

## **BAB II**

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

#### **2.1 Tinjauan Pustaka**

Pembuatan dan penggunaan fitur admin pada suatu sistem seperti ini telah banyak dilakukan sebelumnya dalam sektor perindustrian, yang secara umum memiliki tujuan yang sama yaitu mengatur agar penggunaan sistem dapat berjalan dengan lancar, seperti pengaturan otorisasi pengguna serta memasukkan dan memperbaharui data yang terdapat pada sistem.

(Hartanti, 2003) dalam Penelitian dengan judul Analisis Sistem Informasi Kepegawaian Pemerintah Kotamadya Jakarta Barat Berbasis Web memiliki kesimpulan sebagai berikut:

Sistem informasi kepegawaian adalah rancangan suatu sistem yang terdiri dari kumpulan data yang diorganisir dan berkaitan antara satu tabel dengan tabel lainnya dalam sebuah komputer dan sekumpulan program yang memungkinkan beberapa pemakai untuk dapat mengaksesnya dan memanipulasi file (tabel-tabel) tersebut yang bertujuan untuk membantu mengelola dan mengatur arus orang-orang (pegawai) yang ada dalam lingkungan organisasi juga dapat digunakan untuk membaca informasi yang dibutuhkan

(Rokhmah, 2005) telah melakukan penelitian dengan judul Sistem Informasi Kepegawaian di Dinas Pendidikan Kabupaten Lamongan Berbasis Web dengan studi kasus Universitas Trunojoyo Madura 3. Sistem Informasi dibuat untuk mempermudah dalam pengelolaan dan penyimpanan data maka dapat menghasilkan suatu informasi yang tepat dan akurat. Adanya sistem informasi yang tepat dan akurat dapat mengurangi terjadinya kesalahan yang tidak diinginkan sehingga dapat meningkatkan kinerja yang lebih efisien dan kecepatan operasional instansi. Seperti halnya simpeg pada suatu dinas pendidikan yang memiliki fungsi untuk meningkatkan penyelenggaraan tugas dan fungsi pendidikan. Di mana salah satunya Dinas Pendidikan memiliki banyak data kepegawaian yang perlu untuk disimpan dan diolah.

(Septyana, 2011) dalam Penelitian pertama dengan judul Sistem Informasi Kepegawaian pada Dinas Perhubungan Komunikasi dan Informatika Kabupaten Cilacap memiliki kesimpulan sebagai berikut :

SIMPEG yang merupakan singkatan dari Sistem Informasi Kepegawaian/ Sistem Informasi Manajemen Kepegawaian adalah suatu totalitas yang terpadu terdiri atas perangkat pengolah meliputi pengumpul, prosedur, tenaga pengolah dan perangkat lunak. Perangkat penyimpan meliputi pusat data dan bank data serta perangkat komunikasi yang saling berkaitan, berketergantungan dan saling menentukan dalam rangka penyediaan informasi di bidang kepegawaian. Dengan berdasar pada “SK Keputusan Menteri Dalam Negeri Nomor 17 Tahun 2000 tentang Sistem Informasi Manajemen Kepegawaian Departemen Dalam Negeri dan Pemerintah Daerah”.

Berdasarkan pengujian dan analisis yang telah dilakukan oleh para peneliti, beberapa peneliti di atas tersebut membangun aplikasi perancangan sistem aplikasi berbasis web untuk mendukung kinerja pemerintahan daerah agar lebih efektif dan efisien.

Namun pada penelitian di atas masih terdapat beberapa kekurangan yaitu belum adanya fitur informasi atau *update info*, secara keseluruhan penelitian pada aplikasi di atas membahas tentang pengelolaan data dan penyimpanan data, namun belum memiliki fitur *update info*. Kelebihan dari aplikasi web admin yang penulis kembangkan adalah mempunyai fitur *update info* yang bertujuan untuk menginformasikan berita terbaru yang ada di kantor, khususnya di PT Indocement Tunggal Prakarsa Tbk. sehingga mempermudah dalam penyampaian informasi kepada karyawan.

## **2.2 Landasan Teori**

### **2.2.1 Sistem Administrator**

Sistem Administrator adalah seseorang yang bekerja untuk memelihara dan mengoperasikan sebuah sistem komputer atau jaringan yang berjalan setiap harinya pada suatu organisasi/perusahaan. Menurut Nugraha Pengestu dan Ryan R. Adhisa (2013) Sistem Administrator (dikenal juga sebagai admin, administrator, sysadmin,

site admin, dll) merupakan profesi yang memiliki tugas untuk melakukan administrasi terhadap sistem, pemeliharaan sistem, memiliki kewenangan mengatur hak akses terhadap sistem, serta hal – hal lain yang berhubungan dengan pengaturan operasional sebuah sistem.

Tugas utama seorang sistem administrator adalah memastikan sistem tetap berjalan lancar dalam memberikan pelayanan kepada penggunanya.

### **2.2.2 UML**

*Unified Modelling Language* (UML) adalah alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual (Braun, et. al. 2001). Selain itu merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek (Whitten, et. al. 2004).

Penggunaan UML bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem didalam aplikasi, mendokumentasikan hasil analisa dan desain serta untuk menggambarkan sebuah sistem *software*. Model UML yang dipakai dalam pengembangan aplikasi ini antara lain *Use Case Diagram* dan *Activity Diagram*.

### **2.2.3 Diagram-Diagram UML**

Menurut situs resminya, diagram UML dapat dibedakan atas *Structure diagram* dan *Behavior diagram*. *Structure diagram* adalah diagram yang menampilkan susunan statis sistem dan bagian – bagian pada implementasi dan menunjukkan bagaimana bagian – bagian itu saling terkait satu sama lain. Sedangkan *Behavior diagram* menunjukkan perilaku dinamis objek dalam suatu sistem yang dapat digambarkan sebagai rangkaian perubahan sistem dari waktu ke waktu.

#### **1. Use Case Diagram**

Use Case menurut (Flower, 2005, p141) adalah teknik untuk merekam persyaratan fungsional sebuah sistem. Use Case mendeskripsikan interaksi tipikal antara para pengguna sistem dengan

sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan. Use Case Diagram menampilkan aktor mana yang menggunakan use case mana, use case mana yang memasukkan use case lain dan hubungan antara aktor dan use case. Use case diagram menggambarkan interaksi antara sistem dengan sistem eksternal dan pengguna. Dengan kata lain, secara grafis menggambarkan siapa yang menggunakan sistem dan dengan cara apa pengguna mengharapkan untuk berinteraksi dengan sistem. Pemodelan use case awalnya disusun oleh Dr. Ivar Jacobson pada tahun 1986 dan menjadi populer setelah beliau menerbitkan buku, *Object-Oriented Software Engineering*, pada tahun 1992. Dr. Jacobson menggunakan pemodelan use case sebagai kerangka kerja untuk metodologi objectory-nya dengan sukses digunakannya untuk mengembangkan sistem informasi berorientasi-objek.

## 2. Activity Diagram

*Activity diagram* menurut (Fowler, 2005 : 163) adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. Node pada sebuah *activity diagram* disebut sebagai *action*, sehingga *Activity diagram* menampilkan sebuah *activity* yang tersusun dari *action*. *Activity diagram* Menunjukkan kegiatan dan perubahan dari satu aktivitas ke aktivitas lainnya dengan peristiwa yang terjadi di beberapa bagian dari *siste*.

## 3. Class Diagram

*Class Diagram* merupakan salah satu *Structure diagram* yang menggambarkan struktur sistem yang dirancang. Subsistem atau komponen digunakan sebagai kelas dan antarmuka terkait. *Class diagram* menurut (Munawar, 2005 : 28) merupakan himpunan dari objek-objek yang sejenis. Sebuah objek memiliki keadaan sesaat (*state*) dan perilaku (*behavior*). *State* sebuah objek adalah kondisi objek tersebut yang dinyatakan dalam *attribute/properties*. Sedangkan perilaku suatu objek mendefinisikan bagaimana sebuah objek bertindak/ beraksi dan memberikan reaksi.

*Class* diagram menggambarkan relasi atau hubungan antar *class* dari sebuah sistem. Berikut ini beberapa gambaran relasi yang ada dalam *Class* diagram.

a. Asosiasi

Asosiasi merupakan hubungan antar *class* yang statis. *Class* yang mempunyai relasi asosiasi menggunakan *class* lain sebagai atribut pada dirinya. Asosiasi digambarkan dengan garis panah lurus.

b. Agregasi

Agregasi merupakan relasi yang membuat *class* yang saling berelasi terikat satu sama lain namun tidak terlalu bergantung.

c. Komposisi

Komposisi merupakan relasi agregasi dengan mengikat satu sama lain dengan ikatan yang sangat kuat dan saling bergantung satu sama lain.

d. *Dependency*

Merupakan hubungan antar-*class* di mana *class* yang memiliki relasi *dependency* menggunakan *class* lain sebagai attribute pada method

#### 4. Sequence Diagram

Sequence diagram menurut Munawar (2005 : 187) adalah grafik dua dimensi dimana obyek ditunjukkan dalam dimensi horizontal, sedangkan lifeline ditunjukkan dalam dimensi vertikal.

#### 2.2.4 Entity Relationship Diagram (ERD)

Menurut (Sutanta, 2011:91), “Entity Relationship Diagram (ERD) merupakan suatu model data yang dikembangkan berdasarkan objek.” Entity Relationship Diagram (ERD) digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis.

Bagi perancang atau analis sistem, Entity Relationship Diagram (ERD) berguna untuk memodelkan sistem basis data yang nantinya akan dikembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan

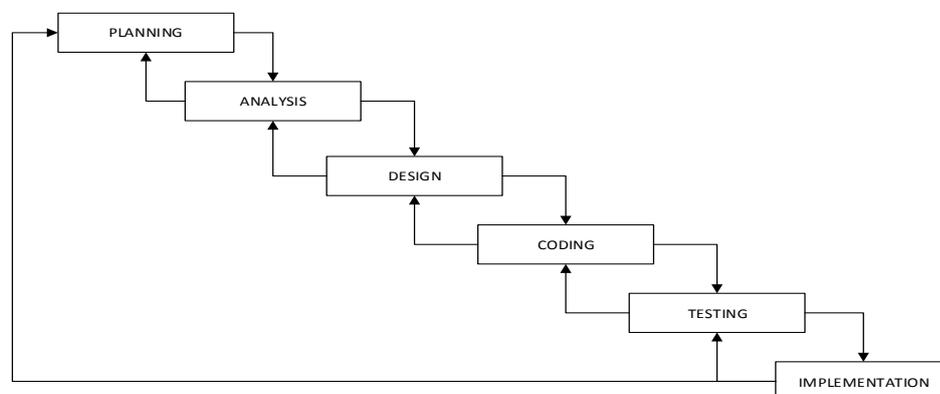
analisis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasiaan antardata didalamnya.

### 2.2.5 Metode Waterfall dalam Software Development Life Cycle

SDLC (*Software Development Life Cycle*) merupakan sebuah siklus pengembangan perangkat lunak yang terdiri dari beberapa tahapan-tahapan penting dalam membangun perangkat lunak yang dilihat dari segi pengembangannya. Tahapan-tahapan tersebut diantaranya : perencanaan (*planning*), analisis (*analysis*), desain (*design*), implementasi (*implementation*), dan uji coba (*testing*). Selain untuk proses pembuatan, SDLC juga penting untuk proses *maintenance* (pemeliharaan) *software*.

Salah satu metode yang terdapat pada model SDLC adalah metode *Waterfall*. Model ini Pertama kali diperkenalkan oleh Winston Royce pada tahun 1970. Model ini merupakan model klasik yang sederhana dengan aliran sistem yang linear, yang artinya suatu tahapan harus selesai terlebih dahulu, sebelum memulai tahapan yang lain. Output dari setiap tahap menjadi *input* bagi tahap berikutnya.

Salah satu referensi yang terkenal mengenai metode waterfall adalah metode yang dikemukakan oleh Yousef Bassil. Menurut (Bassil, 2011), ” Model *Waterfall* SDLC adalah proses pengembangan perangkat lunak yang berurutan (*sequential*) dimana prosesnya dari atas ke bawah (seperti air terjun) melalui tahapan-tahapan yang harus dijalankan untuk keberhasilan pembuatan perangkat lunak”. Untuk lebih memahami metode waterfall menurut Bassil, dapat dilihat pada gambar 2.1.



**Gambar 2.1** Metode *Waterfall*

### 2.2.6 NetBeans IDE

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

Netbeans merupakan software development yang Open Source, dengan kata lain software ini di bawah pengembangan bersama, 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.

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.

The *NetBeans* IDE adalah sebuah lingkungan pengembangan - sebuah kaskas untuk pemrogram menulis, mengompilasi, mencari kesalahan dan menyebarkan program. Netbeans IDE ditulis dalam Java - namun dapat mendukung bahasa pemrograman lain. Terdapat banyak modul untuk memperluas Netbeans IDE. Netbeans IDE adalah sebuah produk bebas dengan tanpa batasan bagaimana digunakan. NetBeans IDE mendukung pengembangan semua tipe 16 aplikasi Java (*J2SE, web, EJB*, dan aplikasi mobile). Fitur lainnya adalah sistem proyek berbasis *Ant, kontrol versi, dan refactoring*.

### 2.2.7 Java

Java merupakan bahasa pemrograman yang disusun oleh James Gosling yang dibantu oleh rekan-rekannya di suatu perusahaan perangkat lunak yang bernama Sun Microsystems, pada tahun 1991. Bahasa pemrograman ini mula-mula diinisialisasi dengan nama "Oak", namun pada tahun 1995 diganti namanya menjadi "Java".

Menurut definisi Sun Microsystem, di dalam buku (Shalahuddin & Rosa, 2010:1) Java adalah nama sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer yang berdiri sendiri (standalone) ataupun pada lingkungan jaringan.

Java berdiri di atas sebuah mesin penterjemah (interpreter) yang diberi nama Java Virtual Machine (JVM). JVM inilah yang akan membaca kode bit (bytecode) dalam file .class dari suatu program sebagai representasi langsung program yang berisi bahasa mesin. Oleh karena itu bahasa Java disebut sebagai bahasa pemrograman yang portable karena dapat dijalankan pada berbagai sistem operasi, asalkan pada sistem operasi tersebut terdapat JVM. Alasan utama pembentukan bahasa Java adalah untuk membuat aplikasi-aplikasi yang dapat diletakkan di berbagai macam perangkat elektronik, sehingga Java harus bersifat tidak bergantung pada platform (platform independent). Itulah yang menyebabkan dalam dunia pemrograman Java dikenal adanya istilah „write once, run everywhere“, yang berarti kode program hanya ditulis sekali, namun dapat dijalankan di bawah kumpulan pustaka (platform) manapun, tanpa harus melakukan perubahan kode program.

### **2.2.8 JSP (Java Server Page)**

Menurut (Foenadioen, 2007) JSP (Java Server Pages) adalah suatu bahasa pemrograman web (scripting) yang bersifat server side yang menggabungkan HTML dengan scripting tag dan program java. Suatu dokumen JSP ditandai dengan berkas ekstensi. JSP Java Server Pages (JSP) merupakan sebuah teknologi servletbased yang digunakan pada web tier untuk menghadirkan dinamik dan statik content. Java Server Pages (JSP) merupakan text-based dan kebanyakan berisi template text HTML yang digabungkan dengan spesifik tags dynamic content”.

### **2.2.9 CSS (Cascading Style Sheet )**

Menurut Shalahuddin dan A.S Rosa (2010), *CSS atau Cascading Style Sheets* adalah suatu fasilitas untuk mempermudah pemeliharaan sebuah halaman web, dengan menggunakan CSS sebuah halaman web dapat diubah tampilannya tanpa harus mengubah dokumen *HTML-nya*.

### 2.2.10 JavaScript

Menurut (Shalahuddin & A.SRosa, 2010) *JavaScript* adalah bahasa script (bahasa pemrograman yang dapat memegang kontrol aplikasi) yang berbasis pada pemrograman *Java*, namun *JavaScript* bukanlah bagian teknologi *Java* dari Sun. *Java* merupakan bahasa pemrograman berorientasi objek murni sedangkan *JavaScript* digunakan secara prosedural.

### 2.2.11 MySQL

Beberapa pengertian MySQL menurut para ahli:

- a. Menurut (Raharjo, 2011:21), “MySQL merupakan RDBMS (atau server database) yang mengelola database dengan cepat menampung dalam jumlah sangat besar dan dapat di akses oleh banyak user”.
- b. Menurut (Kadir, 2008:2), “MySQL adalah sebuah software open source yang digunakan untuk membuat sebuah database.” Berdasarkan pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa MySQL adalah suatu software atau program yang digunakan untuk membuat sebuah database yang bersifat open source.

### 2.2.12 Pengujian *Black Box*

Pengujian sistem menyajikan anomali yang menarik bagi rekayasa perangkat lunak pada proses perangkat lunak, perekrayasa berusaha membangun perangkat lunak dari konsep abstrak ke implementasi yang dapat dilihat, baru kemudian dilakukan pengujian.

Menurut (Pressman, 2002) *Black Box Testing* atau Pengujian Kotak Hitam, berfokus pada persyaratan fungsional dari perangkat lunak. Artinya, teknik *Black Box Testing* memungkinkan untuk mendapatkan set kondisi masukan yang sepenuhnya akan melaksanakan semua persyaratan fungsional untuk suatu program. *Black box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineer* untuk memperoleh *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program.