

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Pengertian Sertifikat Kapal**

Sertifikat kapal adalah syarat atau sistem manajemen keselamatan yang bertujuan untuk menjamin kelayakan operasional kapal dengan aman serta legalitas kapal yang akan berlayar dan mengerjakan sebuah proyek. Kapal Indonesia (Kapal Berbendera Indonesia) yang dinyatakan memenuhi persyaratan keselamatan akan diberikan Sertifikat Keselamatan oleh Menteri. Dalam peraturan Menteri Perhubungan No.17 Tahun 2008 Tentang Sertifikat Keselamatan Kapal diberikan kepada semua jenis kapal ukuran GT 7 (Tujuh *Gross Tonnage*) atau lebih, kecuali kapal perang, kapal negara, dan kapal yang digunakan untuk keperluan olahraga.

Perusahaan yang telah memenuhi persyaratan akan diterbitkan Dokumen Kesesuaian atau *Document of Compliance* (DOC) dan juga akan diterbitkan Sertifikat Manajemen Keselamatan atau *Safety Management Certificate* (SMC). Perusahaan dan kapalnya yang tidak dapat memenuhi persyaratan ISM Code akan menghadapi kesulitan dalam operasionalnya, baik diperairan internasional maupun domestik.

##### **2.1.1 Tujuan Penggunaan Dokumen Sertifikat Kapal**

Diberlakukan adanya dokumen sertifikat kapal bertujuan untuk mencegah terjadinya kecelakaan kapal, manusia, muatan barang/cargo dan harta benda yang berlebihan serta mencegah terjadinya pencemaran lingkungan laut, maka pemerintah mengeluarkan peraturan tentang manajemen keselamatan kapal

& perlindungan lingkungan laut yang dikenal dengan Peraturan International Safety Management (ISM Code) yang juga dikonsolidasikan dalam SOLAS Convention.

### 2.1.2 Jenis-Jenis Dokumen Sertifikat Kapal

Berdasarkan hasil wawancara di PT. Rukina Sukses Abadi dengan objek wawancara bagian admin kapal dan officer kantor bagian dokumen sertifikat kapal yaitu Bapak Rhesa Pradana, adapun jenis-jenis dokumen sertifikat kapal yang harus ada disetiap kapal adalah sebagai berikut :

1. Surat Laut (*Certificate Of Nationality*).
2. Surat Ukur (*International Tonnage Certificate*).
3. Sertifikat keselamatan konstruksi kapal barang (*Cargo Ship Safety Construction Certificate*).
4. Sertifikat keselamatan perlengkapan kapal barang (*Cargo Ship Safety Equipment Certificate*).
5. Sertifikat keselamatan radio kapal barang (*Cargo Ship Safety Radio Certificate*).
6. Sertifikat internasional pencegahan pencemaran oleh minyak (IOPP).
7. Sertifikat internasional pencegahan pencemaran oleh kotoran (ISPP).
8. Sertifikat internasional pencegahan pencemaran oleh udara (IAPP).
9. *Fire Extinguisher Certificate*.
10. *Inspection Certificate*.
11. *International Load Line Exemption Certificate*.
12. Sertifikat Klasifikasi Sementara.
13. Sertifikat Garis Muat (*Load Line Certificate*).

14. *Damage Control Plans and Booklets*, SOLAS 1974, regulations II-1 / 2 3, 2 4, dan MSC.
15. *Continous Synopsis Record (CSR)*.
16. *Cargo Securing Manual*.
17. *Garbage Management Plan*.
18. *Garbage Record Book*.
19. *Oil Record Book*.
20. *Document of Compliance*.
21. *Fire Control Plan/booklet*.
22. *Intact Stability Booklet*, SOLAS 1974, regulations II – 1 / 2 2 dan 1988 LL Protocol, regulations 10.
23. *Ship Security Plan and Associated Records*.
24. *On Board Training and Drills Record*.

## **2.2 Website**

Website atau dalam bahasa Indonesia disebut sebagai situs web merupakan kumpulan berbagai halaman web yang ditulis dengan bahasa HTML yang kemudian bisa dilihat menggunakan software yang disebut web browser (Zaki, 1999: 127). Halaman web bisa berisi file seperti gambar, video, dan sebagainya. Agar dapat diakses, halaman web harus diletakkan di server web untuk kemudian bisa diakses melalui peranti seperti internet, jaringan, dan sebagainya.

### 2.3 Aplikasi

Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, permainan, pelayanan masyarakat, periklanan, atau semua proses yang hampir manusia lakukan (Pramana 2005, 19).

### 2.4 PHP

Menurut Kadir (2008), PHP (akronim dari PHP *Hypertext Preprocessor*) yang merupakan bahasa pemrograman berbasis web yang memiliki kemampuan untuk memproses data dinamis. PHP dikatakan sebagai sebuah *server-side embedded script language* artinya sintaks-sintaks dan perintah yang kita berikan akan sepenuhnya dijalankan oleh *server* tetapi disertakan pada halaman HTML biasa. Aplikasi-aplikasi yang dibangun oleh PHP pada umumnya akan memberikan hasil pada web browser, tetapi prosesnya secara keseluruhan dijalankan di *server*. Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*. Dalam hal ini *client* menggunakan kode-kode PHP untuk mengirimkan permintaan ke *server* (dapat dilihat pada gambar dibawah). Ketika menggunakan PHP sebagai *server-side embedded script language* maka *server* akan melakukan hal-hal sebagai berikut :

1. Membaca permintaan dari *client/browser*.
2. Mencari halaman/page di *server*.
3. Melakukan instruksi yang diberikan oleh PHP untuk melakukan modifikasi pada halaman/page.

Mengirim kembali halaman tersebut kepada *client* melalui internet atau intranet.

## 2.5 *Server*

Menurut Sutisna (2007), *Server* adalah sebuah sistem komputer yang menyediakan jenis layanan tertentu dalam sebuah jaringan komputer. *Server* didukung dengan prosesor yang bersifat *scalable* dan RAM yang besar, juga dilengkapi dengan sistem operasi khusus, yang disebut sebagai sistem operasi jaringan atau *network operating system*. *Server* juga menjalankan perangkat lunak administratif yang mengontrol akses terhadap jaringan dan sumber daya yang terdapat didalamnya, seperti halnya berkas atau alat pencetak (printer), dan memberikan akses kepada workstation anggota jaringan. Adapun jenis dari *server* adalah sebagai berikut :

### 1. *Server* Aplikasi

*Server* yang digunakan untuk menyimpan berbagai macam aplikasi yang dapat diakses oleh *client*, *server* data sendiri digunakan untuk menyimpan data baik yang digunakan *client* secara langsung maupun data yang diproses oleh *server* aplikasi.

### 2. *Server* Data

Berfungsi untuk mengatur lalu lintas di jaringan melalui pengaturan *proxy*. Orang awam lebih mengenal *proxy server* untuk mengkoneksikan komputer *client* ke Internet.

### 3. *Server Proxy*

*Proxy Server* adalah sebuah komputer *server* yang dalam perkembangannya saat ini, dapat berupa system operasi tersendiri yaitu sistem operasi fungsional khusus *Proxy* (seperti *Untangle*, dll.); atau dapat pula berupa program aplikasi yang diinstallkan pada komputer *server* tersebut (seperti *Squid*, *Kerio Winroute Firewall*, *WinGate* dll.). *Proxy Server* memiliki banyak fungsi di dalamnya. Akan tetapi fungsi utama (secara umum) dari *server* ini adalah untuk menjembatani (biasa disebut *gateway*) dan menangani setiap request (permintaan akses) terhadap konten-konten yang berasal baik dari dalam maupun luar jaringan lokal.

### 2.6 Konsep Pemodelan Sistem

*Flowchart* adalah teknik penyusunan instruksi untuk penulisan program computer terstruktur dengan menggunakan gambar-gambar/symbol-symbol. Tujuan utama dari penggunaan *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas dengan menggunakan symbol-symbol standar (Jogiyanto, 1990).

*Data Flow Diagram (DFD)* adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi (Jogiyanto, 1990).

## 2.7 Konsep Basis Data

Basis data adalah koleksi dari data-data yang terkait secara logis dan deskripsi dari data-data tersebut, yang dirancang untuk memenuhi kebutuhan informasi dari suatu organisasi (Yourdon, 1989).

Teknik *Entity Relationship Diagram* ERD merupakan notasi grafis dalam pemodelan data konseptual yang mendeskripsikan hubungan antara penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, karena hal ini relatif kompleks (Yourdon, 1989).

Teknik normalisasi merupakan teknik analisis data yang mengorganisasikan atribut-atribut data dengan cara mengelompokkan sehingga terbentuk entitas yang nonredundan, stabil, dan fleksibel (Yourdon, 1989).

*Structured Query Language* (SQL) adalah bahasa yang bersifat request oriented dan bersifat non-prosedural sehingga lebih mudah untuk dipelajari karena sintaksis yang digunakan hampir menyerupai bahasa yang digunakan oleh manusia untuk berkomunikasi (Yourdon, 1989).. Selain itu juga, SQL bersifat non case sensitif. Banyak vendor pembuat DBMS (Database Management Sistem) yang saat ini menggunakan SQL sebagai standarisasi dalam produk mereka, seperti ORACLE, Microsoft SQL Server, PostgreSQL, dan MySQL(Yourdon, 1989).

## 2.8 Perangkat Lunak Yang Digunakan

MySQL adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user serta menggunakan perintah standar SQL (Structured Query Language). Dengan sifatnya yang open

source, memungkinkan juga user untuk melakukan modifikasi pada source code-nya untuk memenuhi kebutuhan spesifik mereka sendiri (Kadir, 2005).

XAMPP merupakan aplikasi server yang menggabungkan beberapa aplikasi server yang biasa digunakan di web server. Berikut beberapa komponen-komponen yang terdapat pada XAMPP, yaitu : Apache (web server), MySQL (database server), Filezilla FTP server, Mercury Mail (mail server), phpMyAdmin (web-based interface MySQL) (Sutisna, 2007).

## **2.9 SDLC**

System Development Lyfe Cycle (SDLC) adalah keseluruhan proses dalam membangun sistem melalui beberapa langkah. Metode pengembangan perangkat lunak dikenal dengan istilah SDLC (Software Development Life Cycle). Metodologi ini menjadi perhatian sangat istimewa pada proses rekayasa perangkat lunak. Karena dengan metodologi SDLC yang digunakan akan sangat menentukan sukses tidaknya proyek software.

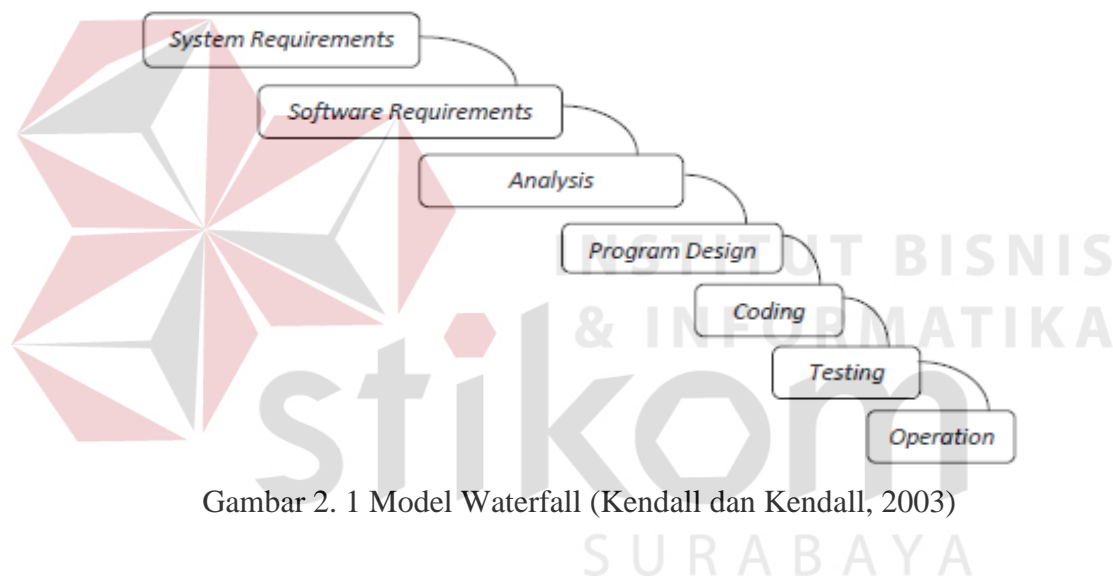
### **2.9.1 Waterfall**

Menurut Kendall dan Kendall (2003), model SDLC air terjun atau waterfall sering juga disebut model sekuensial linier atau alur hidup klasik. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung. Dari kenyataan yang terjadi sangat jarang model air terjun dapat dilakukan sesuai alurnya karena sebab seperti perubahan spesifikasi perangkat lunak terjadi di tengah alur pengembangan, adanya kesulitan bagi pelanggan untuk mendefinisikan semua spesifikasi di awal alur pengembangan. Pelanggan



sering kali membutuhkan contoh untuk menjabarkan spesifikasi kebutuhan sistem lebih lanjut, serta pelanggan tidak mungkin bersabar mengakomodasi perubahan yang diperlukan di akhir alur pengembangan. Dengan berbagai kelemahan yang dimiliki model air terjun namun model ini telah menjadi dasar dari model-model lain dalam melakukan perbaikan model pengembangan perangkat lunak.

Model *waterfall* ini adalah model SDLC yang paling sederhana, dan hanya cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah. Tahapan dari model *waterfall* ini dapat dilihat pada gambar 1.



Gambar 2. 1 Model Waterfall (Kendall dan Kendall, 2003)

### 2.9.2 Fase dalam metode *Waterfall*

Berikut ini akan dijelaskan secara singkat tentang tahapan dalam model *waterfall*, yaitu:

#### 1. *System Requirements*

Merupakan tahap pengumpulan data tentang kondisi awal dari suatu permasalahan yang akan diselesaikan. Data tersebut seperti siapa saja *stakeholder* yang ada, bagaimana keadaan sistem yang sedang digunakan saat ini dan perubahan seperti apa yang diinginkan oleh para *stakeholder* tersebut.

## 2. *Software Requirements*

Tahap selanjutnya adalah mendefinisikan kebutuhan perangkat lunak yang akan dibangun sesuai dengan apa yang diinginkan oleh para *stakeholder*.

## 3. *Analysis*

Tahap ini merupakan tahap mengidentifikasi, menyeleksi, dan merencanakan sistem yang bertujuan untuk mendeteksi dan memberikan solusi terhadap permasalahan yang ada.

## 4. *Program Design*

Tahap ini melakukan desain, pendefinisian dan pengolahan data yang terkait dengan fungsi, desain basis data, pendefinisian pengolahan *database*, waktu eksekusi, mendefinisikan *interface* dan penjelasan tentang *input*, *process*, dan *output*.

## 5. *Coding*

Tahap untuk melakukan pengkodean untuk membangun perangkat lunak sesuai dengan hasil dari desain program sekaligus menyiapkan dokumentasi untuk setiap aktivitas pengkodean.

## 6. *Testing*

Melakukan uji kelayakan perangkat lunak yang telah dibangun sesuai dengan scenario dan *test plan* yang disiapkan.

## 7. *Operations*

Tahap ini adalah pengimplementasian dan instalasi perangkat lunak, dimana perangkat lunak tersebut akan diadaptasi dengan sistem yang lama untuk kemudian dilakukan evaluasi.

## 2.10 *Black Box Testing*

Menurut Rizky (2011), pengertian dari *black box testing* adalah suatu tipe *testing* yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya. Berdasarkan hal tersebut, para tester memandang perangkat lunak seperti layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tetapi cukup dikenai proses *testing* bagian luarnya saja.

*Black box testing* hanya memandang perangkat lunak dari sisi spesifikasi dan kebutuhan yang telah ditentukan pada saat awal perancangan. Keuntungan dari jenis *testing* ini antara lain:

1. Anggota tim tester tidak harus dari seseorang yang memiliki kemampuan teknis di bidang pemrograman.
2. Kesalahan dari perangkat lunak ataupun bug sering ditemukan oleh komponen tester yang berasal dari pengguna.
3. Hasil dari *black box testing* dapat memperjelas kontradiksi ataupun kerancuan yang mungkin timbul dari eksekusi sebuah perangkat lunak.
4. Proses *testing* dapat dilakukan lebih cepat dibandingkan *white box testing*.