

Rekayasa Web

Disiapkan oleh: Umi Proboyekti, S.Kom, MLIS

Pengantar

Rekayasa web adalah proses yang diunakan untuk menciptakan aplikasi web yang berkualitas tinggi. Rekayasa web mengadaptasi rekayasa perangkat lunak dalam hal konsep dasar yang menekankan pada aktifitas teknis dan manajemen. Namun demikian adaptasi tidak secara utuh, tapi dengan perubahan dan penyesuaian. Rekayasa web gabungan antara web publishing (suatu konsep yang berasal dari printed publishing) dan aktifitas rekayasa perangkat lunak. Dikatakan demikian karena desain sebuah aplikasi web menekankan pada desain grafis, desain informasi, teori hypertext, desain sistem dan pemrograman.

Ciri dan sifat WebApp (Web Application)

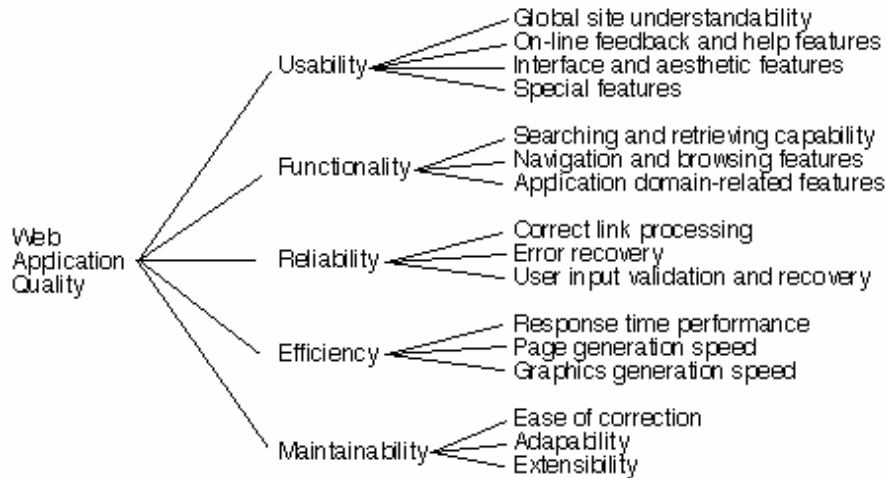
Aplikasi web berbeda dari software lain karena hal-hal dibawah ini:

1. *Network intensive*. Sifat dasar dari WebApp (aplikasi web) adalah aplikasi ini ditujukan untuk berada di jaringan dan memenuhi kebutuhan komunitas yang berbeda.
2. *Content-Driven*. Sebagian besar fungsi dari WebApp adalah untuk menyajikan informasi dalam bentuk teks, grafik, audio dan video ke end user.
3. *Continuous evolution*. Selalu berkembang secara terus menerus.
4. *Document-oriented*. Halaman-halaman situs yang statis akan tetap ada sekalipun sudah ada pemrograman web dengan java atau yang lain.

Selain itu WebApp memiliki karakteristik seperti berikut ini :

1. *Immediacy*. Diperlukan segera untuk memenuhi ditayangkan, dipasarkan dalam waktu singkat.
2. *Security*. Untuk melindungi isi yang sensitif dan menyediakan pengiriman data yang aman, keamanan suatu WebApp harus diterapkan pada seluruh infrastruktur yang mendukung WebApp dan termasuk dalam WebApp sendiri.
3. *Aesthetics*. Daya tarik utama WebApp adalah tampilan dan keindahan. Jika WebApp digunakan untuk memasarkan suatu produk maka sisi estetika harus diperhatikan sebagaimana sisi teknis.

Faktor-faktor yang menentukan kualitas suatu web digambarkan pada gambar 1.



Gambar 1 :Faktor kualitas aplikasi web

Faktor-faktor kualitas pada gambar 1 adalah faktor-faktor yang membantu web developer dalam merancang dan membangun webapp yang dapat diterima dan memenuhi kebutuhan end user yang begitu beragam.

Untuk memenuhi faktor-faktor kualitas tersebut, perancangan dan implementasi webapp terkait dengan 3 teknologi yang sangat penting yaitu: component-based development, security dan standart Internet. Seorang web developer harus mengenal 3 teknologi ini untuk membangun webapp yang berkualitas:

- Component-based development : CORBA,DCOM/COM dan JavaBeans merupakan standar yang memungkinkan web developer menggunakan komponen-komponen yang sudah ada untuk berkomunikasi dengan sistem pada level lain.
- Keamanan: enkripsi, dan firewall
- standart Internet: HTML, XML

Proses Rekayasa Web

Model yang dianggap cocok dan baik untuk rekayasa web adalah model modified waterfall dan spiral.

Modified waterfall

Tahapan dalam modified waterfall adalah :

1. problem definition dan concept exploration
2. requirement analysis specification
3. design prototyping

4. implementation and unit testing
5. integration and system testing
6. operation and maintenance

pada modified waterfall, perbedaan berada pada 2 proses pertama yang dilakukan secara berulang-ulang sehingga disebut *whirlpool*. Tujuannya adalah dapat melengkapinya requirement dan analisis secara lengkap.

Spiral

Pada spiral terbagi beberapa sektor yaitu :

1. determine site objectives and constraints
2. identify and resolve risks
3. develop the deliverables for the iteration and verify that they are correct
4. plan the next iteration

spiral model sangat masuk akal untuk rekayasa web tapi rumit dan sulit dalam pengaturan. Dibandingkan dengan waterfall, tahapan-tahapan pada spiral tidak jelas dimana mulai dan dimana akhir. Pada prakteknya spiral berguna selama perencanaan karena mengurangi resiko dan mendorong tim developer untuk memikirkan apa yang paling penting.

Formulasi dan Analisis sistem berbasis web

Formulasi dan analisis sistem dan aplikasi berbasis web adalah serangkaian aktifitas rekayasa web yang dimulai dengan identifikasi tujuan dan diakhiri dengan pembangunan analisis model atau spesifikasi requirement sistem.

Formulasi

Formulasi memungkinkan klien dan pembangun untuk menetapkan tujuan-tujuan pembangunan web. Beberapa pertanyaan berikut dapat membantu menentukan tujuan :

- Apa motivasi utama pembangunan WebApp?
- Mengapa WebApp diperlukan?
- Siapa yang akan menggunakan WebApp?

Ada dua macam tujuan:

- Informational goals—tujuan dari penyajian isi atau informasi kepada end
- Applicative goals—berkaitan dengan kemampuan yang dimiliki WebApp

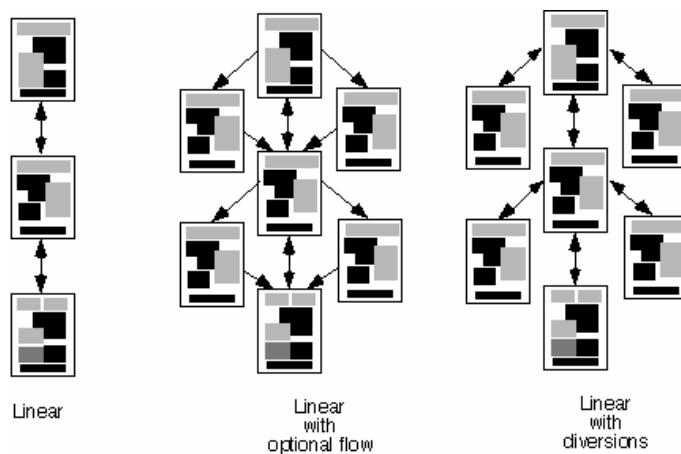
Analisis rekayasa web

Ada 4 tipe analisis dalam rekayasa web:

1. *Content Analysis*. Isi yang akan disajikan oleh WebApp dalam ditentukan formatnya baik itu berupa text, grafik dan image, video, dan audio.
2. *Interaction Analysis*. Cara interaksi antara user dan WebApp dijelaskan.
3. *Functional Analysis*. Menentukan operasi yang akan diaplikasikan pada WebApp dan termasuk di dalamnya fungsi-fungsi yang melakukan proses. Semua operasi dan fungsi dideskripsikan secara detail.
4. *Configuration Analysis*. Lingkungan dan infrastruktur dimana WebApp akan diberada digambarkan secara detail.

Desain Web

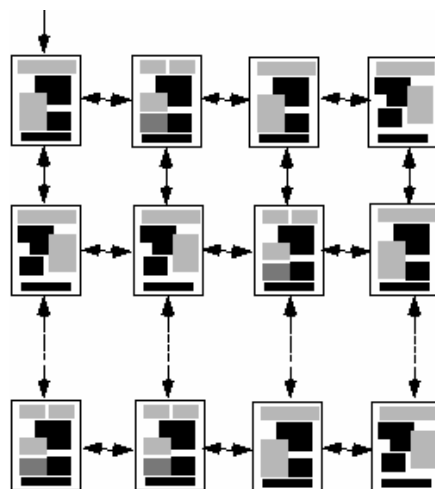
1. *Architectural design*: menggambarkan struktur WebApp



Struktur linier:

- urutan interaksi sudah bisa dipastikan
- misal untuk presentasi tutorial, pemesana produk yang harus mengikuti urutan tertentu

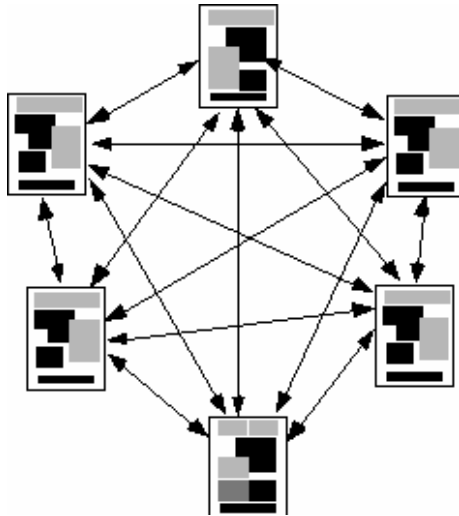
Gambar 2: Struktur linier



Struktur Grid

- isi dapat dikategorikan dalam 2 atau lebih dimensi
- misal: e-commerce menjual handphone. Horizontal adalah katagori berdasarkan feature hp, sedang vertikal adalah merek HP

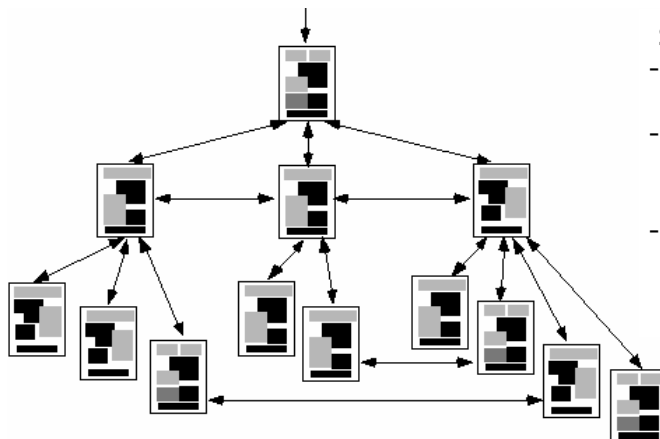
Gambar 3: Struktur Grid



Struktur jaringan:

- komponen pada struktur ini terhubung satu sama lain
- sekalipun bersifat fleksibel, struktur ini membingungkan user

Gambar 4: Struktur jaringan atau "pure web"



Struktur Hirarki:

- struktur paling umum digunakan
- memungkinkan aliran secara horizontal selain jalur vertikal yang umum
- aliran secara horizontal juga bisa mengakibatkan kebingungan user

Gambar 5: Struktur Hirarki

2. *Navigation design*: menentukan navigasi halaman-halaman web.

Setelah arsitektur WebApp sudah terbentuk dan komponen-komponen seperti halaman, scripts, applet dan fungsi lain sudah ada, developer menentukan navigasi yang memungkinkan user mengakses isi WebApp dan layanannya.

Jika user tidak bisa berpindah ke halaman lain dalam web dengan mudah dan cepat maka mungkin karena grafik, dan isi tidak relevant, ini masalah navigasi.

Dalam desain navigasi beberapa hal perlu dilakukan :

- menentukan semantik (arti) dari navigasi untuk user yang berbeda.
- menentukan cara yang tepat: pilihannya adalah text-based links, icons, buttons and switches, and graphical metaphors

3. *Interface design*: membangun interaksi dengan user yang konsisten dan efektif. User interface pada WebApp adalah kesan pertama. Sekalipun nilai isinya baik, kemampuan prosesnya canggih, layanannya lengkap namun jika user interfacenya buruk maka hal lain tidak berguna, karena akan membuat user berpindah ke web lain.

Beberapa petunjuk dalam merancang interface design :

- Server errors, menyebabkan user pindah ke website.
- Membaca di layar monitor lebih lambat 25% dari pada di kertas, karena itu teks jangan terlalu banyak.
- Hindari tanda "under construction".
- User tidak suka scroll. Pastikan informasi cukup dalam satu layar.
- Navigasi menu dan headbar harus konsisten.
- Keindahan tidak seharusnya lebih penting dari pada fungsinya
- Opsi navigasi harus jelas sehingga tahu bagaimana berpindah atau mencari hal lain pada halaman aktif.

Pengujian pada Rekayasa web

1. Check isi/informasi untuk kesalahan yang mungkin terjadi, misalnya salah ketik.
2. design model WebApp di- review untuk menemukan navigation errors.
3. processing components an Web pages diuji.
4. Integration test untuk arsitektur web :
 - Struktur linier, grid, atau hirarki sederhana—seperti pada software dengan pemrograman terstruktur (modular).
 - Struktur hirarki campuran atau network (Web) — seperti pada Object oriented software.
5. Uji WebApp secara keseluruhan setelah disatukan semua komponennya secara lengkap.
6. WebApp yang diimplementasikan pada konfigurasi yang berbeda diuji kompatibilitasnya. Misalnya jika membuat di IE, coba di Netscape, dan Firefox
7. WebApp diuji oleh sekelompok pengguna dengan kemampuan yang berbeda. Bagian yang diuji adalah isi, navigation, kemudahan penggunaan, kehandalan dan unjuk kerja.

Diadaptasi dari:

1. Pressman, Roger.S. "Software Engineering : A Practioner's Approach." 5th . McGrawHill. 2001.
2. Powell, Thomas A. " Web Site Engineering". Prentice Hall. 1998.