

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Sistem Informasi**

Sistem informasi merupakan sebuah sistem dalam suatu organisasi yang memberikan dan menyediakan informasi kepada semua tingkatan dalam organisasi tersebut yang dapat digunakan untuk pengambilan keputusan[13].

#### **2.2 Sistem Pencatatan**

Sistem pencatatan merupakan sebuah sistem yang dapat digunakan untuk mencatat setiap transaksi yang terjadi. Sistem ini terintegrasi dengan sistem lainnya dalam sebuah perusahaan dengan tujuan untuk menunjang pelaksanaan proses bisnis secara menyeluruh. Dengan adanya sistem pencatatan dapat membantu sebuah perusahaan dalam mengontrol setiap aktivitas yang terjadi dan mempermudah dalam pengolahan inventori[14].

#### **2.3 Penjualan**

Penjualan merupakan suatu kegiatan yang dilakukan oleh penjual dan pembeli dimana penjual menawarkan barang atau jasa dan diharapkan dapat memperoleh keuntungan dari pembeli atau dengan kata lain penjualan adalah pemindahan hak kepemilikan atas suatu barang atau jasa.

Penjualan merupakan suatu persetujuan antara kedua belah pihak yaitu penjual dan pembeli dimana penjual menawarkan suatu produk dan pembeli dapat membeli produk tersebut sesuai dengan harga jual yang telah disepakati[15].

#### **1.4 Pembelian**

Pembelian adalah suatu usaha yang dilakukan untuk memenuhi kebutuhan akan barang atau jasa yang diperlukan berdasarkan kualitas dan kuantitas yang sesuai untuk kegiatan operasi dalam periode tertentu oleh perusahaan[16]. Biasanya pembelian dilakukan untuk memenuhi kebutuhan sendiri, kepentingan produksi maupun dijual kembali.

#### **1.5 Usaha Mikro, Kecil dan Menengah**

Pengertian UMKM menurut UU No.20 Tahun 2008 adalah sebuah usaha yang dikelola dan dimiliki oleh perorangan atau sekelompok kecil orang dan jumlah aset, hingga kekayaan maupun pendapatannya akan dibagi sesuai dengan bagiannya masing-masing[2].

- a. Usaha Mikro adalah usaha produktif milik orang perorangan dan/atau badan usaha perorangan yang memenuhi kriteria Usaha Mikro sebagaimana diatur dalam Undang – Undang ini.
- b. Usaha Kecil adalah usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perorangan atau badan usaha yang bukan merupakan anak perusahaan atau bukan cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dari Usaha Kecil sebagaimana dimaksud dalam Undang – Undang ini.
- c. Usaha Menengah adalah usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh perorangan atau badan usaha yang bukan merupakan anak perusahaan atau cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dengan Usaha Kecil atau Usaha Besar dengan jumlah kekayaan bersih atau hasil penjualan tahunan sebagaimana diatur dalam Undang – Undang ini.

## 1.6 Metode Prototyping

Metode *prototyping* merupakan metode pengembangan perangkat lunak yang berupa model fisik kerja sistem dan berfungsi sebagai tahap awal dari sistem. Dengan metode ini akan dihasilkan sebuah *prototype* sistem sebagai perantara pengembang dan *user* agar dapat berinteraksi dalam proses kegiatan pengembangan sistem[17].

*Prototyping* menghasilkan versi awalan yang akan dibangun berdasarkan dari sistem informasi yang dibutuhkan. *Feedback* dari *user* sangat diharapkan dari suatu metode *prototyping* karena metode ini melibatkan *user* dalam memeriksa *input, output, model* dan *user interface*. Metode ini dilakukan untuk melakukan validasi terhadap kebutuhan *user* dengan sistem yang akan dihasilkan[18].



Gambar 2. 1 Metode Prototyping

Sumber : [18]

## 1.7 Unified Modelling Language (UML)

UML merupakan bahasa pemodelan yang kemudian digunakan untuk merancang bagaimana sistem bekerja dan beroperasi dan berinteraksi dengan pengguna. UML mengidentifikasi komponen-komponen proses sebagai objek yang dicirikan oleh properti dan metode yang menggambarkan interaksi dan model yang dihasilkan akan dipahami oleh pengguna[19].

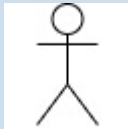

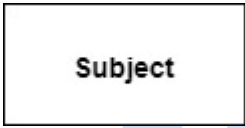

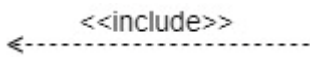
### 1.7.1 Use Case Diagram

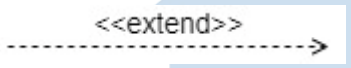

*Use Case Diagram* adalah diagram yang menggambarkan fungsi dasar dari sebuah sistem. *Use case* menjelaskan cara yang sangat sederhana dari fungsi utama suatu sistem. *Use case* digunakan untuk menggambarkan fungsi dari sistem dan sebagai model dialog antara aktor dan sistem. *Use case* dapat membantu bila kita sedang menyusun kebutuhan sistem,

mengkomunikasikan rancangan dengan pengguna dan merancang test untuk desain yang ada pada sistem[20].

**Tabel 2. 1 Simbol Use Case Diagram**

Sumber : [20]

Simbol	Keterangan
 <p>Actor / Role</p>	<p><i>Actor :</i></p> <ul style="list-style-type: none"> <li>- Merupakan orang atau sistem yang mendapatkan manfaat dari dan bersifat eksternal terhadap subjek.</li> <li>- Dapat dikaitkan dengan aktor lain menggunakan asosiasi spesialisasi/superclass, dilambangkan dengan panah dengan panah berongga.</li> </ul>
 <p>Use Case</p>	<p><i>Use Case :</i></p> <ul style="list-style-type: none"> <li>- Adalah elemen utama dari fungsionalitas sistem.</li> <li>- Dapat menyertakan dan memperpanjang <i>use case</i> lain.</li> <li>- Ditempatkan dalam batas sistem.</li> </ul>
 <p>Subject</p>	<p><i>Subject Boundary :</i></p> <ul style="list-style-type: none"> <li>- Tergolong dari nama subjek di dalam atau di atas.</li> <li>- Mewakili ruang lingkup subjek, misalnya sistem atau individu proses bisnis.</li> </ul>
	<p><i>Association Relationship :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk menghubungkan <i>actor</i> dengan <i>use case</i> yang digunakan untuk berinteraksi.</li> </ul>
	<p><i>Include Relationship :</i></p> <ul style="list-style-type: none"> <li>- Merupakan penyertaan fungsionalitas dari satu <i>use case</i> di dalam <i>use case</i> yang lain.</li> <li>- Memiliki panah yang ditarik dari <i>use case</i> dasar ke <i>use case</i> yang akan digunakan.</li> </ul>



Simbol	Keterangan
	<p><i>Extend Relationship :</i></p> <ul style="list-style-type: none"> <li>- Merupakan ekstensi dari <i>use case</i> untuk memasukkan <i>optional behavior</i>.</li> <li>- Memiliki panah yang ditarik dari <i>use case</i> ekstensi ke <i>use case</i> dasar.</li> </ul>
	<p><i>Generalization Relationship :</i></p> <ul style="list-style-type: none"> <li>- Menghubungkan <i>use case</i> yang khusus kepada yang lebih umum.</li> <li>- Memiliki tanda panah yang ditarik dari <i>use case</i> khusus ke <i>use case</i> dasar.</li> </ul>


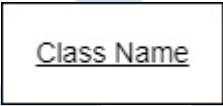
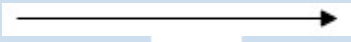
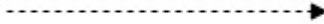



### 1.7.2 Activity Diagram

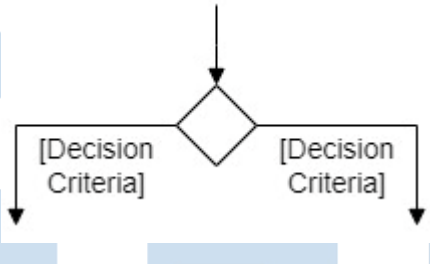
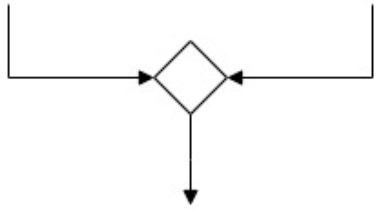
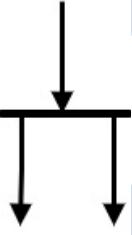
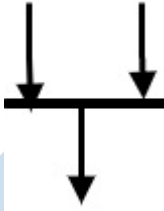

*Activity Diagram* merupakan gambaran tingkah laku proses bisnis dari suatu objek. Dalam beberapa kebiasaan dapat dilihat sebagai diagram aliran data yang digunakan dengan struktur analisis. *Activity diagram* dapat digunakan untuk menggambarkan segala perilaku aliran kerja dan fungsi aktivitas terkait[20].

**Tabel 2. 2 Simbol Activity Diagram**

Sumber : [20]

Simbol	Keterangan
	<p><i>Action :</i></p> <ul style="list-style-type: none"> <li>- Merupakan perilaku yang sederhana dan tidak dapat dijabarkan.</li> <li>- Diberi label sesuai dengan namanya.</li> </ul>
	<p><i>Activity :</i></p>

Simbol	Keterangan
	<ul style="list-style-type: none"> <li>- Digunakan untuk mewakili serangkaian tindakan.</li> <li>- Diberi label sesuai dengan namanya.</li> </ul>
	<p><i>Object Node :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk merepresentasikan suatu objek yang terhubung ke sekumpulan arah gabungan objek.</li> <li>- Diberi label sesuai dengan namanya.</li> </ul>
	<p><i>Control Flow :</i></p> <ul style="list-style-type: none"> <li>- Menunjukkan urutan eksekusi.</li> </ul>
	<p><i>Object Flow :</i></p> <ul style="list-style-type: none"> <li>- Menunjukkan aliran suatu objek dari satu aktivitas(tindakan) ke aktivitas(tindakan) lainnya.</li> </ul>
	<p><i>Initial Node :</i></p> <ul style="list-style-type: none"> <li>- Menggambarkan awal dari serangkaian tindakan atau kegiatan.</li> </ul>
	<p><i>Final – Activity Node :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk menghentikan semua aliran kontrol dan objek mengalir dalam suatu kegiatan atau tindakan.</li> </ul>
	<p><i>Final – Flow Node :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk menghentikan aliran kontrol atau aliran objek tertentu.</li> </ul>

Simbol	Keterangan
	<p><i>Decision Node :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk mewakili kondisi pengujian untuk memastikan bahwa aliran kontrol atau aliran objek hanya turun satu jalur.</li> <li>- Diberi label dengan kriteria keputusan untuk melanjutkan ke jalur tertentu.</li> </ul>
	<p><i>Merge node :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk menyatukan kembali jalur keputusan yang berbeda yang dibuat menggunakan <i>decision node</i>.</li> </ul>
	<p><i>Fork Node :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk membagi perilaku menjadi sekumpulan aktivitas paralel atau bersamaan.</li> </ul>
	<p><i>Join Node :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk menyatukan kembali serangkaian <i>activity</i> paralel atau bersamaan.</li> </ul>
	<p><i>Swimlane :</i></p> <ul style="list-style-type: none"> <li>- Digunakan untuk memecah <i>activity diagram</i> menjadi baris dan kolom untuk menetapkan aktivitas individu (atau tindakan) untuk individu atau objek</li> </ul>

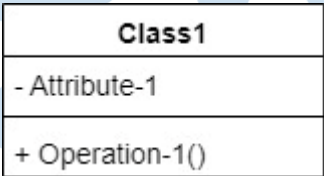
Simbol	Keterangan
	<p>yang bertanggung jawab untuk menjalankan aktivitas (atau tindakan).</p> <ul style="list-style-type: none"> <li>- Diberi label sesuai dengan nama individu atau objek yang bertanggung jawab.</li> </ul>

### 2.7.3 Class Diagram

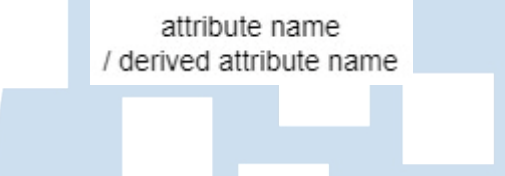
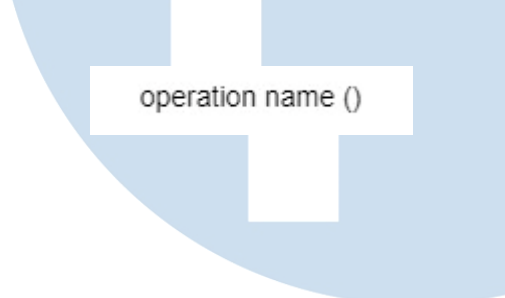
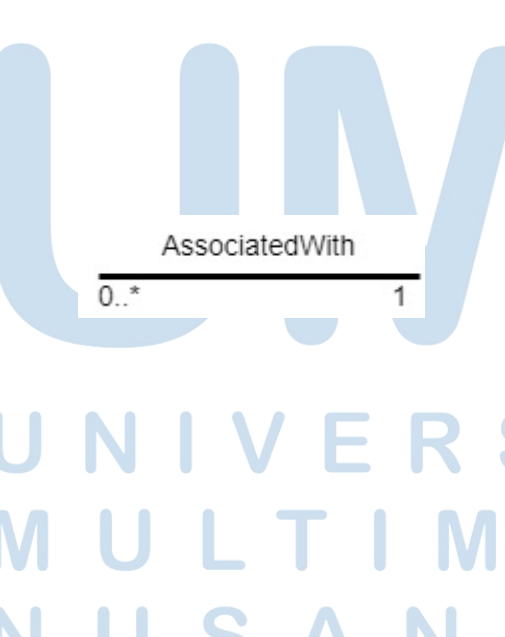
*Class diagram* merupakan sebuah model yang menunjukkan hubungan antar *class* yang bersifat konstan dari waktu ke waktu dalam sistem. *Class diagram* menggambarkan *class* yang mencakup status dan perilaku dengan hubungan antar *class*[20]. Berikut ini merupakan penjelasan tiap simbol dalam *class diagram* :

**Tabel 2. 3 Simbol Class Diagram**

Sumber : [20]

Simbol	Keterangan
	<p><i>A class :</i></p> <ul style="list-style-type: none"> <li>- Mewakili jenis orang, tempat, atau hal yang diperlukan sistem untuk menyimpan informasi.</li> <li>- Memiliki nama yang ditulis dengan menggunakan huruf tebal dan berada di kompartemen atas.</li> <li>- Memiliki daftar atribut yang berada di kompartemen tengah.</li> </ul>



Simbol	Keterangan
	<ul style="list-style-type: none"> <li>- Memiliki daftar operasi yang berada di kompartemen bawah.</li> </ul>
	<p><i>An attribute :</i></p> <ul style="list-style-type: none"> <li>- Menggambarkan keadaan suatu objek.</li> <li>- Dapat diturunkan dari atribut lain.</li> </ul>
	<p><i>An operation :</i></p> <ul style="list-style-type: none"> <li>- Menggambarkan tindakan atau fungsi yang dapat dilakukan oleh <i>class</i>.</li> <li>- Dapat diklasifikasikan sebagai <i>constructor</i>, <i>query</i>, atau <i>update operation</i>.</li> </ul>
	<p><i>An association :</i></p> <ul style="list-style-type: none"> <li>- Merupakan hubungan antara beberapa <i>class</i> atau dengan <i>class</i> itu sendiri.</li> <li>- Diberi label dengan menggunakan kata kerja atau peranan yang mewakili hubungan tersebut.</li> <li>- Berisi simbol multiplisitas, yang mewakili waktu minimum dan maksimum suatu <i>instance class</i> yang dapat diasosiasikan</li> </ul>

Simbol	Keterangan
	dengan <i>instance class</i> terkait.
	<i>A generalization</i> : Menggambarkan hubungan antara beberapa <i>class</i> .
	<i>An aggregation</i> : <ul style="list-style-type: none"> <li>- Mewakili hubungan logis antara beberapa kelas atau dengan kelas itu sendiri.</li> <li>- Merupakan bentuk khusus dari asosiasi.</li> </ul>
	<i>A composition</i> : <ul style="list-style-type: none"> <li>- Mewakili hubungan fisik antara beberapa kelas atau dengan kelas itu sendiri.</li> <li>- Merupakan bentuk khusus dari asosiasi.</li> </ul>

## 1.8 HTML

*HyperText Markup Language* atau HTML merupakan bahasa pemrograman terstruktur dimana dapat menambahkan *script* yang dikembangkan untuk membuat halaman *website* yang dapat ditampilkan atau diakses menggunakan *web browser*. Dengan kode tersebut, dapat diterjemahkan ke *web browser* dalam bentuk tulisan, grafik, maupun media lainnya. Terdapat juga kode lain yang bertujuan menghias dan mengatur gaya tampilan dan layout halaman *web* agar tampak menarik[21].

## 2.9 PHP

PHP (*Hypertext Processor*) merupakan bahasa pemrograman berbasis web. PHP menggunakan bahasa server – side script yang memungkinkan sintaks yang digunakan dan perintah yang diberikan akan sepenuhnya dijalankan oleh server tapi dapat disertakan pada halaman HTML biasa[12].

## 2.10 XAMPP

XAMPP merupakan sebuah perangkat lunak yang terdiri atas beberapa program yaitu Apache dan MySQL yang ditulis dan diterjemahkan dengan bahasa pemrograman Perl dan PHP dan digunakan sebagai server sendiri(localhost)[22].

## 2.11 Penelitian Terdahulu

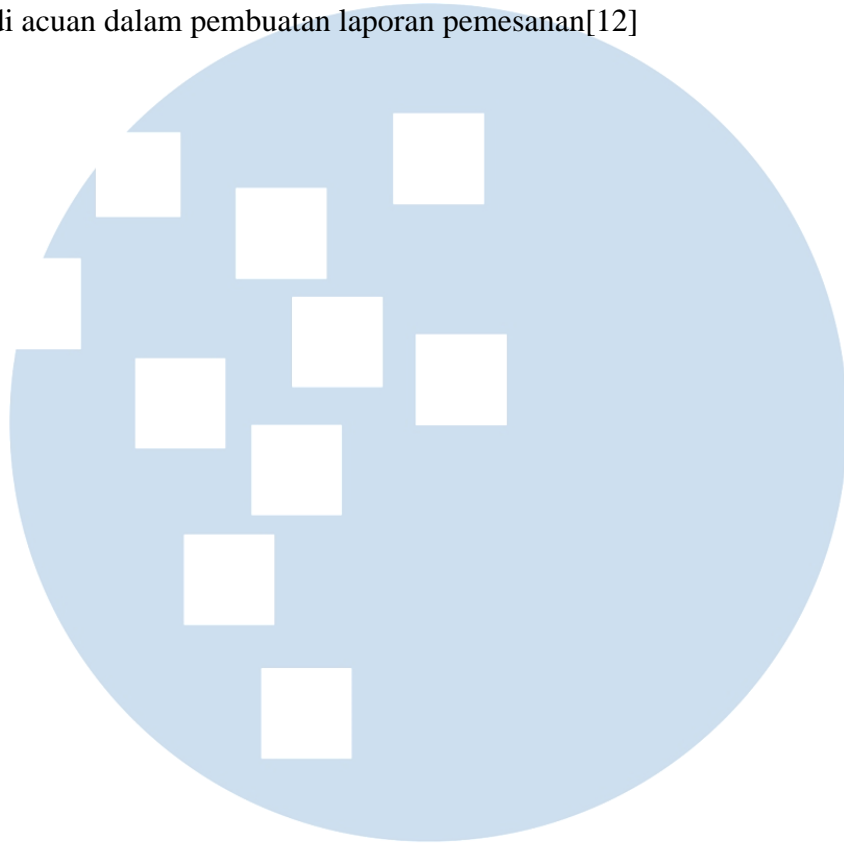
Tabel 2. 4 Penelitian Terdahulu

No.	Jurnal	Hasil
1.	<p><u>Judul Jurnal :</u> Rancang Banun Sistem Informasi Pembelian pada CV. Cimanggis Jaya Depok</p> <p><u>Penulis :</u> Ibnu Rusdi, Ade Sri Mulyani, Inna Herlina</p> <p><u>Tahun :</u> 2020</p> <p><u>Jurnal :</u> Jurnal AKRAB JUARA Vol. 5, No. 2</p>	Dengan adanya sistem yang telah terkomputerisasi pada CV. Cimanggis Jaya, segala proses pembelian dapat berjalan dengan lancar, efektif dan efisien serta pengolahan data dan informasi yang dibutuhkan dapat diperoleh dengan cepat, tepat dan akurat.
2.	<p><u>Judul Jurnal :</u> Pembentukan dan Penerapan Enterprise Resource Planning (ERP) pada UMKM (Usaha Kecil Menengah) Toko Budi dan M-Bisy Mart</p> <p><u>Penulis :</u> Kelly Rossa Sungkono, Riyanarto Sarno, Azzam Jihad Ulhaq, Muhammad Taufiqulsa'di, Isnaini</p>	Aplikasi ini dapat dipantau oleh pemilik secara <i>real-time</i> baik di lingkungan toko maupun di luar lingkungan toko. Fitur – fitur yang terdapat pada aplikasi ini adalah penjualan barang, pembelian barang, laporan penjualan, laporan keuangan, rangking penjualan produk, dan pencatatan stok barang. Dengan adanya aplikasi ini dapat mempermudah

No.	Jurnal	Hasil
	<p>Nurul KurniaSari, Zahrul Zizki Dinanto</p> <p><u>Tahun :</u> 2019</p> <p><u>Jurnal :</u> SEWAGATI Vol.3, No.3</p>	<p><i>monitoring</i> jual beli di kedua toko tersebut.</p>
3.	<p><u>Judul Jurnal :</u> Sistem Informasi Penunjang Pemesanan dan Desain Kue Pada Kue Artisan <i>Online</i> Berbasis Web</p> <p><u>Penulis :</u> Tania Jovita Wibowo, Suryasari, Arnold Aribowo, Andree E. Widjaja</p> <p><u>Tahun :</u> 2019</p> <p><u>Jurnal :</u> ULTIMA InfoSys, Vol.X, No.1</p>	<p>Hasil dari penelitian ini adalah sebuah sistem yang dapat menunjang dalam proses pemesanan dan desain kue pada toko kue artisan <i>online</i>. Dengan sistem ini, dapat memudahkan pembeli dalam memilih kue, melakukan pemesanan, hingga memudahkan dalam melakukan kustomisasi kue. Selain itu sistem ini juga dapat membantu pemilik toko dalam proses pemesanan yang praktis dan efisien.</p>
4.	<p><u>Judul Jurnal :</u> Analisis dan Perancangan Sistem Informasi Penjualan Produk Kesehatan pada PT. ABC</p> <p><u>Penulis :</u> T. Husain</p> <p><u>Tahun :</u> 2017</p> <p><u>Jurnal :</u> ULTIMA InfoSys, Vol.VIII, No.2</p>	<p>Hasil dari penelitian ini adalah sebuah sistem yang dapat memudahkan bagian administrasi dalam memproses data mitra, pembuatan laporan pemesanan, transaksi pembayaran, dan retur. Selain itu sistem juga dapat menyimpan data dengan baik serta pencarian data mitra lebih mudah dan cepat.</p>

Penelitian pertama ini memberikan kontribusi referensi untuk membuat dan merancang sistem pembelian[24]. Pada penelitian kedua ini memberikan gambaran dan contoh untuk pembuatan fitur pembelian barang[7]. Penelitian ketiga memberikan gambaran mengenai perancangan sistem menggunakan metode

prototyping dan gambaran sistem pemesanan[25]. Penelitian keempat ini dapat menjadi acuan dalam pembuatan laporan pemesanan[12]



# UMMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA