

SISTEM INFORMASI PENAMPIL PENGUMUMAN FAKULTAS TEKNIK UNIVERSITAS DIPONEGORO

Fajar Yogi Hernawan, Kodrat Iman Satoto., Eko Didik Widiyanto.

*Program Studi Sistem Komputer, Fakultas Teknik, Universitas Diponegoro
Jalan Prof. Sudharto,SH, Kampus Undip, Tembalang, Semarang, Indonesia 50275*

Abstract - Engineering Faculty is the biggest faculty with the highest number of departements and students in the Diponegoro University. The idea made SIPP (System Information Viewer Announcements Faculty of Engineering, University of Diponegoro) appears is to help disseminate information and stories of pride born from Engineering Faculty to become the impetus and motivation for more achievement. SIPP maked with 5(five) stages: requirement, analysis, design, implementation, testing. SIPP is built using the programming language PHP, HTML, CSS, Bootstrap, Javascript, and MySQL database. SIPP is expected to convey information and the pride stories of the Engineering Faculty in the multimedia. The output of the SIPP is the information shown via LCD TV screen ratio of 16: 9 which blend of images, video and running text. SIPP expected to be implemented in the lobby Engineering Faculty dean and in every department in it. Users SIPP is divided into 2 (two) they are admin faculty as the main admin and admin department as sub-admin. SIPP tested with 2 (two) test that is functionality testing by black box and performance testing with performance-analyzer application. With the implementation of SIPP environment Faculty of Engineering will be an additional alternative for the community Faculty of Engineering to obtain information.

Keywords: Information, SIPP, multimedia, Engineering Faculty.

Abstrak – Fakultas Teknik adalah fakultas dengan jumlah jurusan dan mahasiswa terbanyak di Universitas Diponegoro. Gagasan dibuatnya SIPP (Sistem Informasi Penampil Pengumuman Fakultas Teknik Universitas Diponegoro) muncul adalah untuk membantu menyebarkan informasi dan cerita kebanggaan dari Fakultas Teknik yang menjadi dorongan dan motivasi untuk lebih berprestasi. SIPP dibuat dengan 5(lima) tahap : rencana kebutuhan, analisis kebutuhan, perancangan desain, perancangan sistem, dan pengujian. SIPP dibangun menggunakan bahasa pemrograman PHP, HTML, CSS, Bootstrap, Javascript, dan basisdata MySQL. SIPP diharapkan dapat menyampaikan informasi dan cerita kebanggaan dari Fakultas Teknik dalam bentuk multimedia. Keluaran dari SIPP ditampilkan melalui LCD TV rasio 16 : 9 yang merupakan perpaduan antara gambar, video, dan teks berjalan. SIPP diharapkan dapat dipasang di lobi dekanat Fakultas Teknik dan di setiap jurusan didalamnya. Pengguna SIPP dibagi menjadi 2(dua) yaitu admin fakultas sebagai admin utama dan admin jurusan sebagai sub-admin. SIPP diuji dengan 2(dua) pengujian yaitu uji fungsionalitas dengan *black box* dan uji performansi dengan aplikasi *performnce-analyzer*. Dengan penerapan SIPP akan menjadi alternatif untuk warga Fakultas Teknik mendapatkan informasi.

Kata kunci : informasi, SIPP, multimedia, fakultas teknik.

PENDAHULUAN

Dalam perkembangannya, informasi dikelola secara lebih baik ke dalam suatu sistem yang disebut Sistem Informasi. Sistem Informasi dirancang untuk memudahkan para pencari informasi dalam memenuhi kebutuhannya akan informasi. Sistem informasi menerapkan teknologi informasi dalam implementasinya. Dengan adanya bantuan teknologi informasi yang kita tahu membantu pekerjaan manusia, informasi tidak lagi sulit untuk didapatkan yang dikemas sedemikian rupa agar memudahkan dan membuat nyaman penggunaannya.

Di Fakultas Teknik Universitas Diponegoro Semarang, belum ada sistem yang saling terintegrasi untuk menampilkan dan memberikan informasi kepada setiap pengunjung yang ada di dekanat yang terintegrasi dengan kantor masing-masing jurusan. Dengan adanya Sistem Informasi Penampil Pengumuman di Fakultas Teknik Universitas Diponegoro Semarang, yang selanjutnya ditulis SIPP, diharapkan mampu membantu pengunjung; baik dari kalangan mahasiswa, dosen, akademisi, maupun masyarakat umum dalam mendapatkan informasi umum tentang Fakultas Teknik, maupun jurusan-jurusan yang ada didalamnya.

Penelitian terdahulu dan sejenis sebelum adanya SIPP salah satunya dilakukan oleh Riko^[1] yang berjudul Penampil Informasi Menggunakan Mikrokontroler Arduino 328 Berbasis Web. Pada penelitian ini, media penampil informasi menggunakan LED Matriks berukuran 8 x 64 yang berisi informasi pesan yang akan dikirimkan dengan memanfaatkan komunikasi serial antara mikrokontroler dan *web server* yang ditanamkan pada Raspberry PI. Pada perancangannya, hasil yang diharapkan adalah sistem dapat menampilkan informasi berupa teks berjalan dari kanan ke

kiri dengan jumlah 199 karakter dan mampu dikendalikan dari jarak jauh melalui internet berantarmuka aplikasi web. Penampil Informasi Menggunakan Mikrokontroler Arduino 328 Berbasis Web ini dibangun menggunakan bahasa pemrograman PHP dan basisdata MySQL. Penelitian lain yang sejenis pula oleh Rizal Panuntun^[2] adalah Perancangan Papan Informasi Berbasis Web Menggunakan Raspberry-Pi dimana peneliti mengembangkan *Digital Sinage* dengan membuat aplikasi penampil beserta perangkatnya. Sistem hanya mengembangkan informasi berupa teks dan video, sedangkan keluaran ditampilkan menggunakan raspberry-pi untuk menampilkan ke LCD. Bahasa pemrograman yang digunakan dalam sistem ini adalah PHP dengan basis data MySQL.

METODOLOGI PENELITIAN

Dalam pengembangan SIPP rencana kebutuhan membahas bagaimana SIPP bekerja, aktor yang berperan dalam SIPP, dan gambaran umum topologi SIPP. Analisis kebutuhan membahas gambaran umum penggunaan sistem, kebutuhan fungsional, dan kebutuhan non-fungsional. Perancangan desain membahas perancangan DFD, perancangan basis data, pembuatan diagram alir sistem, dan perancangan antarmuka. Aktivitas perancangan sistem adalah pembuatan basis data dan implementasi antarmuka. Pengujian dalam pengembangan SIPP dilakukan dengan 2 (dua) pengujian yaitu uji performansi dengan mengetes kecepatan akses aplikasi SIPP dan uji *black box*.

PERANCANGAN SISTEM

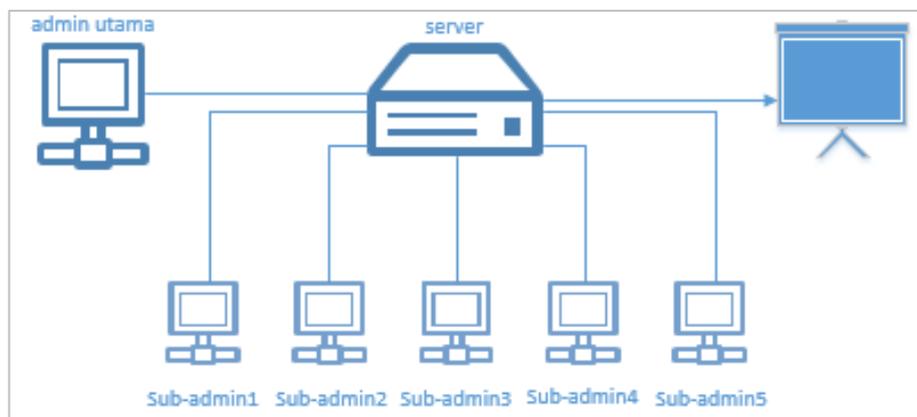
SIPP berfungsi untuk mengelola pengumuman yang akhirnya akan ditampilkan ke penampil yang dipasang disetiap lobi kantor di lingkungan Fakultas

Teknik, Universitas Diponegoro. Dalam menjalankan fungsinya SIPP digunakan oleh aktor yang bernama admin utama dan sub-admin.

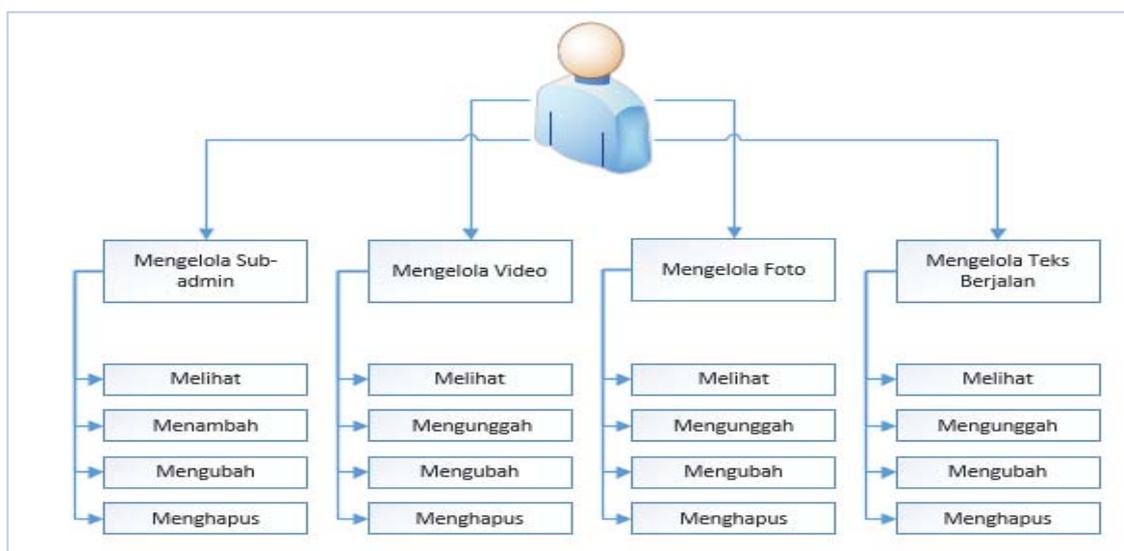
Admin utama bertugas sebagai pengelola dan penyedia informasi ditingkat fakultas dan mengelola fungsi sub-admin. Secara rinci, admin utama bertugas mengunggah informasi berupa teks berjalan, video, dan foto yang akhirnya tampil sebagai pengumuman di penampil. Wewenang lain dari admin utama adalah membuat akun

sub-admin dan mengelolanya. Dan admin utama berhak melihat seluruh data yang telah diunggah oleh sub-admin kecuali *password* akun.

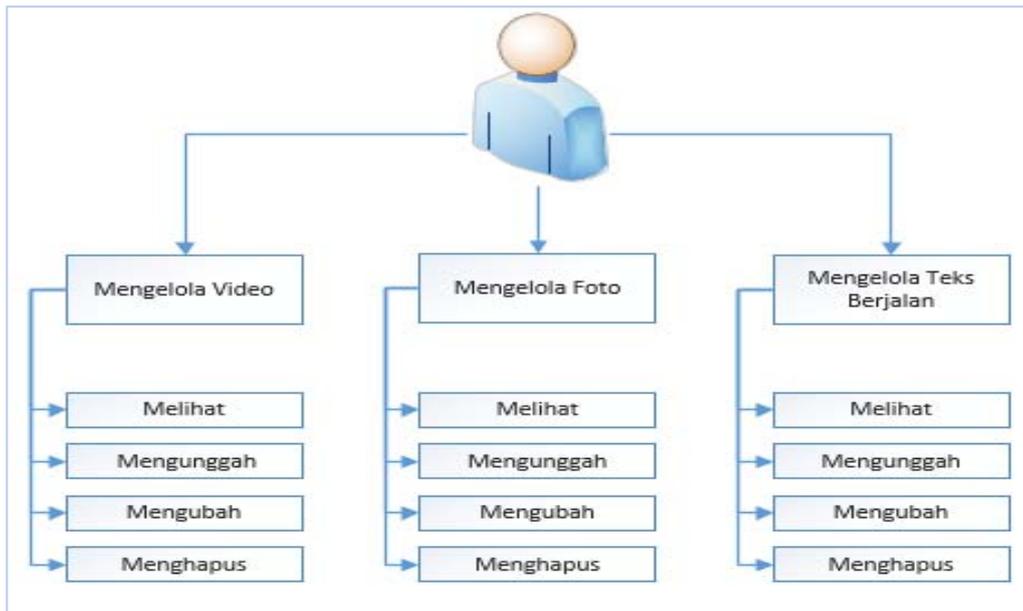
Sub-admin bertugas mengunggah informasi jurusan berupa teks berjalan, video, dan foto. Admin utama juga berkewajiban memperbarui data profilnya sesaat setelah mendapatkan akun. Namun sub-admin tidak berhak untuk melihat data yang dimiliki sub-admin lain.



Gambar 1 Gambaran umum topologi SIPP



Gambar 2 Bagan Aktivitas Admin Utama



Gambar 3 Bagan Aktivitas Sub-Admin

Sub-admin disini adalah petugas dari masing-masing jurusan yang bertugas mengelola sumber informasi jurusan.

SIPP menggunakan LCD TV rasio 16:9 sebagai penampil, server untuk menyimpan data, dan personal komputer di dekanat dan masing-masing jurusan, perangkat audio, dan *web browser* untuk menjalankan sistem. Berikut adalah diagram yang menunjukkan konsep SIPP. Gambar 1 adalah topologi SIPP secara umum. Gambar 2 adalah bagan aktivitas admin utama, sedangkan Gambar 3 adalah bagan aktivitas sub-admin.

IMPLEMENTASI DAN PENGUJIAN

Bab ini membahas implementasi dan pengujian SIPP (Sistem Informasi Penampil Pengumuman Fakultas Teknik Universitas Diponegoro) sehingga menghasilkan suatu aplikasi yang dapat bekerja sebagaimana

mestinya. Pengujian sistem dilakukan dengan menggunakan *web browser* Microsoft Edge. Pada tahap pengujian sistem akan menggunakan pengujian *Black Box* dan uji performansi.

Tahap Pembuatan

Pembuatan aplikasi merupakan tahap menerjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh sistem itu sendiri yang harus dilakukan secara terurut. Tahap pertama yang harus dilakukan adalah pembuatan Basisdata dan dilanjutkan dengan tahap pembuatan aplikasi yang berupa tampilan antar muka dari sistem.

Basisdata yang digunakan untuk membangun SIPP adalah MySQL. Data disimpan dalam bentuk tabel-tabel. Tabel-tabel dalam basisdata SIPP antara lain tabel *user*, tabel *photos*, tabel *text*, tabel *videos*.

Tabel 1 *User*

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 nama	varchar(30)			No	None	
<input type="checkbox"/>	3 password	varchar(30)			No	None	
<input type="checkbox"/>	4 jurusan	varchar(30)			No	None	
<input type="checkbox"/>	5 email	varchar(30)			No	None	
<input type="checkbox"/>	6 telepon	varchar(12)			No	None	
<input type="checkbox"/>	7 otoritas	varchar(30)			No	None	
<input type="checkbox"/>	8 tanya	varchar(1000)			No	None	
<input type="checkbox"/>	9 jawab	varchar(1000)			No	None	
<input type="checkbox"/>	10 status	int(1)			No	None	

Tabel *user* merupakan tabel yang berfungsi untuk penyimpanan data dari pengguna. Struktur data pengguna seperti ditunjukkan pada Tabel 1.

Atribut *id* yang dibuat sebagai *auto_increment* yang bertujuan agar ketika ada data baru yang masuk maka secara otomatis nilai *id* akan bertambah satu. Atribut yang lain yaitu atribut *nama* dan *password* yang digunakan ketika *login* ke halaman *user* dimana pada atribut *password* sebagai kata sandi. Kolom *jurusan*, *email*, dan *telepon* untuk melengkapi data. Kolom *otoritas* digunakan untuk menentukan hak akses dari masing-masing pengguna. Kolom *tanya* dan *jawab* digunakan ketika pengguna

mengalami lupa *password*. *olom* status digunakan untuk penanda, apakah akun pengguna tersebut sedang aktif atau tidak.

Tabel *photos* digunakan untuk menyimpan data foto-foto. Struktur foto ditampilkan dalam Tabel 2. Pada struktur tabel *photos*, terdapat beberapa atribut yaitu atribut *id* yang dibuat sebagai *primary key* yang menandakan bahwa *id* merupakan kunci dari tabel yang unik yang digunakan untuk membedakan data-data yang tersimpan antara satu dengan yang lainnya dan *auto_increment* yang bertujuan agar ketika ada data baru yang masuk maka secara otomatis nilai *id* akan bertambah satu.

Tabel 2 *Photo*

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 id	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 judul	varchar(1000)			No	None	
<input type="checkbox"/>	3 nama	varchar(1000)			No	None	
<input type="checkbox"/>	4 tgl_ditambahkan	varchar(30)			No	None	
<input type="checkbox"/>	5 tgl_kadaluarsa	date			No	None	
<input type="checkbox"/>	6 permanen	int(1)			No	None	
<input type="checkbox"/>	7 sumber	varchar(30)			No	None	

Atribut nama yang berfungsi sebagai penyimpanan nama foto dan atribut status untuk menyimpan status penggunaan foto. Atribut nama ini pula yang digunakan untuk mengambil foto dari folder untuk ditampilkan disistem. Atribut *tgl_ditambahkan* berisi waktu dan tanggal ketika foto diunggah. Atribut *tgl_kadaluarsa* berisi tanggal foto kadaluarsa dan akan dihapus. Atribut *permanen* digunakan untuk status foto apakah permanen atau tidak. Dan

kolom sumber berisi jurusan dari akun pengunggah foto.

Tabel *text* digunakan untuk menyimpan teks yang akan tampil berjalan beserta dengan data-datanya. Struktur data teks berjalan ditampilkan pada Tabel 3.

Pada tabel *teks* ini terdapat atribut id yang dibuat sebagai *primary key* dan *auto increment*. Atribut *konten* berisi teks yang akan ditampilkan berjalan oleh sistem.

Tabel 3 *Text*

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id	int(11)			No	None	AUTO_INCREMENT
2	konten	text			No	None	
3	tgl_ditambahkan	varchar(30)			No	None	
4	tgl_kadaluarsa	date			No	None	
5	permanen	int(1)			No	None	
6	sumber	varchar(30)			No	None	

Atribut *tgl_ditambahkan* berisi waktu dan tanggal ketika teks diunggah. Atribut *tgl_kadaluarsa* berisi tanggal teks kadaluarsa dan akan dihapus. Atribut *permanen* digunakan untuk status teks apakah permanen atau tidak. Dan atribut *sumber* berisi jurusan dari akun pengunggah teks.

ditampilkan pada Tabel 4. Pada tabel ini berisi atribut id yang dibuat sebagai *primary key* dan *auto increment*. Atribut *nama* berisi keterangan nama video yang telah diunggah. Atribut *judul* berisi judul yang pengguna buat pada nama video. Atribut *tgl_ditambahkan* berisi waktu dan tanggal ketika video diunggah.

Tabel *videos* digunakan untuk menyimpan data-data video. Struktur video

Tabel 4 *Videos*

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id	int(11)			No	None	AUTO_INCREMENT
2	nama	text			No	None	
3	judul	text			No	None	
4	tgl_ditambahkan	datetime			No	CURRENT_TIMESTAMP	
5	tgl_kadaluarsa	date			No	None	
6	permanen	int(1)			No	None	
7	sumber	varchar(30)			No	None	

Atribut `tgl_kadaluarsa` berisi tanggal video kadaluarsa dan akan dihapus. Atribut permanen digunakan untuk status teks apakah permanen atau tidak. Dan atribut sumber berisi jurusan dari akun pengunggah teks.

Implementasi antarmuka menggambar tampilan dari aplikasi yang dibangun

yaitu implementasi antarmuka SIPP Berikut ini adalah tampilan menu-menu antarmuka yang terdapat pada SIPP.

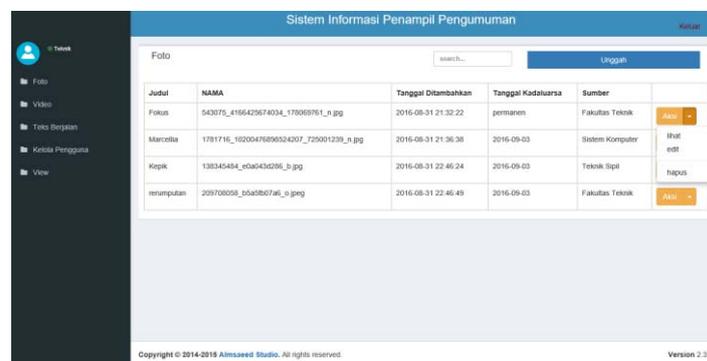
Gambar 4 merupakan halaman awal ketika pengguna membuka SIPP. Pengguna harus memasukkan nama dan *password* dengan benar supaya bisa masuk kesistem.



Gambar 4 Halaman masuk sistem

Gambar 5 adalah halaman awal setelah pengguna melakukan login. Pengguna akan langsung diarahkan ke

halaman data foto sesuai otoritas masing-masing.

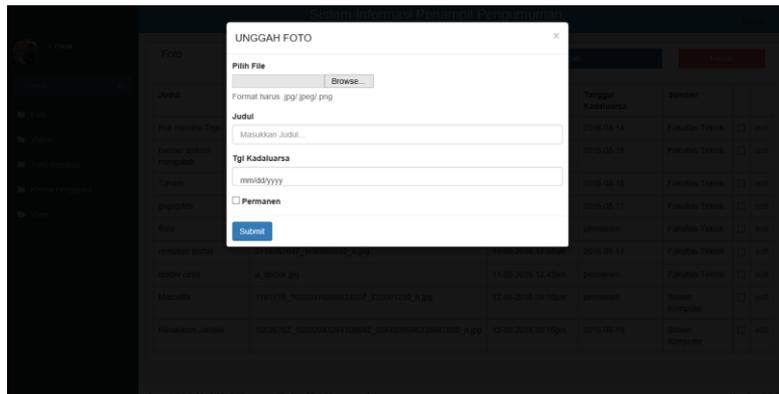


Gambar 5 Tampilan menu Foto otoritas fakultas

Perbedaan yang paling mencolok adalah di tampilan pengguna fakultas ditampilkan semua data foto dari jurusan maupun fakultas. Sedangkan ditampilkan foto jurusan hanya menampilkan data dari jurusan saja. Ketika tombol Unggah di halaman foto ditekan, akan memunculkan modal seperti Gambar 6.

Pengguna hanya diminta untuk memasukkan foto yang akan diunggah, judul

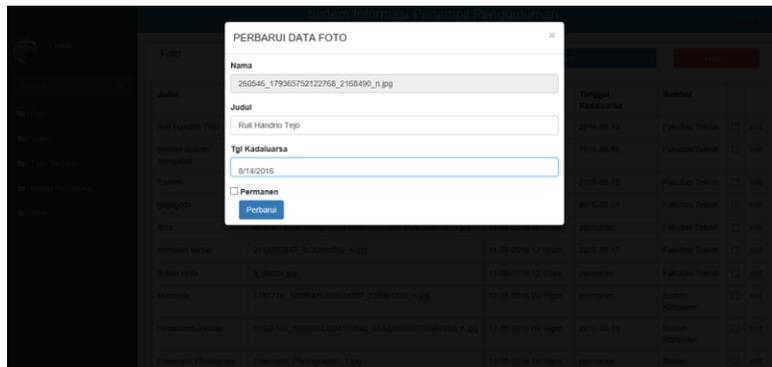
untuk foto, dan data tanggal kadaluarsa. Untuk tanggal ditambahkan dan sumber, akan ditambahkan secara otomatis oleh sistem. Lalu menekan tombol submit untuk meneruskan proses. Data foto yang telah diunggah, akan secara otomatis masuk kedalam tabel data foto. Berikut ini adalah potongan kode untuk mengunggah foto.



Gambar 6 Tampilan modal unggah foto

Pengguna juga diberikan fitur untuk mengubah data. Dengan menekan tombol edit, sistem akan memunculkan modal untuk

mengubah data. Gambar 7 adalah modal untuk memperbarui data.

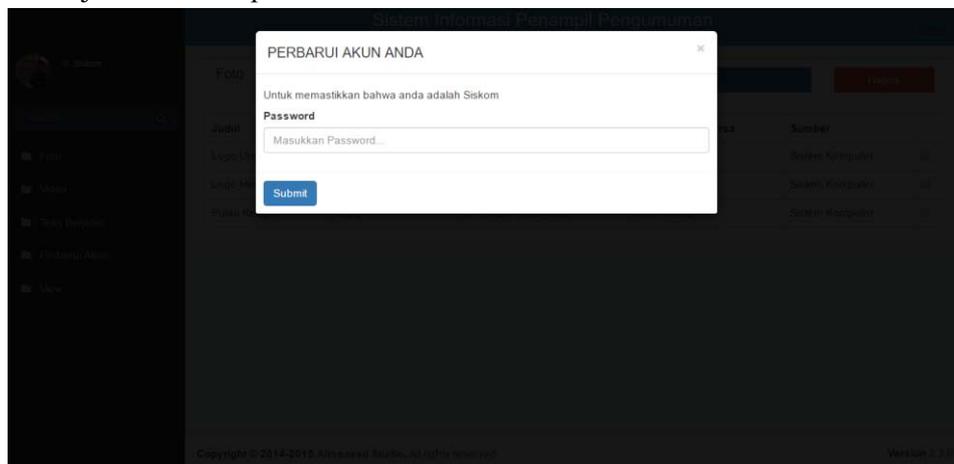


Gambar 7 Tampilan modal untuk perbarui data foto

Tampilan dan fitur yang sama dimiliki oleh halaman video dan teks berjalan.

Halaman perbarui akun hanya dimiliki oleh otoritas jurusan. Setiap kali masuk ke

halaman perbarui akun, pengguna akan diminta *password* lagi sesuai dengan akun yang digunakan seperti pada Gambar 8.



Gambar 8 Tampilan modal untuk masuk ke halaman perbarui akun

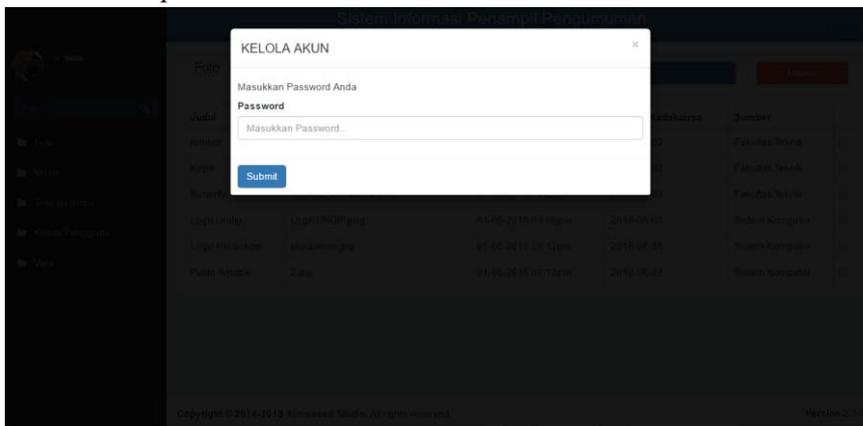


Gambar 9 Halaman edit akun

Setelah *password* dimasukkan dengan benar, maka akan diarahkan ke halaman edit akun seperti pada Gambar 9. Data yang tampil dihalaman edit akun bersifat bisa dirubah. Untuk merubah data, pengguna cukup memperbarui data yang ada lalu menekan tombol “perbarui” atau *enter*

dari *keyboard*.

Menu Kelola Pengguna hanya dimiliki oleh otoritas fakultas. Setiap kali masuk ke halaman kelola akun, pengguna akan dimintai *password* lagi sesuai dengan akun yang digunakan seperti pada Gambar 10.



Gambar 10 Tampilan modal kelola akun

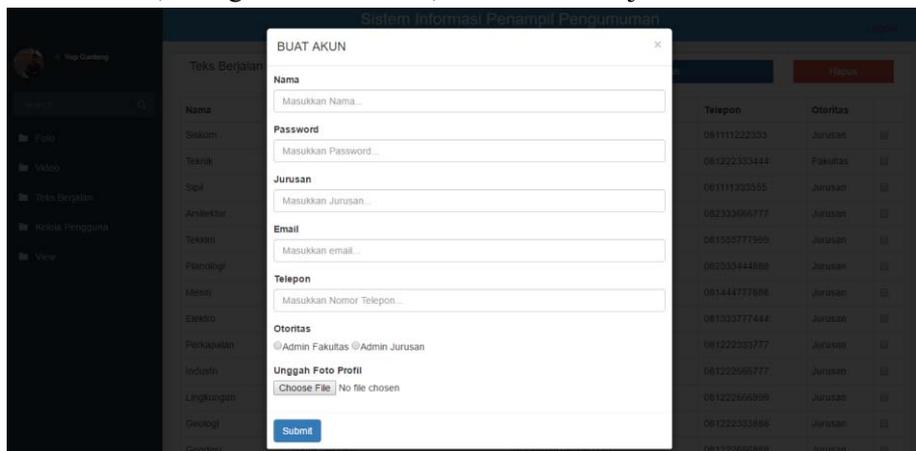


Gambar 10 Tampilan daftar akun

Setelah *password* dimasukkan dengan benar, maka akan diarahkan ke halaman daftar akun yang ditampilkan pada Gambar 10.

Menu kelola pengguna memiliki fitur untuk membuat akun, mengubah data akun,

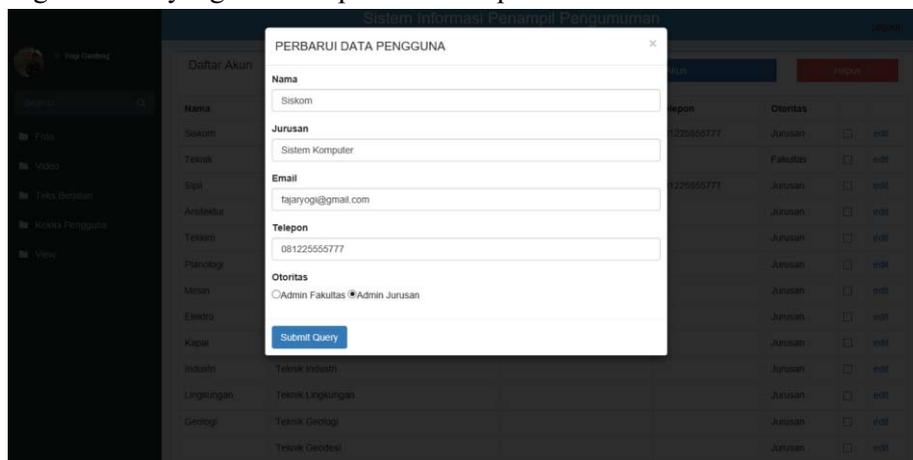
dan menghapus akun. Untuk membuat akun, pengguna perlu menekan tombol buat akun untuk memunculkan modal isian data akun yang akan dibuat. Gambar 11 adalah modal untuk membuat akun. Setelah data diisikan, maka dilanjutkan menekan tombol submit.



Gambar 11 Tampilan modal buat akun

Untuk memperbarui data akun, pengguna harus menekan tombol edit yang sejajar dengan akun yang akan diperbarui

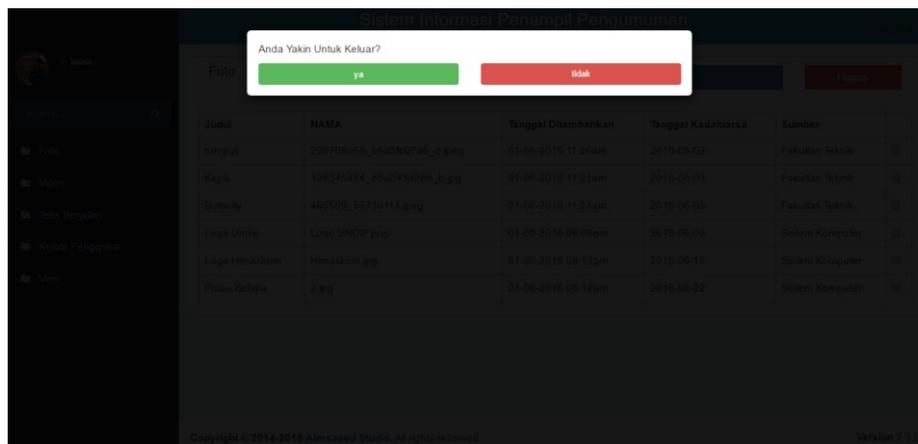
datanya untuk memunculkan modal perbarui data akun. Gambar 12 adalah modal untuk perbarui data akun.



Gambar 12 Modal untuk perbarui data pengguna

Untuk keluar dari SIPP, pengguna perlu menekan tombol keluar yang ada dipojok kanan atas setiap halaman. Setelah menekan

tombol keluar, maka pengguna akan ditanyai dalam bentuk modal.



Gambar 13 Tampilan antarmuka pengguna untuk keluar dari sistem



Gambar 14 Tampilan isi pengumuman pada penampil

Gambar diatas adalah tampilan isi pengumuman pada penampil. Gambar rumput adalah salah satu foto yang tampil. Foto akan tampil secara bergantian dengan bergeser. Gambar sebelah kanan adalah cuplikan video yang tampil. Video diputar secara otomatis ketika halaman dibuka dan berganti secara otomatis pula ketika video habis. Sedangkan tulisan dibawah adalah teks yang sedang berjalan, yang bersambung satu sama lain.

Pengujian

Dalam tahap pengujian untuk aplikasi SIPP ini, dilakukan dengan 2 (dua) pengujian. Pengujian yang pertama adalah pengujian *black box* sedangkan yang kedua

adalah pengujian peformansi. Pada tahap ini merupakan kelanjutan dari tahap implementasi.

Pengujian *black box* dilakukan dengan cara menguji aplikasi dengan memasukkan data ke dalam *form* yang telah disediakan. Tabel 5 berisi data pengujian fungsi login. Dari hasil pengujian didapatkan bahwa, tidak ada permasalahan fungsional yang ditemukan dan dapat disimpulkan bahwa semua tombol dalam SIPP telah berfungsi dengan baik dan semestinya.

Dalam pengujian performansi SIPP, digunakan perangkat lunak bernama 'Performance Analyser' yang tertanam pada Google Chrome dan Mozilla Firefox setelah

pemasangan. Dalam perangkat lunak tersebut terdapat beberapa kategori penilaian. Namun dalam pengujian performansi ini hanya diambil beberapa kategori saja, yaitu *Request to Host*, *Time to First Byte*, *DOM Content Loading*, *DOM Processing*, *Slowest Call*, *Average Call*.

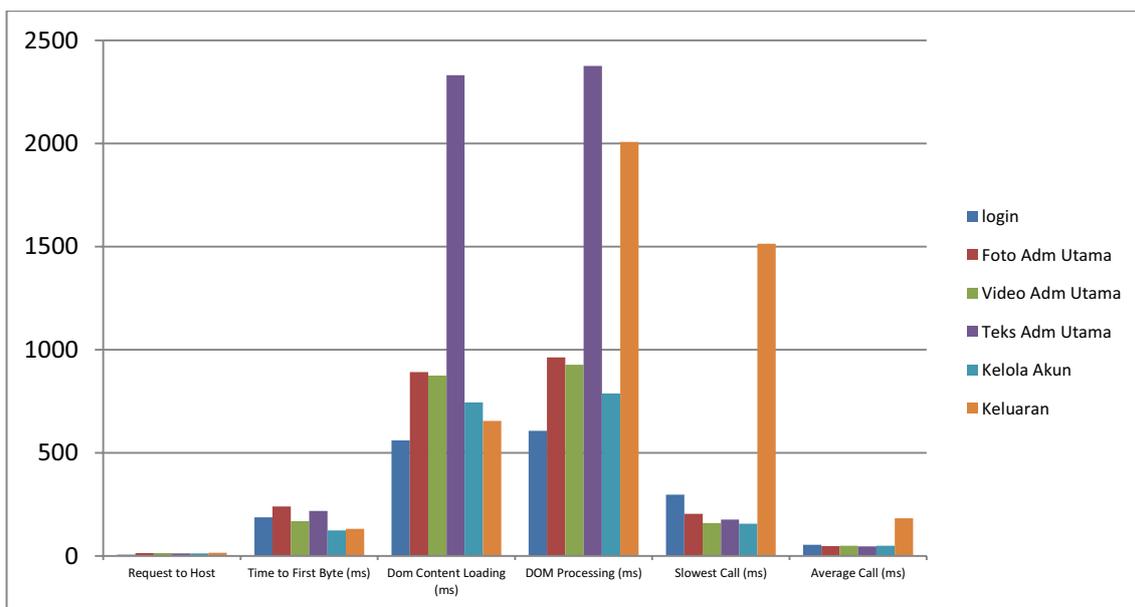
Request to Host adalah banyaknya permintaan yang diajukan ke server. *Time to First Byte (TTFB)* adalah waktu yang diperlukan oleh komputer sejak meminta data hingga diterimanya byte pertama berkas tersebut.

DOM (*Document Object Model*) adalah obyek model standar untuk HTML dan XML yang bersifat *platform independent*. Sebuah *web browser* tidak harus menggunakan DOM untuk menampilkan dokumen HTML. Namun DOM diperlukan oleh JavaScript yang akan mengubah tampilan sebuah website secara dinamis. Dengan kata lain, DOM adalah cara JavaScript melihat suatu halaman HTML.

DOM Content Loading berarti waktu yang diperlukan untuk mengambil kode-kode dari server. *DOM Processing* adalah waktu yang diperlukan untuk menerjemahkan kode-kode menjadi tampilan yang tampil dalam *web browser*. *Slowest call* adalah waktu terlama yang diperlukan untuk menghubungi server. Sementara *average call* adalah waktu rata-rata yang diperlukan untuk menghubungi server.

Metode yang digunakan dalam pengambilan data adalah dengan membuka halaman dari kondisi halaman kosong lalu dimasukkan url halaman yang dikehendaki. Percobaan dilakukan dengan 2 (dua) kondisi berbeda. Kondisi pertama adalah percobaan dilakukan di *web browser* Google Chrome yang notabene tidak menghilangkan cache dan relatif lebih berat untuk diakses. Kondisi kedua adalah percobaan dilakukan di *web browser* Mozilla Firefox incognito yang notabene relatif lebih ringan dan tidak ada cache dalam pengaksesan.

1. Tes Performansi Halaman Admin Utama

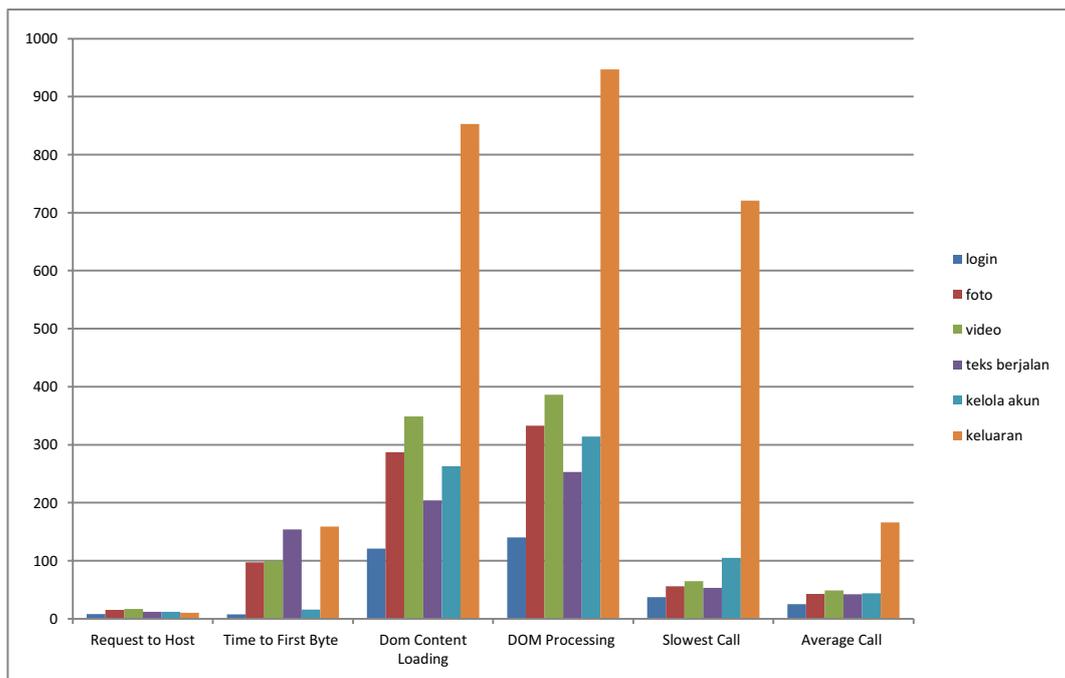


Gambar 15 Grafik Halaman Admin Utama Percobaan 1

Gambar 4.15 merupakan grafik data percobaan 1 performansi pada halaman admin utama yang menggunakan Google Chrome sebagai *web browser*. dari percobaan 1 diperoleh hasil bahwa nilai Request to Host terendah dimiliki oleh halaman login dengan 6 poin dan tertinggi dimiliki halaman keluaran dengan 14 poin. Time to First Byte terendah dimiliki oleh kelola pengguna dengan 124 ms sedangkan yang tertinggi ada halaman foto dengan 240 ms. Dom Content Loading terendah dimiliki halaman login dengan 561 ms sedangkan yang tertinggi halaman teks 2330 ms. DOM Processing terendah dimiliki oleh halaman login dengan 607 ms dan tertinggi dimiliki oleh halaman

teks berjalan dengan 2375 ms Average call terendah dimiliki halaman teks berjalan dengan 47 ms sedangkan yang tertinggi halaman keluaran dengan 182 ms.

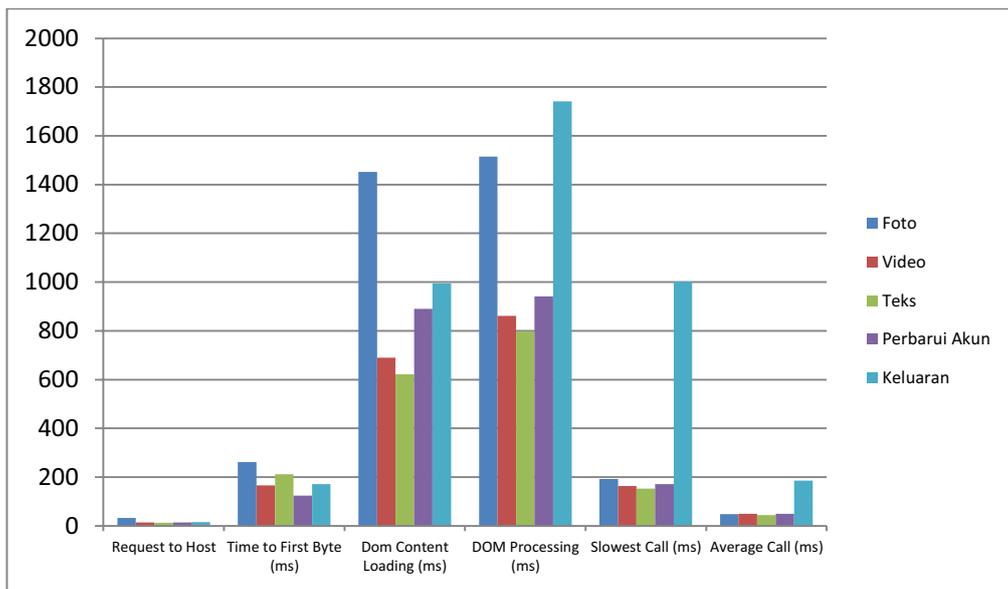
Dibandingkan dengan percobaan 1, percobaan 2 menghasilkan angka yang relatif lebih kecil. Diambil contoh Dom Processing tertinggi dimiliki oleh halaman keluaran dengan 947 ms dengan terendahnya dimiliki oleh halaman login dengan 140 ms. Atau data yang menghasilkan waktu terendah yaitu di Time to First Byte dengan tertinggi dimiliki oleh halaman keluaran dengan 15 ms dan terendah dimiliki oleh halaman login dengan 7,6 ms.



Gambar 16 grafik halaman admin utama percobaan 2

Hal tersebut juga terjadi pada pengujian halaman untuk sub-admin. Pada percobaan 1 halaman sub admin, angka tertinggi ada pada DOM Processing yang dimiliki halaman keluaran dengan 1742 ms dengan terendahnya dimiliki oleh ha-laman

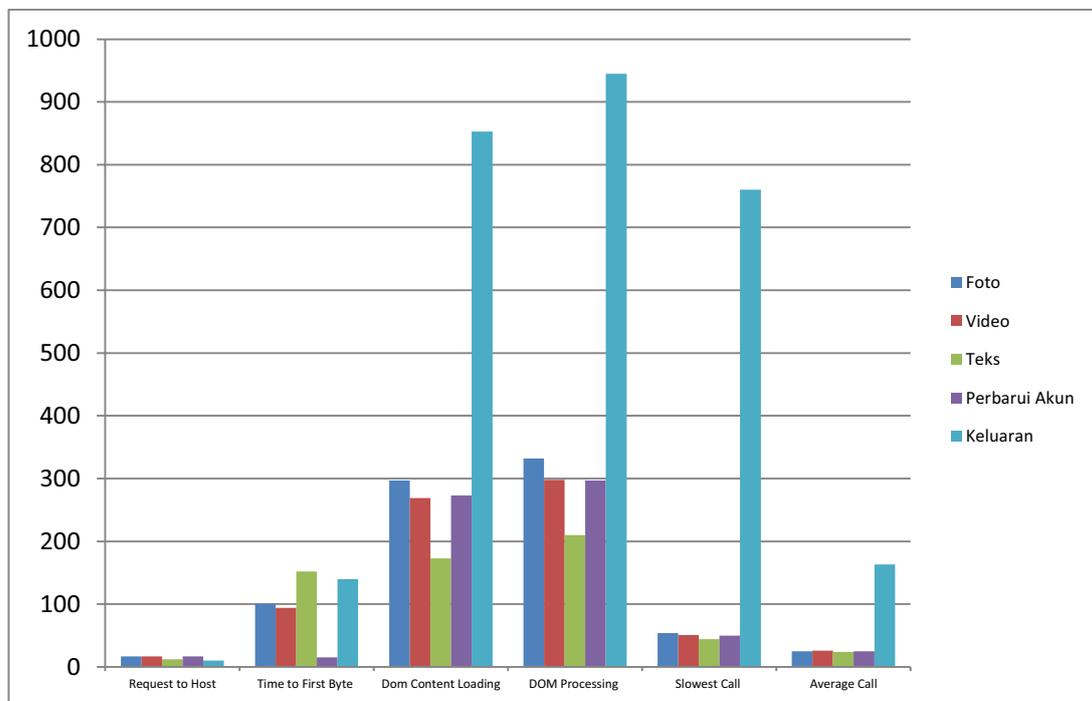
teks berjalan dengan nilai 797 ms. Angka terendah pada percobaan 1 ada pada Average Call dengan nilai tertinggi dimiliki oleh halaman keluaran dengan 186 ms dan nilai terendah dimiliki oleh halaman teks berjalan dengan 44 ms.



Gambar 17 Grafik halaman sub-admin percobaan 1

Berbeda dengan percobaan 1 halaman sub-admin, percobaan 2 relatif menghasilkan angka-angka yang lebih kecil. Angka tertinggi terdapat pada DOM Processing dimana nilai tertinggi dimiliki oleh halaman keluaran dengan 945 ms dan nilai

terendahnya dimiliki halaman teks berjalan dengan nilai 210 ms. Angka terendah terdapat pada Average Call dengan nilai tertinggi dimiliki oleh halaman keluaran dengan 163 ms dan terendah dimiliki oleh halaman teks berjalan dengan 44 ms.



Gambar 18 Grafik halaman sub-admin percobaan 2

Dari kedua percobaan diatas dapat disimpulkan bahwa faktor-faktor yang dapat mempengaruhi cepat lambatnya proses pengaksesan halaman oleh *web browser* antara lain:

1. *Web browser* yang digunakan. Pada kasus ini *web browser* yang digu-nakan adalah Google Chrome dan Mozilla Firefox incognito. Google Chrome memberikan hasil lebih buruk daripada Mozilla Firefox incognito karena memang diketahui bahwa Google Chrome menggu-nakan lebih banyak *resource* daripada Mozilla Firefox incognito.
2. Ada tidaknya *cache* dalam *web browser* juga turut mempengaruhi. Hal ini dapat dibuktikan bahwa menggunakan Mozilla Firefox mode incognito memberikan hasil lebih baik daripada Google Chrome karena mode ini tidak menyimpan *cache* ketika penggunaan.
3. Diperkirakan aktivitas komputer saat pengujian juga turut mempengaruhi. Hal ini diindikasikan demikian karena pada beberapa hasil menunjukkan anomali. Pada percobaan 1 halaman teks berjalan admin utama memiliki DOM Content Loading dan DOM Processing tertinggi, padahal percobaan yang lain menunjukkan bahwa halaman keluaran memiliki hasil tertinggi. Hal ini terjadi dimungkinkan karena aktivitas komputer sedang meningkat sehingga ikut mempengaruhi hasil percobaan.

V. PENUTUP

Dari hasil pembuatan dan pengujian SIPP dapat disimpulkan bahwa dibuatnya SIPP akan membantu penyebaran informasi dilingkungan Fakultas Teknik Univer-sitas Diponegoro dengan lebih menarik dan terstruktur hingga ke tingkat jurusan, penggunaan sistem dan penyalahgunaanya dapat dipantau oleh otoritas yang lebih

berwenang sehingga informasi yang tampil menjadi lebih berkualitas, pengujian SIPP dilakukan menggunakan metode *black box* dan menunjukan bahwa fungsi-fungsi yang terdapat pada aplikasi dapat berfungsi dengan baik dan sesuai dengan fungsionalitasnya masing-masing serta dapat berjalan dengan baik, dari pengujian performansi diketahui bahwa dengan kondisi opotimal, SIPP bisa diakses dengan kecepatan ter-lama 947 ms yaitu halaman keluaran, dan tercepat dengan 44 ms di halaman teks berjalan, dengan kata lain bisa dinyatakan cukup cepat untuk diakses, *web browser* yang digunakan, ada tidaknya *cache* dalam *web browser* ikut mempengaruhi hasil uji performansi.

Saran yang dapat diberikan untuk pengembangan SIPP yang lebih baik dikemudian hari adalah dapat dipertimbangkan untuk menggunakan *framework* dalam pembuatan sistem untuk menyingkat waktu dan tenaga dalam pengerjaan dan pengkodean lebih terstruktur sesuai standar sehingga dapat dilanjutkan oleh pengembang lain, perlu dibuatkan sistem keamanan yang lebih handal jika data yang dikelolakan mulai dikembangkan, perlu dilakukan peng-ambilan data yang lebih teliti, sehingga menghasilkan data yang valid.

DAFTAR PUSTAKA

- [1] Riko, *Penampil Informasi Menggunakan Mikrokontroler 328 Berbasis Web*, 2014.
- [2] Panuntun, Rizal., *Perancangan Papan Informasi Berbasis Web Menggunakan Raspberry-Pi*, 2014.
- [3] Dwiartara, Loka., *Menyelam & Menaklukan Samudra PHP*, 2010.
- [4] Kustiyahningsih, Yeni. Anamisa, Devie Rosa., *Pemrograman Basis Data Berbasis Web Menggunakan PHP &*

- MySQL*, Penerbit Graha Ilmu, DIY, 2011.
- [5] Kadir, A, *Pemrograman Web Mencakup: HTML, CSS, JavaScript dan PHP*. Yogyakarta: ANDI, 2003.
- [6] Ariona, R, *Belajar HTML & CSS: Tutorial Fundamental dalam Mempelajari HTML & CSS*. 2013.
- [7] Khannedy, E.K, *Tutorial Javascript*, 2007.
- [8] Jogiyanto, *Analisis dan Perancangan Sistem*, Andi Offset, Yogyakarta, 2001
- [9] Khanna, E, Bhasin, H, dan Sudha, "Black Box Testing based on Requirement Analysis and Design Specifications," vol. 87, no. 1, pp. 36–40, 2014.
- [10] Rouf, A, "Pengujian perangkat lunak dengan menggunakan metode white box dan black box," pp. 1–7, 2012.