

BAB I

PENDAHULUAN

1.1 Latar Belakang

Selama beberapa tahun belakangan ini perkembangan *Cloud Computing* khususnya *Platform as a Service* sangat populer digunakan oleh perusahaan, lembaga pendidikan dan *developer* pengembang aplikasi untuk mengembangkan aplikasinya menggunakan layanan *Platform as a Service* karena untuk mengembangkan aplikasi cukup mengakses layanan tersebut melalui *website* yang disediakan pada layanan *cloud computing*. Karena abstraksi pada layanan *Platform as a Service* menyembunyikan eksekusi terperinci pada penggunaan alokasi sumber daya fisik (*CPU, memory, disk, dll*), manajemen ekosistem layanan, *operating system*, dan konfigurasi jaringan pada layanan tersebut. *Cloud Computing* khususnya pada layanan *Platform as a Service* juga memiliki kemampuan untuk melakukan otomatisasi *load balancing*, penskalaan sumber daya, dan menyediakan ketersediaan tinggi dan toleransi pada kesalahan komponen yang terdapat pada layanan tersebut.

Semua kemudahan yang disediakan pada layanan *Platform as a Service* memungkinkan pada pengembang aplikasi untuk lebih fokus pada proses pengembangan dan pemrograman aplikasi mereka tanpa harus mengkhawatirkan ketersediaan sumber daya dan masalah infrastruktur yang dibutuhkan untuk mengembangkan aplikasi tersebut. Penggunaan layanan *Platform as a Service* sudah semakin luas dan semakin berkembang, dengan semakin luas penggunaan layanan tersebut dibutuhkan pula suatu sistem yang

dapat memonitoring dan mengawasi layanan tersebut karena sebetulnya dengan memonitoring layanan *Platform as a Service* merupakan kunci untuk melihat seberapa efektifkah proses penskalaan otomatis dan pemanfaatan sumber daya yang digunakan pada layanan tersebut hal ini diperlukan untuk menjaga ketersediaan sumber daya yang tinggi/cukup, serta mengidentifikasi kinerja dari aplikasi yang dibangun pada layanan tersebut. [1]

Beberapa penelitian yang berkaitan dengan memonitoring layanan *Platform as a Service* khususnya memanfaatkan layanan *Openshift Origin* pada *Minishift* lebih memonitoring kepada penggunaan matriks untuk melihat konsumsi sumberdaya pada *Openshift Origin* tersebut seperti penggunaan rata-rata *memory* pada setiap aplikasi yang dikembangkan ataupun penggunaan *CPU* dan *Network* pada aplikasi yang dikembangkan pada layanan *Platform as a Service*. Tetapi belum tersedianya layanan monitoring untuk melihat apakah berhasil atau tidak pengembangan aplikasi pada layanan tersebut, atas dasar itulah maka dikembangkan layanan *monitoring* untuk melihat berhasil atau tidaknya pengembangan aplikasi pada *Openshift Origin* melalui *Minishift* dengan memanfaatkan *Openshift API* dan *Kubernetes API*. Dengan adanya monitoring untuk melihat keberhasilan aplikasi yang berhasil dikembangkan pada layanan *Platform as a Service* dapat diketahui bahwa aplikasi yang dikembangkan tidak berhasil dikembangkan pada layanan tersebut.

1.2 Perumusan Masalah

Dari masalah di atas, dapat diambil rumusan masalah sebagai berikut:

1. Bagaimana mengimplementasikan layanan komputasi awan *Platform as a Services* (PaaS) menggunakan *Openshift Origin* menggunakan *Minishift* yang diimplementasikan pada *Private Cloud*.
2. Bagaimana merancang *Monitoring System* pada layanan *Openshift Origin* menggunakan *Minishift* dengan memanfaatkan *Openshift API* dan *Kubernetes API*.

1.3 Tujuan dan Manfaat

Tujuan yang ingin dicapai dari penelitian ini adalah:

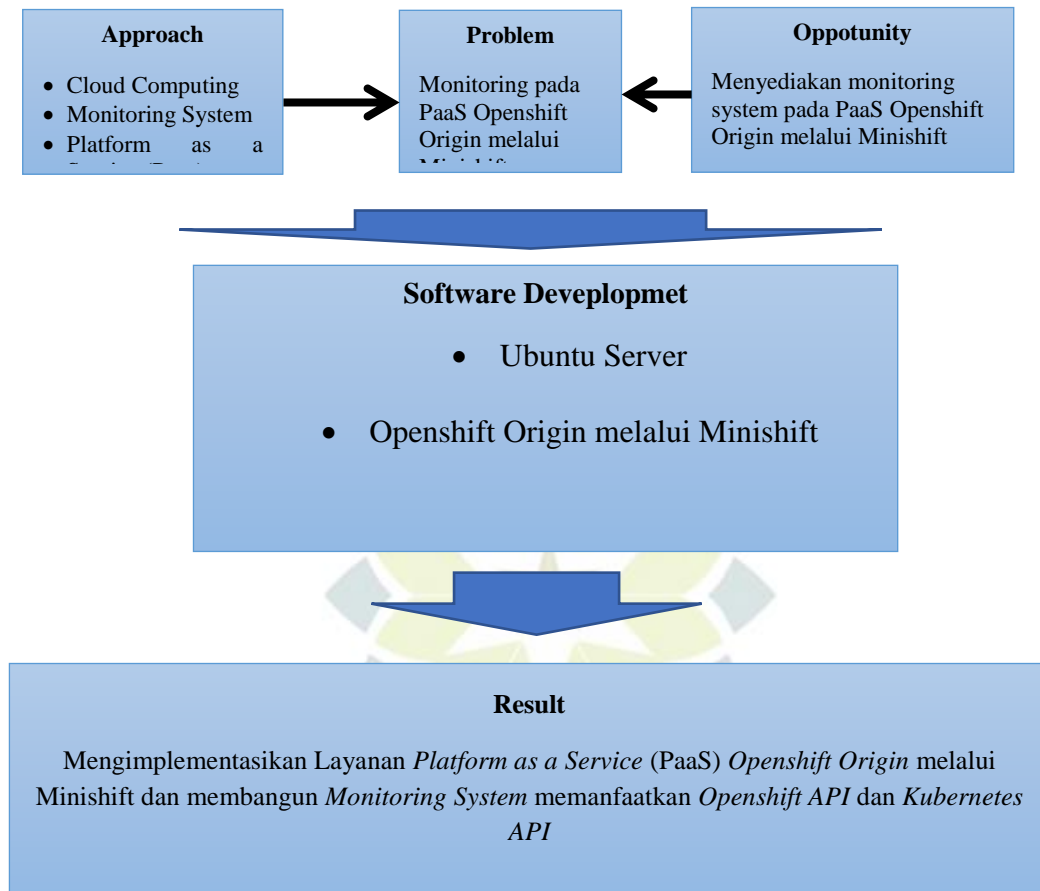
1. Mengimplementasikan layanan komputasi awan *Platform as a Service* (PaaS) pada *Openshift Origin* menggunakan *Minishift* yang akan diimplementasikan pada *Private Cloud*.
2. Merancang *Monitoring System* pada layanan *Openshift Origin* menggunakan *Minishift* dengan memanfaatkan *Openshift API* dan *Kubernetes API*.

1.4 Batasan Masalah

Agar pembahasan penelitian ini lebih terarah dan efektif, maka penulis membatasi pokok pembahasannya sebagai berikut:

1. Mengimplementasikan *private cloud* menggunakan *Ubuntu Server*.
2. Layanan yang disediakan oleh *private cloud computing* adalah *Platform as a Services* (PaaS) menggunakan *Openshift Origin* pada *Minishift*.
3. *Monitoring System* pada layanan *Platform as a Service* (PaaS) yang menggunakan *Openshift Origin* melalui *Minishift*.
4. Indikator yang digunakan adalah terkirimnya file dari *client* ke *server*.
5. Menggunakan *Ubuntu Server* dan *Openshift Origin* melalui *Minishift*.

1.5 Kerangka Pemikiran



Gambar 1.1 Kerangka pemikiran penelitian

Gambar 1.1 di atas memaparkan pemikiran secara kasar mengenai penelitian yang dilakukan untuk mengimplementasikan salah satu layanan *cloud computing* yaitu layanan *platform as a service* yang dikembangkan pada ekosistem *private cloud*. Sistem ini diharapkan dapat mempermudah dalam kegiatan pengembangan aplikasi karena pengembang aplikasi tidak harus memikirkan infrastruktur yang dibutuhkan untuk mengembangkan aplikasi hanya untuk terfokus pada aplikasi apa yang akan dikembangkan. Serta membangun sistem *monitoring* pada layanan *platform as a service* menggunakan *openshift origin* melalui *minishift* memanfaatkan *openshift api* dan *kubernetes api*.

1.6 Metode Penelitian

Metode penelitian yang digunakan dalam penyusunan tugas akhir ini adalah sebagai berikut:

1.6.1 Metode Pengumpulan Data

1. Studi kepustakaan, teknik pengumpulan data dengan memahami dan mengambil data-data dari berbagai *literature*. Berupa: buku, jurnal penelitian, dan artikel baik media cetak atau internet sesuai dengan penelitian yang dibahas.
2. Observasi, teknik yang dilakukan dengan melihat secara langsung ke tempat yang memiliki data yang diperlukan atau melihat secara langsung proses-proses yang terjadi di tempat tersebut.

1.6.2 Metode Pengembangan Sistem

Metode pengembangan sistem menggunakan model pengembangan *prototype*. Tahapan-tahapan yang dilakukan dalam pembuatan *prototype* sebagai berikut:

3. Melakukan komunikasi untuk mendefinisikan sasaran keseluruhan perangkat lunak yang akan dikembangkan dan mengidentifikasi spesifikasi kebutuhan yang saat ini diketahui.
4. Setiap iterasi pembuatan *prototype* direncanakan dengan beserta dengan pemodelannya (dalam bentuk “rancangan cepat”). Suatu rancangan cepat berfokus pada representasi semua aspek yang akan terlihat oleh para pengguna akhir (misalnya rancangan *user interface*). Rancangan cepat akan memulai konstruksi pembuatan *prototype*.

5. *Prototype* akan diserahkan dan akan dilakukan evaluasi-evaluasi tertentu terhadap *prototype* yang telah dibuat sebelumnya, kemudian akhirnya akan memberikan umpan-balik yang akan digunakan untuk memperhalus spesifikasi kebutuhan. Iterasi akan terjadi saat *prototype* diperbaiki untuk memenuhi kebutuhan.

1.6.3 Metode Implementasi Sistem

Terdapat dua tahapan yang dilakukan pada metode implementasi sistem, yaitu pelatihan dan pengujian.

Pada tahap ini mulai dibangun sistem *cloud computing* yang di sesuaikan dengan kebutuhan para pelanggan dan merupakan pengaplikasian dari rancangan yang telah dimodelkan sebelumnya dan sudah dievaluasi oleh para pelanggan.

1.6.4 Metode Analisis Hasil Implementasi Sistem

Tahap akhir dalam pengembangan menggunakan metode *prototype* pengembang kembali berkomunikasi dengan pelanggan setelah sistem terbentuk yang telah dievaluasi sebelumnya, setelah itu pengembang kembali meminta saran kepada pelanggan untuk pengembangan lebih lanjut dan pengembangan sistem kembali pada tahap membangun sistem dengan hasil evaluasi yang diberikan pelanggan.

1.7 Sistematika Penulisan Tugas Akhir

Sistematika penulisan penelitian ini disusun untuk memberikan gambaran umum tentang penelitian yang dijalankan. Sistematika penulisan tugas akhir ini adalah sebagai berikut:

BAB I PENDAHULUAN

Menguraikan tentang latar belakang permasalahan, mencoba merumuskan

inti permasalahan yang dihadapi, menentukan tujuan dan kegunaan penelitian yang kemudian diikuti dengan pembatasan masalah, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Membahas berbagai konsep dasar teori yang berkaitan dengan topik penelitian yang dilakukan dan hal-hal yang berguna dalam proses analisis permasalahan serta tinjauan terhadap penelitian-penelitian serupa yang pernah dilakukan sebelumnya.

BAB III ANALISIS DAN PERANCANGAN

Bab ini membahas analisis terhadap sistem yang dibuat serta bagaimana merancang dan menerapkan *Private Cloud Computing* sebagai *Platform As A Services* (PaaS) melalui *Openshift Origin* dengan *Minishift*. Serta membangun *System Monitoring* pada *Openshift Origin* melalui *Minishift* dengan memanfaatkan *Openshift API* dan *Kubernetes API* untuk membangun *system* tersebut.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Berisi tentang tahapan-tahapan yang dilakukan untuk menerapkan sistem yang telah dirancang. Serta melakukan pengujian terhadap aplikasi yang telah dibangun.

BAB V PENUTUP

Berisi kesimpulan dan saran yang sudah diperoleh dari hasil penulisan laporan.