

Rancang Bangun Aplikasi Pembangkit Halaman HTML untuk Pembuatan Media Cetak Digital Berbasis Windows Metro Style

Sally Indah Khansa, Dwi Sunaryono, Umi Laili Yuhana

Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

e-mail: dwi@its-sby.edu

Abstrak—Aplikasi dibangun dengan memanfaatkan Windows Metro Style agar dapat dijalankan pada sistem operasi Windows 8. Input dari aplikasi pembangkit halaman ini berupa HTML. Aplikasi ini menyediakan editor halaman HTML agar pengguna dapat mengubah isi halaman tanpa harus membuat ulang media dan juga dapat menampilkan daftar media yang pernah dibuat sebelumnya agar dapat dibuka sewaktu-waktu. Pengujian aplikasi dilakukan dengan mengunduh folder yang berisi file HTML, file HTML tersebut sebelumnya harus sudah diurutkan terlebih dahulu jika terdapat lebih dari satu file, contohnya: 01.html, 02.html, dll. File HTML yang sudah diinputkan, dibaca dan digabungkan di dalam aplikasi dengan menggunakan metode HTML tag extraction method. Metode tersebut bertujuan untuk mencari tag <body>, <script>, <h1>, dll. Halaman yang sudah digabungkan ditambahkan skrip menggunakan *plugin* BookBlock dan Booklet agar menyerupai tampilan atau format membuka pada buku maupun majalah. Output aplikasi berupa tampilan media cetak digital yang berjalan pada Windows 8. Dapat dijalankan menggunakan komputer pribadi maupun komputer tablet dengan sistem operasi Windows 8. Karena output aplikasi berupa halaman HTML, halaman tersebut dapat ditampilkan dalam *browser* dengan macam-macam perangkat. Akan tetapi, untuk keberhasilan tampilan halaman tergantung pada *browser* perangkat dan layar monitor perangkat.

Kata Kunci—HTML tag extraction method, media cetak digital, metro app, pembangkit halaman HTML.

I. PENDAHULUAN

Berkembangnya teknologi baru yang bermunculan menciptakan perangkat-perangkat baru yang bertujuan untuk mempermudah manusia menjalankan aktifitasnya seperti berkomunikasi, mencari informasi, dll. Untuk mencari informasi sudah dapat dilakukan menggunakan perangkat, misalnya saja membaca. Membaca yang sebelumnya hanya dapat dilakukan dengan menggunakan media cetak sekarang sudah dapat dilakukan dengan menggunakan perangkat. Salah satu contohnya adalah majalah The Word Magazine yang dijual di <http://itunes.apple.com>.

Dengan banyaknya peminat membaca berita dengan menggunakan perangkat, banyak *developer* membuat aplikasi untuk mempermudah pembuat berita membuat media cetak digital tersebut. Salah satu contohnya adalah Pugpig. Pugpig dikembangkan untuk aplikasi publikasi *platform* HTML5

yang memungkinkan pengguna untuk membuat surat kabar, majalah dan buku-buku secara digital yang dapat ditampilkan untuk iPad, iPhone, dan Android.

Oleh karena aplikasi Pugpig belum dapat diakses melalui *desktop* dan hanya menggunakan perangkat tertentu, maka pada TA ini dibuat sebuah aplikasi pembangkit halaman HTML dikembangkan dengan menggunakan Windows Metro Style. Metro Style sendiri adalah inovasi baru yang ada di Windows 8 yang digunakan untuk membuat model tampilan di Windows 8. Aplikasi pembangkit halaman ini dapat berjalan dengan sistem operasi Windows 8. Hasil dari aplikasi ini dapat ditampilkan di Windows 8 akan tetapi tidak menutup kemungkinan dapat berjalan di sistem operasi dan perangkat lain melalui *browser*.

II. DASAR TEORI

A. Windows Metro Style

Metro Style merupakan perbedaan yang terlihat antara Windows 8 dan Windows versi sebelumnya. Windows versi sebelumnya hanya memiliki aplikasi *desktop*, sedangkan Windows 8 memiliki aplikasi *desktop* dan Metro. Aplikasi Metro ditampilkan pada satu layar penuh. Untuk mengembangkan aplikasi Metro, pada Visual Studio 2012 sudah terintegrasi dengan beberapa bahasa pemrograman. Bagi yang terbiasa dengan HTML5, CSS dapat menggunakan JavaScript untuk mengembangkan aplikasi Metro sedangkan bagi yang terbiasa dengan *framework* .NET dapat menggunakan XAML dengan menggunakan bahasa C#, VB atau C++ [1].

B. HTML Tag Extraction Method

HTML yang dikenal sebagai Hypertext Mark-up Language adalah bahasa untuk menjelaskan halaman web. Dalam hal struktur itu sendiri dari *head* dan *body*. *Head* mengandung informasi deskripsi *browser*. *Body* mengandung isi konteks, gambar dan sesuatu yang perlu ditampilkan. Sebuah file HTML terdiri dari elemen dan tag. Tag digunakan untuk mengatur elemen properti [2].

Metode ini menggunakan *regular expression* (*regex*) yaitu sekelompok karakter yang bisa mendeskripsikan atau mencocokkan dengan beberapa kata. Regex memiliki 2 fungsi yaitu mencari dan mengganti. Mencari suatu pola tertentu dalam teks kemudian menggantinya menjadi pola yang lain.

Regex digunakan oleh banyak teks editor, *utilities*, dan bahasa pemrograman untuk pencarian dan manipulasi teks berdasarkan pola.

C. jQuery Page Flip Plugin

jQuery adalah pustaka JavaScript yang cepat, ringan dan kaya fitur. Dengan adanya jQuery halaman HTML dapat dimanipulasi, diberikan event, animasi, dan AJAX yang jauh lebih sederhana dengan API yang mudah digunakan yang bekerja di banyak *browser*. Dengan kombinasinya yang fleksibel, jQuery telah mengubah cara bahwa jutaan orang menulis JavaScript [3]. Sudah banyak *plugin-plugin* yang dibuat dari jQuery untuk mempermudah pembuat aplikasi. Salah satunya adalah *jQuery Page-Flip Plugin*, yaitu *plugin* jQuery yang dapat memanipulasi dalam sehingga terlihat seperti buku yang dapat dibalik. Berikut adalah beberapa *jQuery Page-Flip Plugin*.

1) BookBlock

BookBlock adalah *plugin* jQuery yang dapat digunakan untuk membuat buklet seperti komponen yang memungkinkan untuk navigasi membalik halaman. Konten apapun dapat digunakan, seperti gambar atau teks. *Plugin* mengubah struktur hanya ketika membalik halaman dan menggunakan beberapa lapisan sebagai bayangan untuk halaman untuk menciptakan lebih realitis [4].

2) Booklet

Booklet adalah alat jQuery untuk menampilkan konten di web dalam tata letak *flipbook*. Booklet dibangun menggunakan pustaka jQuery. Tampilan Booklet dalam satu halaman terdiri dari dua halaman HTML, sehingga terlihat seperti buku terbuka [5].

D. Pugpig

Pugpig merupakan aplikasi publikasi *platform* HTML5 yang memungkinkan pengguna untuk membuat surat kabar, majalah dan buku-buku dengan tampilan yang rapi untuk iPad, iPhone, dan Android. Pugpig mampu menghasilkan tingkat kerapian, kehalusan dan kecepatan yang setara dengan *native application* yaitu aplikasi yang dibuat khusus untuk suatu sistem operasi tertentu (Android, iOS atau Blackberry). Biasanya pembuat sistem operasi menyediakan *tool* (Android Development Tools, dll) dan API khusus bagi para *developer* untuk pembuatan aplikasinya. Pugpig merupakan *hybrid app* yang menggabungkan bit terbaik dari kode *nativ* dengan potongan kode dari HTML, sehingga mampu menghasilkan pengalaman membaca yang menyenangkan pada iPhone, iPad dan Android [6]. Aplikasi Pugpig ini dijalankan dengan menggunakan tools Xcode yang merupakan perangkat lunak untuk *platform* Mac OS.

III. METODOLOGI

A. Arsitektur Perangkat Lunak

Aplikasi pembangkit halaman ini dapat dilakukan oleh pembuat berita baik penulis, penerbit, atau *publisher* yang ingin menjadikan hasil tulisannya secara digital. Karena

output dari aplikasi ini berupa halaman HTML maka pengguna dapat mempublikasikan hasil tulisannya di internet baik menggunakan web pribadi, Blog, Dropbox, dll.

Dokumen hasil aplikasi dimasukkan ke dalam folder Digipub yang berada di dalam folder dokumen komputer pengguna aplikasi. Folder Digipub secara otomatis dibuat ketika aplikasi ini pertama kali dijalankan. Setiap dokumen hasil aplikasi dijadikan satu folder dan dimasukkan ke dalam folder Digipub tersebut. Halaman HTML dapat dibuka ulang menggunakan aplikasi maupun web *browser*. Karena aplikasi ini menggunakan Metro Style yang merupakan keluaran dari Microsoft maka tampilan halaman dalam aplikasi ini secara tetap mengikuti tampilan dari web *browser* IE. Sehingga pengguna aplikasi harus menyesuaikan halaman yang diinputkan dengan web *browser* IE agar dapat ditampilkan dengan maksimal pada aplikasi Digipub Windows 8 ini.

B. Skenario Kasus Pengguna

Dalam pendefinisian kebutuhan perangkat lunak digunakan diagram kasus penggunaan. Kasus penggunaan yang digunakan pada sistem perangkat lunak ini digambarkan secara garis besar pada Gambar 1. Hal tersebut merupakan kebutuhan fungsional yang berisi kumpulan proses bisnis dalam perangkat lunak yang harus dipenuhi. Aktor dalam sistem ini adalah pembuat berita baik penulis, penerbit, distributor, dll. Pembuat berita dapat membuat berita baru, melihat berita maupun mengubah berita.

IV. IMPLEMENTASI

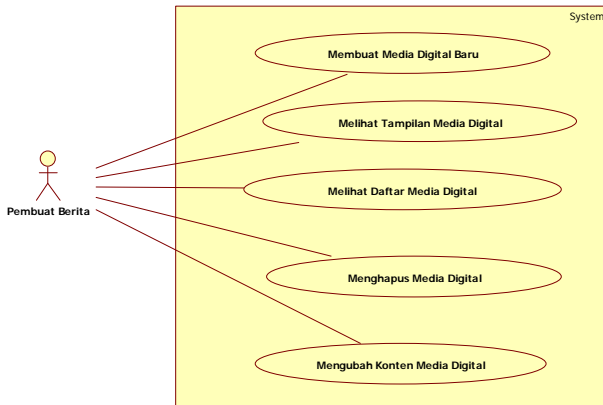
Implementasi pertama pada aplikasi ini adalah proses pembuatan media baru yaitu dengan menggabungkan masukkan halaman HTML dari pengguna. Sebelum menggabungkan halaman dilakukan proses pencarian *body* halaman per-halaman dengan menggunakan *regular expression (regex)*. Berikut adalah sintaks untuk mencari isi *body* halaman. Sintaks tersebut untuk mendapatkan isi dari tag *body*.

```
@"(?m)<body[^>]*>(\w|\W)*?</body[^>]*>"
@"</?body[^>]*?>"
```

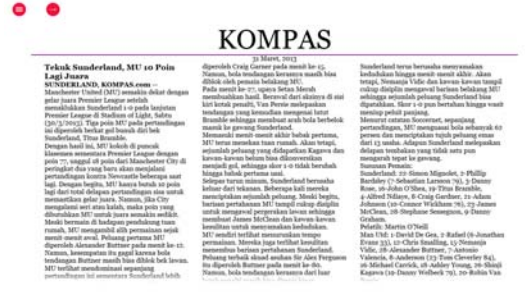
Setelah mendapatkan isi *body* per-halaman, dilakukan modifikasi halaman agar bisa disesuaikan dengan *style flip* buku yang dipakai pada *plugin*. Untuk membuat daftar halaman media, aplikasi juga menggunakan *regex* untuk mencari h1/H1 awal perhalaman. Berikut adalah sintaks untuk mencari h1/H1 awal *body* halaman.

```
<h1[^>]*?>(?!<TagText>.*?)</h1>
mc.Success ? mc.Groups[1].Value.Trim() :
string.Empty
```

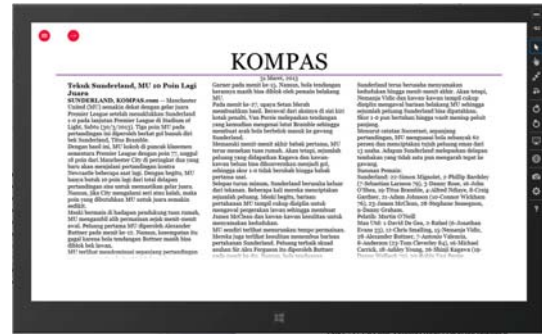
Dari Sintaks tersebut menunjukkan pencarian isi tag h1 dan mengambilnya dengan pada *index* pertama. Untuk proses perubahan media yang telah dibuat pada aplikasi



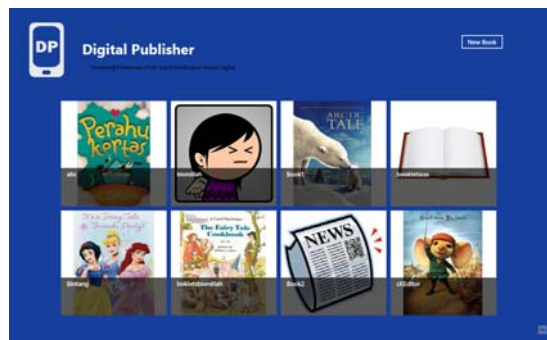
Gambar. 1. Diagram Kasus Pengguna Sistem



Gambar. 2. Tampilan Hasil Pembuatan Koran pada Aplikasi



Gambar. 3. Tampilan Hasil Pembuatan Koran pada Simulator Windows 8



Gambar. 4. Tampilan Hasil Pembuatan Koran pada Simulator

menggunakan plugin TinyMce. Perubahan halaman media hanya berfokus pada perubahan isi media.

Implementasi kedua dari aplikasi ini adalah menampilkan daftar media yang telah dibuat oleh aplikasi. Proses menampilkan daftar media dilakukan dengan membuka direktori folder penyimpanan media cetak digital yang pernah dibuat sebelumnya. Folder penyimpanan aplikasi berada pada dokumen pengguna di dalam folder Digipub. Folder Digipub dibuat ketika aplikasi membuat media pertama kali.

V. UJI COBA

Uji coba pada perangkat aplikasi pembangkit halaman untuk pembuatan media cetak digital ini dilakukan dengan menggunakan sebuah laptop. Pengujian perangkat lunak ini menggunakan metode pengujian *black box* yang berfokus pada kebutuhan fungsional. Uji coba ini dilakukan untuk menguji apakah fungsionalitas yang diidentifikasi pada tahap kebutuhan benar-benar diimplementasi dan bekerja seperti yang semestinya.

A. Uji Coba Membuat Media Baru

Gambar 2 dan 3 merupakan tampilan dari hasil pembuatan media pada aplikasi pembangkit halaman HTML. Gambar 2 merupakan tampilan dalam aplikasi, sedangkan Gambar 3 merupakan tampilan dalam simulator komputer tablet Windows 8. Selain membuat media baru, aplikasi juga dapat melakukan perubahan dan menghapus media yang sudah pernah dibuat sebelumnya. Untuk mengubah media yang

pernah dibuat aplikasi menggunakan *plugin* TinyMce. Aplikasi hanya dapat mengubah isi konten pada media.

B. Uji Coba Menampilkan Media pada Aplikasi dan Beberapa perangkat

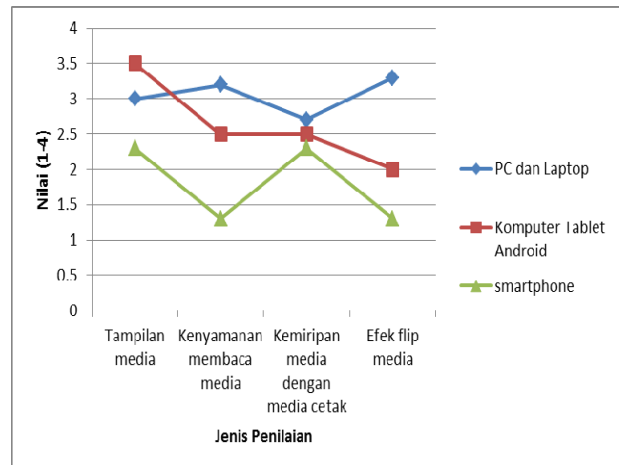
Gambar 6 merupakan tampilan media hasil pembuatan dari aplikasi dilihat dari beberapa perangkat oleh 32 pengguna. Pengujian tampilan dinilai dari indikator (1-4) dimana nilai satu adalah penilaian terendah dan nilai empat merupakan nilai tinggi. Dari Gambar 6 tersebut dapat disimpulkan bahwa tampilan media dari pembuatan aplikasi dapat berjalan dengan baik pada PC/laptop. Pengguna menampilkan media dengan menggunakan *browser* Mozilla, Chrome dan Safari.

Pada komputer tablet, pengguna yang menampilkan halaman dengan menggunakan *browser* Chrome dapat berjalan dengan baik, sedangkan dengan *browser* Browser maupun Opera Mini efek membuka halaman tidak dapat berjalan dengan baik karena *browser* tidak kompetibel dengan CSS3. Untuk tampilan pada *smartphone* tampilan media kurang baik karena ditampilkan pada *browser* Opera Mini yang tidak kompetible pada CSS3. Sedangkan untuk kesesuaian masukan HTML dan penggabungan HTML hasil pembuatan aplikasi adalah sesuai dengan masukan. Untuk animasi membalik halaman, tampilan menggunakan *browser* lebih halus dibandingkan dengan tampilan aplikasi karena *plugin* yang digunakan pada aplikasi lebih mendukung jika ditampilkan pada *browser* Chrome, Mozilla, Opera dan Safari.

C. Evaluasi Pengujian

Dari skenario pengujian kebutuhan fungsional, berikut adalah evaluasi pengujian aplikasi.

1. Proses pembuatan media cetak digital didalam aplikasi berhasil dibuat dengan baik.
2. Proses menampilkan media cetak digital di dalam aplikasi berhasil ditampilkan dengan baik.
3. Proses menampilkan daftar media cetak digital yang pernah dibuat di dalam aplikasi berhasil ditampilkan dengan baik.
4. Proses menghapus media dalam aplikasi berhasil dilakukan dengan baik.
5. Proses pengubahan media dalam aplikasi berhasil dilakukan dengan baik.
6. Proses tampil media dalam bentuk koran baik di Laptop maupun komputer tablet yang ditampilkan dengan menggunakan aplikasi Digipub berhasil ditampilkan sesuai dengan masukan HTML oleh pengguna.
7. Proses tampilan pembuatan media dalam bentuk koran baik dengan menggunakan *browser* secara *localhost* berhasil ditampilkan sesuai dengan masukan HTML oleh pengguna.
8. Proses tampilan pembuatan media dalam bentuk koran baik dengan menggunakan *browser* pada perangkat komputer tablet Samsung Galaxy Tab dan iPhone 5 berhasil ditampilkan akan tetapi kenyamanan dan tampilan kurang baik karena keterbatasan ukuran layar sehingga membutuhkan banyak *scroll*.
9. Proses tampilan pembuatan media dalam bentuk buku atau novel baik di laptop maupun komputer tablet yang ditampilkan dengan menggunakan aplikasi Digipub berhasil ditampilkan sesuai dengan masukan HTML oleh pengguna.
10. Proses tampilan pembuatan media dalam bentuk buku atau novel baik dengan menggunakan *browser localhost* berhasil ditampilkan sesuai dengan masukan HTML oleh pengguna.
11. Proses tampilan pembuatan media dalam bentuk buku atau novel baik dengan menggunakan *browser* pada perangkat komputer tablet Samsung Galaxy Tab dan iPhone 5 berhasil ditampilkan akan tetapi kenyamanan dan tampilan kurang baik karena keterbatasan ukuran layar sehingga membutuhkan banyak *scroll*.
12. Penilaian oleh pengguna melihat tampilan hasil pembuatan aplikasi dengan menggunakan PC/laptop, komputer tablet dan *smartphone*. Aplikasi berhasil ditampilkan dengan baik pada PC/laptop dan komputer tablet yang menggunakan *browser* Chrome, Mozilla, Opera, Safari dan IE. Sedangkan dengan *smartphone browser* yang digunakan tidak kompetibel dengan halaman media dan terbatas dengan ukuran layar monitor.



Gambar. 6. Tampilan Media Hasil Pembuatan Aplikasi pada Beberapa Perangkat oleh 32 Pengguna

VI. KESIMPULAN/RINGKASAN

A. Kesimpulan

Dalam proses pengerjaan dari tahap analisis, desain, implementasi, hingga pengujian didapatkan kesimpulan sebagai berikut :

1. Aplikasi pembangkit halaman HTML mampu mengubah halaman HTML dan menambahkan skrip pada halaman agar tercipta efek membuka tampilan halaman seperti format buku atau majalah.
2. Aplikasi dapat melakukan pembuatan media cetak digital sesuai dengan masukan halaman HTML dari pengguna.
3. Aplikasi pembangkit halaman HTML mampu berjalan pada sistem operasi Windows 8.
4. Hasil pembangkitan halaman HTML mampu ditampilkan di aplikasi baik di dalam PC maupun komputer tablet Windows 8.
5. Aplikasi dapat menghapus dan melakukan perubahan isi halaman media.

B. Saran

Setelah pengerjaan selesai, terdapat beberapa saran yang bertujuan untuk menyempurnakan aplikasi pembangkit halaman HTML ini. Saran untuk pengembangan aplikasi ini antara lain :

1. Aplikasi ini untuk pengembangan selanjutnya dapat menambahkan proses perubahan halaman yang lebih detail misalnya saja dapat mengubah CSS masukan halaman dari pengguna langsung dari aplikasi.
2. Aplikasi ini untuk pengembangan selanjutnya dapat memperbaiki navigasi membuka halaman yang mendukung *touch gesture* agar proses membuka halaman terlihat lebih halus.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Tuhan Yang Maha Esa, karena hanya dengan karunia, rahmat dan hidayah-Nya penulisan tugas akhir ini dapat terselesaikan dengan baik. Penulis juga mengucapkan terima kasih kepada semua pihak yang terkait dalam penyelesaian penulisan artikel ini.

DAFTAR PUSTAKA

- [1] Metro, [Online], (<http://windowmetrostyle.com/>), diakses tanggal 24 September 2012)
- [2] Zhang Xu, Dong Yan. Designing and Implementing of the Webpage Information Extracting Model Based on Tags. 2011 International Conference on Intelligence Science and Information Engineering.
- [3] jQuery, [Online], (<http://jquery.com/>), diakses tanggal 29 Maret 2013)
- [4] BookBlock, [Online], (<http://tympanus.net/codrops/2012/09/03/bookblock-a-content-flip-plugin/>) diakses tanggal 24 September 2012).
- [5] Booklet, [Online], (<http://builtbywill.com/code/booklet/>), diakses tanggal 24 September 2012).
- [6] Pugpig, [Online], (<http://pugpig.com/>), diakses tanggal 24 September 2012).
- [7] TinyMce, [Online], (<http://tinymce.com/>), diakses tanggal 15 Maret 2013).
- [8] Microsoft developer network, Developing Apps, [online], (<http://msdn.microsoft.com/en-us/library/windows/apps>), diakses tanggal 25 September)