

BAB I

PENDAHULUAN

1.1 Latar belakang

Di antara banyaknya masalah di dunia perkembangan aplikasi, ada satu masalah klasik yang sering ditemui oleh pengembang aplikasi dan sistem administrator. Masalah tersebut yaitu perbedaan *platform* sistem operasi yang digunakan para pengembang aplikasi dan sistem administrator, maka diperlukannya rancangan arsitektur platform virtualisasi yang lebih baik dan terdistribusi. Pengembang aplikasi dan sistem administrator ada yang menggunakan macos, linux ataupun windows. Variasi *platform* di antara pengguna linux pun akan beragam, baik distro yang digunakan, maupun versi perangkat lunak yang terpasang di dalamnya. Intinya adalah tidak semua pengembang aplikasi menggunakan lingkungan kerja yang sama persis. Teknologi *virtualisasi* menawarkan perluasan penggunaan pada infrastruktur teknologi informasi menjadi lebih tinggi dengan biaya yang lebih rendah. karena *virtualisasi* memanfaatkan sumber daya komputer secara optimal sehingga dapat digunakan untuk berbagai kebutuhan, setiap kali penambahan aplikasi atau fitur baru maka diperlukan juga menambahkan mesin komputer beserta sistem operasi, sehingga biaya untuk lisensi perangkat lunak makin bertambah dengan *virtualisasi* penambahan komputer tidak diperlukan karena sebuah server dapat digunakan secara Bersama-sama dengan berbagai aplikasi yang akan diperlukan. Saat ini teknologi *virtualisasi* kontainer semakin populer terutama docker di mana dapat memasang aplikasi kedalam sebuah kontainer.

Docker adalah sebuah proyek terbuka yang menyediakan *platform* terbuka untuk pengembang aplikasi mau pun sistem administrator untuk dapat membangun, mengemas, dan menjalankan aplikasi di mana pun sebagai sebuah

wadah kontainer yang ringan. docker awal mulanya dikembangkan oleh *Solomon hykes* sebagai proyek internal di *dotcloud*, sebuah perusahaan PAAS.

Sistem kontainer docker melakukan pembungkusan pada bagian-bagian perangkat lunak dalam berkas sistem secara lengkap yang berisi semua kebutuhan yang diperlukan dalam menjalankan kode, *runtime*, sistem alat dan sistem perpustakaan yang dapat dijalankan pada *server*, sehingga perangkat lunak dapat berjalan sama meskipun dalam lingkungan yang berbeda. Kontainer berjalan pada satu komputer tunggal yang dapat dipasang berbagai sistem operasi dalam *kernel* yang sama dapat mengurangi penggunaan RAM dibangun dengan berkas sistem yang berlapis sehingga penggunaan memori menjadi lebih efisien.

1.2 Perumusan Masalah

Berdasarkan penjelasan di atas pada bagian latar belakang, maka penulis merumuskan berbagai masalah yang akan dianalisis dengan penelitian yang akan penulis yaitu:

1. Bagaimana rancangan dan arsitektur *platform* virtualisasi berbasis docker?
2. Bagaimana keefektifan rancangan arsitektur docker sebagai *platform* virtualisasi berbasis kontainer.
3. Bagaimana membuktikan kompatibilitas kontainer docker untuk menjalankan aplikasi yang telah berjalan di atas *platform* sistem operasi yang berbeda.
4. Bagaimana docker dapat digunakan untuk membantu proses kontrol versi sebuah aplikasi.

1.3 Tujuan dan Manfaat Penelitian

1.3.1 Tujuan

Tujuan yang ingin dicapai dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Merancang Arsitektur *platform* virtualisasi berbasis docker yang mudah digunakan dan cepat dalam implementasi serta lebih terdistribusi.
2. Menguji efektifitas rancangan asitektur docker sebagai platform virtualisasi berbasis kontainer.
3. Menguji kompatibilitas docker sebagai platform virtualisasi untuk mengatasi permasalahan terkait ketergantungan lintas *platform*.
4. Melakukan pengujian kontrol versi untuk aplikasi.

1.3.2 Manfaat Penelitian

Dalam penelitian ini manfaat yang didapatkan adalah sebagai berikut:

1. Memahami cara kerja dan penggunaan docker sebagai salah satu *platform* berbasis kontainer.
2. Menghasilkan karya tugas akhir di bidang virtualisasi berbasis docker yang dapat menjadi salah satu rujukan karya tulis dan tugas akhir yang bermanfaat bagi dunia akademis dan dunia industri.

1.4 Batasan Masalah

Setelah menelaah rumusan masalah di atas, maka didapat beberapa batasan masalah di antaranya:

1. Perangkat lunak docker dipasang pada sistem operasi Ubuntu versi 17.10.
2. Manajemen sistem docker menggunakan alat yang telah tersedia (*built in*) pada perangkat lunak docker dan alat manajemen *rancher server*.

3. Pada uji kompatibilitas docker untuk mengatasi permasalahan terkait ketergantungan lintas *platform* hanya menguji sebuah aplikasi web yaitu joomla.

4. Dalam penelitian ini tidak dilakukan pengujian keamanan pada sistem docker.

1.5 Metode Penelitian

Dalam tulisan ini penulis menggunakan metode kualitatif deskriptif dan eksperimental dengan menggunakan perangkat lunak virtualisasi dan membuat simulasi yang disesuaikan dengan kasus aslinya. Adapun teknik pengumpulan datanya dilakukan dengan dua unsur pengumpulan data:

1. Studi pustaka data diperoleh dari tulisan-tulisan dan panduan di internet yang berkaitan dengan masalah yang penulis bahas.
2. Simulasi dilakukan dengan percobaan pembuatan pemodelan dengan berdasarkan panduan yang berkaitan dengan masalah yang di bahas.

1.6 Sistematika Penulisan

Sistematika penulisan tugas akhir yang digunakan penulis adalah sebagai berikut:

- **BAB I PENDAHULUAN**

Dalam bab ini penulis menjelaskan tentang latar belakang, identifikasi/perumusan masalah dan batasan masalah, tujuan dan manfaat, metode penulisan dan sistematika penulisan.

- **BAB II KAJIAN LITERATUR**

Dalam bab ini penulis menjelaskan mengenai kajian literatur serta teori yang berkaitan dengan judul tugas akhir ini.

- **BAB III METODOLOGI PENELITIAN**

Dalam bab ini akan menjelaskan mengenai metode dalam melakukan penelitian mengacu pada apa saja yang akan dianalisis oleh peneliti.

- **BAB IV ANALISIS DAN PERANCANGAN**

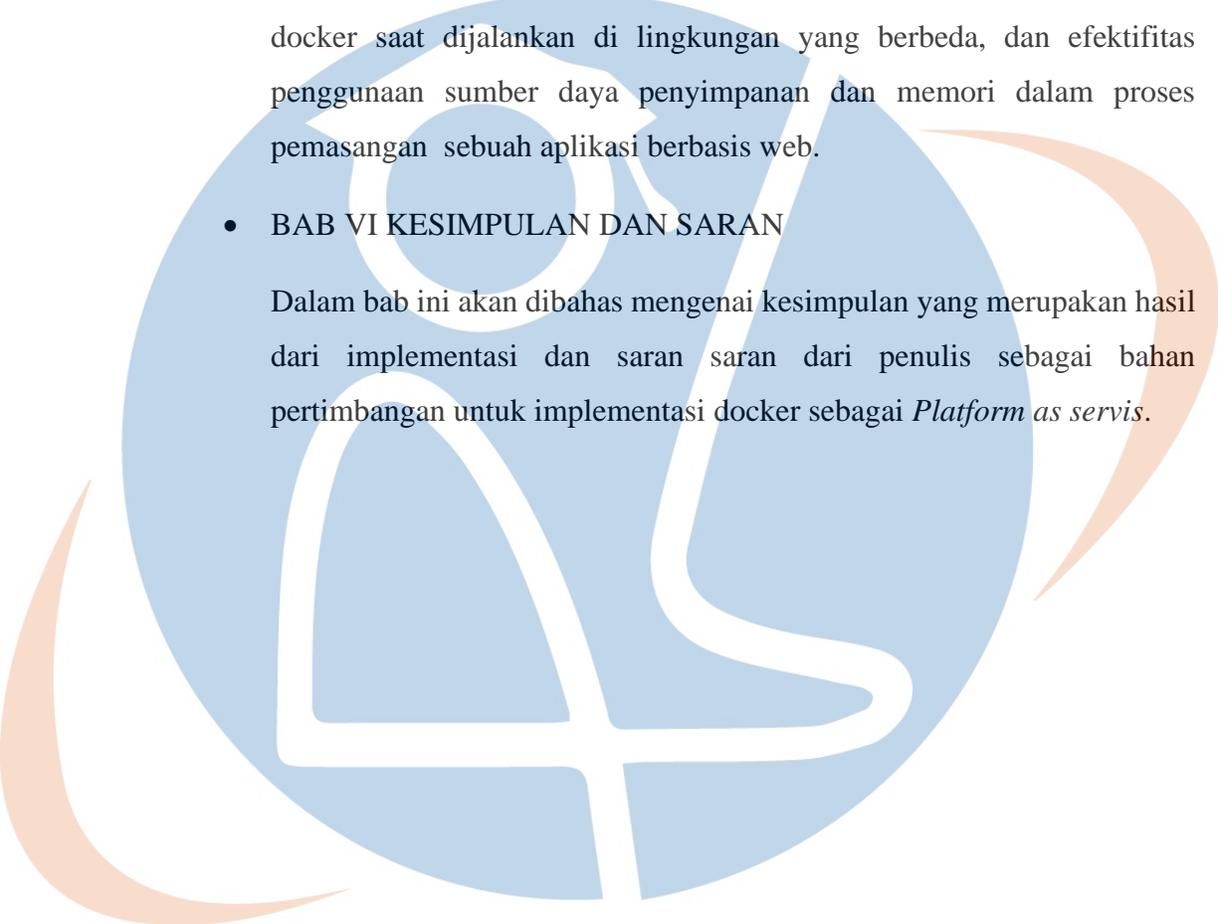
Dalam bab ini menjelaskan mengenai analisis dan perancangan pengujian beserta tahapan-tahapan implementasinya.

- **BAB V IMPLEMENTASI DAN EVALUASI**

Dalam bab ini menjelaskan mengenai pengujian kompatibilitas Aplikasi docker saat dijalankan di lingkungan yang berbeda, dan efektifitas penggunaan sumber daya penyimpanan dan memori dalam proses pemasangan sebuah aplikasi berbasis web.

- **BAB VI KESIMPULAN DAN SARAN**

Dalam bab ini akan dibahas mengenai kesimpulan yang merupakan hasil dari implementasi dan saran saran dari penulis sebagai bahan pertimbangan untuk implementasi docker sebagai *Platform as servis*.



STT - NF