

BAB VI KESIMPULAN DAN SARAN

Pada Bab VI akan dibahas mengenai kesimpulan dari hasil yang telah dicapai pada pembuatan aplikasi JobDesk. Beserta saran untuk pengembangan aplikasi.

6.1. Kesimpulan

Aplikasi JobDesk diharapkan dapat mengoptimasi kerja setiap *Project Manager* maupun *Member* untuk *me-manage project* yang sedang dilakukan. Fitur-fitur aplikasi yang ada diharapkan dapat memfasilitasi setiap pengguna untuk menetapkan tugas, berkomunikasi, bertukar data. Perkembangan proyek diharapkan tidak terganggu ketika para anggota tidak dapat berkumpul di suatu tempat, meskipun para anggota tim sedang berada di kota bahkan negara yang berbeda, tetapi dengan akses internet setiap anggota dapat saling memantau perkembangannya. Pemberian tanggung jawab juga diharapkan dapat dilakukan meskipun *Project Manager* tidak dapat bertemu langsung dengan *Member*.

Penggunaan *framework Vaadin* bagi mereka yang memiliki pengetahuan *Java*, sangat mudah untuk pembuat *web application* yang dinamis, karena tidak perlu memiliki pengetahuan bahasa *web programming*, seperti *HTML*, *CSS*, *AJAX*, bahkan *JSP* sekalipun. *Developer* hanya akan bekerja di lingkungan *Java*. Namun *Developer* ada baiknya memiliki beberapa pengetahuan *web* seperti *CSS*, *HTML*, karena *Vaadin* dapat memfasilitasi pembuatan *custom component* sesuai yang dibutuhkan. *Framework Vaadin* secara *default* telah memiliki hasil tampilan *user interface* yang menarik. Namun memiliki tingkat kesulitan yang cukup sulit untuk memodifikasi tampilan *Vaadin*, seperti warna dan bentuk.

Berbeda dengan rencana awal menggunakan bahasa *web programming*. *Developer* harus memiliki pengetahuan *HTML*, *CSS*, *AJAX*, *JavaScript*, *DHTML*, *jQuery*, *PHP* untuk mengembangkan aplikasi web yang dinamis. Hal ini menyebabkan penggantian *platform* menjadi *Java* menggunakan *framework Vaadin*. Namun bila *developer* memiliki pengetahuan tentang *web programming*, aplikasi web yang dihasilkan akan lebih mudah untuk melakukan *customize* sesuai dengan kebutuhan.

Teknologi *Hibernate* juga digunakan untuk memudahkan implementasi antara *Java* dengan *database*. Penggunaan *Hibernate* tidak tergantung pada 1 jenis *database*, *developer* dapat dengan mudah mengganti *database*, misal dari *MySQL* ke *Oracle*. Teknologi *Hibernate* akan memfasilitasi *query* ke *database*, namun diperlukan pengetahuan *Hibernate Query Language* dan *XML*.

6.2. Saran

Saran untuk pengembangan aplikasi:

1. Dapat terhubung pada *server svn*, sehingga *file-file* yang ada dapat lebih mudah dipantau perkembangan perubahannya.
2. Mengganti *platform* ke bahasa pemrograman *web*, sehingga dapat lebih disesuaikan dengan kebutuhan, karena *framework Vaadin* tidak mudah untuk dilakukan *customize*.