

1. Pendahuluan

1.1 Latar belakang

Saat ini pengembangan website mengarah kepada apa yang disebut dengan *Rich Interaction Application (RIA)*, berbeda dengan pengembangan website pada saat awal kemunculannya yang bersifat *page-based*, pengembangan mengarah kepada *micro-content*. Dan *RIA* membuat *software usability* (observasi pada kebutuhan dan kebiasaan pengguna komputer) yang biasanya diterapkan pada *desktop application* menjadi bagian dari pengembangan website.

Berdasarkan sejarahnya teknik-teknik usability pada umumnya diterapkan pada aplikasi *desktop*, karena pada saat itu tool dan teknologi pengembangan website belum dapat menerapkannya. Diawali dengan kemunculan *Dynamic HTML (DHTML)* yang mengkombinasikan *JavaScript* dan *CascadingStyleSheet (CSS)* pada halaman web membuat jarak pemisah pada pengalaman pengguna menggunakan aplikasi desktop dan aplikasi berbasis web semakin dekat. Dan dengan kemunculan *Asynchronous JavaScript and XML (AJAX)* membuat jarak itu menjadi semakin dekat.

Pengembang AJAX dari Adaptive Path memberikan definisi pada AJAX sebagai sekumpulan teknologi yang membolehkan aplikasi-aplikasi web menyediakan *rich interaction*, informasi pada saat ini dan antarmuka yang dinamis tanpa perlu melakukan refresh pada keseluruhan halaman web[4]. AJAX memiliki perbedaan model interaksi dengan aplikasi web klasik. Pada aplikasi web klasik model interaksinya biasa disebut *request-response*, yaitu pengguna melalui *browser* mengirim *request* ke server, kemudian pengguna menunggu *response* dari server, selama menunggu *response* dari server pengguna tidak dapat melakukan hal lainnya. Sedangkan pada AJAX menggunakan sebuah intermediari *AJAX engine* yang berada diantara pengguna dan server. Pengguna melalui web browser melakukan *request*, *request* tersebut biasanya dilakukan pada elemen sebuah halaman web yang kemudian *request* tersebut akan ditangani oleh *AJAX engine*. *AJAX engine* memproses *request* dari pengguna di belakang layar, sehingga memungkinkan interaksi antara pengguna dengan server terjadi asinkronus. Karena pemrosesan *request* terjadi dibelakang layar jadi pengguna tidak pernah lagi menemukan permulaan berupa *window browser* yang kosong. Model interaksi aplikasi web berbasis AJAX dapat meningkatkan interaktivitas dan usability dari pengguna.

AJAX dengan semua keunggulannya dari definisi dan model interaksi diatas memiliki masalah dan tantangan, dengan AJAX sebuah web dikonstruksi sebagai sebuah *single web page* yaitu mencoba untuk menampilkan banyak fungsionalitas pada sebuah halaman web dan proses pengupdateaan isi pada website tidak harus merefresh seluruh halaman, tetapi hanya bagian-bagian dari halaman web saja yang diupdate, hal ini membuat *back button* pada web browser tidak dapat memberikan hasil sesuai yang diharapkan oleh pemakai, karena *history* pada web browser adalah *URL* dari halaman yang dikunjungi oleh pemakai, sedangkan pada aplikasi web berbasis AJAX hanya merubah state dari aplikasi. Karena AJAX

telah "merusak" paradigma *history* dan *URL* pada web, teknik *bookmarking* menjadi sebuah problema.

Metode-metode yang umumnya digunakan dalam teknik *bookmarking* pada aplikasi web adalah hidden Iframe, URL Fragment dan Really Simple History. Really Simple History menggunakan dua buah kelas JavaScript, DhtmlHistory dan HistoryStorage. DhtmlHistory menspesifikasikan lokasi baru dan data-data yang berhubungan dengan *history*. Sedangkan HistoryStorage digunakan untuk tempat penyimpanan data-data *history*. Pada halaman web yang normal jika kita meninggalkan halaman tersebut maka web browser akan melakukan unload dan membersihkan seluruh status dari aplikasi dan JavaScript. Dengan ini jika pengguna meninggalkan suatu halaman web, kemudian menggunakan *backbutton*, data dari *history* dapat diakses melalui HistoryStorage.

1.2 Perumusan masalah

Permasalahan yang akan dijabarkan dan akan diteliti adalah sebagai berikut :

1. Bagaimana menerapkan *bookmarking* pada sebuah web berdasarkan state dari aplikasi.
2. Bagaimana membuat sebuah aplikasi web berbasis AJAX agar dapat memiliki *history* berdasarkan state dari aplikasi.
3. Bagaimana menerapkan sebuah *back button* agar ketika user menekannya user mendapatkan hasil sesuai dengan yang diinginkannya.
4. Bagaimana membuat sebuah aplikasi web berbasis AJAX dengan fitur-fitur yang mendukung *rich-interaction*.

1.3 Tujuan

Penulisan tugas akhir ini bertujuan untuk :

1. Menganalisis teknik-teknik *bookmarking* yang digunakan pada aplikasi web yang dibangun.
2. Memberikan solusi pada permasalahan pada web berbasis AJAX seperti pada paradigma *history*, *URL*, *backbutton* dan *bookmarking*.
3. Menciptakan aplikasi web berbasis AJAX dengan fitur-fitur yang memiliki *rich-interaction* sebagai sarana untuk menerapkan teknik-teknik *bookmarking*. Aplikasi web yang akan diimplementasikan akan memiliki banyak elemen pada setiap halamannya, seperti frame, tab menu, tree menu dan memiliki banyak *link*.

1.4 Metodologi penyelesaian masalah

1. Studi literatur, dilakukan dengan mencari informasi dan referensi dari buku, jurnal, artikel maupun internet yang berkaitan dengan topik ini.
2. Mempelajari konsep dari teknik – teknik yang digunakan untuk melakukan bookmark pada aplikasi web berbasis AJAX.
3. Membangun perangkat lunak dengan tahapan sebagai berikut :
 - a. Analisis
Akan menganalisa pembuatan web dengan menggunakan AJAX. Menganalisa teknik-teknik *bookmarking* yang akan digunakan. Menganalisa kebutuhan akan *AJAX Engine* yang akan digunakan dan bahasa *script* yang akan digunakan untuk pembuatan aplikasi web berbasis AJAX.

- b. Perancangan
Pada tahap ini akan dibuat perancangan dari aplikasi yang akan dibuat mulai dari bahasa script yang akan digunakan, *AJAX Engine* yang akan digunakan, sampai dengan interface untuk aplikasi dan teknik-teknik *bookmarking* yang akan digunakan.
 - c. Implementasi
Akan diimplementasikan sebuah aplikasi web berbasis AJAX yang mendukung *Rich Internet Application* sebagai sarana untuk menerapkan teknik-teknik *bookmarking*.
 - d. Pengujian
Metode pengujian dilakukan dengan tujuan untuk mendapatkan kesalahan-kesalahan pada pada teknik *bookmarking* yang diimplementasikan. Serta untuk membuktikan bahwa teknik-teknik *bookmarking* sudah berjalan sesuai dengan analisa dan perancangan yang telah dilakukan.
4. Penyusunan laporan