

ANALISIS UNJUK KERJA APLIKASI *DIGITAL SIGNAGE* XIBO PADA *SINGLE BOARD COMPUTER* RASPBERRY PI

Bimo Sunarfri Hantono¹⁾, Guntur Dharma Putra²⁾

^{1), 2)} *eSystems Lab, Jurusan Teknik Elektro dan Teknologi Informasi*
Universitas Gadjah Mada
Jl. Grafika No.2, Kampus UGM, Yogyakarta 55281
Email : bhe@ugm.ac.id¹⁾, guntur.dharma@mail.ugm.ac.id²⁾

Abstrak

Di era digital, papan informasi bermigrasi dari bentuk konvensional ke bentuk digital, sehingga dikenal istilah digital signage. Bentuk digital signage yang kerap digunakan adalah komputer yang terhubung ke layar monitor untuk menampilkan informasi. Salah satu aplikasi digital signage yang mendukung manajemen konten dan penjadwalan berbasis Web adalah Xibo. Sayangnya, implementasi Xibo pada umumnya menggunakan komputer desktop yang terbilang kurang efisien jika hanya digunakan sebagai peranti digital signage. Xibo pun juga tidak memiliki dukungan sama sekali dalam implementasinya pada single board computer seperti Raspberry Pi. Penelitian ini bertujuan untuk mengimplementasikan Xibo pada Raspberry Pi dan menganalisis unjuk kerjanya. Pada awalnya akan dilakukan porting aplikasi Xibo client dari Ubuntu dengan melakukan kompilasi kode sumber dari dependensi aplikasi Xibo sehingga dapat diinstallkan pada Raspberry Pi. Kemudian segala fitur utama yang dimiliki oleh Xibo diuji dan dianalisis unjuk kerjanya untuk kemudian dibandingkan dengan Xibo client di Windows yang mana memiliki dukungan penuh dari pengembangnya.

Kata kunci: Digital Signage, Single Board Computer, Raspberry Pi, Xibo.

1. Pendahuluan

Saat ini sangat umum kita perhatikan suatu papan informasi yang ada di perkantoran, di gedung perkuliahan, di bandar udara [4] maupun stasiun kereta api yang berupa suatu monitor layar datar yang dikendalikan oleh suatu komputer *desktop*. Dari satu sisi hal ini lebih mudah dioperasikan daripada menggunakan metoda lama menggunakan papan pengumuman ataupun papan penunjuk arah. Dengan menggunakan komputer informasi bisa lebih dinamis dan media yang ditampilkan lebih beragam. Papan informasi berbasis komputer tersebut saat ini biasanya memanfaatkan komputer *desktop* yang terhubung jaringan dan mampu untuk dikendalikan secara jarak jauh untuk mengatur apa yang akan ditampilkan [4]. Penggunaan komputer *desktop* yang hidup setiap waktu tersebut mengakibatkan

pemborosan penggunaan energi listrik, selain itu kemampuan satu unit komputer yang hanya dipergunakan untuk menampilkan informasi yang relatif statis juga dirasa terlalu berlebihan dan dilain sisi biaya pembelian beserta pemeliharaan unit (*hardware* dan *software*) yang juga tidak murah.

2. Penelitian Serupa

Devin K. Farmer memanfaatkan *Raspberry Pi* sebagai papan informasi di perpustakaan [1]. Apa yang dilakukannya adalah menggunakan *LCD Display* sebagai media penampil dan informasi yang ditampilkan berupa video berformat mp4. Pada penelitian tersebut belum ada interaksi antara klien dan *server* dalam pertukaran data.

Adrian O. Connor [2] dalam proposal penelitiannya menyebutkan bahwa ia akan membuat papan informasi berbasis *Raspberry Pi* dan menyambungkannya ke jaringan komputer. Komunikasi yang terjalin menggunakan protokol *ZeroMQ* dan klien menggunakan aplikasi buatannya sendiri. Sedangkan pengendalian dilakukan jarak jauh dengan komputer lain.

Penelitian tentang implementasi Xibo pernah dilakukan untuk papan informasi di PT GMF Aero Asia [3]. Billy Lukmanfiandy dalam laporan kerja praktiknya menyebutkan bahwa ia menggunakan komputer *desktop* sebagai Xibo *client* yang nantinya menampilkan informasi dari *server*. Sayangnya, penggunaan komputer *desktop* di sini adalah suatu bentuk pemborosan karena sumber daya dan energi yang digunakan tidak sebanding dengan apa yang dikerjakan.

3. Metodologi

Langkah-langkah dalam penelitian ini yaitu:

1. Analisis Kebutuhan: analisis kebutuhan fungsional, kebutuhan non-fungsional, dan kebutuhan melakukan penelitian.
2. Persiapan Xibo *client*: meliputi instalasi sistem operasi Raspbian sebelum penginstalan Xibo.
3. Persiapan Xibo *server*: instalasi perangkat-perangkat lunak yang diperlukan guna berjalannya server Xibo.

4. Konfigurasi sistem: konfigurasi yang dilakukan untuk menghubungkan *client* dan *server* pada satu jaringan untuk eksperimen.
5. Pengujian Kinerja Xibo: tahap utama dari penelitian, nantinya hasil akan dibandingkan dengan kinerja Xibo pada komputer *desktop*.

3.1. Analisis Kebutuhan

Penelitian ini menggunakan alat dan bahan seperti berikut:

- Dua buah *Raspberry Pi* Model B,
- Dua buah Monitor Zyrex ZM 845,
- Laptop MacBook Pro,
- Satu unit komputer *desktop* berbasis Windows,
- Kabel Ethernet secukupnya dan sebuah Hub.

3.2. Persiapan Xibo Client

Aplikasi Xibo tidak bisa langsung diinstal ke dalam *Raspberry Pi*. Instalasi yang dilakukan adalah melakukan *porting* python *client* untuk Ubuntu agar dapat berjalan di *Raspberry Pi* dengan sistem operasi Raspbian.

Untuk melakukan *porting*, diperlukan beberapa *dependency library* guna meng-*compile* kode-kode sesumbernya. Library yang diperlukan antara lain: *berkelium*, *libbrowsermode*, dan *libavg* [5]. Proses *compile* sendiri memakan waktu kurang lebih tiga hari dikarenakan *resource* terbatas dari *Raspberry Pi*. Namun, proses ini lebih stabil dibandingkan *crosscompiling* [5].

3. Persiapan Xibo Server

Xibo dapat berjalan pada sistem operasi berbasis Windows dan *NIX (Linux, BSD, Mac OS) [6] namun dengan syarat sistem operasi tersebut sudah memiliki beberapa aplikasi yang digunakan untuk Xibo berjalan. Karena Xibo *server* adalah aplikasi berbasis Web, kebutuhannya adalah Web server, PHP, dan MySQL [7]. Namun PHP yang diperlukan harus sudah memiliki *library* MySQL, GD, JSON, dan module SOAP [7].

Aplikasi Xibo *server* dapat diunduh secara gratis [6] dalam bentuk *file-file* PHP. *File-file* ini kemudian di-*extract* ke dalam direktori web server anda untuk kemudian dilakukan instalasi guna mengkonfigurasi kata sandi untuk admin dan menghubungkan Xibo dengan *server* basis data.

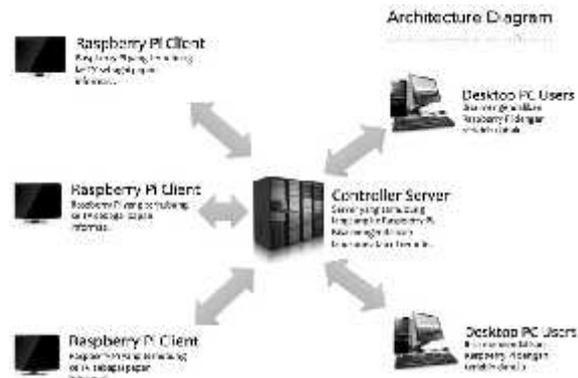
3.4. Konfigurasi sistem

Untuk menampilkan informasi, *Raspberry Pi* harus terhubung dengan layar monitor yang memiliki *port* DVI atau HDMI. *Raspberry Pi* tidak mempunyai *port* VGA yang umum digunakan pada sebuah komputer. Namun, *Raspberry Pi* memiliki *port* HDMI yang bisa dihubungkan ke DVI menggunakan *converter*. Setelah siap, *Raspberry Pi* dapat dimodelkan seperti terlihat pada Gambar 1.



Gambar 1. Pemodelan sistem client.

Setelah *client* dan *server* siap, keduanya harus saling dihubungkan ke dalam satu jaringan yang dapat saling berkomunikasi. Secara *default* *Raspberry Pi* Model B sudah memiliki *port ethernet*. Oleh karena itu, antara *client* dan *server* saling terhubung melalui kabel *ethernet* yang terkoneksi melalui sebuah Hub. Arsitektur dari sistem dapat dilihat pada Gambar 2.



Gambar 2. Arsitektur dari sistem yang digunakan.

3.5. Uji Coba Kinerja Xibo

Eksperimen yang dilakukan mengujikan fitur-fitur utama yang dimiliki oleh Xibo. Selain dari itu, responsivitas dari pengendalian *client* oleh *server* juga akan diuji. Fitur-fitur Xibo yang akan diujikan antara lain:

- Penjadwalan dan responsivitas,
- informasi teks dengan dan tanpa animasi,
- gambar statis,
- animasi flash,
- halaman Web, dan
- powerpoint.

4. Hasil dan Pembahasan

4.1. Menampilkan Informasi Berbasis Teks

Uji coba ini ditujukan untuk mengujikan kemampuan Xibo pada *Raspberry Pi* dan komputer berbasis Windows untuk menampilkan informasi paling dasar dari fitur yang ada pada *Raspberry Pi* yaitu menampilkan informasi berbasis teks.

Saat dilakukan uji coba di laboratorium, Xibo pada *Raspberry Pi* terbukti mampu menampilkan informasi berbasis teks. Begitu pula pada komputer *desktop* berbasis Windows, informasi berbasis teks dapat tertampil.

4.2. Menampilkan Informasi Berbasis Teks dengan Animasi

Uji coba ini ditujukan untuk menguji kemampuan Xibo pada *Raspberry Pi* dan komputer berbasis Windows untuk menampilkan informasi berbasis teks namun dengan sedikit animasi sederhana.

Raspberry Pi dapat menampilkan animasi, namun terkesan patah-patah karena kemampuan komputasi grafis dari *Raspberry Pi* yang sangat terbatas. Sedangkan komputer *desktop* berbasis Windows dapat menampilkan animasi dengan lancar.

4.3. Menampilkan Gambar

Uji coba ini ditujukan untuk menguji kemampuan Xibo pada *Raspberry Pi* dan komputer berbasis Windows untuk menampilkan gambar statis.

Hasil uji laboratorium juga membuktikan kedua Xibo yang terinstall pada sistem yang berbeda mampu menampilkan gambar diam (*still image*).

4.4. Menampilkan Halaman Web

Uji coba ini ditujukan untuk menguji kemampuan lanjut Xibo pada *Raspberry Pi* dan komputer berbasis Windows untuk menampilkan halaman Web. Penampilan halaman Web membutuhkan klien yang juga terkoneksi dengan internet.

Hasil uji coba menunjukkan bahwa *Raspberry Pi* dan komputer *desktop* mampu menampilkan halaman web dengan baik. Bahkan Xibo juga mendukung animasi dengan JavaScript.

4.5. Menampilkan Animasi Flash

Uji coba ini ditujukan untuk menguji kemampuan lanjut Xibo pada *Raspberry Pi* dan komputer berbasis Windows untuk menampilkan gambar animasi berbasis flash.

Sangat disayangkan bahwa *Raspberry Pi* tidak mampu menampilkan animasi flash. Saat *Raspberry Pi* dijadwalkan untuk menampilkan animasi berbasis flash, hanya tampilan layar hitam yang muncul.

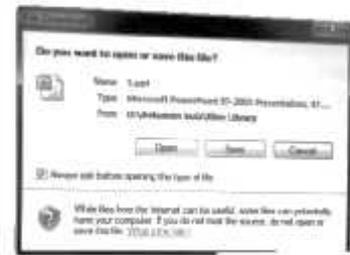
Lain halnya dengan *Raspberry Pi*, komputer *desktop* justru mampu menampilkan animasi flash dengan lancar.

4.6. Menampilkan Presentasi Powerpoint

Uji coba ini ditujukan untuk menguji kemampuan lanjut Xibo pada *Raspberry Pi* dan komputer berbasis Windows untuk menampilkan file ppt dari Microsoft Power Point.

Hasil percobaan menunjukkan bahwa *Raspberry Pi* yang terinstall aplikasi klien Xibo tidak mampu menampilkan file Power Point. Saat dijadwalkan, seperti animasi flash, hanya tertampil layar hitam.

Begitu pula dengan komputer *desktop*, namun komputer *desktop* menunjukkan dialog untuk menyimpan atau membuka file ppt seperti dapat dilihat pada Gambar 3.



Gambar 3. Komputer *desktop* menampilkan dialog file ppt.

Pada dialog tersebut, jika tombol *open* ditekan maka akan tertampil jendela Power Point baru menampilkan file ppt tersebut. Jika tombol *save* ditekan, akan muncul dialog lain untuk memilih direktori di mana file ingin disimpan. Jadi, dapat disimpulkan bahwa komputer *desktop* juga tidak mampu menampilkan file ppt.

4.7. Responsivitas Klien dan Server

Keenam uji coba di atas, diujikan dengan penjadwalan waktu. Jadi tiap klien diberikan slot waktu tertentu untuk menampilkan informasi tertentu. Uji coba ini ditujukan untuk menguji responsivitas dari klien dan *server*.



Gambar 4. Uji coba penjadwalan pada komputer *desktop*.

Uji coba penjadwalan pada komputer *desktop* berjalan hampir tanpa *delay*, hanya berselang kurang dari sepuluh detik dari jadwal, tampilan pada klien sudah langsung

berubah. Pada Gambar 4 dapat dilihat bahwa pada pukul 11:18:05 komputer menampilkan animasi flash dan selang 15 menit kemudian, pada pukul 11:33:05 komputer menampilkan halaman Web seperti jadwal.



Gambar 5. Uji coba penjadwalan pada Raspberry Pi.

Raspberry Pi juga merespon penjadwalan dari server, namun dengan waktu tunda sekitar satu menit. Seperti yang dapat dilihat pada Gambar 5, pada pukul 2:43:48, Raspberry Pi A dan B menampilkan informasi teks, dan pada pukul 3:08:49, kedua Raspberry sudah menunjukkan informasi yang berbeda yaitu gambar dan teks beranimasi.

4.8. Hasil Penelitian

Hasil ketujuh uji coba di atas dapat disimpulkan dalam Tabel 1.

Tabel 1. Tabel perbandingan kinerja Xibo di Raspberry dan Desktop-PC.

	Raspberry Pi	Desktop-PC
Menampilkan Informasi Teks	Bisa	Bisa
Menampilkan Informasi Teks dengan Animasi	Bisa (patah-patah)	Bisa
Menampilkan Gambar	Bisa	Bisa
Menampilkan Halaman Web	Bisa	Bisa
Menampilkan Animasi Flash	Tidak	Bisa
Menampilkan Slide Powerpoint	Tidak	Tidak
Responsivitas (delay)	+/- 1 menit	< 10 detik

5. Kesimpulan dan Saran untuk Penelitian Selanjutnya

Penelitian ini meneliti tentang unjuk kerja aplikasi digital signage Xibo dalam implementasinya pada single board computer Raspberry Pi. Hasil penelitian menunjukkan bahwa aplikasi Xibo dapat diimplementasikan pada Raspberry Pi dengan beberapa instalasi dan kompilasi dari library yang diperlukan untuk Xibo. Namun seperti yang dapat dilihat pada hasil penelitian, kinerja aplikasi Xibo pada Raspberry Pi kurang baik bila dibandingkan aplikasi Xibo yang terinstall pada komputer Windows. Xibo pada Raspberry Pi memiliki kemampuan yang sama dengan python client di Ubuntu, namun dengan kecepatan yang kurang dikarenakan resource yang terbatas.

Penelitian berikutnya lebih baik berorientasi pada pengembangan aplikasi papan informasi digital terdistribusi khusus untuk Raspberry Pi karena kebutuhannya yang khusus. Faktor-faktor yang perlu diperhatikan antara lain:

- Kemudahan dalam meng-install klien,
- Pengendalian yang realtime, dan
- Kemampuan untuk membuka animasi flash atau Powerpoint.

Daftar Pustaka

[1] D. K. Farmer, "Raspberry Pi in the Library", Devin K. Farmer, [Online]. Tersedia: <http://devinfarmer.com/library/14-raspberry-pi-in-the-library> [Diakses 11 April 2013].

[2] A. O. Connor, "Writing a display-board application for Raspberry Pi", Raspberry Pi Forum, [Online]. Tersedia: <http://www.raspberrypi.org/phpBB3/viewtopic.php?f=31&t=6339>, May 22, 2012 [Diakses 11 April 2013].

[3] B. Lukmanfiandy, dkk, *Perancangan Dan Implementasi Digital Signage Pada Media Informasi PT GMF AEROASIA Menggunakan XIBO*, Yogyakarta: Jurusan Teknik Elektro dan Teknologi Informasi UGM, 2013.

[4] M. Arif, *Kualitas Pelayanan Publik di Bandara Internasional Sultan Hasanuddin Makassar*, Makassar: Sekolah Pascasarjana UNHAS, 2011.

[5] Hellbender, "Building Berkeley, libbrowsernode and libAVG on the RaspberryPi and installing them for use with the Xibo package.", Raspberry Pi Project, [Online]. Tersedia: http://rpiserver.breggen.nl/building_xibo.html [Diakses 26 October 2013].

[6] D. Garner, dkk, "Xibo – Digital Signage Documentation.", Xibo Digital Signage, [Online]. Tersedia: <http://xibo.org.uk/docs/> [Diakses 27 October 2013].

[7] D. Garner, dkk, "What is Xibo?", Xibo Digital Signage, [Online]. Tersedia: <http://xibo.org.uk/about/> [Diakses 27 October 2013].

Biodata Penulis

Bimo Sunarfri Hantono, memperoleh gelar Sarjana Teknik (S.T.), Jurusan Teknik Elektro, Universitas Gadjah Mada, Yogyakarta, lulus tahun 2000. Memperoleh gelar Master of Engineering (M.Eng.), Electrical and Electronic Engineering Department, Nanyang Technology University, Singapore, lulus tahun 2006. Saat ini menjadi Dosen di Jurusan Teknik Elektro

dan Teknologi Informasi, Universitas Gadjah Mada,
Yogyakarta.

Guntur Dharma Putra, saat ini sedang menyelesaikan skripsi untuk memperoleh gelar Sarjana Teknik (S.T.), di Jurusan Teknik Elektro dan Teknologi Informasi, Universitas Gadjah Mada, Yogyakarta.

