

SISTEM MONITORING SERVER CORE BANKING :

STUDI KASUS BANK DANAMON INDONESIA

Sofian Sidiq¹, Prof. Dr. Moedjiono, M.Sc²

Universitas Muhammadiyah Tangerang / Fakultas Teknik,
Program Studi Informatika

Jl. Perintis Kemerdekaan 1/33 Cikokol Kota Tangerang TLP. 55793251, 55772949,
55793802, 55736926.

Email¹: sofian.sidiq@gmail.com., Email²: moedjiono@depkominfo.go.id,

ABSTRAK

Sistem monitoring *server core banking* dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama atau memperbaiki sistem yang telah ada di suatu perusahaan atau yayasan, termasuk Bank Danamon Indonesia. Sejak berdirinya sampai saat ini belum memiliki sebuah aplikasi yang dapat memberikan informasi untuk memantau server secara detail. Ruang lingkup sistem monitoring yang dikembangkan meliputi: OS (*CPU Utilization, IO dan Memory*), *Java Virtual Machine* dan *ATM*. Metode pengembangan sistem monitoring menggunakan model *Waterfall*. Metode pengumpulan data dilakukan dengan observasi, studi pustaka, dan wawancara terhadap sampel yang dipilih dengan menggunakan metode *purposive sampling*. Kualitas perangkat lunak yang dihasilkan diuji berdasarkan empat karakteristik kualitas perangkat lunak model ISO 9126, yaitu: *functionality, reliability, usability, dan efficiency* menggunakan metode kuesioner. Hasil penelitian berupa perangkat lunak sistem monitoring yang memiliki tingkat kualitas sangat baik dengan total **88,20%**.

Kata Kunci: Sistem Monitoring, *JVM, Waterfall*, Analisis dan Perancangan Berorientasi Obyek, ISO 9126

ABSTRACT

*Server monitoring core banking system may mean constructing a new system to replace the old system or improve existing systems in a company or foundation, including Bank Danamon Indonesia (BDI). Since its establishment until now have not an application that monitoring service system. The scope of personnel monitoring JVM system include: active thread, memory, up time dan connection pool. This research is type of Applied Research (Applied Research). Information systems development method using the Waterfall model. Method of data collection is done by observation, study, and interviews with a selected sample using purposive sampling method. Mechanical testing of the system with black-box testing approach. The quality of the resulting software was tested by four characteristics of software quality model of ISO 9126, namely: functionality, reliability, usability, and efficiency using the questionnaire. The results are in the form of good quality monitoring system software are expected with total **88,20%**.*

Keywords: Monitoring Systems, *JVM, Waterfall, Object-Oriented Analysis and Design, ISO 9126*

I PENDAHULUAN

Sistem teknologi informasi saat ini berkembang sangat pesat. Sistem teknologi informasi biasanya diterapkan di organisasi yang menggunakan komputer sebagai alat bantu utama dalam membantu tercapainya tujuan organisasi, perusahaan maupun instansi yang bersangkutan. Untuk mendapatkan hasil kerja yang efektif dan efisien, organisasi harus mengembangkan suatu sistem teknologi informasi yang memungkinkan orang-orang mempunyai kesempatan berinteraksi dan memanfaatkan sistem teknologi tersebut untuk membantu mencapai tujuan mereka. Pesatnya perkembangan teknologi informasi saat ini mendorong akan kebutuhan informasi yang cepat dan akurat, dalam proses bisnis di organisasi, baik perusahaan maupun instansi. Penggunaan sistem informasi memungkinkan adanya otomatisasi pekerjaan dan fungsi pelayanan untuk mewujudkan pelayanan yang baik seperti yang dibutuhkan.

Prinsip kerja dari sistem monitoring ini adalah memonitor OS, *Java Virtual Machine*, *ATM* dan mengirimkan *summary / alert* melalui email lalu menampilkannya ke sebuah desktop. Sistem monitoring ini memungkinkan untuk mengetahui apabila terjadi masalah seperti database, aplikasi server mati, *ATM reject dll*, sehingga memudahkan mengetahui suatu masalah yang harus di selidiki untuk segera dilakukan perbaikan. Selain itu dapat juga digunakan untuk melakukan analisa dan membandingkan kinerja dari sistem dari waktu ke waktu.

1.1 Identifikasi Masalah

Berdasarkan latar belakang permasalahan dalam penelitian ini, maka permasalahan dapat diidentifikasi sebagai berikut:

1. Bagaimana memonitoring server secara periodik agar informasi yang dibutuhkan tersedia dengan baik.
2. Belum adanya aplikasi yang memonitor sistem secara berkala

1.2 Rumusan Masalah

Berdasarkan latar belakang, identifikasi dan pembatasan masalah, maka permasalahan penelitian dapat dirumuskan sebagai berikut:

1. Bagaimana membangun purwarupa aplikasi monitoring yang menyediakan informasi yang dibutuhkan?
2. Bagaimana tingkat kualitas perangkat lunak sistem aplikasi monitoring *core banking* yang dihasilkan jika diukur mengadaptasi karakteristik kualitas perangkat lunak model ISO 9126?

II LANDASAN TEORI DAN KERANGKA PEMIKIRAN

2.1 Sistem Monitoring

Sistem monitoring digunakan secara berulang kali untuk mengumpulkan data dari berbagai insiden yang dimonitor oleh sistem yang bertujuan untuk pemberitahuan secara *real-time*, seperti verifikasi sistem kesehatan, analisis kinerja dan perencanaan kapasitas longitudinal. Berbagai matrix biasanya dikumpulkan oleh sistem ini termasuk CPU, memori, jaringan atau pemanfaatan disk dan waktu respon dari lamanya jaringan yang menyebabkan terjadinya anomali *log system*. Untuk mengumpulkan informasi tersebut secara berulang kali administrator otomatis menggunakan alat monitoring sistem.

Menurut Badger penulis sistem monitoring *open source zenoss*. Sistem monitoring merupakan aplikasi berbasis web yang di instal pada sebuah server sebagai sentral kendali administrator untuk mengakses sistem monitoring melalui antarmuka web yang memungkinkan untuk melakukan:

- a. Manajemen perangkat (*device management*), biasanya disini ditempatkannya konfigurasi manajemen database untuk menyimpan semua perubahan dan histori data.
- b. Ketersediaan dan kinerja pemantauan (*availability and performance monitoring*), biasanya aplikasi yang digunakan adalah ICMP dan SNMP. Selain itu juga berfungsi untuk mengumpulkan data *time series* dan memberikan analisis grafis.
- c. Manajemen peristiwa (*event management*), sebagai sumber data untuk tanda-tanda masalah termasuk *syslog* ketersediaan dan kinerja monitor SNMP traps dan event log windows.
- d. Sistem laporan (*system report generation*), merupakan suatu set paket laporan standart yang memungkinkan untuk melihat apa yang terjadi. Laporan mengintegrasikan dengan management perangkat, monitor kinerja, peristiwa dan fungsi pengguna.

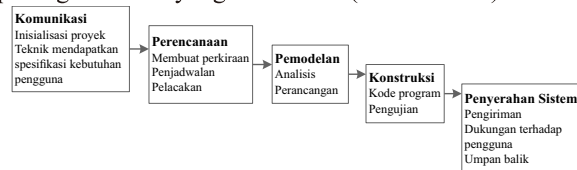
Pengguna dan manajemen sinyal (*user and alert management*), semua kejadian tersebut dapat dilakukan oleh administrator melalui portal web^[1]

2.2 Metode Pengembangan Sistem Model Waterfall

Dalam rekayasa perangkat lunak, konsep SDLC mendasari berbagai jenis metodologi pengembangan perangkat lunak. Metodologi-metodologi ini membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi, yaitu proses pengembangan perangkat lunak. Beberapa proses pengembangan sistem dengan SDLC

adalah: *waterfall*, *prototyping*, *incremental*, *spiral*, dan *RAD*.

Metodologi pengembangan sistem model *waterfall* menurut Pressman adalah proses pengembangan sistem yang menyiratkan pendekatan yang sistematis dan berurutan (sekuensial) pada pengembangan perangkat lunak, dimana proses pengembangan tersebut mengalir secara teratur ke bawah sehingga terlihat seperti air terjun [2]. Proses pengembangan dimulai dengan spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan-tahapan perencanaan (*planning*), pemodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem perangkat lunak ke pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan berkelanjutan pada perangkat lunak yang dihasilkan (Gambar II-1).



Gambar II-1 Model *Waterfall* [2]

2.3 Tinjauan Studi

Penelitian mengenai sistem monitoring pada server telah banyak yang meneliti, beberapa penelitian terdahulu yaitu penelitian yang dilakukan oleh Penelitian yang dilakukan oleh Nova Dwi Kurniawan [3] dalam “Implementasi Sistem Monitor Server Berbasis Web Dengan Memanfaatkan SNMP (Simple Network Management Protocol) Dengan Metode Prototyping” adalah Keberadaan tool jaringan yaitu system monitor jaringan sangat mempermudah dalam pengelolaan berbagai perangkat jaringan diantaranya server. Server ini harus bekerja 24 jam sehingga harus selalu dimonitor dan dipelihara kesehatan sistemnya secara berkala. Untuk menghindari overload (kelebihan beban) pada suatu waktu tertentu. Sistem ini dikembangkan dengan menggunakan metode prototyping yang mempunyai kelebihan memperluas alur SDLC (Software Development Life Cycle) serta mempersingkat waktu dalam pengembangan sistem. Dengan adanya sistem monitor server ini diharapkan para administrator jaringan dapat melakukan monitoring dengan metode peringatan dini yaitu dengan sistem pelaporan berupa email dan SMS yang akan dikirimkan jika terjadi overload atau nilai

diluar jangkauan pada sensor tertentu yang telah ditetapkan sebelumnya oleh administrator

Penelitian yang dilakukan oleh Riza dan Idris [4] dalam “Aplikasi Monitoring Proses pada Komputer Server dengan Menggunakan Java” Komputer monitoring merupakan komputer utama yang akan dibangun dalam proyek akhir ini. Dalam komputer ini akan dibuat sebuah aplikasi yang dapat melakukan monitoring terhadap service yang ada pada masing-masing server. Dengan adanya aplikasi tersebut diharapkan service-service yang sedang berjalan pada server dapat dikontrol penggunaannya

Penelitian yang dilakukan oleh Ahmad Holis [5] dalam “Perancangan Dan Implementasi Monitoring Jaringan Lokal Menggunakan Sistem Kerja Backdoor” Komputer monitoring merupakan komputer utama yang akan dibangun dalam proyek akhir ini. Dalam komputer ini akan dibuat sebuah aplikasi yang dapat melakukan monitoring terhadap service yang ada pada masing-masing server. Dengan adanya aplikasi tersebut diharapkan service-service yang sedang berjalan pada server dapat dikontrol penggunaannya

2.4 Hipotesis

Berdasarkan kerangka konsep yang telah dikemukakan maka pernyataan penelitian ini dapat dirumuskan sebagai berikut:

1. Diduga sistem monitoring pada server core banking ini mampu menyediakan informasi kondisi server secara berkala.
2. Diduga kualitas perangkat lunak sistem monitoring yang dihasilkan jika diukur mengadaptasi kualitas perangkat lunak model ISO 9126 adalah baik.

III METODOLOGI DAN RANCANGAN PENELITIAN

3.1 Metode Penelitian

Penelitian pengembangan sistem monitoring pada *core banking* dengan Studi Bank Danamon Indonesia yang dilakukan merupakan jenis Penelitian Terapan (*Applied Research*). Hasil penelitian dapat langsung diterapkan untuk memecahkan permasalahan yang dihadapi. [6]

Dalam penelitian ini akan menerapkan teori pengembangan sistem informasi menggunakan metode pengembangan sistem model *waterfall*, analisis dan perancangan sistem dengan pendekatan berorientasi obyek, implementasi hasil analisis dan perancangan

menggunakan pemrograman berbasis Java, pengujian validasi menggunakan *Focus Group Discussion*, serta pengujian kualitas perangkat lunak yang dihasilkan menggunakan model ISO 9126. Hasil penelitian berupa aplikasi sistem monitoring di bank Danamon Indonesia langsung dapat diterapkan untuk mendapatkan informasi secara berkala

3.1.2 Metode Pemilihan Sampel

Metode pengambilan sampel dalam penelitian ini adalah *purposive sampling*. Pengambilan sampel dengan *purposive sampling* merupakan teknik pengambilan sampel dengan mengambil responden yang terpilih betul oleh peneliti menurut ciri-ciri spesifik yang dimiliki oleh sampel tersebut.^[7]

Adapun kriteria pemilihan responden sebagai sampel penelitian ini adalah:

1. Memiliki tugas dan tanggung jawab sebagai supervisor atau pimpinan yang bekerja selama lebih dari 1 tahun.
2. Memiliki pengalaman di bidangnya selama lebih dari 1 tahun.

Responden dalam penelitian ini adalah Supervisor IT dan staff IT. Pemilihan responden sampel ini dengan pertimbangan berdasarkan kriteria yang sudah dijelaskan sebelumnya, sehingga sampel tersebut sangat representatif.

3.3 Metode Pengumpulan Data

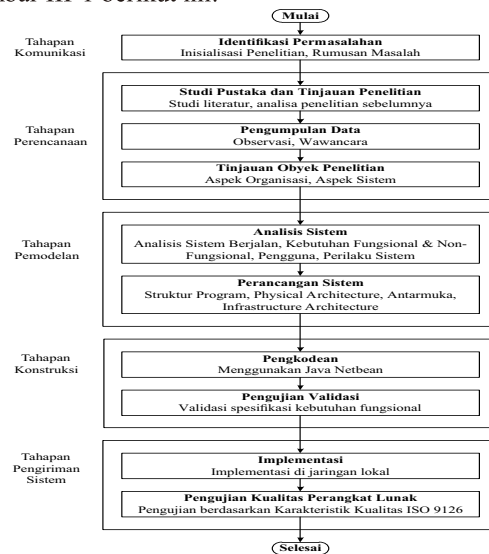
Metode pengumpulan data yang digunakan dalam penelitian ini adalah:

1. Metode Wawancara. Wawancara dengan pihak-pihak yang berkaitan dengan penelitian. Teknik wawancara dilakukan dengan wawancara berstruktur.^[7] Dalam wawancara tersebut peneliti telah menyiapkan daftar pertanyaan yang berkaitan dengan pengembangan sistem monitoring service JVM pada server core banking. Responden dalam wawancara ini adalah *SPV IT Application Analyst* dan *SPV IT Infrastructure*. Pertanyaan-pertanyaan untuk mendapatkan data yang terkait dengan sistem monitoring yang sedang berjalan saat ini, kebutuhan fungsional, nonfungsional, dan pengguna untuk sistem yang akan dikembangkan.
2. Metode Observasi. Observasi atau pembuatan langsung sistem monitoring. Teknik yang dilakukan dengan observasi berstruktur dengan menyiapkan data – data maupun informasi yang dibutuhkan selama pembuatan aplikasi sistem monitoring.
3. Metode Studi Pustaka. Metode pengumpulan data yang diperoleh dengan mempelajari,

meneliti, dan membaca buku, jurnal, skripsi, tesis yang berhubungan dengan sistem monitoring.

3.4 Langkah-Langkah Penelitian

Dalam pengembangan sistem informasi, keseluruhan proses yang dilalui harus melalui beberapa tahapan. Dalam penelitian ini digunakan metode pengembangan sistem informasi model *Waterfall*. Tahapan yang dilakukan meliputi: komunikasi, perencanaan, pemodelan, konstruksi, dan pengiriman sistem. Langkah-langkah pada tahapan pelaksanaan penelitian dapat dilihat dalam bentuk diagram alir pada Gambar III-1 berikut ini:



Gambar III-1 Langkah-langkah Penelitian

IV PEMBAHASAN HASIL PENELITIAN

4.1 Analisis Sistem

4.1.1 Analisis Sistem yang Sedang Berjalan

Proses analisis sistem mendeskripsikan apa yang harus dilakukan oleh sistem untuk memenuhi kebutuhan informasi pengguna. Analisis sistem akan menjawab pertanyaan apa yang akan dikerjakan oleh sistem, siapa yang akan menggunakan sistem, dan dimana serta kapan sistem tersebut akan digunakan. Kegiatan analisis sistem yang berjalan dilakukan dengan pendekatan analisis berorientasi *objek* untuk sistem yang dirancang, dimaksudkan untuk menitik beratkan kepada fungsionalitas sistem yang berjalan. Selanjutnya dari hasil analisis akan divisualisasi dan didokumentasikan dengan *Unified Modeling Language (UML)* melalui *Use Case Diagram* dengan pertimbangan diagram tersebut dianggap mewakili secara keseluruhan sistem yang berjalan yang dapat dimengerti oleh pengguna.

4.1.2 Sistem Yang dikembangkan

Penbuatan sistem monitoring server core banking dapat berarti menyusun suatu sistem agar dapat memenuhi kebutuhan akan informasi server core banking. Sistem monitoring yang dibuat merupakan sistem monitoring yang menangani beberapa aspek mulai dari proses monitoring OS (*CPU Utilization, IO dan Memory*), *Java Virtual Machine* dan *ATM*. Berdasarkan wawancara dan observasi, diperoleh informasi bahwa Bank Danamon memiliki 2 server database 2 server aplikasi dan 10 server sentralize. Tujuan pembuatan sistem dalam penelitian ini adalah membangun aplikasi sistem monitoring berbasis desktop yang dapat menampilkan data dan informasi status server secara berkala.

Sistem yang dibuat dalam penelitian ini diharapkan akan lebih meningkatkan kewaspadaan terhadap server server core banking. Sistem ini diharapkan dapat memberikan informasi data server yang dibutuhkan secara periodik, dapat memberikan laporan yang dibutuhkan kepada setiap bagian yang membutuhkan maupun pimpinan apabila informasi sewaktu-waktu diperlukan.

4.1.3 Analisis Kebutuhan Fungsional & Nonfungsional

Tahap analisis kebutuhan fungsional sistem akan membahas mengenai fungsi-fungsi yang diperlukan dalam pembangunan sistem. Berdasarkan hasil analisis proses, identifikasi kebutuhan data dan informasi, maka dianalisis juga beberapa fungsi yang harus tersedia di dalam sistem. Hal ini dilakukan untuk memenuhi kebutuhan data dan informasi yang diperlukan oleh pengguna.

Berdasarkan hasil wawancara dan observasi ada beberapa permintaan fitur untuk aplikasi ini. kebutuhan – kebutuhan tersebut adalah sebagai berikut:

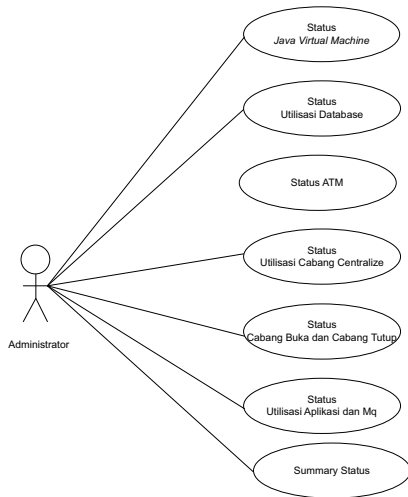
Tabel IV-1 Functional dan Non Functional

Functional	
Saya ingin sistem dapat	
1	Sistem dapat menampilkan informasi status java virtual machine
2	Sistem dapat menampilkan informasi status database
3	Sistem dapat menampilkan informasi status ATM

4	Sistem dapat menampilkan informasi status utilisasi cabang centralize
5	Sistem dapat menampilkan informasi status cabang buka dan cabang tutup
6	Sistem dapat menampilkan informasi status utilisasi aplikasi dan mq
7	Sistem dapat menampilkan informasi status server secara keseluruhan
Non Functional	
Saya ingin sistem dapat	
1	Menampilkan pesan pada layar jika terdapat suatu masalah
2	Hanya membutuhkan waktu 15 menit untuk mendapatkan informasi berikutnya untuk status server
3	Keseluruhan informasi dikirimkan dalam bentuk email
4	Informasi ditampilkan dalam layar besar

4.1.3.1 Use Case Diagram

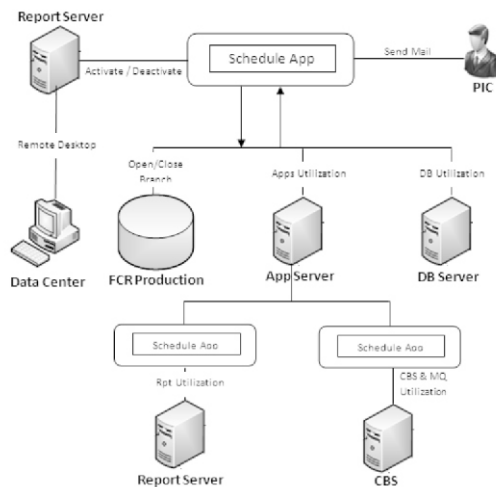
Berdasarkan spesifikasi kebutuhan fungsional dan *actor* yang terlibat dalam sistem, maka dapat dimodelkan dengan *use case diagram*. *Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. *Use case* juga menggambarkan interaksi yang terjadi dalam sistem, yang memberi gambaran *user* atau *actor* yang berhubungan dengan sistem dan hal-hal yang berhubungan dengan user di dalam sistem. Berikut adalah interaksi user dalam sistem monitoring server pada bank danamon Indonesia:



Gambar IV-2 Use Case Diagram Monitoring

4.2 Perancangan infrastructure dan arsitektur

Dalam pembuatan sistem monitoring server core banking dilakukan perancangan arsitektur moiting server sebagai berikut:



Gambar IV-3 Design Sistem Monitoring

Penjelasan gambar diatas sebagai berikut:

1. Data Center akan melakukan remote desktop untuk monitoring Aplikasi Server.
2. Data Center akan mengaktifkan atau menonaktifkan ATM monitoring sesuai request.
3. Aplikasi ini akan tereksekusi agar mendapatkan log setiap 15 menit.

4. Aplikasi dalam laporan akan membagi dan meringkas setiap transaksi berdasarkan log.
5. Aplikasi akan mengirimkan summary atau alert setiap jamnya atau 15 menit jika ada masalah pada server

4.2.1 Konstruksi Aplikasi

4.2.1.1 Konstruksi Parsing File

Bagian ini akan menjelaskan *source code* pembuatan dari log server yang akan di parsing menjadi sebuah informasi:

a. Kontruksi parsing file I

Gambar IV-4 Konstruksi Parsing File I

Dari code diatas dijelaskan bahwa log yang dihasilkan dari sistem di parsing (parse) menjadi bentuk text.

b. Konstruksi parsing file II

Gambar IV-5 Konstruksi Parsing File II

Dari code diatas dijelaskan bahwa log yang dihasilkan dari sistem di parsing (parse) menjadi informasi untuk kebutuhan interface status java viartual machine.

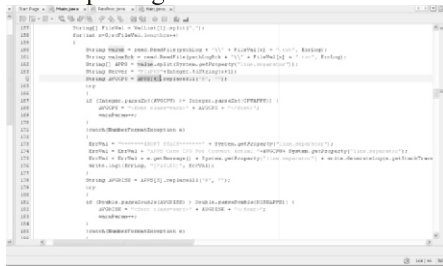
c. Konstruksi parsing file III



Gambar IV-6 Konstruksi Parsing File III

Dari code diatas dijelaskan bahwa log yang dihasilkan dari sistem di parsing (parse) menjadi informasi untuk kebutuhan interface status ATM.

d. Konstruksi parsing file IV



Gambar IV-7 Konstruksi Parsing File IV

Dari code diatas dijelaskan bahwa log yang dihasilkan dari sistem di parsing (parse) menjadi informasi untuk kebutuhan interface status utilisasi cabang centralize, status cabang buka dan cabang tutup dan status utilisasi aplikasi dan MQ.

4.2.1.2 Konstruksi Halaman Status Java Virtual Machine

Bagian ini akan menjelaskan struktur code yang dibuat agar interface dapat berfungsi sebagai mana mestinya:



Gambar IV-8 Konstruksi status java virtual machine

Dari code diatas adalah struktur kode untuk pembuatan interface java virtual machine.

4.2.1.3 Konstruksi Halaman Status Database

Bagian ini akan menjelaskan struktur code yang dibuat agar interface dapat berfungsi sebagai mana mestinya:



Gambar IV-9 Konstruksi status database

Dari code diatas adalah struktur kode untuk pembuatan interface database.

4.2.1.4 Konstruksi Halaman Status ATM

Bagian ini akan menjelaskan struktur code yang dibuat agar interface dapat berfungsi sebagai mana mestinya:



Gambar IV-10 Konstruksi status ATM

Dari code diatas adalah struktur kode untuk pembuatan interface ATM.

4.2.1.5 Konstruksi Halaman Status Utilisasi Cabang Centralize

Bagian ini akan menjelaskan struktur code yang dibuat agar interface dapat berfungsi sebagai mana mestinya:




Gambar IV-11 Konstruksi utilisasi cabang centralize

Dari code diatas adalah struktur kode untuk pembuatan *interface utilisasi cabang centralize*.

4.2.1.6 Konstruksi Halaman Status Cabang Buka dan Tutup

Bagian ini akan menjelaskan struktur *code* yang dibuat agar interface dapat berfungsi sebagai mana mestinya:



Gambar IV-12 Konstruksi status cabang buka dan tutup

Dari code diatas adalah struktur kode untuk pembuatan *interface cabang buka dan cabang tutup*.

4.2.1.7 Konstruksi Halaman Status Utilisasi aplikasi dan Mq

Bagian ini akan menjelaskan struktur *code* yang dibuat agar interface dapat berfungsi sebagai mana mestinya:



Gambar IV-13 Konstruksi status utilisasi aplikasi dan mq

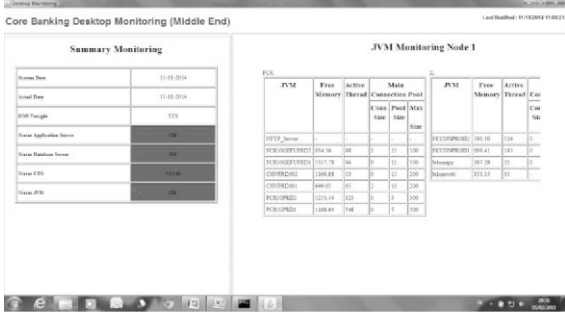
Dari code diatas adalah struktur kode untuk pembuatan *interface utilisasi aplikasi dan MQ*.

4.2.2 Konstruksi Antarmuka

Bagian ini akan menjelaskan implementasi atau kontruksi berdasarkan hasil rancangan tampilan aplikasi sistem monitoring yang sudah dibahas dalam perancangan antarmuka pengguna. Untuk menjelaskan hasil konstruksi tersebut akan diberikan *prototype* dari masing-masing tampilan pada aplikasi yang dibangun.

4.2.2.1 Tampilan Halaman Status Java Virtual Machine

Tampilan dibuat ringkas serta mudah dipahami, selain itu bertujuan mengutamakan fungsi monitoring.

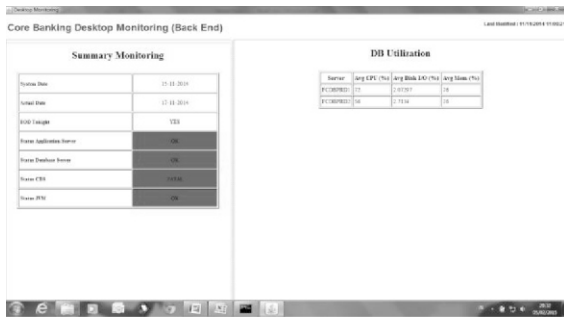


Gambar IV-14 Tampilan Halaman Status Java Virtual Machine

Dari gambar diatas menampilkan informasi summary monitoring untuk menandakan bahwa statusnya normal, warning atau fatal dan menampilkan informasi java virtual machine secara detail

4.2.2.2 Tampilan Halaman Status Database

Tampilan dibuat ringkas serta mudah dipahami, selain itu bertujuan mengutamakan fungsi monitoring.

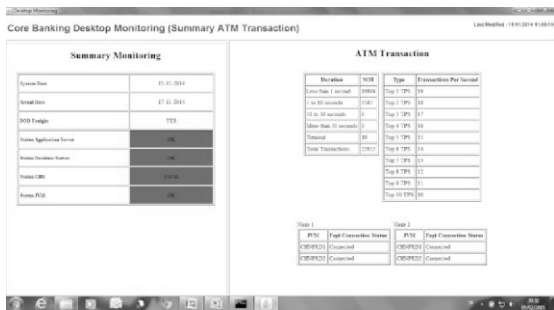


Gambar IV-15 Tampilan Halaman Status Database

Dari gambar diatas menampilkan informasi summary monitoring untuk menandakan bahwa statusnya normal, warning atau fatal dan menampilkan informasi database secara detail

4.2.2.3 Tampilan Halaman Status ATM

Tampilan dibuat ringkas serta mudah dipahami, selain itu bertujuan mengutamakan fungsi monitoring.

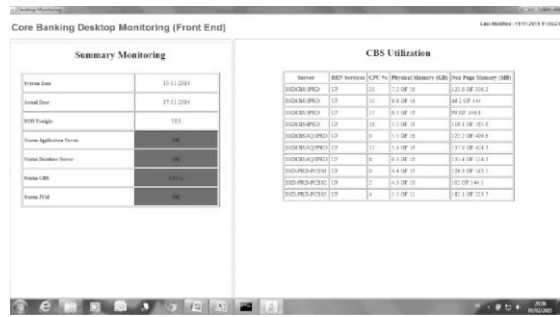


Gambar IV-16 Tampilan Halaman Status ATM

Dari gambar diatas menampilkan informasi summary monitoring untuk menandakan bahwa statusnya normal, warning atau fatal dan menampilkan informasi ATM secara detail

4.2.2.4 Tampilan Halaman Status Utilisasi Cabang Centralize

Tampilan dibuat ringkas serta mudah dipahami, selain itu bertujuan mengutamakan fungsi monitoring.

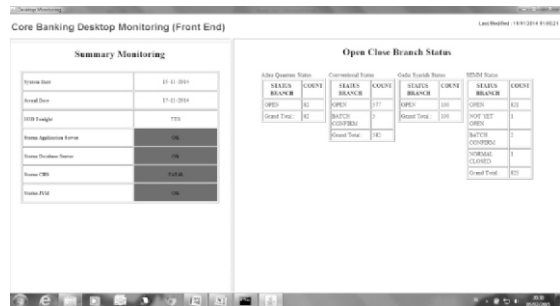


Gambar IV-17 Tampilan Halaman Status Utilisasi Cabang Centralize

Dari gambar diatas menampilkan informasi summary monitoring untuk menandakan bahwa statusnya normal, warning atau fatal dan menampilkan informasi utilisasi cabang centralize secara detail

4.2.2.5 Tampilan Halaman Status Cabang Buka dan Tutup

Tampilan dibuat ringkas serta mudah dipahami, selain itu bertujuan mengutamakan fungsi monitoring.

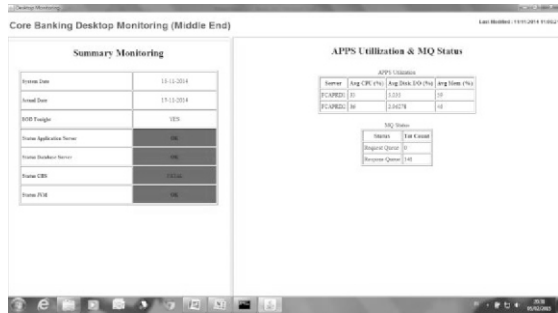


Gambar IV-18 Tampilan Halaman Status Cabang Buka dan Tutup

Dari gambar diatas menampilkan informasi summary monitoring untuk menandakan bahwa statusnya normal, warning atau fatal dan menampilkan informasi status cabang buka dan tutup secara detail

4.2.2.6 Tampilan Halaman Status Utilisasi aplikasi dan Mq

Tampilan dibuat ringkas serta mudah dipahami, selain itu bertujuan mengutamakan fungsi monitoring.



Gambar IV-19 Tampilan Halaman Status Utilisasi aplikasi dan Mq

Dari gambar diatas menampilkan informasi summary monitoring untuk menandakan bahwa statusnya normal, warning atau fatal dan menampilkan informasi status utilisasi aplikasi dan MQ secara detail

4.3 Pengujian Kualitas

Berdasarkan analisis data yang diperoleh dari kuesioner, berikut rekapitulasi hasil pengujian kualitas berdasarkan empat aspek kualitas perangkat lunak menurut ISO 9126:

Tabel IV-2 Hasil Pengujian ISO 9126

Aspek	Skor Aktual	Skor Ideal	% Skor Aktual	Kriteria
Functionality	116	125	92,8%	Sangat Baik
Reliability	102	125	81,60%	Baik
Usability	176	200	88,00%	Sangat Baik
Efficiency	47	50	94%	Sangat Baik
Total	441	500	88,20%	Sangat Baik

Berdasarkan tabel di atas dapat disimpulkan bahwa tingkat kualitas monitoring server ini secara keseluruhan dalam kriteria Sangat Baik, dengan persentase 88,20%. Aspek kualitas tertinggi adalah berdasarkan aspek *Efficiency* dengan persentase sebesar 94%, sedangkan aspek kualitas terendah adalah dari aspek *Reliability* dengan persentase sebesar 81,60%.

4.4 Kesimpulan Hasil Pengujian Kualitas Dan Pembuktian Hipotesis

Dari beberapa pengujian yang dilakukan maka hasil yang di dapatkan dapat dilihat pada tabel berikut :

Tabel IV-2 Hasil Pengujian Kualitas

No	Teknik pengujian	keberhasilan	Kriteria
1	Forum Group Discussion (FGD)	100 %	Diterima keseluruhan
2	ISO 9126	88,20 %	Sangat Baik

Berdasarkan hasil pengujian, pengujian untuk hipotesis kedua dalam penelitian ini dibuktikan bahwa kualitas monitoring server yang dihasilkan jika diukur mengadaptsi 4 karakteristik berdasarkan kualitas perangkat lunak model ISO 9126 mencapai harapan semula yaitu Sangat Baik. Hasil akhir kualitas perangkat lunak menurut responden adalah Sangat Baik dengan persentase tanggapan responden rata-rata sebesar 88,20%.

4.5 Rencana Implementasi Sistem

Rencana implementasi sistem merupakan tahap awal dari penerapan sistem dan tujuan dari kegiatan implementasinya adalah agar sistem dapat beroperasi sesuai dengan yang diharapkan. Dapat dijelaskan bahwa implementasi sistem merupakan tahap akhir dalam siklus pengembangan sistem menggunakan metode *Waterfall*.

4.5.1 Tahapan Implementasi Sistem

Dalam proses implementasi sistem monitoring server core banking diperlukan beberapa tahapan perencanaan untuk implementasi sistem dan penerapan sistem sistem monitoring server core banking. Tahapan tersebut adalah sebagai berikut:

Tabel IV-4 Rencana Implementasi Sistem

No	Kegiatan	Bulan 1				Bulan 2			
		1	2	3	4	1	2	3	4
1	Pengadaan Hardware dan Software								

2	Instalasi Hardware dan Software								
3	Pemilihan Operator								
4	Pelatihan Pengguna								
5	Sosialisasi Kepada Pimpinan								
6	Ujicoba Sistem								
7	Evaluasi dan Perbaikan Sistem								

4.5.2 Kegiatan Implementasi Sistem

Langkah-langkah kegiatan implementasi sistem adalah sebagai berikut:

1. **Pengadaan Hardware Dan Software.** Hardware dan yang software yang diperlukan untuk menunjang sistem ini adalah:
 - a. Hardware berupa PC Server, yang akan diinstal sistem operasi server. Spesifikasi hardware PC server tidak dibahas dalam penelitian ini dikarenakan peneliti tidak melakukan observasi terhadap aspek biaya dari obyek penelitian.
 - b. Software sistem operasi Server, termasuk didalamnya adalah aplikasi Web Server dan Database Server. Sistem operasi yang digunakan adalah Ubuntu Server versi LTS (*Long Time Support*), Web Server menggunakan Apache dan Database Server menggunakan MySQL. Karena semua software tersebut merupakan Free/Open Source Software, maka tidak diperlukan biaya lisensi untuk menggunakannya.
2. **Instalasi Hardware dan Software.** Setelah hardware untuk Server dan software-nya tersedia maka selanjutnya adalah proses instalasi software pada PC server. Instalasi yang dilakukan meliputi: instalasi sistem operasi server, instalasi web server dan database server serta instalasi aplikasi sistem informasi manajemen kepegawaian yang sudah dibangun.
3. **Pemilihan Operator.** Salah satu faktor yang mempunyai peranan penting dalam penerapan

sistem di Danamon berupa aplikasi sistem monitoring, yaitu operator. Untuk Administrator sistem sudah ada, yaitu kepala IT Engineering yang selama ini menangani software dan hardware. Operator yang dimaksudkan adalah operator yang akan memantau aplikasi monitoring, apakah akan ditentukan berdasarkan struktur organisasi yang sudah ada di IT Danamon atau melakukan penambahan personalia tertentu. Peranan operator adalah memantau status server, sehingga hal ini menjadi sangat penting.

4. **Pelatihan Pengguna.** Pengguna atau *user* yang telah dipilih akan mengikuti pelatihan agar memahami sistem yang baru tersebut dan dapat menjalankannya dengan baik dan tepat. Pelatihan ini tidak hanya meningkatkan keahlian/ketrampilan pengguna, namun juga memudahkan penerimaan pengguna terhadap sistem baru.
5. **Sosialisasi Kepada Pimpinan.** Kegiatan sosialisasi dilakukan untuk memberikan informasi tentang keberadaan sistem monitoring kepada semua pimpinan yang merupakan pengguna dari aplikasi tersebut. Dalam kegiatan sosialisasi dilakukan dengan memberikan surat edaran bahwa aplikasi sudah siap dipergunakan dan disertakan juga panduan penggunaan aplikasi.
6. **Uji Coba Sistem Baru.** Pengujian sistem baru harus dilakukan untuk mengetahui apakah sistem yang baru tersebut dapat berjalan sesuai dengan yang direncanakan atau tidak.
7. **Evaluasi Sistem Baru Dan Perbaikan.** Kegiatan evaluasi ini bertujuan untuk mengetahui hasil dari sistem yang baru dibandingkan dengan hasil yang dicapai oleh sistem lama.

V kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan uraian pada bab-bab sebelumnya, maka dapat disampaikan beberapa kesimpulan sebagai berikut:

1. Model analisis, perancangan dan implementasi sistem monitoring server core banking dapat berfungsi menyediakan informasi status server dengan menggunakan metode pengembangan model *waterfall*. Kesimpulan ini berdasarkan hasil pengujian dengan pendekatan *blackbox* yang menggunakan *Focus Group Discussion* yang telah dilaksanakan. Semua responden sebagai informan dalam penelitian menyatakan semua spesifikasi kebutuhan fungsional untuk pengguna dan fungsi sistem secara keseluruhan dalam menyediakan informasi status sever core banking dapat disetujui.

2. Tingkat kualitas sistem monitoring server core banking pada Bank Danamon Indonesia yang dihasilkan mendadaptasi berdasarkan empat karakteristik model ISO 9126, yaitu: *functionality*, *reliability*, *usability*, dan *efficiency* dapat mencapai kriteria Sangat Baik.

5.2 Saran

Berdasarkan hasil penelitian, implikasi dan kesimpulan, selanjutnya peneliti dapat memberikan saran yaitu penelitian berkaitan dengan pengembangan sistem monitoring core banking ini dapat diperluas lagi ruang lingkup penelitiannya. Penelitian selanjutnya dapat dikembangkan bukan hanya untuk perangkat komputer/PC yang menggunakan windows 7/XP akan tetapi dapat dikembangkan untuk sistem operasi mobile seperti Android, Blackberry, Windows Phone dan iOS Apple agar dapat mengakomodir semua kebutuhan user. Untuk rencana implementasi aplikasi ini akan diinstall di server yang telah ada bersama dengan aplikasi core banking lainnya. Untuk sistem monitoring sever core banking ini diperlukan operator yang melakukan monitoring di aplikasi. Untuk rencana implementasi aplikasi ini akan diinstall di server yang telah ada bersama dengan aplikasi akademik lainnya. Untuk sistem monitoring ini diperlukan operator yang melakukan monitoring di aplikasi. Dimana operator ini juga bertugas untuk mengawasi dan mengawal jalannya *sistem monitoring core banking* agar sistem dapat berjalan dengan baik. Selanjutnya perlu diadakan sosialisasi tentang keberadaan aplikasi sistem monitoring sever core banking kepada semua pengguna yaitu IT.

DAFTAR PUSTAKA

- [1] Badger, Michael, *Zennos Core Network And System Monitoring*, Packt Publishing, Ltd, Birmngham, 2008
- [2] Pressman, S, Roger. *Software Engineering: A Practitioner's Approach, 7th ed.* Dialihbahasakan oleh Adi Nugroho, J, Leopold Nikijuluw George dan et.al. Yogyakarta: ANDI, 2012.
- [3] Nova, Kurniawan Dwi. *Implementasi Sistem Monitor Server Berbasis WEB Dengan Memanfaatkan SNMP (Simple Network Management Protocol) Dengan Metode Prototyping.* Skripsi, Fakultas Ilmu Komputer, 2012.
- [4] Nurfaiza, Riza, Winarno, Idris. *Aplikasi Monitoring Proses pada Komputer Server dengan Menggunakan Java.* Surabaya: ITS, 2010
- [5] Holis, Ahmad. *Perancangan Dan Implementasi Monitoring Jaringan Lokal Menggunakan Sistem Kerja Backdoor.* Yogyakarta: Universitas Islam Negeri Sunan Kalijaga, 2011
- [6] Moedjiono. *Pedoman Penelitian, Penyusunan dan Penilaian Tesis (V.5).* Jakarta: Universitas Budi Luhur, 2012. <http://pascasarjana.budiluhur.ac.id>
- [7] Nasution, S. *Metode Research.* Jakarta: Bumi Aksara, 2009.
- [8] Al-Qutaish, Rafa, E. "Quality Models in Software Engineering Literature: An Analytical and Comparative Study." *Journal of American Science* 6 (2010): 166-175.