

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Penelitian Terkait**

Peneliti sebelumnya telah melakukan sejumlah penelitian yang relevan, salah satunya dilakukan oleh Raja Sabarudin (2020) yang membuat sistem manajemen proses pemesanan konveksi dan sablon dengan teknik prototype dan menghasilkan berbagai kesimpulan dari Sistem informasi pemesanan konveksi dan sablon berbasis web dan dapat diakses oleh tiga pihak yaitu direktur, *administrator*, dan *customer* sesuai penilaian bab. Pencatatan data, *user*, pelanggan, produk, kategori, pemesanan, daftar pemesanan, laporan pemesanan, penempatan pesanan, dan pengelolaan data profil, semuanya merupakan bagian dari sistem informasi pemesanan konveksi dan sablon yang dikembangkan. Direksi dan administrator dapat mengakses aplikasi untuk pelanggan dan bagian internal perusahaan di sini. Setiap bagian data terkini, termasuk *level*, kategori, produk, pengguna, pelanggan, data pesanan, laporan pesanan, dan laporan jurnal, dapat dikelola oleh direktur. Administrator memiliki akses untuk melihat produk, mengelola data pesanan, laporan pesanan, profil pribadi, dan data pelanggan. Pelanggan dapat memesan, mengubag daftar pesanan, dan profil pribadi untuk sementara. Laporan pemesanan merupakan hasil dari system informasi pemesanan. Sistem informasi pemesanan konveksi dan sablon mempermudah pemasukan data dan pencatatan transaksi, meminimalisir

human *error*, menyimpan data dengan aman, dan menjamin keakuratan. dan Klien yang tinggal jauh dari bisnis masih dapat melakukan pemesanan.[2]

Sistem informasi dirancang dengan teknik mengembangkan objek dengan menggunakan gambar *perspektif* berdasarkan ide-ide yang ditemukan dalam konteks permasalahan dunia nyata. OOAD berdasarkan Mathiassen (dalam Purwaningtias, 2018) Khususnya perancangan dan analisis sistem menggunakan pendekatan objek. Meskipun OOD adalah metode mengarahkan arsitektur perangkat lunak berdasarkan operasi objek, OOA adalah metode analitis yang melihat persyaratan (persyaratan yang harus dipenuhi sistem) dalam kaitannya dengan kelas dan objek yang ditemui dalam suatu perusahaan. Menurut Setiawan dkk (2019), suatu sistem atau subsistem. Menggunakan teknik desain dan analisis berorientasi objek meningkatkan kegunaan kembali sistem dan membuat sistem lebih dinamis. Setelah analisis dan desain berorientasi objek, pemodelan mengambil bentuk diagram yang dibuat dalam *Unified Modeling Language* (UML). Model atau teknik untuk membuat pengembangan perangkat lunak berorientasi objek disebut UML. Selain itu, UML menawarkan pedoman untuk membuat sistem cetak biru, yang terdiri dari skema database, konsep proses bisnis, kelas bahasa pemrograman, dan komponen sistem perangkat lunak. (Sonata & Winda Sari, 2019)

Untuk membangun sistem informasi yang dinamis dan memenuhi kebutuhan pengguna, proyek ini bermaksud merancang sistem informasi akuntansi UKM Konveksi Bim *Collection* berbasis website dengan

menggunakan metode OOAD. Dengan adanya sistem informasi pembukuan berbasis website ini diharapkan pengelolaan data keuangan dan pelaporan di Konveksi Bim *Collection* menjadi lebih efektif, efisien, mudah digunakan, mengurangi penggunaan ruang penyimpanan fisik dan mudah diakses oleh pemilik usaha dari mana saja dan kapan saja sehingga dapat digunakan sebagai acuan dalam pengambilan keputusan terhadap perkembangan usaha di masa depan[3]

Penelitian Agung Kurnia, Satria Audria Sakti, Rahmat Tulah (2022) adalah judul "Pemesanan Sablon Berbasis Web Pada Toko Jape". Pelajari tentang pengembangan sistem informasi P. Bahasa Pemrograman PHP dan MySQL. Dengan menghilangkan catatan manual dan pembatasan pasar, sistem ini bertujuan untuk berdagang secara efisien dan meningkatkan jumlah pelanggan. Solusi yang disarankan mencakup peningkatan laporan penjualan dan pembuatan situs web untuk pemesanan online. Artikel ini juga menjelaskan penggunaan bahasa pemrograman PHP dan UML dalam desain sistem. [4]

Menurut H. Husain, Mochamad Wahyudi, M. Safi'i, dan Muhammad Zarlis (2020), antarmuka pengguna merupakan aspek penting dalam suatu sistem atau program karena berinteraksi langsung dengan pengguna. Saat mengembangkan sistem *e-commerce*, desain antarmuka pengguna sangat penting untuk dipertimbangkan karena berdampak langsung pada pengguna. Preferensi pembeli dapat dipengaruhi oleh desain antarmuka pengguna yang menarik secara visual dan intuitif. Namun antarmuka pengguna dirancang

dengan buruk dapat menyebabkan pengunjung mengalami disorientasi saat mengunjungi situs web dalam waktu lama atau bahkan kecil kemungkinannya untuk melakukan pembelian.[5]

Pemasaran digital merupakan salah satu keunggulan teknologi yang dimanfaatkan oleh bisnis dan konsumen untuk berjualan (Pramadyanto, 2022). Pemasaran digital adalah strategi bisnis yang melibatkan penggunaan teknologi, termasuk media sosial, *e-commerce*, dan online, untuk memasarkan barang. [6]

## **2.2 Landasan Teori**

### **2.2.1 MYSQL**

MySQL adalah perangkat lunak yang ditujukan untuk mengelola database SQL, juga dikenal sebagai sistem manajemen database (DBMS). Yang mendukung penggunaan multi-jalur dan multi-pengguna. Dalam bahasa pemrograman PHP, terdapat dua ekstensi yang dapat dipakai untuk berinteraksi dengan MySQL, yaitu MySQL dan MySQL. Kedua ekstensi ini berperan dalam menghubungkan basis data MySQL dengan bahasa pemrograman yang digunakan. MySQL adalah versi yang telah diperbarui dan ditingkatkan dari MySQL, sehingga pengembang sistem disarankan untuk beralih dari MySQL ke MySQL. [7]

*Structured Query Language* memudahkan pengambilan data dalam *database* (SQL). Situs web (seperti PHP) atau aplikasi *non-web* (seperti *Visual Basic*) dapat mengakses data dalam *database*.

*MySQL* awalnya dikembangkan sebagai proyek internal untuk memenuhi kebutuhan manajemen basis data di tempat mereka bekerja. Kesuksesan awal proyek ini segera menarik perhatian komunitas pengembang, dan *MySQL* menjadi proyek open-source yang populer. Dengan fokus pada kecepatan, kinerja, dan kemudahan penggunaan,

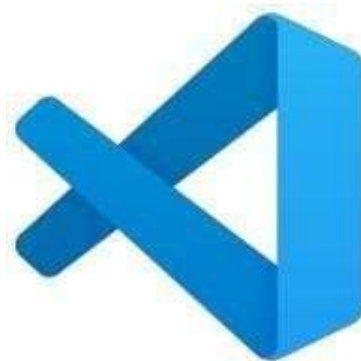
*MySQL* terus berkembang seiring waktu, dengan *MySQL AB* didirikan pada tahun 1996 untuk mendukung pengembangan dan pemasaran *MySQL*. Berbagai versi baru dari *MySQL* dirilis secara teratur dengan peningkatan fitur dan keamanan yang signifikan. Meskipun telah mengalami serangkaian akuisisi, termasuk oleh Sun Microsystems dan kemudian *Oracle Corporation*, *MySQL* tetap menjadi perangkat lunak *open-source* yang banyak digunakan di seluruh dunia.

*MySQL* adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan basis data yang mampu menerima dan mengirim data dengan sangat cepat, multiuser serta menggunakan perintah standar SQL (*Structured Query Language*). *MySQL* memiliki dua bentuk lisensi, yaitu Perangkat Lunak Bebas dan *Shareware*. *MySQL* kami yang biasa 88 yang digunakan adalah *MySQL Free Software* berada di bawah GNU/GPL (Lisensi Publik Umum)

### 2.2.2 *Visual Studio Code*

Dalam *Visual Studio Code*, Anda dapat membuat aplikasi konsol atau aplikasi bisnis dan konsumen, atau Anda dapat membuat komponen aplikasi, semuanya dalam satu program. Dalam sesuatu bentuk aplikasi *console*.

*Visual Studio Code* berasal dari proyek bernama "Monaco", yang digunakan dalam layanan *cloud-based Visual Studio Online*. Dirancang untuk memfasilitasi pengembangan yang cepat dan efektif, *Visual Studio Code* menawarkan kemampuan termasuk penyelesaian kode otomatis, *debugging*, penyorotan *sintaksis*, dan integrasi dengan berbagai alat pengembangan. Berbeda dengan *Visual Studio*, yang merupakan lingkungan pengembangan terpadu (IDE) yang lebih besar dan kompleks,



Gambar 2 1 *Visual Studio Code*

### 2.2.3 PHP

PHP atau *Hypertext Preprocessor* adalah bahasa pemrograman yang digunakan disisi server yang sangat populer untuk pengembangan web dinamis. Meskipun dikenal sebagai bahasa untuk membuat halaman web. PHP adalah singkatan dari *PHP Hypertext Processor*, suatu bahasa komputer untuk skrip sisi server yang terintegrasi dalam HTML (Fajar Junaedi, 2005: 14).

*Hypertext Preprocessor* atau biasa yang disebut PHP merupakan bahasa pemrograman pelengkap *Hyper Text Markup Language* (HTML) yang memungkinkan dibuatnya aplikasi web dinamis untuk pengolahan data, pemrosesan data dari *user via form*, membuat buku tamu, toko online, dan sebagainya dalam bentuk tampilan halaman *web*. [8]

PHP digunakan untuk membuat suatu situs *web* menjadi dinamis, yang seolah-olah dapat berinteraksi dua arah. Selain itu, PHP memiliki keunggulan karena merupakan sistem terbuka (*open source*), artinya siapa pun yang tertarik dengan bahasa pemrograman dapat mempelajari dan mengembangkannya dari mana saja di dunia, dan juga mampu mendukung dengan perangkat lunak database. Selain ASP (*Active Server Page*) buatan *Microsoft* yang notabene bersifat komersil, dan *JAVA*, PHP yang secara sintaksnya mirip dengan bahasa pemograman [9]

#### 2.2.4 HTML

Sebuah website atau *homepage* dapat dibuat dengan menggunakan bahasa markup HTML (*HypeText Markup Language*). Semua dokumen di Internet ditulis dalam HTML. HTML mendukung semua jenis dokumen, termasuk *hyperlink* yang dapat diklik, foto, dokumen multimedia, dan formulir yang dapat diisi. [10]

Dokumen *HTML*, hanyalah sebuah dokumen teks biasa dan disebut sebagai Markup Bahasa markup yang menyertakan tag HTML, kode markup yang menentukan format tampilan dokumen dikenal sebagai HTML.

#### 2.2.5 UML (*Unified Modelling Language*)

Struktur dan perilaku sistem perangkat lunak dijelaskan, dirancang, dan didokumentasikan menggunakan UML, bahasa pemodelan yang digunakan dalam rekayasa perangkat lunak.

Sejarah UML dimulai pada awal 1990 ketika Grady Booch, Ivar Jacobson, dan James Rumbaugh, yang dikenal sebagai "Tiga Amigos", bekerja sama untuk menggabungkan metode pemodelan yang mereka kembangkan sendiri (*Booch*, *OMT*, dan *OOSE*) menjadi satu bahasa pemodelan yang lebih kuat dan bersatu. Pada tahun 1994, mereka memulai proyek bersama untuk mengembangkan UML sebagai standar industri. Pada tahun 1997, UML versi 1.1 diterbitkan sebagai spesifikasi oleh *Object Management Group* (OMG), sebuah organisasi industri yang mengembangkan dan memelihara standar untuk teknologi



berbasis objek. Sejak itu, UML telah mengalami beberapa rilis dengan peningkatan fitur dan penyesuaian standar, termasuk UML 2.0 pada tahun 2005[11]

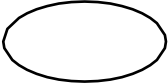



Model atau metode untuk membuat pengembangan perangkat lunak berorientasi objek dikenal dengan istilah UML. Selain itu, UML menawarkan pedoman untuk menghasilkan sistem cetak biru, yang terdiri dari skema *database*, konsep proses bisnis, kelas bahasa pemrograman, dan komponen sistem perangkat lunak.

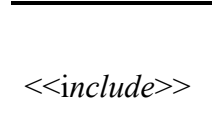
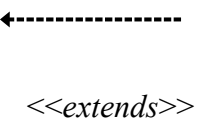
Dalam Perancangan berorientasi objek berbasis UML menggunakan alat bantu sebagai berikut:

1. *Use Case Diagram*

*Diagram use case* memodelkan bagaimana sistem informasi yang akan dikembangkan akan berperilaku. Use case menjelaskan bagaimana satu atau lebih aktor berinteraksi dengan sistem informasi yang akan dikembangkan. kasus penggunaan biasanya digunakan untuk menentukan fungsi mana yang ada dalam sistem informasi dan siapa yang berwenang untuk menggunakannya. Simbol Diagram Kasus yang digunakan dalam *Use Case Diagram* [12]

Tabel 2 1 Simbol *Use case*



No	Simbol	Keterangan
1		<i>Use case</i> : Interaksi dan abstraksi sistem dan aktor.
2		Aktor : Saat berbicara dengan <i>use case</i> , aktor mengambil peran sebagai seseorang, sistem yang berbeda, atau alat.
3		Asosiasi antara aktor dan <i>use case</i> Garis tanpa panah yang mewakili hubungan antara aktor dan kasus penggunaan menunjukkan data, dan bukannya bukan siapa atau apa yang meminta interaksi.
4		Asosiasi antara aktor dan <i>use case</i> yang menggunakan Panah terbuka digunakan untuk menunjukkan hubungan antara aktor dan <i>use case</i> ketika aktor berinteraksi dengan sistem secara pasif.


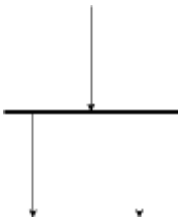
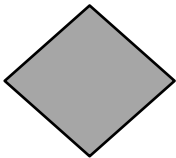
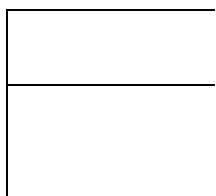
No	Simbol	Keterangan
5	 <<include>>	<i>Include</i> , Menunjukkan spesialisasi <i>actor</i> untuk dapat berpartisipasi dengan <i>use case</i> .
6	 <<extends>>	<i>Extend</i> , Merupakan perluasan dari <i>use case</i> yang berbeda jika prasyarat atau ketentuan terpenuhi.

2. Diagram aktivitas (*Activity diagram*)

Diagram aktivitas menggambarkan alur kerja sistem atau proses bisnis antar aktivitas. Tabel di bawah ini menampilkan simbol-simbol yang digunakan dalam diagram aktivitas.

Tabel 2 2 Simbol *Activity diagram*

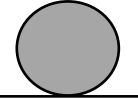
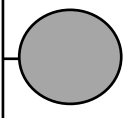
No	simbol	keterangan
1		<i>Start point</i> , terletak di kiri atas, menandakan dimulainya kegiatan.
2		<i>End Point</i> , akhir aktivitas.

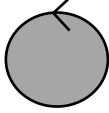

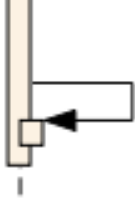


No	simbol	keterangan
3		<i>Activities</i> , menggambar kan suatu proses/kegiatan bisnis.
4		<i>Fork</i> /percabangan digunakan untuk menunjukkan kegiatan yang dikakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
5		<i>Decision points</i> , gambaran tentang kemungkinan benar dan salah dalam pengambilan keputusan.
6		<i>Swimlane</i> , Pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

### 3. Diagram Urutan (*Sequence Diagram*)

*Sequence Diagram* menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *Sequence Diagram* Bisa dilihat tabel di bawah.

Tabel 2 3 *Sequence Diagram*

No	Simbol	Keterangan
1		<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
2		<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interfaces</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>form entry</i> dan <i>form cetak</i> .

3		<p><i>Control class</i>, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.</p>
4		<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
5		<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
6		<p><i>Activation</i>, mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.</p>
7		<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>