

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Setiap pengerjaan suatu proyek perangkat lunak maupun mengembangkan suatu perangkat lunak tentunya ingin menghasilkan suatu perangkat lunak atau *software* yang baik yang mampu dengan mudah untuk digunakan dan dikembangkan lagi kedepannya, serta mudah dalam proses pemeliharaan. Untuk menghasilkan perangkat lunak seperti yang sudah disebutkan tadi, selain membutuhkan keahlian pemrograman yang mumpuni maka diperlukan juga suatu metode yang tepat dan sesuai dalam pengerjaan dan pengembangannya. Dengan menggunakan metode yang tepat maka tahapan pengerjaan suatu perangkat lunak akan lebih terjadwal dengan baik sehingga mampu menghasilkan suatu perangkat lunak yang baik pula. Selain menentukan metode pengembangan suatu perangkat lunak Salah satu hal yang perlu diperhatikan dalam proses pengembangan suatu perangkat lunak adalah menentukan standar dan Mengukur kualitas perangkat lunak yang dikerjakan serta mengukur kebutuhan sumber daya perangkat lunak yang perlukan dalam pengembangan.

Kebutuhan sumber daya pengembangan perangkat lunak dapat diukur dengan dua cara yakni dengan melakukan pengukuran secara langsung dan pengukuran tidak langsung. Roger S. Pressman, Ph.D. dan Bruce R. Maxim, Ph.D. dalam bukunya yang berjudul “*Software Engineering A Practitioner’s Approach*” mengatakan bahwa pengukuran kualitas suatu perangkat lunak secara langsung dapat dilakukan dengan menggunakan metode *Lines of Code* (LOC), sedangkan untuk melakukan pengukuran tidak langsung metode yang paling sering digunakan adalah metode *function point* (FP) (Pressman and Maxim, 2011). Kedua metode ini jika dibandingkan memiliki perbedaan yang sangat signifikan dimana dalam penggunaan metode LOC perangkat lunak diukur berdasarkan bahasa pemrograman yang digunakan dalam mengembangkan suatu perangkat lunak (DENNIS, HALEY WIXOM and M. ROTH, 2011). Jadi hasil pengukuran dengan menggunakan metode LOC akan menghasilkan nilai yang berbeda untuk setiap bahasa pemrograman yang berbeda. Walaupun perangkat lunak tersebut

mempunyai fungsionalitas yang sama. Sedangkan FP adalah metode pengukuran perangkat lunak yang komponennya terdiri dari beberapa komponen yang mempengaruhi fungsionalitas suatu perangkat lunak yakni *External Input* (EI), *External Input* (EI), *External Inquiry* (EQ), *Internal Logical File* (ILF), dan *External Interface File* (EIF).

Pengukuran kebutuhan sumber daya pengembangan suatu perangkat lunak menggunakan metode FP (*Function point*) harus dilakukan oleh orang yang sudah ahli dalam *Function point*. Namun pada kenyataannya Saat ini masih banyak yang belum memahami dan bahkan belum mengetahui metode yang digunakan untuk mengukur kebutuhan sumber daya pengembangan suatu perangkat lunak terutama metode *function point*. Hal ini dikarenakan pada sebagian orang yang mampu mengidentifikasi DFD secara manual tetapi tidak mengetahui penerapannya pada metode *function point*, sebagian lagi ada yang memahami tentang metode *function point* tetapi masih sulit untuk mengidentifikasi atau membaca alur ERD dan DFD secara manual sehingga mengakibatkan kesalahan yang fatal dalam mengukur kualitas perangkat lunak.

Melihat permasalahan tersebut dimana dalam menghitung dan mengukur perangkat lunak serta menentukan kebutuhan akan sumberdaya dengan metode *function point* diperlukan kemampuan pemahaman dalam mengidentifikasi atau menganalisa DFD serta mengetahui proses mendapatkan nilai *function point* yang benar. Maka pada penelitian ini akan dibuat sebuah sistem yang mampu memudahkan dalam menentukan serta mencari nilai *function point* pada suatu perangkat lunak. Sistem akan menentukan nilai *function point* berdasarkan DFD perangkat lunak yang sebelumnya telah diubah kedalam bahasa *xml* dengan menggunakan aplikasi visual paradigm. Nantinya nilai *function point* yang diperoleh dapat diunduh dalam bentuk *file pdf* dan menjadi sebuah laporan pelengkap dalam pengembangan suatu perangkat lunak serta dapat memudahkan dalam melakukan evaluasi pengembangan perangkat lunak.

## 1.2 Rumusan Masalah

Pengukuran perangkat lunak dengan menggunakan metode *function point* masih jarang dilakukan dan perhitungan untuk menghasilkan nilai *function point* yang benar memerlukan orang yang telah ahli dalam bidang tersebut. Dari permasalahan tersebut, dapat dirumuskan sebuah masalah yakni bagaimana menghasilkan sebuah sistem yang dapat memudahkan dan menghitung nilai *function point* dari sebuah proyek perangkat lunak dengan mengharapkan bahwa nilai dari hasil perhitungan *function point* tersebut cukup akurat?

## 1.3 Batasan Masalah

Batasan-batasan masalah dalam penelitian ini mencakup sebagai berikut:

- a. Sistem dapat menghitung nilai *function point* dengan menginputkan parameter-parameter yang dibutuhkan secara manual
- b. Sistem dapat melakukan perhitungan otomatis dengan syarat DFD dan perangkat lunak yang akan diukur telah di ubah ke dalam bahasa *xml* menggunakan aplikasi *Visual Paradigm* versi 5.0 dimana *file xml* yang dihasilkan adalah *xml* versi 1.0, sedangkan untuk sistem yang dibangun menggunakan PHP versi 5.
- c. Perhitungan FP yang digunakan berdasarkan pada perhitungan FP yang dikemukakan oleh Albreth.

## 1.4 Tujuan Penelitian

- a. Tujuan tugas akhir ini adalah membuat Sistem pengukuran kebutuhan sumber daya perangkat lunak dengan menggunakan metode *function point*.
- b. Sistem mampu mendapatkan dan menghitung nilai 5 elemen *function point* yakni *External Inputs*, *External Outputs* dan *External Inquiry* serta *Internal Logical File* dan *External Interface File*.

## 1.5 Manfaat Penelitian

Menghasilkan sebuah sistem yang dapat membantu untuk menghitung *function point* sebuah sistem berdasarkan berdasarkan hasil *eksport* DFD kedalam *file xml* sistem tersebut

## **1.6 Sitematika Penulisan**

Berikut adalah sistematika penulisan laporan yang peneliti lakukan:

### **BAB I PENDAHULUAN**

Pada BAB I, peneliti menjelaskan latar belakang pembuatan sistem, perumusan masalah, pembatasan masalah dalam pembuatan sistem, tujuan pembuatan sistem, manfaat pembuatan sistem, dan sistematika penulisan laporan.

### **BAB II KAJIAN PUSTAKA**

Pada BAB II, peneliti memaparkan landasan teori yang digunakan sebagai penunjang untuk membuat tugas akhir ini.

### **BAB III PERANCANGAN SISTEM**

Pada BAB III, berisi mengenai analisa bisnis proses, pengumpulan data, pemodelan sistem, *use case diagram* dan *activity diagram* serta desain *user interface*.

### **BAB IV IMPLEMENTASI DAN PENGUJIAN**

Pada BAB IV, berisi tentang penerapan sistem dan tampilan sistem secara umum, serta hasil pengujian sistem.

### **BAB V PENUTUP**

Pada BAB V, berisi tentang kesimpulan dan saran dari peneliti terhadap penelitian yang telah dilakukan