

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka merupakan penelusuran terhadap berbagai *literature* hasil penelitian sebelumnya yang relevan atau memiliki keterkaitan dengan permasalahan yang telah penulis pelajari sebelumnya.

Dalam penelitian Michael Christian Wibisono (2013), untuk memudahkan pencatatan *stock* barang dan aset perusahaan, maka dibuat aplikasi untuk melakukan pencatatan *stock* dengan menggunakan *scan barcode*. Aplikasi ini dibuat pada *mobile device* berbasis sistem operasi Android karena berguna sebagai telepon genggam, *mobile device* juga memiliki fasilitas yang *modern* seperti kamera dan koneksi internet menggunakan *wifi* sehingga dapat pula dijadikan sebagai *barcode reader* yang dapat terhubung dengan server perusahaan. Selain itu harga yang ditawarkan untuk dapat memperoleh sebuah *mobile device* adalah lebih terjangkau daripada sebuah alat *barcode reader*.

Sedangkan menurut penelitian Ronny Doan P. Simanjuntak (2018), perlu dibangun suatu sistem informasi manajemen aset yang dapat memproses data cepat dan terpusat. Metode yang digunakan dalam membangun Sistem Informasi Manajemen Aset siklus SDLC (*System Development Life Cycle*) ini adalah dengan model *Waterfall* yang dimulai menganalisa kebutuhan *software* yang menyediakan registrasi aset, pemeliharaan, perpindahan, penghapusan dan pelaporan. Perancangan arsitektur aplikasi ini menggunakan bahasa pemodelan UML, arsitektur data menggunakan ERD dan LRS. Pemograman terstruktur dengan pemograman *Java Netbeans*. Pengujian aplikasi menggunakan *black box* dari sistem informasi manajemen, diharapkan seluruh data aset tercatat dengan baik, proses pengelolaan data cepat, jelas dan terpusat, pengelolaan data lebih efektif dan efisien serta sistem pelaporan dapat dilakukan disetiap saat tergantung kebutuhan.

Dalam penelitian Adventus Angga Kurniawan (2015), proses inventarisasi akan terkait dengan masalah pengadaan barang dari SKPD. Proses inventarisasi juga penting agar semua aset yang terdapat dalam SKPD bisa didata sehingga dapat diketahui *stock opname* atau persediaan di gudang untuk operasional SKPD.

2.2 Landasan Teori

2.2.1 Desktop Application

Menurut Dew Omenn (2013), *desktop application* atau aplikasi *desktop* merupakan suatu aplikasi yang dapat berjalan sendiri atau independen tanpa menggunakan *browser* atau koneksi internet disuatu komputer.

Desktop Application atau *software* bisa dikatakan sebagai perangkat lunak yang berjalan pada sebuah komputer dan dioperasikan langsung oleh manusia. Perangkat lunak ini berfungsi mengatur sistem kerja pada komputer melalui sebuah perintah yang diberikan oleh penggunanya.

2.2.2 Bahasa Pemograman Java

Menurut Adi Nugroho (2009), *Java* merupakan Bahasa pemrograman berorientasi objek untuk pengembangan aplikasi mandiri atau tanpa jaringan internet, aplikasi berbasis internet, aplikasi untuk perangkat cerdas yang dapat berkomunikasi lewat internet atau jaringan komunikasi. Melalui teknologi *Java*, dimungkinkan perangkat *audio stereo* dirumah terhubung dengan jaringan komputer. *Java* tidak hanya untuk membuat *applet* yang memerintah halaman web tapi juga *Java* telah menjadi bahasa untuk pengembangan aplikasi skala *enterprise* berbasis jaringan besar.

2.2.3 Netbeans IDE

Menurut Heiko Bock (2009), *NetBeans* adalah *Integrated Development Environment (IDE)* berbasis *Java* dari *Sun Microsystems* yang berjalan di atas *Swing*. *Swing* adalah sebuah teknologi *Java* untuk pengembangan aplikasi *desktop* yang dapat bejalan di berbagai macam *platforms* seperti Windows, Linux, Mac OS X dan Solaris.

2.2.4 MySQL

Menurut Junaedi (2005), MySQL adalah jenis *database server* yang sangat terkenal dan sangat banyak digunakan untuk membangun aplikasi web yang menggunakan *database* sebagai sumber dan pengelolaan datanya.

MySQL adalah sebuah *Database Management System (DBMS)* populer yang memiliki fungsi *Relational Database Manajemen System (RDBMS)*. Selain itu MySQL software merupakan suatu aplikasi yang sifatnya *open source* dan juga memiliki kinerja sangat cepat, *reliable*, dan mudah untuk digunakan serta bekerja dengan arsitektur *client*

server atau *embedded systems*. Dikarenakan faktor *open source* dan populer tersebut maka cocok untuk mendemonstrasikan proses replikasi basis data yang digunakan.

2.2.5 Unified Modeling Language (UML)

Menurut Booch (2005) UML adalah Bahasa standar untuk membuat rancangan *software*. UML biasanya digunakan untuk menggambarkan dan membangun, dokumen artifak dari *software intensive system*. UML merupakan suatu perangkat lunak (*software*) yang dipergunakan dalam merancang desain sebuah sistem.

2.2.6 Entity Relationship Diagram (ERD)

Menurut Loonam (2010), *Entity Relationship Diagram* (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi yang ada, biasanya oleh *System Analyst* dalam tahap analisis persyaratan proyek pengembangan sistem. Seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain *database* relasional yang mendasari sistem informasi yang dikembangkan.

2.2.7 iReport

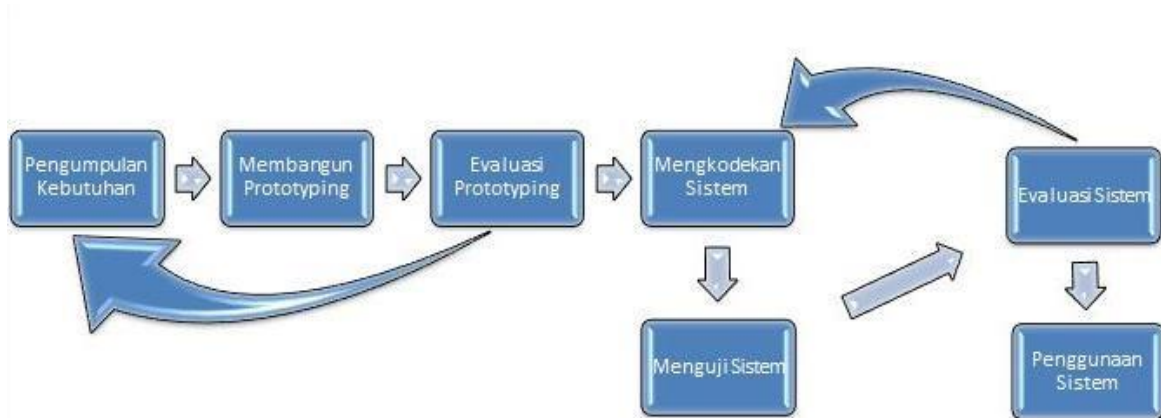
Menurut Hendra Kurniawan (2011), iReport adalah *report designer visual* yang dibangun pada JasperReport. iReport bersifat intuitif dan mudah digunakan untuk pembangun laporan visual atau desainer untuk JasperReport dan tertulis dalam kitab Java. Sebagai alternatif, terdapat *tools* iReport (dengan *library* JasperReport) yang dapat membantu dalam pembuatan laporan. *Library* JasperReport sendiri merupakan Java Library (JAR) yang bersifat *open* dan dirancang untuk menambahkan kemampuan pelaporan (*reporting capabilities*) pada aplikasi *Java*.

2.2.8 System Development Life Cycle (SDLC)

Dalam perancangan dan pembangunan aplikasi pendataan aset ini, peneliti menggunakan metode *Software Development Life Cycle* (SDLC). Metode ini dipakai untuk proses pengembangan sistem dalam cara yang terstruktur dan metodis yang mengharuskan tahap life cycle dari awal ide sampai sistem final dan dilakukan secara beraturan.

SDLC yang digunakan adalah model *software prototyping*, menurut Howard (1997) yaitu salah satu pendekatan dalam rekayasa perangkat lunak yang secara langsung mendemonstrasikan bagaimana sebuah perangkat lunak atau komponen-komponen perangkat lunak akan bekerja dalam lingkungannya sebelum tahapan konstruksi aktual dilakukan.

Salah satu ciri dari metodologi ini adalah *system developer*, klien dan pengguna dapat bereksperimen dan melihat cara kerja sistem sejak awal. Model ini juga memiliki tahapan seperti pada gambar 2.1 , yaitu:



Gambar 2. 1 Tahapan prototyping

1. Pengumpulan kebutuhan

Mendefinisikan format perangkat lunak, mengidentifikasi kebutuhan, dan gambaran sistem yang akan dibangun.

2. Membangun *prototyping*

Membuat rancangan sementara sistem yang mudah dipahami pelanggan (contohnya membuat input dan format output).

3. Evaluasi *prototyping*

Evaluasi yang dilakukan oleh pelanggan apakah *prototyping* sudah sesuai keinginan atau belum. Jika belum, *developer* dapat mengulang rancangan prototype dari tahapan 1 dan 2.

4. Mengkodekan sistem

Jika sudah disepakati oleh pelanggan, maka tahap selanjutnya adalah membangun sistem sesuai bahasa pemrograman.

5. Menguji sistem

Setelah sistem berhasil dibangun, sistem ini dapat di uji coba dengan *White box*, pengujian arsitektur, *Basis Path*, *Black Box* dan lain-lain.

6. Evaluasi sistem

Evaluasi yang dilakukan oleh pelanggan apakah sistem sesuai keinginan atau tidak.

Jika tidak, maka developer dapat mengulang dari tahapan 4 dan 5.

7. Menggunakan sistem

Software yang sudah dibangun dapat digunakan.