

## **BAB II**

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

#### **2.1. Tinjauan Pustaka**

Dengan pesatnya laju perkembangan teknologi, sistem inventarisasi data akan lebih efektif jika menggunakan sebuah aplikasi khusus. Berdasarkan penelitian yang telah dilakukan sebelumnya, pembuatan aplikasi khusus ini akan membuat pengolahan dokumen lebih efektif dan penyimpanan dokumen akan lebih aman.

Dina Apriani (2013) melakukan sebuah penelitian yang berjudul “APLIKASI PENGOLAHAN DATA INVENTARIS BARANG PADA PT. ASURANSI JIWASRAYA (Persero) PALEMBANG”. Penelitian tersebut bertujuan untuk menghasilkan aplikasi inventaris barang pada PT. Asuransi Jiwasraya (Persero) Palembang, karena sebelumnya inventaris barang masih dilakukan dengan menggunakan *Microsoft Word*. Pada penelitian tersebut pembuatan aplikasi menggunakan Visual Basic.

Pembuatan aplikasi mengenai inventaris barang juga sudah pernah dilakukan sebelumnya dalam penelitian yang berjudul “SISTEM INFORMASI PENGELOLAAN DATA INVENTARIS PADA PERCETAKAN CV. EMJI SPS” (Kurniawan 2013). Penelitian tersebut bertujuan untuk memperbaiki sistem sebelumnya yang masih manual, sehingga sering terjadi kesalahan saat memasukan data barang. Perancangan aplikasi tersebut menggunakan bahasa pemrograman Java.

Yuhendra dan Rendi Poerwanta (2014), pada jurnalnya yang berjudul “PERANCANGAN SISTEM INVENTORY SPARE PARTS MOBIL PADA CV. AUTO PARTS TOYOTA BERBASIS APLIKASI JAVA”, merancang sebuah sistem *inventory* yang berfungsi untuk pencatatan dan pemantauan data *spare parts* pada CV. Auto Parts Toyota. Sistem ini dirancang untuk menggantikan sistem

sebelumnya yang masih menggunakan *Microsoft Excel*, karena data yang ada tidak terstruktur dengan baik. Perancangan sistem ini menggunakan bahasa pemrograman *java* dan database *MySQL*.

Menurut analisis yang dilakukan dari hasil ketiga penelitian sebelumnya, dapat disimpulkan bahwa aplikasi dapat memperbaiki sistem yang digunakan sebelumnya. Namun aplikasi yang dibuat hanya berupa pencatatan barang masuk dan keluar. Dalam skripsi ini, penulis membutuhkan aplikasi yang tidak hanya berupa aplikasi pencatatan barang masuk dan keluar saja. Berdasarkan kekurangan dari ketiga penelitian sebelumnya, maka penulis dalam skripsi ini menambahkan fitur-fitur sebagai berikut:

1. Sistem dapat menyimpan data inventaris kemudian mengekspor ke file *Microsoft Excel*.
2. Sistem dapat membuat *report* secara otomatis.
3. Sistem dapat mencetak *form* pemesanan secara otomatis.

## **2.2. Landasan Teori**

### **2.2.1. Pengertian Inventarisasi**

Menurut Sugiana (2013) Inventarisasi aset adalah serangkaian kegiatan untuk melakukan pendataan, pencatatan, pelaporan hasil pendataan aset, dan mendokumentasikannya baik aset berwujud maupun aset tidak berwujud pada suatu waktu tertentu. Sedangkan menurut PP No. 27 2014: "Inventarisasi adalah kegiatan untuk melakukan pendataan, pencatatan, dan pelaporan hasil pendataan Barang Milik Negara atau Daerah". Dan menurut KBBI inventarisasi merupakan pencatatan atau pendaftaran barang-barang milik kantor, (sekolah, rumah tangga, dan sebagainya) yang dipakai dalam melaksanakan tugas. Jadi dari ketiga pengertian diatas dapat disimpulkan bahwa inventarisasi merupakan semua kegiatan yang berhubungan dengan pencatatan barang-barang milik suatu instansi baik itu barang yang diadakan dengan pembelian menggunakan anggaran maupun barang yang merupakan hibah atau pemberian.

Di dalam buku “Manajemen Aset Pariwisata”, Dr. A Sugiana (2013) menjelaskan bahwa inventarisasi merupakan tahap ketiga setelah tahap perencanaan kebutuhan dan tahap pengadaan. Tujuan dari dilakukannya inventarisasi adalah sebagai berikut:

1. Untuk menciptakan tertib administrasi,
2. Untuk pengamanan terhadap asset yang ada,
3. Untuk melakukan pengendalian dan pengawasan pada asset.

Jenis barang atau asset yang perlu diinventarisasi ada dua, yaitu:

1. Barang atau asset yang berwujud, contohnya tanah, bangunan, peralatan atau perlengkapan, dll.
2. Barang atau asset yang tidak berwujud, contohnya hak paten, hak cipta, hak merek, dll.

### **2.2.2. Pengertian Aplikasi *Dekstop***

Aplikasi *dekstop* adalah suatu aplikasi yang dapat berjalan sendiri atau independen dalam suatu komputer tanpa menggunakan *browser* atau tanpa terkoneksi *Internet* (Konixbam,2009). Salah satu bahasa pemograman yang dapat digunakan untuk megembangkan aplikasi *dekstop* adalah bahasa pemograman C#.

### **2.2.3. Bahasa Pemograman C#**

C# (dibaca: C sharp) merupakan bahasa pemograman yang berorientasi objek, C# dikembangkan oleh *Microsoft*. Bahasa pemograman C# telah di setandarisasi secara Internasional oleh ECMA. Bahasa pemograman C# telah digunakan dalam membangun berbagai aplikasi, seperti aplikasi *dekstop* dan aplikasi berbasis *web* serta aplikasi berbasis *web service*. Bahasa pemograman C# memiliki kelebihan dibandingkan dengan bahasa pemograman lain, yaitu:

1. Bahasa pemrograman C# dibuat sebagai bahasa pemrograman yang bersifat *general-purpose*,
2. Berorientasi objek (*Object-Oriented Language*),
3. Modern,
4. Sederhana,
5. *Powerfull* dan Fleksibel,
6. Modular (Abid Alfian Syakir, 2015).

#### **2.2.4. *Windows Presentation Foundation (WPF)***

*Windows Presentation Foundation (WPF)* adalah framework UI yang digunakan untuk membuat aplikasi dekstop. WPF mendukung serangkaian fitur dalam pengembangan aplikasi, seperti *resources*, *control*, grafik, *layout*, *data binding*, dan keamanan. WPF menggunakan *Extensible Application Markup Language (XAML)* untuk menyediakan model deklaratif pemrograman aplikasi (Microsoft, 2017).

#### **2.2.5. *SQLite***

SQLite merupakan sebuah aplikasi *embedded SQL database*, yang artinya tidak seperti pada aplikasi database SQL lainnya yang membutuhkan server untuk memproses database, SQLite akan membaca dan juga menulis langsung file pada *disk* lengkap dengan tabel, *trigger*, *views*, dan *indeces* (SQLite, 2017).

#### **2.2.6. *Unified Markup Language (UML)***

*Unified Markup Language (UML)* merupakan bahasa pemodelan secara grafis yang digunakan untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan seluruh rancangan aplikasi perangkat lunak. Model UML yang dipakai dalam pengembangan aplikasi yaitu model *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*.

1. *Use Case Diagram*

*Use Case Diagram* mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem yang dibuat. Dapat dikatakan *Use Case* digunakan untuk mengetahui fungsi yang ada di dalam sistem dan siapa saja yang berhak menggunakan fungsi-fungsi yang dibutuhkan.

2. *Activity Diagram*

*Activity Diagram* merupakan diagram yang digunakan untuk menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

3. *Class Diagram*

*Class Diagram* merupakan diagram yang digunakan untuk menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class Diagram* secara khas meliputi: Nama Kelas (*Class Name*), Atribut (*Attributes*), Operasi (*Operations*), dan Relasi (*Relationships*).

### 2.2.7. *Black Box Testing*

Pengujian pada perangkat lunak dapat dilakukan dengan 2 metode yaitu metode *white box testing* dan metode *black box testing*. Dari kedua metode tersebut dalam skripsi dipilih menggunakan *black box testing*. Perangkat lunak memerlukan seperangkat tes untuk pencarian kesalahan fungsi-fungsi dalam aplikasi sehingga dalam hal ini *black box testing* lebih sesuai dibandingkan *white box testing*. Pengujian ini digunakan untuk mengetahui apakah fungsi-fungsi dalam perangkat lunak sudah sesuai dengan yang diharapkan atau belum.

Menurut Roger S. Pressman (2010), *black box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineer* untuk memperoleh *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program. *Black box testing* berusaha untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau fungsi yang hilang,
2. Kesalahan antarmuka,
3. Kesalahan dalam struktur data atau akses database eksternal,
4. Kesalahan kinerja,
5. Kesalahan inisialisasi dan pemutusan kesalahan.