

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang Masalah

*Spare Part* (suku cadang) adalah komponen dari peralatan kontrol instrumen dan elektrik yang dicadangkan untuk perbaikan atau penggantian bagian mesin yang mengalami kerusakan. Suku cadang merupakan faktor utama yang menentukan jalannya sistem kontrol elektronik dalam suatu mesin. Sehingga suku cadang ini mempunyai peranan yang cukup besar dalam serangkaian proses produksi sebuah perusahaan. PT. Sinergi Utama Bandung adalah salah satu perusahaan yang bergerak di bidang penjualan jasa *Maintenance, Service and Development*. Pelayanan jasa PT. Sinergi Utama Bandung dilakukan oleh tim Workshop yang terdiri dari Kepala Teknisi sebagai penerima perintah kerja dari Direktur Perusahaan, Teknisi yang bertanggung jawab melakukan pekerjaan sesuai dengan perintah kerja dibawah pengawasan Kepala Teknisi, Bagian Gudang sebagai yang bertanggung jawab penuh atas stok *spare part* serta alur keluar masuk *spare part*. Pelayanan di PT. Sinergi Utama kurang maksimal apabila sistem informasi tentang *spare part* tidak tersedia, dimana informasi mengenai ketersediaan *spare part* yang dibutuhkan untuk menyelesaikan pekerjaan dapat diperoleh.

Sistem pengajuan *spare part* yang sedang berjalan masih mengandalkan kertas sebagai media utamanya. Hal ini menyebabkan teknisi sulit untuk mengetahui ketersediaan *spare part* yang dibutuhkan, karena ketersediaan *spare part* hanya tersedia di Bagian Gudang. Ketersediaan *spare part* inipun menumpuk dan tidak tersusun dengan rapih di bagian gudang. Bagian Gudang sering mengeluh karena dalam proses pencarian *spare part* sering terjadi pengecekan berulang pada *spare part* yang sama. Hal ini sering dikeluhkan oleh teknisi karena memperlambat dalam menyelesaikan pekerjaannya.

Pada sistem pengajuan *spare part* yang sedang berjalan Teknisi harus menulis kebutuhannya lalu mengajukannya pada Kepala Teknisi untuk

mendapat persetujuan, setelah itu Teknisi pergi ke Bagian Gudang untuk mendapatkan *spare part* yang dibutuhkannya. Sistem pengajuan ini memerlukan beberapa kali pengecekan oleh Kepala Teknisi yang berkoordinasi dengan Bagian Gudang. Sehingga sering terjadi keterlambatan penyelesaian pekerjaan oleh Teknisi yang diakibatkan oleh lambatnya koordinasi dari Kepala Teknisi dan Bagian Gudang.

Lambatnya koordinasi ini disebabkan oleh dokumen-dokumen yang masih berupa kertas menumpuk sehingga sulit untuk mencari daftar stok *spare part* yang tersedia di gudang. Koordinasi antara Kepala Teknisi dengan Bagian Gudang juga sering terkendala oleh beberapa dokumen alur keluar masuk *spare part* yang hilang.

## **1.2 Identifikasi Masalah**

Berdasarkan latar belakang yang telah dijelaskan diatas, maka penulis mengidentifikasi masalah-masalah yang ada sebagai berikut:

1. Sistem pengajuan *spare part* yang ada saat ini belum terkomputasi sehingga Bagian Gudang sulit untuk mengetahui letak posisi *spare part* dan jumlah yang tersedia.
2. Untuk mengajukan *spare part* yang dibutuhkan Teknisi harus menulis kebutuhannya lalu mengajukannya pada Kepala Teknisi untuk mendapat persetujuan, setelah itu Teknisi pergi ke Bagian Gudang untuk mendapatkan *spare part* yang dibutuhkannya.
3. Pembuatan laporan stok *spare part* dan alur masuk keluarnya *spare part* masih mengacu pada catatan-catatan berupa kertas sehingga sering terjadi kesalahan dikarenakan hilangnya beberapa catatan.

## **1.3 Maksud dan Tujuan**

Berdasarkan permasalahan yang telah diuraikan sebelumnya, maka maksud dari penelitian ini yaitu membangun dan mengimplementasikan RFID untuk aplikasi *Spare Part Ordering System* dan tujuannya adalah:

1. Mempermudah Teknisi dalam mengajukan *spare part* yang dibutuhkan.

2. Penerapan teknologi *Radio Frequency Identification* ( RFID ) sebagai alat bantu pencarian *spare part*.
3. Mempermudah pembuatan laporan stok *spare part* dan alur masuk keluar *spare part* oleh Bagian Gudang.

#### **1.4 Batasan Masalah**

Batasan yang dimaksud adalah untuk membatasi ruang lingkup pekerjaan dan dilaksanakan dengan tujuan memperkecil masalah yang ada pada bagian umum serta pihak – pihak yang bersangkutan.

1. Aplikasi yang akan dibangun berbasis Web.
2. Pengelolaan data yang digunakan menggunakan Database Mysql.
3. Ruang lingkup penelitian dibatasi hanya di Bagian Gudang saja.
4. Aplikasi yang dibangun memanfaatkan teknologi RFID yang memanfaatkan kartu RFID yang di dekatkan pada RFID Reader untuk mengirim data *spare part*.
5. *Spare part* yang digunakan memiliki nilai ekonomis yang tinggi dan tidak umum dipasaran.
6. Alat pendeteksi RFID hanya diperuntukan mencari *spare part* yang sudah diajukan.

#### **1.5 Metodologi Penelitian**

Metodologi penelitian adalah sebuah cara sistematis yang dilakukan dalam proses penelitian, yang bertujuan untuk membuat penelitian lebih teratur dengan menggunakan metode dan langkah-langkah yang telah ditentukan agar memenuhi kebutuhan penelitian secara maksimal[1]. Metodologi yang digunakan dalam penelitian adalah metode penelitian kuantitatif. Kuantitatif adalah metode penelitian yang bersifat deskriptif dan penggunaan analisis lebih banyak didalamnya[2].

Metode penelitian yang akan dilakukan pada penelitian ini dapat diuraikan seperti dibawah ini.

### 1.5.1 Metode Pengumpulan Data

Metode pengumpulan data dibutuhkan sebagai dasar dari penelitian yang dilakukan. Metode pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut:

#### a. Wawancara

Teknik pengumpulan data dengan melakukan wawancara langsung dengan pihak-pihak yang terlibat dalam pengajuan *spare part*.

#### b. Observasi

Pada teknik pengambilan data dengan cara observasi, mahasiswa terjun langsung sebagai *Helper* Teknisi yang membantu melakukan pengajuan *spare part* dan melakukan pengamatan serta pencatatan sehingga di peroleh data yang sebenar – benarnya.

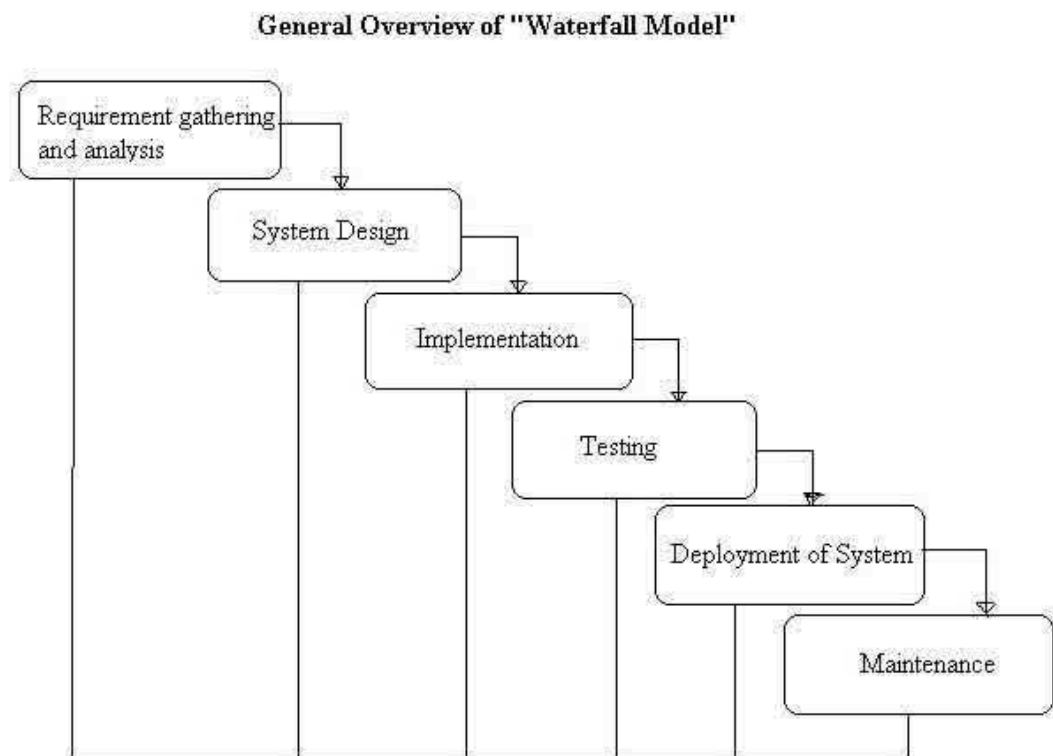
#### c. Studi Pustaka

Pengambilan data dengan mencari referensi yang ada baik berupa buku, ataupun berupa artikel yang berkaitan dengan yang sedang dibahas.

### 1.5.2 Metode Pembangunan Perangkat Lunak

Metode pengembangan perangkat lunak yang digunakan adalah Metode SDLC (*System Development Life Cycle*) dengan model *waterfall*. Model ini memberikan pendekatan-pendekatan sistematis dan berurutan bagi pengembangan piranti lunak. Berikut gambar pengembangan sistem perangkat lunak dengan proses SDLC dengan model *waterfall*:

Model *Waterfall* merupakan salah satu metode SLDC (*System Development Life Cycle*) yang mempunyai ciri khas pengerjaan dalam setiap fase dalam *waterfall* harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya. Artinya fokus dalam fase dapat dilakukan dengan maksimal karena tidak adanya pengerjaan yang sifatnya paralel. Model *Waterfall* dapat dilihat pada **Gambar 1.1 [1]**



**Gambar 1.1 SDLC Model Waterfall**

Sumber : [www.tutorialspoint.com/sdlc/sdlc\\_waterfal\\_model.htm](http://www.tutorialspoint.com/sdlc/sdlc_waterfal_model.htm)

Penjelasan dari tahap-tahap model Waterfall adalah sebagai berikut:

1. *Requirement gathering and analysis*

Tahapan ini adalah tahapan di mana *developer* mengumpulkan informasi selengkap-lengkapnyanya dari *client*, apa yang sebenarnya *client* butuhkan dengan solusi *software application* yang ditawarkan oleh *developer*. *Developer* harus mampu memahami maksud *client* dengan secermat mungkin tanpa ada kesalah pahaman dan harus mengumpulkan semua informasi kebutuhan *client* bukan sebagian. Setelah itu kemudian *developer* menentukan kira-kira teknologi apa yang bakal dipakai.

## 2. *System Design*

Tahapan ini *developer* merancang desain dari *User Interface* (UI), *User Experience* (UX), mendesain *database*, mendesain alur program dan rancangan sistem lainnya. Jadi tahapan ini intinya untuk merancang rancangan sistem dan juga mendesign tampilan antarmuka aplikasi. Meskipun banyak literatur yang menerangkan konsep diagram-diagram sistem seperti misalnya UML (*Unified Modeling Language*), ERD (*Entity Relationship Diagram*), namun terkadang tidak perlu semua diagram dibutuhkan pada tahapan ini karena bergantung pada kebutuhan sistem.

## 3. *Implementation*

Tahapan ini adalah tahapan dimana *developer* menulis kode-kode program aplikasi. Bergantung pada platformnya juga. Apabila platform aplikasi yang dipakai yaitu web maka tahapan dimulai dari *Slicing* atau proses mengubah rancangan design UI/UX yang sebelumnya dirancang oleh desainer menjadi tampilan web statis dengan bahasa markup tertentu seperti HTML, CSS dengan dibubuhi bahasa *client side scripting language* contohnya *javascript* atau *coffescript*. Baru setelah tahapan *slicing* selesai dilaksanakan maka memasuki tahap penulisan kode *server side* seperti PHP, Java, Node.js dan lain-lain. Jadi beda platform aplikasi maka beda tahapan pada implementasi ini

## 4. *Testing*

Proses ini akan menguji kode program yang telah dibuat dengan memfokuskan pada bagian dalam piranti lunak. Tujuannya untuk memastikan bahwa semua pernyataan telah diuji dan memastikan juga bahwa *input* yang digunakan akan menghasilkan *output* yang sesuai. Pada tahap ini pengujian dibagi menjadi dua bagian, pengujian internal dan pengujian eksternal. Pengujian internal bertujuan menggambarkan bahwa semua statemen sudah dilakukan pengujian, sedangkan

pengujian eksternal bertujuan untuk menemukan kesalahan serta memastikan *output* yang dihasilkan sesuai dengan yang diharapkan

#### 5. *Deployment Of System*

Tahapan ini maksudnya yaitu pemasangan aplikasi di lapangan. Tergantung platformnya juga. Kalau misalkan platform web berarti pemasangan web pada jaringan komputer yang direncanakan sebelumnya. Apabila jaringan internet berarti pemasangan web pada penyedia jasa hosting. Apabila jaringan intranet suatu instansi berarti pemasangan web pada web server milik instansi tersebut. Apabila aplikasi berupa aplikasi desktop berarti developer tersebut harus membuat installer-nya terlebih dahulu sebelum aplikasi akan dipasang pada komputer client. Jadi berbeda platform berbeda cara instalasi bahkan beda bahasa pemrograman pada platform yang sama deployment process nya pun bisa berbeda

#### 6. *Maintenance*

Proses ini dilakukan setelah piranti lunak telah digunakan oleh pemakai atau konsumen. Perubahan akan dilakukan jika terdapat kesalahan, oleh karena itu piranti lunak harus disesuaikan lagi untuk menampung perubahan kebutuhan yang diinginkan konsumen

### **1.6 Sistematika Penulisan**

Disusun untuk memberikan gambaran secara umum mengenai penulisan penyusunan tugas akhir ini.

#### **BAB 1 PENDAHULUAN**

Bab ini membahas mengenai latar belakang yang diambil rumusan masalah sistem pengajuan *spare part* yang terjadi di PT. Sinergi Utama, maksud dan tujuan yaitu menjelaskan tentang pencapaianakirdan menjawab masalah yang ada, batasan masalah yaitu menjelaskan tentang batasan-batasan pada sistem yang dibuat, metodologi penelitian dan sistematika penelitian.

## **BAB 2 LANDASAN TEORI**

Bab ini membahas mengenai sejarah, logo instansi, visi – misi, badan hukum instansi, bidang jasa pekerjaan, personal dan lingkup layanan, struktur organisasi dan deskripsi pekerjaannya serta landasan teori yang berkaitan dengan topik penelitian yang dilakukan yaitu pembangunan aplikasi sistem pengajuan *spare part* dan alat-alat yang mendukung dalam pembangunan aplikasi pengajuan ini.

## **BAB 3 ANALISIS DAN PERANCANGAN SISTEM**

Bab ini menjelaskan analisis yang berkaitan dengan sistem yaitu analisis masalah yang menjelaskan permasalahan-permasalahan yang terdapat pada sistem sehingga aplikasi pengajuan *spare part* tersebut harus dibangun, analisis sistem yang sedang berjalan menjelaskan prosedur yang bisa dilakukan oleh sistem pada sistem tersebut, memberikan informasi spesifikasi kebutuhan non fungsional yang mencakup pengguna sebagai target yang menggunakan aplikasi nantinya, *software* sebagai perangkat yang mendukung penggunaan aplikasi, dan *hardware* sebagai perangkat yang mendukung penggunaan aplikasi secara fisik. Selanjutnya, digunakan *tools* untuk membangun perancangan dalam bentuk diagram-diagram maupun antarmuka sistem.

## **BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM**

Bab ini membahas implementasi dari hasil analisis dan perancangan yang telah dibuat disertai juga dengan hasil pengujian dari sistem yang dibangun ini sehingga diketahui apakah sistem yang dibangun sudah memenuhi syarat sebagai aplikasi yang *user-friendly*.

## **BAB 5 KESIMPULAN DAN SARAN**

Bab ini membahas tentang kesimpulan yang sudah diperoleh dari hasil penulisan tugas akhir dan saran mengenai pengembangan aplikasi yang dibangun untuk masa yang akan datang agar mendapatkan pencapaian yang maksimal dan dapat bermanfaat dalam penggunaannya.