



# **BAB II**

# **TINJAUAN PUSTAKA DAN**

# **LANDASAN TEORI**

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Berdasarkan penelitian yang saya baca dari jurnal Fuad, H., Budiman, A., & Kurniasari, D. pada tahun 2018 mengenai Perancangan Sistem Informasi Pemesanan Paket Pernikahan Berbasis Web *Study Kasus Di Wedding Organizer PJ Management*. Penulis menuliskan beberapa masalah yang dialami pada PJ management diantaranya ialah konsumen yang akan memesan diharuskan datang ke tempat tersebut, kemudian untuk media promosi hanya dilakukan dari mulut ke mulut sehingga masyarakat yang mengenal PJ Management belum banyak. Hasil penelitian ini adalah membuat sebuah sistem informasi pemesanan berbasis web yang dapat digunakan oleh dua aktor yaitu admin dan member. Tools yang digunakan untuk membuat sistem tersebut berupa *XAMPP*, *PHP*, dan *Adobe Dreamweaver*. Sistem ini diharapkan dapat menghemat waktu dalam melakukan pemesanan, dan dapat menjadi media promosi [2].

Penelitian kedua yang saya baca dari Deti, N. pada tahun 2018. Sistem Informasi *Wedding Organizer Berbasis Web (Study Kasus: Cv. Yakin Group)*, sistem informasi ini dibuat dengan tujuan mempermudah para member melakukan pemesanan dibuat menggunakan metode prototype dengan kelebihan adanya komunikasi yang efektif antara pengembang dan *user* atau pengguna. sistem ini dibuat oleh penulis berlatar belakang karena pencatatan pemesanan pada cv tersebut masih sangat sederhana yaitu menggunakan buku besar dan ditulis menggunakan tangan, kemudian data- data yang ada masih belum terorganisir. Dengan adanya masalah keadaan yang seperti itu mengakibatkan tingkat kesalahan untuk pencatatan pemesanan relatif besar, dan laporan yang belum terorganisir dan masih ditulis menggunakan kertas, suatu saat dapat mengakibatkan data hilang. Hasil dari penelitian ini adalah penulis membuat sistem informasi *Wedding Organizer* berbasis web yang digunakan untuk membuat data laporan pemesanan terorganisir, sistem ini dibuat oleh penulis dengan menggunakan

bahasa *PHP* dan *MySQL*, dan pengujian sistem yang digunakan adalah *GUI*. Sistem ini dapat digunakan oleh 3 aktor yaitu konsumen, admin, dan pemilik [3].

Penelitian selanjutnya yang penulis baca dari Lusti, H., & Masya, F. (2020). Analisa Perancangan Sistem Informasi Pemesanan Pada *Wedding Organizer* Berbasis Web. Penelitian yang dilakukan oleh penulis berlatar belakang dari masalah konsumen yang sulit untuk menemui pemilik jasa *Wedding Organizer* tersebut karena si pemilik tidak selalu berada dirumah. Kemudian masalah lain yaitu para calon pengantin atau konsumen, jika ingin mencari informasi tentang layanan jasa tersebut harus datang ke lokasi. Sehingga penulis membuat sistem ini dengan harapan agar pemesanan layanan jasa tersebut lebih hemat waktu dan tenaga, serta dapat membantu pemilik untuk mendapatkan pelanggan baru dan mempermudah komunikasi antar pelanggan mengenai jasa yang akan dipesan. Sistem ini digunakan oleh 2 *user* yaitu *user* calon pengantin dan *user owner*. Calon pengantin dapat melakukan pendaftaran, melihat informasi paket, dan melakukan pemesanan. Owner dapat melakukan upload informasi paket pernikahan, mengelola pemesanan dan membuat laporan pemesanan[4].

Penelitian lain yang saya baca ialah dari penelitian yang hampir serupa oleh Cindy Rizky Syafrina tentang Sistem booking online pada Anisa Salon berbasis web. Hasil penelitian ini adalah mempunyai 2 pengguna yaitu Admin dan *user*. Sistem yang dibuat hanya digunakan untuk penyewaan jasa *Wedding Organizer* pada Anisa salon, pembayaran dilakukan secara tunai, serta sistem tidak membahas tentang penggajian karyawan. Sistem ini menggunakan metode pengembangan sistem *waterfall*, dan menggunakan metode pengujian sistem *blackbox*. Alat penelitian yang digunakan untuk penerapan coding menggunakan *Visual Studio Code*, dan menggunakan *MySQL 5.5.27* untuk mengelola *database* [5].

Berdasarkan beberapa penelitian yang sudah penulis baca, maka penulis akan mengembangkan sistem informasi pemesanan *Wedding Organizer* dengan metode *waterfall*, dengan menggunakan bahasa *PHP* dan *database MySQL* untuk membuat sistem yang dapat digunakan oleh admin, member vendor, keuangan, dan pemilik yang mencakup pengelolaan pemesanan jasa *Wedding Organizer* organizer, pengelolaan produk dan jasa yang ditawarkan, pengelolaan akun member/pelanggan dan vendor, dan pengelolaan

laporan transaksi penyewaan. Sistem yang akan dibangun diharapkan akan membantu pelanggan pada saat pemesanan agar menghemat waktu dan tenaga, dan membantu pemilik untuk mengelola bisnis yang sudah berjalan sebelumnya menjadi terorganisir dan terkomputerisasi

## **2.2 Landasan Teori**

Guna menunjang proses penelitian dibutuhkan teori-teori yang mendasar. Teori yang digunakan dalam penelitian ini adalah:

### **2.2.1 Sistem Informasi**

Sistem informasi adalah suatu proses kombinasi antara perangkat lunak, perangkat keras dan manusia yang dilakukan untuk mengolah data menjadi informasi yang digunakan untuk mendukung pengambilan keputusan dan kendali dalam suatu organisasi [6].

### **2.2.2 Pemesanan**

Pemesanan adalah sebuah perjanjian yang dilakukan oleh kedua belah pihak atau lebih untuk memberikan jasa atau produk dan untuk memakai jasa atau produk yang ditawarkan pada periode tertentu guna memenuhi kebutuhannya. Lanjutan dari pemesanan adalah pengiriman produk (produk) agar sampai ke tangan konsumen, atau melakukan tindakan berupa jasa secara langsung kepada konsumen [7].

### **2.2.3 WhatsApp**

WhatsApp merupakan sebuah aplikasi yang digunakan para user untuk mengirimkan pesan berupa text, foto, video, dokumen secara cepat dan aman. WhatsApp bersifat cross platform yang berarti meskipun user WhatsApp dari bermacam jenis ponsel, WhatsApp tetap dapat terhubung antar user [8].

### **2.2.4 Notifikasi**

Notifikasi menurut KBBI adalah pemberitahuan atau kabar tentang penawaran produk dan sebagainya. Notifikasi yang berhubungan dengan sistem dapat diartikan sebagai penyampaian pemberitahuan yang diberikan oleh sistem kepada pengguna [9].

### **2.2.5 Website**

Website adalah suatu halaman yang saling terhubung melalui jaringan internet, website merupakan layanan yang disediakan oleh

internet yang paling banyak digunakan karena memiliki kelebihan mampu menyajikan informasi berupa gambar, teks maupun video. Adapun kegunaan website adalah sebagai media penyedia informasi, sebagai media promosi, maupun sebagai media peningkatan pendapatan bagi perusahaan[10] .

### **2.2.6 Rekayasa Perangkat Lunak**

Rekayasa perangkat lunak (RPL) adalah disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahapan kebutuhan pengguna, menentukan spesifikasi kebutuhan pengguna, desain, pengkodean, pengujian hingga pemeliharaan setelah sistem digunakan tujuan dari rekayasa perangkat lunak adalah menciptakan perangkat lunak yang dapat memenuhi kebutuhan pengguna secara efektif dan efisien [11].

### **2.2.7 Metode Pengembangan Sistem**

Untuk mengembangkan sistem penulis menggunakan metode *waterfall*. Keuntungan menggunakan metode ini adalah tahapannya terstruktur sehingga kualitas perangkat lunak yang dibuat baik karena proses pengerjannya mendetail dan terperinci sehingga meminimalisir potensi kesalahan yang akan dilakukan. Model *Waterfall* adalah sebuah metode pengembangan sistem dengan pendekatan alur hidup perangkat lunak sekuensial atau secara berurutan. Dalam proses implementasi model *Waterfall* ini, sebuah langkah akan diselesaikan terlebih dahulu dari tahapan yang pertama sebelum melanjutkan ke tahapan yang berikutnya .Tahapan dari metode *waterfall* adalah sebagai berikut[1]:

#### **1. Requirement Definition**

Tahap ini merupakan tahap awal yang dilakukan untuk menentukan batasan, dan tujuan sistem yang ditentukan melalui konsultasi dengan *user* sistem. Persyaratan tersebut kemudian didefinisikan secara rinci yang digunakan sebagai spesifikasi sistem.

#### **2. System and Software Design**

Pada tahap ini dilakukan untuk menentukan arsitektur sistem secara keseluruhan dengan melakukan perancangan desain sistem perangkat keras, dan perangkat lunak.

#### **3. Implementation and Unit Testing**

Pada tahap ini, design yang sudah dibuat kemudian di implementasikan ke dalam kode yang bisa dibaca oleh komputer. Pembuatan sistem ini menggunakan bahasa pemrograman PHP dan menggunakan database MySQL

#### 4. *Integration and System Testing*

Tahap ini adalah tahap integrasi dan pengujian sistem, agar sistem yang telah dibuat menjamin bahwa persyaratan sistem telah dipenuhi dan memberikan hasil yang diharapkan.

#### 5. *Operation and Maintenance*

Selanjutnya adalah tahap pemeliharaan, mungkin akan ada kesalahan yang tidak terdeteksi pada tahap terdahulu maka dilakukan pembaharuan atau perbaikan terhadap sistem tersebut.

### 2.3 Metode Pengujian Sistem


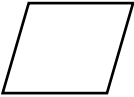

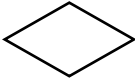
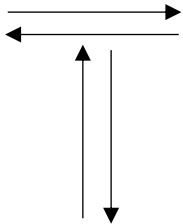
Pengujian sistem ini menggunakan metode *black box*, metode *black box* adalah pengujian yang dilakukan hanya dengan mengamati hasil eksekusi melalui data uji dalam memeriksa fungsi sistem [12]. Metode *black box* dapat menemukan kesalahan dalam kategori berikut:



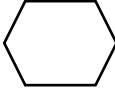
1. Fungsi yang tidak benar atau hilang.
2. Kesalahan interface
3. Kesalahan dalam struktur data atau akses basisdata eksternal
4. Inisialisasi dan kesalahan terminasi
5. Validitas fungsional
6. Kesensitifan sistem terhadap nilai input tertentu
7. Batasan dari suatu data

#### 2.3.1 Flowchart

*Flowchart* merupakan *Diagram* alir yang berbentuk simbol-simbol tertentu yang mudah dipahami dan digunakan untuk menjelaskan tahapan penyelesaian masalah. *Flowchart* bertujuan untuk menggambarkan suatu prosedur penyelesaian masalah secara tertata dan sederhana [13]. Berikut adalah simbol-simbol *Flowchart* beserta fungsinya.

**Tabel 2. 1** *Flowchart*

No	Simbol	Nama	Keterangan
1.		Terminator	Simbol yang menyatakan awal atau akhir suatu program
2.		Input/Output	Simbol yang menyatakan input (memasukan data) atau output (hasil dari suatu proses yang berjalan)
3.		Process	Simbol yang menyatakan sebuah proses yang dilakukan oleh komputer.
4.		Decission	Simbol yang menggambarkan pengambilan keputusan yang memberikan jawaban “ya” atau “tidak”
5.		Flow	Simbol ini disebut juga connecting line atau garis yang menyambungkan antar simbol

6.		Document	Simbol yang menyatakan input berasal dari dokumen atau output yang dicetak.
7.		Manual Operation	Simbol yang menunjukkan pengolahan data yang tidak dilakukan oleh komputer
8.		Preparation	Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal.

### 2.3.2 UML ( Unified Modelling Language)

UML (Unified Modelling Language) adalah salah satu standar bahasa yang dapat digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML adalah sebuah bahasa pemodelan yang berdasarkan *Diagram* atau grafik dengan sistem pengembangan perangkat lunak berbasis OOP (*Object Oriented Programming*) [14].


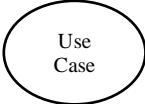

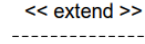
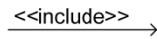

#### a. Usecase Diagram

Use Case *Diagram* adalah gambaran dari sistem yang akan dibuat melalui sebuah interaksi antara satu aktor atau lebih. Setiap aktor memiliki tugasnya masing-masing dan saling berkaitan satu



sama lain [15]. Simbol-simbol yang digunakan dalam *Usecase Diagram* tertera pada **Tabel 2.1** dibawah ini:



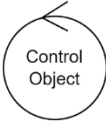

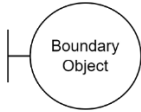
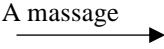

**Tabel 2. 2** Use Case Diagram

No	Simbol	Nama	Keterangan
1.	 Actor	Aktor	Mewakili peran orang yang berhubungan langsung dengan <i>Usecase</i> dan berinteraksi dengan sistem.
2.	 Use Case	Use Case	fungsi yang berfungsi sebagai tempat pertukaran pesan antara unit dengan actor yang biasanya dinyatakan dengan menggunakan kata kerja.
3.		Asosiasi/ Association	Sebagai penghubung antara aktor dan <i>Usecase</i> .
4.		Extend	Sebagai relasi use case tambahan yang bisa berdiri sendiri.
5.		Include	Sebagai relasi use case tambahan
6.		Generalization	Hubungan umum khusus antar dua buah <i>Usecase</i> , dimana arah panah mengarah pada usecase yang umum.

#### b. *Sequence Diagram*

*Sequence Diagram* merupakan *Diagram* yang menggambarkan interaksi antar objek dalam sistem (termasuk pengguna, *display/form*) berupa pesan yang digambarkan terhadap waktu. *Sequence Diagram* biasa digunakan untuk menggambarkan suatu langkah langkah untuk menghasilkan sebuah output [16]. Simbol-simbol yang terdapat dalam *Sequence Diagram* ditunjukkan pada **Tabel 2.2** dibawah ini:

**Tabel 2. 3** *Sequence Diagram*

No	Simbol	Nama	Keterangan
1.		<i>Object lifeline</i>	Mempresentasikan entitas tunggal dalam <i>Sequence Diagram</i> .
2.		Aktor	Menggambarkan orang/ proses yang terlibat dalam sistem
3.		<i>Control Object</i>	Digunakan untuk menghubungkan tabel dengan boundry
4.		<i>Entity Object</i>	Menggambarkan hubungan yang akan dilakukan antar sistem
5.		<i>Boundary Object</i>	Digunakan untuk menggambarkan sebuah form
6.		<i>Message</i>	Menggambarkan pengiriman pesan
7.		Waktu aktif	Menyatakan sebuah objek dalam keadaan aktif dan melakukan interaksi

### 2.3.3 Pengertian Basis Data

Basis data adalah sekumpulan data yang terintegrasi dan terorganisir yang umumnya diakses dan disimpan dari suatu komputer

secara elektronik. Pada saat pangkalan data semakin kompleks, maka pangkalan data akan dikembangkan menjadi teknik perancangan dan pemodelan secara formal [17].

Basis data diolah dengan menggunakan perangkat lunak yang biasa disebut *database* manajemen sistem (DBMS). DBMS adalah perangkat lunak yang dirancang untuk melakukan pengaturan dan pengelolaan data secara lebih mudah. Salah satu jenis DBMS yang berhubungan antar tabel adalah *Relationship database* manajemen sistem (RDBMS), contohnya seperti *Oracle*, *MySQL*, *Ms SQL Server* dan *Ms Access* [17].

MySQL adalah *database* server yang bebas di akses dengan lisensi GNU *General Public License* (GPL) yang dapat dipakai untuk keperluan komersial maupun keperluan pribadi. MySQL merupakan *server database* yang mendukung bahasa *database* SQL sebagai bahasa yang interaktif dalam mengelola data [17].

Untuk dapat mengakses dan mengolah data base, pengguna harus menggunakan DDL dan DML, adapun pengertian dari keduanya adalah sebagai berikut:

#### 1. DDL (*Data Definition Language*)

DDL digunakan untuk mengubah, mendefinisikan serta menghapus data objek objek yang ada dalam data base. Perintah dari DDL seperti: *DROP*, *CREATE*, *ALTER*.


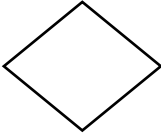
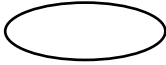

#### 2. DML (*Data Manipulation Language*)

DML digunakan untuk memasukan, memperbaharui ataupun menghapus data dalam suatu tabel, perintah dari DML yaitu:

### 2.3.4 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* adalah model data yang dikembangkan berdasarkan objek untuk menjelaskan secara logis hubungan antar basis data kepada pengguna. *ERD* dapat membantu perancang dalam menganalisis dan perancangan basis data karena dapat memberikan gambaran data yang dibutuhkan dan keselarasan antar data [18]. Adapun simbol-simbol yang digunakan untuk membuat *ERD* ditunjukkan pada tabel 2.3 dibawah ini:

**Tabel 2. 4 ERD**

No	Simbol	Nama	Keterangan
1.		Entitas	Simbol yang menyatakan kumpulan objek yang dapat diidentifikasi secara unik.
2.		Relasi	Simbol yang menggambarkan hubungan yang terjadi antara salah satu atau lebih entitas.
3,		Atribut	Simbol yang menyatakan penjelasan detail dari entitas.
4.		Link	Penghubung antara himpunan relasi dengan himpunan entitas dengan atribut lainnya.

LEMBAR INI SENGAJA DIKOSONGKAN



# **BAB III**

# **METODOLOGI DAN**

# **PERANCANGAN SISTEM**

## **BAB III**

### **METODOLOGI DAN PERANCANGAN SISTEM**

#### **3.1 Metodologi**

Proses pembuatan sistem yang dilakukan oleh penulis menggunakan metode *waterfall* karena metode ini memfokuskan mempermudah interaksi antara member dan pengembang yang dapat berfungsi mengatasi ketidak selarasan antara pengembang dan pengguna sistem. Berikut adalah alat, bahan dan jalan penelitian dalam pembuatan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias.

##### **3.1.1 Bahan Penelitian**

Bahan penelitian yang digunakan dalam pembangunan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias dibagi menjadi dua data yaitu data primer dan data sekunder. Data data tersebut meliputi:

##### **1. Data Primer**

Data primer merupakan data yang diperoleh secara langsung dari narasumber. Data primer yang dibutuhkan dalam pembangunan sistem ini yaitu data produk/jasa dan data alur pemesanan

##### **2. Data Sekunder**

Data sekunder merupakan data yang diperoleh secara tidak langsung. Data sekunder yang dibutuhkan dalam pembangunan sistem ini meliputi jurnal penelitian, kajian pustaka berupa buku, serta artikel internet.

### 3.1.2 Alat Penelitian

Dalam pembuatan sistem ini, penulis menggunakan alat penelitian yang berupa beberapa aplikasi dan perangkat keras komputer. Adapun alat pembuatan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias ditunjukkan pada **Tabel 3.1**

**Tabel 3. 1** Alat Penelitian

No	Kebutuhan Perangkat Lunak	Keterangan
1.	<i>Windows 10 64-bit</i>	Sistem operasi pada komputer yang digunakan sebagai sarana pengembangan sistem
2.	<i>Sublim text 3</i>	Perangkat lunak yang berupa text Editor yang digunakan untuk menerapkan Coding untuk membuat Sistem Informasi Pemesanan Jasa <i>Wedding Organizer</i> Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias.
3.	<i>Microsoft Office Word 2016</i>	Perangkat lunak yang digunakan untuk menyusun laporan secara rinci mengenai tugas akhir yang dibuat.



4.	<i>Google Chrome</i>	Browser yang digunakan untuk menjalankan sistem yang dibuat.
5.	<i>Xampp</i>	Perangkat lunak yang berfungsi sebagai server lokal yang dapat menyimpan data website dan dapat diakses secara offline.
6.	<i>Draw io</i>	Pembuatan <i>Flowchart, Usecase Diagram</i> dan <i>Sequence Diagram</i> .
7.	<i>Balsamiq Mockups 3</i>	Software yang digunakan untuk membuat perancangan antarmuka sistem

### 3.1.3 Analisa Kebutuhan Pengguna

Bagian ini berisi secara rinci kebutuhan pengguna dari masing masing aktor au pengguna yang terlibat dalam sistem. Kebutuhan pengguna dapat dilihat pada **Tabel 3.2**

**Tabel 3. 2** Analisa Kebutuhan Pengguna

No	User	Hak Akses
1.	Admin	<i>Login</i> Mengelola data kategori Mengelola data member Mengelola data vendor Mengelola data tim Melihat data pesanan
2.	Member	Registrasi <i>Login</i> Mengelola profil member Mengelola pemesanan produk
3.	Vendor	Registrasi

		<i>Login</i> Mengelola profil vendor Mengelola data produk Mengelola data pesanan produk setiap vendor Lihat kategori
4.	Keuangan	<i>Login</i> Mengelola data pesanan
5.	Pemilik	<i>Login</i> Lihat data member Cetak data member Lihat data vendor Cetak data vendor Lihat data tim Cetak data tim Lihat data kategori Liha data produk Cetak data produk Lihat laporan keuangan Cetak laporan keuangan.

### 3.1.4 Analisa Kebutuhan Eksternal

#### A. Analisa Kebutuhan Antarmuka Perangkat Lunak

Analisa kebutuhan antarmuka perangkat lunak digunakan untuk merancang, mengembangkan, menjalankan serta untuk mendokumentasi pembangunan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan

Notifikasi Whatsapp Di Indah Permata Rias. Analisa kebutuhan perangkat lunak yang digunakan dapat dilihat pada **Tabel 3.3** dan

**Tabel 3. 3** Analisa kebutuhan Perangkat Lunak

No	Kebutuhan Perangkat Lunak	Keterangan
1.	<i>Windows 10</i>	Sistem operasi komputer yang digunakan untuk pengembangan sistem
2.	<i>MySQL</i>	Perangkat lunak yang digunakan untuk pengelolaan <i>database</i>
3	<i>Google Chrome</i>	Browser yang digunakan untuk menjalankan sistem yang dibuat
4.	<i>Sublim Text 3</i>	Perangkat lunak yang digunakan untuk implementasi coding dalam pembuatan sistem

#### **B. Analisa Kebutuhan Perangkat Keras**

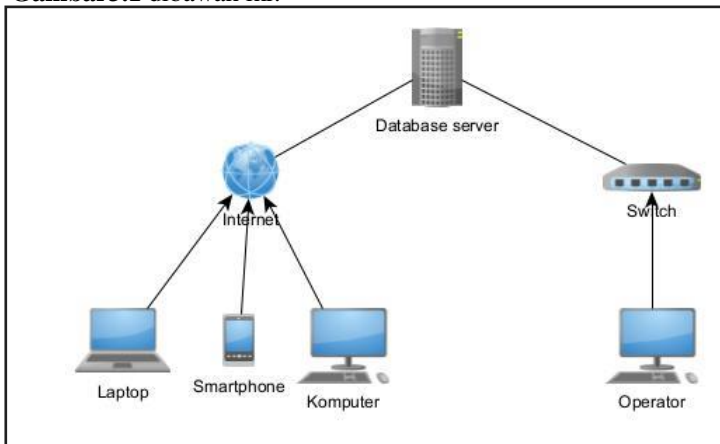
Kebutuhan perangkat keras dalam pembangunan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias adalah perangkat keras komputer dengan spesifikasi yang cukup atau lebih baik. Kebutuhan perangkat keras dapat dilihat pada **Table 3.4** dibawah ini:

**Tabel 3. 4** Analisa kebutuhan perangkat keras

No	Kebutuhan Perangkat Keras
1.	<i>Processor intel i5 2.5 GHz</i>
2.	RAM 4 GB
3.	<i>Harddisk : 500 GB</i>
4.	VGA NVIDIA GeForce 920MX

#### **C. Analisis Kebutuhan Antarmuka Jaringan**

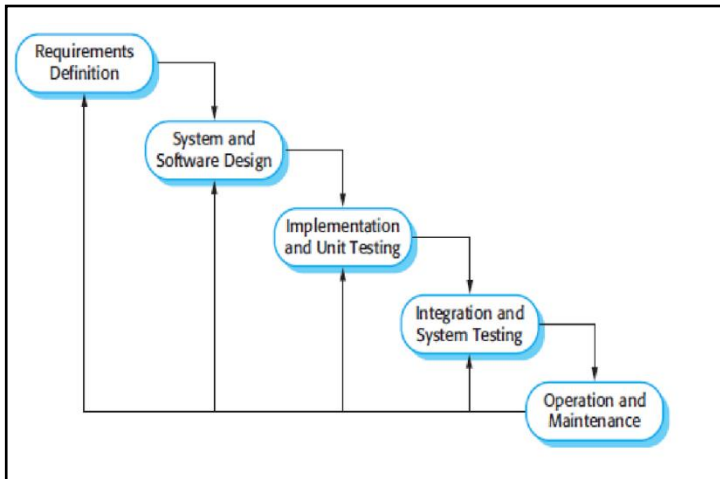
Analisis kebutuhan antarmuka jaringan digunakan untuk menganalisa jaringan yang digunakan untuk menjalankan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias. Untuk mengakses data pada server dan dalam transmisi data menggunakan media internet. Pengguna dapat mengakses sistem dengan menggunakan perangkat smarhphone , laptop dan komputer. Analisa kebutuhan antarmuka jaringan dapat dilihat pada **Gambar3.1** dibawah ini:



**Gambar 3. 1** Kebutuhan Antarmuka Jaringan

### 3.1.5 Jalan Penelitian

Jalan penelitian adalah tahapan yang digunakan dalam pengembangan Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Notifikasi Menggunakan Whatsapp Di Indah Permata Rias dengan menggunakan metode *waterfall*. Tahapan dari metode *waterfall* dapat dilihat pada **Gambar 3.2**



**Gambar 3. 2** Metode *Waterfall*

### 1. *Requirement Definition*

Tahap ini merupakan tahap analisa kebutuhan sistem yang akan dikembangkan. Pada tahap ini penulis mengumpulkan informasi tentang sistem yang sedang berjalan dengan melakukan observasi dan wawancara kepada pemilik Indah Permata Rias mengenai proses bisnis yang sedang berjalan. Tahapan ini mencakup analisis kebutuhan pengguna, analisis kebutuhan perangkat lunak, kebutuhan perangkat keras

### 2. *System and Software Design*

Pada tahap ini dilakukan kegiatan pembuatan desain antarmuka pengguna dan perancangan database. Proses Perancangan dimulai dengan pembuatan *Flowchart*, *usecase Diagram*, *Sequence Diagram*, dan *entity relationship Diagram* yang bertujuan untuk membantu memberikan gambaran dan proses dalam pembangunan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Notifikasi Menggunakan WhatsApp di Indah Permata Rias

### 3. *Implementation and Unit Testing*

Pada tahap ini, design yang sudah dibuat kemudian di implementasikan ke dalam kode yang bisa dibaca oleh komputer. Pembuatan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Notifikasi Menggunakan WhatsApp di Indah Permata Rias ini menggunakan bahasa pemrograman PHP dan menggunakan *database MySQL*

#### 4. *Integration and Unit Testing*

Tahap ini adalah tahap pengujian Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Notifikasi Menggunakan WhatsApp di Indah Permata Rias yang berfokus pada bagian internal perangkat lunak secara logis yang digunakan untuk memastikan setiap pernyataan telah diuji dan pada bagian fungsi eksternal dilakukan pengujian untuk menemukan kesalahan (*error*) dan memastikan *input* akan memberikan hasil yang diharapkan. Tahapan pengujian pada sistem ini menggunakan metode pengujian *black-box testing*.

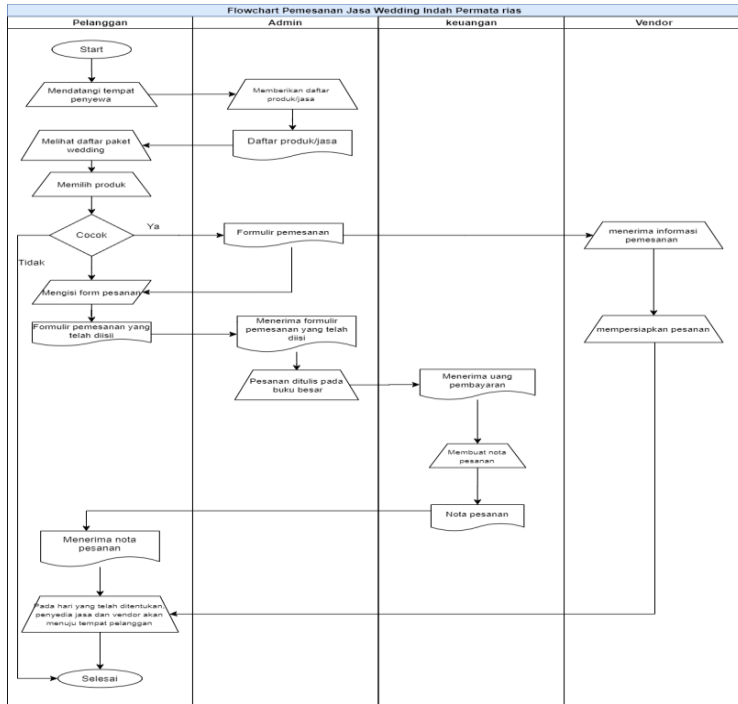
#### 5. *Operation and Maintenance*

Selanjutnya adalah tahap penggunaan sistem, pengujian dan pemeliharaan sistem yang telah dibuat. Tahap ini dilakukan oleh *user* untuk mengoperasikan sistem apabila sistem dinilai perlu ditambahkan fitur baru dan memiliki permasalahan maka peneliti akan melakukan pemeliharaan atau perbaikan sistem.

### **3.2 Perancangan Sistem**

#### **3.2.1 Analisis Sistem yang sedang berjalan**

Proses pemesanan jasa *Wedding Organizer* pada Indah Permata Rias yang sedang berjalan digambarkan pada *Diagram* alir dibawah ini yang ditunjukkan pada **Gambar 3.3**



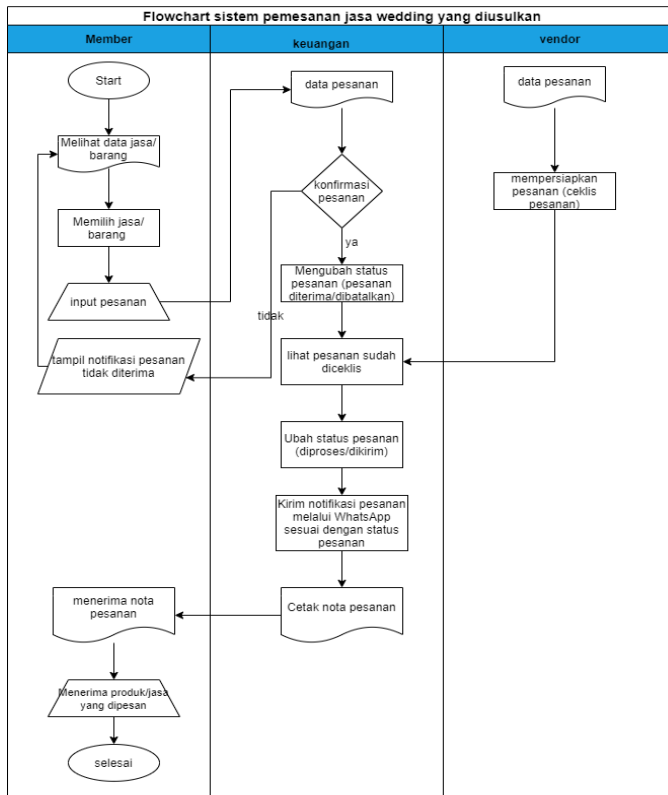
**Gambar 3. 3** Analisa sitem yang sedang berjalan

Proses pemesanan yang sedang di Indah Permata Rias dimulai dari pelanggan mendatangi tempat penyewa untuk menanyakan mengenai produk apa saja yang ditawarkan, kemudian admin memberikan informasi produk kepada pelanggan. Pelanggan memilih produk/jasa yang tersedia, jika pelanggan merasa cocok selanjutnya pelanggan akan mengisi formulir pemesanan, lalu admin akan mencatat pemesanan pada buku besar, dan memberikan informasi kepada vendor mengenai produk yang dipesan. Setelah menerima informasi vendor akan melakukan persiapan pesanan. Setelah itu pelanggan harus melakukan pembayaran ke bagian keuangan ,setelah bagian keuangan menerima uang, maka pelanggan akan diberi nota pesanan. Kemudian penyedia jasa dan tim vendor

akan menuju tempat pelanggan yang telah memesan pada hari yang telah ditentukan.

### 3.2.2 Rancangan sistem yang akan dibangun

A. Gambaran alur sistem pemesanan produk/jasa yang diusulkan ditunjukkan pada **Gambar 3.4**



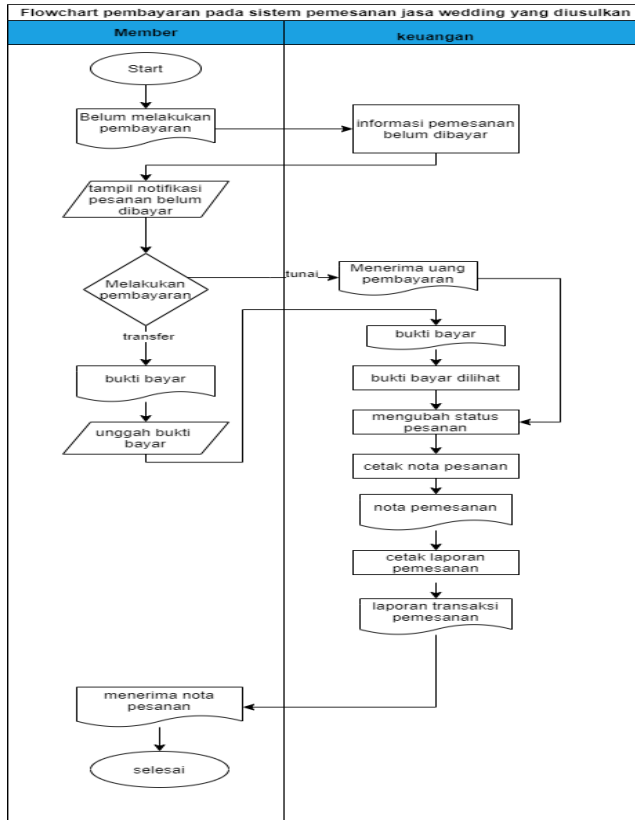
**Gambar 3.4** Flowchart sistem pemesanan jasa *Wedding Organizer* yang akan diusulkan



Alur yang digunakan dalam sistem pemesanan yang diajukan adalah:

1. Member melihat data produk dan jasa yang ditawarkan
2. Member memilih produk/jasa yang ditawarkan
3. Member menginputkan pesanan
4. Data pesanan akan di lihat oleh keuangan, dan akan dikonfirmasi, jika pesanan tidak di terima maka keuangan akan mengubah status pesanan diterima. jika tidak diterima keuangan akan mengubah status pesanan dibatalkan.
5. Pesanan yang masuk akan masuk pada menu pesanan tiap vendor.
6. Venndor melakukan ceklis produk jika produk sudah disiapkan
7. Jika produk sudah siap, maka keuangan akan mengubah status pesanan menjadi diproses/ dikirm.
8. Notifikasi pesanan akan dikirim oleh keuangan sesuai dengan status pesanan melalui WhatsApp
9. Keuangan mencetak nota pesanan
10. Member menerima nota pesanan dan menerima produk/jasa yang telah dipesan
11. Selesai

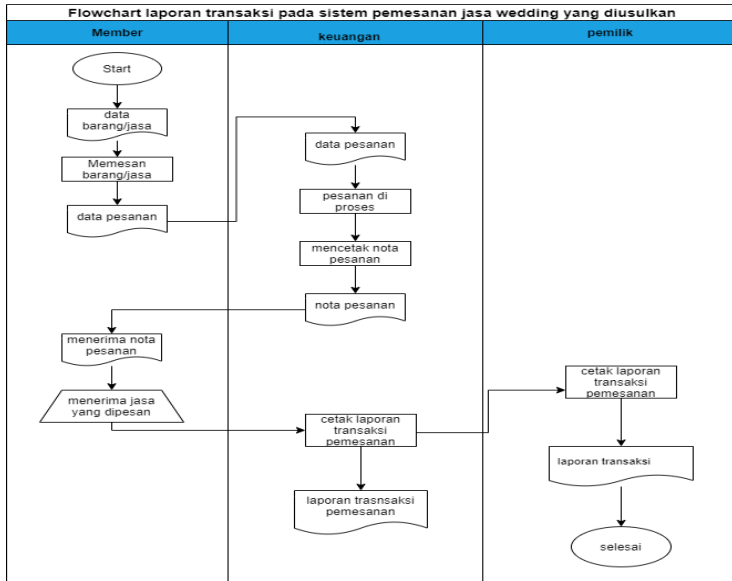
- B. Gambaran alur pembayaran pada sistem jasa *Wedding Organizer* yang akan diusulkan ditunjukkan pada **Gambar 3.5** dibawah ini



**Gambar 3.5** Sequence pembayaran sistem yang diusulkan

Alur pembayaran dalam sistem pemesanan jasa *Wedding Organizer* yang diajukan adalah:

1. tampil data member yang belum melakukan pembayaran
  2. keuangan akan mengirimkan notifikasi kepada member yang belum melakukan pembayaran.
  3. Member memilih akan melakukan pembayaran menggunakan transfer atau tunai, jika secara tunai maka member akan menemui bagian keuangan untuk membayar.
  4. Member yang memilih pembayaran secara transfer harus mengunggah bukti bayar.
  5. Kemudian bukti bayar akan di lihat terlebih dahulu oleh keuangan
  6. Jika pembayaran telah dilakukan, maka keuangan akan mengubah status pemesanan
  7. Bagian keuangan dapat mencetak nota pesanan dan mencetak laporan pemesanan
  8. Pada hari yang telah ditentukan, member akan menerima nota pesanan
  9. Selesai.
- C. Gambaran alur laporan transaksi pada sistem sistem jasa *Wedding Organizer* yang akan diusulkan ditunjukkan pada **Gambar 3.6** dibawah ini



**Gambar 3. 6** Flowchart laporan transaksi yang diusulkan

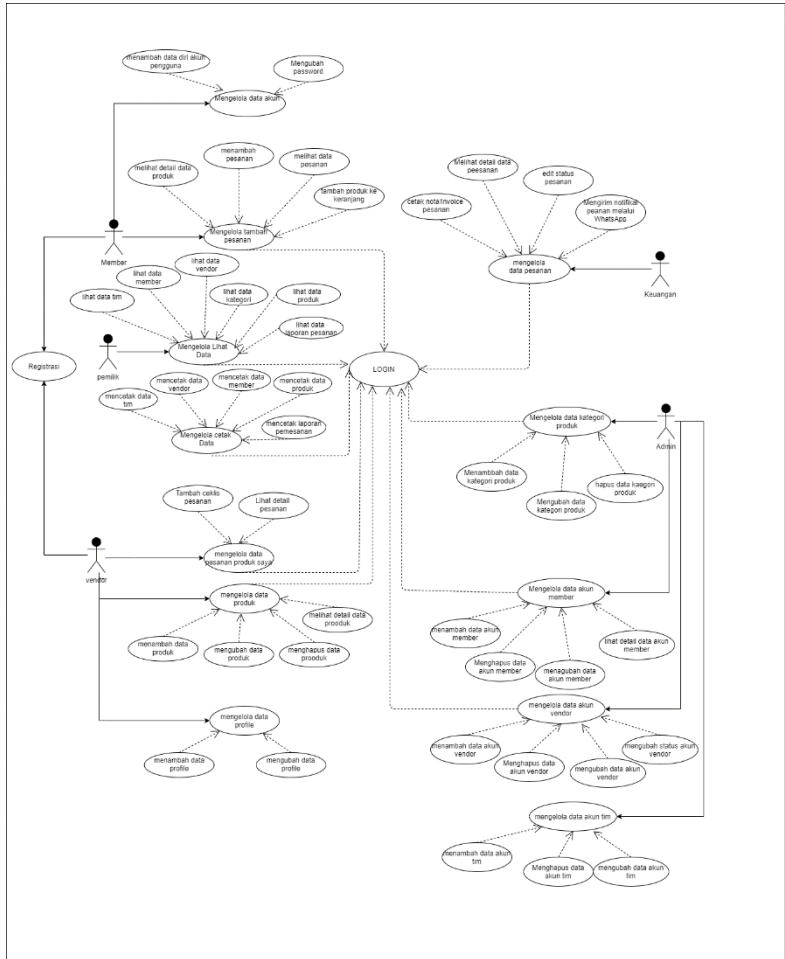
Alur laporan transaksi dalam sistem yang diusulkan adalah:

1. Member memesan produk/ jasa
2. Member melakukan pembayaran
3. keuangan memproses pesanan
4. keuangan mencetak nota pesanan
5. member menerima nota pesanan dan menerima pesanan
6. keuangan dan pemilik dapat melakukan cetak laporan transaksi pemesanan, sesuai data yang diinginkan.
7. selesai

### 3.2.3 Aliran Informasi

#### A. Usecase Diagram

*Usecase Diagram* mengenai Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias ditunjukkan pada **Gambar 3.5**



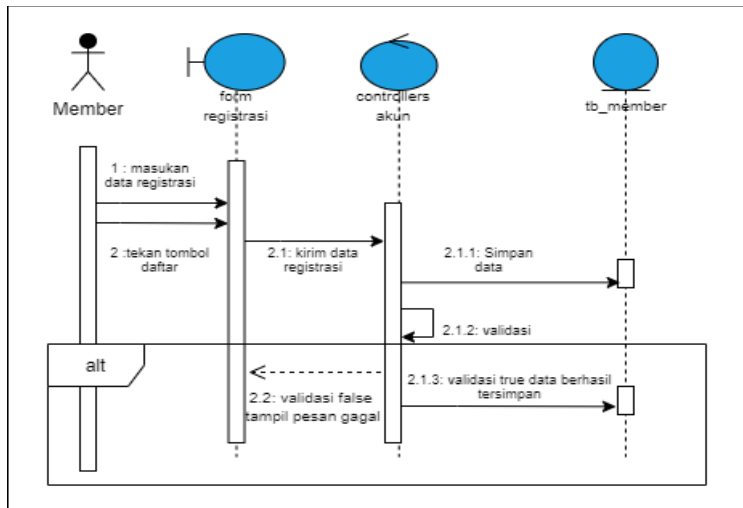
**Gambar 3. 7** *Usecase* perancangan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias

Pada **Gambar 3.7** menjelaskan bahwa Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias mempunyai 5 aktor yaitu member, admin, vendor, pemilik dan keuangan. Member dapat melakukan *Login*, registrasi, mengelola profil pengguna yang terdiri dari menambah data diri pengguna dan mengubah data diri pengguna, member juga dapat melakukan olah data tambah pesanan yang berupa melihat data produk/jasa, menambah pesanan, melihat data pesanan dan menambah pesanan ke dalam keranjang, selanjutnya terdapat aktor pemilik yang dapat melakukan *Login*, mengelola lihat data yang berupa lihat data tim, lihat data member, lihat data vendor, lihat data kategori, lihat ata produk, dan lihat data laporan transaksi pemesanan. Pemilik juga dapat mengelola cetak data yang meliputi cetak data tim, cetak data member, cetak data vendor dan cetak laporan transaksi pemesanan. Kemudian ada aktor vendor yang dapat melakukan *Login* dan registrasi dapat mengelola pesanan produk saya yang berupa tambah ceklis dan lihat detail pesanan, vendor juga dapat mengelola data produk yang berupa menambah data produk, mengubah data produk, menghapus data produk, serta melihat detail data produk. Aktor vendor juga dapat melakukan olah data profile yang terdiri dari tambah data profile dan ubah data profile. Selanjutnya terdapat aktor keuangan yang dapat melakukan *Login* dan dapat mengelola data pemesanan yang beruppa lihat detail data pesanan, ubah status pesanan, mengirim notifikasi pesanan melalui WhasApp dan mencetak nota pesanan. Lalu yang terakhir terdapat aktor Admin yang dapat melakukan *Login*, mengelola data kategori produk yang berupa menambah data kategori produk, mengubah kategori produk, dan menghapus kategori produk. Admin dapat mengelola data member yang berupa tambah data member, ubah data member, hapus data member, dan lihat detail data member. Selain itu, admin dapa mengelola data vendor yang meliputi tambah data vendor, hapus data vendor, ubah data vendor, dan mengubah status vendor. Data tim juga dikelola oleh admin yang emeliputi tambah akun tim, hapus akun tim, dan ubah akun tim.

## **B. *Sequence Diagram***

## 1. Sequence Diagram registrasi Member

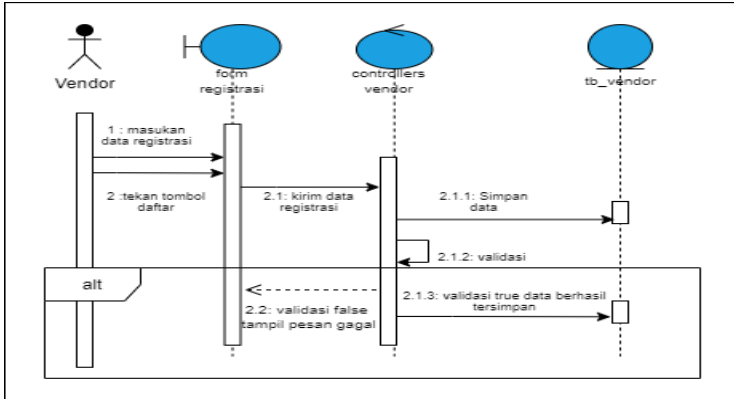
Rancangan proses registrasi member dimulai dari member memasukan data nama, *email*, dan *password* kedalam form registrasi, kemudian data yang telah dimasukan akan dikirim ke control data, kemudian *controllers* akun akan menyimpan data tersebut kedalam *tb\_member*. Jika data berhasil tersimpan maka member akan diarahkan ke halaman beranda. Tetapi jika validasi salah maka akan kembali ke form registrasi. Rancangan proses registrasi member di tunjukan pada **Gambar 3.8**



**Gambar 3.8** Sequence Diagram Registrasi Member

## 2. Sequence Registrasi Vendor

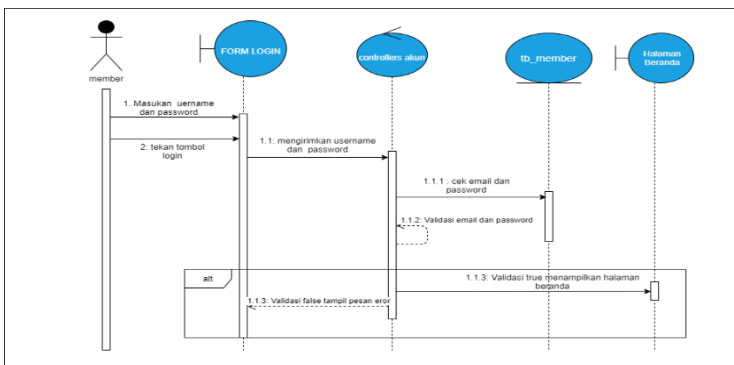
Rancangan proses registrasi vendor dimulai dari vendor memasukan data vendor, *email*, dan *password* kedalam form registrasi, kemudian data yang telah dimasukan akan dikirim ke *control data*, kemudian *controllers welcome* akan menyimpan data tersebut kedalam *tb\_vendor*. Jika data berhasil tersimpan maka vendor akan diarahkan ke halaman beranda. Tetapi jika validasi salah maka akan kembali ke form registrasi. Rancangan proses registrasi vendor di tunjukan pada **Gambar 3.9**



Gambar 3. 9 Sequence Diagram Registrasi Vendor

### 3. Sequence Diagram Login Member

Rancangan proses *login* yang digunakan oleh member dimulai dengan member memasukkan *username* dan *password* ke dalam form *login*. Lalu, data yang telah dimasukan akan di kirim ke *controllers* akun, lalu data akan dicocokkan dengan *tb\_member* , jika data tersebut belum terdaftar maka akan muncul psan gagal /*error*. Apabila *username* dan *password* sudah terdaftar maka *login* controller akan melakukan proses *login* ke halaman beranda . Rancangan proses *login* member digambarkan pada **Gambar 3.10**

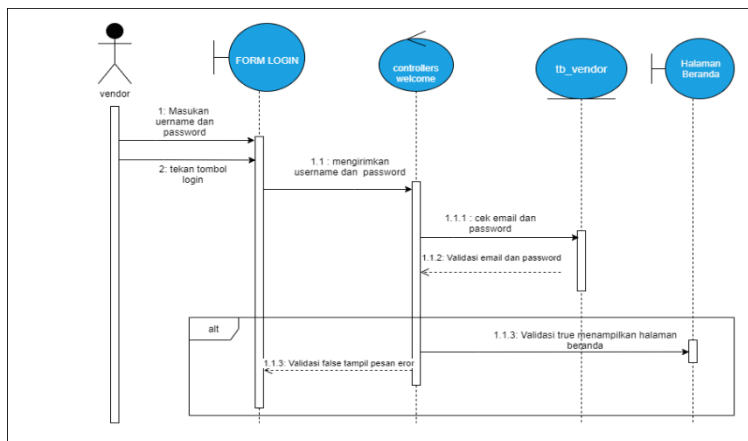


Gambar 3. 10 Sequence Login Member



#### 4. *Sequence Diagram Login Vendor*

Rancangan proses *login* yang digunakan oleh vendor dimulai dengan *user* memasukkan *username* dan *password* ke dalam form *login*. Lalu, data yang telah dimasukan akan di kirim ke *controllers welcome*, lalu data akan dicocokkan dengan *tb\_vendor*, jika data tersebut belum terdaftar maka akan muncul psan gagal */error*. Apabila *username* dan *password* sudah terdaftar maka *login* controller akan melakukan proses *login* ke halaman beranda. Rancangan proses *login* vendor digambarkan pada **Gambar 3.11**

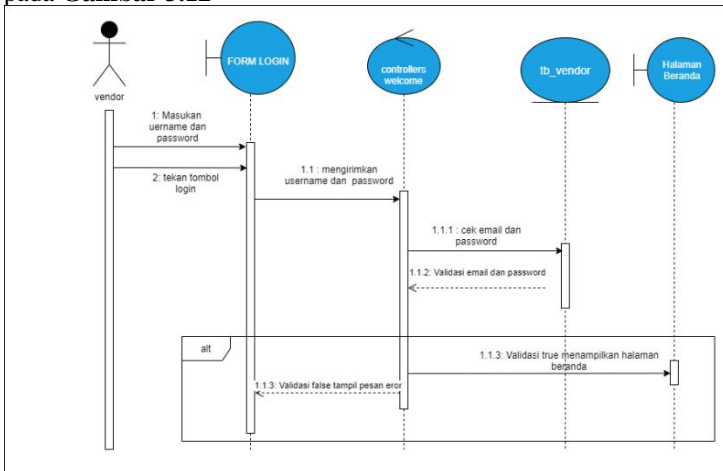


**Gambar 3. 11** *Sequence Diagram Login Vendor*

#### 5. *Sequence Diagram Login Tim (admin,keuangan, dan pemilik)*

Rancangan proses *login* yang digunakan oleh admin,keuangan dan pemilik dimulai dengan *user* memasukkan *username* dan *password* ke dalam form *login*. Lalu, data yang telah dimasukan akan di kirim ke *controllers welcome*, lalu data akan dicocokkan dengan *tb\_admin*, jika data tersebut belum terdaftar maka akan muncul pesan gagal */error*. Apabila *username* dan *password* sudah terdaftar maka *login* controller akan melakukan proses *login* ke halaman beranda

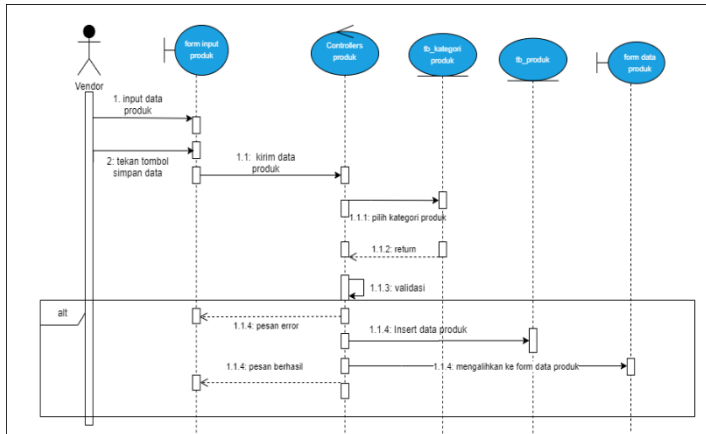
masing-masing aktor. Rancangan proses *login* vendor digambarkan pada **Gambar 3.12**



**Gambar 3. 12** *Sequence Diagram Login Admin,Keuangan, dan Pemilik*

## 6. *Sequence Diagram Vendor Menambah Data Produk*

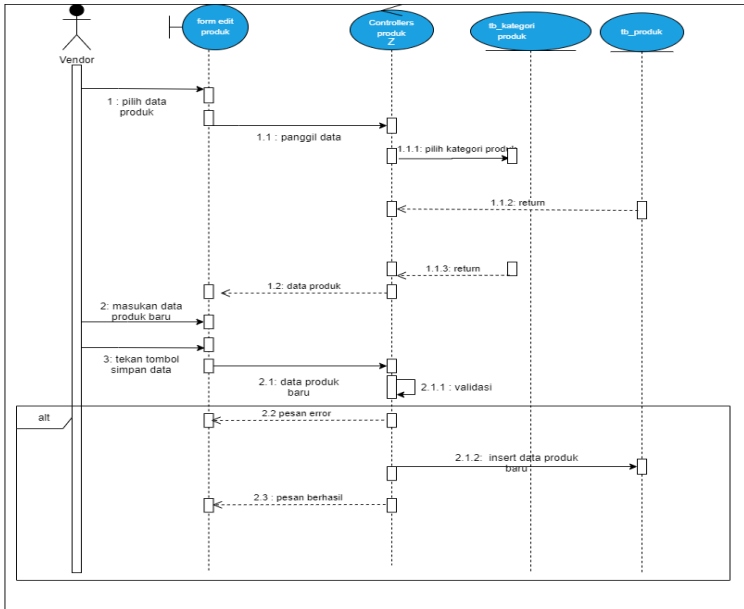
Rancangan proses menambah data produk yang dilakukan oleh vendor diawali dengan vendor menginputkka data produk pada halaman produk dan menekan tombol simpan data, kemudian UI produk akan memanggil data kategori dari tabel kategori untuk ditampilkan pada form input produk. Vendor akan memilih kategori produk yang diinput, dan memasukan gambar produk. *Controllers* produk akan mengirimkan data yang dikirim untuk dilakukan validasi, jika *true* maka data produk akan disimpan ke dalam *tb produk* dan akan muncul pesan data berhasil diinput, tetapi jika *false* maka data produk tidak tersimpan dan akan dikembalikan pada halaman *input* data produk dan akan menampilkan pesan data gagal disimpan. *Sequence Diagram* menambah data produk yang dilakukan oleh vendor ditunjukkan pada **Gambar 3.13**



**Gambar 3. 13** Sequence Diagram tambah produk

## 7. Sequence Diagram Vendor Mengubah Data Produk

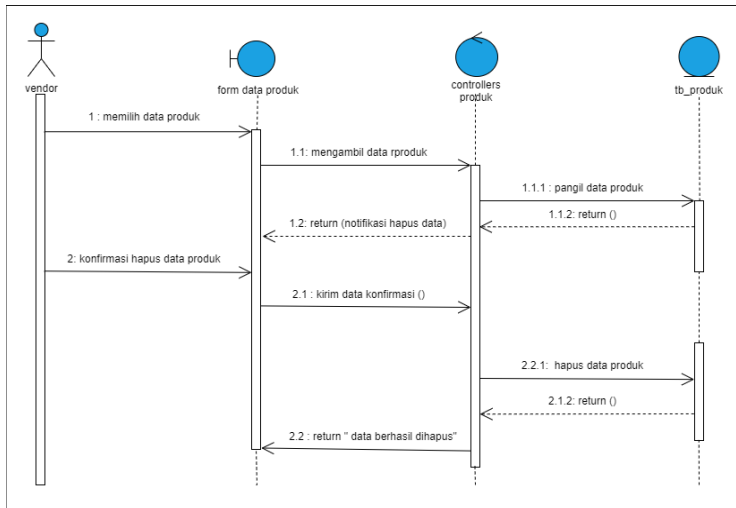
Rancangan proses mengubah data produk yang dilakukan oleh vendor diawali dengan vendor memilih data produk yang akan diubah, lalu data akan disampaikan ke *controllers* produk dan memanggil kategori produk pada *database* *tb\_kategori*, dan memanggil data produk untuk ditampilkan pada form *edit* produk. Kemudian, data yang di panggil akan muncul dan vendor dapat melakukan pengubahan data, dengan memasukkan data produk yang baru. Data produk baru akan divalidasi oleh *controllers* jika *true* maka data produk yang baru akan disimpan ke dalam *tb* produk tetapi jika *false* maka data poduk tidak tersimpan dan akan menampilkan pesan gagal. Sequence Diagram mengubah data produk yang dilakukan oleh vendor ditunjukan pada **Gambar 3.14**



**Gambar 3. 14** *Sequence Diagram* ubah data produk

## 8. *Sequence Diagram* Hapus Data Produk Pada Vendor

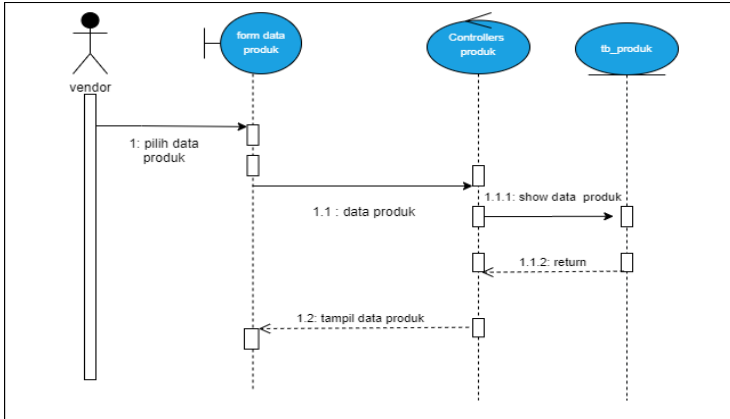
Rancangan proses menghapus data produk yang dilakukan oleh vendor diawali dengan vendor memilih data produk yang akan dihapus, lalu data akan dikirimkan ke *controller*, kemudian *controller* mengirimkan konfirmasi hapus data *by 'id'*. Jika vendor mengkonfirmasi penghapusan data akan di validasi oleh *controller* produk Setelah itu, data produk yang sudah dipilih akan dihapus dan akan dikembalikan pada halaman data produk dan akan menampilkan pesan data berhasil dihapus. *Sequence Diagram* menghapus data produk yang dilakukan oleh vendor ditunjukkan pada **Gambar 3.15**



**Gambar 3. 15** *Sequence Diagram* Menghapus Data Produk

## 9. *Sequence Diagram* Melihat Data Produk

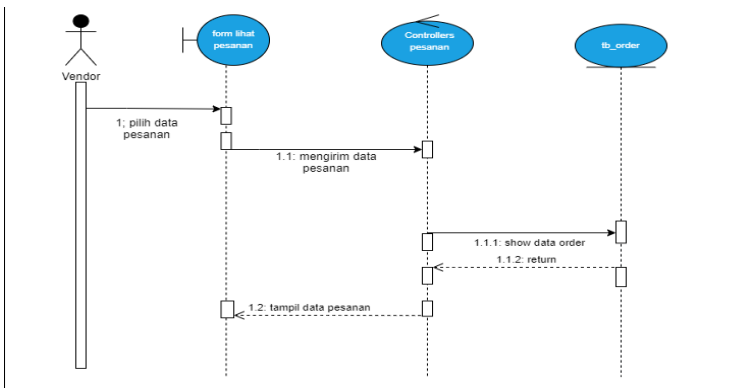
Rancangan proses melihat data produk yang dilakukan oleh vendor diawali dengan vendor memilih data produk yang akan dilihat, lalu *controller* akan memanggil data produk pada tabel produk. Lalu data akan ditampilkan pada form data produk. *Sequence Diagram* melihat data produk yang dilakukan oleh vendor ditunjukkan pada **Gambar 3.16**



**Gambar 3. 16** *Sequence Diagram* Melihat Data Produk

## 10. *Sequence Diagram* Vendor Melihat Data Pemesanan

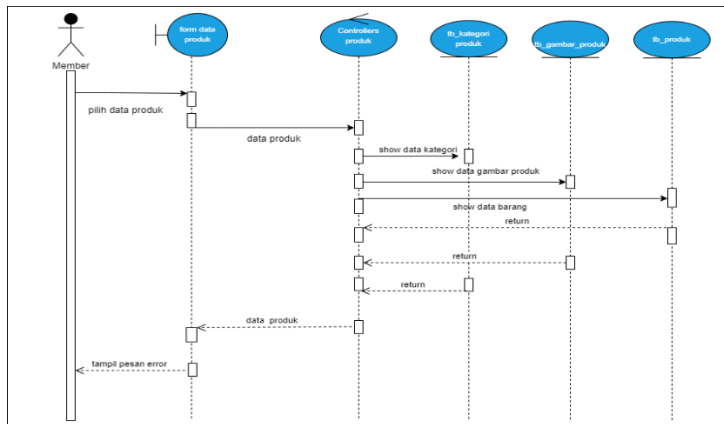
Rancangan proses untuk vendor melihat data data pemesanan adalah dimulai dari vendor melihat form pesanan kemudian akan diolah pada *controllers* pesanan, kemudian *controllers* akan memanggil data *invoice* pada *tb\_Invoice*, dan memanggil data produk pada *tb\_produk*, kemudian akan dilakukan validasi jika berhasil maka data order akan muncul pada halaman pesanan, tetapi jika gagal maka akan tampil pesan gagal/*error*. *Sequence Diagram* melihat riwayat pesanan ditunjukkan pada **Gambar 3.17**



**Gambar 3. 17** Sequence *Diagram* Vendor Melihat Data Pemesanan

### 11. Sequence *Diagram* Member Melihat Data Produk

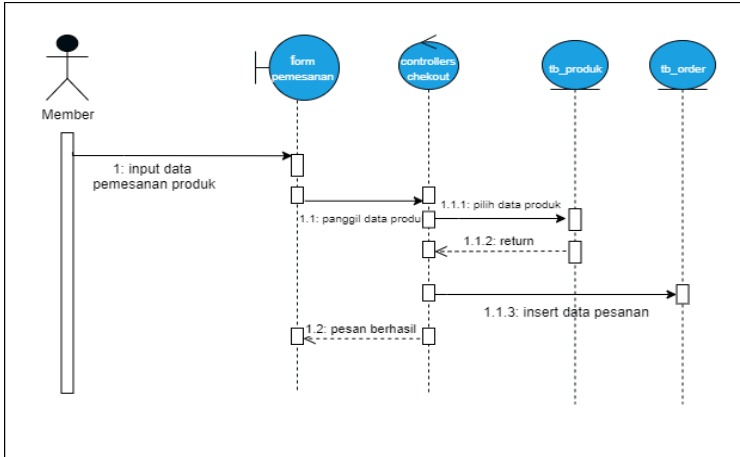
Rancangan proses melihat data produk yang dilakukan oleh member diawali dengan vendor memilih data produk yang akan dilihat, lalu *controller* akan memanggil data kategori dari *tb\_kategori* dan data gambar produk pada *tb\_gambar\_produk*. Lalu data akan ditampilkan pada form data produk. *Sequence Diagram* melihat data produk yang dilakukan oleh member ditunjukkan pada **Gambar 3.18**



**Gambar 3. 18** Sequence *Diagram* Member Melihat Data Produk

### 12. Sequence *Diagram* Member Menambah Pesanan

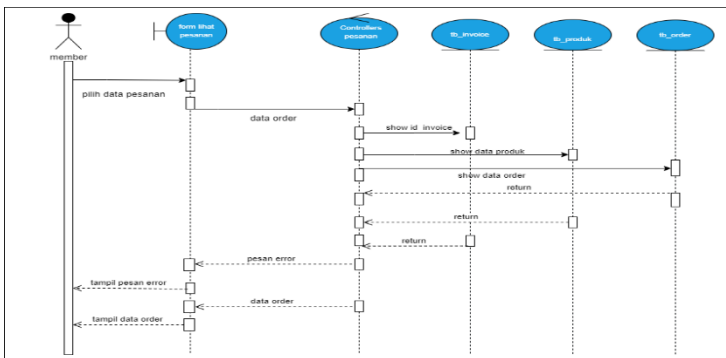
Rancangan proses menambah pesanan produk yang dilakukan oleh member diawali dengan member menginput data pesanan pada form pemesanan, lalu data produk akan pada tabel produk akan dipanggil dan ditampilkan pada halaman input pemesanan. Member dapat memilih produk yang akan dipesan lalu data pesanan tersebut akan dikirimkan ke *controllers checkout* kemudian oleh *controllers checkout* akan diproses, apabila true maka data akan disimpan pada *tb\_order*. *Sequence Diagram* memesan produk yang dilakukan oleh member ditunjukkan pada **Gambar 3.19**



**Gambar 3. 19** Sequence Diagram Member Menambah Pesanan

### 13. Sequence Diagram Member Melihat Data Pesanan

Rancangan proses untuk member melihat data data pemesanan adalah dimulai dari member melihat form pesanan kemudian akan diolah pada *controllers* order, kemudian *controllers* akan memanggil data *Invoice* pada tabel *Invoice*, dan memanggil data produk pada tabel produk, kemudian akan dilakukan validasi jika berhasil maka data order akan muncul pada halaman pesanan, tetapi jika gagal maka akan tampil pesan gagal. *Sequence Diagram* melihat riwayat pesanan ditunjukkan pada **Gambar 3.20**

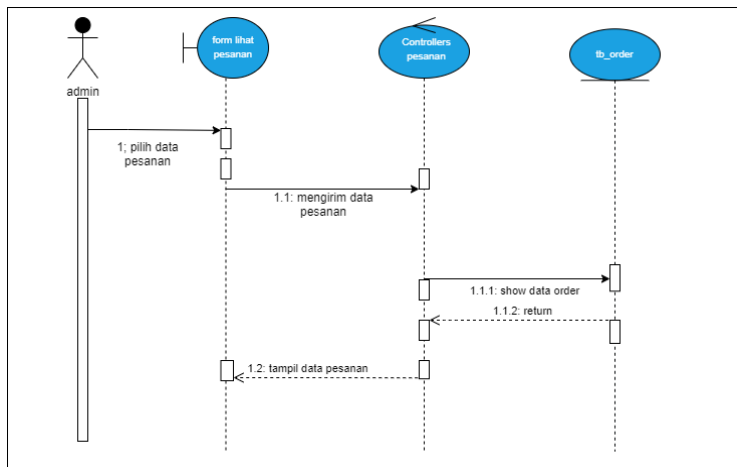




**Gambar 3. 20** *Sequence Diagram* Member Melihat Data Pesanan

#### 14. *Sequence Diagram* Admin Melihat data pesanan

Rancangan proses untuk admin melihat data data pemesanan adalah dimulai dari member melihat form pesanan kemudian akan diolah pada *controllers* pesanan, kemudian *controllers* akan memanggil data *invoice* pada tabel *invoice*, dan memanggil data produk pada tabel produk, kemudian akan dilakukan validasi jika berhasil maka data order akan muncul pada halaman pesanan, tetapi jika gagal maka akan tampil pesan gagal. *Sequence Diagram* melihat data pesanan yang dilakukan oleh admin ditunjukkan pada **Gambar 3.19**

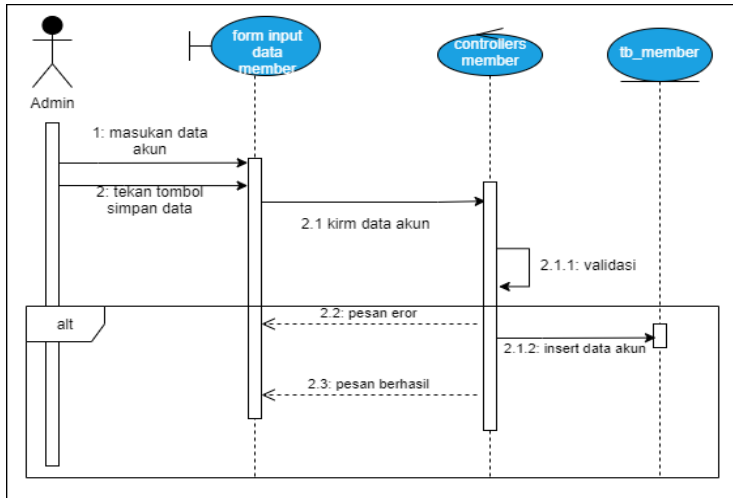


**Gambar 3. 21** *Sequence Diagram* Admin Melihat Data Pesanan

#### 15. *Sequence Diagram* Admin Input Data Member

Proses input data member yang dilakukan oleh Admin , dimulai dari admin melakukan input data pada form input data member, kemudian data dikirimkan ke *controller* member untuk dilakukan validasi, jika data gagal ditambah atau data tidak sesuai dan kurang lengkap maka akan muncul pesan error, jika berhasil maka data akan disimpan pada *tb\_member* dan akan muncul data berhasil disimpan.

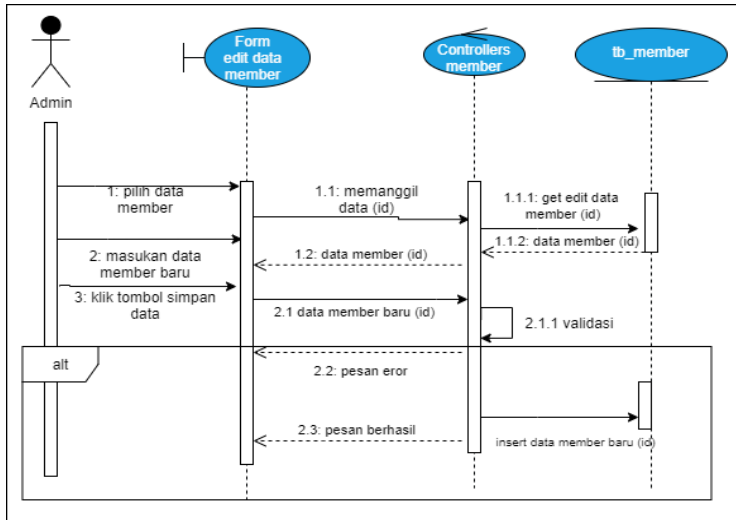
*Sequence Diagram* menambah data member yang dilakukan oleh admin ditunjukkan pada **Gambar 3.20**



**Gambar 3. 22** *Sequence Diagram* Admin Menambah Data Member

### 16. *Sequence Diagram* Admin Edit Data Member

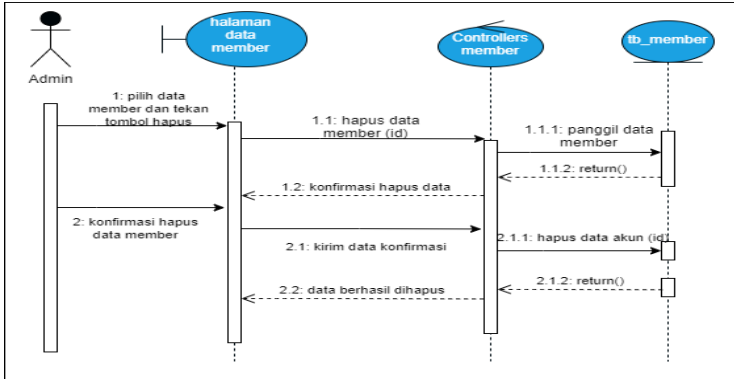
Proses *Edit* data member yang dilakukan oleh Admin , dimulai dari admin memilih data akun yang akan di*Edit* pada form *Edit* data member, kemudian data dikirimkan ke *controllers* member, lalu *controllers* memanggil data member melalui 'id' dari *tb\_member*. Lalu data akan muncul dan admin dapat melakukan input data yang baru. Data yang baru akan di validasi pada *controller* jika data gagal *Edit* maka akan muncul pesan gagal/*error* kepada admin namun jika data benar atau valid maka *controller* akan meng-*insert* data baru pada *database* dan *controller* akan memberikan pesan berhasil, Gambaran alur *Edit* data member yang dilakukan oleh admin ditunjukkan pada **Gambar 3.23**



**Gambar 3. 23** Sequence Diagram Admin Edit Data Member

### 17. Sequence Diagram Admin Hapus Data Member

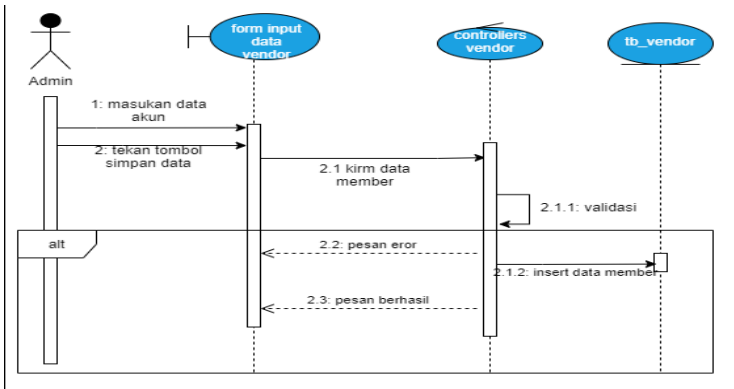
Proses hapus data member yang dilakukan oleh Admin diawali dengan admin memilih data member yang akan dihapus pada halaman data member, kemudian data akan dikirimkan ke *controllers member*, lalu *controllers* akan mengirimkan konfirmasi penghapusan data by 'id', jika Admin menerima maka data yang dipilih akan dihapus pada *database*, tetapi jika validasi tidak diterima maka data gagal dihapus, dan menampilkan notifikasi data gagal dihapus. *Sequence Diagram* menghapus data member yang dilakukan oleh admin ditunjukkan pada **Gambar 3.24**



Gambar 3. 24 Sequence Diagram Admin Hapus Data Member

**18. Sequence Diagram Admin Input Data Vendor**

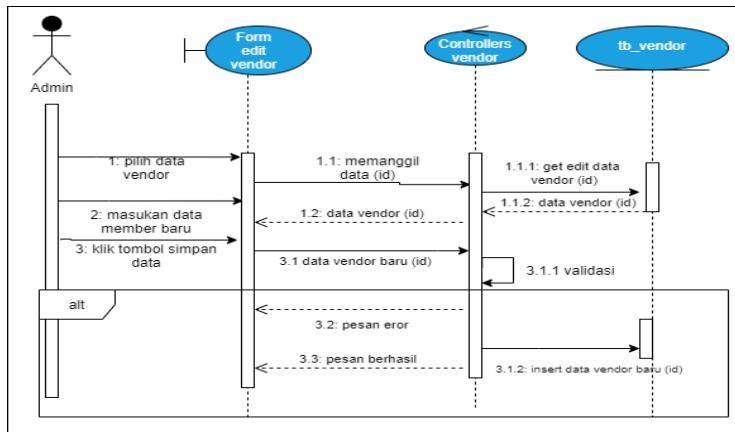
Proses input data vendor yang dilakukan oleh Admin , dimulai dari admin melakukan input data pada form input data member, kemudian data dikirimkan ke *controller* vendor untuk dilakukan validasi, jika data gagal ditambah atau data tidak sesuai dan kurang lengkap maka akan muncul pesan error, jika berhasil maka data akan disimpan pada *tb\_vendor* dan akan muncul data berhasil disimpan. *Sequence Diagram* menambah data vendor yang dilakukan oleh admin ditunjukkan pada **Gambar 3.25**



Gambar 3. 25 Sequence Diagram Tambah Data Vendor

### 19. Sequence Diagram Admin Edit Data Vendor

Proses *Edit* data vendor yang dilakukan oleh Admin , dimulai dari admin memilih data vendor yang akan di*Edit* pada form *Edit* data vendor, kemudian data dikirimkan ke *controllers* vendor, lalu *controllers* memanggil data akun melalui 'id' dari *tb\_vendor*. Lalu data akan muncul dan admin dapat melakukan input data yang baru. Data yang baru akan di validasi pada *controller* jika data gagal d*Edit* maka akan muncul pesan gagal/*error* kepada admin namun jika data benar atau valid maka *controller* akan meng-*insert* data baru pada *database* dan *controller* akan memberikan pesan berhasil, Gambaran alur *Edit* data vendor yang dilakukan oleh admin ditunjukkan pada **Gambar 3.26**

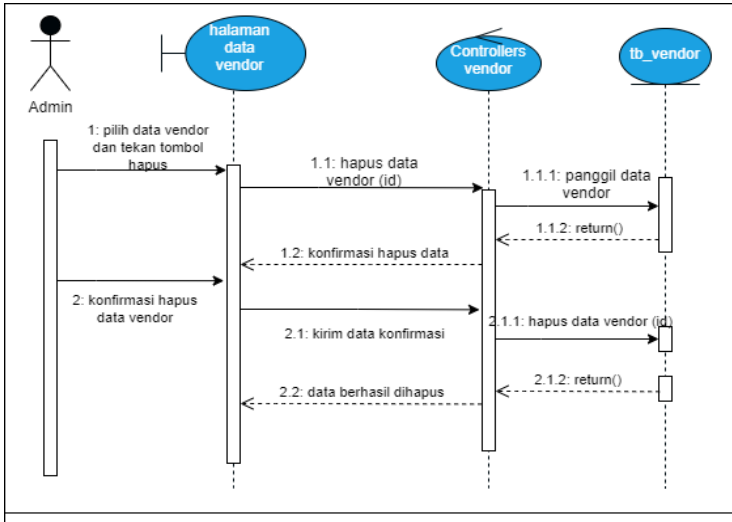


**Gambar 3. 26** Sequence Diagram Ubah Data Vendor

### 20. Sequence Diagram Admin Hapus Data Vendor

Proses hapus data member yang dilakukan oleh Admin diawali dengan admin memilih data vendor yang akan dihapus pada halaman data member , kemudian data akan dikirimkan ke *controllers* vendor, lalu *controllers* akan mengirimkan konfirmasi penghapusan data by 'id' , jika Admin menerima maka data yang dipilih akan dihapus pada *database*, tetapi jika validasi tidak diterima maka data gagal dihapus, dan menampilkan notifikasi data gagal dihapus. *Sequence Diagram*

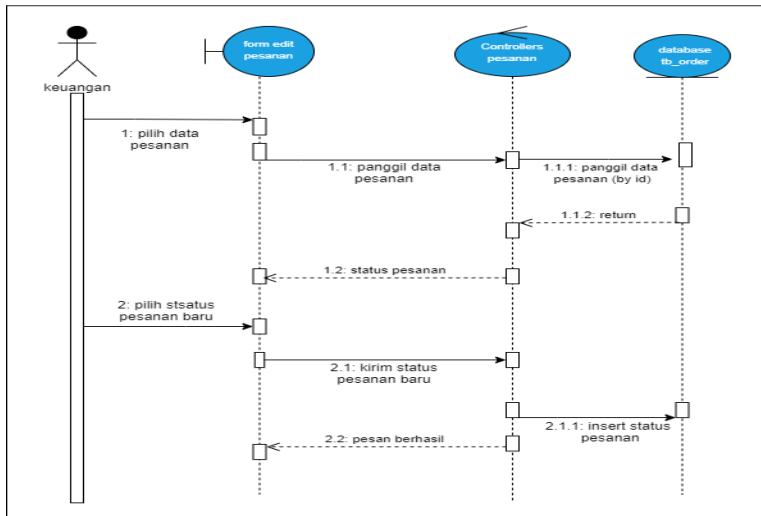
menghapus data vendor yang dilakukan oleh admin ditunjukkan pada **Gambar 3.27**



**Gambar 3. 27** *Sequence Diagram Hapus Data Vendor*

## 21. *Sequence Diagram Keuangan Update Status Pesanan*

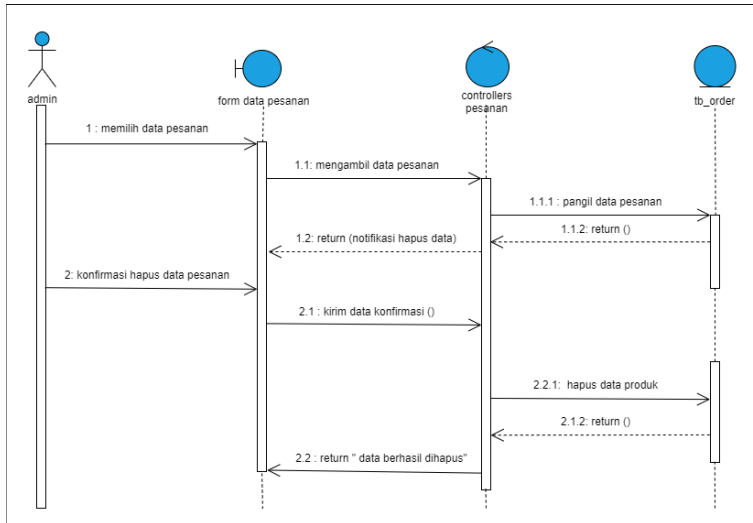
Rancangan proses mengupdate status pesanan yang dilakukan oleh keuangan diawali dengan keuangan memilih data pesanan yang akan diubah statusnya, lalu data akan disampaikan ke *controllers* pesanan dan memanggil data pesanan pada tabel *order*, ditampilkan pada form *Edit* status *Invoice*. Kemudian, data yang di panggil akan muncul dan admin dapat melakukan pengubahan sttus pesanan. Satus pesanan yang baru akan divalidasi oleh *controllers* jika *true* maka pesanan yang dipilih berhasil diubah statusnya dan data akan disimpan ke dalam *database* *tb order*. *Sequence Diagram* melakukan update status pesanan yang dilakukan oleh keuangan ditunjukkan pada **Gambar 3.28**



**Gambar 3. 28** *Sequence Diagram* Keuangan Update Status Pesanan

## 22. *Sequence Diagram* Admin Hapus Data Pesanan

Proses hapus data akun yang dilakukan oleh Admin diawali dengan admin memilih data pesanan yang akan dihapus pada form pesanan, kemudian data akan dikirimkan ke *controllers* pesanan, lalu *controllers* akan mengirimkan konfirmasi penghapusan data by 'id', jika admin menerima maka data yang dipilih akan dihapus pada *database* tabel *order*, tetapi jika validasi tidak diterima maka data gagal dihapus, dan menampilkan notifikasi data gagal dihapus. *Sequence Diagram* menghapus data pesanan yang dilakukan oleh admin ditunjukkan pada **Gambar 3.29**

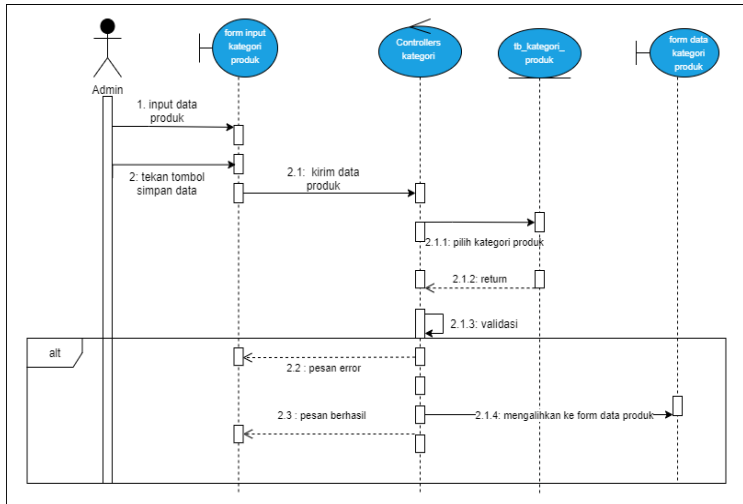


**Gambar 3. 29** *Sequence Diagram Admin Hapus Data Pesanan*

### 23. *Sequence Diagram Admin Tambah Data Kategori*

Proses admin menambahkan data dimulai dari admin memasukan data kategori produk pada form input kategori, kemudian oleh *controllers* kategori produk, data akan diolah dan dilakukan validasi, jika data yang dimasukan lengkap dan sesuai maka data akan tersimpan pada tabel kategori produk, dan admin akan dialihkan untuk melihat data yang diinput pada menu kategori produk. *Sequence Diagram* menambah data kategori produk yang dilakukan oleh admin ditunjukkan pada **Gambar 3.30**

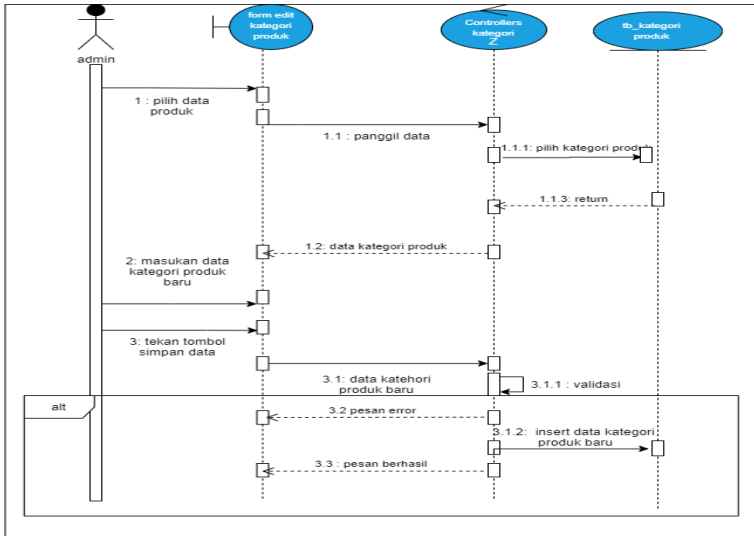




**Gambar 3. 30** Sequence Diagram Tambah Data Kategori Produk

#### 24. Sequence Digram Edit Data Kategori Produk

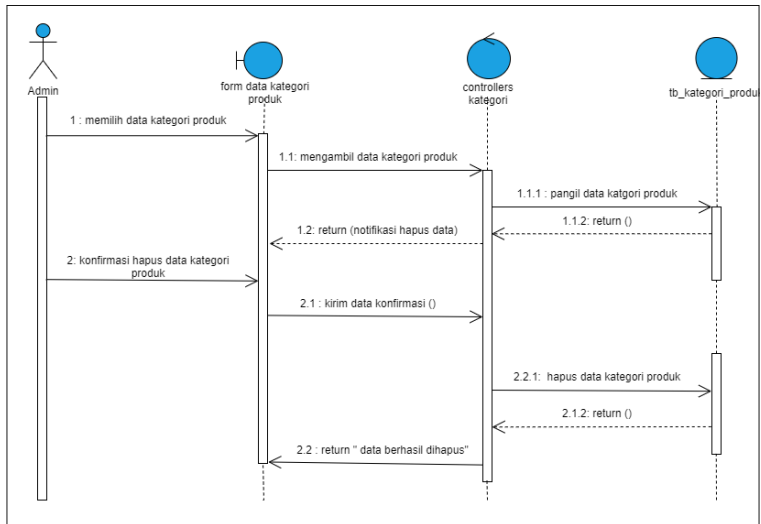
Proses *Edit* data katagori produk yang dilakukan oleh admin, dimulai dari admin memilih data kategori lalu menekan tombol *Edit*, setelah itu *controllers* akan melaukan panggilan data dari tabel kategori produk, dan akan ditampilkan pada form *Edit* data. Kemudian, admin akan memasukan data kategori produk yang baru dan akan dilakukan validasi oleh *conrollers* kategori produk, jika data yang dimasukan valid maka data baru akan didimpan kedalam tabel kategori produk dan akan menampilkan pesan berhasil, tetapi jika data tidak valid maka data tbaru tidak tersimpan dan sistem akan menampilkan pesan *error/gagal*. Sequence Diagram ubah data kategori produk dapat dilihat pada **Gambar3.31**



**Gambar 3. 31** *Sequence Diagram Edit Data Kategori Produk*

## 25. *Sequence Digram Hapus Data Kategori Produk*

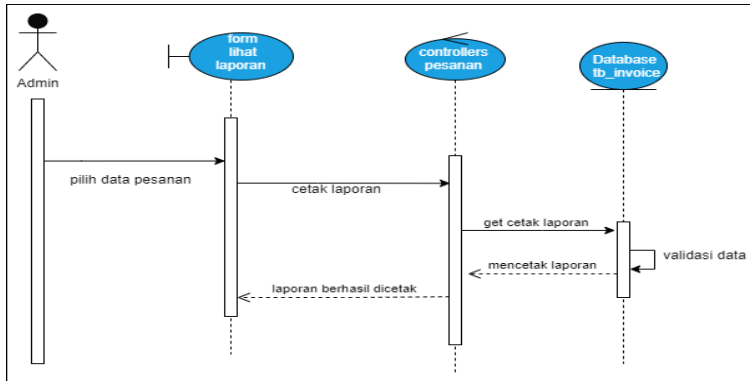
Proses hapus data kategori produk yang dilakukan oleh admin dimulai dari admin memilih data yang akan dihapus pada halaman kategori produk, kemudian pada *controllers* akan memproses dengan mengirimkan konfirmasi hapus data, jika pesan dikonfirmasi maka akan melakukan penghapusan data. Jika sistem berhasil maka data pada database akan terhapus, dan akan menampilkan pesan berhasil, tetapi jika sistem gagal, maka data yang dipilih gagal terhapus pada tabel kategori produk dan akan menampilkan pesan gagal. *Sequence Diagram* hapus data kategori dapat dilihat pada **Gambar3.32**



**Gambar 3. 32** *Sequence Diagram* Hapus Data Kategori Produk

## 26. *Sequence Diagram* Admin Cetak Invoice Pemesanan

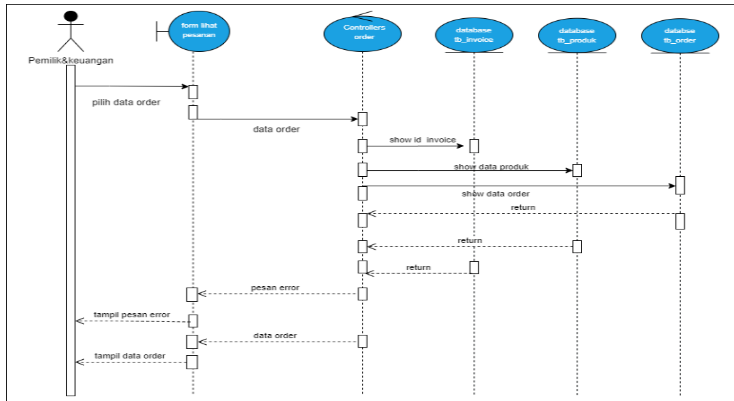
Proses mencetak laporan pemesana yang dilakukan oleh Admin dimulai dari admin memilih menu pesanan, maka pada sistem akan menampilkan data tersebut, kemudian admin menekan tombol print dan pada *controller* akan melakukan validasi sesuai dengan tabel *Invoice*, jika berhasil maka *Invoice* pemesanan akan berhasil dicetak. *Sequence Diagram* mencetak laporan pesanan yang dilakukan oleh admin ditunjukkan pada **Gambar 3.33**



**Gambar 3. 33** *Sequence Diagram* Admin Mencetak Invoice pesanan

## 27. *Sequence Diagram* pemilik dan Keuangan Lihat Data Pesanan

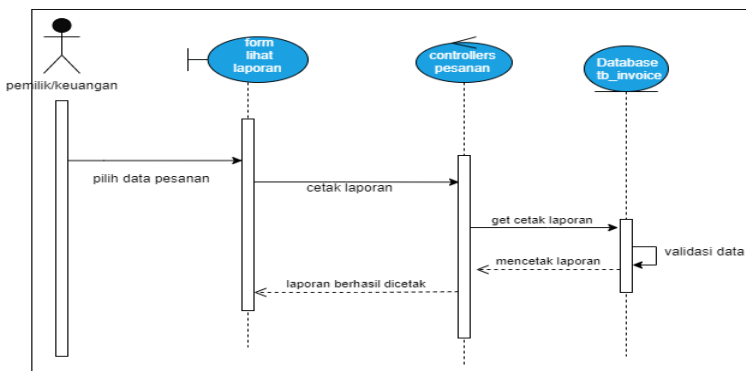
Rancangan proses untuk pemilik dan keuangan untuk melihat data pemesanan adalah dimulai dari member melihat form pesanan kemudian akan diolah pada *controllers* pesanan, kemudian *controllers* akan memanggil data Invoice pada tabel Invoice, dan memanggil data produk pada tabel produk, kemudian akan dilakukan validasi jika berhasil maka data order akan muncul pada halaman pesanan, tetapi jika gagal maka akan tampil pesan gagal. *Sequence Diagram* melihat data pesanan yang dilakukan oleh admin ditunjukkan pada **Gambar 3.34**



**Gambar 3. 34** Sequence Diagram Pemilik dan Keuangan Lihat Data Pesanan

## 28. Sequence Diagram Pemilik dan Keuangan Cetak Invoice

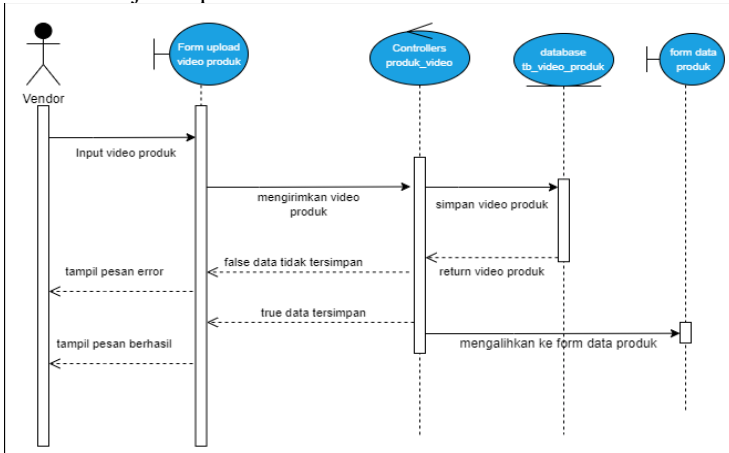
Proses mencetak laporan pemesana yang dilakukan oleh pemilik dan keuangan dimulai dari admin memilih menu pesanan, maka pada sistem akan menampilkan data tersebut, kemudian admin menekan tombol print dan pada *controller* akan melakukan validasi sesuai dengan tabel *Invoice*, lalu jika berhasil maka *Invoice* pemesanan akan berhasil dicetak. *Sequence Diagram* mencetak laporan pesanan yang dilakukan oleh admin ditunjukkan pada **Gambar 3.35**



**Gambar 3. 35** *Sequence Diagram* Cetak Invoice Pemesanan

### 29. *Sequence Diagram* Vendor Tambah Video Produk

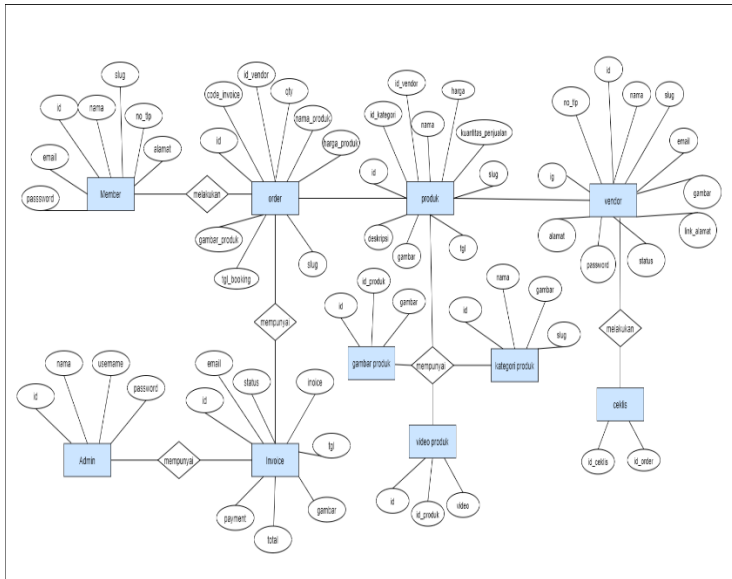
Proses menambah video produk yang dilakukan oleh vendor dimulai dari vendor menambahkan video produk pada form *upload* video produk, kemudian data akan dikirimkan ke *controllers* produk\_video dan akan dilakukan penyimpanan ke dalam *tb-video\_produk*. Jika data yang dimasukkan benar maka data akan tersimpan dan vendor akan dialihkan ke halaman data produk. *Sequence Diagram* mencetak laporan pesanan yang dilakukan oleh admin ditunjukkan pada **Gambar 3.36**



**Gambar 3. 36** *Sequence Diagram* Tambah Video

### C. Entity Relationship Diagram

Erd atau *Entity Relationship Diagram* merupakan *Diagram* struktural yang digunakan dalam membuat desain *database*. Erd Pada sistem yang diusulkan terdiri dari member yang dapat melakukan order, kemudian data riwayat order akan mempunyai *Invoice* yang dapat di akses oleh admin. Produk yang di kelola oleh vendor mempunyai katageori produk, gambar produk, dan video produk. Gambaran ERD yang diancang dapat dilihat pada **Gambar3.37**



**Gambar 3. 37** ERD Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias

Perancangan *database* pada Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias memiliki beberapa tabel yang terdiri atas tabel member, tabel order, tabel produk, tabel vendor, tabel admin, tabel *Invoice*, tabel kategori, tabel gambar produk dan tabel video produk Adapun struktur dari masing-masing tabel adalah sebagai berikut:

1. Tabel Admin
 

Nama Tabel	: tb_admin
Fungsi	: mengelola data admin
Primary Key	: id

**Tabel 3. 5** Tabel Admin

No	Nama field	Tipe data	Panjang	Keterangan
1.	id	int	11	<i>Primary key</i>
2.	nama	varchar	30	-
3.	username	varchar	30	-
4.	password	varchar	255	-

## 2. Tabel Member

Nama Tabel : tb\_member  
 Fungsi : mengelola data member  
 Primary Key : id

**Tabel 3. 6** Tabel Member

No	Nama field	Tipe data	Panjang	Keterangan
1.	id	int	11	<i>Primary key</i>
2.	nama	varchar	30	-
3.	slug	varchar	50	-
4.	email	varchar	50	Foreign key
5.	no_telp	varchar	13	-
6.	alamat	text	-	
7.	password	varchar	10	-

## 3. Tabel vendor

Nama Tabel : tb\_vendor  
 Fungsi : mengelola data vendor  
 Primary Key : id

**Tabel 3. 7** Tabel Vendor

No	Nama field	Tipe data	Panjang	Keterangan
1.	id	int	11	<i>Primary key</i>
2.	nama	varchar	30	-
3.	slug	varchar	50	-
4.	email	varchar	50	-
5.	no_telp	varchar	13	-
6.	ig	varchar	25	-
7.	alamat	text	-	
8.	password	varchar	10	-



9.	status	enum	-	
10.	gambar	varchar	255	-

4. Tabel produk  
 Nama Tabel : tb\_produk  
 Fungsi : mengelola data produk  
 Primary Key : id

**Tabel 3. 8** Tabel Produk

No	Nama field	Tipe data	Panjang	Keterangan
1.	id	int	11	<i>Primary key</i>
2.	Id_kategori	int	11	<i>Foreign key</i>
3.	Id_vendor	int	11	<i>Foreign key</i>
4.	nama	varchar	30	-
5.	harga	bigint	20	-
6.	kuantitas_penjualan	int	2	-
7.	deskripsi	Text	-	-
8.	gambar	varchar	255	-
9.	tgl	date	-	-
10	slug	varchar	50	-

5. Tabel orders  
 Nama Tabel : tb\_orders  
 Fungsi : mengelola data pesanana  
 Primary Key : id

**Tabel 3. 9** Tabel Order

No	Nama field	Tipe data	Panjang	Keterangan
1.	id	int	11	<i>Primary key</i>
2.	code_Invoice	varchar	50	<i>Foreign key</i>
3.	id_vendor	int	11	
4.	qty	int	3	-
5.	Nama_produk	varchar	50	-
6.	Harga_produk	bigint	8	-

7.	gambar_produk	varchar	255	-
8.	Tgl_booking	date	-	-
9.	slug	varchar	50	-
10	Nama_vendor	varchar	50	-

## 6. Tabel kategori produk

Nama Tabel : tb\_kategori\_produk  
 Fungsi : mengelola data produk  
 Primary Key : id

**Tabel 3. 10** Tabel Kategori Produk

No	Nama field	Tipe data	Panjang	Keterangan
1.	id	int	11	<i>Primary key</i>
2.	nama	varchar	30	-
3.	gambar	varchar	255	-
4.	slug	varchar	255	-

7. Tabel *Invoice*

Nama Tabel : tb\_*Invoice*  
 Fungsi : mengelola data riwayat pesanan  
 Primary Key : id

**Tabel 3. 11** Tabel *Invoice*

No	Nama field	Tipe data	Panjang	Keterangan
1.	id	int	11	<i>Primary key</i>
2.	email	varchar	50	<i>Foreign key</i>
3.	<i>Invoice</i>	varchar	50	<i>Foreign key</i>
4.	tgl	datetime	-	-
5.	status	enum	255	-
6.	payment	tinytext	-	-
7.	total	int	11	-
8.	gambar	varchar	255	-

## 8. Tabel Gambar Produk

Nama Tabel : tb\_*Invoice*

Fungsi : mengelola data riwayat pesanan  
 Primary Key : id

**Tabel 3. 12** Tabel Gambar Produk

No	Nama field	Tipe data	Panjang	Keterangan
1.	id	int	11	<i>Primary key</i>
2.	Id_produk	int	11	<i>Foreign key</i>
3.	gambar	varchar	255	-

9. Tabel Ceklis  
 Nama Tabel : tb\_ceklis  
 Fungsi : mengelola persiapan produk  
 Primary Key : id

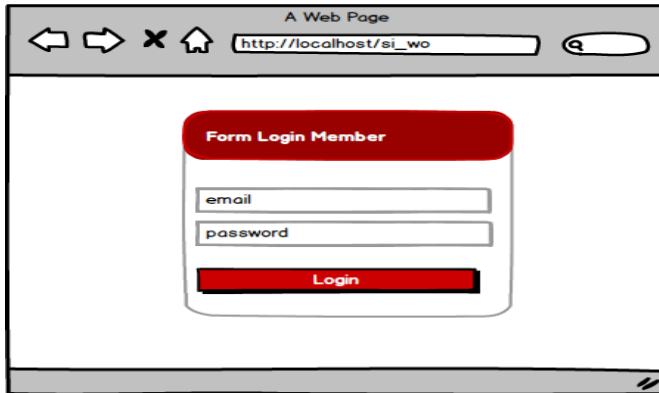
**Tabel 3. 13** Tabel Ceklis

No	Nama field	Tipe data	Panjang	Keterangan
1.	Id_ceklis	int	11	<i>Primary key</i>
2.	Id_order	int	11	-

### 3.2.4 Perancangan Antarmuka

Berikut merupakan perancangan antar muka yang akan digunakan dalam pengembangan Sistem Informasi Pemesanan Jasa *Wedding Organizer* Berbasis Web Dengan Menggunakan Notifikasi Whatsapp Di Indah Permata Rias :

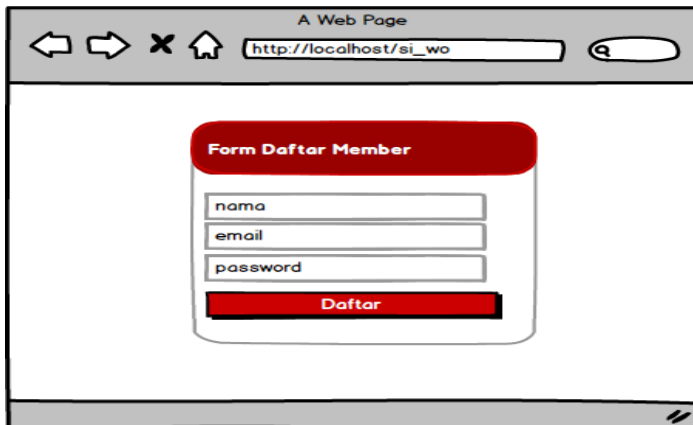
1. Rancangan Antarmuka Halaman *Login* member



**Gambar 3. 38** Rancangan Halaman *Login* Member

Rancangan Antarmuka yang tertera pada **Gambar 3.38** menjelaskan tentang gambaran halaman *Login* pada sistem yang akan dibuat. Pada halaman *Login* terdapat sebuah *form* yang terdiri dari *email* dan *password* yang harus diisi sebelum masuk ke halaman website

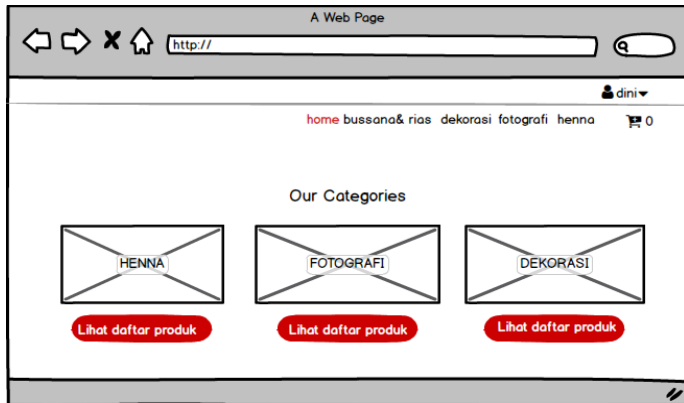
## 2. Rancangan Antarmuka Registrasi member



**Gambar 3. 39** Rancangan Halaman Registrasi Member

Rancangan Antarmuka yang tertera pada **Gambar 3.39** menjelaskan tentang gambaran halaman registrasi pada sistem yang akan dibuat. Pada halaman registrasi terdapat sebuah *form* yang terdiri dari nama, *email*, dan *password* yang harus diisi.

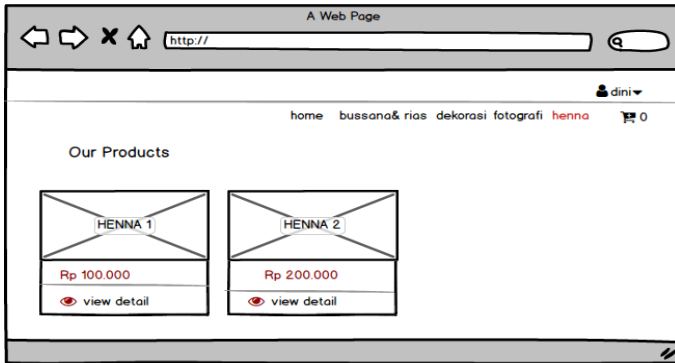
### 3. Rancangan Antarmuka halaman beranda *user*/pengguna



**Gambar 3. 40** Rancangan Beranda Member

Rancangan Antarmuka yang tertera pada **Gambar 3.40** menjelaskan tentang gambaran halaman beranda member yang berisi gambar mengenai kategori produk dan jasa yang ditawarkan. Pada halaman ini juga terdapat keranjang belanja yang digunakan untuk menampung produk yang akan dipesan oleh member.

### 4. Rancangan Antarmuka Halaman Produk Pada Member



**Gambar 3. 41** Rancangan Halaman Kategori

Rancangan Antarmuka yang tertera pada **Gambar 3.41** menjelaskan tentang gambaran halaman lihat produk yang dapat dilihat oleh member . halaman ini berisi foto produk dan harga berdasarkan kategori yang dipilih. Untuk melihat detail produk maka member harus menekan tombol *view detail*.

##### 5. Rancangan Antarmuka Halaman Lihat Detail Produk

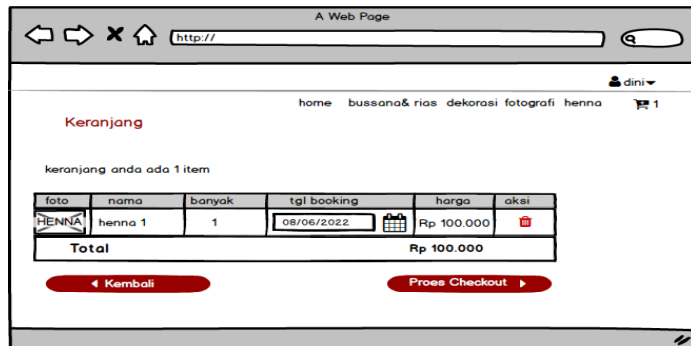


**Gambar 3. 42** Perancangan Halaman Lihat Produk

Rancangan antarmuka halaman lihat detail produk yang tertera pada **Gambar 3.42** berisi mengenai detail produk secara rinci mulai dari kategori produk, harga, stok, tanggal produk

diposting, dan deskripsi produk. Pada halaman ini juga terdapat tombol tambah ke keranjang yang digunakan untuk menambahkan produk tersebut ke keranjang sebelum dilakukan pemesanan, sebelum menambahkan produk ke keranjang member diharuskan mengisi tanggal pemesanan terlebih dahulu.

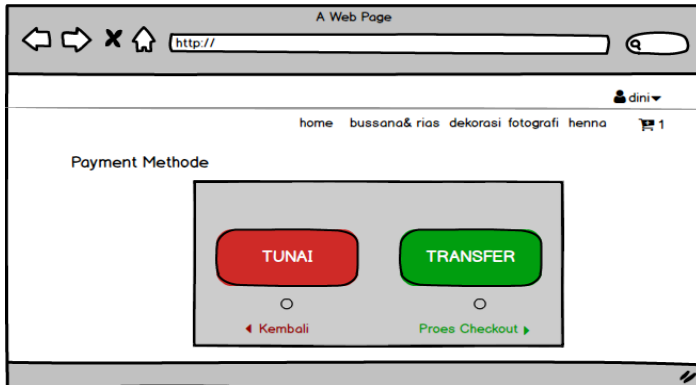
#### 6. Rancangan Antarmuka Halaman Keranjang Pada Member



**Gambar 3. 43** Perancangan Halaman Keranjang

Rancangan Antarmuka yang tertera pada **Gambar 3.43** menjelaskan tentang gambaran halaman keranjang yang berisi tabel dengan kolom foto, nama produk, tanggal booking, harga dan aksi hapus yang digunakan untuk menghapus produk yang terdapat dalam keranjang. Pada halaman ini terdapat tombol kembali yang digunakan oleh member jika belum ingin melakukan pemesanan, dan tombol proses checkout yang digunakan untuk melakukan proses pesanan.

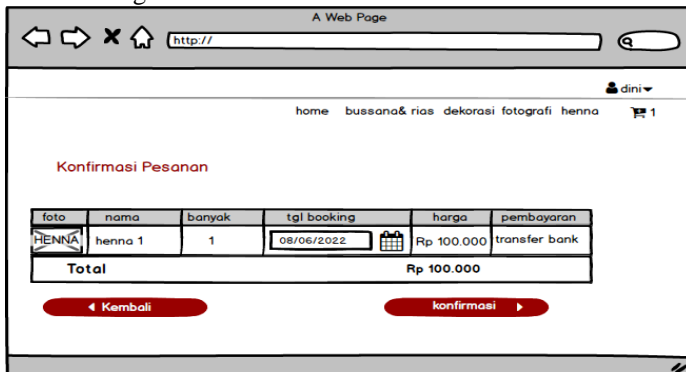
#### 7. Rancangan Antarmuka Halaman Pilih Pembayaran



**Gambar 3. 44** Perancangan Halaman Pilih Pembayaran

Rancangan Antarmuka yang tertera pada **Gambar 3.44** menjelaskan tentang halaman pilih pembayaran. Member bisa memilih pembayaran yang akan dilakukan dengan cara tunai atau transfer dengan mengklik tombol radio yang ada. Kemudian untuk melanjutkan pesanan maka member harus memilih tombol proses pesanan, tetapi jika tidak ingin melanjutkan proses pesanan member dapat menekan ombol kembali.

#### 8. Rancangan Antarmuka Halaman Konfirmasi Pesanan

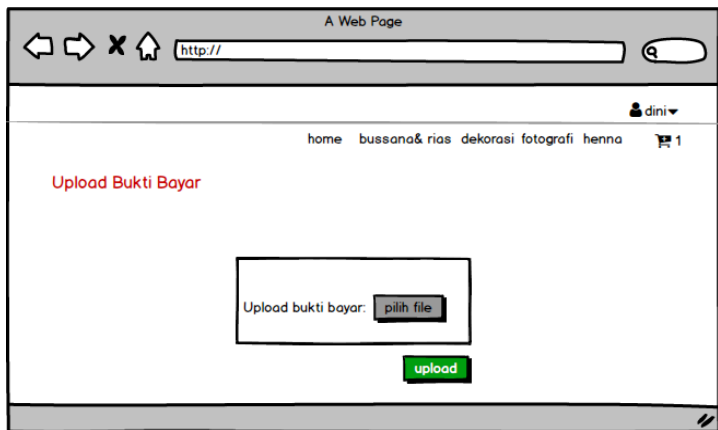


**Gambar 3. 45** Perancangan Konfirmasi Pesanan



Rancangan Antarmuka yang tertera pada **Gambar 3.45** menjelaskan tentang gambaran halaman konfirmasi pesanan yang berisi tabel dengan kolom foto, nama produk, tanggal booking, harga, pembayaran dan total harga pesanan. Pada halaman ini terdapat tombol kembali yang digunakan oleh member jika belum ingin melakukan konfirmasi pemesanan, dan tombol konfirmasi yang digunakan untuk mengkonfirmasi pesanan.

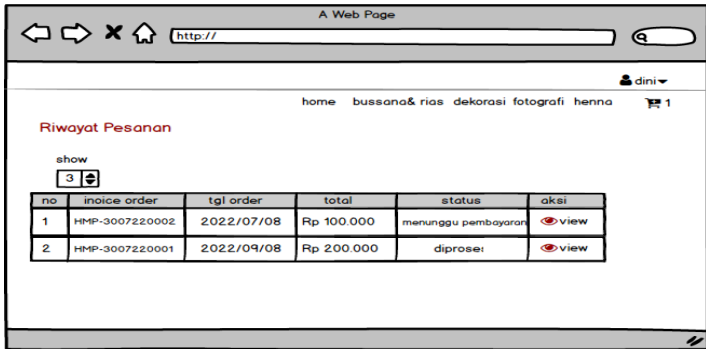
#### 9. Halaman *Upload* Bukti Pembayaran



**Gambar 3. 46** Perancangan Halaman *Upload* Bukti Pembayaran

Rancangan Antarmuka yang tertera pada **Gambar 3.46** menjelaskan tentang halaman pilih pembayaran. Member bisa memilih pembayaran yang akan dilakukan dengan cara tunai atau transfer dengan mengklik tombol yang ada.

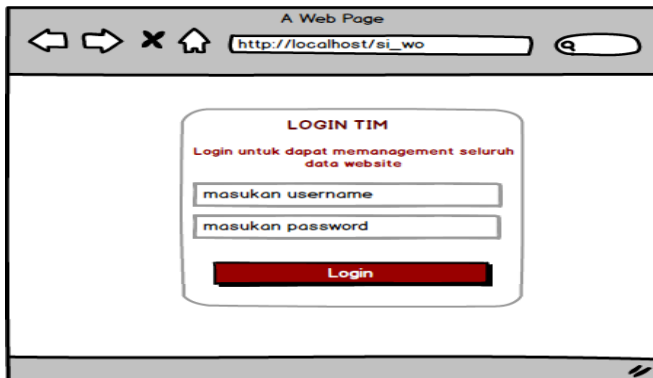
#### 10. Rancangan Antarmuka Halaman Riwayat Pesanan



**Gambar 3. 47** Perancangan Halaman Riwayat Pesanan

Rancangan Antarmuka yang tertera pada **Gambar 3.47** menjelaskan tentang halaman lihat riwayat pesanan yang berbentuk tabel. Pada halaman ini terdapat status pesanan yang dapat dilihat oleh member, dan pada tabel riwayat pesanan terdapat aksi *view* yang digunakan untuk melihat detail data pesanan.

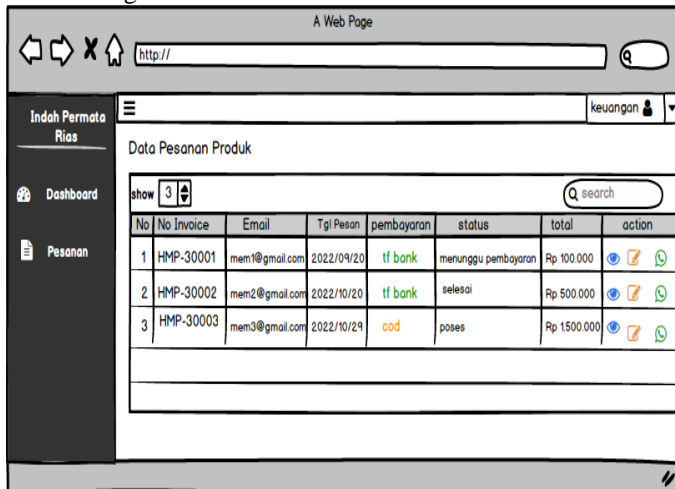
#### 11. Rancangan Antarmuka Halaman *Login* Tim (admin, keuangan, da pemilik)



**Gambar 3. 48** Perancangan Antarmuka *Login* Tim

Rancangan Antarmuka yang tertera pada **Gambar 3.48** menjelaskan tentang gambaran halaman *login* yang digunakan oleh tim yaitu admin, keuangan, dan. Pada halaman ini terdapat sebuah *form* yang terdiri dari *username* dan *password* yang harus diisi dan terdapat tombol *login* yang digunakan untuk masuk kedalam sistem.

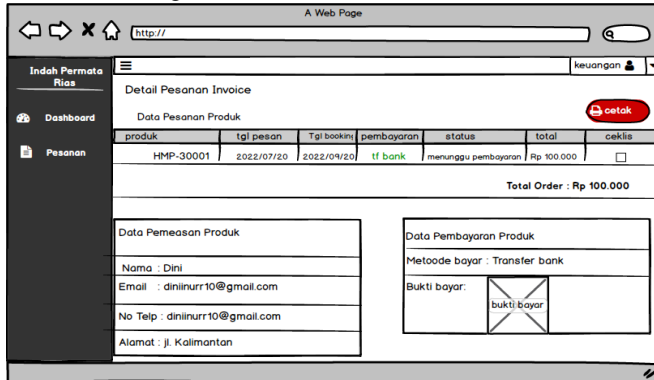
## 12. Rancangan Antarmuka Halaman Lihat Data Pesanan Pada Keuangan



**Gambar 3. 49** Perancangan Halaman Lihat Data Pesanan Pada Keuangan

Rancangan Antarmuka yang tertera pada **Gambar 3.49** menjelaskan mengenai halaman lihat data pesanan yang berupa tabel dengan kolom nomor, nomor *Invoice*, email, tanggal pesan, pembayaran, status, total dan aksi yang berupa lihat detail pesanan, *Edit* status pesanan, dan kirim notifikasi pesanan melalui WhatsApp kepada member. Pada halaman ini juga terdapat *search box* yang digunakan untuk mencari data yang diinginkan.

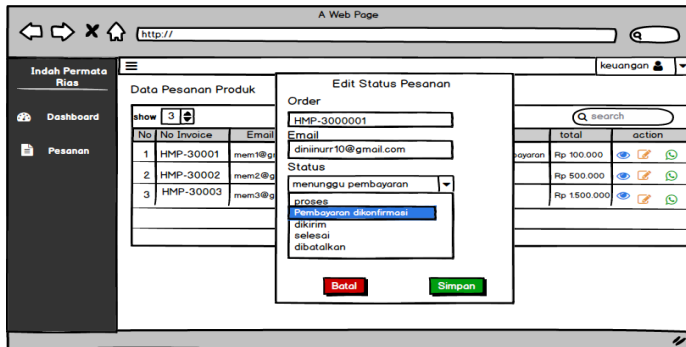
### 13. Rancangan Antarmuka Halaman Lihat Detail Pesanan Pada Keuangan



**Gambar 3. 50** Perancangan Lihat Detail Data Pesanan Pada Keuangan

Rancangan antar muka halaman yang tertera pada **Gambar3.50** menggambarkan mengenai detail pesanan yang dapat dilihat oleh aktor keuangan. Pada halaman ini terdapat tiga tabel yaitu tabel data pesanan produk, tabel data pemesan produk, dan tabel data pembayaran produk. Pada tabel data pesanan produk terdapat *checkbox* yang digunakan oleh vendor untuk menyiapkan pesanan, jika pada bagian tersebut sudah terdapat ceklis berarti vendor telah menyiapkan pesanan dan bagian keuangan dapat mengganti status pesanan.

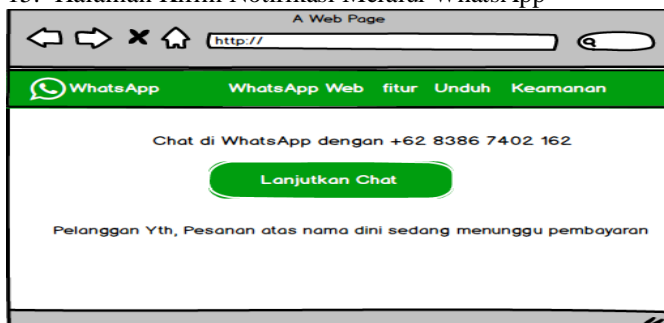
### 14. Rancangan Antarmuka Ubah Status Pesanan Pada Keuangan



**Gambar 3. 51** Perancangan Ubah Status Pesanan Pada keuangan

Rancangan antar muka halaman yang tertera pada **Gambar3.51** menggambarkan mengenai halaman ubah status pesanan yang dapat dilakukan oleh aktor keuangan. Pada halaman ini terdapat form yang berisi no Invoice, emmail dan status. Bagian keuangan dapat mengatur status pesanan member sesuai dengan keadaan pesanan, tekan tombol simpan untuk menyimpan status pesanan yang baru, dan tekan tombol batal untuk membatalkan mengubah status pesanan.

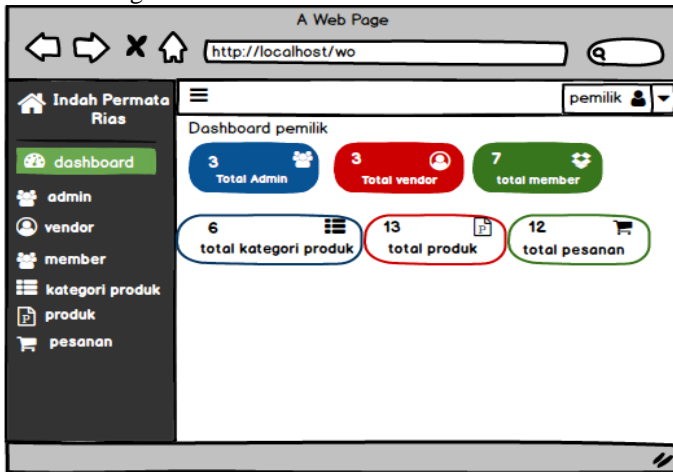
#### 15. Halaman Kirim Notifikasi Melalui WhatsApp



**Gambar 3. 52** Perancangan Halaman Kirim Notifikasi Pesanan Melalui WhatApp

Rancangan antar muka halaman yang tertera pada **Gambar3.52** menggambarkan mengenai halaman yang digunakan oleh keuangan untuk mengirimkan notifikasi pesan sesuai dengan status pesanan melalui WhatsApp kepada member.

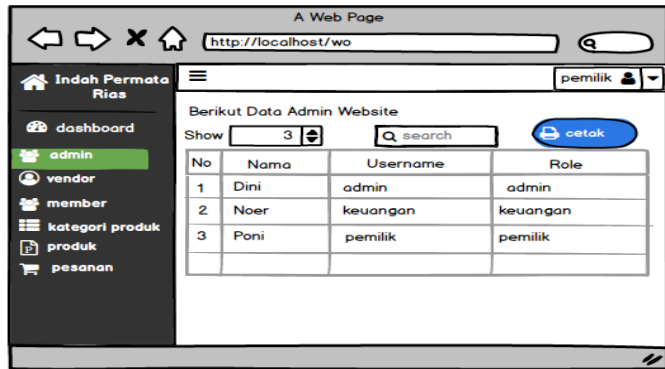
#### 16. Rancangan Antarmuka Halaman *Dashboard* Pemilik



**Gambar 3. 53** Perancangan Halaman *Dashboard* Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar3.53** menggambarkan mengenai halaman *Dashboard* atau halaman yang pertama kali dilihat oleh aktor pemilik ketika sudah berhasil *Login*. Pada halaman ini menampilkan data otal admin, total vendor, total member, total kategori produk, total produk, dan total pesanan.

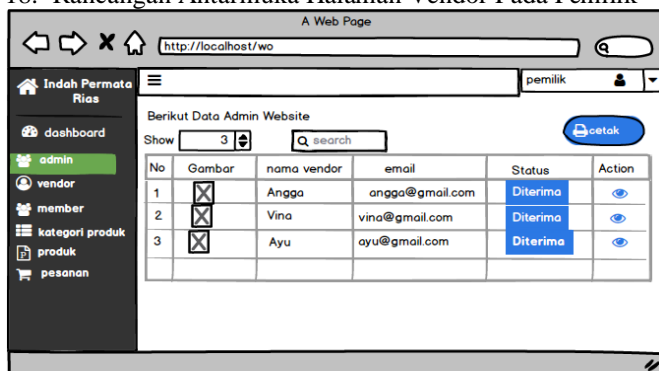
#### 17. Rancangan Antarmuka Halaman Admin Pada Pemilik



**Gambar 3. 54** Perancangan Halaman Data Admin Pada Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar3.54** menggambarkan mengenai halaman data admin yang terdapat dalam tabel dengan kolom nomor, nama, *username*, dan *role*. Aktor pemilik dapat melakukan cetak data admin dengan menekan tombol cetak, dan dapat melakukan pencarian data pada *search box*.

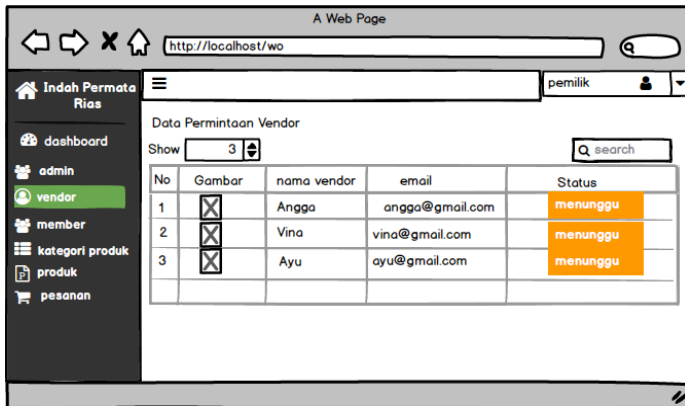
#### 18. Rancangan Antarmuka Halaman Vendor Pada Pemilik



**Gambar 3. 55** Perancangan Halaman Data Vendor Pada Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar 3.55** menggambarkan mengenai halaman data vendor yang terdapat dalam tabel dengan kolom nomor, nama vendor, *email*, status dan *action* lihat detail yang digunakan untuk melihat detail data vendor. Aktor pemilik dapat melakukan cetak data vendor dengan menekan tombol cetak, dan dapat melakukan pencarian data pada *search box*.

#### 19. Rancangan Antarmuka Halaman Permintaan Vendor Pada Pemilik

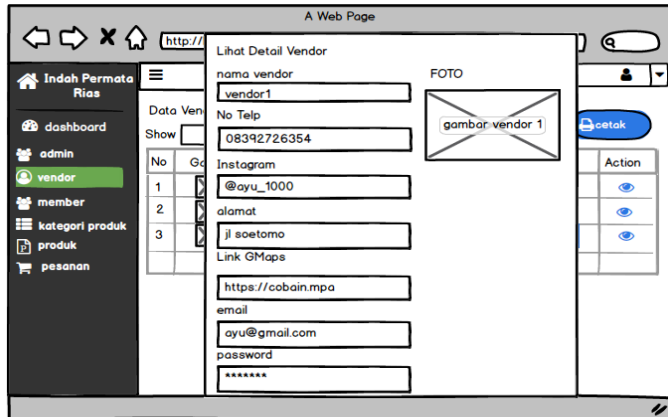


**Gambar 3. 56** Perancangan Halaman Permintaan Vendor Pada Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar 3.56** menggambarkan mengenai halaman permintaan vendor yang terdapat dalam tabel dengan kolom nomor, gambar, nama vendor, *email*, dan status. Jika status masih menunggu maka vendor tersebut belum diterima oleh bagian admin, untuk dapat bergabung maka vendor harus diterima terlebih dahulu oleh bagian admin. Aktor pemilik dapat melakukan pencarian data pada *search box*.



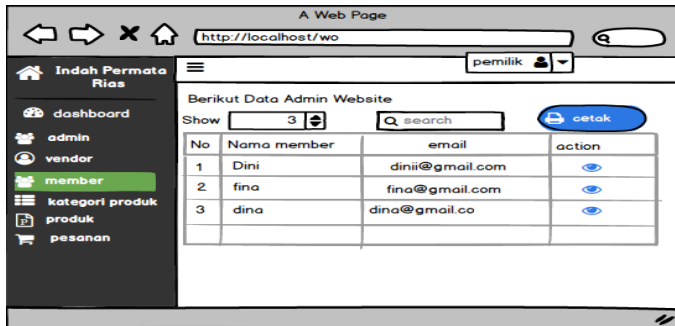
## 20. Rancangan Antarmuka Halaman Lihat Detail Vendor Pada Pemilik



**Gambar 3. 57** Perancangan Halaman Lihat Detail Vendor Pada Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar 3.57** menggambarkan mengenai halaman lihat detail data vendor yang dapat dilihat oleh aktor pemilik yang terdapat dalam sebuah form yang berisi nama vendor, nomor telephone, insatgram, alamat, *email*, *password*, dan gambar.

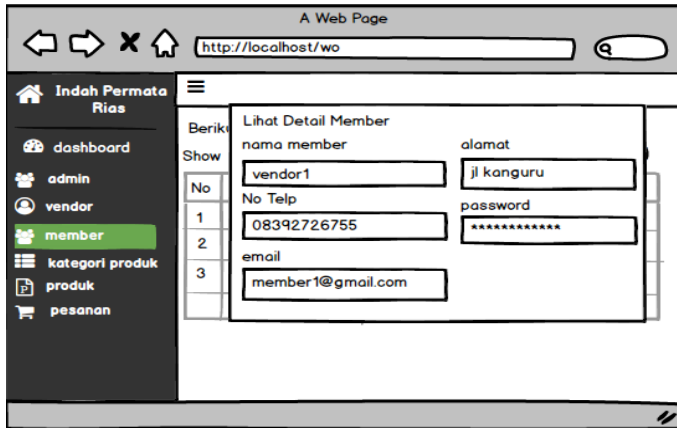
## 21. Rancangan Antarmuka Halaman Data Member Pada Pemilik



**Gambar 3.58** Perancangan Halaman Lihat Data Member Pada Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar 3.58** menggambarkan mengenai halaman data member yang terdapat dalam tabel dengan kolom nomor, nama member, *email*, dan *action* lihat detail yang digunakan untuk melihat detail data member. Aktor pemilik dapat melakukan cetak data member dengan menekan tombol cetak, dan dapat melakukan pencarian data pada *search box*.

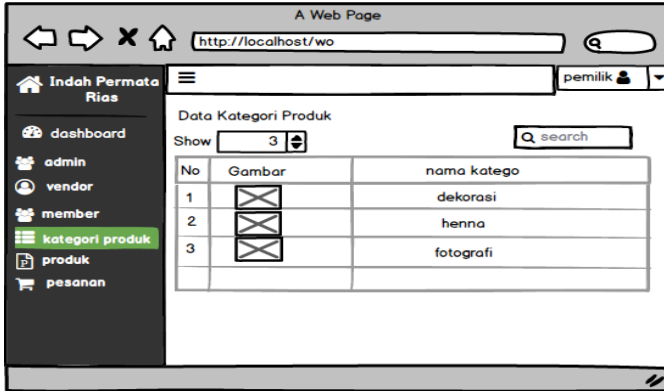
## 22. Rancangan Antarmuka Halaman Lihat Detail Data Member Pada Pemilik



**Gambar 3. 59** Prancangan Halaman Lihat Detail Data Member Pada Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar 3.59** menggambarkan mengenai halaman lihat detail data member yang dapat dilihat oleh aktor pemilik yang terdapat dalam sebuah form yang berisi nama member, nomor telephone, alamat, *email* dan *password*.

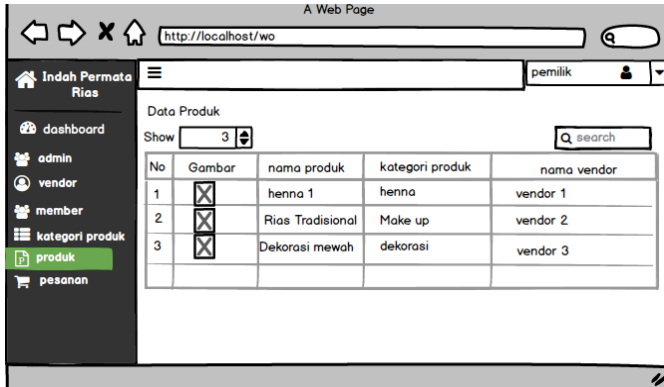
### 23. Rancangan Antarmuka Halaman Data Kategori Pada Pemilik



**Gambar 3. 60** Perancangan Halaman Kategori Pada Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar 3.60** menggambarkan mengenai halaman kategori yang terdapat dalam tabel dengan kolom nomor, gambar, dan nama kategori. Aktor pemilik dapat pencarian data pada *search box*.

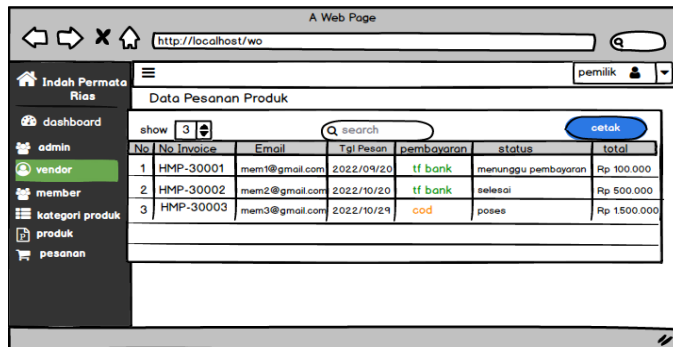
#### 24. Rancangan Antarmuka Halaman Data Produk Pada Pemilik



**Gambar 3. 61** Perancangan Halaman Data Produk Pada Pemilik

Rancangan antar muka halaman yang tertera pada **Gambar 3.61** menggambarkan mengenai halaman produk yang dapat dilihat oleh aktor pemilik yang terdapat dalam tabel dengan kolom nomor, gambar, nama pproduk,dan nama kategori. Aktor pemilik dapat pencarian data pada *search box*.

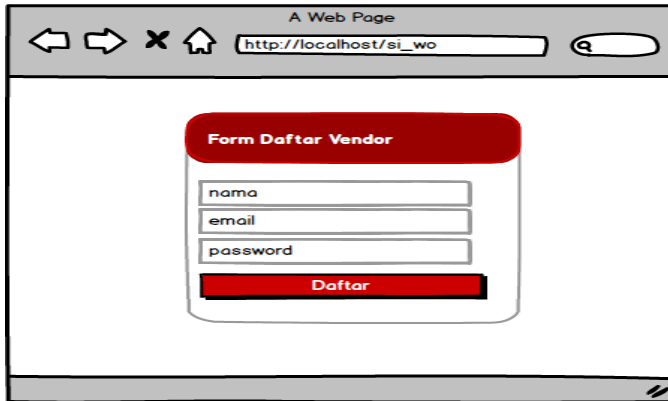
## 25. Rancangan Antarmuka Halaman Data Pesanan Pada Pemilik



**Gambar 3. 62** Perancangan Halaman Data Pesanan Pada Pemilik

Rancangan Antarmuka yang tertera pada **Gambar 3.62** menjelaskan mengenai halaman lihat data pesanan yang berupa tabel dengan kolom nomor, nomor *Invoice*, email, tanggal pesan,pembayaran, status, dan total. Pada halaman ini pemilik dapat melakukan cetak laporan dengan menekan tombol cetak dan pada halaman ini terdapat *search box* yang digunakan untuk mencari data yang diinginkan.

## 26. Rancangan Antarmuka Halaman Registrasi Vendor

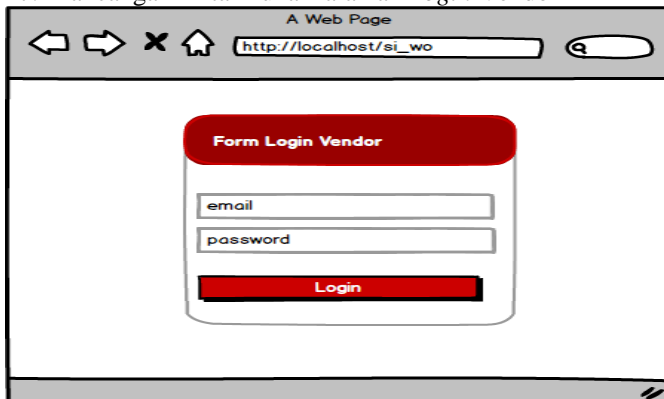


A screenshot of a web browser window titled "A Web Page" with the address bar showing "http://localhost/si\_wo". The main content area displays a registration form titled "Form Daftar Vendor". The form consists of three input fields labeled "nama", "email", and "password", followed by a red button labeled "Daftar".

**Gambar 3. 63** Perancangan Halaman Registrasi Vendor

Rancangan Antarmuka yang tertera pada **Gambar 3.63** menjelaskan mengenai gambaran halaman registrasi pada sistem yang akan dibuat. Pada halaman rgistrasi terdapat sebuah *form* yang terdiri dari nama,*email*, dan *password* yang harus diisi. Tekan tombol daftar untuk menyimpan data.

#### 27. Rancangan Antarmuka Halaman *Login* Vendor

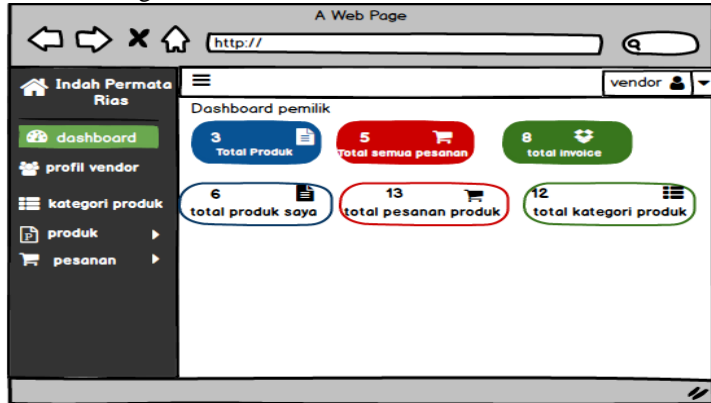


A screenshot of a web browser window titled "A Web Page" with the address bar showing "http://localhost/si\_wo". The main content area displays a login form titled "Form Login Vendor". The form consists of two input fields labeled "email" and "password", followed by a red button labeled "Login".

**Gambar 3. 64** Perancangan Halaman *Login* Vendor

Rancangan Antarmuka yang tertera pada **Gambar 3.64** menjelaskan mengenai gambaran halaman *Login* vendor pada sistem yang akan dibuat. Pada halaman *Login* terdapat sebuah *form* yang terdiri dari *email*, dan *password* yang harus diisi. Tekan tombol *Login* untuk dapat masuk kedalam sistem.

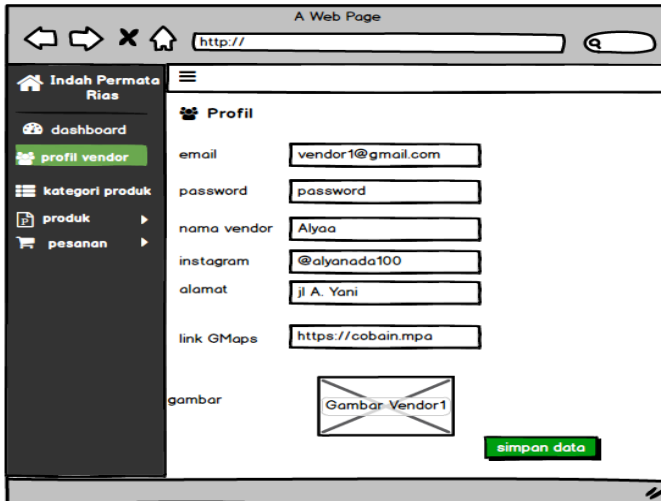
## 28. Rancangan Antarmuka Halaman *Dashboard* Vendor



**Gambar 3. 65** Perancangan Halaman *Dashboard* Vendor

Rancangan antar muka halaman yang tertera pada **Gambar3.65** menggambarkan mengenai halaman *Dashboard* atau halaman yang pertama kali dilihat oleh aktor vendor ketika sudah berhasil *Login*. Pada halaman ini menampilkan data total semua produk, total semua pesanan, total *Invoice*, total produk saya, total pesanan produk saya, dan total kategori produk. Pada halaman ini juga terdapat menu yang dapat digunakan oleh cendor untuk mengelola data pesanan dan data produk.

## 29. Rancangan Antarmuka Halaman Profil Vendor

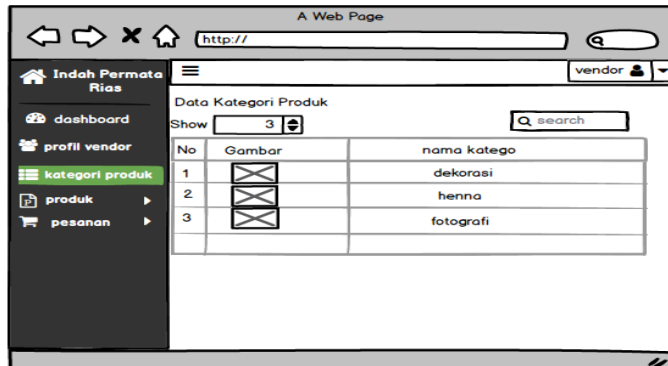


**Gambar 3. 66** Perancangan Halaman Profil Vendor

Rancangan antar muka halaman yang tertera pada **Gambar3.66** menggambarkan mengenai halaman profil vendor, haaman ini digunakan oleh vendor untuk menambah atau mengubah data profil yang terdiri dari nama vendor, instagram, alamat, serta gambar vendor. Sebelum mengubah atau menambah data profil, vendor harus memasukan password terlebih dahulu.

### 30. Rancangan Antarmuka Halaman Kategori Produk Pada Vendor

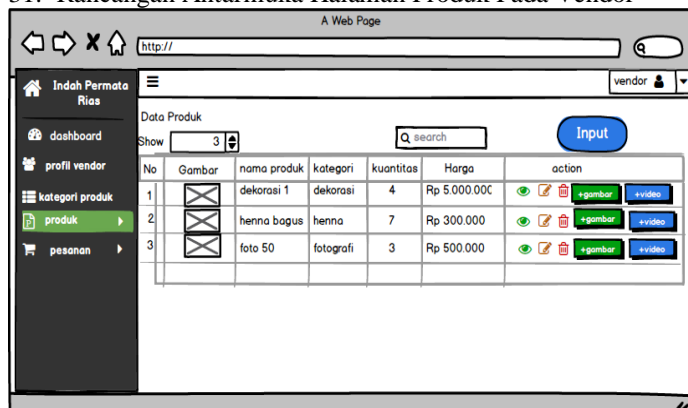




**Gambar 3. 67** Perancangan Halaman Kategori Produk Pada Vendor

Rancangan antar muka halaman yang tertera pada **Gambar 3.67** menggambarkan mengenai halaman kategori yang terdapat dalam tabel dengan kolom nomor, gambar, dan nama kategori. Aktor vendor dapat pencarian data pada *search box*.

### 31. Rancangan Antarmuka Halaman Produk Pada Vendor



**Gambar 3. 68** Perancangan Halaman Produk Pada Vendor

Rancangan antar muka halaman yang tertera pada **Gambar 3.68** menggambarkan mengenai halaman produk yang terdapat dalam tabel dengan kolom nomor, gambar, dan nama produk, kategori, kuantitas, harga, dan aksi lihat detail, *Edit*, hapus dan tambah data pada tombol input. Aktor vendor dapat pencarian data pada *search box*.

### 32. Rncangan Antarmuka Halaman Tambah Produk Pada Vendor

**Gambar 3. 69** Perancangan Halaman Tambah Data Produk

Rancangan antar muka halaman yang tertera pada **Gambar 3.69** menggambarkan mengenai halaman tambah data produk, untuk menambah data produk, vendor harus mengisi semua data yang ada pada form input produk, kemudian tekan simpan data untuk meyimpan data produk.

### 33. Rancangan Antarmuka Halaman Ubah Data Produk

A Web Page  
http://

Indah Permata Rias

dashboard  
profil vendor  
kategori produk  
produk  
pesanan

Form Input Produk

kategori: pilih kategori

nama produk: henna 1

Harga: Rp.

kuantitas: jumlah

dekripsi: masukan text

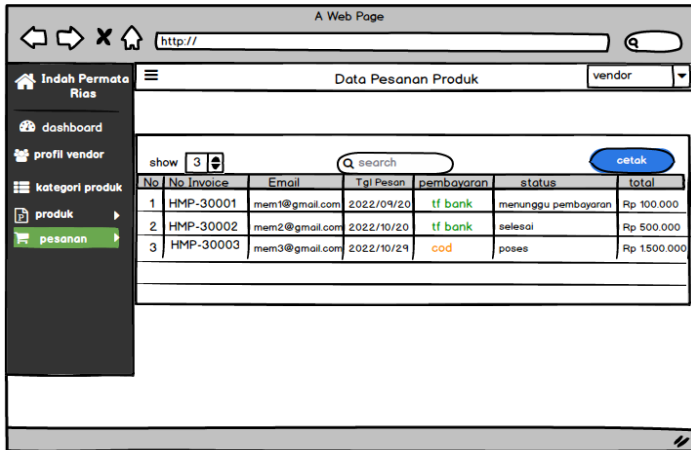
gambar: pilih file, Gambar henna

edit data

**Gambar 3. 70** Perancangan Halaman Ubah Data Produk

Rancangan antar muka halaman yang tertera pada **Gambar 3.70** menggambarkan mengenai halaman ubah data produk, untuk mengubah data produk, vendor harus mengisi semua data yang ada pada form *Edit* produk, kemudian tekan simpan data untuk menyimpan data produk yang baru.

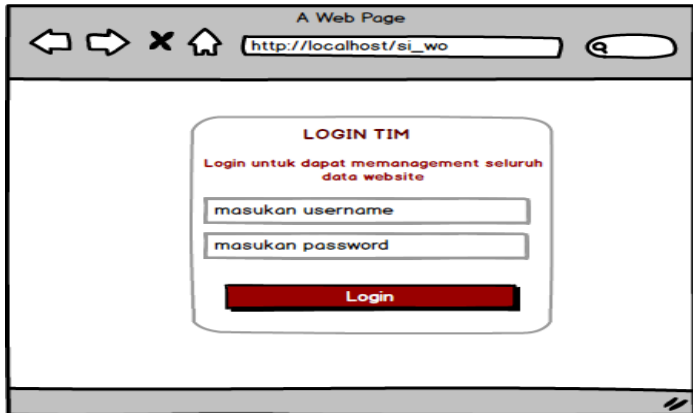
#### 34. Rancangan Antarmuka Halaman Lihat Data Pesanan



**Gambar 3. 71** Perancangan Antarmuka Halaman Lihat Data Pesanan Pada Vendor

Rancangan antar muka halaman yang tertera pada **Gambar 3.71** menggambarkan mengenai halaman lihat data pesanan yang terdapat dalam tabel dengan kolom nomor, nomor *Invoice*, email, tanggal pesan, pembayaran, status dan toal. Aktor vendor dapat melakukan cetak data pesanan dengan menekan tombol cetak dan dapat melakukan pencarian data pada *search box*.

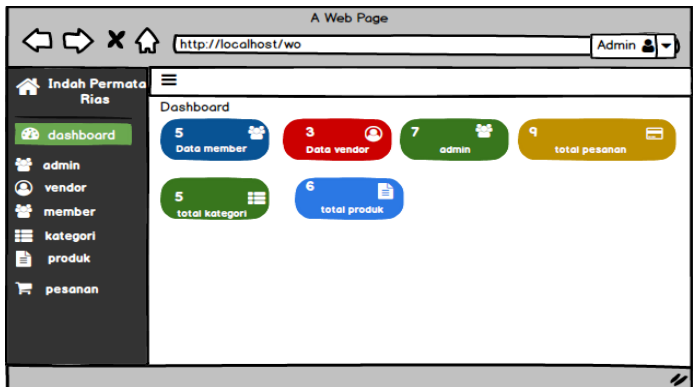
35. Halaman *Login* Tim (admin, keuangan dan pemilik)



**Gambar 3. 72** Perancangan Halaman *Login* Tim

Rancangan antar muka halaman yang tertera pada **Gambar 3.72** menggambarkan mengenai halaman *login* yang digunakan oleh tim (admin, keuangan dan pemilik), Aktor tim harus memasukan *username* dan *password* untuk dapat masuk kedalam sistem.

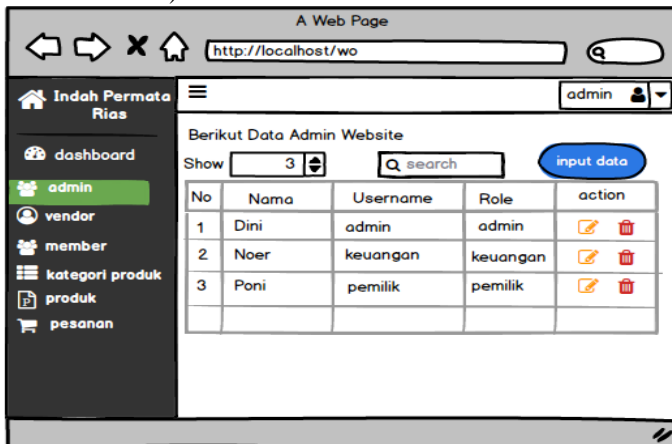
### 36. Rancangan Antarmuka Halaman *Dashboard* Pemilik



**Gambar 3. 73** Perancangan Halaman *Dashboard* Admin

Rancangan antar muka halaman yang tertera pada **Gambar3.73** menggambarkan mengenai halaman *Dashboard* atau halaman yang pertama kali dilihat oleh aktor pemilik ketika sudah berhasil *Login*. Pada halaman ini menampilkan data total semua produk, total semua pesanan, total *Invoice*, dan total kategori produk. Pada halaman ini juga terdapat menu yang digunakan untuk mengelola data member, tim, vendor, dan mengelola data kategori.

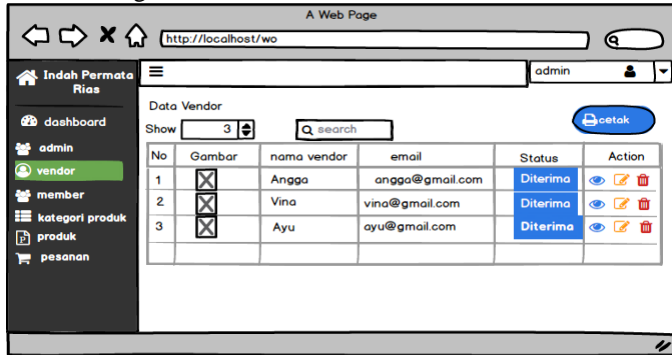
### 37. Rancangan Antarmuka Halaman Tim (admin, keuangan dan vendor)



**Gambar 3. 74** Perancangan Halaman Tim

Rancangan antar muka halaman yang tertera pada **Gambar3.74** menggambarkan mengenai halaman data admin yang terdapat dalam tabel dengan kolom nomor, nama, *username*, *role*, dan aksi yang berupa aksi *edit*, data dan hapus data. Aktor admin dapat melakukan cetak data admin dengan menekan tombol cetak, dan dapat melakukan pencarian data pada *search box*.

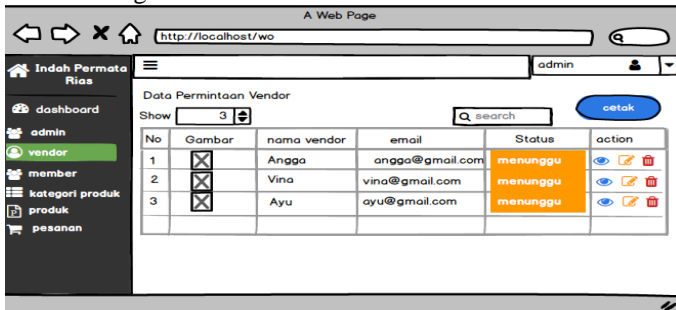
### 38. Rancangan Antarmuka Halaman Vendor



**Gambar 3. 75** Perancangan Halaman Vendor Pada Admin

Rancangan antar muka halaman yang tertera pada **Gambar 3.75** menggambarkan mengenai halaman vendor yang terdapat dalam tabel dengan kolom nomor, gambar, nama vendor, *email*, dan status. Admin dapat mengubah status pada vendor diterima atau ditolak. Admin juga dapat melakukan lihat detail data vendor, *Edit* data vendor dan hapus data vendor, selain itu admin juga dapat melakukan pencarian data pada *search box*.

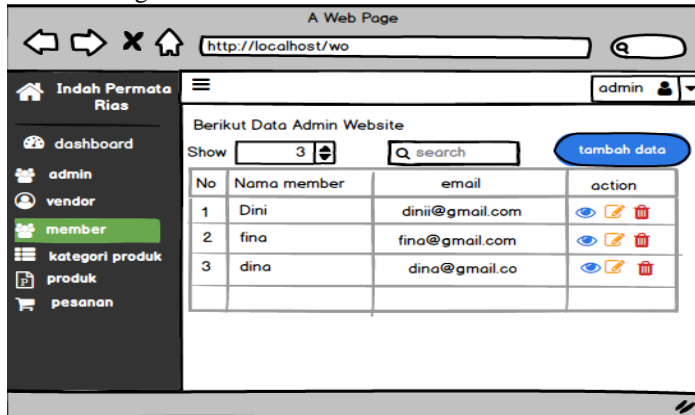
### 39. Rancangan Antarmuka Halaman Permintaan Vendor



**Gambar 3. 76** Perancangan Halaman Permintaan Vendor Pada Admin

Rancangan antar muka halaman yang tertera pada **Gambar 3.76** menggambarkan mengenai halaman permintaan vendor yang terdapat dalam tabel dengan kolom nomor, gambar, nama vendor, *email*, dan status. Jika status masih menunggu maka vendor tersebut belum diterima, untuk dapat bergabung maka vendor harus diterima terlebih dahulu oleh bagian admin. Aktor admin dapat melakukan ubah status vendor, hapus data, lihat detail vendor yang mendaftar, dan tambah data vendor. Admin juga dapat melakukan pencarian data pada *search box*.

#### 40. Rancangan Antarmuka Halaman Member

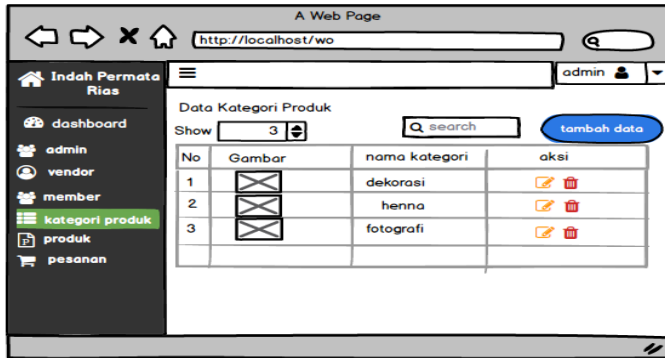


**Gambar 3. 77** Perancangan Halaman Member Pada Admin

Rancangan antar muka halaman yang tertera pada **Gambar 3.77** menggambarkan mengenai halaman member yang telah melakukan pendaftaran. Admin dapat melakukan lihat detail data member, input data, hapus data dan ubah data serta dapat melakukan pencarian data pada *search box*.

#### 41. Rancangan Antarmuka Halaman Kategori Produk

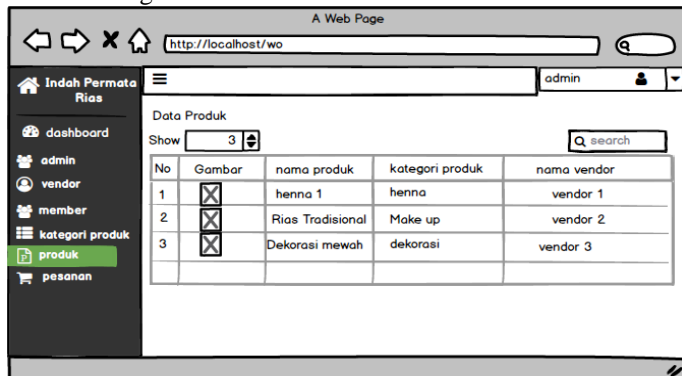




**Gambar 3. 78** Perancangan Halaman Kategori Produk Pada Admin

Rancangan antar muka halaman yang tertera pada **Gambar 3.78** menggambarkan mengenai halaman kategori produk, halaman ini digunakan oleh admin untuk mengelola data kategori produk yang meliputi tambah data, *Edit* data, dan hapus data. Pada halaman ini juga terdapat *searchbox* yang dapat digunakan untuk mencari data yang diinginkan.

#### 42. Rancangan Antarmuka Halaman Produk



**Gambar 3. 79** Perancangan Halaman Produk

Rancangan antar muka halaman yang tertera pada **Gambar 3.79** menggambarkan mengenai halaman produk, halaman ini digunakan oleh admin untuk melihat produk yang telah diupload oleh vendor. Pada halaman ini juga terdapat *searchbox* yang dapat digunakan untuk mencari data yang diinginkan.

#### 43. Rancangan Antarmuka Halaman Pesanan

The screenshot shows a web browser window with the URL 'http://localhost/wo'. The page title is 'Indah Permata Rias' and the user is logged in as 'admin'. The main content area is titled 'Data Pesanan Produk' and contains a table with the following data:

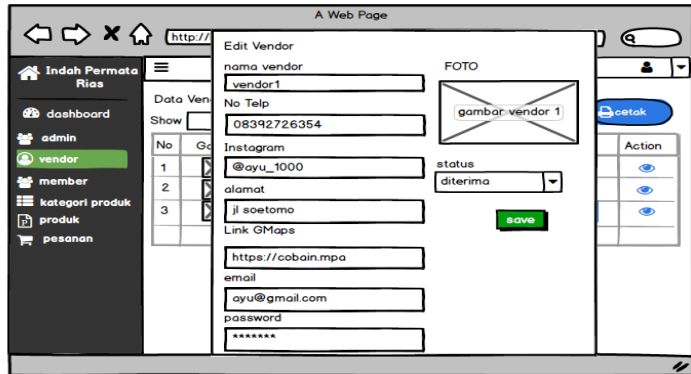
No	No Invoice	Email	Tgl Pesan	pembayaran	status	total
1	HMP-30001	mem1@gmail.com	2022/09/20	tf bank	menunggu pembayaran	Rp 100.000
2	HMP-30002	mem2@gmail.com	2022/10/20	tf bank	selesai	Rp 500.000
3	HMP-30003	mem3@gmail.com	2022/10/29	cod	poses	Rp 1500.000

Additional interface elements include a sidebar with navigation options (dashboard, admin, vendor, member, kategori produk, produk, pesanan), a search box, and a 'cetak' button.

**Gambar 3. 80** Perancangan Halaman Pesanan Pada Admin

Rancangan antar muka halaman yang tertera pada **Gambar 3.80** menggambarkan mengenai halaman pesanan yang berbentuk tabel, dengan kolom nomor, nomor *Invoice*, email, tanggal pesan, pemabayaran, status, dan total. Admin dapat melakukan cetak data laporan pesanan. Pada halaman ini juga terdapat *searchbox* yang dapat digunakan untuk mencari data yang diinginkan.

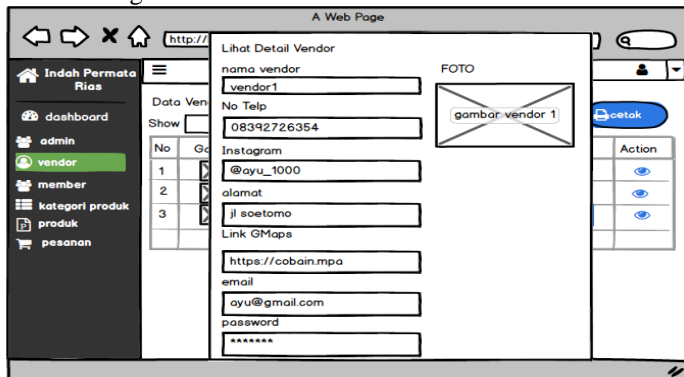
#### 44. Rancangan Antarmuka Halaman Edit Data Vendor



**Gambar 3. 81** Rancangan Antarmuka Halaman Edit Vendor

Rancangan antar muka halaman yang tertera pada **Gambar 3.81** menggambarkan mengenai halaman edit vendor yang dilakukan oleh aktor admin. Vendor dapat melakukan ubah data vendor dan ubah data status vendor. Tekan tombol *save* untuk menyimpan data ke dalam *database*.

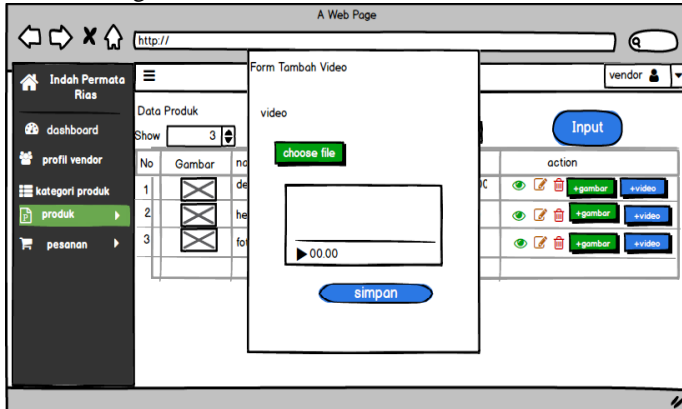
#### 45. Rancangan Antarmuka Halaman Lihat Detail Data Vendor



**Gambar 3. 82** Rancangan Antarmuka Halaman Lihat Detail Data Vendor

Rancangan antar muka halaman yang tertera pada **Gambar 3.82** menggambarkan mengenai halaman lihat data vendor yang dilakukan oleh aktor admin. Admin dapat melihat detail dari data vendor yang berupa nama vendor, nomor telepon, instagram, alamat, link *Google Maps*, *email*, *password* dan foto vendor telah melakukan registrasi pada sistem

#### 46. Rancangan Antarmuka Halaman Tambah Video



**Gambar 3. 83** Rancangan Antarmuka Halaman Lihat Detail Data Vendor

Rancangan antar muka halaman yang tertera pada **Gambar 3.82** menggambarkan halaman tambah video produk yang dilakukan oleh vendor. Vendor dapat melakukan tambah video dengan menekan tombol *choose file* dan vendor juga dapat melakukan penyimpanan data video dengan menekan tombol *simpan*.

### 3.2.5 Skenario Pengujian Sistem

Skenario pengujian sistem berisi kemungkinan-kemungkinan yang akan terjadi pada saat sedang melakukan pengujian sistem. Berikut adalah skenario pengujian pada beberapa proses:

1. Skenario Pengujian Proses *Login*

Dalam pengujian pada proses ini dilakukan oleh aktor dengan memasukan *password* dan *username* untuk dapat *login* ke dalam sistem, Jika *username* dan *password* sesuai maka aktor akan masuk ke dalam sistem sesuai dengan *role* aktor. Tetapi jika *username* dan *password* salah maka akan muncul pesan *error*. Skenario Pengujian *login* dapat dilihat pada **Tabel 3.14**

**Tabel 3. 14** Skenario Pengujian *Login*

Idefifikasi	
Nama proses	<i>Login</i>
Tujuan	Masuk ke dalam sistem.
Deskripsi	
Aktor	<ul style="list-style-type: none"> <li>- Member</li> <li>- Admin</li> <li>- Vendor</li> <li>- Pemilik</li> <li>- Keuangan</li> </ul>
Skenario Pengujian	
Kondisi Awal	
Aktor berada di halaman <i>Login</i>	
Aktor	Sistem
<ul style="list-style-type: none"> <li>- Aktor memasukkan <i>email</i> dan <i>password</i>.</li> <li>- Aktor menekan tombol <i>Login</i>.</li> </ul>	<ul style="list-style-type: none"> <li>- Sistem akan melakukan pencocokan data yang dimasukkan dengan data yang tersimpan di <i>database</i></li> </ul>
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
<ul style="list-style-type: none"> <li>- Aktor akan masuk ke halaman utama sesuai dengan <i>role id</i> masing-masing</li> </ul>	<ul style="list-style-type: none"> <li>- Aktor tetap di halaman <i>Login</i>.</li> </ul>

	- Sistem akan menampilkan pesan <i>error</i> , <i>username</i> atau <i>password</i> salah.
--	--

## 2. Skenario Pengujian Registrasi

Pengujian pada proses ini dilakukan oleh aktor member dan vendor dengan mengisi *username*,*email*,no telepon, alamat dan *password* ke dalam form yang tersedia. Skenario pengujian *login* dapat dilihat pada **Tabel 3.15**

**Tabel 3. 15** Skenario Pengujian Registrasi

Ideftifikasi	
Nama proses	Registrasi
Tujuan	Memasukan data registrasi kedalam tabel member dan vendor.
Deskripsi	
Aktor	- Member - Vendor
Skenario Pengujian	
Kondisi Awal	
Aktor berada di halaman Registrasi	
Aktor	Sistem
- Aktor data pada form registrasi	- Sistem akan menyimpan data registrasi pada <i>database</i> .
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Menampilkan registrasi berhasil	- Aktor tetap di halaman registrasi - Akan muncul pesan <i>username</i> sudah terdaftar,

	jika <i>username</i> yang dimasukkan sudah ada pada <i>database</i> .
--	---

### 3. Skenario Tambah Data Produk

Pengujian pada proses ini dilakukan oleh aktor vendor dengan memasukkan data produk ke dalam form upload data produk. Skenario Pengujian tambah data produk dapat dilihat pada **Tabel 3.16**

**Tabel 3. 16** Skenrio Pengujian Tambah Data Produk

Ideftifikasi	
Nama proses	Tambah data produk
Tujuan	Memasukan atau menambah data produk kedalam tabel produk
Deskripsi	
Aktor	Vendor
Skenario Pengujian	
Kondisi Awal	
Data produk tidak ada dalam tabel produk	
Aktor	Sistem
- Aktor memasukan data produk pada form tambah data produk	- Sistem akan menyimpan data produk pada tabel produk
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Menampilkan pesan data berhasil ditambah	- Menampilkan pesan data gagal ditambahkan - Sistem tidak menyimpan data

- data akan disimpan pada tabel produk	
--	--

#### 4. Skenario Pengujian Ubah Data Produk

Pengujian pada proses ini dilakukan oleh aktor vendor dengan memasukan data produk yaang baru ke dalam form ubah data produk.Skenario Pengujian ubah data produk dapat dilihat pada **Tabel 3.17**

**Tabel 3. 17** Skenario Pengujian Ubah Data Produk

Ideftifikasi	
Nama proses	Ubah data produk
Tujuan	Mengubah data produk kedalam tabel produk
Deskripsi	
Aktor	Vendor
Skenario Pengujian	
Kondisi Awal	
Data produk ada dalam tabel produk	
Aktor	Sistem
<ul style="list-style-type: none"> <li>- Aktor memilih data produk yang akan di<i>Edit</i></li> <li>- Aktor memasukan data baru pada form ubah data produk</li> </ul>	<ul style="list-style-type: none"> <li>- Sistem akan menampilkan data produk yang dipilih</li> <li>- Sistem akan menyimpan dan merubah data produk baru pada tabel produk</li> </ul>
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
<ul style="list-style-type: none"> <li>- Menampilkan pesan data berhasil diubah</li> </ul>	<ul style="list-style-type: none"> <li>- Menampilkan pesan data gagal diubah</li> </ul>



- data akan disimpan pada tabel produk	- Sistem tidak menyimpan data
--	-------------------------------

#### 5. Skenario Pengujian Hapus Data Produk

Pengujian pada proses ini dilakukan oleh aktor vendor untuk menghapus data produk yang tersimpan pada tabel produk. Skenario Pengujian hapus data produk dapat dilihat pada **Tabel 3.18**

**Tabel 3. 18** Skenario Pengujian Hapus Data Produk

Identifikasi	
Nama proses	Hapus data produk
Tujuan	Menghapus data produk kedalam tabel produk
Deskripsi	
Aktor	Vendor
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
<ul style="list-style-type: none"> <li>- Aktor memilih data produk yang akan di<i>Edit</i></li> <li>- Aktor menekan tombol hapus</li> </ul>	<ul style="list-style-type: none"> <li>- Sistem akan menghapus data produk yang dipilih dari tabel produk</li> </ul>
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
<ul style="list-style-type: none"> <li>- Menampilkan pesan data berhasil dihapus</li> <li>- Menampilkan data yang berurung.</li> </ul>	<ul style="list-style-type: none"> <li>- Mampilkan pesan data gagal diubah</li> <li>- Sistem tidak menghapus data yang dipilih</li> </ul>

#### 6. Skenario Pengujian Lihat Data Produk

Pengujian pada proses ini dilakukan oleh aktor vendor untuk melihat data produk yang tersimpan pada tabel produk. Skenario Pengujian lihat data produk dapat dilihat pada **Tabel 3.19**

**Tabel 3. 19** Skenario Pengujian Lihat Data Produk

Identifikasi	
Nama proses	Lihat data produk
Tujuan	Melihat data produk pada tabel produk
Deskripsi	
Aktor	Vendor
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
- Aktor memilih data produk	- Sistem menampilkan data produk yang tersimpan
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan data produk	- Sistem tidak menampilkan data produk

#### 7. Skenario Pengujian Tambah Data Member

Pengujian pada proses ini dilakukan oleh aktor admin untuk menambah data member yang tersimpan pada tabel member. Skenario Pengujian tambah data member dapat dilihat pada **Tabel 3.20**

**Tabel 3. 20** Skenario Pengujian Tambah Data Member

Idefifikasi	
Nama proses	Tambah data member
Tujuan	Menambah data member pada tabel member
Deskripsi	
Aktor	Admin
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
- Aktor admin memasukan data member pada form tambah data member	- Sistem menyimpan data yang diinputkan pada tabel member
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan pesan data berhasil ditambah - Data disimpan pada tabel member	- Sistem menampilkan pesan data gagal ditambahkan - Sistem tidak menyimpan data member

#### 8. Skenario Pengujian Lihat Data Member

Pengujian pada proses ini dilakukan oleh aktor admin untuk melihat data member yang tersimpan pada tabel member. Skenario Pengujian lihat data member dapat dilihat pada **Tabel 3.21**

**Tabel 3. 21** Skenario Pengujian Lihat Data Member

Ideftifikasi	
Nama proses	Lihat data member
Tujuan	Melihat data member pada tabel member
Deskripsi	
Aktor	Admin
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
- Aktor memilih data member	- Sistem menampilkan data member yang tersimpan
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan data member	- Sistem tidak menampilkan data member

#### 9. Skenario Pengujian Tambah Vendor

Pengujian pada proses ini dilakukan oleh aktor admin untuk menambah data vendor yang tersimpan pada tabel vendor.Skenario Pengujian tambah data vendor dapat dilihat pada **Tabel 3.22**

**Tabel 3. 22** Skenario Pengujian Tambah Vendor

Ideftifikasi	
Nama proses	Tambah data vendor

Tujuan	Menambah data vendor pada tabel vendor
Deskripsi	
Aktor	Admin
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
Aktor admin memasukkan data vendor pada form tambah data vendor	- Sistem menyimpan data yang diinputkan pada tabel vendor
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan pesan data berhasil ditambah - Data disimpan pada tabel vendor	- Sistem menampilkan pesan data gagal ditambahkan - Sistem tidak menyimpan data vendor

#### 10. Skenario Pengujian Lihat Data Vendor

Pengujian pada proses ini dilakukan oleh aktor admin untuk melihat data vendor yang tersimpan pada tabel vendor. Skenario Pengujian lihat data vendor dapat dilihat pada

#### **Tabel 3.23**

**Tabel 3. 23** Skenario Pengujian Lihat Data Vendor

Ideftifikasi	
Nama proses	Lihat data vendor
Tujuan	Melihat data vendor pada tabel vendor

Deskripsi	
Aktor	Admin
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
- Aktor memilih data vendor	- Sistem menampilkan data vendor yang tersimpan
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan data vendor	- Sistem tidak menampilkan data vendor

11. Skenario Pengujian verifikasi data vendor

Pengujian pada proses ini dilakukan oleh aktor admin untuk memverifikasi data vendor yang terdaftar dan data akan tersimpan pada tabel vendor. Skenario Pengujian verifikasi data vendor dapat dilihat pada **Tabel 3.24**

**Tabel 3. 24** Skenario Pengujian verifikasi data vendor

Identifikasi	
Nama proses	Verifikasi data vendor
Tujuan	Memverifikasi data vendor yang mendaftar
Deskripsi	
Aktor	Admin
Skenario Pengujian	
Kondisi Awal	

Status pada data vendor yang mendaftar belum terverifikasi	
Aktor	Sistem
- Aktor memilih data yang belum terverifikasi	- Sistem melakukan verifikasi dan mengirimkan data ke dalam tabel vendor
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan pesan data berhasil diverifikasi - Tampilan status pada data berubah menjadi data terverifikasi	- Sistem menampilkan pesan data gagal diverifikasi - Data tidak disimpan pada tabel vendor

## 12. Skenario Pengujian Melihat Data Produk

Pengujian pada proses ini dilakukan oleh aktor member untuk melihat data produk yang tersimpan pada tabel produk. Skenario Pengujian lihat data produk dapat dilihat pada **Tabel 3.25**

**Tabel 3. 25** Skenario Pengujian Melihat Data Produk

Ideftifikasi	
Nama proses	Lihat data produk
Tujuan	Melihat data produk pada tabel produk
Deskripsi	
Aktor	Member
Skenario Pengujian	

Kondisi Awal	
Aktor	Sistem
- Aktor memilih data produk	- Sistem menampilkan data produk yang tersimpan
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan data produk	- Sistem tidak menampilkan data produk

### 13. Skenario Pengujian Membuat Pesanan

Pengujian pada proses ini dilakukan oleh aktor member/member untuk membuat atau menambah pesanan dengan menambah produk yang akan dipesan, lalu member mengisikan data pada form pemesanan, selanjutnya data akan disimpan pada tabel order. Skenario Pengujian membuat pesanan dapat dilihat pada **Tabel 3.26**

**Tabel 3. 26** Skenario Pengujian Membuat Pesanan

Ideftifikasi	
Nama proses	Membuat pesanan
Tujuan	Menambah pesanan
Deskripsi	
Aktor	Member/member
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
- Aktor memilih produk	- Sistem akan menyimpan data pesanan pada tabel order



- Aktor menekan tombol pesan - Aktor mengisi form pesanan	
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Data pesanan akan tampil pada halaman lihat pesanan	- Sistem menampilkan pesan produk gagal dipesan - Sistem tidak menyimpan data pesanan pada tabel order

#### 14. Skenario Pengujian Lihat Status Pesanan

Pengujian pada proses ini dilakukan oleh aktor member/member untuk melihat status pesanan sudah dikonfirmasi atau belum. Proses ini dimulai dengan aktor meenekan tombol menu pesanan yang ada pada sistem, jika berhasil, maka data pesanan akan muncul disertai status pesanan. Skenario pengujian lihat status pesanan dapat dilihat pada **Tabel 3.27**

**Tabel 3. 27** Skenario Pengujian Lihat Status Pesanan

Ideftifikasi	
Nama proses	Lihat Status Pesanan
Tujuan	Lihat Status Pesanan
Deskripsi	
Aktor	Member/member
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem

- Aktor menekan tombol menu pesanan	- Sistem akan menampilkan data pesanan
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Data pesanan akan tampil disertai status pesanan baik menunggu pembayaran, pesanan diproses, pesanan ditolak atau status pesanan selesai	- Sistem tidak menampilkan data pesanan yang ada pada tabel order

#### 15. Skenario Pengujian Cetak *Invoice*

Pengujian pada proses ini dilakukan oleh aktor member/member untuk mencetak *Invoice* tau riwayat pesanan yang ada pada tabel *Invoice* jika berhasil, maka data pesanan akan muncul dan member dapat menekan tombol print untuk mencetaknya, jika gagal maka sistem tidak dapat menampilkan data *Invoice* yang ada pada tabel *Invoice*. Skenario pengujian cetak *Invoice* dapat dilihat pada **Tabel 3.28**

**Tabel 3. 28** Skenario Pengujian Cetak *Invoice*

Identifikasi	
Nama proses	Mencetak <i>Invoice</i>
Tujuan	Mencetak <i>Invoice</i>
Deskripsi	
Aktor	Member/member
Skenario Pengujian	
Kondisi Awal	

Aktor	Sistem
- Aktor menekan tombol cetak <i>Invoice</i>	- Sistem akan menampilkan data yang ada pada tabel <i>Invoice</i> - Sistem mencetak <i>Invoice</i>
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Data <i>Invoice</i> berhasil dicetak	- Sistem tidak dapat mencetak data <i>Invoice</i> .

#### 16. Skenario Pengujian Lihat Data pesanan

Pengujian pada proses ini dilakukan oleh aktor admin untuk melihat data pesanan yang masuk, dengan menekan tombol menu pesanan. Jika berhasil, maka sistem akan menampilkan data pesanan yang ada pada tabel order, Jika Gagal maka sistem tidak menampilkan data pesanan. Skenario pengujian lihat data pesanan yang dilakukan oleh admin dapat dilihat pada **Tabel 3.29**

**Tabel 3. 29** Skenario Pengujian Lihat Data pesanan

Ideftifikasi	
Nama proses	Lihat data pesanan
Tujuan	Lihat data pesanan
Deskripsi	
Aktor	Admin Keuangan Pemilik Vendor
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem

- Aktor menekan tombol menu pesanan	- Sistem akan menampilkan data pesanan
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem menampilkan data pesanan yang masuk	- Sistem tidak menampilkan data pesanan yang ada pada tabel order

### 17. Skenario Pengujian Ubah Status Pesanan

Pengujian pada proses ini dilakukan oleh aktor keuangan untuk mengubah status pada pesanan yang masuk menjadi pesanan diproses, atau pesanan selesai. Skenario ubah sttus pesanan yang dilakukan oleh keuangan dapat dilihat pada **Tabel 3.30**

**Tabel 3. 30** Skenario Pengujian Ubah Status Pesanan

Idefifikasi	
Nama proses	Mengubah status pesanan
Tujuan	Mengubah status pesanan
Deskripsi	
Aktor	Keuangan
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
- Aktor admin memilih menu pesanan - Aktor admin memilih data pesanan yang akan	- Sistem akan menampilkan data pesanan - Sistem akan menampilkan data pesanan yang dipilih dengan sttus yang telah berubah

diubah statusnya dengan menekan tombol ubah status	
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem menampilkan data pesanan dengan status yang telah berubah menjadi pesanan di proses atau pesanan selesai	- Sistem tidak menampilkan status pada data yang dipilih tidak berubah

#### 18. Skenario Pengujian Pengiriman Notifikasi melalui *WhatsApp*

Pengujian pada proses ini dilakukan oleh aktor keuangan untuk mengirimkan notifikasi melalui *WhatsApp* kepada member. Proses ini dimulai dengan aktor melihat data pesanan, kemudian jika ingin mengirimkan notifikasi pesanan sesuai dengan status pesanan maka aktor harus menekan tombol *WhatsApp*, yang kemudian sistem akan mengirimkan notifikasi sesuai dengan status pesanan. Skenario pengiriman notifikasi melalui *WhatsApp* yang dilakukan oleh keuangan dapat dilihat pada **Tabel 3.31**

**Tabel 3. 31** Skenario Pengujian Kirim Notifikasi Pesanan Melalui *WhatsApp*

Identifikasi	
Nama proses	Mengirim notifikasi melalui <i>WhatsApp</i>
Tujuan	Mengirim notifikasi melalui <i>WhatsApp</i>
Deskripsi	

Aktor	Keuangan
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
<ul style="list-style-type: none"> <li>- Aktor admin menekan menu pesanan</li> <li>- Aktor admin memilih data pesanan yang akan dikirimkan notifikasi melalui <i>WhatsApp</i></li> </ul>	<ul style="list-style-type: none"> <li>- Sistem akan menampilkan data pesanan</li> <li>- Sistem akan melakukan pengiriman notifikasi kepada member melalui <i>WhatsApp</i></li> </ul>
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
<ul style="list-style-type: none"> <li>- Pesan notifikasi yang dikirim akan masuk ke member melalui <i>WhatsApp</i></li> </ul>	<ul style="list-style-type: none"> <li>- <i>WhatsApp</i> tidak dapat mengirim pesan notifikasi kepada member</li> </ul>

#### 19. Skenario Pengujian Cetak Laporan Transaksi

Pengujian pada proses ini dilakukan oleh aktor admin untuk mencetak *Invoice* tau riwayat pesanan yang ada pada tabel *Invoice* jika berhasil, maka data pesanan akan muncul dan admin dapat menekan tombol print untuk mencetaknya, jika gagal maka sistem tidak dapat mencetak data . Skenario pengujian cetak Laporan Transaksi dapat dilihat pada **Tabel 3.32**

**Tabel 3. 32** Skenario Pengujian Cetak Laporan Transaksi

Ideftifikasi	
Nama proses	Mencetak <i>Invoice</i>
Tujuan	Mencetak <i>Invoice</i>

Deskripsi	
Aktor	Admin Keuangan Pemilik vendor
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem
- Aktor menekan tombol cetak <i>Invoice</i>	- Sistem akan menampilkan data yang ada pada tabel <i>Invoice</i> - Sistem mencetak <i>Invoice</i>
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Data <i>Invoice</i> berhasil dicetak	- Sistem tidak dapat mencetak data <i>Invoice</i> .

#### 20. Skenario Pengujian Lihat Data Member

Pengujian pada proses ini dilakukan oleh aktor pemilik untuk melihat data member yang tersimpan pada tabel member. Skenario Pengujian lihat data member dapat dilihat pada **Tabel 3.33**

**Tabel 3. 33** Skenario Pengujian Lihat Data Member

Ideftifikasi	
Nama proses	Lihat data member
Tujuan	Melihat data member pada tabel member
Deskripsi	
Aktor	Pemilik
Skenario Pengujian	

Kondisi Awal	
Aktor	Sistem
- Aktor memilih data member	- Sistem menampilkan data member yang tersimpan
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan data member	- Sistem tidak menampilkan data member

#### 21. Skenario Tambah Data Kategori Produk

Pengujian pada proses ini dilakukan oleh aktor admin dengan memasukan data kategori produk ke dalam form upload data kategori produk. Skenario Pengujian tambah data kategori produk dapat dilihat pada **Tabel 3.34**

**Tabel 3. 34** Skenario Pengujian Tambah Data Kategori Produk

Identifikasi	
Nama proses	Tambah data kategori produk
Tujuan	Memasukan atau menambah data kategori produk kedalam tabel kategori produk
Deskripsi	
Aktor	Admin
Skenario Pengujian	
Kondisi Awal	
Data produk tidak ada dalam tabel produk	
Aktor	Sistem



- Aktor memasukan data kategori produk pada form tambah data produk	- Sistem akan menyimpan data kategori produk pada tabel kategori produk
<b>Kondisi Akhir</b>	
<b>Kondisi Berhasil</b>	<b>Kondisi Gagal</b>
- Menampilkan pesan data berhasil ditambah - data akan disimpan pada tabel kategori produk	- Menampilkan pesan data gagal ditambahkan - Sistem tidak menyimpan data

## 22. Skenario Pengujian Ubah Data kategori Produk

Pengujian pada proses ini dilakukan oleh aktor admin dengan memasukan data kategori produk yang baru ke dalam form ubah data kategori produk. Skenario Pengujian ubah data kategori produk dapat dilihat pada **Tabel 3.17**

**Tabel 3. 35** Skenario Pengujian Ubah Data Produk

<b>Ideftifikasi</b>	
Nama proses	Ubah data produk
Tujuan	Mengubah data kategori produk kedalam tabel produk
<b>Deskripsi</b>	
Aktor	Admin
<b>Skenario Pengujian</b>	
<b>Kondisi Awal</b>	
Data produk ada dalam tabel produk	
Aktor	Sistem

<ul style="list-style-type: none"> <li>- Aktor memilih data kategori produk yang akan di <i>Edit</i></li> <li>- Aktor memasukan data baru pada form ubah data kategori produk</li> </ul>	<ul style="list-style-type: none"> <li>- Sistem akan menampilkan data produk yang dipilih</li> <li>- Sistem akan menyimpan dan merubah data produk baru pada tabel produk</li> </ul>
Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
<ul style="list-style-type: none"> <li>- Menampilkan pesan data berhasil diubah</li> <li>- data akan disimpan pada tabel kategori produk</li> </ul>	<ul style="list-style-type: none"> <li>- Menampilkan pesan data kategori gagal diubah</li> <li>- Sistem tidak menyimpan data</li> </ul>

### 23. Skenario Pengujian Hapus Data Produk

Pengujian pada proses ini dilakukan oleh aktor admin untuk menghapus data kategori produk yang tersimpan pada tabel kategori produk. Skenario Pengujian hapus kategori data produk dapat dilihat pada **Tabel 3.36**

**Tabel 3. 36** Skenario Pengujian Hapus Data Produk

Ideftifikasi	
Nama proses	Hapus data kategori produk
Tujuan	Menghapus data kategori produk kedalam tabel kategori produk
Deskripsi	
Aktor	Admin
Skenario Pengujian	
Kondisi Awal	
Aktor	Sistem

<ul style="list-style-type: none"> <li>- Aktor memilih data kategori produk yang akan di<i>Edit</i></li> <li>- Aktor menekan tombol hapus</li> </ul>	<ul style="list-style-type: none"> <li>- Sistem akan menghapus data kategori produk yang dipilih dari tabel produk</li> </ul>
<b>Kondisi Akhir</b>	
Kondisi Berhasil	Kondisi Gagal
<ul style="list-style-type: none"> <li>- Menampilkan pesan data kategori berhasil dihapus</li> <li>- Menampilkan data yang berurung.</li> </ul>	<ul style="list-style-type: none"> <li>- Mampilkan pesan data gagal diubah</li> <li>- Sistem tidakmenghapus data yang dipilih</li> </ul>

#### 24. Skenario Pengujian Lihat Data Kategori Produk

Pengujian pada proses ini dilakukan oleh aktor admin untuk melihat data kategori produk yang tersimpan pada tabel produk. Skenario Pengujian lihat data kategori produk dapat dilihat pada **Tabel 3.37**

**Tabel 3. 37** Skenario Pengujian Lihat Data Produk

<b>Ideftifikasi</b>	
Nama proses	Lihat data kategori produk
Tujuan	Melihat data kategori produk pada tabel produk
<b>Deskripsi</b>	
Aktor	admin
<b>Skenario Pengujian</b>	
<b>Kondisi Awal</b>	
Aktor	Sistem
<ul style="list-style-type: none"> <li>- Aktor memilih data kategori produk</li> </ul>	<ul style="list-style-type: none"> <li>- Sistem menampilkan data kategori produk yang tersimpan</li> </ul>

Kondisi Akhir	
Kondisi Berhasil	Kondisi Gagal
- Sistem Menampilkan data kategori produk	- Sistem tidak menampilkan data kategori produk