

1. Pendahuluan

Semakin pesatnya perkembangan arus internet dan teknologi informasi juga memacu pesatnya perkembangan teknologi *website*. Sehingga sebuah *website* tidak lagi hanya dilihat dari informasi yang disajikan tetapi juga dari desain *layout*. Kreasi dan dukungan multimedia yang tersedia, menuntut para pengembang *web* (*web programmer*) untuk selalu berkreasi menciptakan ide-ide baru baik dalam desain *layout*, penggunaan aplikasi *web server*, maupun *operating system* yang dipakai, agar *website* yang dibangun tidak kandas dalam persaingan.

Perkembangan internet ini, diikuti pula dengan perkembangan HTML (*Hyper Text Markup Language*) yang merupakan bahasa standar yang paling umum atau yang paling sering digunakan oleh para *web programmer* dalam membangun sebuah aplikasi *web*. HTML yang umum digunakan saat ini adalah HTML versi 4.01. HTML versi 5 atau HTML5 menjadi bahasa HTML baru yang hadir dengan tambahan fitur-fitur baru yang berbeda dengan HTML4.01. Sejauh mana perbandingan atau perbedaan yang ada antara HTML4.01 dan HTML5 inilah, yang diangkat dalam penelitian yang akan dilakukan yaitu menganalisis dan melakukan perbandingan antara kedua versi HTML, yang meliputi penulisan *script* dan desain *interface*, *tag*, atribut dan *input types* yang digunakan, penanganan dokumen multimedia (video), validasi data, kecepatan akses dan kompatibilitas terhadap *browser*.

2. Kajian Pustaka

Dalam Laporan W3C yang berjudul “HTML5 Differences From HTML4”, HTML4 menjadi Rekomendasi W3C pada tahun 1997. Meskipun terus melayani banyak fitur inti dari HTML, tetapi tidak memberikan informasi yang cukup untuk membangun implementasi yang saling beroperasi satu sama lain dan lebih penting dengan konten-konten yang kritis. Hal yang sama berlaku untuk XHTML1, yang mendefinisikan sebuah serialisasi XML untuk HTML4, dan DOM Level 2 HTML, yang mendefinisikan *JavaScript* API untuk HTML dan XHTML. HTML5 akan menggantikan dokumen-dokumen ini. W3C menerbitkan HTML5, yang merupakan revisi besar kelima dari HTML. WHATWG menerbitkan HTML, yang merupakan superset dari HTML5. “Perbedaan dari HTML4 HTML5” menggambarkan perbedaan dokumen-dokumen dari HTML4, dan kasus-kasus pemanggilan dimana HTML berbeda dari HTML5 [1].

Sama seperti sebuah *software*, HTML memiliki beberapa versi. Versi terakhir yang diperkenalkan adalah HTML5. Walaupun belum semua *browser* dapat mengimplementasikan *tag-tag* dalam HTML5, HTML5 memiliki kapabilitas baru dibanding versi sebelumnya, HTML4.01. Misalnya saja, HTML5 memiliki *tag* baru bernama `<audio>` dan `<video>` yang secara khusus berfungsi memasang suara atau musik dan *video* ke dalam *website*. Ketika menggunakan HTML4.01, cara untuk memasang *video* dan *audio* ke dalam *website* adalah dengan memanfaatkan *tag* `<embed>` yang sebenarnya berfungsi sangat luas [2].

Penelitian yang berjudul “Pemanfaatan *File* Format HTML Sebagai Wahana Cetak Pada Pemrograman Komputer”, menghasilkan piranti lunak pengolahan HTML. Pembuatan tampilan program menggunakan HTML sangat membantu

dan mempermudah pemrograman komputer untuk mendapatkan hasil keluaran program yang menarik dengan cara yang cepat dan mudah [3].

Dalam penelitian yang berjudul “Pencitraan 3 Dimensi Berbasis *JavaScript Object Notation (JSON)* Pada *Hyper Text Markup Language 5 (HTML5)*” Penggunaan aplikasi berbasis *website* semakin berkembang dan digunakan untuk banyak hal, mulai dari Sistem Informasi hingga *Game-Online*. *Hypertext Markup Language 5 (HTML5)* adalah salah satu teknologi terbaru dalam standar *World Wide Web*. Standar ini memperkenalkan fitur baru seperti memutar video ataupun *drag and drop*, sebelum ada HTML5, fitur untuk memutar video bergantung pada *plugins* pihak ketiga di penjelajah *website* seperti *Adobe Flash* atau *Microsoft Silverlight*. HTML5 membuat mekanisme yang lebih mudah dalam pencitraan, baik itu hanya *audio*, video, dua dimensi hingga tiga dimensi sekalipun. Teknologi yang sangat fenomenal dari HTML5 adalah adanya fitur *Web-Based Graphic Library (WebGL)* yang diadaptasi dari aplikasi *desktop* agar *website* dapat menjalankan sebuah Interaktif tiga Dimensi yang dimasukkan dalam *JavaScript* dan adanya *tag Canvas* yang dapat digunakan untuk pencitraan tiga Dimensi dengan berbasis teks yang dikemas dalam *JavaScript Object Notation (JSON)*. JSON adalah suatu format ringkas pertukaran data dalam komputer yang digunakan untuk merepresentasikan struktur data sederhana dan sebuah objek yang didasarkan pada *JavaScript* walaupun tidak bergantung pada suatu bahasa [4].

Berdasarkan penelitian terkait penggunaan HTML dalam pembuatan *website* maka melalui penelitian ini, akan dilakukan analisis dan perbandingan terhadap HTML4.01 dan HTML5 berdasarkan aplikasi *web online music store* yang dibangun. Melalui penelitian yang dilakukan diharapkan dapat memberikan gambaran dan informasi yang jelas bagi para *web designer* tentang perbedaan dan perbandingan HTML4.01 dan HTML5 dan penggunaannya dalam membangun aplikasi *web* yang sesuai dengan kebutuhan.

Web

Web atau situs dapat diartikan sebagai kumpulan halaman yang digunakan untuk menampilkan teks, gambar diam atau gerak, animasi, suara atau gabungan semuanya itu baik bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, dimana masing-masing dihubungkan dengan jaringan-jaringan yang saling terkait (*hyperlink*) [5].

HTML4.01

HTML4.01 adalah simbol-simbol atau *tag-tag* yang dituliskan dalam sebuah *file* yang dimaksudkan untuk menampilkan halaman pada *web browser*. *Tag-tag* HTML4.01 selalu diawali dengan `<x>` dan diakhiri dengan `</x>` dimana *x tag* HTML4.01 seperti `b`, `i`, `u` dan sebagainya. Namun ada juga *tag* yang tidak diakhiri dengan tanda `</x>` seperti `tag
`, `<input>` dan lainnya [6].

Seperti umumnya dokumen lain, dokumen HTML4.01 terdiri dari teks-teks dan bahkan lebih dari itu, dokumen HTML4.01 juga dapat mengandung suatu gambar, suara, ataupun video. Satu hal yang membedakan dokumen HTML4.01 dengan dokumen-dokumen lainnya adalah adanya elemen-elemen HTML4.01

beserta *tag-tag*nya. Elemen dan *tag* HTML4.01 berfungsi untuk memformat atau menandai suatu bagian tertentu dari dokumen HTML4.01 dan juga untuk menentukan struktur bagian tersebut dalam dokumen HTML4.01. Elemen dan *Tag* inilah yang merupakan ciri utama dari suatu dokumen HTML. Secara garis besar, untuk menuliskan sebuah dokumen HTML4.01 dibutuhkan kerangka penulisan dengan *tag-tag* dasar, yaitu: *HTML*, *HEAD*, *TITLE* dan *BODY* [7]. Struktur dasar HTML4.01 ditunjukkan pada Gambar 1.

```
<html>
<head>
<title>
// judul atau nama halaman web
</title>
</head>
<body>
// isi web
</body>
</html>
```

Gambar 1 Struktur Dasar HTML4.01

HTML5

HTML5 adalah versi terbaru dari HTML, dan merupakan standar baru untuk HTML, XHTML, dan HTML DOM. HTML5 sendiri sampai saat ini masih terus dikembangkan dan mayoritas *browser* modern sudah bisa mendukung HTML5. Ada beberapa aturan yang diterapkan untuk HTML5, seperti, fitur-fitur baru harus berbasis HTML, CSS, DOM, dan *JavaScript*, mengurangi kebutuhan *plugin* eksternal (contohnya *flash*), *error handling* yang lebih baik, *markup* tambahan untuk menggantikan *scripting*, HTML5 harus bisa diakses dari piranti manapun atau *device independent*, proses pengembangan harus bisa dilihat oleh publik [8].

Struktur dasar HTML5 ditunjukkan pada Gambar 2.

```
<!DOCTYPE html>
<html>
<head>
<title>
// judul halaman web
</title>
</head>
<body>
// isi halaman web
</body>
</html>
```

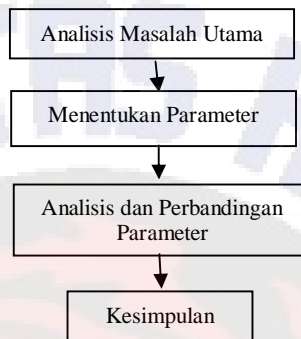
Gambar 2. Struktur Dasar HTML5

ASP

ASP bekerja pada *web server* dan merupakan *server side scripting*. Berbeda dengan *VBScripting* yang bekerja pada *client* atau disebut juga *client side scripting*. Meskipun demikian ASP menggunakan *VBScript* sebagai bahasa dasar untuk pemrogramannya. Karena bersifat *server side scripting* maka ASP dapat bekerja pada semua *web browser* [9].

3. Tahapan penelitian

Tahapan penelitian disusun dalam urutan-urutan langkah kerja dalam sebuah diagram alir sehingga dapat memfokuskan analisis dan perbandingan yang dilakukan. Tahapan penelitian dalam analisis dan perbandingan HTML4.01 dan HTML5 ditunjukkan pada Gambar 3.



Gambar 3 Tahapan Penelitian

Tahapan penelitian yang dilakukan, dijelaskan sebagai berikut:

1. Analisis Masalah Utama. Mengelompokkan dan memilah data menjadi satuan yang dapat dikelola. Dengan demikian perlu adanya analisis terhadap masalah utama untuk menemukan pokok-pokok permasalahan yang dapat dijadikan parameter untuk memfokuskan penelitian tentang HTML4.01 dan HTML5.
2. Menentukan Parameter. Berdasarkan analisis terhadap masalah utama, maka akan ditentukan hal-hal pokok yang akan dijadikan sebagai parameter perbandingan antara HTML4.01 dan HTML5.
3. Analisis dan Perbandingan Parameter. Pada tahap ini, akan dilakukan proses analisis dan perbandingan secara *side by side* terhadap masalah atau parameter HTML4.01 dan HTML5 yang telah ditemukan.
4. Kesimpulan. Kesimpulan dari hasil analisis dan perbandingan yang dilakukan pada HTML4.01 dan HTML5.

4. Gambaran Umum Aplikasi

Aplikasi yang dibangun untuk melakukan pengujian analisis dan perbandingan antara HTML4.01 dan HTML5 adalah aplikasi *Web Online Music Store*.

HTML4.01

Sebagai seorang *user* yang ingin melakukan pemesanan atau pembelian maka dapat mengakses halaman *default* yang menjadi halaman utama aplikasi ini yaitu di <http://aspider.ws/jeyzone/default.aspx>. Semua halaman ada dalam aplikasi *web* ini menggunakan satu *MasterPage.Master* yang sama. Kode program untuk *master page* dapat dilihat pada Kode Program 1, Kode Program 2, Kode Program 3 dan Kode Program 4.

Kode Program 1 Kode Program MasterPages.Master1

```
1.<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/DTD/strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
2.<head>
3.<meta http-equiv="content-type" content="text/html; charset=utf-8" />
4.<title>Music Store</title>
5.<meta name="keywords" content="" />
6.<meta name="description" content="" />
7.<link href="styles.css" rel="stylesheet" type="text/css" media="screen" />
8.<link rel="stylesheet" href="nivo-slider.css" type="text/css" media="screen"
/>
9.<script type='text/javascript' src="jwplayer.js"></script>
10.<asp:ContentPlaceholder ID="headHolder" runat="server">
11.</asp:ContentPlaceholder>
12.</head>
```

Kode Program 1 baris ke-1 mengidentifikasi kepada *browser* tentang jenis dokumen yang digunakan, `<meta>` menjelaskan tentang konten atau isi dari dokumen HTML, dan *script type* berfungsi untuk menuliskan kode *javascript*.

Kode Program 2 Kode Program MasterPages.Master2

```
1.<div id="wrap">
2.<div id="header">
3.<div id="menu">
4.<ul>
5.<li class="but4_menu"><a href="default.aspx">Home</a></li>
6.<li class="but4_menu"><a href="gallery.aspx">Gallery</a></li>
7.<li class="but4_menu"><a href="cartpage.aspx">Cart</a></li>
8.<% if (Session["member"] != null)
10.    { %>
11.<li class="but4_menu"><a href="default.aspx?logout=1">Log out</a></li>
12.    <%
13.    }
14.else
15.    {
16.%>
17.<li class="but4_menu"><a href="signup.aspx">Sign Up</a></li>
18.</ul>
19.</div>
```

Kode Program 2 adalah kode program pada *file MasterPage.master* dan digunakan oleh semua halaman yang menggunakan *template MasterPage.master*. Kode program tersebut berfungsi untuk menampilkan menu navigasi pada halaman *web*. Kode program baris ke-5 sampai baris ke-7 berfungsi menampilkan *home*, *gallery* dan *cart*. Khusus menu *Log out* dan *Sign up* tidak akan muncul bersamaan. Menu *Sign Up* pada baris kode ke-17 muncul ketika pengunjung *web* belum melakukan *login* sebagai *Member*. Sedangkan menu *Log out* pada baris kode ke-11 hanya muncul ketika pengunjung telah melakukan *login*.

Kode Program 3 Kode Program MasterPages.Master3

```
1.<div id="logo">
2.<h1>
3.<a href="#">QUANTUM MUSIC</a></h1>
4.  <a href="#"><small>Online Music Store</small></a>
5.  </div>
6.  <asp:ContentPlaceholder ID="rollerHolder" runat="server">
7.  </asp:ContentPlaceholder>
8.  </div>
9.  div style="clear: both">
10.</div>
11.<div id="content">
12.<asp:ContentPlaceholder ID="contentHolder" runat="server">
13.</asp:ContentPlaceholder>
14.</div>
```

Kode Program 3 merupakan kode program pada file *MasterPage.master* dan digunakan oleh semua halaman yang menggunakan *template MasterPage.master*. Kode Program 3 merupakan bagian isi dari halaman *web*. Tag `<ContentPlaceHolder>` berfungsi menyediakan area yang dapat diimplementasikan sendiri oleh tiap halaman yang menggunakan *MasterPage.master*.

Kode Program 4 Kode Program *MasterPages.Master4*

```

1.<div class="footer_top_colum2">
2.<h3>
3. Latest Stuff</h3>
4. <ul class="ls">
5.<%
6. var items = d.Produks.OrderByDescending(x => x.Tanggal).Take(5);
7. foreach (Produk p in items)
8.{
9.%>
10.<li><a href='DetailProduk.aspx?kode=<%=p.Kode %>'>
11.<%=p.Nama %></a></li>
12.<%
13.%>
14. </ul>
15. </div>
16. <div class="footer_top_colum3">
17. <h3>
18. Member Log In</h3>
19. <a id="lgn" name="lgn"></a>
20. <asp:MultiView ID="MultiView1" runat="server">
21. <asp:View ID="loginView" runat="server">
22. <asp:Login ID="memberLogin" runat="server" 23.DisplayRememberMe="False"
    TextLayout="TextOnTop"
24. TitleText=" OnAuthenticate="memberLogin_Authenticate">
25. </asp:Login>
26. </asp:View>
27. <asp:View ID="logoutView" runat="server">

```

```

28.         Welcome,
29. <asp:Literal ID="memberNameLiteral" 30.runat="server"></asp:Literal>
31. <br />
32. <asp:LinkButton ID="logoutButton" runat="server"
33.     OnClick="logoutButton_Click">[Logout]</asp:LinkButton>
34. </asp:View>
35. </asp:MultiView>
36. <div class="button_box">
37. </div>

```

Kode Program 4 merupakan bagian dari *file MasterPage.master*, yang berfungsi untuk menampilkan menu *latest stuff* dan *Log in* di bagian *footer*. Kode program baris ke-6 berfungsi untuk menampilkan daftar 5 (lima) produk terbaru. Kode program baris ke-18 sampai ke-25 merupakan pengecekan terhadap *member* yang melakukan *log in*.

HTML5

Desain yang digunakan dalam pembuatan aplikasi untuk *online music store* dengan HTML5 menggunakan desain yang sama dengan desain HTML4.01 sehingga tidak ada perbedaan desain *interface* antara HTML4.01 dengan HTML5. Kode Program untuk *MasterPage.Master online music store* HTML5 dapat dilihat pada Kode Program 5.

Kode Program 5 Kode Program MasterPages.Master HTML51

```
1.<!DOCTYPE HTML>
2.<html>
3.<head>
4.<meta http-equiv="content-type" content="text/html; charset=utf-8" />
5.<title>Music Store (HTML5)</title>
6.<meta name="keywords" content="" />
7.<meta name="description" content="" />
8.<link href="styles.css" rel="stylesheet" type="text/css" media="screen" />
9.<link rel="stylesheet" href="nivo-slider.css" type="text/css" media="screen"
10./>
11.<link href="video-js.css" rel="stylesheet" type="text/css" />
12.<script src="video.js" type="text/javascript"></script>
13.<asp:ContentPlaceholder ID="headHolder" runat="server">
14.</asp:ContentPlaceholder>
15.</head>
```

Dalam mendefinisikan menu yang digunakan dalam aplikasi *online music store*, HTML5 memiliki beberapa perbedaan dengan HTML4.01 dalam penulisan *scriptnya*. Penulisan *script* untuk HTML5 dapat dilihat pada Kode Program 6.

Kode Program 6 Kode Program MasterPages.Master HTML52

```
1.<body>
2.<form id="Form1" runat="server">
3.<section id="bg_top">
4.<section id="wrap_bg">
5.<section id="wrap">
6.<header>
7.<nav>
8.<ul>
9.<li class="but4_menu"><a href="default.aspx">Home</a></li>
10.<li class="but4_menu"><a href="gallery.aspx">Gallery</a></li>
11.<li class="but4_menu"><a href="cartpage.aspx">Cart</a></li>
12.<% if (Session["member"] != null)
13.    { %>
14.<li class="but4_menu"><a href="default.aspx?logout=1">Log out</a></li>
15.    <%
```

dilakukan dengan beberapa urutan langkah kerja, yang akan digunakan dalam melakukan penelitian terhadap masalah yang diangkat.

Analisis Masalah Utama

Masalah utama yang menjadi fokus dalam penelitian ini adalah HTML4.01 dan HTML5. Hadirnya HTML5 menjadi terobosan baru bagi para *web designer* yang selama ini bekerja menggunakan HTML4.01. HTML5 hadir dengan berbagai fitur terbaru yang dikatakan mempermudah para *web designer* dalam bekerja. Banyak *web designer*, baik profesional maupun pemula yang kemudian mulai mencoba beralih dari HTML4.01 ke HTML5, karena tawaran kemudahan yang dihadirkan oleh HTML5. Masalah-masalah seperti kompatibilitas terhadap *browser* menjadi hal yang paling sering ditemukan. Dalam penelitian ini, akan dilakukan analisis dan perbandingan antara HTML4.01 dan HTML5.

Menentukan Parameter Analisis dan Perbandingan

Sebelum melakukan analisis dan perbandingan terhadap HTML4.01 dan HTML5, maka dilakukan pengelompokan terhadap hal-hal yang akan menjadi parameter perbandingan. Hal ini dilakukan untuk mempermudah dan memfokuskan proses analisis dan perbandingan yang dilakukan. Pengelompokan

parameter yang dianalisis dan dibandingkan adalah sebagai berikut: (1) Desain *Interface*; (2) *Tag*, atribut dan *input types* yang digunakan; (3) Penanganan dokumen multimedia (Video); (4) Validasi data; (5) Kecepatan Akses; (6) Kompatibilitas terhadap *browser*; (7) Pengaruh penggunaan *javascript* pada HTML 4.01 terhadap kecepatan akses dan penggunaan memori. Setelah melakukan pengelompokan terhadap hal-hal yang menjadi parameter yang akan dianalisis dan dibandingkan, maka proses analisis dan perbandingan terhadap HTML4.01 dan HTML5 dapat dilakukan.

Tabel Hasil Analisis dan Perbandingan HTML4.01 dan HTML5

Tabel 1 Hasil Analisis dan Perbandingan HTML4.01 dan HTML5

No	Parameter Yang Diuji	Pengujian		Penjelasan
		HTML4.01	HTML5	
1	Desain <i>Interface</i>	Desain halaman menggunakan CSS.	Desain halaman menggunakan CSS.	Desain <i>interface</i> secara umum tidak terlihat adanya perbedaan untuk dibandingkan, perbandingan difokuskan terhadap menu atau konten yang dimuat oleh aplikasi ke dalam halaman <i>web</i> . Analisis dilakukan terhadap konten atau media <i>player</i> yang ditampilkan pada halaman utama dan halaman <i>gallery</i> .
		Media <i>player</i> pada halaman utama ditampilkan bersama <i>control bar</i> .	Media <i>player</i> tanpa <i>control bar</i> .	

2	Tag, Atribut, Input type dan elemen	<pre><div id="bg_top"> <div id="header"> <div id="footer"> <div id="menu"></pre>	<pre><section id = " bg_top " > <header> <foother> <nav> <email> <video> <pattern> <placeholder> <Range> <x-webkit- speech ></pre>	<p>HTML4.01 menggunakan <code><div></code> untuk mengelompokkan <i>file</i> HTML sesuai fungsinya. Setiap fungsi akan ditandai dengan id masing-masing. Semakin banyak fungsi yang digunakan, semakin banyak div yang dipakai. Untuk HTML5, fungsi <code><div></code> digantikan sesuai fungsi yang akan dipanggil. Contoh untuk membuat <i>header</i>, HTML4.01 mendefinisikan <code><div id="header"></code> dan HTML5 <code><header></code>. Penulisan <i>script</i> HTML5 relatif mempermudah <i>web programmer</i> dalam mengelompokkan fungsi-fungsi yang sejenis. Untuk menampilkan dokumen multimedia seperti video, HTML4.01 menggunakan <i>javascript</i> dan <i>plugin</i> tambahan. HTML5 menggunakan tag <code><video></code> yang memanfaatkan <i>library javascript</i>. Validasi <i>email</i> HTML4.01 menggunakan <i>regular expresiion</i>, HTML5 menggunakan <code><emai></code>. Untuk membuat <i>shadow</i> pada kolom <i>form</i>, HTML4.01 menggunakan <i>javascript</i>, dan HTML5 menggunakan tag <code><placeholder></code>. <code><Range></code> dan <code><x-webkit-speech></code> hanya ada pada HTML 5. Fungsi ini digunakan peneliti untuk melakukan uji kompabilitas HTML5 terhadap <i>browser</i>.</p>
3	Penanganan Dokumen Multimedia (video)	Menggunakan fungsi <i>javascript</i> dan <i>flash player</i>	Menggunakan tag <code><video></code>	<p>Dalam menangani dokument multimedia pada aplikasi yang dibangun, contohnya video, HTML4.01 menggunakan fungsi <i>javascript</i> berikut:</p> <pre><script type = " text/javascript "> jwplayer('mediaplayer_3').setup('flashplayer': 'player.swf', 'id': 'playerID', 'width': '455', 'height': '176', 'file': 'media/SONOR Arena Kit- PROMO ViDEO.flv' }); </script></pre> <p>HTML5 menggunakan standar penulisan <i>script</i> tersendiri.</p> <pre><video class="video-js vjs- default-skin" controls preload="auto" width="455"</pre>

```

height="176" data-setup="{}">
<source src='media/SONOR
Arena Kit- PROMO ViDEO.flv'
type="video/mp4" />
Your browser does not support
the video tag.
</video>

```

HTML4.01 menyisipkan fungsi untuk memanggil *flash player*. Jika perangkat komputer tidak memiliki *flash player*, maka dokumen multimedia yang diminta tidak dapat ditampilkan. HTML5 memanfaatkan tag `<video>` dengan memanggil fungsi *library javascript* `video.js`. Dengan fungsi ini, maka HTML5 dapat menampilkan dokumen multimedia tanpa harus menggunakan fungsi atau *plugin* tambahan seperti *flash*

4	Validasi Data <i>Username</i>	Menggunkan fungsi <i>javascript</i>	Menggunkan <i>input type</i> <code><username></code> dan atribut <code><pattern></code>
---	----------------------------------	---	---

Javascript yang digunakan dalam aplikasi HTML4.01 untuk validasi *username*:

```

function isUsernameValid(uid) {
  var lookupTable =
  "abcdefghijklmnopqrstuvwxy
  ABCDEFGHIJKLMNOPQRSTU
  VWXYZ0123456789";
  if (uid.length < 6) return false;
  for (i = 0; i < uid.length; i++) {
    var char = uid.charAt(i); var 6.
    index =
    lookupTable.indexOf(char); if
    (index == -1) {return false;}
  }
}

```

HTML5 menggunakan Fungsi HTML murni:

```

<input type="username" name =
"username" type = "text"
required placeholder="ketik
username disini."
autofocus pattern="[A-Za-z0-
9]{6,12}"title= " Panjang
Username 6 s/d 12 karakter
terdiri dari huruf dan angka
saja."required />

```

Dalam melakukan validasi terhadap *username*, HTML4.01 Menggunakan *lookupTabel* untuk mendaftarkan karakter yang diperbolehkan dalam penulisan *username*. Panjang karakter yang

diperbolehkan dalam pengisian dicek menggunakan *javascript*. HTML5 menggunakan *input type <username>* untuk mendefinisikan fungsi *file username*. HTML5 menggunakan atribut *<pattern>* untuk mendaftarkan karakter-karakter yang diperbolehkan dalam penulisan *user name*. *<pattern>* dapat langsung memasukkan batasan minimal dan maksimal karakter yang diperbolehkan. Penggunaan *username* dan *pattern* relatif lebih mudah dibandingkan penggunaan fungsi *javascript* dalam validasi *username*.

Validasi Data <email>	Menggunakan fungsi <i>javascript reguler expression</i>	Menggunakan fungsi HTML5 <i>input type email</i>
--------------------------	---	--

HTML4.01 Menggunakan fungsi *reguler ekspresi* untuk melakukan validasi *email*. *reguler ekspresi* yang digunakan dalam aplikasi:
var emailReg = /^[^\w-\.]+@([\w-]+\.)+[\w-]{2,4}\$/;

Untuk HTML5 menggunakan *script HTML5*:
<input type="email" name="email" type="email" required/>.

Penggunaan *email* relatif memudahkan seorang *web designer* dalam melakukan validasi data untuk *email*. *Email* memungkinkan *web designer* tidak lagi menggunakan *reguler expression*. Tetapi dalam penggunaannya, seorang *web designer* harus memperhatikan versi *browser* yang digunakan apakah mendukung HTML5 secara penuh atau tidak.

5	Kecepatan Akses	Pengujian:	Pengujian:	Dalam melakukan uji kecepatan akses pada HTML4.01 dan HTML5, digunakan lima <i>browser</i> yaitu, <i>Mozilla Firefox 12.0</i> , <i>Google Chrome 19</i> , <i>Internet Explorer 9.0</i> , <i>Opera 11.64</i> dan <i>Safari 5.1.7</i> . Pengukuran waktu dihitung sejak <i>browser</i> mulai memuat halaman <i>web</i> sampai ditampilkan dengan sempurna ke dalam <i>browser</i> .
	<i>Mozilla Firefox 12.0</i>	1: 24.68 detik 2: 22.56 detik	1: 19.68 detik 2: 16.85 detik	
	<i>Google Chrome 19</i>	1: 24.23 detik 2: 23.64 detik	1: 19.33 detik 2: 18.92 detik	
	<i>Internet Explorer 9.0</i>	1: 27.87 detik 2: 24.59 detik	1: 17.10 detik 2: 17.01 detik	• <i>Genuine Intel(R) Dual Core CPU 2140 @ 1.60 GHz</i>

Opera 11.64	1: 30.69 detik 2: 28.69 detik	1: 16.70 detik 2: 15.88 detik	<ul style="list-style-type: none"> • <i>Memori</i> 4.00 GB • <i>Hardisk</i> 320 GB • <i>System Type:</i> 64-bit <i>Operating System</i>
Safari 5.1.7	1: 30.69 detik 2: 21.18 detik	1: 17.11 detik 2: 16.50 detik	Dalam uji kecepatan akses, pengujian dilakukan sebanyak dua kali. Kecepatan akses internet yang digunakan untuk setiap pengujian juga berbeda. Untuk pengujian pertama, pengujian menggunakan internet dengan kecepatan akses rata-rata 70 KB/sec, dan pengujian kedua, menggunakan kecepatan akses rata-rata 80-85 KB/sec. Berdasarkan hasil uji kecepatan akses pada Tabel 4.7, terlihat perbedaan waktu yang cukup lama antara HTML4.01 dan HTML5 saat dimuat ke dalam halaman browser. Rata-rata kecepatan akses sebuah aplikasi dengan HTML4.01 dimuat oleh browser adalah 25.87 detik, dan untuk HTML5 mempunyai rata-rata kecepatan akses 17.50 detik. Ada selisih waktu 8.37 detik antara HTML4.01 dan HTML5

Rata-rata kecepatan: 25.87 detik 17.50 detik

Uji Kompatibilitas

Pengujian untuk uji kompatibilitas yang dilakukan, hanya dikhususkan untuk HTML5. Hal ini untuk menguji tingkat kompatibilitas antara browser yang digunakan dengan HTML5. HTML4.01 tidak disertakan dalam uji kompatibilitas, karena HTML4.01 adalah bahasa HTML yang telah umum digunakan sehingga tingkat kompatibilitasnya tentu lebih baik. Uji kompatibilitas terhadap HTML5 dilakukan dengan menyisipkan beberapa tag baru ke dalam script HTML yang digunakan di dalam aplikasi.

- Input Types Range.

Kode Program 15 Kode Program Range

```

1. <tr>
2.   <td class="style2">
3.     Umur Anda tahun ini
4.   </td>
5.   <td>
6.     <script type="text/javascript">
7.       function displayNilai() {
8.         var u = $("#umur").val();
9.         $("#umurSpan").html(u);
10.        }
11.      </script>
12.      <input type="range" min="0" max="100" name="umur"
13.      id="umur" value="20" onchange="displayNilai();" />
14.      <span id="umurSpan"></span>
15.    </td>
16.  </tr>

```

Kode Program 15 disisipkan ke dalam halaman *sign up* untuk dilihat apakah fungsi ini dapat berjalan pada *browser-browser* yang akan digunakan. Berdasarkan hasil pengujian, hanya *Google Chrome 19*, *Safari 5.1.7* dan *Opera 11.64* yang mendukung *input types Range*. Sedangkan untuk *Mozilla Firefox 12* dan *Internet Explorer 9*, fungsi *range* yang merupakan *slider* diubah menjadi bentuk *input* biasa. Tabel 2 menunjukkan hasil pengujian dari kelima *browser* yang digunakan.

Tabel 2 Hasil Pengujian *Input Types Range*

Mozilla Firefox Versi 12	Umur Anda tahun ini	<input type="text" value="20"/>	20
Internet Eksplorer Versi 9	Umur Anda tahun ini	<input type="text" value="20"/>	20
Google Chrome Versi 19	Umur Anda tahun ini	<input type="range" value="20"/>	20
Opera Versi 11.64	Umur Anda tahun ini	<input type="range" value="20"/>	20
Safari 5.1.7	Umur Anda tahun ini	<input type="range" value="20"/>	20

- Elemen *x-webkit-speech*

Elemen *x-webkit-speech* merupakan fitur baru HTML5 yang dirancang untuk mengisi *form* dengan perintah suara. Elemen *x-webkit-speech* disisipkan ke dalam *script* HTML5 dengan tujuan untuk melakukan uji kompatibilitas. Berdasarkan uji aplikasi pada *browser*, ditemukan hanya *Google chrome 19* yang mendukung penggunaan elemen *x-webkit-speech*. Kode Program yang menggunakan elemen *x-webkit-speech* di dalam aplikasi yang dibangun, ditunjukkan pada Kode Program 16.

Kode Program 4.25 Kode Program *X-webkit-speech*

```

1. <tr>
2.   <td class="style2">
3.     Nama</td>
4.   <td>
5.     <input required="nama" name="nama" type="text" required
x-
webkit-speech placeholder="nama anda."required/></td>
6. </tr>

```

Hasil implementasi *x-webkit-speech* yang kompatibel dengan *browser Google chrome 19* dapat dilihat pada Gambar 4.

Gambar 4 Implementasi *x-webkit-speech*

Pengujian selanjutnya dilakukan dengan menggunakan *website testhtml5.com* yang memberikan informasi mengenai seberapa besar *browser* yang digunakan mendukung HTML5. Hasil pengujian yang dilakukan menggunakan *website testhtml5.com*, ditampilkan pada Tabel 3

Tabel 3 Hasil Pengujian Kompatibilitas *Mozilla Firefox 12*

Mozilla Firefox 12	Mendukung	Tidak Mendukung
<!DOCTYPE html>	V	
triggers standards mode		
canvas element	V	
2D context	V	
Text	V	
video element	V	
Subtitle support		V
Poster image support		
MPEG-4 support		V
H.264 support		V
Ogg Theora support	V	
WebM support	V	
PCM audio support	V	
AAC support		V
MP3 support		V
Ogg Vorbis support	V	
WebM support	V	
HTML5 tokenizer	V	
XMLHttpRequest Level 2	V	
WebSocket	V	
Cross-document messaging	V	
WebM support		V
Grouping content elements	V	
ext-level semantic elements	V	
Dynamic markup insertion	V	
input type=text	V	
input type=search	V	
input type=tel	V	
input type=url	V	
input type=datetime		V
input type=date		V
input type=month		V
input type=week		V
input type=time		V
input type=datetime-local		V
input type=checkbox	V	
input type=number	V	
input type=image	V	
keygen		V
datalist	V	
textarea	V	
Geolocation	V	
Device Orientation	V	
3D context		V
FileReader API	V	
FileSystem API		V
Session Storage	V	

Local Storage	V	
IndexedDB	V	
Web Workers	V	
Shared Workers	V	
Access the webcam		V
Web Notifications		V
Full screen support	V	
Text selection	V	
Page Visibility	V	

Hasil pengujian kompatibilitas *Internet Explorer* 9 terhadap HTML5, dapat dilihat pada Tabel 4.

Tabel 4 Hasil Pengujian Kompatibilitas *Internet Explorer* 9

IE 9	Mendukung	Tidak Mendukung
<!DOCTYPE html>	V	
triggers standards mode		
canvas element	V	
2D context	V	
Text	V	
video element	V	
Subtitle support		V
Poster image support		V
MPEG-4 support		V
H.264 support	V	V
Ogg Theora support		V
WebM support		V
PCM audio support		V
AAC support	V	
MP3 support	V	
Ogg Vorbis support		
WebM support		V
HTML5 tokenizer		V
XMLHttpRequest Level 2		V
WebSocket		V
Cross-document messaging	V	
WebM support		V
Grouping content elements	V	
ext-level semantic elements	V	
Dynamic markup insertion	V	
input type=text	V	
input type=search		V
input type=tel		V
input type=url		V
input type=datetime		V
input type=date		V
input type=month		V
input type=week		V
input type=time		V
input type=datetime-local		V
input type=checkbox	V	
input type=number		V
input type=image	V	
keygen		V

datalist		V
textarea	V	
Geolocation	V	
Device Orientation		V
3D context		V
FileReader API		V
FileSystem API		V
Session Storage	V	
Local Storage	V	
IndexedDB		V
Web Workers		V
Shared Workers		V
Access the webcam		V
Web Notifications		V
Full screen support		V
Text selection		V
Page Visibility		V

Data pada Tabel 3 dan Tabel 4, merupakan beberapa data hasil uji kompatibilitas HTML5 terhadap *browser*. *Browser* yang dipakai dalam pengujian yaitu *Mozilla Firefox 12* dan *Internet explorer 9*. Dalam uji kompatibilitas, *Mozilla Firefox 12* cukup banyak mendukung penggunaan HTML5, meskipun belum secara penuh. *Internet explorer 9* memiliki tingkat kompatibilitas paling rendah dalam dukungan terhadap HTML5. Berdasarkan data pada Tabel 3 dan Tabel 4, maka dapat disimpulkan bahwa kompatibilitas *browser* terhadap HTML5 masih cukup kurang, terutama *internet explorer*. Dibandingkan dengan HTML4.01, maka dari sisi dukungan atau kompatibilitas terhadap *browser*, HTML4.01 masih lebih baik dibandingkan HTML5. Walaupun dikatakan bahwa sebagian besar *web browser* telah mendukung HTML5, tetapi berdasarkan pada hasil uji coba yang dilakukan dengan menggunakan *prototype* aplikasi dan juga hasil pengujian dengan *testhtml5.com* menunjukkan bahwa *browser* belum dapat mendukung HTML5 sepenuhnya. Berbeda dengan HTML4.01 yang didukung penuh oleh semua *browser*.

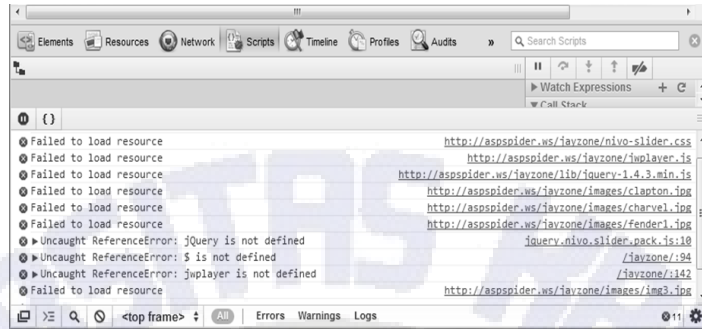
Analisis Pengaruh Penggunaan JavaScript Pada Kecepatan Akses dan Penggunaan Memori Aplikasi.

Berdasarkan hasil analisis pada proses validasi data dan kecepatan akses, dilakukan analisis lebih lanjut terhadap pengaruh penggunaan *javascript* pada aplikasi berbasis HTML4.01 yang dibangun.

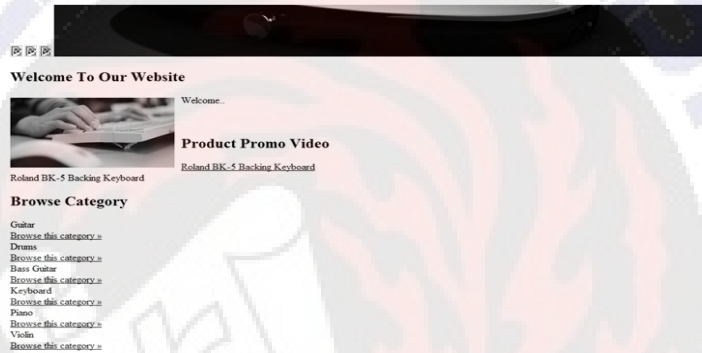
Dengan menggunakan *developer tools network* dilakukan pengujian proses eksekusi dan pengiriman data dari *server* ke *browser*. Pengujian dilakukan dengan menggunakan *browser Google Chrome* dan akses internet menggunakan *wifi.uksw.edu*. Berdasarkan pengujian, ditemukan beberapa permasalahan yaitu:

1. Pada akses internet dengan kecepatan yang terbatas, *file* kode *javascript* dan *css* pada *browser* yang digunakan untuk menguji, memiliki prioritas rendah untuk diunduh. Sehingga *file-file* tersebut sering gagal diunduh.
2. Jika *file javascript* gagal diunduh, maka halaman HTML yang menggunakan *file* tersebut tidak akan dapat menggunakan kode-kode program untuk melakukan fungsinya, misalnya: memutar *file* video. Kegagalan *javascript*

yang diunduh dan pengaruhnya pada aplikasi dapat dilihat pada Gambar 6 dan Gambar 7.



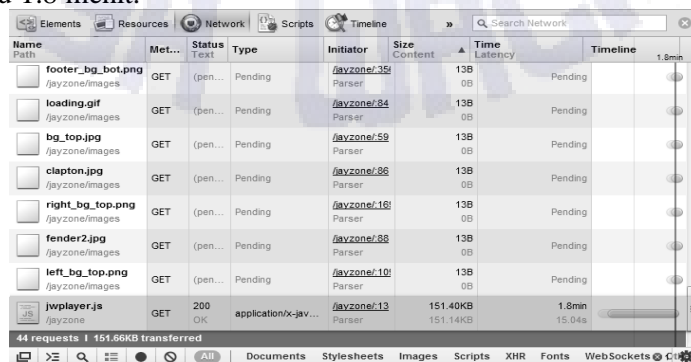
Gambar 6 Javascript Gagal Diunduh



Gambar 7 Pengaruh Pada Aplikasi

3. Penggunaan *javascript* relatif memperbanyak kode program yang digunakan pada aplikasi *online music store* HTML4.01. Kode *javascript* yang diunduh berefek menambah jumlah data yang harus diunduh oleh *browser*. Semakin panjang kode program, semakin besar pula jumlah *byte* yang dilewatkan pada koneksi internet.

Penggunaan *javascript* juga mempengaruhi waktu yang diperlukan oleh *browser* untuk mengunduh *file* dari *server*. Pada Gambar 8, bagian yang diterangkan dengan warna hijau, merupakan kode *javascript* yang digunakan untuk memutar video. Ukuran *file javascript* tersebut adalah 151.40KB, diunduh dalam waktu 1.8 menit.



Gambar 8 Javascript Yang Diunduh

Berdasarkan hasil analisis dan uji coba, perbandingan antara HTML 4.01 dengan HTML5 pada masalah ini yaitu, pada HTML5, beberapa validasi seperti validasi *email*, validasi angka untuk masukan nomor telepon, tidak menggunakan kode *javascript*. Hal ini memberikan keuntungan yaitu, file *javascript* tidak perlu diunduh, sehingga tidak terjadi masalah gagal unduh seperti pada HTML 4.1., juga mengurangi jumlah *byte* yang harus diunduh oleh *browser*.

Berdasarkan Hasil analisis dan perbandingan yang dilakukan, berikut ini dijabarkan beberapa kelebihan dan kekurangan HTML4.01 dan HTML5. **Kelebihan HTML4.01** adalah, (1) Mendukung banyak format video, karena menggunakan *plugin* untuk memutar video. *Plugin* memungkinkan seorang *web designer* dapat menentukan prediksi format apa saja yang dapat digunakan; (2) Tingkat kompatibilitas terhadap *browser* dan sistem operasi tinggi. **Sedangkan kelebihan HTML5** adalah, (1) Penulisan *script* lebih singkat sehingga mudah dihafal; (2) Menerapkan standar dalam penulisan *script* untuk memuat *file* video ke halaman *browser*; (3) Tidak menggunakan *plugin* tambahan untuk menampilkan video; (4) Memiliki kemampuan untuk menciptakan media *player* sendiri dengan memanfaatkan *video.js* dari *library java* dan *CSS*; (5) Validasi data lebih mudah dengan *tag*, atribut dan *input types* yang menjadi teknologi baru sehingga mengurangi penggunaan *javascript*; (6) Proses *loading* lebih cepat.

Kekurangan HTML4.01 adalah, (1) Dalam mendefinisikan `<doctype>` terdapat tiga penulisan *script*. Memberikan kebingungan kepada *web designer* dalam memilih `<doctype>`; (2) Tidak memiliki standar penulisan *script* untuk menampilkan dokumen multimedia misalnya video; (3) Sangat tergantung pada *plugin* tambahan untuk memutar video, walaupun sebenarnya menguntungkan karena dapat memainkan banyak format video. Tetapi hal ini menyebabkan *user* harus selalu menambahkan *plugin* tambahan; (4) Validasi data banyak mengandalkan *javascript* sehingga cukup rumit dan penulisannya panjang sehingga sulit dihafal; (5) Memerlukan waktu *loading* yang lebih lama untuk dapat ditampilkan ke dalam halaman *browser*. **Sedangkan Kekurangan HTML5** adalah, (1) Terbatas dalam mendukung format video; (2) Kompatibilitas dengan *browser* kurang.

Simpulan

Berdasarkan hasil penelitian yang dilakukan tentang analisis dan pembuatan *prototype* aplikasi, disimpulkan beberapa hal yaitu, (1) HTML5 memiliki standar penulisan *script* untuk memuat dokumen multimedia seperti video; (2) HTML4.01 tergantung pada penggunaan *javascript* dalam melakukan beberapa fungsinya jika dibandingkan dengan HTML5 yang tidak begitu tergantung pada penggunaan *javascript*; (3) Permasalahan serius yang dihadapi dalam penggunaan HTML5 adalah kompatibilitas dengan *browser*. Dalam menggunakan HTML5, seorang *web designer* harus benar-benar memperhatikan *browser* yang akan dipakai. (4) Penggunaan HTML4.01 masih dapat menjadi pilihan utama bagi *web designer*, dikarenakan HTML4.01 dapat dikatakan sebagai bahasa HTML yang paling stabil, yang digunakan sekarang. Berbeda dengan HTML5 yang hanya dapat kompatibel dengan beberapa *browser*; (5) Dibandingkan dengan penggunaan *javascript* seperti pada HTML4.01, penggunaan *script* HTML5 selain relatif mudah dan singkat, HTML5 lebih

menghemat penggunaan memori, relatif mempercepat proses pengiriman data dari *server* ke *browser* dan memperkecil kemungkinan terjadinya kegagalan unduh oleh *browser*.

Saran untuk penelitian ke depan yang dapat dilakukan adalah dengan menganalisis dan membandingkan lebih lanjut fungsi setiap elemen yang ada di dalam HTML4.01 dan HTML5; dan Mengkolaborasikan HTML5 sebagai teknologi baru dengan bahasa pemrograman lain, terutama dalam penanganan dokumen-dokumen multimedia, sehingga dapat membangun aplikasi berbasis *web* yang lebih fleksibel, sesuai kebutuhan dan perkembangan teknologi ke depan.

Daftar Pustaka

- [1] Van Kesteren, Anne., Pieter, Simon., 2012. *HTML5 Differences From HTML4*, www.w3c.org/TR/html5-diff/. Diakses 15 Juni 2012.
- [2] Jubilee Enterprise, 2011. *Step by Step HTML5: Cara Cepat Dan Mudah Menguasai Script HTML5*. Jakarta:Gramedia.
- [3] Widodo, Joko Purwadi., 2007. *Pemanfaatan File Format HTML Sebagai Wahana Cetak Pada Pemrograman Komputer*. Surakarta: Fakultas Teknik, Universitas Sebelas Maret.
- [4] Thabit, Nabel Amir., 2011. *Pencitraan 3 Dimensi Berbasis JavaScript Object Notation (JSON) Pada Hyper Text Markup Language 5 (HTML5)*. Bandung: Program Studi Ilmu Komputer, Universitas Pendidikan Indonesia.
- [5] Saputro W Hendro., *Pengertian Website dan Unsur-Unsurnya*, <http://www.balebengong.net/topik/teknologi/2007/08/01/pengertian-website-dan-unsur-unsurnya>. Diakses tanggal 20 Maret 2012.
- [6] Astamal Rio., 2005. *e-book:Mastering Code HTML*.
- [7] Khannedy Kurniawan Eko., 2008. *Aplikasi IT 1- HTML*.
- [8] Winarno Edy ST, M.Eng, Zaki Ali, SmitDev Community., 2010, *Easy Web Programming With PHP Plus HTML5*. Jakarta: Gramedia.
- [9] Gunawan Johanes, 2008. *e-book : Cepat Mahir ASP*, <http://www.ilmukomputer.org/2008/11/25/cepat-mahir-asp/>. Diakses 14 Maret 2012.
- [10] Sugianto, Bambang, 2012. <http://www.satublogs.com/web-design/tren-web-design-di-tahun-2012-yang-harus-anda-ikuti.html>. Diakses 25 April 2012.
- [11] Heshan, What is VideoJS?, <http://video.heidisotf.com/documentation/what=videojs>. Diakses 27 Maret 2012