

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sistem Informasi**

Sistem informasi menurut Cegielski pada bukunya yang berjudul *Introduction to Information System - Fifth Edition* menyatakan, sistem informasi adalah proses mengumpulkan, memproses, menganalisis, dan menyebarkan informasi untuk tujuan tertentu (Cegielski et al., 2014).

#### **2.2 Pelayanan**

Pelayanan menurut Kamus Besar Bahasa Indonesia (KBBI) adalah sebagai suatu usaha untuk membantu menyiapkan atau mengurus apa yang diperlukan orang lain. Sedangkan menurut Rahmadana, beliau menyatakan bahwa pelayanan adalah kegiatan yang dilakukan oleh seseorang atau sekelompok orang dengan landasan factor materi melalui sistem, prosedur atau metode tertentu dalam rangka usaha memenuhi kepentingan orang lain sesuai dengan haknya (Rahmadana et al., 2020).

#### **2.3 Website**

*Website* menurut Sibero pada bukunya yang berjudul *Web Programming Power Pack* menyatakan bahwa, *website* adalah suatu sistem yang berkaitan dengan dokumen digunakan sebagai media untuk menampilkan teks, gambar, multimedia, dan lainnya pada jaringan internet (Sibero et al., 2013).

#### **2.4 Basis Data (Database)**

*Database* menurut Ramakrishnan pada bukunya yang berjudul *Sistem Manajemen Database Edisi 3* menyatakan, basis data (*database*) adalah kumpulan data yang menjelaskan satu atau lebih relasi dari sebuah organisasi (Ramakrishnan & Gehrke, 2003).

#### **2.5 DBMS (Database Management System)**

DBMS Menurut Connolly pada bukunya yang berjudul *Database System: A Practical Approach to Design, Implementation, and Management, Fourth Edition* menyatakan bahwa, *Database Management System (DBMS)* adalah sebuah sistem perangkat lunak yang memungkinkan user untuk mendefinisikan, membuat, memelihara, dan mengontrol akses ke basis data (Connolly & Begg, 2005).

## 2.6 MySQL

MySQL Menurut Sibero pada bukunya yang berjudul *Web Programming Power Pack* menyatakan bahwa, MySQL atau dibaca “My Sekuel” adalah suatu RDBMS (*Relational Database Management System*) yaitu sistem perangkat lunak yang menjalankan fungsi pengolahan data (Sibero, 2013).

## 2.7 PHP (*Hypertext Preprocessor*)

PHP menurut Hikmah pada bukunya yang berjudul *Cara Cepat Membangun Website* dari Nol menyatakan bahwa, PHP tergolong sebagai bahasa pemrograman yang sangat cocok dikembangkan dalam lingkungan *web*, karena PHP bisa diletakkan pada *script* HTML atau sebaliknya. PHP dikhususkan untuk pengembangan *web* dinamis (Hikmah, 2015)

## 2.8 Framework

*Framework* menurut Raharjo pada bukunya yang berjudul *Belajar Otodidak Framework CodeIgniter* menyatakan bahwa, *Framework* atau sering *web framework* adalah suatu kumpulan kode berupa pustaka (*library*) dan alat (*tool*) yang dipadukan sedemikian rupa menjadi satu kerangka kerja (*framework*) guna memudahkan dan mempercepat proses pengembangan aplikasi *web* (Budi Raharjo, 2015). *Framework web* untuk PHP: CodeIgniter, Yii, Zend *Framework*, Laravel, Symfony, dsb. Sebagian besar dari *framework* yang tertera sebelumnya mengimplementasikan pola desain atau arsitektur *Model, View, Controller* (MVC), yang memisahkan bagian kode untuk penanganan proses bisnis dengan bagian kode untuk keperluan presentasi (tampilan).

## 2.9 CodeIgniter

CodeIgniter menurut Raharjo pada bukunya yang berjudul *Belajar Otodidak Framework CodeIgniter* menyatakan bahwa, CodeIgniter adalah *framework web* untuk bahasa pemrograman PHP yang dibuat oleh Rick Ellis pada tahun 2006, penemu dan pendiri EllisLab. EllisLab adalah suatu tim yang berdiri pada tahun 2002 dan bergerak di bidang pembuatan *software* dan *tool* untuk para pengembang *web*. Menurut Raharjo, CodeIgniter merupakan sebuah *toolkit* yang ditujukan untuk orang yang ingin membangun aplikasi *web* dengan bahasa pemrograman PHP (Raharjo, 2015).

## 2.10 UML (*Unified Modeling Language*)

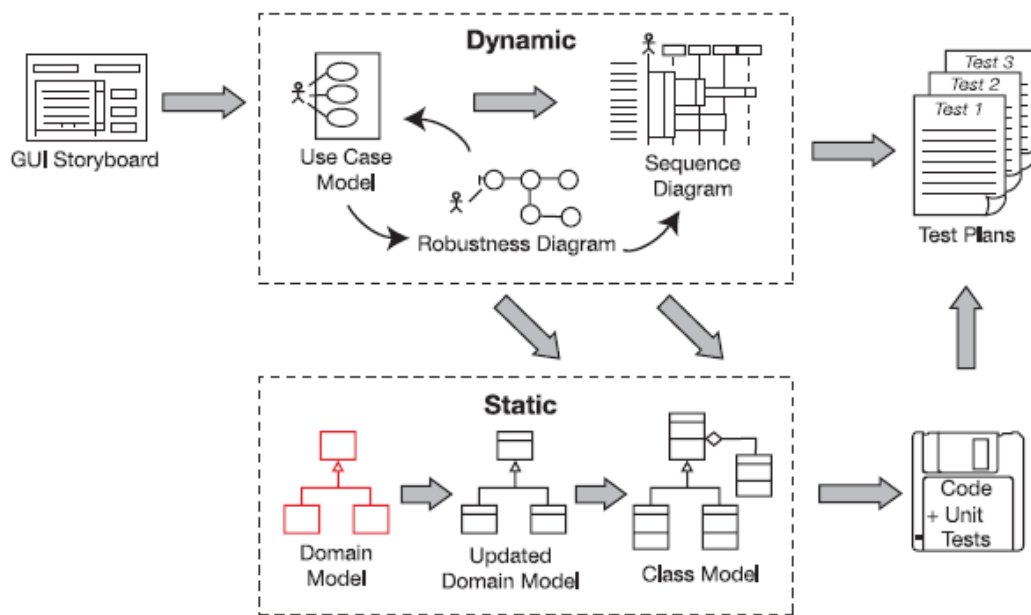
UML Menurut Shalahuddin pada bukunya yang berjudul *Rekayasa*

Perangkat Lunak Terstruktur dan Berorientasi Objek menyatakan bahwa, UML (*Unified Modeling 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 (Sukamto & Shalahuddin, 2014). UML (*Unified Modeling Language*) memiliki diagram-diagram yang digunakan dalam pembuatan aplikasi berorientasi objek, diantaranya :

1. *Use Case Diagram*, merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi tersebut.
2. *Activity Diagram*, menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.
3. *Class Diagram*, menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class* memiliki apa yang disebut atribut dan metode atau operasi.
4. *Sequence Diagram*, menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Untuk menggambarkan diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki *class* yang diinstansiasi menjadi objek itu.

### **2.11 *ICONIX Process***

*ICONIX Process* menurut Rosenberg pada bukunya yang berjudul *Use Case Driven Object Modeling with UML* menyatakan bahwa, *ICONIX Process* adalah metode perancangan atau pengembangan yang mengacu pada *use case*. *Use case* ditentukan pada awal pengembangan yang dimana *use case* tersebut menjadi dasar dalam penentuan model dari sistem yang sedang dibuat (Rosenberg & Stephens, 2007).



**Gambar 2.1** *ICONIX Process (Rosenberg & Stephens, 2007)*

*ICONIX Process* terdiri dari 4 tahap, yaitu:

1. *Requirements*
  - a. *Functional Requirements*, merupakan aktivitas dimana data-data dikumpulkan serta diolah sesuai dengan kebutuhan fungsional yang nantinya diperlukan di dalam pembangunan atau pembangunan perangkat lunak. Hal ini menjadi modal dalam pembangunan atau pembangunan perangkat lunak dimana semua kebutuhan mulai dari kebutuhan fungsional maupun non fungsional dianalisa dan diolah sehingga menjadi satu bagian kebutuhan terhadap pembangunan atau pembangunan perangkat lunak.
  - b. *Domain Modeling*, merupakan tahapan pada bagian statis UML dimana data-data yang didapatkan berasal dari kebutuhan fungsional maupun non fungsional yang diekstrak menjadi beberapa bagian untuk dapat dihubungkan sesuai kebutuhan perangkat lunak.
  - c. *GUI Storyboard*, merupakan tahapan pembangunan tampilan antarmuka pengguna.
  - d. *Use Case Modeling*, merupakan aktivitas dimana bagian dari tahapan *ICONIX Process* dilakukan sebuah pengidentifikasian

terhadap aktor serta aktivitas kegiatan proses bisnis yang sedang berjalan sehingga memaparkan terhadap apa saja kegiatan yang dilakukan pengguna yang kaitannya terhadap tanggapan sistem.

2. *Analysis and Preliminary Design*

- a. *Robustness Analysis*, merupakan pengembangan dari tahapan analisa kemudian dilakukan proses tahapan desain.
- b. *Update Domain Model*, merupakan tahapan pengembangan dengan menghilangkan beberapa *class* yang *redundant* atau ambigu serta menambahkan beberapa *class* yang tidak ada serta atribut di dalam pemodelan domain jika terdapat objek baru yang muncul.

3. *Detailed Design*

- a. *Sequence Diagram*, merupakan tahapan pemodelan *Sequence Diagram* dimana disusun terhadap diagram alir yang dilanjutkan dari tahapan *Robustness Diagram*. Berisi aktivitas yang dilakukan oleh pengguna ketika akan berinteraksi langsung dengan sistem yang dirancang atau dibangun.
- b. *Update Domain Model*, merupakan tahapan penambahan model yang didasarkan pada hasil pengembangan *Sequence Diagram* yang disesuaikan terhadap kebutuhan perangkat lunak.

4. *Implementation*

- a. *Coding/Unit Testing*, merupakan tahapan dimulainya implementasi sistem dimana dilakukan proses pembuatan *coding* atau proses penerjemahan setelah pengembangan model yang dirancang.
- b. *Integration and Scenario Testing*, merupakan tahapan yang dilanjutkan setelah proses pembuatan *coding* atau proses penerjemahan ke dalam tahapan pengujian (*Testing*). Tujuannya, agar pengembangan model sistem yang telah dibuat dapat diukur apakah telah sesuai dengan yang diinginkan oleh pengguna maupun pengembang. Pengujian (*Testing*) dapat dilakukan dengan *white box Testing* atau *Black Box Testing*.

**2.12 ERD (*Entity Relationship Diagram*)**

ERD menurut Kusri pada bukunya yang berjudul Tuntunan Praktis

membangun sistem informasi Akuntansi Dengan Visual Basic dan Microsoft SQL Server menyatakan bahwa, ERD merupakan notasi grafis dalam pemodelan data konseptual yang mendeskripsikan hubungan antardata, karena hal ini relatif kompleks (Kusrini & Koniyo, 2007). Untuk itu ERD dibagi menjadi dua jenis model, yaitu :

a. *Conceptual Data Model (CDM)*

CDM merupakan konsep yang berkaitan dengan pandangan pemakai terhadap data yang disimpan dalam basis data. CDM dibuat sudah dalam bentuk tabel-tabel tanpa tipe data yang menggambarkan relasi antar tabel untuk keperluan implementasi ke basis data. CDM merupakan hasil penjabaran lebih lanjut dari ERD (Rosa, 2013).

b. *Physical Data Model (PDM)*

PDM merupakan model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data. Setiap tabel mempunyai sejumlah kolom di mana setiap kolom memiliki nama yang unik beserta tipe datanya. PDM merupakan konsep yang menerangkan detail dari bagaimana data di simpan di dalam basis data. PDM sudah merupakan bentuk fisik perancangan basis data yang sudah siap diimplementasikan ke dalam DBMS (Rosa, 2013).