

# PAGE HIT MONITORING SYSTEM PADA ENTERPRISE RESOURCE PLANNING (ERP II) SERVER

**Radix Rascalía<sup>1)</sup>, Anggun Fadhlín Librianti<sup>2)</sup>, dan Ari Raharjo<sup>3)</sup>**

<sup>1, 2, 3)</sup>Manajemen Informatika, Politeknik Manufaktur Astra

Komplek Astra International Tbk – Gedung B, Jalan Gaya Motor Raya no 8 Sunter II,  
Jakarta, 14330

Telp : (021) 6519555, Fax : (021) 6519821

E-mail : [radix.rascalía@polman.astra.ac.id](mailto:radix.rascalía@polman.astra.ac.id)<sup>1)</sup>, [anggun.librianti@polman.astra.ac.id](mailto:anggun.librianti@polman.astra.ac.id)<sup>2)</sup>,  
[ariraharjo07@gmail.com](mailto:ariraharjo07@gmail.com)<sup>3)</sup>

---

## Abstrak

*PT. X menjalankan proses pemeriksaan hit halaman dalam sebuah portal aplikasi, untuk memeriksa performa server. Sebelumnya, proses tersebut dilaksanakan secara manual. Page Hit Monitoring System merupakan sistem terintegrasi untuk membantu sistem management dalam memonitor seluruh aplikasi dalam portal Pygmalion Scheduled Synchronization (PSS) di divisi IT. Sistem terdiri dari 4 aplikasi, yaitu SQL Server Integration Services (SSIS), SQL Server Analysis Services (SSAS), SQL Server Reporting Services (SSRS), dan Monitoring PSS. Monitoring PSS berfungsi untuk mengetahui count, average, max dan min hit dari setiap halaman yang diakses oleh pengguna. Aplikasi Monitoring PSS dikembangkan dengan bahasa pemrograman ASP.NET Framework v4, web server IIS 7, dan basisdata SQL Server 2008 R2. Sistem tersebut berhasil mempercepat waktu pengunggahan data server ke basisdata. Selain itu, laporan data hit dapat dilihat sesuai dengan interval waktu yang ditentukan oleh pengguna. Dan jika terjadi kegagalan proses, maka data akan disimpan dalam sebuah status log.*

**Kata kunci:** Business Intelligence, page hit monitoring, pelaporan

## Abstract

*PT. X running an examination page hit process in a portal application, to test server performance. Previously, the process is carried out manually. Page Hits Monitoring System is an integrated system to assist management system controls all applications in Pygmalion Scheduled Synchronization (PSS) portal in IT division. The system consists of four applications, namely SQL Server Integration Services (SSIS), SQL Server Analysis Services (SSAS), SQL Server Reporting Services (SSRS), and Monitoring PSS. Monitoring PSS serves to determine count, average, max and min hit of every page accessed by users. Monitoring PSS application developed with ASP.NET programming language Framework v4, IIS 7 web server, and database SQL Server 2008 R2. The system successfully accelerate time uploading data to database server. In addition, data hit reports can be seen by the interval of time specified by users. And if there is a failure process, then data will be stored in a log status.*

**Keywords:** Business Intelligence, page hit monitoring, reporting

## 1. PENDAHULUAN

Perkembangan ilmu pengetahuan dan teknologi saat ini berjalan sangat cepat. Untuk meningkatkan teknologi, perusahaan membutuhkan informasi yang tepat dan akurat. Informasi sangat berpengaruh dalam perkembangan perusahaan, karena akan mempermudah pimpinan dalam mengambil keputusan. PT. X adalah sebuah perusahaan multinasional yang bergerak dalam tujuh bidang. Untuk menjalankan proses bisnisnya, saat ini PT. X mengembangkan sebuah sistem *Extended Enterprise Resources Planning* (ERP II) yang mengelola berbagai bisnisnya dan anak perusahaannya. Penerapan sistem ERP diimplementasikan dengan aplikasi-aplikasi yang tergabung dalam suatu portal yang bernama *Pygmalion Scheduled Synchronization* (PSS).

Saat ini, proses pemeriksaan *hit* halaman aplikasi dalam PSS dilihat dari *log file* yang secara otomatis terbuat setiap hari untuk masing-masing *server*. *Log file* ini digunakan untuk mengetahui seberapa sering suatu halaman dalam aplikasi PSS dipakai (*hit*), kemudian dikonversi ke dalam bentuk *spreadsheet*. Namun, *file* yang dihasilkan masih berupa tabel yang berisi jutaan data. Untuk itu, maka dibutuhkan sebuah sistem informasi yang menampilkan data statistik serta grafik yang dihasilkan dari *log file* tersebut. Data statistik dan grafik dibuat dengan menggunakan teknologi *Bussiness Intelligence* (BI) sebagai alat bantu dalam pengambilan keputusan.

Keseluruhan sistem akan dibagi menjadi dua proses. Proses pertama yaitu proses harian, merupakan pencatatan *log hit* yang berlangsung secara otomatis untuk pengunduhan data dari server ke basisdata. Proses kedua yaitu proses dinamis, dimana DBA dapat melakukan proses pengunduhan berdasarkan permintaan pengguna untuk waktu yang berada di luar proses harian. Dalam pengembangannya, metodologi yang digunakan yaitu *Extreme Programming* (XP). Tahapan pengembangan sistem dengan menggunakan XP yaitu *Exploration, Planning, Iteration to Release, Productionizing, dan Maintenance* [1].

## 2. TINJAUAN PUSTAKA

*Page Hit Monitoring system* PSS adalah sebuah sistem yang berfokus pada kemampuan analisis data untuk menghasilkan sebuah informasi. Setiap *hit* yang terjadi pada aplikasi PSS akan tercatat dalam *log server*. Kumpulan *log* ini akan dikelola dengan melewati beberapa proses yang berbeda, sehingga menjadi sebuah informasi baru. Informasi inilah yang kemudian akan menjadi dasar pengambilan keputusan bagi pihak manajemen terhadap performa kinerja aplikasi PSS. Dalam sistem yang dibangun terdapat beberapa aplikasi yang digunakan untuk menunjang kerja seluruh proses sistem. Aplikasi yang digunakan, yaitu *SQL Server Integration Sistem* (SSIS) *Monitoring PSS*, *SQL Server Analysis Sistem* (SSAS) *Monitoring PSS*, *SQL Server Reporting Sistem* (SSRS) *Monitoring PSS*, dan *Monitoring PSS Application*.

### 2.1 Extreme Programming (XP)

Menurut Scott Ambler, XP memiliki lima *value* utama, yaitu: *communication, simplicity, feedback, courage* dan *humility* [1]. XP memiliki beberapa keunggulan, antara lain:

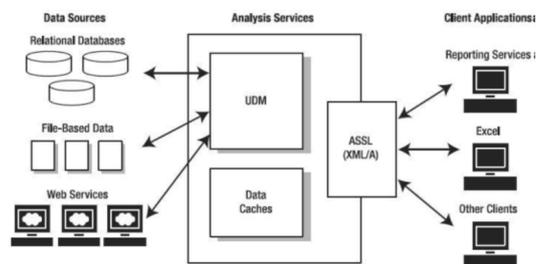
1. Cocok untuk pengembangan perangkat lunak dengan waktu yang singkat,
2. Setiap anggota dalam tim dapat mengerjakan tiap tahapan dalam metode ini, tanpa harus menunggu tahapan lain selesai,
3. Setiap kebutuhan *user* dapat diselesaikan dengan cepat dan baik, dan
4. Dalam pengimplementasian, *user* dapat ikut serta dalam hal pengujian kelayakan dan kebutuhan aplikasi yang dibangun.

### 2.2 Business Intelligence (BI)

Secara umum, BI merupakan sebuah proses untuk melakukan ekstraksi data operasional perusahaan dan mengumpulkannya dalam sebuah *data warehouse*. Selanjutnya data tersebut diproses menggunakan berbagai analisis statistik dalam proses data mining, sehingga didapat berbagai kecenderungan atau pattern dari data. Hasil penyederhanaan dan peringkasan ini disajikan kepada *end user* yang biasanya merupakan pengambil keputusan bisnis. BI memiliki 3 elemen yang menjadi pendukung dalam pengembangannya. Elemen – elemen tersebut, yaitu *data warehouse, data mining, dan online analytical processing*.

### 2.3 SQL Server 2008 Analysis Sistem (SSAS)

*Analysis Services* adalah teknologi untuk OLAP (*Online Analytical Processing*) dan *data mining*. *Development* OLAP menggunakan *Business Intelligence Development Studio* (BIDS) [2]. Dengan menggunakan BIDS, dapat dibangun *core component* dari *analysis services* (*data source, data views, dimension, value, cube, dan sebagainya*). Arsitektur SSAS dapat dilihat pada gambar 1.



Gambar 1: Arsitektur SSAS [2]

### 2.3 SQL Server 2008 Integration Sistem (SSIS)

*SQL Server Integration Services* (SSIS) adalah sebuah *tools* yang digunakan untuk melakukan proses *Extract, Transform, and Load* (ETL) dan diklasifikasikan sebagai fitur *Business Intelligence* (BI) [2]. ETL adalah proses untuk mengumpulkan data dari berbagai sumber (*Extract*), memindahkannya (*Transform*), dan untuk menyimpannya ke sistem lain (*Load*). Dalam kaitannya dengan BI, SSIS adalah fitur yang digunakan untuk

menarik data dari relational database atau file untuk kemudian hasilnya disimpan ke dalam data warehouse.

## 2.4 SQL Server Reporting Sistem (SSRS)

SQL Server Reporting Services (SSRS) adalah platform laporan berbasis server yang menyediakan fungsionalitas pembuatan laporan untuk berbagai sumber data [2]. Terdapat berbagai format untuk menampilkan dan mengeksport laporan ke aplikasi lain, dan mensubskripsi report yang telah dipublikasi. Dalam kaitannya dengan BI, SSRS adalah fitur untuk menampilkan data dari basis data multi dimensional yang membantu melakukan analisis data.

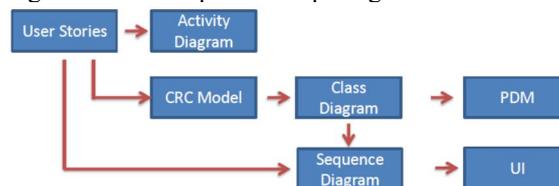
## 2.5 Unified Modeling Language (UML)

UML merupakan standarisasi dalam teknik pembuatan diagram untuk setiap model pengembangan sistem project dari analisis hingga implementasi. Tahapan analisis dibagi menjadi 3 model, yaitu *functional*, *structural* dan *behavioral* [3]. Masing-masing tahapan memiliki artefak yang akan menjadi masukan dalam proses selanjutnya.

## 3. ANALISIS DAN DESAIN

*Page Hit Monitoring system* PSS merupakan suatu sistem terintegrasi yang membantu *system management* dalam memonitor penggunaan aplikasi-aplikasi yang tergabung dalam aplikasi PSS. Sistem ini bertujuan untuk menyajikan data statistik dan grafik dari penggunaan (*hit*) aplikasi PSS berdasarkan kondisi dan tanggal tertentu sesuai dengan keinginan user. Statistik dan grafik yang diperoleh digunakan sebagai alat bantu dalam pembuatan keputusan. PSS *Application Performance Monitoring system* terdiri dari 4 aplikasi yang saling terintegrasi, yaitu aplikasi SSIS, SSAS, SSRS, dan *Monitoring* PSS. SSIS, SSAS, dan SSRS merupakan aplikasi yang bekerja secara background sehingga *users* tidak merasakan secara langsung fungsi dari ketiga aplikasi ini.

Setiap *hit* yang dilakukan oleh *user* pada aplikasi PSS akan tercatat dalam *server log* PSS. Seluruh *log* inilah yang kemudian akan dijadikan bahan dalam proses pembuatan laporan. *Log files* akan dikelompokkan sesuai tanggal hit, dan disatukan ke dalam file .tsv. Data yang sudah terkumpul dalam file .tsv kemudian dikelola dengan memasukkannya ke dalam basis data, untuk selanjutnya dianalisis dan menjadi sebuah laporan yang berbentuk tabel dan grafik. Skema pembangunan sistem dapat dilihat pada gambar 2.



Gambar 2: Skema Pengembangan Sistem

### 3.1 Keadaan Saat Kini (Current)

Proses dimulai dari *request user* ke bagian *Database Administrator* (DBA) melalui email ataupun bertemu langsung untuk melakukan pengunggahan data yang berkaitan dengan proses hit aplikasi. Kemudian DBA mengambil *files log* yang berasal dari satu atau lebih server berdasarkan tanggal dan kondisi tertentu sesuai dengan permintaan user dan disalin ke dalam satu dan dijadikan satu file .tsv. File .tsv yang dihasilkan kemudian diunggah sehingga diperoleh data mentah yang berkaitan dengan proses hit aplikasi. Setelah proses pengunggahan berhasil, data yang sudah ada kemudian disalin ke dalam bentuk spreadsheet file. File ini kemudian diberikan ke user melalui email.

### 3.2 Business Process Automation

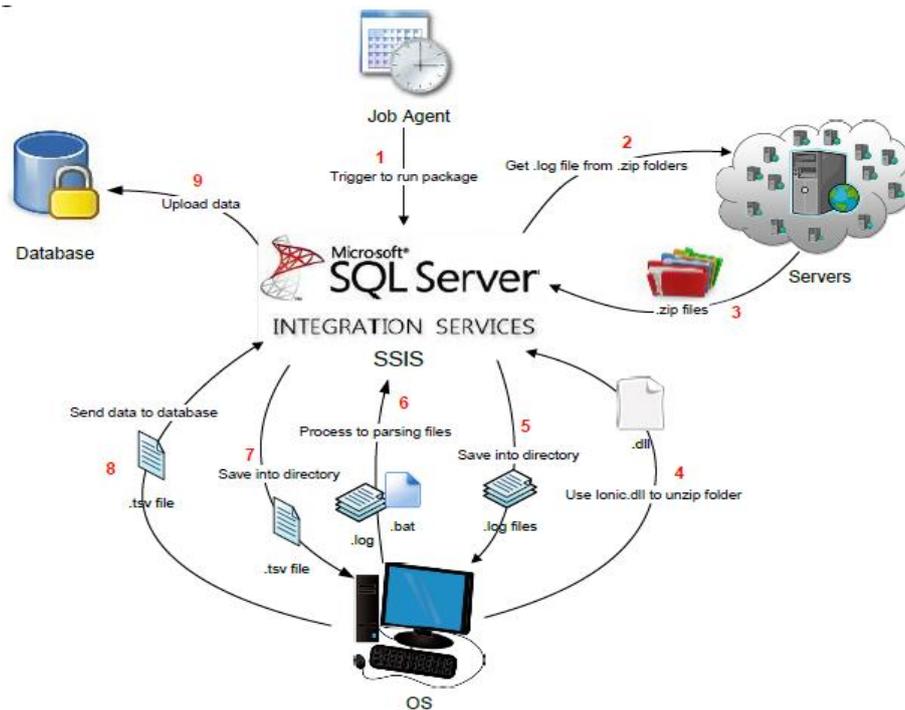
Teknik pembuatan sistem yang digunakan adalah *Business Process Automation* (BPA). Tujuan BPA adalah mengaplikasikan proses bisnis yang sudah ada ke dalam teknologi komputer agar lebih efisien. Dengan menggunakan BPA, analisis akan menghabiskan waktu lebih untuk mengumpulkan informasi tentang sistem, dan mengembangkan model detail untuk sistem tersebut.

### 3.3 Keadaan Setelah Pengembangan (Improvement)

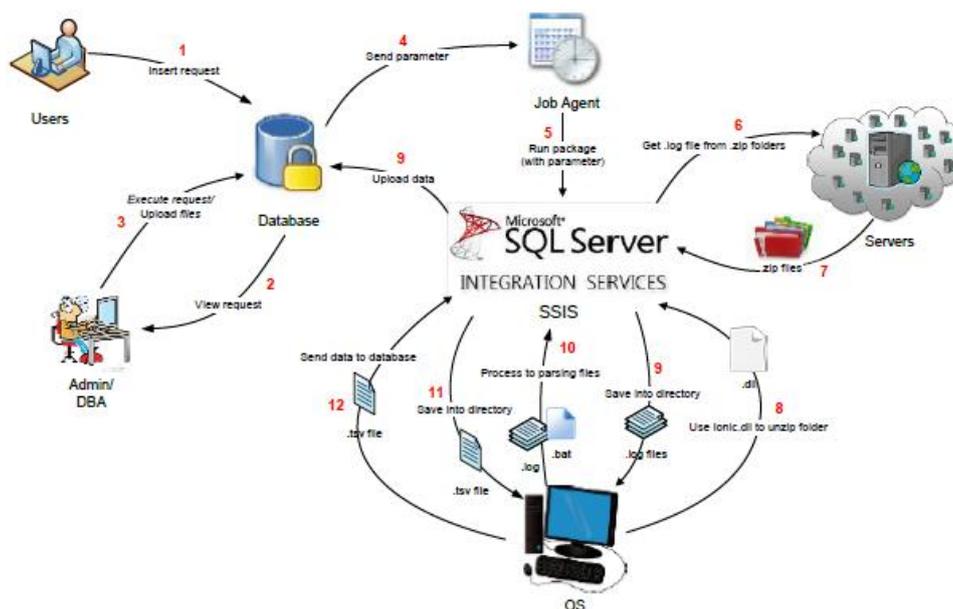
Pada pengembangan aplikasi, peran DBA digantikan *platform* SSIS yang bertugas untuk membangun sistem yang handal untuk integrasi data. Data yang diintegrasikan oleh SSIS diunggah ke basis data secara otomatis oleh sistem. User dapat menggunakan aplikasi *Monitoring* PSS untuk meminta laporan data. *Page Hit Monitoring system* PSS dibangun dengan menggunakan arsitektur 3-tier. 3-tier membagi proses loading sistem ke dalam 3 bentuk lapisan. Pertama, komputer client menjalankan *graphical user interface* (GUI). Kedua, aplikasi *server* menjalankan *business logic*. Dan ketiga, basis data. Client merupakan komputer yang melakukan suatu permintaan data atau layanan, sedangkan server merupakan komputer yang menyediakan data atau layanan yang diminta oleh client.

Gambar pengembangan sistem proses *daily* dan *dynamic* dapat dilihat pada gambar 3 dan 4.

Pada proses harian, *package* dijalankan berdasarkan jadwal yang telah ditentukan setiap harinya. Secara otomatis, *log* akan diambil dari *server* dalam bentuk terkompresi dan disimpan ke dalam sebuah direktori. Sebelum disimpan, file dikembalikan ke dalam bentuk *log*. *Log file* ini akan diparsing ke dalam bentuk *flat file*, kemudian diimpor ke dalam basisdata. Secara umum, proses dinamis sama dengan proses harian. Namun, pada proses dinamis *package* akan dijalankan berdasarkan permintaan dari pengguna aplikasi. Data yang disimpan dalam basisdata adalah data 30 hari ke belakang. Sehingga untuk data yang tidak tersedia harus diminta kembali melalui aplikasi. Pengguna akan melakukan permintaan pada DBA. Permintaan yang masuk akan dieksekusi oleh DBA. Pada saat eksekusi inilah, *package* akan menjalankan proses yang sama dengan proses harian. Perbedaannya terletak pada parameter tambahan di proses dinamis untuk tanggal yang diminta.



Gambar 3: Daily Process



Gambar 4: Dynamic Process

## 1.4 Karakteristik Pengguna

Karakteristik pengguna yang terlibat dalam aplikasi *Monitoring PSS* dapat dilihat pada Tabel 1.

Tabel 1: Karakteristik Pengguna Aplikasi

Karakteristik	Tugas
<i>Super Admin</i>	<i>View status log, view log detail, delete dynamic data, view request, view statistic, manage user, setting</i>
<i>Admin (Database Administrator)</i>	<i>View status log, view log detail, delete dynamic data, view request, view statistic</i>
<i>User</i>	<i>Input request, view request, view statistic</i>

## 4. DESKRIPSI PERANCANGAN SISTEM

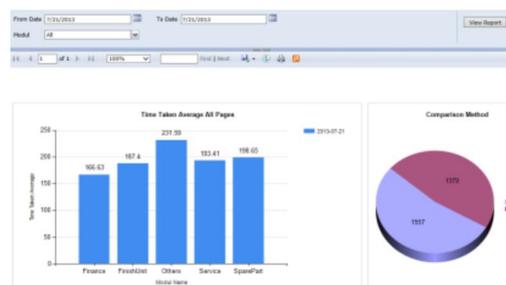
*Physical Data Model* (PDM) merupakan gambaran basis data secara detail dalam bentuk fisik [3]. Penggambaran rancangan PDM memperlihatkan struktur penyimpanan data yang benar pada basis data yang digunakan serta hubungan diantaranya. PDM didapatkan dari artefak *Class Diagram* yang telah dibuat sebelumnya. PDM dibentuk dengan melakukan mapping *Class, Method, Attribut* serta Asosiasi antar class ke dalam bentuk PDM. Penggambaran rancangan PDM yang digunakan pada *Page Hit Monitoring system PSS* menggunakan tipe basis data berupa *Relational Database Management Sistem* (RDBMS). RDBMS merupakan tipe basis data yang banyak digunakan dan hanya mendukung tipe data simple seperti: *int, float, varchar, char*, dan sebagainya.

Dalam menunjang pengelolaan data pada *Page Hit Monitoring system PSS*, dibuat sebuah basis data yang berisi tabel-tabel. Terdapat 2 basis data yang digunakan, yaitu basis data OLAP dan transaksi. Pada basis data transaksi, terdapat 9 tabel yang berisi data permintaan, log, server, dan pengguna aplikasi. Data konfigurasi disimpan dalam file XML. Sedangkan basis data OLAP berisi 6 tabel, dengan 1 tabel utama dan 5 table dimensi. Tabel server pada basis data OLAP terintegrasi dengan tabel server di basis data transaksi. Kedua tabel ini berisi *update* data server yang digunakan sebagai sumber data dalam pengelolaan perhitungan hit.

## 5. IMPLEMENTASI DAN PENGUJIAN

Implementasi dan pengujian termasuk ke dalam tahap *productionizing* dan *maintenance* pada tahapan XP. Pada tahap ini akan dihasilkan dokumentasi uji *testing*, *system documentation* dan *user documentation*. *System documentation* yaitu keseluruhan dokumentasi dari *Page Hit Monitoring system PSS* ini. Sedangkan *user documentation* meliputi dokumentasi yang dibutuhkan oleh user dalam menggunakan aplikasi monitoring PSS. Dokumentasi yang dihasilkan, yaitu: *Standard Operation (SOP)* dan *Technical Guide*. SOP merupakan dokumentasi yang berisi cara penggunaan dari aplikasi *monitoring PSS*. Sedangkan *Technical Guide* merupakan dokumentasi yang berisi tahapan peng-*install*-an aplikasi *monitoring PSS*.

Pengujian meliputi seluruh form yang ada pada sistem. Fungsional yang diujikan meliputi proses fungsi aplikasi dan validasi sesuai dengan proses yang dilakukan pada setiap form. Selama proses pengujian, terdapat beberapa *requirement* tambahan sesuai dengan permintaan *user*. *Requirement* tersebut antara lain: perubahan bentuk *user interface* pada form aplikasi *manage user* agar dibuat lebih *user friendly*, pengunduhan pada form aplikasi *UploadLog* dibuat agar *multiselect* untuk *upload browse* dan sebagainya. Contoh *user interface* dapat dilihat pada gambar 5.



Gambar 5: Tampilan Laporan Aplikasi

Hasil pengujian terhadap *Page Hit Monitoring System PSS* adalah:

1. Sistem mempermudah proses pengunggahan data, baik data *log* harian maupun sesuai dengan permintaan user melalui *form input request* pada aplikasi *Monitoring PSS*,
2. Sistem mempermudah user melakukan permintaan tanggal yang tidak tersedia di dalam basis data,
3. Sistem mengurangi kerja DBA melakukan eksekusi *approval* terhadap permintaan user,

4. Sistem memberikan informasi dan fungsi eksekusi data log yang tidak berhasil diunggah ke basis data,
5. Sistem menyediakan laporan dalam bentuk grafik dan rincian tabel kepada *user*

Tabel 2 dan 3 menunjukkan proses *daily* dan *dynamic*, dalam perbandingan waktu sebelum dan sesudah pembuatan sistem, dengan jumlah *sample* 2 juta data.

Tabel 2: Hasil Pengujian *Daily Process*

Kegiatan (daily)	Sebelum	Sesudah
	(dalam menit)	
Upload report	360	0
Delivery data	5	0
Pembuatan grafik	3	0
Analisis data sesuai hirarki	10	0.1
<b>TOTAL WAKTU</b>	<b>378</b>	<b>0.1</b>

Tabel 3: Hasil Pengujian *Dynamic Process*

Kegiatan (dynamic)	Sebelum	Sesudah
	(dalam menit)	
Request data	5	5
Acceptance request	5	1
Upload report	20	10
Delivery data	5	0
Pembuatan grafik	3	0
Analisis data sesuai hirarki	10	0.1
<b>TOTAL WAKTU</b>	<b>48</b>	<b>16.1</b>

## 6. SIMPULAN DAN SARAN

Berdasarkan pembangunan sistem dengan didukung beberapa teori, maka selama proses pembuatan sistem dari awal hingga selesai dapat diambil simpulan dan saran, sebagai berikut:

### 6.1 Simpulan

Berdasarkan pengujian yang dilakukan di bab 5, dapat diambil kesimpulan bahwa proses pengunduhan *log hit* dapat dilaksanakan secara otomatis, permintaan *users* untuk interval tanggal waktu tertentu dapat dipenuhi, laporan ditunjukkan dalam bentuk tabel dan grafik, dan disediakan laporan pencatatan status pengunggahan data. Dari keseluruhan proses, dengan implementasi sistem ini mengalami efisiensi waktu pada proses harian sebanyak 378 menit. Sedangkan untuk proses *dynamic* sebanyak 32 menit. Efisiensi waktu tersebut disebabkan oleh pengunggahan data otomatis setiap harinya, sehingga mengurangi *load* waktu pada saat pembuatan laporan di akhir bulan atau pada interval waktu tertentu. Dengan efisiensi waktu tersebut, maka proses pengambilan keputusan juga akan semakin cepat. Pihak manajemen tidak perlu lagi menunggu proses manual yang terlalu lama. Dan jika diperlukan laporan mendesak, tidak akan membuang waktu lama karena user dapat langsung meminta dan mengakses laporan sesuai dengan interval tanggal yang diinginkan.

Namun demikian, masih ada yang perlu disempurnakan dalam sistem, yaitu *report* yang dihasilkan masih bersifat statis dan hanya dapat digunakan dengan baik pada *browser* Internet Explorer (IE) 8 dan IE10, sedangkan untuk *browser* lain *report* tidak *compatible* atau tidak berjalan dengan baik. Dari hasil tersebut, disimpulkan bahwa tujuan pembangunan sistem tercapai. Namun, ada kekurangan pada akses *report* yang terbatas *browser* tertentu. Hal ini disebabkan karena terbatasnya *support tools* yang digunakan.

### 6.2 Saran

Berdasarkan pengalaman dalam proses pembuatan sistem, ada beberapa hal yang perlu diperbaiki. Pembuatan *report* yang dihasilkan dapat dikembangkan dengan *tools* BI lainnya seperti *microstrategy* sehingga *report* yang dihasilkan dapat bersifat dinamis. Selain itu, untuk *report* yang dihasilkan lebih baik jika dapat dijalankan di semua *browser*, terutama *browser* yang umum digunakan.

## 7. DAFTAR RUJUKAN

- [1] Ambler, Scott. 2002. Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process. New York : Wiley.
- [2] Wirama, Kasim et all. 2008. The Essential Business Intelligence in Microsoft SQL Server 2008. Jakarta.
- [3] Dennis, Alan et al. 2009. Sistem Analysis and Design with UML Version 2.0. New York: Wiley.