data sistem yang akan dibuat, berikut ini merupakan rancangan struktur tabel tersebut:

#### 1. Struktur Tabel *Login*

Tabel masuk digunakan untuk menyimpan data *email*, *password*, selengkapnya mengenai struktur tabel ini dapat dilihat pada tabel III.5 di bawah ini:

**Table III. 5 Table Masuk**

Nama Database	Wedding Organizer
---------------	-------------------

Nama Table		Login	
No.	Nama Field	Tipe Data	Primary Key
1.	Email	Varchar (50)	<input type="checkbox"/>
2.	Password	Varchar (20)	

## 2. Struktur Tabel Data Wedding

Tabel data wedding digunakan untuk menyimpan data nama wo, alamat wo, kategori wo, no. hp wo, *email*. selengkapnya mengenai struktur tabel ini dapat dilihat pada tabel III.6 di bawah ini:

**Table III. 6 Table Data Wedding**

Nama Database		Wedding Organizer	
Nama Table		Data Wedding	
No.	Nama Field	Tipe Data	Primary Key
1.	Nama_wo	Varchar (50)	<input type="checkbox"/>
2.	Alamat_wo	Varchar (20)	
3.	Kategori_wo	Varchar (50)	
4.	No_wo	Int (15)	
5.	Email_wo	Varchar (50)	

## 3. Struktur Tabel Kategori

Tabel kategori digunakan untuk menyimpan data pakaian wanita, pakaian pria, sewa tempat, dekorasi, hiburan, makanan. selengkapnya mengenai struktur tabel ini dapat dilihat pada tabel III.7 di bawah ini:

**Table III. 7 Table Kategori**

Nama Database	Wedding Organizer		
Nama Table	Kategori		
<b>No.</b>	<b>Nama Field</b>	<b>Tipe Data</b>	<b>Primary Key</b>
1.	Pakaian_w	Varchar (50)	
2.	Pakaian_p	Varchar (50)	
3.	Sewa_tmp	Varchar (50)	
4.	Dekor	Varchar (50)	
5.	Hiburan	Varchar (50)	
6.	Makanan	Varchar (50)	

#### 4. Struktur Tabel Pemesanan

Tabel pemesanan digunakan untuk menyimpan data pakaian wanita, pakaian pria, sewa tempat, dekorasi, hiburan, makanan. selengkapnya mengenai struktur tabel ini dapat dilihat pada tabel III.8 di bawah ini:

**Tabel III.8. Tabel Pemesanan**

Nama Database	Wedding Organizer		
Nama Table	Pemesanan		
<b>No.</b>	<b>Nama Field</b>	<b>Tipe Data</b>	<b>Primary Key</b>
1.	Nama_wo	Varchar (50)	
2.	Kategori	Varchar (50)	
3.	Harga	Varchar (50)	

4.	Jumlah	Varchar (50)	
5.	Alamat_pl	Varchar (50)	
2.	Np_pl	Varchar (50)	
2.	Tgl_acara	Varchar (50)	

### III.5. Desain Interface

Tahap perancangan berikutnya yaitu desain sistem secara detail yang meliputi desain *output* sistem, desain *input* sistem, dan desain *database*.

#### 1. Desain Form Login

Desain *Form* Login untuk melakukan pengolahan data login dapat terlihat seperti pada gambar III.7 berikut:

**Gambar III.7 Desain Form Login**

#### 2. Desain Form Home

Desain *Form* untuk melakukan pengolahan data home dapat terlihat seperti pada gambar III.8 berikut:

<b>WEDDING ORGANIZER</b>		
SEARCH		
GAMBAR 1	GAMBAR 2	GAMBAR 3
GAMBAR 4	GAMBAR 5	GAMBAR 6
HOME	KATEGORI	PAKETAN
PROFILE		

**Gambar III.8 Desain Form Home**

### 3. Desain Form Kategori

Desain *Form* Kategori untuk melakukan pengolahan data kategori dapat terlihat seperti pada gambar III.9 berikut:

<b>KATEGORI</b>		
SEARCH		
<i><b>PAKAJIAN WANITA</b></i>	<i><b>PAKAJIAN PRIA</b></i>	<i><b>SEWA TEMPAT</b></i>
<i><b>DEKORASI</b></i>	<i><b>HIBURAN</b></i>	<i><b>SOUVENIR</b></i>
HOME	PAKETAN	PROFILE

**Gambar III.9 Desain Form Kategori**

### 4. Desain Form Paketan

Desain *Form* untuk melakukan pengolahan data paketan dapat terlihat seperti pada gambar III.10 berikut:

**Gambar III.10 Desain Form Paketan**

### 5. Desain Form Pemesanan

Desain *Form* untuk melakukan pengolahan data pemesanan dapat terlihat seperti pada gambar III.12 berikut:

DAFTAR PESANAN						
NO.	NAMA PELANGGAN	PESANAN	JUMLAH	ALAMAT	NO. TELP	TANGGAL PELAKSANAAN

**Gambar III.12 Desain Form Pemesanan**