ISSN : 1978-0087

APLIKASI PENGAMBILAN INFORMASI SITUS BERITA SECARA OTOMATIS MENGGUNAKAN *REGULAR EXPRESSION* DENGAN PLATFORM MICROSOFT .NET

Ario Koesnan [1)], Yudhi Purwananto [2)], Faisal Johan Atletiko [2)]

[1, 2, 3)] *Teknik Informatika , FTIF, ITS, Surabaya, e-mail : Sherlock_h30@yahoo.com*

Abstract, *Mosts news website nowadays are rich-content websites where they do not only provide news but also advertisements. Many of the advertisement banners in a news website are large-sized files. Therefore, it would take more time to download a news webpages. The purpose of this research is to help user with limited internet bandwidth to abtain news-only content from a news websites omitting its advertisements. Many news websites have common structures in indexing and directories. Regular expression can be used to adopt the structure of a news websites. By using windows services, the application will run in background to download news-only content. The results of this experiments showed that this application can download the contents of a news websites in background and users can read them offline. File size to be downloaded can also be reduced significantly.*

Keywords : Regular expression, Windows Service, Background process, News Content

Belakangan ini, masyarakat banyak disuguhkan berbagai macam situs berita mulai dari yang *thin content* dengan kategori yang sedikit seperti *bintang-indonesia.com* sampai situs yang menjadi portal berita yang *rich content* masa kini dan sering diakses dengan bermacam-macam kategorinya semacam *detik.com*.

Berdasarkan pengamatan penulis, semakin lama situs *rich content* semakin memberikan ruang yang banyak untuk penempatan *banner* atau iklan dengan alasan untuk meningkatkan pendapatan situs. Sebagai akibatnya dalam mendapatkan informasi, pengguna harus terbebani oleh *banner* atau iklan tersebut. Sehingga kebanyakan pengguna hanya mengakses situs-situs berita tersebut untuk membaca berita terbaru yang umumnya tersaji melalui kepala berita (*headline*) berita pada halaman utama situs tersebut. Banyak diantara pengguna, terutama dari SOHO (*Small Office Home Office*) dengan *bandwidth* koneksi internet terbatas, jarang mengakses berita utama dari *headline* yang sebagian besar digunakan untuk men-*download* gambar-gambar maupun *banner-banner* iklan dari situs berita yang kebanyakan belum tentu berguna dan tidak berhubungan dengan berita yang dibaca.

Sejalan dengan itu, proses pengambilan informasi secara keseluruhan dari situs berita secara otomatis ini akan memberikan kemudahan dan keefisienan bagi pengguna dalam memperoleh berita yang lengkap tanpa harus mengakses langsung situs-situs berita tersebut. Proses pengambilan informasi secara otomatis ini juga sangat memudahkan pengarsipan berita dan memahami kronologis kejadian dari berita tersebut.

Tujuan pembuatan aplikasi ini adalah membuat perangkat lunak yang bekerja sebagai proses *background* (*background process*) yang melakukan pengambilan berita dari situs-situs berita yang telah ditentukan pengguna. Pengambilan dilakukan secara otomatis dan periodik tanpa harus mengakses langsung ke situs berita tersebut dengan melakukan penyaringan (*filtering*) terhadap situs berita sesuai skenario yang telah terdefinisi dalam *profile* yang dikenali melalui *regular expression*.

Permasalahan yang diangkat dalam aplikasi ini adalah :

Bagaimana membuat suatu sistem aplikasi berbasis *web service* untuk men-*download* secara otomatis informasi berita utama dari satu atau lebih situs berita.

Bagaimana merancang dan membuat suatu skenario melalui *regular expression* yang mampu menyaring informasi utama dari situs berita yang memiliki berbagai macam jenis struktur baik *file* maupun datanya. Bagaimana mempelajari dan mengetahui site map situs,

metode pengambilan informasi, serta memberikan saringan (*filter*) yang sesuai pada situs-situs berita yang berbagai macam jenisnya tersebut.

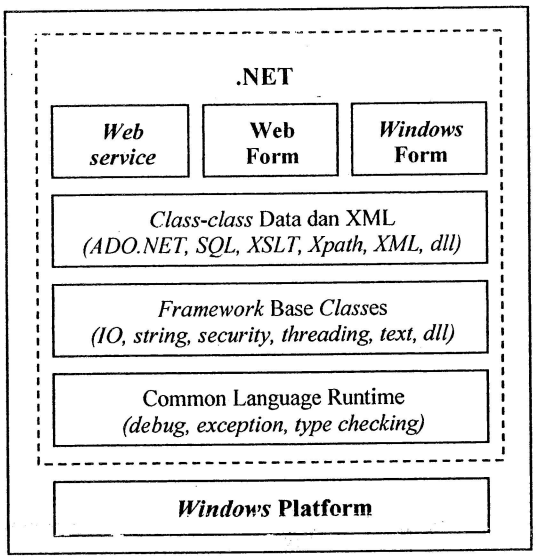
Manfaat dari pembuatan aplikasi ini adalah:

1. Memudahkan pengguna dengan bandwidth internet terbatas untuk mendapatkan berita secara cepat.
2. Pengguna dapat tetap memperoleh informasi berita terbaru tanpa perlu melakukan browsing ke situs berita.

FRAMEWORK .NET

Framework .NET adalah sebuah *platform computing* baru yang dirancang untuk menyederhanakan pengembangan aplikasi pada lingkungan terdistribusi pada internet [08]. Bahasa yang didukung oleh *Framework* .NET adalah Visual Basic.NET, C++, ASP.NET, Jscript.NET dan C#.

Pada gambar 1 terlihat framework .NET berada diatas sistem operasi, yang berupa sistem operasi buatan Microsoft, yaitu *Windows* 2000 (*Server* dan profesional), *Windows* NT 4.0 (*Server* dan workstation), *Windows* Millenium Edition, *Windows* 98 dan *Windows* XP Profesional.



Gambar 1. Framework .NET

WINDOWS SERVICE

Windows service sebelumnya disebut *NT service*. Fungsi utama dari suatu *windows service* adalah untuk menjalankan aplikasi sebagai *background*. *Windows service* mulai berjalan jauh sebelum pengguna melakukan *log*/masuk ke sistem (jika *windows service* tersebut diatur untuk mulai/*start* pada proses *boot-up*). Sebuah *windows service* juga dapat diatur sehingga pengguna harus melakukan *start* secara manual.

Windows service memiliki proses sendiri dan oleh karena itu ia dapat berjalan dengan sangat efisien. Secara normal, *windows service* tidak memiliki antarmuka.

WEB SERVICE

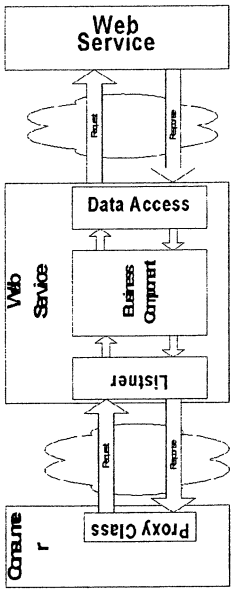
Suatu *Web service* adalah sebuah komponen yang menyediakan layanan untuk pelanggan yang menggunakan protokol internet standar (HTTP, XML) untuk mengakses layanan-layanan ini. *Web service* adalah komponen terprogram yang dapat diakses melalui protokol internet. *Web service* menggunakan XML dan *Simple Object Access Protocol* (SOAP) untuk berkomunikasi dengan pelanggan. XML menyediakan bahasa terstandarisasi untuk pertukaran data dalam format yang diterima secara luas. SOAP adalah protokol berbasis XML yang sederhana yang berjalan melalui HTTP untuk melakukan pertukaran data dalam lingkungan heterogen yang terdistribusi. Gambar 2 menunjukkan arsitektur dari *web service*.

Ketika menggunakan Visual Studio untuk membuat dan menggunakan *web service*, kita tidak perlu mengetahui arsitektur ini. .NET *Framework* mengkonversi semuanya, kita dapat membuat dan menggunakan *web service* tanpa terlebih dahulu harus mempelajari XML dan SOAP.

REGULAR EXPRESSION

Regular expression menyediakan metode yang kuat, fleksibel dan efisien untuk memproses teks [04]. Notasi pencocokan pola yang luas memungkinkan teks dengan ukuran besar dapat diparsing dengan cepat untuk menemukan pola karakter tertentu; untuk mengekstrak, mengedit, mengganti, atau menghapus *substring* teks; atau menambahkan *string* terekstraksi ke dalam koleksi untuk menghasilkan laporan. Untuk banyak aplikasi yang berhubungan dengan *string* (misalnya pemrosesan HTML, *parsing* log file, dan *parsing* HTTP header), *regular expression* merupakan *tool* yang diperlukan. Dengan desain yang disesuaikan dengan *regular expression* dari Perl 5, *Regular expression*

.NET *Framework* meliputi fitur pencocokan dari kanan ke kiri (right-to-left matching) dan kompilasi sambil jalan (on-the-fly compilation).



Gambar 2. Arsitektur Web Service

Bahasa *regular expression* didesain dan dioptimasi untuk memanipulasi teks. Bahasanya terdiri dari dua tipe karakter dasar: karakter teks literal (normal) dan metakarakter-metakarakter.

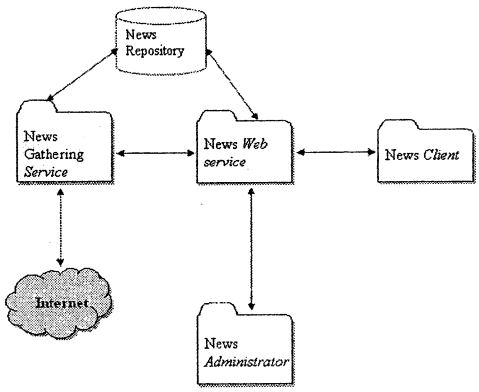
Metakarakter ? dan * adalah metakarakter yang sudah familiar dan digunakan dengan DOS file sistem untuk mewakili suatu karakter apapun atau kumpulan karakter. Perintah DOS COPY *.DOC A: memerintahkan sistem untuk menyalin file apa saja dengan ekstensi .DOC ke dalam disket di *drive* A. Metakarakter * mewakili nama file apa saja didepan ekstensi .DOC. *Regular expression* juga menyediakan pencarian yang lebih kompleks. Sebagai contoh, *regular expression* (?<char>\w)\k<char>, menggunakan grup bernama dan backreferencing, mencari karakter berdekatan yang berpasangan.

Framework .NET memiliki clas-class built-in untuk memproses *regular expression*, yaitu class **Regex, Match, MatchCollection, Group Collection, CaptureCollection, Group,** dan **Capture.**

METODOLOGI

Arsitektur Sistem

News Administrator mengatur segala hal yang berhubungan dengan *inserting*, *updating* dan *deleting* data ke *database*. Aplikasi inilah yang menangani data-data baru yang akan masuk ke sistem, seperti situs baru, alamat URL dan sebagainya. Demikian juga jika ada perubahan maupun penghapusan terhadap *database*. *News Web service* berfungsi sebagai *service* yang mengantarai *server* dan *client*. Fungsi-fungsi yang digunakan *client* umumnya yang berhubungan dengan akses ke *database*, sedangkan *server* menggunakan fungsi-fungsi untuk *parsing* (*filtering*). *News Client* merupakan aplikasi bagi pengguna (*client*) berisi antarmuka untuk membaca dan mencari berita. *News Gathering Service* berfungsi sebagai *server* yang bertugas mengambil berita secara berkala. Aplikasi ini merupakan *windows service* yang berjalan sebagai proses *background* pada saat mengambil berita. Gambar 3 menunjukkan arsitektur sistem



Gambar 3. Arsitektur Sistem

Pengguna Aplikasi

Ada tiga macam pengguna aplikasi, yaitu *Administrator User*, *Administrator Regex* dan *Client*.

1. ***Administrator User***
   Pengguna ini berfungsi sebagai operator yang bertugas memasukkan data-data ke dalam database, maka *skill* yang diperlukan adalah *entry* data.
2. ***Administrator Regex***
   Pengguna ini bertugas menganalisis situs berita dan membuat formula *regex*-nya. *Skill* yang diperlukan adalah mengerti *Regular Expression* dan HTML.
3. ***Client***
   Pengguna ini berfungsi sebagai *client* yang memerlukan informasi utama dari situs berita tanpa mengutamakan iklan/*banner*,

gambar, dan menu/*link* lainnya, hanya berita utamanya saja.

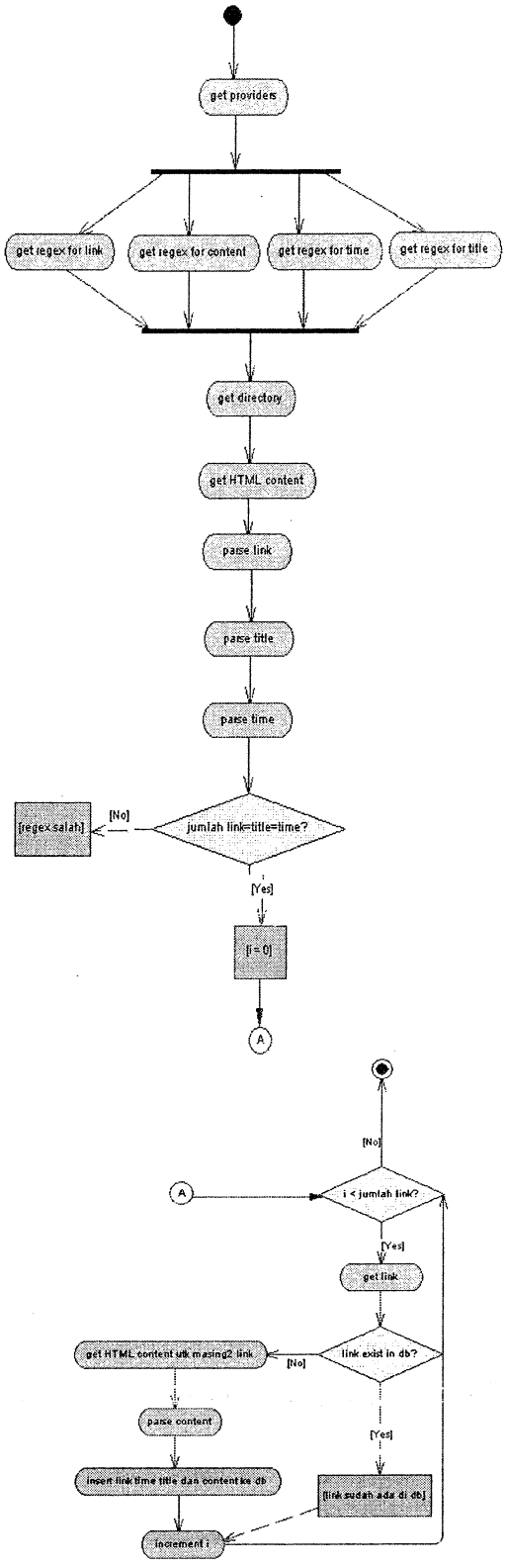
Perancangan Skenario Regular Expression

Ada beberapa langkah yang harus dilakukan sebelum merancang skenario ini, yaitu sebagai berikut:

1. ***Pengenalan Situs***
   Dalam tahap ini dilakukan pengenalan pola situs, mulai dari struktur situs, penamaan direktori, URL, dan lain-lain yang diperlukan bagi perancangan *regex* nantinya. Untuk mengenali suatu situs harus didapatkan *source HTML* dari situs tersebut. Umumnya diambil file yang mengandung indeks berita dari situs yang bersangkutan. Dari *source* ini kemudian diambil pola dengan melihat *tag-tag* khusus yang menentukan posisi *link* indeks berita, waktu berita, judul bahkan isi berita itu sendiri.
2. ***Pembuatan regex untuk link, time, dan title***
   Setelah mengetahui pola situs kita dapat membuat formula *regex* untuk memfilter aspek-aspek yang kita perlukan. Yang paling penting di sini adalah *filtering* terhadap *link* indeks berita, karena *link-link* inilah yang nantinya akan merujuk kepada isi/*content* beritanya. Untuk dapat mengambil *link-link* yang sesuai, harus diketahui dimana posisi mulai dan akhir dari *link-link* yang akan diambil. Selain *link-link* tersebut, dilakukan pula *filtering* untuk mendapatkan waktu berita (*time*) dan judul berita (*title*) juga dengan *regex* tertentu yang dibuat berdasarkan pengenalan pola situs yang telah dilakukan sebelumnya.
3. ***Pembuatan regex untuk content***
   Setelah mendapatkan *link-link* indeks berita yang akan diambil, dilakukan pengambilan content/isi berita juga dengan *regex* hasil pengenalan pola dari *link* yang merujuk ke berita utama tersebut.

Activity Diagram Pengambilan Berita

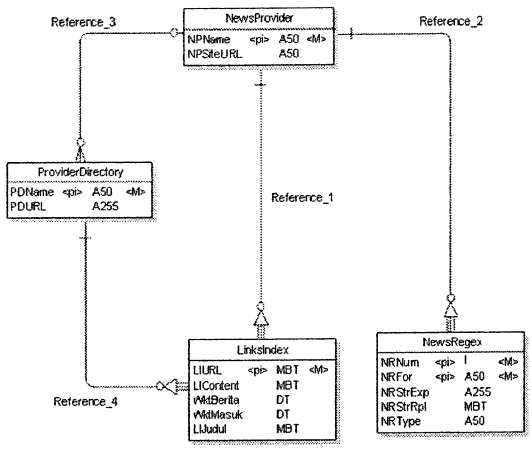
Diagram ini menunjukkan aliran proses utama yang terjadi pada perangkat lunak ini, yaitu proses pengambilan berita. Proses ini dilakukan sebagai proses background (*windows service*) pada *server*. Gambar 4 menunjukkan aliran proses yang terjadi saat *service* melakukan pengambilan berita.



Gambar 5. Activity Diagram Pengambilan Berita

Perancangan CDM dan PDM

Rancangan *conceptual diagram* (CDM) dari *database* sistem perangkat lunak ini, dapat dilihat pada gambar 6. Sedangkan diagram PDM ditunjukkan oleh gambar 7.

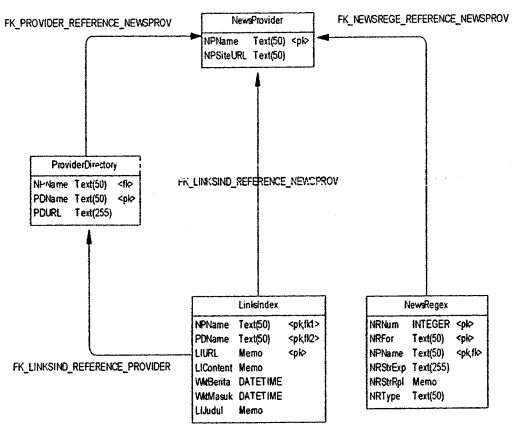


Gambar 6. Diagram CDM

PEMBAHASAN

Implementasi Rancangan Sistem

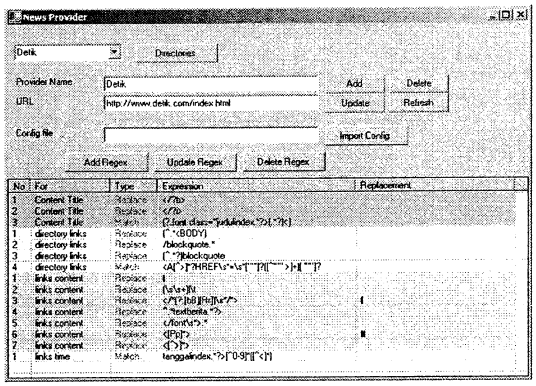
Untuk mengimplementasikan rancangan sistem *client-server* ini, digunakan tools Microsoft Visual Studio .NET dalam membuat *web service*. Bahasa pemrograman yang digunakan adalah VB.NET. Web service tidak menghasilkan bentuk antarmuka melainkan hanya API bagi sistem lainnya. Deskripsi API ini dijelaskan dalam bentuk WSDL. *Web service* ini dirancang dengan tujuan sebagai jembatan antara *Client* dengan *Server*. Metode lain yang biasa dipakai adalah *remoting/socket*.



Gambar 7. Diagram PDM

Aplikasi Administrator

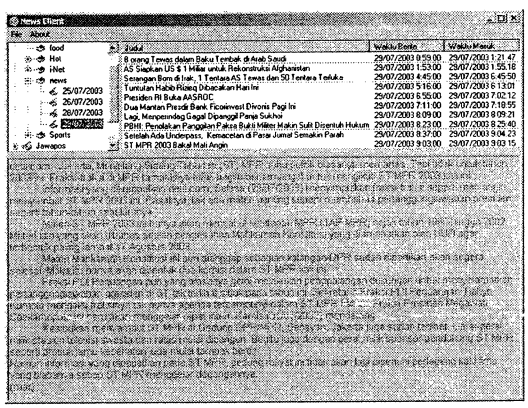
Aplikasi *Administrator* menggunakan *web service* untuk memanggil fungsi-fungsi untuk memanipulasi tabel-tabel dalam *database*, yaitu tabel *NewsProvider*, *ProviderDirectory*, dan *NewsRegex*. Tampilan awal dari aplikasi *Administrator* ditunjukkan pada gambar 8.



Gambar 8. Tampilan Aplikasi Administrator

Aplikasi Client

Aplikasi *Client* menggunakan *web service* untuk mengambil isi berita secara *offline* dari *database* untuk dibaca oleh *client*/pengguna. Aplikasi *Client* dibuat menyerupai Microsoft Outlook (*MS.Outlook-like*) untuk memudahkan interaksi dengan *client*/pengguna. Tampilan awal aplikasi ini ditunjukkan pada gambar 9.



Gambar 9. Tampilan Aplikasi Client

Uji Coba

Pada bagian ini akan dilakukan uji coba terhadap proses pengambilan berita. Situs-situs berita yang akan digunakan sebagai analisa

meliputi detik (www.detik.com), republika (www.republika.co.id) dan kompas (www.kompas.co.id). Analisa ini dilakukan untuk mengetahui apakah proses pengambilan berita berjalan dengan lancar dengan menghasilkan *content* yang benar dan sesuai untuk tiap-tiap direktori dari situs berita tersebut. Uji coba juga dilakukan untuk menguji apakah proses *client-server* dapat berjalan dengan benar dan akurat.

Pada uji coba ini dilakukan perancangan skenario *regex* dan pengambilan berita terhadap tiga situs berita yang dianggap mampu merepresentasikan situs-situs berita di Indonesia. Situs-situs tersebut adalah detik, kompas dan republika. Pengambilan berita ini sebenarnya dilakukan oleh *server* secara berkala dan dalam bentuk proses background. Namun dalam uji coba ini akan menggunakan fungsi di *web service* untuk dapat memantau keberhasilan mengambil informasi berita per penyedia situs.

Skenario uji coba yang lain adalah dengan membandingkan antara *browsing* ke situs secara langsung dibandingkan dengan memakai aplikasi ini. Tujuannya adalah membandingkan besar/ukuran *file* yang dihasilkan. Uji coba juga ini dapat dijadikan referensi untuk mengetahui bahwa aplikasi ini dapat menghemat *bandwidth*.

1) *Uji coba Situs www.detik.com*

Uji coba ini dilakukan pada situs detik dengan direktori *Finance*, *food*, *Hot*, *i-Net*, *news*,dan *Sports*. Gambar 9 dan 10 menunjukkan data direktori dan regex situs detik.

- Detik http://www.detik.com/index.html

PDName	PDURL
Sports	http://www1.detik.com/indeksberita/index.cfm?fuseaction=index.detikSports
news	http://www1.detik.com/indeksberita/index.cfm?fuseaction=index.detikNews
i-Net	http://www1.detik.com/indeksberita/index.cfm?fuseaction=index.detiki-Net
Hot	http://www1.detik.com/indeksberita/index.cfm?fuseaction=index.detikHot
food	http://www1.detik.com/indeksberita/index.cfm?fuseaction=index.detikFood
Finance	http://www1.detik.com/indeksberita/index.cfm?fuseaction=index.detikFinance

Gambar 9. Data Direktori Detik

NRNum	NREr	NPName	NRStrExp	NRStrRpl	NRType
1	1	Detik	(*.*<BODY)		1
2	1	Detik	/blockquote.*		1
3	1	Detik	(*.*?)blockquote		1
4	1	Detik	<A[^>]*?HREF\s*=\s*["']?([^" >]+)[^>]*		2
1	2	Detik	□		1
2	2	Detik	(\s\s+)[\t		1
3	2	Detik	</?(?:[bB][Rr])\s*/*>	□	1
4	2	Detik	*.*textberita.*?>		1
5	2	Detik	</font\s*>.*		1
6	2	Detik	<[Pp][^>	□□	1
7	2	Detik	<[^>]*[^>		1
1	3	Detik	tanggalindex.*?>(*0-9]*)[\[^<]*)		2
1	4	Detik	</?b>		1
2	4	Detik	</?i>		1
3	4	Detik	(?:font class="judulindex.*?>(.*?)<)		2

Gambar 10. Data Regex Detik

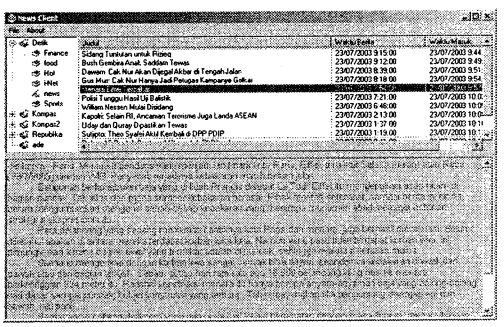
Setelah *web service* di-*run*, fungsi *parseDirectory* dipanggil, kemudian parameter-parameter dimasukkan, yaitu situs penyedia berita, direktori dan *includeContent* (untuk pilihan termasuk *content*/tidak) yang akan diambil *main content* beritanya. Setelah fungsi ini selesai, maka informasi berita telah berhasil masuk ke *database*, yaitu ke tabel *LinksIndex* untuk ditampilkan ke *client*. Gambar 11 menunjukkan table link dan data-datanya.

NPName	PDName	LIURL	LIContent
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-003800.shtml	detik.com - Yogyakarta, Meski
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-004051.shtml	detik.com - Jakarta, Duta Beso
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-011758.shtml	detik.com - Solo, Keinginan Th
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-013647.shtml	detik.com - Jakarta, Setelah be
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-021320.shtml	detik.com - Jakarta, Kapolri Jer
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-064507.shtml	detik.com - Banda Aceh, Menu
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-072015.shtml	detik.com - Jakarta, Polda Met
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-075228.shtml	detik.com - Paris, Menara leget
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-081818.shtml	detik.com - Jakarta, Budayawat
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-083816.shtml	detik.com - Jakarta, Menurut pe
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-091036.shtml	detik.com - Jakarta, Kabar tewa
Detik	News	http://www.detik.com/penstiwa/2003/07/23/20030723-091528.shtml	detik.com - Jakarta, Setelah be

Gambar 11. Tabel setelah *link* dan berita masuk

WktBerita	WktMasuk	LIJudul
23/07/2003 0:09:00	23/07/2003 10:18:07	DPD Golkar Yogya Tolak Calonkan Akbar Tandjung
23/07/2003 0:41:00	23/07/2003 10:15:44	Dubes AS Bantah Bantuan Militer untuk RI Ditunda
23/07/2003 1:19:00	23/07/2003 10:13:47	Sutjipto: Theo Syafei Aktif Kembali di DPP PDIP
23/07/2003 1:37:00	23/07/2003 10:10:56	Uday dan Qusay Dipastikan Tewas
23/07/2003 2:13:00	23/07/2003 10:08:50	Kapolri: Selain RI, Ancaman Terorisme Juga Landa ASEAN
23/07/2003 6:46:00	23/07/2003 10:05:33	William Nessen Mulai Disidang
23/07/2003 7:21:00	23/07/2003 10:02:21	Polisi Tunggu Hasil Uji Balistik
23/07/2003 7:54:00	23/07/2003 9:57:20	Menara Eiffel Terbakar
23/07/2003 8:18:00	23/07/2003 9:54:23	Gus Mus: Cak Nur Hanya Jadi Petugas Kampanye Golkar
23/07/2003 8:39:00	23/07/2003 9:51:57	Dawam: Cak Nur Akan Dijegal Akbar di Tengah Jalan
23/07/2003 9:12:00	23/07/2003 9:49:13	Bush Gembira Anak Saddam Tewas
23/07/2003 9:15:00	23/07/2003 9:44:13	Sidang Tuntutan untuk Rizieq

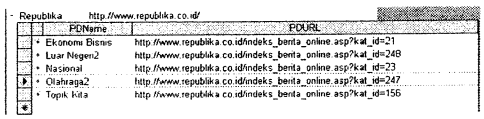
Gambar 12 menunjukkan aplikasi client ketika melihat isi berita.



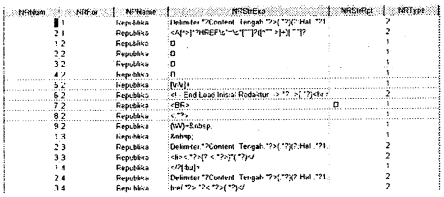
Gambar 12. Tampilan Aplikasi Client untuk Situs Detik

2) *Uji coba Situs www.republika.co.id*

Uji coba ini dilakukan pada situs republika dengan direktori *Ekonomi Bisnis*, *Luar Negeri*, *Nasional*, *Olahraga*, dan *Topik Kita*. Gambar 13 dan 14 menunjukkan data direktori dan regex situs republika.

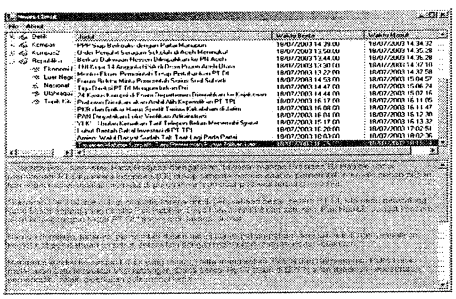


Gambar 13. Data Direktori Republika



Gambar 14. Data Regex Republika

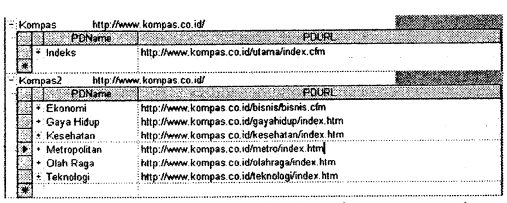
Gambar 15 menunjukkan aplikasi client ketika membaca berita.



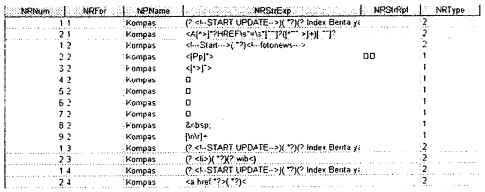
Gambar 15. Tampilan Aplikasi Client untuk Situs Republika

3) *Uji coba Situs www.kompas.com*

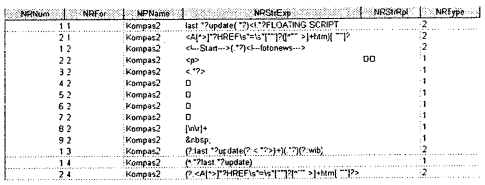
Uji coba ini dilakukan pada situs kompas dengan direktori *Ekonomi*, *Gaya Hidup*, *Kesehatan*, *Teknologi*, *Metropolitan*, *Olah Raga*, dan *Topik Kita*. Skenario *regex* untuk kompas ini sengaja dibagi menjadi dua, yaitu kompas dengan direktori *Indeks*, dan kompas2 dengan direktori-direktori seperti diuraikan di atas. Hal ini dilakukan karena pola antara indeks berita utama kompas dengan direktori-direktori lainnya berlainan. Gambar 16, 17, dan 18 menunjukkan data direktori dan regex situs kompas.



Gambar 16. Data Direktori Kompas dan Kompas2

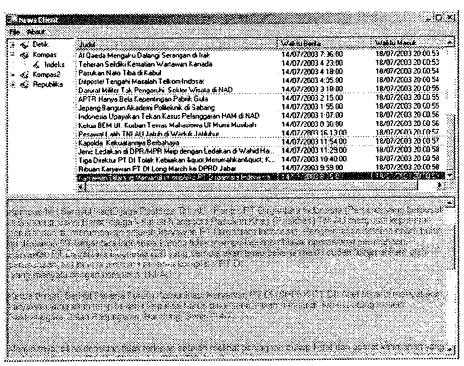


Gambar 17. Data Regex Kompas

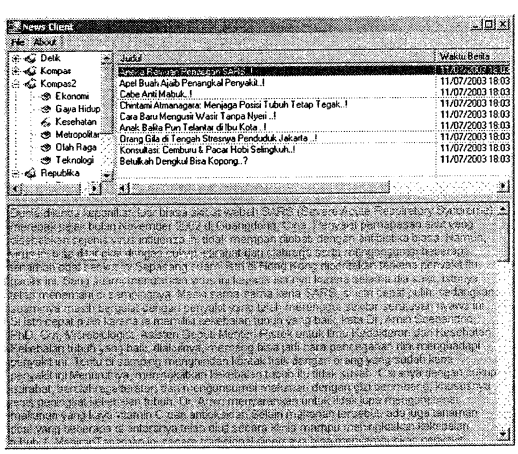


Gambar 18. Data Regex Kompas2

Gambar 19 dan 20 menunjukkan aplikasi client ketika mengakses informasi berita Kompas.



Gambar 19. Tampilan Aplikasi Client untuk Situs Kompas



Gambar 20. Tampilan Aplikasi Client untuk Situs Kompas2

4) Ujicoba perbandingan ukuran file

Telah dilakukan ujicoba untuk membandingkan ukuran file yang harus didownload jika pengguna langsung melakukan browsing ke sebuah halaman situs berita Detik dengan ukuran file berita (dalam ekstensi .txt) yang didownload oleh aplikasi ini. Untuk kasus browsing langsung, diperoleh data ukuran file yang didownload adalah 94,5 KB dengan *size-on-disk* adalah 184 KB. Sedangkan ukuran file berita yang didownload oleh aplikasi ini adalah 1,49 KB dengan *size-on-disk* adalah 4,00 KB.

SIMPULAN

1) Untuk mengambil informasi berita utama/*main content* dari suatu situs berita yang memiliki pola/struktur situs yang tetap atau memiliki web content management, digunakan suatu formula *Regular Expression* yang sesuai dengan pola situs masing-masing.
2) Perancangan dan pembuatan skenario *Regular Expression* dapat dilakukan dengan mempelajari struktur/pola suatu situs.
3) Formula *Regular Expression* dirumuskan dalam satu *file* konfigurasi untuk setiap penyedia berita oleh *Administrator Regex* yang kemudian *file* konfigurasi tersebut akan diimport dan dimasukkan ke database melalui aplikasi *News Administrator* oleh *Administrator User*.
4) Satu situs penyedia berita dapat memiliki lebih dari satu pola/struktur situs sehingga memiliki lebih dari satu formula *Regular Expression*.
5) Teknologi *Windows Service* digunakan sebagai aplikasi yang berjalan dalam proses *background* untuk melakukan pengambilan berita secara otomatis.
6) Teknologi *Web Service* dengan protokol HTTP dapat menjadi penghubung antara *Server* dan *Client*.

Untuk pengembangan :

1) Penggunaan *pattern recognition* untuk dapat mengenali pola/struktur situs secara dinamis.
2) Penelitian lebih lanjut untuk dapat menghasilkan formula *Regular Expression* secara otomatis.

DAFTAR RUJUKAN

[01] Haryanto, Steve. *Majalah Masterweb MWMag*. Indonesia: Oktober 2001, November 2001, *issue* 04, *issue* 06

[02] Jain, Jayesh. *Creating A Windows Service in VB.NET*. www.devarticles.com. Oktober 2002

[03] JISC TechWatch Report: *Content Management Systems*. http://www.jisc.ac.uk/techwatch/reports/tsw_01-02.pdf. September 2001

[04] Microsoft Service Digital Network (MSDN). *.NET Framework Developer's Guide .NET Framework Regular Expressions*, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconcomregularexpressions.asp, 2002

[05] *Regex Library*; www.regexlib.com

[06] Regular Expresison di .NET; http://www.techmedia-online.com/tutorials/t02070101.asp?id=1953

[07] Rockford Lhotka. *Implementing a Background Process in Visual Basic .NET*, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnadvnet/html/vbnet09272002.asp, Magenic Technologies, 2002

[08] Thai, Thuan, Huang Q. Lam. *.NET Framework Essential*. 1st Edition, O'Reilly, 2001

[09] *VB.NET Fast Track*. Syngress, 2001

[10] Situs-situs berita; www.republika.co.id, www.jawapos.co.id, www.kompas.co.id, www.surya.co.id, www.detik.com, www.reuters.com, www.astaga.com, news.bbc.co.uk dan situs-situs berita lainnya