

IMPLEMENTASI SISTEM INFORMASI RUMAH SAKIT UNTUK SUBSISTEM LABORATORIUM DENGAN FRAMEWORK PRADO

Eko Handoyo, Agung Budi Prasetyo, Dwi Mardiatmo Nur Hidayat *)

Abstract

Nowadays, the need of an information for some hospital is getting bigger. This Hospital Information System application is made to be based on web by using Prado framework based on PHP programming language and MySQL as its databases. This research include application scheme using object oriented approach, database scheme, and also examination from application as a whole. In its making, this application adapted with the requirement of a hospital in general such as facility for requesting of examination disease at laboratory, queue information of test request, record-keeping of test result of a patient at medical laboratory, pathology, and bacteriology, blood request for patient at blood bank of hospital, and also result information of laboratory test of a patient. The result of the application examination show this information system can work if input condition have been full filed. Process which desirable for example login and user registration process, data addition, data change, data abolition, and the data seeking at system. This application earn to give queue information of test request, service for the blood requesting, and also result information of laboratory test of a patient.

Keywords : Hospital Information System, Prado Framework

Pendahuluan

Perkembangan teknologi informasi telah mengubah manusia dalam menyelesaikan semua pekerjaannya, tidak hanya dalam pekerjaannya saja tetapi dalam segala aspek kehidupan manusia, seperti pada saat pencarian informasi. Jika dahulu manusia mencari informasi sebatas pada buku, media cetak, maupun secara lisan, sekarang lebih banyak mencari informasi tersebut melalui internet. Secara tidak langsung dapat dikatakan semua serba terkomputerisasi.

Rumah sakit sebagai salah satu institusi pelayanan umum membutuhkan keberadaan suatu sistem informasi yang akurat dan andal, serta cukup memadai untuk meningkatkan pelayanannya kepada para pasien serta lingkungan yang terkait lainnya. Dengan lingkup pelayanan yang begitu luas, tentunya banyak sekali permasalahan kompleks yang terjadi dalam proses pelayanan di rumah sakit. Banyaknya variabel di rumah sakit turut menentukan kecepatan arus informasi yang dibutuhkan oleh pengguna dan lingkungan rumah sakit.

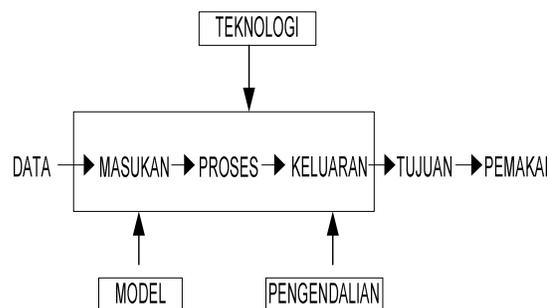
Pengelolaan data di rumah sakit merupakan salah satu komponen yang penting dalam mewujudkan suatu sistem informasi di rumah sakit. Pengelolaan data secara manual, mempunyai banyak kelemahan, selain membutuhkan waktu yang lama, keakuratannya juga kurang dapat diterima, karena kemungkinan kesalahan sangat besar. Dengan dukungan teknologi informasi yang ada sekarang ini, pekerjaan pengelolaan data dengan cara manual dapat digantikan dengan suatu sistem informasi dengan menggunakan komputer. Selain lebih cepat dan mudah, pengelolaan data juga menjadi lebih akurat.

Penelitian ini membahas perancangan dan pengembangan sistem informasi rumah sakit untuk subsistem laboratorium yang berbasis web. Aplikasi ini diguna-

kan untuk membantu bagian laboratorium di rumah sakit untuk menyediakan pelayanan dan informasi kesehatan bagi pasien sehingga dapat mendukung kelancaran kegiatan manajemen di suatu rumah sakit sehari-hari.

Merancang dan membangun sebuah aplikasi sistem informasi rumah sakit untuk manajemen laboratorium berbasis web dengan menggunakan *framework* Prado yang menggunakan bahasa pemrograman PHP dengan menggunakan MySQL sebagai basis datanya. Sistem informasi ini diharapkan dapat bermanfaat untuk membantu dalam mengolah data di laboratorium rumah sakit.

Sistem informasi adalah sekumpulan komponen pembentuk sistem yang mempunyai keterkaitan antara satu komponen dengan komponen lainnya yang bertujuan menghasilkan suatu informasi dalam suatu bidang tertentu. Dalam sistem informasi diperlukan klasifikasi alur informasi, hal ini disebabkan keanekaragaman kebutuhan akan suatu informasi oleh pengguna informasi. Kriteria dari sistem informasi antara lain, fleksibel, efektif dan efisien^[1]. Secara garis besar komponen yang terkait dengan suatu sistem informasi dapat dilihat pada Gambar 1 berikut.



Gambar 1 Komponen sistem informasi

*) Staf Pengajar Teknik Elektro Fakultas Teknik
Universitas Diponegoro

Sistem Informasi Rumah Sakit (SIRS) adalah suatu tatanan yang berurusan dengan pengumpulan data, pengelolaan data, penyajian informasi, analisis dan penyimpulan informasi serta penyampaian informasi yang dibutuhkan untuk kegiatan rumah sakit^[1].

Sebuah sistem informasi rumah sakit idealnya mencakup integrasi fungsi-fungsi klinikal (medis), keuangan, serta manajemen yang nantinya merupakan subsistem dari sebuah sistem informasi rumah sakit. Subsistem ini merupakan unsur dari sistem informasi rumah sakit yang tugasnya menyiapkan informasi berdasarkan fungsi-fungsi yang ada untuk menyederhanakan pelayanan pada suatu rumah sakit.

Fungsi utama dari rumah sakit yang pada umumnya adalah pelayanan kesehatan, serta pasien sebagai objek dari fungsi utama rumah sakit, dukungan operasional berupa tenaga kerja, keuangan, sarana dan prasarana, serta sistem manajemen yang dibutuhkan untuk mengelola suatu rumah sakit. Maka berdasarkan pertimbangan tersebut suatu sistem informasi

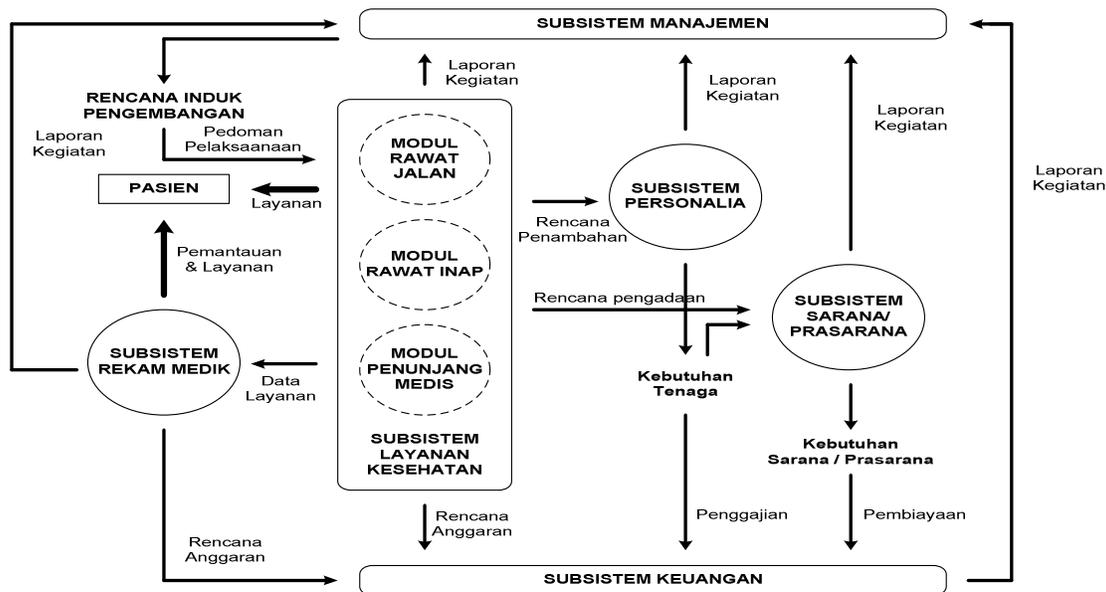
rumah sakit terdiri dari beberapa subsistem sebagai berikut :

1. Subsistem Layanan Kesehatan. Subsistem Rekam Medis.
2. Subsistem Personalialia.
3. Subsistem Keuangan.
4. Subsistem Sarana/Prasarana.
5. Subsistem Manajemen Rumah Sakit.

Subsistem tersebut diatas kemudian dijabarkan lagi ke dalam modul-modul yang sifatnya lebih spesifik. Subsistem Layanan Kesehatan dapat dijabarkan lebih lanjut menjadi :

1. Modul Rawat Jalan.
2. Modul Rawat Inap.
3. Modul Layanan Penunjang Medis.

Skema rancang bangun SIRS secara global ini dapat dilihat pada Gambar 2.2. Pada gambar tersebut diberikan contoh hubungan antara satu subsistem dengan subsistem lainnya. Rancangan global SIRS berisi penjabaran SIRS menjadi subsistem, modul, submodul dan aplikasi.



Gambar 2 Rancangan global sistem informasi rumah sakit

- 1.
- 1.1
- 1.2

Pelayanan Laboratorium merupakan salah satu pelayanan penunjang di lingkungan rumah sakit dalam menjalankan fungsinya untuk memberikan pelayanan kepada masyarakat.

Divisi Laboratorium bertanggung jawab terhadap pemberian jasa pelayanan yang berhubungan dengan pemeriksaan penunjang medis laboratorium baik pemeriksaan medis, patologi, dan mikrobiologi bagi seorang pasien. Selanjutnya saat ini Divisi Laboratorium juga diberikan tanggung jawab sebagai pengelola Bank Darah.

Pada masa yang akan datang beberapa konsep baru telah disepakati untuk digunakan di lingkungan Divisi Laboratorium. Hal pertama yang diperkenalkan dan akan dilaksanakan adalah order manajemen yaitu pemanfaatan pelayanan permintaan dan penyampaian hasil pemeriksaan dengan memanfaatkan fasilitas komputer secara *online*. Fasilitas ini dapat dimanfaatkan oleh semua unit pengguna. Status atau proses permintaan layanan termasuk hasil pemeriksaan dapat dipantau / dilihat langsung melalui fasilitas komputer.

Metodologi penelitian

Di dalam pengembangan perangkat lunak, suatu *framework* digambarkan sebagai suatu struktur pendukung dimana perancangan perangkat lunak yang lain dapat terorganisir dan dikembangkan^[2]. Suatu *framework* dapat meliputi program pendukung, kumpulan kode-kode program (*libraries*), suatu bahasa *scripting*, atau perangkat lunak lain untuk membantu mengembangkan dan menggabungkan komponen - komponen yang berbeda menjadi satu dari suatu perancangan perangkat lunak.

Prado adalah sebuah *framework* pemrograman berbasis komponen dan *event-driven* untuk pengembangan aplikasi web pada PHP 5. PRADO merupakan singkatan dari PHP *Rapid Application Development Object-oriented Framework* ini dibuat oleh Qiang Xue dan telah menjadi pemenang dalam Zend PHP 5 *Coding Contest*.

Teknik yang digunakan *framework* Prado sangatlah berbeda. Pembangunan aplikasi web menggunakan Prado melibatkan banyak komponen (yang telah dibuat oleh pengembangnya), *men-setting* properti, dan memberikan tugas pada komponen berupa *event*. Jika seorang programmer mempunyai pengalaman dengan pemrograman desktop menggunakan Visual Basic atau Delphi, maka pemrograman web dengan menggunakan *framework* Prado sangatlah serupa.

Sebuah komponen Prado adalah kombinasi file spesifikasi (ditulis dengan bahasa XML), *template* HTML, dan *page class*. Komponen-komponen Prado digabungkan untuk membangun komponen yang lebih besar atau halaman web yang utuh. Prado membutuhkan PHP 5 dengan Simple XML dan Simple PHP Library (SPL). Untuk web server dapat digunakan Apache HTTP Server maupun Windows IIS dan mendukung Sistem Operasi Windows Server 2003, Windows XP, Windows 2000, Mac OS X, FreeBSD, RedHat Linux, Fedora Linux, dan Gentoo Linux.

Konsep Prado yang *component-based* dan *event-driven* memberikan banyak keuntungan bagi programmer web. Berikut keuntungan dengan menggunakan Prado:

1. *Reusability*, komponen-komponen dalam Prado dapat digunakan ulang.
2. *Ease of use*, komponen-komponen dalam Prado sangat mudah digunakan. Komponen juga dapat dibuat sendiri dengan menurunkan *class* yang sudah ada sesuai dengan kebutuhan.
3. *Robustness*, Prado membebaskan pengembang program (*developer*) dari kejenuhan dalam menulis kode-kode. Semua kode ditulis dalam format objek, method, dan properti. Tidak seperti pemrograman PHP yang biasa digunakan.
4. *Performance*, Prado menggunakan teknik *cache* untuk menjamin *performance* aplikasi. Dengan

adanya *cache* ini, ia tidak perlu mem-*parser* ulang kode XML yang dibuat.

5. *Team Integration*, Prado memisahkan *business logic* dan *presentation logic*. Yang dimaksudkan adalah pembuatan layout tampilan (*template*) dengan kode program (*class*). Pembuatan keduanya dilakukan pada file yang terpisah. Dengan demikian, aplikasi berbasis Prado dapat dilakukan dalam sebuah tim dengan personal yang berbeda.

Di dalam *framework* Prado terdapat susunan file-file standar sebagai berikut :

2. File .htaccess

File ini berfungsi untuk mengatur hak akses suatu isi folder, dalam hal melihat isi folder dari suatu browser. File ini hanya berisi kode :

- deny from all
- File application.spec
File ini berfungsi untuk menspesifikasi aplikasi, antara lain konfigurasi atribut *default*, menentukan lokasi kode utama *framework* Prado, namespace yang digunakan ,dan lain-lain.
- File *page display* (index.php)
File ini merupakan file yang akan diakses oleh user, dalam file ini berisi letak file utama Prado dan letak file aplikasi sistem informasi rumah sakit.
- File *page template* (.tpl)
File inilah yang bertanggung jawab terhadap tampilan yang dilihat user.
- File *page class* (.php)
File ini lebih banyak berhubungan dengan proses bisnis atau biasa disebut dengan *business logic*.

Koneksi Prado Dengan Basisdata

Untuk melakukan koneksi ke database, Prado memanfaatkan *database abstract layer*, ADOdb. ADOdb adalah class yang ditulis menggunakan bahasa PHP yang berfungsi sebagai *data tier*, dan akan membantu mengatasi perbedaan antara penggunaan database. Cukup dengan menuliskan sebuah kode, maka koneksi dapat dilakukan ke berbagai macam database seperti MySQL, SQLite, SQL Server, Oracle, DB2, Interbase, PostgreSQL, dan sebagainya.

Aplikasi yang dibangun bernama Sistem Informasi Rumah Sakit untuk Subsistem Laboratorium. Subsistem Laboratorium merupakan salah satu bagian dari Sistem Informasi Rumah Sakit secara keseluruhan, tujuan dari pembagian ini adalah untuk mempermudah pemahaman Sistem Informasi Rumah Sakit yang begitu kompleks, sehingga dengan adanya pembagian ini diharapkan penanganan pada subsistem laboratorium dapat lebih detail dan efektif disamping itu juga menghemat waktu dibanding menangani sistem secara keseluruhan.

Divisi Laboratorium bertanggung jawab terhadap pemberian jasa pelayanan yang berhubungan dengan pemeriksaan penunjang medis laboratorium baik pemeriksaan medis, patologi, mikrobiologi, dan sebagai pengelola bank darah bagi seorang pasien.

Aplikasi Sistem Informasi Rumah Sakit ini pada sisi server merupakan sebuah aplikasi berbasis web (*web-based application*) yang berfungsi untuk menerima masukan dari aplikasi sisi klien dan juga berfungsi untuk mengatur basis data. Perancangan aplikasi sisi server ini menggunakan analisis pemodelan berorientasi objek.

Use Case.

Uraian berikut ini adalah use case dari Sistem Informasi Rumah Sakit pada Subsystem Laboratorium. Kata yang bercetak tebal menggambarkan calon objek sedangkan kata yang bercetak miring menggambarkan operasi.

Persiapan Sistem

Sistem ini mempunyai empat jenis pengguna, yaitu administrator sistem, dokter, staf lab, dan user. Setelah sistem ter-*install*, administrator sistem yang mempunyai hak akses tertinggi dalam sistem perlu *menyediakan* informasi-informasi yang nantinya akan dimasukkan ke dalam sistem. Administrator sistem juga memiliki kewenangan untuk *mengelola* informasi-informasi tersebut. ID pengguna dan kata kunci yang digunakan oleh administrator sistem untuk memasuki sistem dapat *diubah* untuk menjamin keamanan akses. Sebelum mengakses sistem, pengguna harus melakukan register terlebih dahulu yang kemudian *diaktifasi* oleh administrator, setelah itu pengguna dapat mengakses ke dalam sistem informasi rumah sakit sesuai dengan role-nya.

User

Setelah melakukan registrasi, user akan *mendapatkan* ID pengguna dan kata kunci. Administrator harus *mengaktifkan* user yang telah melakukan registrasi agar user tersebut dapat *mengakses* sistem. User yang sudah *diaktifkan* dapat *mengakses* sistem informasi rumah sakit dan dapat *ikut serta* dalam aktifitas di dalam sistem. Kata kunci yang digunakan oleh user juga dapat *diubah* untuk menjamin keamanan. User hanya dapat mencari dan melihat informasi pasien serta melihat informasi antrian order lab.

Staf Lab

Sama seperti pengguna lainnya, Staf Lab setelah *teraktifasi* kemudian *mendapatkan* id dan kata kunci yang digunakan untuk masuk ke dalam sistem.. Setelah Staf Lab *melakukan login* maka Staf Lab dapat *melakukan* berbagai macam *aktivitas* diantaranya melihat informasi antrian order pasien, *mengisikan* dan *mengedit* permintaan test laboratorium pasien, *mengisikan* hasil test laboratorium medis, *mengedit* parameter test, *mengisikan* hasil test laboratorium patologi, *mengisikan* dan *mengedit* permintaan pada bank darah, serta mencari dan melihat

informasi hasil test laboratorium seorang pasien. Staf Lab juga dapat *mengubah* kata kunci milik mereka.

Dokter

Dokter juga *mendapatkan* id dan kata kunci sebagai hak akses ke dalam sistem setelah *melakukan* registrasi dan *teraktifasi*. Setelah dokter *melakukan login* maka dokter dapat *melakukan aktivitas* diantaranya melihat data pasien, melihat informasi antrian order lab dan melihat informasi hasil test laboratorium seorang pasien. Dokter juga dapat *mengubah* kata kunci milik mereka.

Administrator

Administrator *memiliki* peranan yang paling besar di dalam aktivitas sistem ini. Administrator *memiliki* hak akses yang paling lengkap dibandingkan user-user lainnya. Administrator *mempunyai* kewajiban dalam *mengatur* administrasi user diantaranya *mengaktifasi* pengguna yang sudah *melakukan* registrasi ke dalam sistem. Administrator juga berhak *menonaktifkan* seorang user. Untuk keperluan manajemen informasi Administrator dapat *menambahkan* menu baru atau *mengedit* menu dan informasi yang sudah ada sebelumnya. Selain itu administrator juga dapat *mengubah* kata kunci miliknya.

Identifikasi Objek.

Setelah proses dilanjutkan dengan indentifikasi calon objek kemudian diikuti oleh hasil seleksi calon objek. Berikut ini merupakan daftar spesifikasi atribut dari masing-masing objek atau kelas.

User = ID + password + role + status

Dokter = ID + password + role + status

Staf Lab = ID + password + role + status

Administrator = ID + password

Antrian order = tanggal order + nama pasien + divisi laboratorium + kode test + tanggal test + waktu test + nama dokter + nama petugas

Permintaan test = tanggal order + nama pasien + divisi laboratorium + kode test + tanggal test + waktu test + nama dokter + nama petugas + diagnosa + catatan

Hasil Test = nama pasien + tanggal + waktu + parameter test

Setelah pengerjaan use case dan indentifikasi objek, tahapan selanjutnya melakukan pemodelan kelas-tanggungjawab-kolaborator (CRC). Metode ini disusun berdasarkan indentifikasi objek atau kelas. Objek merupakan instansiasi dari kelas, untuk mempermudah pada saat pembuatan program, tiap kelas diberi nama yang lebih sederhana.

Setelah melakukan pemodelan kelas-tanggungjawab-kolaborator maka dapat dibuat model hubungan antar objek dengan mengkaji kartu indeks model CRC, tanggung jawab dan kolaborator. Dari pembuatan kartu index CRC selanjutnya dapat dibuat diagram kelas. Diagram kelas memberikan gambaran tentang kelas-kelas apa saja yang perlu dibuat untuk membangun aplikasi, lengkap dengan atribut dan operasinya.

Pemodelan objek yang dilakukan antara lain melalui CRC dan model hubungan antar objek berfungsi untuk mempresentasikan elemen statis dari model analisis berorientasi objek. Langkah selanjutnya adalah memodelkan elemen dinamis dari model analisis, hal ini dilakukan dengan memodelkan tingkah laku objek, antara lain dengan membangun diagram runtun (*sequence diagram*) dan *statechart diagram*. *Sequence diagram* merupakan diagram interaksi yang menekankan urutan waktu dari sebuah pesan. Diagram ini menunjukkan kumpulan objek dan pesan yang dibawa maupun diterima objek tersebut. *Statechart diagram* merupakan diagram yang menggambarkan sebuah sistem secara dinamis. Model tingkah laku objek menunjukkan bagaimana sistem akan merespon kejadian atau stimulus eksternal.

Hasil dan pembahasan

Pengguna SIRS Subsistem Laboratorium

Pengguna dari sistem informasi rumah sakit subsistem laboratorium terdiri dari 4 jenis, yaitu :

1. Administrator, merupakan pemegang hak akses paling tinggi dalam sistem.
2. Dokter
3. Staf Lab
4. User

Antarmuka sistem

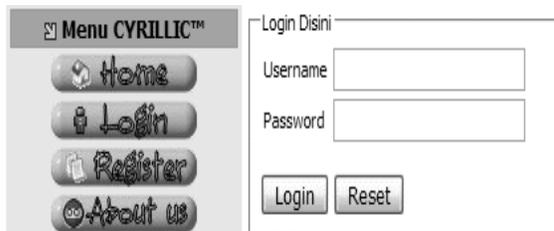
Aplikasi SIRS mempunyai halaman utama yang berisikan link ke form login yang digunakan untuk autentifikasi pengguna agar bisa mengakses sistem, link ke form registrasi yang digunakan untuk pendaftaran pengguna, serta link ke halaman about us yang memuat informasi pembuat aplikasi SIRS ini.



Gambar 3 Tampilan halaman muka

Antarmuka form login

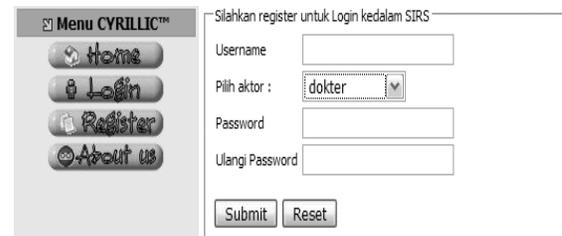
Antamuka Login memuat form login yang dapat digunakan untuk tempat autentifikasi bagi pengguna. Setelah memasukkan username dan password yang benar, maka pengguna berhak untuk masuk dan mengakses SIRS.



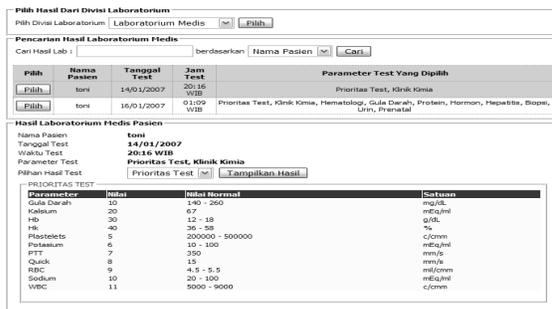
Gambar 4 Tampilan halaman login

Antarmuka form registrasi

Antamuka Registrasi memuat form registrasi yang dapat digunakan untuk tempat pendaftaran bagi seorang pengguna. Setelah melakukan registrasi maka seorang pengguna berhak untuk mengakses SIRS..



Gambar 5 Tampilan halaman registrasi



Gambar 12 Tampilan halaman informasi hasil test laboratorium

Pengujian aplikasi

Aplikasi dinyatakan berhasil melewati pengujian apabila dalam percobaan-percobaan berikut tidak aplikasi tidak menjadi *error* atau melaksanakan perintah dengan salah. Pengujian aplikasi baik pada sisi klien maupun server menggunakan metode kotak hitam (*black box*).

Dalam metode pengujian *black box*, aplikasi diberikan berbagai macam kondisi masukan, kemudian keluaran yang dihasilkan sistem dibandingkan dengan keluaran yang diharapkan.

Prosedur pengujian aplikasi dilakukan seperti pada pengujian aplikasi sisi klien. Aplikasi dihadapkan pada berbagai kondisi yang memungkinkan untuk menimbulkan kesalahan, seperti misalnya pengisian *form* yang salah maupun tidak lengkap. Aplikasi harus mampu mengenali kondisi-kondisi tersebut dan menyiapkan fasilitas untuk mengatasinya, sehingga tidak menjadi *error* atau menjalankan fungsi yang salah.

Pengujian dilakukan antara lain pada form login. Adapun pengujiannya adalah sebagai berikut.

Tabel 1. Hasil pengujian black box proses login.

Input		Output	
Username	Password	Output diharapkan	Output Sistem
(-)	(-)	Pesan: Masukkan Username. Masukkan Password.	Pesan: Masukkan Username. Masukkan Password.
(-)	√	Pesan: Masukkan Username.	Pesan: Masukkan Username.
√	(-)	Pesan: Masukkan Password.	Pesan: Masukkan Password.
√(*)	√	Pesan: Login Gagal! Username atau Password anda salah.	Pesan: Login Gagal! Username atau Password anda salah.
√	√(**)	Pesan: Login Gagal! Username atau Password anda salah.	Pesan: Login Gagal! Username atau Password anda salah.
√	√	Tampilan Halaman index Pesan: Selamat datang	Tampilan Halaman index Pesan: Selamat datang

Keterangan :

- √ : Data diisi.
- √(*) : Username yang diinputkan tidak ada dalam basisdata.
- √(**) : Password yang diinputkan salah.
- (-) : Data kosong.

Pada pengujian proses login, ada tiga kondisi yang harus dipenuhi pada saat melakukan login. Ketiga proses itu yaitu yang pertama adalah melengkapi semua *field* login. Kedua adalah bahwa *username* terdapat dalam basis data. Ketiga adalah jika *password* yang diinputkan cocok dengan password yang tersimpan dalam basis data. Proses query hanya akan dijalankan apabila ketiga syarat tersebut dipenuhi. Apabila syarat tidak terpenuhi maka proses login tidak akan berhasil dan menampilkan pesan peringatan pada pengguna.

Hasil pengujian menggunakan metode diatas, aplikasi tidak menjadi *error* atau menampilkan keluaran yang salah. Semua kondisi penggunaan telah diujikan dan berhasil, dengan demikian aplikasi telah berhasil melewati pengujian.

Kesimpulan

Dari Penelitian pembuatan aplikasi Sistem Informasi Rumah Sakit Subsistem Laboratorium ini dapat diambil beberapa kesimpulan antara lain :

1. Berdasarkan hasil pengujian dengan metode kotak hitam (*black box*), aplikasi berbasis web yang dibangun yaitu SIRS Subsistem Laboratorium telah sesuai dengan yang diharapkan dan dapat berfungsi dengan baik.
2. Aplikasi SIRS Subsistem Laboratorium ini berfungsi sebagai pendukung dalam kegiatan pelayanan laboratorium di rumah sakit.
3. Dengan konsep *framework* Prado yang berbasis-komponen dan event driven, Prado memberikan keuntungan yang banyak dalam pengembangan aplikasi berbasis web.
4. Dari hasil pengujian dapat disimpulkan bahwa aplikasi SIRS Subsistem Laboratorium ini telah sesuai dengan fungsi-fungsi yang dimiliki oleh laboratorium di suatu rumah sakit pada umumnya.

Daftar Pustaka

1. Sabarguna, MARS, Dr. dr. H. Boy S., *Sistem Informasi Rumah Sakit*, Penerbit Konsorsium Rumah Sakit Jateng - DIY, 2005
2. Siswoutomo, Wiwit, *PHP Enterprise Kiat Jitu Membangun Web Skala Besar*, Penerbit PT. Elex Media Komputindo Jakarta, 2005
3. Kadir, A., *Konsep dan Tuntunan Praktis Basis Data*, Penerbit Andi Yogyakarta, 1999
4. Fowler, Martin, *UML Distilled Edisi 3 Panduan Singkat Tentang Bahasa Pemodelan Objek Standar*, Penerbit Andi Yogyakarta, 2005
5. Suhendar, A, S.Si, Hariman Gunadi S.Si., MT., *Visual Modeling Menggunakan UML dan Rational Rose*, Penerbit Informatika Bandung, 2002
6. Prasetyo, D. D., *Kolaborasi PHP dan MySQL untuk Membuat Web Database yang Interaktif*, PT. Elex Media Komputindo Jakarta, 2003
7. Siswoutomo, Wiwit, *Membangun Web Service Open Source Menggunakan PHP*, Penerbit PT. Elex Media Komputindo Jakarta, 2005
8. Azis, M.Kom, Ir. M. Farid, *Object Oriented Programming Dengan PHP5*, Penerbit PT. Elex Media Komputindo Jakarta, 2005
9. Siswoutomo, Wiwit, *PHP Undercover Mengungkap Rahasia Pemrograman PHP*, Penerbit PT. Elex Media Komputindo Jakarta, 2005
10. Siswoutomo, Wiwit, *Membuat Aplikasi Database Berbasis Web*, Penerbit PT. Elex Media Komputindo Jakarta, 2005
11. Azis, M.Kom, Ir. M. Farid, *Pemrograman PHP4 Bagi Web Programmer*, Penerbit PT. Elex Media Komputindo Jakarta, 2001