

**APLIKASI BERBASIS WEB BAKERY  
UNTUK PEMASARAN PRODUK UMKM  
(SOFTWARE REQUIREMENT SYSTEM)**



**Disusun Oleh :**

Moch. Rama Maulana	:	1203232085
Anjay	:	12032
Miaw	:	12032

**PROGRAM STUDI S1 INFORMATIKA  
FAKULTAS INFORMATIKA  
UNIVERSITAS TELKOM SURABAYA  
2025**

## **BAB I**

### **PENDAHULUAN**

#### **1. Tujuan**

Dokumen ini merupakan dokumen perancangan untuk aplikasi berbasis web Toko Roti UMKM yang digunakan sebagai media promosi dan penjualan produk roti secara daring. Tujuan dari penulisan Software Design Document (SDD) ini adalah untuk memberikan penjelasan mengenai hasil proses perancangan sistem, termasuk penyempurnaan hasil rancangan tersebut untuk merepresentasikan aplikasi Toko Roti UMKM, baik berupa gambaran umum maupun penjelasan detail dan menyeluruh.

Software Design Document ini akan digunakan sebagai acuan informasi bagi tim Software Engineering, baik itu perancang perangkat lunak maupun pengembang yang akan menggunakan dokumen ini sebagai pedoman dalam mengimplementasikan sistem berdasarkan hasil rancangan yang telah dibuat. Selain itu, dokumen ini juga akan menjadi bahan analisis dalam proses pengembangan dan pemeliharaan sistem di masa mendatang. Dengan adanya dokumen ini diharapkan proses perancangan dapat berjalan lebih terarah dan fokus, sehingga mampu meminimalkan kesalahan serta memastikan bahwa hasil implementasi sesuai dengan tujuan dan kebutuhan pengguna.

#### **2. Ruang Lingkup**

Software Design Document aplikasi Toko Roti UMKM ini menjabarkan rancangan aplikasi penjualan roti secara daring dengan fitur utama guest checkout, yang memungkinkan pelanggan melakukan pembelian tanpa perlu membuat akun. Aplikasi ini akan dirancang dengan memperhatikan aspek arsitektural, prosedural, dan antarmuka menggunakan bahasa pemrograman PHP dengan framework Laravel, alat bantu perancangan Figma, serta basis data MySQL. Lingkungan pengembangan menggunakan Laragon sebagai server lokal, dengan perangkat keras berupa personal computer, keyboard, dan mouse.

## **BAB II**

### **RANCANGAN LINGKUNGAN IMPLEMENTASI**

#### **1. Lingkungan Perangkat Keras**

Perangkat keras yang digunakan pada saat pengimplementasian aplikasi Bakery adalah sebagai berikut :

- a. *Personal Computer*, dengan spesifikasi :
  - Mikroprosesor : Intel i5 12th Gen
  - Memori : 16 GB
  - Monitor : LCD 15 inch
  - Media Penyimpanan : Harddisk 500 GB
  - *Keyboard*
  - *Mouse*

#### **2. Lingkungan Perangkat Lunak**

Perangkat lunak yang digunakan pada saat pengimplementasian aplikasi Bakery adalah sebagai berikut :

- a. Sistem Operasi : Microsoft Windows 11
- b. Bahasa Pemrograman : PHP, HTML, dan Javascript.
- c. Perangkat tambahan :
  - Laragon sebagai software yang mengemas MySQL, PHP, dan Apache. Sehingga nantinya aplikasi yang dibuat dapat dengan mudah terhubung ke database melalui localhost
  - Web browser seperti Google Chrome atau Mozilla Firefox yang digunakan untuk melihat tampilan aplikasi dalam halaman web.

### 3. Fitur – Fitur Aplikasi

Dalam website Bakery terdapat 2 Role nantinya yaitu Admin dan Customer

#### **Fitur web (Customer) :**

- Tambah produk ke cart
- Ubah jumlah produk pada cart
- Hapus produk dari cart
- Bersihkan cart
- Urutkan produk berdasarkan nama dan harga
- Filter produk berdasarkan kategori
- Cari produk berdasarkan nama

#### **Admin :**

- Manage produk (CRUD)
- Manage kategori (CRUD)
- Manage pesanan (CRUD)
- Manage single page (CRUD)

## BAB III

### PERANCANGAN DATA

#### 1. Daftar Tabel

##### a. Tabel Orders

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<code>id</code> 📄	bigint		UNSIGNED	No	None		AUTO_INCREMENT	Change  Drop  More
2	<code>customer_name</code>	varchar(255)	utf8mb4_unicode_ci		No	None			Change  Drop  More
3	<code>customer_postcode</code>	varchar(255)	utf8mb4_unicode_ci		No	None			Change  Drop  More
4	<code>customer_city</code>	varchar(255)	utf8mb4_unicode_ci		No	None			Change  Drop  More
5	<code>customer_address</code>	longtext	utf8mb4_unicode_ci		No	None			Change  Drop  More
6	<code>customer_whatsapp</code>	varchar(255)	utf8mb4_unicode_ci		No	None			Change  Drop  More
7	<code>receipt</code>	varchar(255)	utf8mb4_unicode_ci		No	None			Change  Drop  More
8	<code>shipping_receipt</code>	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change  Drop  More
9	<code>status</code>	enum('Pending', 'Paid', 'Processing', 'Shipped', '...')	utf8mb4_unicode_ci		No	None			Change  Drop  More
10	<code>total</code>	int			No	None			Change  Drop  More
11	<code>paid_date</code>	timestamp			Yes	NULL			Change  Drop  More
12	<code>xendit_url</code>	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change  Drop  More
13	<code>xendit_id</code>	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change  Drop  More
14	<code>created_at</code>	timestamp			Yes	NULL			Change  Drop  More
15	<code>updated_at</code>	timestamp			Yes	NULL			Change  Drop  More

##### b. Tabel Products

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<code>id</code> 📄	bigint		UNSIGNED	No	None		AUTO_INCREMENT	Change  Drop  More
2	<code>title</code>	varchar(255)	utf8mb4_unicode_ci		No	None			Change  Drop  More
3	<code>image</code>	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change  Drop  More
4	<code>body</code>	longtext	utf8mb4_unicode_ci		Yes	NULL			Change  Drop  More
5	<code>price</code>	int			No	None			Change  Drop  More
6	<code>stock</code>	int			No	None			Change  Drop  More
7	<code>created_at</code>	timestamp			Yes	NULL			Change  Drop  More
8	<code>updated_at</code>	timestamp			Yes	NULL			Change  Drop  More

##### c. Tabel Categories

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<code>id</code> 📄	bigint		UNSIGNED	No	None		AUTO_INCREMENT	Change  Drop  More
2	<code>title</code>	varchar(255)	utf8mb4_unicode_ci		No	None			Change  Drop  More
3	<code>image</code>	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change  Drop  More
4	<code>created_at</code>	timestamp			Yes	NULL			Change  Drop  More
5	<code>updated_at</code>	timestamp			Yes	NULL			Change  Drop  More

#### d. Tabel Product Categories

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<code>id</code> 	bigint		UNSIGNED	No	<i>None</i>		AUTO_INCREMENT	 Change  Drop More
<input type="checkbox"/>	2	<code>product_id</code> 	bigint		UNSIGNED	No	<i>None</i>			 Change  Drop More
<input type="checkbox"/>	3	<code>category_id</code> 	bigint		UNSIGNED	No	<i>None</i>			 Change  Drop More
<input type="checkbox"/>	4	<code>created_at</code>	timestamp			Yes	<i>NULL</i>			 Change  Drop More
<input type="checkbox"/>	5	<code>updated_at</code>	timestamp			Yes	<i>NULL</i>			 Change  Drop More

#### e. Tabel Order Products

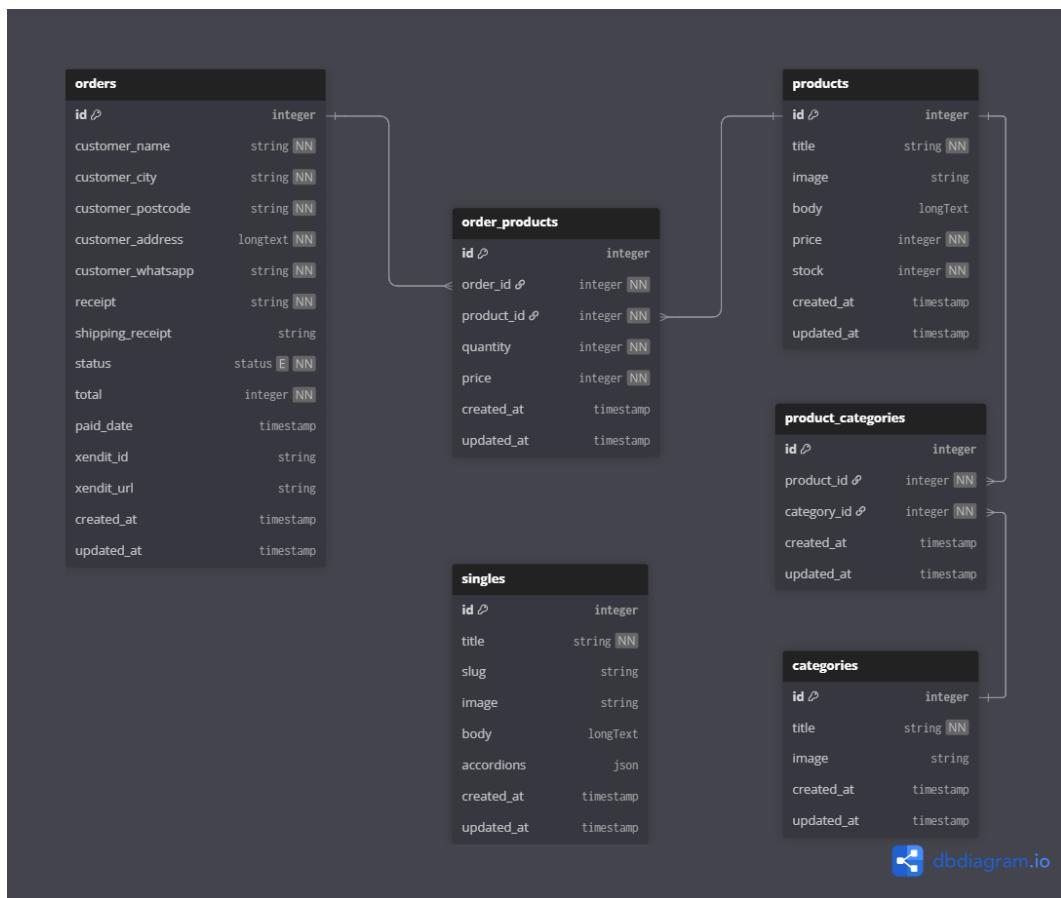
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<code>id</code> 	bigint		UNSIGNED	No	<i>None</i>		AUTO_INCREMENT	 Change  Drop More
<input type="checkbox"/>	2	<code>order_id</code> 	bigint		UNSIGNED	No	<i>None</i>			 Change  Drop More
<input type="checkbox"/>	3	<code>product_id</code> 	bigint		UNSIGNED	No	<i>None</i>			 Change  Drop More
<input type="checkbox"/>	4	<code>quantity</code>	int			No	1			 Change  Drop More
<input type="checkbox"/>	5	<code>price</code>	int			No	<i>None</i>			 Change  Drop More
<input type="checkbox"/>	6	<code>created_at</code>	timestamp			Yes	<i>NULL</i>			 Change  Drop More
<input type="checkbox"/>	7	<code>updated_at</code>	timestamp			Yes	<i>NULL</i>			 Change  Drop More

#### f. Tabel Singles

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<code>id</code> 	bigint		UNSIGNED	No	<i>None</i>		AUTO_INCREMENT	 Change  Drop More
<input type="checkbox"/>	2	<code>title</code>	varchar(255)	utf8mb4_unicode_ci		No	<i>None</i>			 Change  Drop More
<input type="checkbox"/>	3	<code>slug</code> 	varchar(255)	utf8mb4_unicode_ci		No	<i>None</i>			 Change  Drop More
<input type="checkbox"/>	4	<code>image</code>	varchar(255)	utf8mb4_unicode_ci		Yes	<i>NULL</i>			 Change  Drop More
<input type="checkbox"/>	5	<code>body</code>	longtext	utf8mb4_unicode_ci		Yes	<i>NULL</i>			 Change  Drop More
<input type="checkbox"/>	6	<code>accordions</code>	json			Yes	<i>NULL</i>			 Change  Drop More
<input type="checkbox"/>	7	<code>created_at</code>	timestamp			Yes	<i>NULL</i>			 Change  Drop More
<input type="checkbox"/>	8	<code>updated_at</code>	timestamp			Yes	<i>NULL</i>			 Change  Drop More

## BAB IV

### DIAGRAM RELASI DATABASE



## BAB V

### USECASE SCENARO

#### 1. Login (Admin)

<b>Objective</b>	Actor masuk ke sistem
<b>Actors</b>	Admin
<b>Pre-Condition</b>	Actor berada di halaman admin login
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor memasukkan email dan password. Selanjutnya menekan tombol "masuk."</li> <li>2. Sistem mengautentifikasi data akun yang dimasukkan.</li> <li>3. Jika akun berhasil diautentifikasi, sistem akan mengarahkan actor ke admin panel.</li> </ol>
<b>Alternative Flow</b>	<ol style="list-style-type: none"> <li>1. Jika akun gagal diautentifikasi sistem, maka akan muncul peringatan dan meminta actor untuk memasukkan email dan password yang benar.</li> <li>2. Actor memasukkan email dan password yang benar.</li> <li>3. Selanjutnya, actor menekan tombol "masuk."</li> <li>4. Sistem mengautentifikasi data akun yang dimasukkan.</li> <li>5. Jika akun berhasil diautentifikasi, sistem akan mengarahkan actor ke admin panel.</li> </ol>
<b>Post-Condition</b>	Actor masuk ke admin panel sistem.

#### 2. Tambah produk ke cart

<b>Objective</b>	Memungkinkan actor untuk menambah produk ke cart
<b>Actors</b>	Customer
<b>Pre-Condition</b>	Produk yang ingin ditambahkan harus tersedia di sistem
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor menjelajahi product di katalog.</li> <li>2. Actor menemukan product yang diinginkan.</li> </ol>

	<ul style="list-style-type: none"> <li>3. Actor menekan tombol “ADD TO CART” pada halaman katalog.</li> <li>4. Sistem menambahkan product ke daftar cart actor dan memunculkan notifikasi</li> <li>5. Actor dapat mengakses cart mereka melalui menu navbar.</li> </ul>
<b>Alternative Flow</b>	Jika actor mencoba menambahkan product yang sudah ada di cart, sistem menambahkan jumlah produk di cart.
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>1. Product berhasil ditambahkan ke cart actor.</li> <li>2. Actor dapat melihat product di cart melalui navbar.</li> </ul>

### 3. Ubah jumlah produk di cart

<b>Objective</b>	Memungkinkan actor untuk mengubah jumlah produk di cart
<b>Actors</b>	Customer
<b>Pre-Condition</b>	Actor sudah menambahkan produk ke cart
<b>Main Flow</b>	<ul style="list-style-type: none"> <li>1. Actor membuka daftar cart melalui navbar</li> <li>2. Actor mengubah form angka pada produk yang ingin diubah</li> <li>3. Sistem menampilkan notifikasi dan mengubah total harga di cart</li> </ul>
<b>Alternative Flow</b>	Jika jumlah yang dimasukkan kurang dari 0, jumlah otomatis menjadi 1
<b>Post-Condition</b>	Jumlah produk dalam cart berubah

### 4. Hapus produk dari cart

<b>Objective</b>	Memungkinkan actor untuk mencari menghapus produk dari cart
<b>Actors</b>	Customer
<b>Pre-Condition</b>	Actor sudah menambahkan produk ke cart
<b>Main Flow</b>	<ul style="list-style-type: none"> <li>1. Actor membuka daftar cart melalui navbar</li> </ul>

	<ol style="list-style-type: none"> <li>2. Actor menekan ikon “” pada salah satu produk di dalam cart</li> <li>3. Sistem menghapus produk dari cart, menampilkan notifikasi, dan mengubah total harga di cart</li> </ol>
<b>Alternative Flow</b>	-
<b>Post-Condition</b>	Produk yang dipilih dihapus dari cart

#### 5. Bersihkan cart

<b>Objective</b>	Memungkinkan actor untuk mencari menghapus produk dari cart
<b>Actors</b>	Customer
<b>Pre-Condition</b>	Actor sudah menambahkan produk ke cart
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor membuka daftar cart melalui navbar</li> <li>2. Actor menekan tulisan “ Hapus Semua” pada salah satu produk di dalam cart</li> <li>3. Sistem menghapus semua produk di cart, menampilkan notifikasi, dan mengubah total harga di cart jadi 0</li> </ol>
<b>Alternative Flow</b>	-
<b>Post-Condition</b>	Tidak ada produk di dalam cart

#### 6. Urutkan Produk Berdasarkan Nama dan Harga

<b>Objective</b>	Memungkinkan actor untuk mengurutkan daftar produk di katalog
<b>Actors</b>	Customer
<b>Pre-Condition</b>	Actor berada di halaman katalog
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor menekan tombol sort di kanan atas daftar produk</li> <li>2. Actor memilih jenis urutan yang dipilih</li> <li>3. Sistem menampilkan daftar produk berdasarkan urutan yang dipilih</li> </ol>

<b>Alternative Flow</b>	Jika id sort tidak ditemukan, sistem akan menampilkan daftar produk dengan urutan A-Z
<b>Post-Condition</b>	Urutan daftar produk berubah sesuai sort yang dipilih

#### 7. Filter Produk Berdasarkan Kategori

<b>Objective</b>	Memungkinkan actor untuk memfilter produk berdasarkan kategori
<b>Actors</b>	Customer
<b>Pre-Condition</b>	Actor berada di halaman katalog
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor menekan kategori yang diinginkan di samping daftar produk</li> <li>2. Sistem akan menampilkan produk yang terkait dengan kategori yang dipilih</li> </ol>
<b>Alternative Flow</b>	Jika id kategori tidak ditemukan, daftar produk yang ditampilkan tidak ada
<b>Post-Condition</b>	Daftar produk hanya menampilkan sesuai dengan kategori yang dipilih

#### 8. Cari Produk Berdasarkan Nama

<b>Objective</b>	Memungkinkan actor untuk mencari produk berdasarkan nama
<b>Actors</b>	Customer
<b>Pre-Condition</b>	Actor berada di halaman katalog
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor mengisi form "Cari produk" di atas daftar produk lalu menekan enter</li> <li>2. Sistem akan menampilkan produk dengan nama yang terkait dengan keyword</li> </ol>
<b>Alternative Flow</b>	Jika produk tidak ditemukan, daftar produk yang ditampilkan tidak ada
<b>Post-Condition</b>	Daftar produk hanya menampilkan sesuai dengan keyword yang dimasukkan

## 9. Manage produk (CRUD)

<b>Objective</b>	Memungkinkan actor untuk manage produk
<b>Actors</b>	Admin
<b>Pre-Condition</b>	Actor berada di halaman admin products
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor memilih produk yang ingin dimanage</li> <li>2. Actor mengubah data kemudian menekan tombol "simpan"</li> <li>3. Sistem akan menyimpan perubahan ke database lalu mengarahkan actor ke halaman admin products</li> </ol>
<b>Alternative Flow</b>	Actor bisa membuat produk baru dengan menekan tombol "New product"
<b>Post-Condition</b>	Data produk berhasil diubah

## 10. Manage kategori (CRUD)

<b>Objective</b>	Memungkinkan actor untuk manage kategori
<b>Actors</b>	Admin
<b>Pre-Condition</b>	Actor berada di halaman admin categories
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor memilih kategori yang ingin dimanage</li> <li>2. Actor mengubah data kemudian menekan tombol "simpan"</li> <li>3. Sistem akan menyimpan perubahan ke database lalu mengarahkan actor ke halaman admin categories</li> </ol>
<b>Alternative Flow</b>	Actor bisa membuat kategori baru dengan menekan tombol "New categories"
<b>Post-Condition</b>	Data kategori berhasil diubah

## 11. Manage pesanan (CRUD)

<b>Objective</b>	Memungkinkan actor untuk manage pesanan
<b>Actors</b>	Admin

<b>Pre-Condition</b>	Actor berada di halaman admin orders
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor memilih pesanan yang ingin dimanage</li> <li>2. Actor mengubah data kemudian menekan tombol “simpan”</li> <li>3. Sistem akan menyimpan perubahan ke database lalu mengarahkan actor ke halaman admin orders</li> </ol>
<b>Alternative Flow</b>	-
<b>Post-Condition</b>	Data pesanan berhasil diubah

## 12. Manage single page (CRUD)

<b>Objective</b>	Memungkinkan actor untuk manage single page
<b>Actors</b>	Admin
<b>Pre-Condition</b>	Actor berada di halaman admin singles
<b>Main Flow</b>	<ol style="list-style-type: none"> <li>1. Actor memilih single page yang ingin dimanage</li> <li>2. Actor mengubah data kemudian menekan tombol “simpan”</li> <li>3. Sistem akan menyimpan perubahan ke database lalu mengarahkan actor ke halaman admin singles</li> </ol>
<b>Alternative Flow</b>	Actor bisa membuat single page baru dengan menekan tombol “New single”
<b>Post-Condition</b>	Data single page berhasil diubah