

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

A. Sistem

Menurut Rusdiana dan Irfan (2014:29) sistem merupakan kumpulan dari beberapa bagian yang memiliki keterkaitan dan saling bekerja sama serta membentuk suatu kesatuan untuk mencapai tujuan dari sistem tersebut.

Menurut Rusmawan (2019:28) “sistem adalah suatu jaringan atau komponen-komponen yang membentuk satu kesatuan yang terkumpul untuk menyelesaikan suatu tujuan tertentu.”

Menurut Mulyani (2017:218) “sistem berorientasi objek adalah suatu sistem yang kompleks dipecahkan berdasarkan komponen-komponen yang membentuknya, kemudian komponen tersebut dikategorikan berdasarkan objek”, sedangkan *Object Oriented Programming* (OOP) adalah metode yang diimplementasikan dengan pengorganisasian atas kerjasama dari sekumpulan objek, dimana masing-masing objek tersebut adalah sebuah *instance* dari kelas dan semua kelas adalah anggota hirarki yang direlasikan melalui pewaris dari class utamanya.

B. Konsep Dasar Web

1. *Internet*

Menurut Kadir (2014:306) “*Internet* merupakan contoh jaringan terbesar yang menghubungkan jutaan komputer yang tersebar di seluruh penjuru dunia dan tidak terikat pada suatu organisasi pun.”

Internet berasal dari kata *interconnected-networking*. Secara umum *internet* dapat diartikan sebagai jaringan global yang menghubungkan suatu jaringan (*network*) untuk mendapatkan informasi dan komunikasi dengan mudah dan bebas di *internet* tanpa ada batas.

2. Website

Menurut Sidik & Pohan (2014:6) “*World Wide Web* merupakan suatu layanan penyajian informasi di *internet* dengan menggunakan HTML. *Web* identik dengan *internet*, karena kepopulerannya saat ini, *web* sudah menjadi *interface* aplikasi untuk melakukan transaksi dan sajian informasi yang lengkap dari seluruh dunia.”

3. Web Server

Menurut Kadir (2014:329) “*Web server* adalah aplikasi yang berguna untuk melayani permintaan halaman *web* dan mengirimkan dokumen *HTML* yang diminta ke klien.”

4. Web Browser

Menurut Kadir (2014:186) “*Web browser* atau biasa disebut *browser* adalah perangkat lunak yang berguna untuk mengakses informasi *web* ataupun untuk melakukan transaksi via *web*”.

C. Basis Data

Menurut Sutanta dalam (Rusmawan:2019:39) berpendapat bahwa “basis data dipahami sebagai suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (kalaupun ada maka kerangkapan data tersebut harus seminimal mungkin dan terkontrol (*controlled redundancy*), data disimpan dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali, data dapat digunakan satu atau lebih program-program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan dengan program yang akan menggunakannya, data disimpan sedemikian rupa sehingga proses penambahan, pengembalian, dan modifikasi data dapat dilakukan dengan mudah dan terkontrol.” Aplikasi basis data yang digunakan dalam perancangan *website* ini antara lain :

1. MySQL (*My Structured Query Language*)

Menurut Firmansyah & Udi (2018:185) “MySQL adalah *database Server* yang sangat populer dan banyak digunakan untuk menangani data yang disajikan di halaman *web*.”

Menurut Rusmawan (2019:97) “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia.”

2. XAMPP

Menurut Wicaksono dalam (Zamaludin, Yusnaeni, & Amelia, 2016:21) “Xampp adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolah data MYSQL di komputer lokal”.

3. PHPMYAdmin

Menurut Madcoms (2016:148) “*PhpMyAdmin* adalah sebuah aplikasi *open source* memudahkan manajemen *MySQL*”. *PhpMyAdmin* sebuah perangkat lunak berbasis *web* yang sangat populer untuk administrasi basis data *MySQL*.

Menurut Atmoko & Arizona (2016:39) “*PhpMyAdmin* adalah *MySQL client* yang berupa *web* dan umumnya tersedia di *server* PHP seperti XAMPP maupun *server* komersial lainnya *MySQL client* yang paling sederhana adalah berupa *command line*, namun ternyata malah jarang diperoleh di *server* hosting-an karena pada umumnya *server* yang diperjual belikan oleh *reseller* berupa *sharedshot*”.

D. Model Pengembangan Perangkat Lunak

Menurut Shalahuddin dan Sukamto (2018:28) “Model SDLC air terjun (*waterfall*) sering disebut model sekuensial linier (*sequential linear*) atau hidup klasik (*classic life cycle*)”. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau berurutan dimulai dari analisis, desain, pengkodean, dan pengujian, yang terbagi menjadi empat tahapan, yaitu:

1. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi perangkat lunak pada tahap ini perlu didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentransaksi 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 ke 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 dan fungsional dan memastikan bahwa bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai yang diinginkan menggunakan *black box*.

2.2. Teori Pendukung

A. *Entity Relationship Diagram*

Menurut Rusmawan (2019:63) menyimpulkan bahwa "ERD merupakan gambaran grafis dari suatu model data yang menyertakan deskripsi *detail* dari seluruh entitas (*entity*), hubungan (*relationship*), dan batasan (*constraint*) untuk memenuhi kebutuhan sistem analis dalam menyelesaikan pengembangan sebuah sistem."

B. *Logical Record Structure*

Menurut Puspitasari (2016:229) "LRS merupakan sebuah model *system* yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan pemodelan tertentu dalam kaitannya dengan konversi ke LRS."

C. *Unified Modelling Language*

Menurut Maharani (2018:44) "UML (*Unified Modelling Language*) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk masalah visualisasi, merancang dan mendokumentasikan sistem piranti lunak."

Terdapat beberapa diagram UML yang sering digunakan dalam pengembangan sebuah sistem, yaitu:

1. Menurut Maharani (2018:45) "*Use Case Diagram* adalah diagram yang menggambarkan interaksi antara pengguna (*actor*) dengan sistem informasi yang akan dibuat. *Use case* menggambarkan siapa saja *actor* yang terlibat dan fungsi apa saja yang dapat digunakan *actor* pada sistem informasi tersebut."

2. Menurut Maharani (2018:49) “*Activity Diagram* atau Diagram Aktivitas menggambarkan urutan kegiatan atau urutan aktivitas dari sebuah sistem. Tujuan dibuatnya *Activity Diagram* adalah untuk memudahkan dalam memahami proses bisnis sistem.”
3. Menurut Maharani (2018:53) “*Sequence Diagram* menggambarkan interaksi antar obyek dalam sistem. *Sequence Diagram* digunakan untuk menggambarkan skenario pada *use case*. Jumlah *Sequence Diagram* harus sama dengan jumlah *Use Case*.”
4. Menurut Maharani (2018:56) “*Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek serta hubungan satu sama lain seperti *inheritance*, *association* dan lain-lain.”

