

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Teori Terkait**

Dalam penulisan proposal ini, penulis terinspirasi dan merujuk pada berbagai penelitian terdahulu yang berfokus pada sistem inventaris gudang berbasis *web*. Beberapa penelitian yang menjadi referensi utama dalam tugas akhir ini antara lain :

Jurnal oleh Huda Fadhilah dan Yuli Asriningtias yang berjudul “Rancang Bangun Aplikasi Sistem Manajemen Gudang Berbasis *Web Mobile* pada CV. Intan Mulia Abadi” pada tahun 2023 yang membahas tentang merancang dan membuat aplikasi sistem manajemen gudang berbasis *web mobile* di CV. Intan Mulia Abadi agar meningkatkan efisiensi dan efektivitas pengelolaan data barang dan meningkatkan kinerja perusahaan. Berdasarkan hasil penelitian rancang bangun aplikasi sistem manajemen gudang berbasis *web mobile* yang dapat diakses melalui *web* ataupun *smartphone* dapat membantu pengelolaan data barang di CV. Intan Mulia Abadi sehingga pengelolaan data barang menjadi lebih efektif dan efisien[3].

Jurnal oleh Rendy Setiawan dan Alam Santosa yang berjudul “Perancangan Sistem Informasi Gudang K3LH-DP PT. Dirgantara Indonesia (Persero)” pada tahun 2018 yang membahas tentang dibutuhkannya sistem yang dapat memperbaiki pengolahan data. Sebagai solusi untuk mengatasi permasalahan-permasalahan yang ada di gudang K3LH-DP PT. Dirgantara

Indonesia (Persero) yaitu permasalahan-permasalahan yang ada yaitu kesalahan data yang rentan terjadi (duplikasi atau kehilangan data), laporan barang gudang tidak dapat disediakan secara langsung, kebutuhan waktu dan tenaga yang banyak dalam pencatatan maupun pengolahan data. Data dicatat dan diproses dengan *software Microsoft Excel* yang belum mempunyai integrasi data[4].

## **2.1 Landasan Teori**

### **2.1.1 Inventory Management System**

Dikutip dari jurnal.id menyebutkan bahwa *Inventory Management System* memiliki makna pengaturan persediaan dan berkaitan dengan aktivitas logistik sebuah perusahaan. Berdasarkan kutipan diatas kesimpulan umum *Inventory management sytem* ialah aktifitas yang dilakukan perusahaan untuk mengontrol ketersediaan barang baik yang sudah ada maupun yang akan ada untuk dijual ataupun untuk digunakan dalam suatu produksi[5].

### **2.1.2 Web Server**

*Web server* adalah sebuah perangkat lunak, *server* yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML[6].

### 2.1.3 Website

Pada dasarnya *web* merupakan suatu kumpulan *hyperlink* yang menuju alamat satu kealamat lainnya dengan bahasa HTML (*hypertext markup language*). *Website* atau situs dapat diartikan sebagai “kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan/atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman yang saling mempunyai hubungan”[7].

### 2.1.4 PHP

Menurut PHP adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan *web*. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman pemrograman umum. PHP di kembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh The PHP Group. Situs resmi PHP beralamat di <http://www.php.net>. PHP disebut bahasa bahasa pemrograman *server side* karena PHP diproses pada komputer *server*. Hal ini berbeda dibandingkan dengan bahasa pemrograman *client-side* seperti *JavaScript* yang diproses pada *web browser (client)*[8].

### 2.1.5 MySQL

MySQL adalah sebuah implementasi dari sistem sistem manajemen yang pada basis basis data yang relasional (RDBMS) yang

didistribusikan secara gratis. Setiap setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya turunan salah satu konsep utama dalam basis basis dan data data yang telah ada sebelumnya; SQL (*Structured Query Language*). SQL adalah sebuah inti konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data data, yang memungkinkan pada pengoperasian data dikerjakan dengan mudah secara otomatis[8].

#### **2.1.6 XAMPP**

XAMPP adalah salah satu paket *installer* yang berisi *Apache* yang merupakan *web server* tempat menyimpan file yang diperlukan *website*, dan *Phpmyadmin* sebagai aplikasi yang digunakan untuk perancangan *database MySQL*[9].

#### **2.1.7 Bootstrap**

*Bootstrap* adalah paket aplikasi siap pakai untuk membuat *front-end* sebuah *website*. Bisa dikatakan, *bootstrap* adalah *template* desain *web* dengan fitur plus. *Bootstrap* diciptakan untuk mempermudah proses desain *web* bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai HTML dan CSS, anda pun siap menggunakan *bootstrap*[9].

### 2.1.8 UML

*Unified Modeling Language* atau lebih sering dikenal dengan sebutan UML, adalah salah satu metode dalam teknik rekayasa perangkat lunak yang digunakan untuk menggambarkan alur dan cara kerja sistem, fungsi, tujuan dan mekanisme kontrol sistem tersebut. Dalam teknik rekayasa perangkat lunak bidang analisis dan perancangan sistem informasi, saat ini lebih banyak menggunakan gabungan dari konsep pemrograman berorientasi objek dengan teknik pembuatan perangkat lunak, dimana suatu sistem dilihat sebagai objek tersendiri yang sudah mencakup data dan proses atau dapat bekerja secara mandiri dalam satu set sistem (*package*). Pada tahun 1997, UML diterima sebagai standar teknik rekayasa perangkat lunak untuk pengembangan objek. Dalam teknik perancangan sistem informasi.

UML sebagai sebuah bahasa yang memberikan *vocabulary* dan tatanan penulisan kata-kata dalam *'MS Word'* untuk kegunaan komunikasi. Sebuah bahasa model adalah sebuah bahasa yang mempunyai *vocabulary* dan konsep tatanan / aturan penulisan serta secara fisik mempresentasikan dari sebuah sistem. Seperti halnya UML adalah sebuah bahasa standard untuk pengembangan sebuah *software* yang dapat menyampaikan bagaimana membuat dan membentuk model-model, tetapi tidak menyampaikan apa dan kapan model yang seharusnya dibuat yang merupakan salah satu proses implementasi pengembangan *software*. UML tidak hanya merupakan

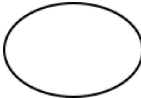




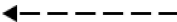
sebuah bahasa pemrograman visual saja, namun juga dapat secara langsung dihubungkan ke berbagai bahasa pemrograman, seperti JAVA, C++, Visual Basic, atau bahkan dihubungkan secara langsung ke dalam sebuah object-oriented *database*. Begitu juga mengenai pendokumentasian dapat dilakukan seperti; *requirements*, arsitektur, *design*, *source code*, *project plan*, *tests*, dan *prototypes*. Untuk dapat memahami UML membutuhkan bentuk konsep dari sebuah bahasa model, dan mempelajari 3 (tiga) elemen utama dari UML seperti *building block*, aturan-aturan yang menyatakan bagaimana *building block* diletakkan secara bersamaan, dan beberapa mekanisme umum[10].

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use case*

*Use case diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu:




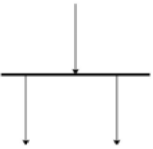
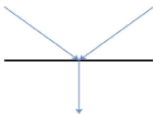
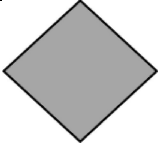

Tabel 2. 1 Simbol *Use Case Diagram*

No	Simbol	Keterangan
1		<i>Use case</i> menggambarkan fungsionalitas yang disediakan system sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja.
2		<i>Actor</i> adalah <i>Abstraction</i> dari orang atau, system yang lain yang mengaktifkan fungsi dari target <i>system</i> . Untuk mengidentifikasi <i>actor</i> harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target <i>system</i> . Orang atau <i>system</i> bisa muncul.
3		Asosiasi antar <i>actor</i> dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengidentifikasi siapa atau apa yang berinteraksi secara langsung dan bukannya mengindikasikan data.
4		Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
5		<i>Include</i> , merupakan di dalam <i>use case</i> lain ( <i>required</i> ) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
6		<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi

## 1) Diagram Aktivitas

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* yaitu:

Tabel 2. 2 Simbol *Activity Diagram*

No	Simbol	Keterangan
1		<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
2		<i>End Point</i> , akhir aktivitas.
3		<i>Activities</i> , menggambar kan suatu proses/kegiatan bisnis.
4		<i>Fork</i> /percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan.
5		<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
6		<i>Decision points</i> menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i> .
7		<i>Swimlane points</i> pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

### 2.1.9 CodeIgniter

*CodeIgniter* merupakan salah satu dari sekian banyak *framework* PHP yang ada. *CodeIgniter* dikembangkan oleh Rick Ellis, tujuan dari pembuatan *framework* *CodeIgniter* ini menurut *user*

manualnya adalah untuk menghasilkan *framework* yang akan dapat digunakan untuk pengembangan proyek pembuatan situs *web* secara lebih cepat dibandingkan dengan pembuatan situs *web* dengan cara koding secara manual, dengan menyediakan banyak sekali pustaka yang dibutuhkan dalam pembuatan situs *web*, dengan antarmuka yang sederhana dan struktur logika untuk mengakses pustaka yang dibutuhkan.

*CodeIgniter* menerapkan lingkungan pengembangan dengan metode MVC (*Model View Controller*). MVC memisahkan antara logika pembuatan kode dengan pembuatan *template* atau tampilan situs *web*. Penggunaan MVC membuat pembuatan sebuah proyek situs *web* menjadi lebih terstruktur dan lebih sederhana. Secara sederhana konsep MVC terdiri dari tiga bagian yaitu bagian *Model*, bagian *View* dan bagian *Controller*[11].