

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

2.1.1. Pengertian Persediaan

Menurut Sofyan Assauri dalam (Jiang, Wang, & Wang, 2018) Persediaan barang ialah sebagai suatu aktiva lancar yang meliputi barang-barang yang merupakan milik perusahaan dengan maksud supaya dijual dalam suatu periode usaha normal ataupun persediaan barang-barang yang masih dalam pekerjaan sebuah proses menghemat ongkos penyimpanan dan pemeliharaan terhadap persediaan tersebut.

2.1.2. Pengertian Sistem

Menurut Tohari (2014:2) “Sistem adalah kumpulan atau himpunan dari unsur atau variable yang saling terkait, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan”.

1. Pengertian Sistem Informasi

Menurut Pratama (2014:15) menyatakan bahwa “sistem informasi merupakan gabungan dari empat bagian utama. Keempat bagian utama tersebut mencakup perangkat lunak (*software*), perangkat keras (*hardware*), infrastruktur dan sumber daya manusia (SDM)”.

2. Pengertian Sistem Berorientasi Objek (OOP)

“Pemrograman berorientasi objek adalah suatu cara baru dalam berpikir serta berlogika untuk menghadapi masalah-masalah yang akan dicoba atasi dengan bantuan komputer” (Fadallah & Rosyida, 2018).

2.1.3. Program

1. NetBeans IDE 8.2

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 berjalan di berbagai macam *platforms* seperti *Windows*, *Linux*, *Mac OS X* dan *Solaris* (Heiko Bock, 2009) dalam (Rahmat Fauzi Siregar, 2017).

Suatu IDE adalah lingkup pemrograman yang diintegrasikan kedalam suatu aplikasi perangkat lunak yang menyediakan pembangun *Graphic User Interface* (GUI), suatu text atau kode editor, suatu compiler atau interpreter dan suatu debugger. NetBeans merupakan software development yang *Open Source*, dengan kata lain *software* ini di bawah pengembangan bersama dan bebas biaya. NetBeans merupakan sebuah proyek kode terbuka yang sukses dengan pengguna yang sangat luas, komunitas yang terus tumbuh, dan memiliki hampir 100 mitra. Sun Microsystems mendirikan proyek kode terbuka NetBeans pada bulan Juni 2000 dan terus menjadi sponsor utama. Saat ini terdapat dua produk : NetBeans IDE dan Netbeans Platform (Heiko Bock, 2009) (Rahmat Fauzi Siregar, 2017).

2. MySQL

Menurut Sutanto dalam (Purba, 2015), My Structured Query Language (MySQL) adalah sebuah perangkat lunak sistem manajemen basis data yang multithread, multi-user, dengan sekitar 6 juta instalasi

diseluruh dunia. MySQL AB membuat MySQL teredia sebagai perangkat lunak gratis dibawah lisensi GNU GPL (General Public License).

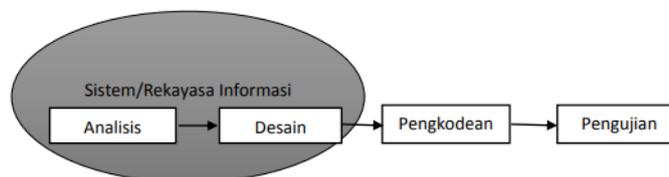
Menurut Faisal dalam (Risdiyansyah, 2017) bahwa “MySQL merupakan database server yang bersifat multiuser dan multi-threaded. SQL adalah bahasa database standar yang memudahkan penyimpanan, pengubahan dan akses informasi. Pada MySQL dikenal istilah database dan tabel. Tabel adalah sebuah struktur data dua dimensi yang terdiri dari baris-baris record dan kolom.

2.1.4. Basis Data

Menurut Rosa dan Shalahuddin (2015:43) “basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat”.

2.1.5. Metode Pengembangan Perangkat Lunak

Metode yang digunakan pada pengembangan perangkat lunak ini menggunakan waterfall Rosa dan Shalahuddin (2015:28) yang terbagi dalam beberapa bagian, yaitu :



Sumber : Rosa dan Shalahuddin (2015:28)

Gambar II.1 Tahapan Metode Waterfall

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat

lunak seperti apa yang dibutuhkan oleh *user* . Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan kode program

Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan keinginan.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirim ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan

mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.2. Teori Pendukung

2.2.1. *Entity Relation Diagram (ERD)*

Menurut Rosa dan Shalahuddin (2015:50) “ ERD digunakan untuk pemodelan basis data relasional sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD”.

2.2.2. *Logical Record Structure (LRS)*

Menurut Friyadie dalam Taufik (2017) “sebelum tabel dibentuk dari field atau atribut entitas secara fisik atau level internal, maka harus dibuatkan suatu bentuk relational model yang dibuat secara logic atau level external dan konsep, dari pernyataan tersebut dibutuhkan yang disebut dengan *Logical Record Structure (LRS)*”

2.2.3. *Unified Modelling Language*

Menurut Rosa Dan Shalahuddin (2015:133) “UML (Unified Modelling Language) adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasikan objek.

Menurut Rosa dan Shalahuddin (2015: 140) UML ini terdiri dari 13 macam diagram namun hanya beberapa diagram yang digunakan, diantaranya :

1. Activity Diagram

Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Rosa dan Shalahuddin, 2015: 161).

Menurut Rosa dan Shalahuddin (2015: 161) Diagram aktivitas juga banyak digunakan untuk mendefinisi hal-hal berikut :

- a. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan .
- b. Urutan atau pengelompokan tampilan dari sistem / user interface dimana setiap aktivitas dianggap memiliki sebuah rancangan antar muka tampilan.
- c. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kamus ujinya .
- d. Rancangan menu yang ditampilkan pada perangkat lunak.

2. *Class Diagram*

Menurut Rosa dan Shalahuddin (2015: 141) diagram kelas atau *class* diagram menggambarkan struktur sistem dari segi pendefinisian kelaskelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut pola dan metode atau operasi ;

- a. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
- b. Operasi atau metode adalah fungsi yang dimiliki oleh suatu kelas.

3. *Use Case Diagram*

Menurut Rosa dan Shalahuddin (2015: 155) *Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami.

Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut Aktor dan *use case* yaitu :

- a. Aktor merupakan orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- b. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor

4. *Sequence* Diagram

Menurut Rosa dan Shalahuddin (2015: 165) diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.