RANCANG BANGUN SISTEM ENTERPRISE RUMAH SAKIT DENGAN MENGGUNAKAN DESIGN PATTERN MODEL VIEW CONTROLLER (MVC) (Studi Kasus Poliklinik Universitas Brawijaya)

Afiqie Fadhihansah, Denny Sagita Rusdianto, S.Kom, M.Kom, Novanto Yudistira, S.Kom, M.Sc Program Studi Informatika Universitas Brawijaya Malang E-mail: afiqie.fadhihansah@yahoo.com

ABSTRACT

Researchers designing, designing and implementing the Hospital Enterprise System Using Design Pattern Model View Controller (MVC). MVC concept is the concept of separation between the display and database logic. The benefits of this concept is, make the coding more simple logic, since it has been separated from the code to look and make programmers can work separately with designers. Programmers working on logic, while the designer struggling with the design and look of [SOF-07].

Analysis of software requirements consist of: business modeling (as-is) and modeling system (to-be). Throughout the implementation phase of the analysis and design of the system will be implemented, namely through the process of coding (coding) using PHP programming language and is built with the Model View Controller design pattern (MVC). The testing phase of software testing will be done. It is intended to identify the weaknesses that exist in the application that has been created. Application testing is done in three ways, namely: Testing Functionality; Usability Testing for Non-functionality; Integration Testing.

Based on the results of testing Usability, Functionality testing and Integration testing has shown that the system was feasible and could be used for the benefit of the clinic in the early stages. The results of the questionnaire showed that users have to accept the system of the Hospital Enterprise System using the MVC design pattern. Results of testing functional and non-functional indicates that the system is 100% valid.

Key words: enterprise systems, hospitals, design pattern, MVC.

ABSTRAK

Peneliti merancang, mendesain serta mengimplementasikan Sistem Enterprise Rumah Sakit dengan Menggunakan Design Pattern Model View Controller (MVC). Konsep MVC adalah konsep pemisahan antara logic dengan tampilan dan database. Manfaat konsep ini adalah, membuat coding logic lebih simple, karena sudah di pisah dengan code untuk tampilan dan membuat programmer dapat bekerja secara terpisah dengan designer. Programmer mengerjakan logic, sedangkan designer berkutat dengan desain dan tampilan [SOF-07].

Analisis kebutuhan perangkat lunak terdiri dari: pemodelan bisnis (as-is) dan pemodelan sistem (to-be). Tahap implementasi seluruh analisis dan perancangan sistem akan diimplementasikan, yaitu melalui proses pengkodean (coding) menggunakan bahasa pemrograman PHP dan dibangung dengan design pattern Model View Controller (MVC). Tahap pengujian akan dilakukan pengujian perangkat lunak. Hal ini dimaksudkan untuk mengetahui kelemahan yang ada di dalam aplikasi yang telah selesai dibuat. Pengujian aplikasi ini dilakukan dengan tiga cara, yakni: Pengujian Functionality; Pengujian Usability for Non-functionality; Pengujian Integration.

Berdasarkan hasil pengujian Usability, pengujian Functionality dan pengujian Integration telah menunjukkan bahwa sistem telah layak dan dapat digunakan untuk kepentingan poliklinik pada tahap awal. Hasil kuisioner didapatkan bahwa pengguna telah dapat menerima sistem Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC ini. Hasil pengujian fungsional dan nonfungsional menunjukkan bahwa sistem 100% valid.

Kata kunci: sistem enterprise, rumah sakit, design pattern, MVC

1. PENDAHULUAN

1.1 Latar belakang

TIK (Teknologi Informasi dan Komunikasi) tidak dapat dipungkiri lagi saat ini telah menjadi salah satu keunggulan kompetitif dalam bersaing bagi suatu negara. Sebagai produk yang dihasilkan dari kreativitas dan kemampuan ilmu pengetahuan, TIK menjadi sandaran masa depan yang tidak akan habis untuk digali dan dikembangkan.

Salah satu contoh sebagian kecil bidang TIK adalah bagaimana pasar aplikasi berkembang dengan pesat di seluruh dunia. Misalnya dalam hal kesehatan, kemajuan teknologi menuntut setiap rumah sakit berlomba-lomba memaksimalkan pelayanan kepada masyarakat, khususnya untuk pasien, salah satunya dengan menerapkan sistem informasi rumah sakit. Sistem informasi rumah sakit saat ini menjadi kebutuhan di tiap rumah sakit, karena bertujuan untuk penghematan kertas, juga alur informasi yang didapat juga lebih cepat dan efisien. Namun saat ini sistem informasi rumah sakit masih belum banyak diterapkan di tiap rumah sakit, seperti di Kota Malang, Jawa Timur. Rumah Sakit Saiful Anwar yang menjadi rumah sakit terbesar di Kota Malang pun juga belum menerapkan sistem informasi terintegrasi.

Peneliti merancang, mendesain serta mengimplementasikan Sistem Enterprise Rumah Sakit dengan Menggunakan Design Pattern Model View Controller (MVC). Konsep MVC adalah konsep pemisahan antara logic dengan tampilan dan database. Manfaat konsep ini adalah, membuat coding logic lebih simple, karena sudah di pisah dengan code untuk tampilan dan membuat programmer dapat bekerja secara terpisah dengan Programmer mengerjakan sedangkan designer berkutat dengan desain dan tampilan [SOF-07].

ERP merupakan sebuah kerangka kerja transaksi berkala perusahaan yang meghubungkan antarproses. ERP bekerja sebagai kekuatan lintas fungsional perusahaan yang mengintegrasikan dan mengautomatisasi berbagai proses bisnis internal dan sistem informasi termasuk manufaktur, logistik, distribusi, akuntansi, keuangan dan sumber daya manusia sebuah perusahaan [ENT-02:35-37].

ERP menjadi sistem yang tak terpisahkan dan sifatnya penting dalam proses bisnis secara

keseluruhan dari beberapa organisasi, dan merupakan salah satu investasi terbesar teknologi informasi untuk beberapa perusahaan. ERP merupakan salah satu sektor yang paling cepat berkembang dari industri sistem informasi [LEO-10].

Rumah sakit terintegrasi membutuhkan perencanaan dan sistem kontrol secara terpusat untuk merencanakan proses pada pasien dan kapasitas yang diperlukan. ERP pada rumah sakit dapat sangat berguna untuk perencanaan dan mengendalikan proses deterministik [MER-04]

Pembeda Sistem Enterprise Rumah Sakit dengan sistem yang lain adalah terlihat pada alur validasi, bisnis proses serta pembagian tugas berbasis modul. Studi kasus peneliti dilakukan di Poliklinik Universitas Brawijaya Malang, hal ini dikarenakan belum adanya sistem informasi di Poliklinik Universitas Brawijaya Malang. Sehingga diperlukan sebuah sistem untuk membantu bisnis proses yang ada di Poliklinik Universitas Brawijaya Malang.

Terdapat enam modul yang akan peneliti buat, yaitu modul Front Office, Medicare, Human Resource, Financial, Chasier, Pharmacy dan Medical Records. Tiap modul memiliki fungsi dan tujuan yang berbeda berdasarkan aktor yang menjalankannya.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah diuraikan di atas maka didapat rumusan masalah yang meliputi :

- Bagaimana menganalisis kebutuhan dari pengguna dan merancang aplikasi Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC?
- Bagaimana mengimplementasikan pembuatan aplikasi Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC?
- Bagaimana menguji aplikasi Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC?

1.3 Batasan masalah

Lingkungan Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC dibatasi pada Poliklinik Universitas Brawijaya Malang

1.4 Tujuan

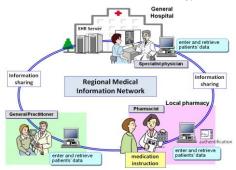
Menganalisis dan merancang kebutuhan dari pengguna, mengimplementasikan pembuatan dan menguji aplikasi Sistem Enterprise Rumah Sakit dengan menggunakan design pattern Model View Controller (MVC)

1.5 Manfaat

Pembuatan Sistem Enterprise Rumah Sakit sebagai aplikasi yang mudah digunakan dan dibutuhkan di pelayanan kesehatan seperti rumah sakit.

2. TINJAUAN PUSTAKA

2.1 Health Information Technology



Gambar 1. Contoh Alur Informasi dan Layanan dalam Komunitas Di Lingkup Medis yang Diterapkan di Jepang.

Sumber: [CHO-11:162]

Sebuah penelitian dari Chon Abraham, 2011 dalam International Journal of Medical Informatics 80 (2011) halaman 157–170 mengenai upaya Jepang untuk menggunakan Health Information Technology (HIT) dalam melakukan reformasi kesehatan. Hasil dari studi kasus masyarakat Yamashina Kyoto bidang medis sebagai bukti konsep untuk kebijakan masa lalu dan sekarang di Jepang yang mendalam untuk proliferasi proyek sukses di Jepang dan adopsi HIT pada umumnya.

Peluncuran sistem yang memungkinkan untuk diterapkan ke setiap rumah sakit menjadi pilihan warga Jepang. Masyarakat membutuhkan penyatuan sumber daya dan menggunakan HIT untuk membantu mengurangi resiko pada rumah sakit utama di daerah pedesaan dan meningkatkan kemampuan perawatan dari masyarakat di seluruh pedesaan.

Selain itu, biaya administrasi perawatan meningkat akibat kenaikan asuransi terkait dengan layanan untuk populasi lansia yang tumbuh pesat. Perawatan umum sistem di Jepang, di mana pasien biasanya mengasumsikan 30% dari biaya-biaya sedangkan 70% sisanya dibayar dan dibebankan oleh negara dan pemerintah daerah atau asuransi kesehatan masyarakat yang didanai oleh perusahaan yang menaunginya. Semua pilihan asuransi mengoperasikan fasilitas perawatan terpisah dalam masyarakat dengan mayoritas dioperasikan oleh swasta, namun, pasien dapat mengunjungi salah satu fasilitas terlepas dari cakupan asuransi mereka.

Klaim pro-cessing ditangani untuk setiap fraksi oleh organisasi seperti Asuransi Klaim Ulasan & Layanan Penggantian. Elektronik klaim (e-klaim atau on-line klaim), sebagai jenis HIT, siap untuk membantu mengurangi biaya yang terkait dengan pengumpulan data pasien (hard copy atau soft copy) dari entitas berbagai kesehatan melayani pasien di seluruh daerah, kurang dari 20% dari kesehatan. Pooling klaim seluruh fasilitas administrasi pengolahan di masyarakat untuk pasien dilayani di berbagai pemerintah dan swasta. Juga, kebijakan penasihat di Jepang percaya bahwa penyatuan data ini lebih baik harus mengaktifkan analisis praktik dengan bukti untuk kualitas dan efektivitas biaya medis, yang pada gilirannya dapat digunakan untuk menentukan penilaian kesehatan masyarakat dan protokol senilai mereplikasi di komunitas medis.

Seperti halnya dalam sistem kesehatan, pasien ditangani oleh para tenaga profesional yang tersedia di daerah, nasional, dan swasta yang dioperasikan layanan dalam perawatan. HIT memungkinkan cara untuk mengkonsolidasikan elektronik medis dan administrasi terhadap pasien dengan harapan kualitas efisiensi dan pelayanan meningkatkan dalam peningkatan informasi. [CHO-11:158-159].

2.2 Code Igniter

Selayang Pandang Code Igniter

Code Igniter adalah sebuah framework PHP. Framework itu sendiri adalah suatu kerangka kerja yang berupa sekumpulan folder yang memuat filefile php yang menyediakan class libraries, helpers, plugins dan lainnya. Framework menyediakan konfigurasi dan teknik coding tertentu.

Mendapatkan Code Igniter

Code Igniter dapat di peroleh secara gratis, dengan mendownloadnya di www.codeigniter.com.

Instalasi Code Igniter

Karena berupa folder php, maka instalasi Code Igniter, cukup dengan mengekstrak file kompresi yang di dapat dari situs www.codeigniter.com dan menempatkannya di dalam folder directory web server Anda. Kemudian edit file config.php yang ada di application/config/config.php dan set base URL anda, misal: http://localhost/ci. Jika anda hendak menggunakan database. edit file database.php di yang application/config/database.php. Setting database anda di file tsb.

Setelah di lakukan instalasi dan setting, test lah Code Igniter anda dengan membuka browser anda dan mengetikkan url yang anda taruh di file config.php di atas, misal: http://localhost/ci.

Jika muncul halaman welcome ala Code Igniter, maka Code Igniter telah siap di gunakan.

Konsep Model-View-Controller (MVC)

Konsep MVC adalah konsep pemisahan antara logic dengan tampilan dan database. Manfaat konsep ini adalah, membuat coding logic lebih simple, karena sudah di pisah dengan code untuk tampilan dan membuat programmer dapat bekerja secara terpisah dengan designer. Programmer mengerjakan logic, sedangkan designer berkutat dengan design dan tampilan.

Model merupakan code struktur data. Model berisi fungsi di dalam pengolahan database. Script Sql masuk di model. View merupakan code untuk menampilkan tampilan suatu program. Tampilan dapat berupa web page, header, footer dan apa saja yang berjenis tampilan. Sedangkan controller merupakan code untuk logic, algoritma dan sebagai penghubung antara model, view, dan sumber lain yang di perlukan untuk mengolah HTTP request dan generate web page.

CI menerapkan pola MVC yang flexible, karena model dapat tidak di gunakan. Anda dapat hanya menggunakan Controller dan View saja dalam menggunakan CI tanpa Model. Jika anda tidak memerlukan pemisahan di dalam struktur data dan database atau menganggap penggunaan model hanya menambah kompleks aplikasi dengan keuntungan yang kurang sebanding, maka anda dapat tidak menggunakan model. [SOF-07]

3. METODE PENELITIAN

Bab ini membahas metode yang digunakan dalam penelitian yang terdiri dari studi literatur,

metode analisis, metode perancangan sistem, metode implementasi, metode pengujian serta pengambilan kesimpulan dan saran. Berikut adalah diagram alir dari metodologi penelitian yang dilakukan



Gambar 3. 1 Alur Pelaksanaan Kegiatan Sumber: Perancangan

Pelaksanaan skripsi ini, dibagi menjadi 5 tahapan, yaitu tahap studi literatur, tahap analisis kebutuhan, tahap perancangan sistem, tahap implementasi dan tahap pengujian.

3.1 Tahap Studi Literatur

Studi literature menjelaskan dasar teori yang digunakan untuk menunjang penulisan skripsi. Teori-teori pendukung tersebut meliputi: Health Information Technology; ERP; Rekayasa Perangkat Lunak; Unified Modeling Language; Pengujian Perangkat Lunak; Code Igniter; Basis Data; MySQL

3.2 Tahap Analisis Kebutuhan

Analisis sistem merupakan tahap yang paling penting dalam suatu pengembangan sebuah aplikasi, karena kesalahan pada tahap analisis sistem akan menyebabkan kesalahan pada tahap selanjutnya. Dengan adanya proses ini, diharapkan dapat menentukan sejauh mana aplikasi yang dibuat tersebut dapat mencapai target. Dari proses tersebut akan dihasilkan suatu gambaran sistem yang kemungkinan memiliki kesalahan-kesalahan ataupun kelemahan-kelemahan sehingga dimungkinkan dilakukan perbaikan.

Dalam sebuah sistem pasti sering terjadi suatu kendala untuk mencapai suatu tujuan dan sering kali kita menyadari bahwa masalah itu terjadi setelah sistem yang berjalan tidak benar. Jika terjadi masalah kita harus bisa menemukan penyebabnya, karena masalah timbul tidak dengan sendirinya melainkan ada sebab yang menimbulkan masalah

tersebut. Tujuan dilakukannya analisis sistem yaitu untuk memahami sistem yang sedang berjalan.

Analisis kebutuhan dilakukan dengan menentukan kebuuhan apa saja yang dibutuhkan untuk membangun Sistem Informasi Rumah Sakit berbasis Enterprise Resource Planning. Proses Analisis ini meliputi:

- Pemodelan *usecase model* dengan as is dan to be
- Pemodelan activity diagram model bisnis
- Pemodelan activity diagram model sistem

Metode analisis yang digunakan adalah Object Oriented Analysis dengan menggunakan bahasa UML (Unified Modelling Language). Fungsi-fungsi yang dapat dilakukan oleh aplikasi ini antara lain: CRUD proses bisnis Sistem Enterprise Rumah Sakit melakukan pelaporan mengenai front office, medicare, pharmacy, medical records, financial and accounting, human resoruce.

Kebutuhan yang digunakan dalam pembuatan aplikasi ini meliputi:

3.2.1 Kebutuhan Fungsional

Kebutuhan fungsionalistas berisi proses-proses apa saja yang nantinya dapat dilakukan oleh sistem, dibutuhkan sebuah sistem yang mampu melakukan fungsi-fungsi seperti :

- Sistem mampu berjalan dengan baik waktu running.
- Sistem mampu menjalankan fungsi-fungsi yang ada pada progam
- Aplikasi dapat berjalan dengan baik sesuai dengan kontrol dari user.

Sistem dapat melakukan penghitungan ketika sedang dijalankan, sesuai dengan keinginan dan harapan Poliklinik Universitas Brawijaya.

3.2.2 Kebutuhan Non Fungsional

3.2.2.1 Kebutuhan perangkat keras

Perangkat keras yang dibutuhkan sebagai sarana penunjang dalam pembuatan Sistem Enterprise Rumah Sakit ini, yaitu berupa seperangkat PC (Personal Computer).

3.2.2.2 Kebutuhan perangkat lunak

Kebutuhan perangkat lunak meliputi:

- Microsoft Windows 8 Proffesional sebagai sistem operasi
- NetBeans IDE 7.4 sebagai platform pengembangan
- XAMPP sebagai tool pembuatan server local
- Astah UML sebagai tool pembuatan diagram pemodelan sistem

3.3 Tahap Perancangan Sistem

Aplikasi ini dirancang dengan menggunakan proses model incremental. Dalam model ini client mengidentifikasi secara garis besar layanan yang diberikan sistem. Model proses incremental mempunyai fase-fase perancangan software, yaitu: penentuan kebutuhan, spesifikasi kebutuhan, perancangan arsitektur software, desain software dan pengkodean serta pengujian [IAN-06].

Kelebihan dari model proses ini antara lain adalah proses development yang lebih cepat, lebih mudah dalam mengetahui kebutuhan pengguna, dan resource yang dibutuhkan untuk melakukan perubahan terhadap perangkat lunak yang dikembangkan lebih sedikit [IAN-11:33].

3.4 Tahap Implementasi

Pada tahap ini, seluruh analisis dan perancangan sistem akan diimplementasikan, yaitu melalui proses pengkodean (coding) menggunakan bahasa pemrograman PHP dan dibangung dengan design pattern Model View Controller (MVC).

3.5 Tahap Pengujian

Pada tahap ini akan dilakukan pengujian perangkat lunak. Hal ini dimaksudkan untuk mengetahui kelemahan yang ada di dalam aplikasi yang telah selesai dibuat. Pengujian aplikasi ini dilakukan dengan tiga cara, yakni:

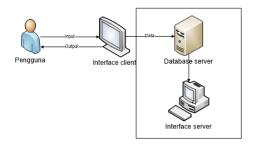
- Pengujian Functionality
- Pengujian Usability for Non-functionality
- Pengujian Integration

4. PERANCANGAN DAN IMPLEMENTASI

Pada tahap ini akan dibahas mengenai arsitektur, kerangka kerja dan analisis kebutuhan yang akan digunakan dalam pengembangn sistem ini.

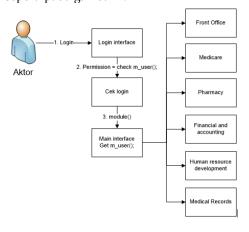
4.1 Arsitektur Sistem Rumah Sakit berbasis ERP

Aplikasi Sistem Enterprise Rumah Sakit dapat melakukan kegiatan registrasi pasien, rawat pasien, transaksi obat, pembayaran pasien dan pengaturan pengguna. Gambar 4.1 berikut menunjukkan arsitektur sistem secara umum:



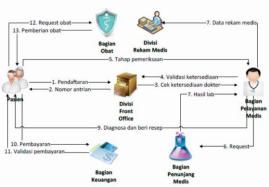
Gambar 4.1 Arsitektur Sistem Sumber: Perancangan

Aplikasi yang dikembangkan bersifat *thin client* sehingga kinerja operasi aplikasi ringan di sisi client. Basis data aplikasi diletakkan di server oleh karenanya dalam menggunakan aplikasi ini, client hanya mengirim dan menerima data yang sudah diproses di server. Dari arsitektur tersebut, aplikasi yang akan dikembangkan memiiki struktur kelas seperti pada gambar 4.2



Gambar 4.2 Komunikasi dengan diagram aplikasi Sumber: Perancangan

Pada tahap (1) aplikasi menerima data masukan berupa username dan password di antarmuka login. Sistem melakukan pengecekan login dari user berupa hak akses (2). Setelah hak akses pengguna yang login diketahui maka aplikasi akan menampilkan antarmuka aplikasi (3) menurut hak akses pengguna apakah front office, medicare, pharmacy, medical records, financial and accounting, human resoruce.



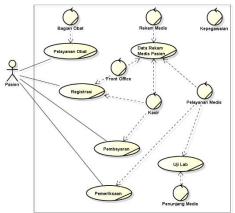
Gambar 4.3 Integrasi antarmodul Sumber: Perancangan

4.2 Analisis Kebutuhan

Pada tahap ini proses analisis kebutuhan dibagi menjadi dua, yakni pemodelan bisnis dengan usecase (as-is) dan pemodelan sistem (to-be). Aplikasi yang akan dibuat pada peneilitan ini adalah aplikasi yang digunakan untuk mengotomatisasi proses bisnis pada rumah sakit. Analisis ini akan membandingkan proses bisnis sebelum adanya sistem informasi berbasis komputer (as-is) dan proses bisnis yang berbasis komputer (to-be).

4.2.1 Pemodelan Bisnis dengan Usecase

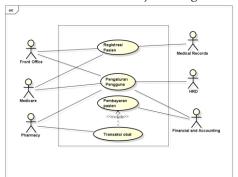
Pada bagian ini akan dibahas mengenai prosesproses yang saat ini terjadi pada Poliklinik Universitas Brawijaya Malang sebelum menggunakan Sistem Enterprise Rumah Sakit. Adapun setian kegiatan dapat dijelaskan oleh Gambar 4.4:



Gambar 4.4 Usecase Model Bisnis (AS-IS) Sumber: Perancangan

4.2.1 Pemodelan Sistem dengan Usecase

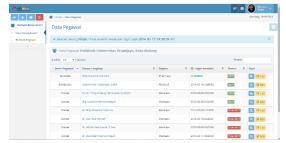
Dari penjelasan proses bisnis pada sub bab sebelumnya maka dikembangkan suatu aplikasi berbasis ERP dan dibuat dengan menggunakan design pattern MVC yang dapat mengurangi aktivitas manual menjadi otomatisasi. Dalam pengembangan sistem ini perubahan analisis kebutuhan dari sistem menjadi sebagai berikut:



Gambar 4.5 Usecase Model Sistem (to-be) Sumber: Perancangan

4.3 Implementasi

4.3.1 Human Resource



Gambar 4.6 Salah satu tampilan di modul HR Sumber: Implementasi

4.3.2 Pharmacy



Gambar 4.7 Salah satu tampilan di modul Pharmacy Sumber: Implementasi

4.3.3 Medical Records



Gambar 4.8 Salah satu tampilan di modul MR Sumber: Implementasi

4.3.4 Front Office



Gambar 4.9 Salah satu tampilan di modul Front Office

Sumber: Implementasi

4.3.5 Financial



Gambar 4.10 Salah satu tampilan di modul Financial

Sumber: Implementasi

4.3.6 Medicare



Gambar 4.11 Salah satu tampilan di modul Medicare Sumber: Implementasi

5. ANALISIS DAN PENGUJIAN

5.1 Pengujian Functionality

Dalam melakukan pengujian functionality maka mengacu kepada dokumen SRS yang telah dibuat sebelumnya. Proses analisis terhadap hasil pengujian Functionality dilakukan dengan melihat kinerja aplikasi. Berdasarkan hasil pengujian tersebut dapat disimpulkan bahwa Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC telah berjalan sesuai dengan kebutuhan sementara bisnis proses Poliklinik Universitas Brawijaya. Hasil yang didapat adalah valid dan sesuai dengan kebutuhan SRS.

5.2 Pengujian Usability for Non-functional Testing (Information)

Pengujian usability melibatkan penyebaran kuisioner terhadap pengguna dimana dalam kasus ini adalah dokter, perawat, staff rekam medis, staff kasir, staff kepegawaian, staff depo obat, dan staff front office masing-masing sebanyak satu orang.



Gambar 5. 1 Hasil Kuisioner Sumber: Pengujian

Pengujian Usability dilakukan dengan melihat hasil wawancara dan kuisioner yang diberikan kepada staff Poliklinik Universitas Brawijaya. Pertanyaan-pertanyaan meliputi: Tampilan aplikasi mudah dipahami; Aplikasi turut membantu mempermudah kinerja dalam pengguna; Tidak terjadi bug/error/kesalahan pada aplikasi; Aplikasi mudah digunakan atau dijalankan; Menu-menu pada aplikasi sudah sesuai dengan kebutuhan; Aplikasi berjalan dengan baik tanpa kendala. Setiap pertanyaan diberi nilai antara 1 s.d 4 (1: Tidak setuju; 2: Kurang setuju; 3: Setuju: 4: Sangat setuju).

Berdasarkan hasil pengujian tersebut dapat disimpulkan bahwa Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC dapat diterima dan diterapkan di Poliklinik Universitas Brawijaya dengan memberikan pelatihan terlebih dahulu kepada setiap pengguna.

5.3 Pengujian Integration

Pengujian integrasi dilakukan untuk mengetahui interasi masing-masing modul sesuai dengan sequence diagram. Hasil dari pengujian integrasi adalah valid dan sesuai dengan kebutuhan sequence diagram.

6. KESIMPULAN

- 1. Untuk menganalisis kebutuhan sistem, dilakukan survei langsung pada Poliklinik Universitas Brawijaya dalam menyusun gambaran proses bisnis yang ada pada saat itu (as-is). Dari hasil analisis tersebut dibuat desain perancangan untuk proses rancang bangun Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC.
- Sistem Enterprise Rumah Sakit dibangun menggunakan aplikasi berbasis web yang dibangun di atas design pattern MVC (Model View Controller) dan framework codeigniter.
- 3. Berdasarkan hasil pengujian Usability, pengujian Functionality dan pengujian Integration telah menunjukkan bahwa sistem telah layak dan dapat digunakan untuk kepentingan poliklinik pada tahap awal. Hasil kuisioner didapatkan bahwa pengguna telah dapat menerima sistem Sistem Enterprise Rumah Sakit dengan menggunakan design pattern MVC ini. Hasil pengujian fungsional dan non-fungsional menunjukkan bahwa sistem 100% valid

Adapun saran dari peneliti adalah sebagai berikut:

- Untuk pengembangan selanjutnya disarankan menambah fitur yang terdapat dalam setiap modul serta menambah modul yang belum tercakup secara keseluruhan.
- Untuk pengembangan mobilitas aplikasi dapat juga dikembangan versi mobile.

7. DAFTAR PUSTAKA

[MIK-11] Akiyama, Miki. 2011. "Impact of information sharing through electric medical network between pharmacist and physician on patient care in Japan". Journal of Keio University.

[CHO-11] Chon Abraham. 2011. "Transforming healthcare with information technology in Japan: A review of policy, people, and progress". International Journal of Medical Informatics, Vol 80, hal. 157-170.

- [FAU-08] Fauci AS, et al. 2008. Harrison's Principles of Internal Medicine 17th edition. New York: The McGraw-Hill Companies, Inc.
- [GDS-11] Goods. 2011. Interest Your Child in Educational Computer Aplikasi. http://www.edmontonnaoshweek.com (Online 5 Maret 2013).
- [LEO-10] Jackson, Leonard A. 2010. "Enterprise resource planning systems: revolutionizing lodging human resources management". Kemmons Wilson School of Hospitality & Resort Management, University of Memphis, Memphis, Tennessee, USA.
- [MER-04] Merode, Godefridus G Van, etc. 2004. Enterprise Resource Planning for Hospitals. International journal of Medical Informatics, Vol 73, hal. 493-501.
- [PRE-01] Pressman, Roger S. 2001. Software Engineering: A Practitioner's Approach, Fifth Edition. McGraw Hill.
- [ROS-11] A. S., Rosa & Shalahudin, M. 2011. Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek). Bandung: Modula.
- [SEG-07] S. Egenfeldt-Nielsen. 2007. Overview of research on the educational use of video aplikasis. Digital kompetense. Nordic Journal of Digital Literacy.
- [SOF-07] Sofwan, Akhmad. 2007. Belajar PHP
 dengan Framework Code Igniter.
 Komunitas eLearning
 IlmuKomputer.Com
- [SOM-03] Sommerville, Ian. 2003. Software

 Engineering (Rekayasa Perangkat Lunak) /

 Edisi 6 / Jilid 1. Jakarta: Penerbit Erlangga
- [SOM-06] Sommerville, Ian. 2006. Software

 Engineering (Rekayasa Perangkat Lunak) /

 Edisi / Jilid. Jakarta: Penerbit Erlangga
- [MAR-09] Graban, Mark. 2009. "Statistics on Healthcare Quality and Patient Safety Problems Errors & Harm (Updated April 10, 2013)". http://www.leanblog.org/2009/08/statistics-on-healthcare-quality-and/. Diakses pada tanggal 22 Oktober 2013
- [JDU-08] Rubin, Jeff dan Cihsnell Dana. 2008, "Handbook of Usability Testing", Second Edition, Willey Publishing, Indianapolis.
- [AGS-09] Mudaliar, Garde, dan Sharma, D.P. 2009, "Educational Resource Planning - A Framework for Educational Institutions",

- Vol.09, No. 978-0-7695-3884-6, Hal. 488 493
- [SSS-04] Shehab, E.M, dkk. 2004,"Enterprise Resource Planning An Integrated Review", Emerald Group Publishing Limited
- [SDL-13] "SDLC Tutorial", Tutorialspoint.com.