

MODEL VISUALISASI ARSITEKTUR INFORMASI

Oleh : Hieronimus Edhi Nugroho

ABSTRAKSI

Berbagai model untuk memvisualkan pengembangan sistem informasi telah diajukan dan digunakan untuk berbagi proyek sistem informasi. Pengembangan sistem informasi berbasis web mempunyai karakteristik yang berbeda dengan sistem informasi lain. Karakteristik yang berbeda tersebut membutuhkan model visualisasi yang tepat. Kajian terhadap model DFD dan OOD menunjukkan bahwa karakteristik web tidak dapat ditunjukkan dengan jelas sehingga diperlukan model visualisasi yang berbeda. Penulis mengusulkan sejumlah model untuk menggambarkan situs web atau tepatnya arsitektur informasi yang merupakan dasar pengembangan situs web.

1. PENDAHULUAN

World Wide Web atau lebih dikenal sebagai Web merupakan salah satu perkembangan teknologi jaringan komputer. Pada awalnya, teknologi web dimaksudkan untuk menampilkan informasi (teks) statis secara baku tanpa bergantung kepada perangkat keras dan perangkat lunak yang digunakan oleh pemakai, akan tetapi pada perkembangannya teknologi ini berkembang sehingga memungkinkan pemakai memperoleh informasi secara dinamis. Informasi dinamis ini diperoleh dari hasil sebuah proses yang dilakukan di server.

Mengingat bahwa teknologi web pada dasarnya adalah cara mengatur tampilan informasi dan bukan dimaksudkan sebagai antarmuka pemakai maka teknologi ini mempunyai keterbatasan-keterbatasan dalam hal antarmuka pemakai. Tetapi dengan perkembangan yang terjadi saat ini, ada berbagai macam

aplikasi yang dijalankan di server web dan hasilnya diterima oleh pemakai sebagai informasi yang terformat.

Salah satu aspek yang penting dalam mengembangkan sebuah aplikasi adalah komunikasi antara pengembang dengan pemakai. Komunikasi ini dimaksudkan untuk menyamakan pemahaman antara pengembang dengan pemakai mengenai aplikasi yang akan dibangun.

Pada umumnya pengembang memodelkan aplikasi yang dibangun dengan menggunakan model visual. Untuk aplikasi non web sudah tersedia sejumlah model visual yang dapat digunakan, diagram-diagram tersebut antara lain : DFD, ERD, Flowchart, Structure Chart, UML dan sebagainya. Akan tetapi model visual tersebut ditujukan untuk pengembangan aplikasi non web sehingga karakteristik-karakteristik khusus dari

aplikasi web seringkali tidak tergambarkan dengan tepat.

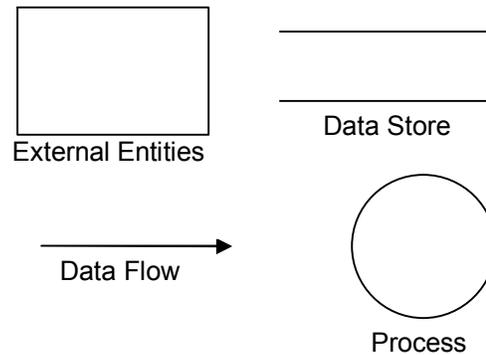
Penulis merasa tertarik untuk mengembangkan sebuah model visual yang secara tepat menggambarkan keseluruhan aspek dari pengembangan perangkat lunak berbasis web.

2. KAJIAN TERHADAP PENELITIAN SEBELUMNYA.

Pressman (2000) menunjukkan bahwa salah satu aspek yang cukup menentukan terhadap keberhasilan pengembangan sebuah sistem adalah komunikasi antara pengembang dengan pengguna. Pengembang harus dengan tepat menyampaikan gagasan dan pemahaman pengembang terhadap sistem yang akan dibangun sehingga tercapai pemahaman yang sama antara pengembang dan pengguna. Pada umumnya komunikasi gagasan tersebut diwujudkan melalui sejumlah model visual yang menggambarkan bagaimana sistem bekerja serta siapa yang terlibat dalam sistem. Model visual yang digunakan bergantung terhadap model pengembangan sistem yang digunakan oleh pengembang. Dalam bagian ini penulis akan menunjukkan dua pendekatan (metoda) utama yang digunakan dalam mengembangkan sebuah sistem, yaitu : SAD (*Structured Analysis and Design*) serta OOA/OOD (*Object Oriented Analysis / Object Oriented Design*). Kajian ditekankan terhadap model visual yang digunakan dalam masing-masing metoda, dalam kajian tersebut akan dilihat pula

mengapa model visual yang dikembangkan di dua metoda tersebut tidak cocok digunakan untuk menggambarkan pengembangan aplikasi web.

DeMarco (1979) telah mengembangkan metodologi pengembangan aplikasi yang dinamakan Structured Analysis (SA) yang kemudian berkembang menjadi Structured Analysis and Design (SAD). DeMarco mengembangkan sejumlah simbol untuk menggambarkan berbagai aspek dari aplikasi. Simbol-simbol tersebut memperlihatkan bagaimana data berubah setelah melewati berbagai proses, selain itu simbol-simbol tersebut juga dapat menggambarkan kaitan antara data dengan pemasok data maupun penerima data. Kumpulan simbol yang menggambarkan apa yang dilakukan oleh aplikasi dan pengaruhnya terhadap data dinamakan Data Flow Diagram (DFD). DFD dimaksudkan untuk menggambarkan hasil analisa. Hasil analisa tersebut kemudian akan digunakan sebagai dasar dalam merancang struktur program pada tahap perancangan. Pada tahap perancangan tersebut proses-proses yang berhubungan sangat erat dikelompokkan dan kemudian ditentukan hirarki dari proses, hirarki proses digambarkan dengan menggunakan structured chart. Modul-modul program serta komunikasi antar modul dikembangkan berdasarkan apa yang ditemukan pada *structured chart*. Simbol-simbol yang digunakan dalam DFD beserta arti dari simbol tersebut diperlihatkan pada Gambar 1.



Gambar 1. Simbol-simbol dalam Data Flow Diagram (DFD) (Yourdon, 1988)

Pada sekitar tahun 1980 muncul perkembangan baru dalam membuat program yaitu dengan dikenalkannya konsep *object oriented* (OO) (Meyer, 1988). Pada SAD definisi data dengan program merupakan hal yang berbeda sementara pada OO definisi data disatukan dengan definisi program. Program yang dibuat dengan menggunakan konsep OO disusun berdasarkan sekumpulan objek. Definisi umum mengenai objek disebut kelas. Beberapa konsep penting tentang OO antara lain :

1. **Inheritance atau Pewarisan.** Konsep ini menyatakan bahwa sebuah kelas dapat menurunkan atribut dan metodenya ke kelas lain.
2. **Encapsulation atau Pengkapsulan.** Definisi data dan modul berada di satu tempat yang sama sehingga kaitan antara data terhadap modul terlihat sangat jelas.
3. **Polymorphism.** Sebuah pemanggilan nama modul akan menggunakan implementasi yang berbeda

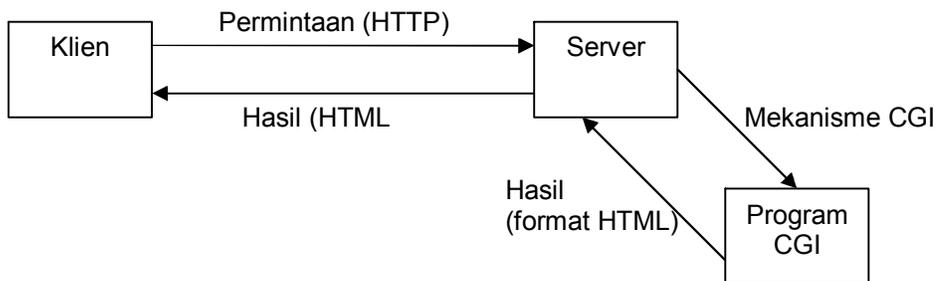
bergantung kepada objek yang aktif saat dilakukan pemanggilan tersebut. Konsep polymorphism membuat pemrogram tidak perlu membuat nama modul yang berbeda untuk menggambarkan fungsi-fungsi yang sama tetapi berbeda pada implementasinya.

Aplikasi yang dikembangkan dengan menggunakan pendekatan objek dapat digambarkan dengan berbagai macam cara, antara lain : Fusion, Booch, OMT, UML. Saat ini belum ada kesepakatan mengenai model visual yang harus digunakan untuk menggambarkan aplikasi dengan pendekatan objek meskipun faktanya banyak orang menggunakan Unified Modelling Language (UML) sebagai model visual.

UML menggunakan tiga macam diagram untuk menggambarkan hasil analisa dan perancangan. Ketiga diagram tersebut adalah : Use Case, Class Diagram, Sequence Diagram. Use Case digunakan untuk menggambarkan kaitan antara aktor

dengan kegiatan yang dilakukan oleh tiap aktor dan hubungannya dengan aktor lain. Class Diagram digunakan untuk menggambarkan struktur kelas serta kaitan antar kelas. Sequence Diagram merupakan diagram yang menggambarkan urutan event (kegiatan).

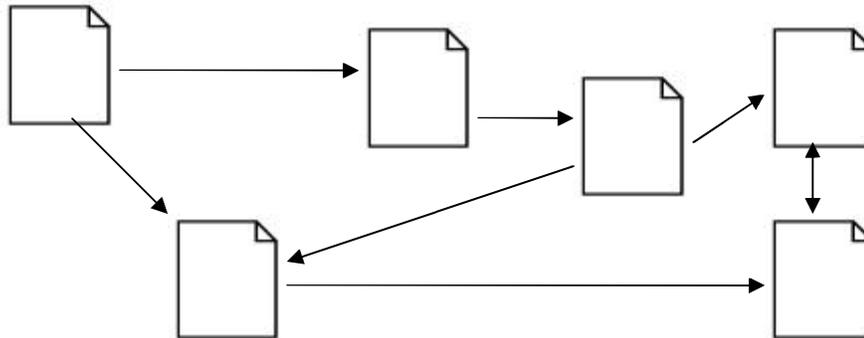
Aplikasi Web didefinisikan sebagai aplikasi yang dijalankan melalui sebuah perantara yaitu web server. Perbedaan komunikasi antara pemakai (klien) dengan server untuk aplikasi non web dengan aplikasi web diperlihatkan pada Gambar 2.



Gambar 2. Mekanisme Komunikasi Klien dengan Server Web (December, 1999)

Dari Gambar 2 terlihat bahwa aplikasi web mendapat data dari klien melalui web server dan memberikan hasil proses ke klien juga melalui web server. Data dikirim dalam bentuk untaian data (*stream*) dengan format seperti disyaratkan pada protokol HTTP (Fielding, 1997) dan hasil proses dikirim dengan format HTML (HyperText Markup Language) dan diterima oleh pemakai sebagai halaman web.

December (1999) menunjukkan bahwa pada aplikasi web pemakai melakukan navigasi (penelusuran) dengan menggunakan sejumlah *link* (kaitan). *Link* didefinisikan sebagai teks atau gambar yang apabila diklik akan membawa pemakai ke lokasi yang berbeda. Ini berarti bahwa pada aplikasi web pemakai bergerak dari satu informasi ke informasi lain, pergerakan informasi tersebut digambarkan oleh December seperti pada Gambar 3.



Gambar 3. Pergerakan Informasi (December, 1999)

Informasi yang diperoleh oleh pemakai dapat berupa informasi statis atau informasi dinamis. Informasi statis adalah informasi yang perubahannya terjadi karena memang sengaja untuk diubah sedangkan informasi dinamis adalah informasi yang isinya bergantung kepada data yang dikirim oleh klien.

Beberapa peneliti (Sacha, 2002; Hurst, 2000; Instone, 2000; Lash, 2002) telah menunjukkan bahwa arsitektur informasi merupakan hal yang sangat penting dalam merancang aplikasi web. Pengetahuan mengenai arsitektur informasi akan menjadi modal bagi perancangan navigasi (Mc Govern, 2001; McLaughlin, 2001). Arsitektur Informasi merupakan pengetahuan dan seni mengorganisasikan informasi sehingga pengunjung dapat dipenuhi kebutuhan informasinya secara tepat dan cepat (Shiple, 2001; West, 2002; Wyllys, 2000). Arsitektur informasi mengelompokkan informasi dan mengatur arus informasi sehingga pemakai dapat menemukan informasi dengan cepat. Studi yang dilakukan oleh Resenfeld (2001) maupun oleh Schleicher (2001) menunjukkan

bahwa pemanfaatan Arsitektur Informasi pada perancangan aplikasi web membuat tingkat kepuasan pengunjung halaman web menjadi lebih baik dibanding tanpa menggunakan Arsitektur Informasi.

Persoalan muncul ketika pengembang akan menggambarkan arsitektur informasi tersebut. Gerret (2002) mengusulkan sejumlah model visual yang dapat digunakan untuk menggambarkan arsitektur informasi, akan tetapi model tersebut mempunyai kelemahan yaitu tidak dapat menggambarkan perbedaan antara informasi statis dan informasi dinamis. Selain itu model visual yang diusulkan oleh Gerret juga tidak dapat menggambarkan *eksternal link*.

Meskipun perubahan data dalam *Data Flow Diagram* dapat dianggap sebagai navigasi, karena pemakai pindah dari satu posisi data (informasi) ke data lain (informasi lain), akan tetapi DFD tidak dapat menggambarkan perpindahan ke informasi lain tanpa melalui sebuah proses (informasi statis). DFD juga tidak dapat menggambarkan konsep mengenai *eksternal link*.

Kritik utama untuk diagram-diagram OO sebagai model visual bagi aplikasi web adalah halaman web berbeda dengan objek atau kelas. Halaman web tidak dapat memenuhi dua prinsip utama dalam OO. Meskipun halaman web dapat menerapkan prinsip *polymorphism* dalam batas-batas tertentu, misalnya : halaman web yang berisi daftar mata kuliah akan ditampilkan secara berbeda bergantung kepada nomor induk dari mahasiswa yang melihat halaman tersebut, tetapi halaman web tidak dapat memenuhi prinsip pewarisan dan pengkapsulan. Halaman web tidak dapat mewariskan isinya ke halaman lain.

Berdasarkan kajian diatas dapat disimpulkan bahwa model visual yang saat ini digunakan dalam dua metoda pengembangan sistem tidak cocok digunakan untuk menggambarkan pengembangan aplikasi web. Untuk itu perlu dicari alternatif model visual yang dapat secara tepat menggambarkan seluruh aspek pengembangan aplikasi web agar komunikasi antara pengembang dan pengguna dapat dilakukan dengan baik.

3. Usulan Model Visualisasi

Gerret (2002) mengusulkan sejumlah model visual untuk menggambarkan arsitektur informasi. Konsep yang mendasari usulan Garret adalah :



<Nama>



<Nama>

1. Sistem menunjukkan jalur (*paths*) kepada pemakai.
2. Pemakai berjalan sepanjang jalur melalui sejumlah aksi (*actions*)
3. Aksi tersebut menyebabkan sistem menghasilkan sejumlah hasil (*results*)

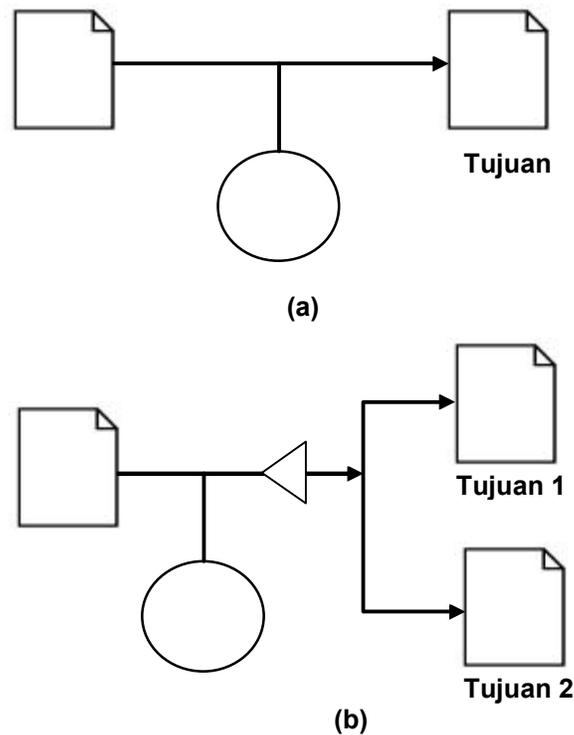
Meskipun model visual yang diusulkan oleh Garret sudah dapat digunakan dalam menggambarkan arsitektur informasi, akan tetapi model tersebut mempunyai kelemahan dimana diagram yang digunakan tidak dapat menunjukkan relasi antara kelompok informasi dengan proses yang dibutuhkan untuk menghasilkan informasi tersebut.

Dengan mendasarkan pada konsep yang disajikan oleh Garret maka penulis mengusulkan model visual yang dapat menghubungkan kelompok informasi dengan proses yang diperlukan untuk menghasilkan halaman web tersebut. Informasi yang berisi informasi statis digambarkan sebagai sebuah halaman seperti diperlihatkan pada Gambar 4.a Apabila Informasi mempunyai informasi yang lebih rinci maka kelompok informasi tersebut dapat digambarkan dengan menggunakan komponen pada Gambar 4.b

Gambar 4. (a) kiri: Simbol Kelompok Informasi Tunggal; (b) kanan: Simbol Kelompok Informasi Jamak

Kelompok informasi yang berisi informasi dinamis digambarkan seperti kelompok informasi statis tetapi dengan menghubungkan kelompok informasi tersebut ke proses yang diperlukan untuk menghasilkan kelompok informasi itu.

Diagram yang digunakan diperlihatkan pada Gambar 5.a Apabila sebuah proses menghasilkan dua atau lebih kemungkinan hasil maka dapat digunakan tanda segitiga untuk menunjukkan kemungkinan yang muncul. (Gambar 5.b)



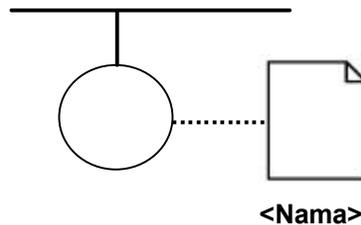
Gambar 5. (a) Informasi dinamis yang dihasilkan melalui sebuah proses ; (b) Proses yang menghasilkan kemungkinan dua informasi

Informasi yang bersifat dinamis seringkali diimplementasikan menggunakan template (pola). Keuntungan dari pemakaian template antara lain :

1. Menyediakan antar muka yang baku.
2. Mempersingkat waktu pengembangan

3. Memudahkan perubahan tampilan informasi.

Untuk menggambarkan bahwa sebuah proses menggunakan template maka proses tersebut dihubungkan ke diagram halaman dengan menggunakan sebuah garis putus-putus seperti diperlihatkan pada Gambar 6.



Gambar 6. Pemakaian Template

Jalur informasi digambarkan sebagai sebuah panah dengan arah panah menunjukkan arah informasi berikutnya

yang dapat diakses oleh pemakai (Gambar 7).



Gambar 7. Komponen Arah informasi

Keterangan mengenai kondisi yang menyebabkan jalur tersebut dipilih dapat diletakkan di atas atau di bawah tanda panah.

Situs eksternal adalah situs yang berada di luar situs yang sedang diakses oleh pemakai. Halaman yang berada di situs eksternal digambarkan dengan menggunakan *rounded rectangle* seperti dicontohkan pada Gambar 8. Sebuah

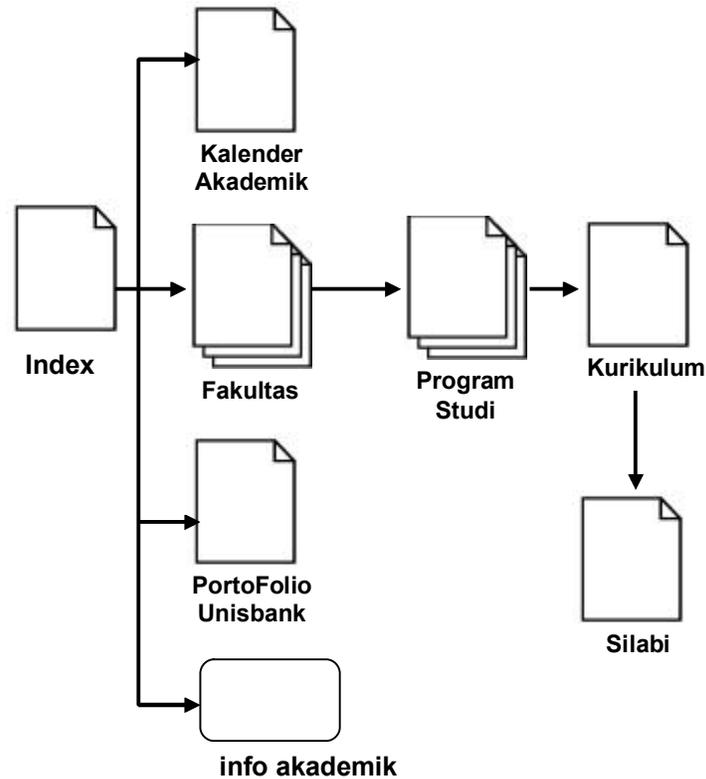
halaman dianggap berada di situs eksternal apabila alamat URL dari situs tersebut tidak sama, sebagian atau seluruhnya, dari alamat situs yang sedang diakses. Sebagai contoh alamat URL : <http://www.unisbank.ac.id/info> akan dianggap sebagai situs eksternal apabila pemakai sedang mengakses halaman web yang berada di alamat : http://www.unisbank.ac.id/info_dosen



Gambar 8. Komponen Situs Eksternal

Sebagai contoh pemakaian diagram arsitektur informasi dapat dilihat pada Gambar 9. Diagram pada Gambar 9

merupakan diagram arsitektur informasi dari situs Universitas Stikubank.

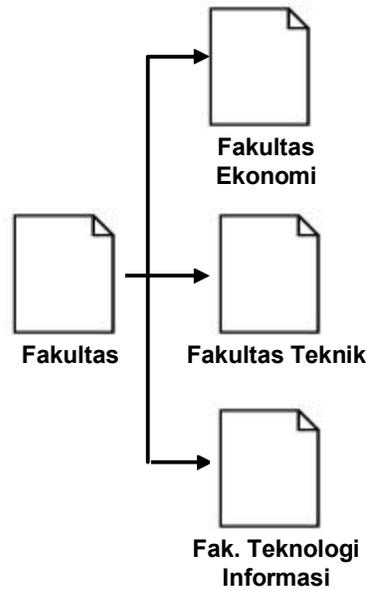


Gambar 9. Diagram Arsitektur Informasi Situs Unisbank

Untuk menghindari kerumitan yang tidak perlu maka diagram arsitektur informasi dapat dibagi menjadi beberapa tingkat dengan tiap tingkat menunjukkan rincian dari kelompok informasi sebelumnya. Komponen informasi yang akan dirinci digambarkan dengan

menggunakan simbol komponen informasi jamak.

Sebagai contoh kelompok informasi Fakultas pada Gambar 10 masih dapat dirinci lagi menjadi diagram seperti diperlihatkan pada Gambar 10.



Gambar 10 Diagram Rinci untuk Kelompok Informasi Fakultas.

4. KESIMPULAN

Dari pembahasan dapat diambil kesimpulan bahwa usulan mengenai model visualisasi arsitektur informasi telah dapat dengan tepat menggambarkan struktur

situs web. Model yang diajukan juga dapat digunakan untuk menunjukkan perubahan halaman web sesuai dengan kegiatan yang dilakukan oleh pengunjung.

5. DAFTAR PUSTAKA

- , 2002, *eCommerce Development Methodology*, The Technology Backbone Corp.
- , 2002, Heavyweight vs. lightweight methodologies, <http://www.zdnet.com.au/builder/manage/project/story/0,20000035082.20270383,00.htm>
- Cohen, Sacha., 2002, *Becoming an Information Architect – an Interview with Mattie Langenberg*, <http://technology.monster.com/articles/infoarchitect/>
- December, John., 1999, *HTML Unleashed*, SAM Publishing.
- Gerret, Jesse James., 2002, *A visual vocabulary for describing information architecture and interaction design*, <http://www.jjg.net/ia/visvocab>
- Hagedorn, Kat., 2000, *The Information Architecture Glossary*, <http://argus-acia.com>
- Hurst, Mark., 2000, *About Information Architecture*, <http://www.goodexperience.com/columns/040300infoarch.html>
- Instone, Keith., 2000, *Information Architecture and Personalization*, <http://argus-acia.com>
- Katerattanakul, Pairin., Siau, Lincoln Keng., 1999, *Measuring Information Quality of Web Sites*, in Proceedings of International Conference on Information Systems pages 279-285, December 13 – 15, 1999, Charlotte, North Carolina, USA.
- Lash, Jeff., 2002, *The Age of Information Architecture*, http://www.digital-web.com/columns/anythinggoes/anythinggoes_2002_08.shtml
- Lee, Seung C., 1998, *IDM: A Methodology for intranet design*, in Proceedings of International Conference on Information Systems pages 51-67, December 13-16, 1998, Helsinki, Finland.
- Mc Govern, Gerry., 2001, *Information Architecture versus Graphic Design*, http://www.clickz.com/design/site_design/article.php/945631.
- McLaughlin, Mark., 2001, *Apply Usability Methodologies in Intranet Information Architecture in a Real World Context*, Internet Journal Magazines, July 2001
- McLaughlin, Mark., 2001, *Apply Usability Methodologies in Intranet Information Architecture in a Real World Context Part II*, Internet Journal Magazines, August 2001.

- Nielsen, Jakob., 1998, *Usability Testing of Advanced Web Concepts*, <http://www.sun.com/980113/sunonnet/concepts.html>
- Pressman, Roger., 2000, *Software Engineering: A Practitioner Approach*, 4nd Edition, Mc Graw Hill.
- Resenfeld, Louis., 2000, *Special Report: Design Usability – Seven Pitfalls to Avoid in Information Architecture*, Internet World Magazine, 2000.
- Schleicher, Dennis., Kush, Jennifer., 2001, *Retail Ecologies, E-Commerce, and Information Architecture*, <http://argus-acia.com>
- Shiple, John., 1998, *Information Architectural Tutorial*, <http://hotwired.lycos.com>
- Toub, Steve., 2000, *Evaluating Information Architecture – A practical guide to assesing web site organization*, <http://argus-acia.com>
- West, Aaron., 2002, *The Art of Information Architecture*, <http://www.iboost.com/build/backend/arch/644.htm>
- Whitley, Edgar A., 1998, *Method-ism in Practice: Investigating The Relationship between Method and Undersanding in Web Page Design*, in Proceedings of International Conference on Information Systems pages 51-67, December 13-16, 1998, Helsinki, Finland.
- Wyllys, R.E., 2000, *Information Architecture*, <http://www.gslis.utexas.edu/~138613dw/readings/InfoArchitecture.html>.