

Pengembangan Sistem *Monitoring* Server Hosting cPanel Berbasis Web Menggunakan *Script Monitoring* Pada PT. Tujuh Ion Indonesia

Purwanto*¹, Sufajar Butsianto², Edora³

^{1,2}Teknik Informatika, Universitas Pelita Bangsa, Bekasi

e-mail: *¹purwanto94@mhs.pelitabangsa.ac.id, ²sufajar@pelitabangsa.ac.id,
³edora@pelitabangsa.ac.id

Abstrak

PT. Tujuh Ion Indonesia, sebuah perusahaan teknologi yang fokus pada layanan hosting dan domain, sangat bergantung pada kinerja server untuk solusi hosting mereka. Untuk menjaga stabilitas server, pemantauan baik dari perangkat keras maupun perangkat lunak sangat penting. Penelitian ini bertujuan mengembangkan sistem pemantauan server berbasis web menggunakan skrip. Sistem ini memungkinkan pemantauan berbagai aspek kritis server, seperti penggunaan CPU, RAM, hard disk, dan aplikasi pendukung yang penting dalam waktu nyata. Data ditampilkan melalui antarmuka web yang mudah digunakan. Hasil penelitian menunjukkan efektivitas sistem dalam pemantauan server secara real-time. Pengguna dapat melacak kinerja server, memperoleh informasi yang akurat, dan dengan cepat mengidentifikasi masalah. Selain itu, sistem ini mendukung pemberitahuan melalui email dan Telegram untuk batasan tertentu atau masalah yang muncul. Secara keseluruhan, sistem pemantauan server berbasis web dengan skrip ini memberikan solusi efektif bagi PT. Tujuh Ion Indonesia dalam menjaga stabilitas kinerja server. Ini meningkatkan efisiensi pemantauan server dan memudahkan respons cepat terhadap masalah yang muncul.

Kata kunci: Sistem monitoring, Server Hosting, Web, Script Monitoring

Abstract

PT. Tujuh Ion Indonesia, a technology company that specializes in hosting and domain services, heavily relies on server performance for their hosting solutions. To maintain server stability, both hardware and software monitoring are crucial. This research aims to develop a web-based server hosting monitoring system using scripting. The system allows monitoring critical server aspects, such as CPU, RAM, hard disk usage, and essential supporting applications in real-time. Data is displayed through an intuitive web interface. The study demonstrates the system's effectiveness in real-time server monitoring. Users can track server performance, obtain accurate information, and swiftly identify issues. Additionally, the system supports notifications through email and Telegram for threshold breaches or problems. In conclusion, this web-based server hosting monitoring system with scripting provides an effective solution for PT. Tujuh Ion Indonesia to maintain server performance stability. It enhances server monitoring efficiency and facilitates responsive actions to emerging issues.

Keywords: Monitoring System, Server Hosting, Web, Monitoring Scripts



1. PENDAHULUAN

Di era yang semakin canggih saat ini, layanan hosting dan domain semakin diminati oleh masyarakat. Permintaan akan platform online yang stabil dan dapat diandalkan terus meningkat. PT. Tujuh Ion Indonesia, sebagai penyedia layanan hosting dan domain, mengambil manfaat dari tren ini. Dengan bertambahnya pelanggan yang berlangganan hosting, jumlah server juga meningkat. Namun, dengan pertumbuhan tersebut, PT. Tujuh Ion Indonesia harus memastikan kinerja server tetap optimal. Ini melibatkan pemantauan yang cermat, pemeliharaan rutin, dan peningkatan kapasitas sesuai kebutuhan. Dengan menjaga kinerja server tetap optimal, PT. Tujuh Ion Indonesia dapat menyediakan layanan hosting dan domain yang handal bagi pelanggannya.

Monitoring adalah proses mengumpulkan informasi dengan usaha minimum untuk membuat keputusan pengendalian pada waktu yang tepat [1]. *Monitoring* server adalah aspek penting dalam menjaga stabilitas dan kinerja server. Kegiatan pemantauan dilakukan untuk memastikan layanan server berjalan normal. Proses pemantauan mencakup pemantauan perangkat keras dan perangkat lunak pada server. Namun, gangguan server bisa terjadi tiba-tiba, baik akibat faktor internal maupun eksternal. Menurut Bapak Ngubaedilah, Manajer Server Administrator PT. Tujuh Ion Indonesia, sistem pemantauan saat ini dianggap kurang efektif karena tidak memiliki fitur *notifikasi*, yang mengakibatkan keterlambatan dalam menangani masalah server. Hal ini dapat merugikan klien jika server hosting mati dan membuat situs web klien menjadi tidak dapat diakses.

Sistem *monitoring* yang digunakan saat ini adalah sistem *monitoring* berbasis web. Namun, sistem ini hanya menampilkan informasi penting seperti status *uptime*, penggunaan *CPU*, dan antrian email. Meskipun parameter tersebut penting, masih banyak faktor lain yang perlu dipantau, seperti penggunaan *CPU*, penggunaan memori, ruang disk, dan aplikasi penting dalam server hosting. Salah satu kelemahannya adalah tidak adanya fitur *notifikasi*, yang memungkinkan *server administrator* menerima informasi segera tentang masalah atau perubahan pada server. Tanpa fitur notifikasi, *server administrator* PT. Tujuh Ion Indonesia harus secara manual memeriksa sistem pemantauan secara berkala, yang mengakibatkan keterlambatan dalam mendeteksi dan menangani masalah.

Untuk mengatasi masalah tersebut, maka perlu dibuat sistem *monitoring* server dengan fitur *notifikasi* serta menampilkan statistik server penting seperti penggunaan *CPU*, penggunaan *memory*, ruang disk, dan layanan aplikasi krusial. Sistem *monitoring* server berbasis web menggunakan bahasa pemrograman PHP dan MySQL, memungkinkan akses administrator dari berbagai perangkat yang terhubung ke internet. Perkembangan aplikasi berbasis web pesat karena memiliki beberapa kelebihan jika dibanding aplikasi berbasis desktop, antara lain realtime-update, bisa diakses darimana saja, data tersimpan pada server sehingga akses data bisa diatur sesuai dengan kebutuhan, dan *cross platform* yang artinya dapat diakses melalui perangkat komputer dengan berbagai macam sistem operasi [2]. Sistem *monitoring* dengan menggunakan *script monitoring* mampu melakukan proses pemantauan server, menjalankan fitur notifikasi dan *controlling server* dengan tingkat keberhasilan mencapai 100% [3]. Dengan adanya sistem *monitoring* server ini, diharapkan masalah server yang tidak terdeteksi dapat segera diatasi dengan cepat, meningkatkan kinerja server, dan menjaga agar server tetap berjalan dengan optimal.

Yanto, Juli dan kawan-kawan pada tahun 2017 membuat sistem monitoring server menggunakan *Nagios*. Sistem *monitoring* dirancang menggunakan *tools Nagios* yang memiliki fitur notifikasi email, menyalakan service dengan otomatis dan monitoring kode status *HTTPD* pada suatu website. Sistem monitoring *nagios* memonitoring status *host* dan *service* suatu server secara terpusat dalam satu web interface, sehingga memudahkan seorang sistem administrator

dalam memonitoring kondisi keseluruhan server. Dengan adanya fitur notifikasi email jika terjadi kondisi tidak normal pada server maka sistem akan mengirimkan notifikasi ke email. Perbedaan dengan sistem monitoring yang penulis buat adalah sistem monitoring ini hanya mendukung fitur notifikasi ke email saja tidak ke aplikasi chat yang mana akan lebih mudah dicek dari smartphone administrator [4].

Michael, Alehandrew dan kawan-kawan pada tahun 2019 membuat sistem *monitoring* server dengan menggunakan *SNMP*. Fungsi sistem ini adalah melakukan pemantauan segala aktifitas pada server. Sehingga jika terdeteksi masalah pada server maka administrator dapat mengetahuinya dengan cepat. Fitur autostart service juga sangat berguna jika terdapat service pada server yang mati secara tiba-tiba. *Output* dari sistem monitoring ini adalah sebuah web yang menampilkan beberapa informasi mengenai server yang dipantau. Perbedaan dengan sistem monitoring yang penulis buat adalah sistem monitoring ini belum mendukung fitur notifikasi, sehingga pemantauan oleh administrator harus dilakukan secara manual secara berkala melalui web interface [5].

Rahman, Dede dan kawan-kawan pada tahun 2020 membuat sistem monitoring server dengan *Prometheus* dan *grafana*. Dengan menggunakan sistem ini, administrator dapat dengan cepat dan mudah dalam memperoleh informasi tentang kondisi server, dari mana saja dan kapan saja. Sistem ini juga dirancang untuk mengirimkan notifikasi atau peringatan mengenai status CPU, Memory atau service apache dan mysql ada yang down. Melalui Grafana, sistem akan mengirimkan pesan peringatan ke telegram. Perbedaan dengan sistem monitoring yang penulis buat adalah sistem monitoring ini hanya memantau perangkat keras *CPU* dan *RAM* tidak memantau hardisk yang mana merupakan salah satu komponen penting pada server yang berfungsi menyimpan data [6].

2. METODE PENELITIAN

Metode yang digunakan pada penelitian ini adalah metode *Software Development Life Cycle (SDLC)* model *waterfall*. *Software Development Life Cycle*, selanjutnya disingkat dengan *SDLC*, merupakan siklus pengembangan sistem. Pengembangan sistem teknik (*engineering system development*). Fungsi *SDLC* adalah menggambarkan langkah-langkah utama dalam pengembangan sebuah sistem. Secara umum tahapan *SDLC* terbagi dalam lima proses antara lain analisis, desain, implementasi, pengujian, dan pemeliharaan. Setiap proses dalam *SDLC* dapat dijelaskan berdasarkan tujuan (*purpose*) dan hasil kegiatannya (*deliverable*) [7].

Model *waterfall* adalah metode pengembangan perangkat lunak yang digunakan dalam penelitian deskriptif-kualitatif. Model *waterfall* merupakan sebuah model pengembangan perangkat lunak yang menggunakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisa, desain, pengkodean, pengujian dan tahap pendukung/support [8]. Model *Waterfall* adalah proses desain berurutan di mana kemajuan kegiatan dilihat sebagai bentuk aliran dari atas terus ke bawah (seperti air terjun) melalui beberapa fase [9].

Berikut ini penjelasan mengenai tahapan-tahapan metode penelitian:

1. Analisis

Penulis melakukan analisis kebutuhan sistem monitoring server yang melibatkan identifikasi dan pemahaman terhadap kebutuhan dari PT. Tujuh Ion Indonesia. Ini meliputi memahami tujuan sistem, fungsi sistem yang diharapkan dan kinerja yang diinginkan.

2. Desain

Pada tahap desain, penulis membuat rancangan desain sistem monitoring yang akan dibuat yaitu berupa diagram yang masih bersifat sementara. Berikut ini adalah perancangan *software* pada sistem *monitoring* server hosting cpanel pada PT. Tujuh Ion Indonesia.

3. Coding

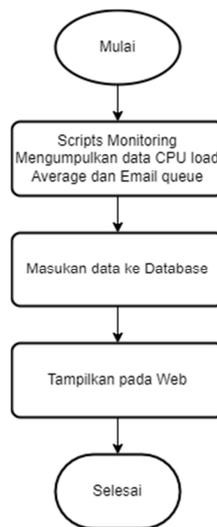
Setelah desain sistem selesai, langkah berikutnya adalah pengkodean. Pada tahap ini, penulis akan menterjemahkan desain sistem menjadi kode program yang dapat dijalankan oleh komputer. Penulis akan menggunakan Bahasa pemrograman *PHP*, *HTML*, *CSS*, dan *Javascript* untuk membuat sistem monitoring berbasis web. Kemudian *script monitoring* akan menggunakan bahasa *bashscript*.

4. Pengujian

Setelah pengkodean selesai, langkah selanjutnya adalah pengujian. Pada tahap ini, penulis akan melakukan serangkaian pengujian terhadap sistem monitoring server yang dikembangkan. Pengujian ini bertujuan untuk memverifikasi bahwa sistem berfungsi sesuai dengan spesifikasi yang telah ditentukan dan memastikan bahwa sistem dapat berfungsi sebagaimana mestinya.

3. ANALISA DAN PERANCANGAN

3.1 Analisis Sistem Berjalan



Gambar 1. Flowchart Sistem Monitoring Berjalan

Gambar 1 menampilkan flowchart sistem monitoring berjalan. Sistem berjalan saat ini merupakan sistem monitoring berbasis web dengan menggunakan *script monitoring*. *Script monitoring* mengumpulkan data monitoring pada server yang kemudian di *insert* ke database *MySQL*. Lalu data monitoring diolah dan ditampilkan pada web. Data monitoring yang dikumpulkan adalah *uptime*, *CPU load*, dan *Email Queue*. Namun sistem ini masih dinilai kurang efektif dikarenakan ada beberapa kekurangan diantaranya adalah sebagai berikut:

1. Masih ada faktor penting lainnya yang tidak termonitor antara lain Penggunaan *CPU*, *Memory*, *Hardisk*, *Network*, *Trafic Email* dan *Service Aplikasi* penting pada server hosting.
2. Tidak adanya fitur notifikasi yang mengakibatkan server administrator harus memeriksa sistem monitoring secara manual memeriksa sistem monitoring secara berkala untuk mendapatkan informasi tentang masalah atau perubahan pada server. Keterbatasan ini menyebabkan keterlambatan dalam mendeteksi dan merespons masalah.

3.2 Perancangan Sistem

1. Flowchart Sistem



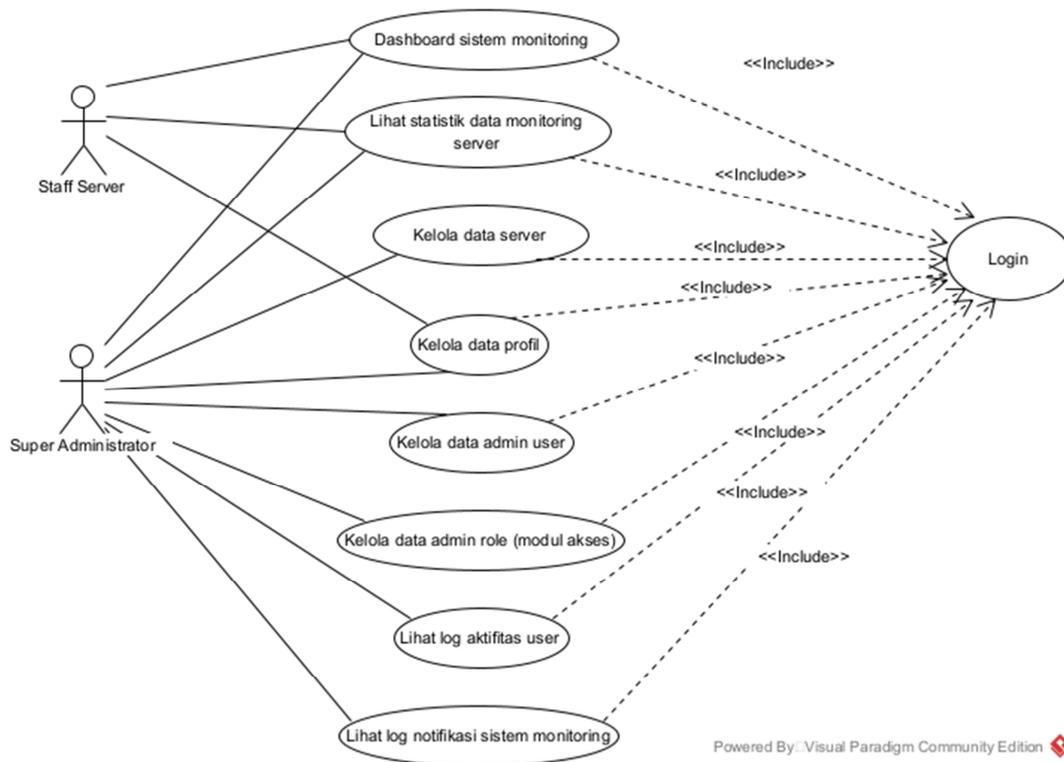
Gambar 2. Flowchart Sistem Monitoring Baru

Flowchart adalah penggambaran dalam bentuk grafik dari langkah-langkah dan urutan prosedur suatu program [10]. Gambar 2 menampilkan flowchart sistem monitoring baru. Sistem monitoring ini merupakan sistem monitoring berbasis web dengan menggunakan script monitoring. Namun data monitoring yang dikumpulkan lebih lengkap yaitu mencakup data statistic *CPU*, *Memory*, *Hardisk*, *Network*, *Traffic Email* dan monitoring *service* aplikasi penting pada server hosting. Kemudian pada sistem monitoring ini sudah ada fitur notifikasi email dan telegram yang akan mengirimkan informasi kepada administrator jika terjadi

masalah pada server, sehingga administrator dapat dengan cepat dan tepat dalam mendeteksi dan merespon masalah pada server.

2. Use Case Diagram

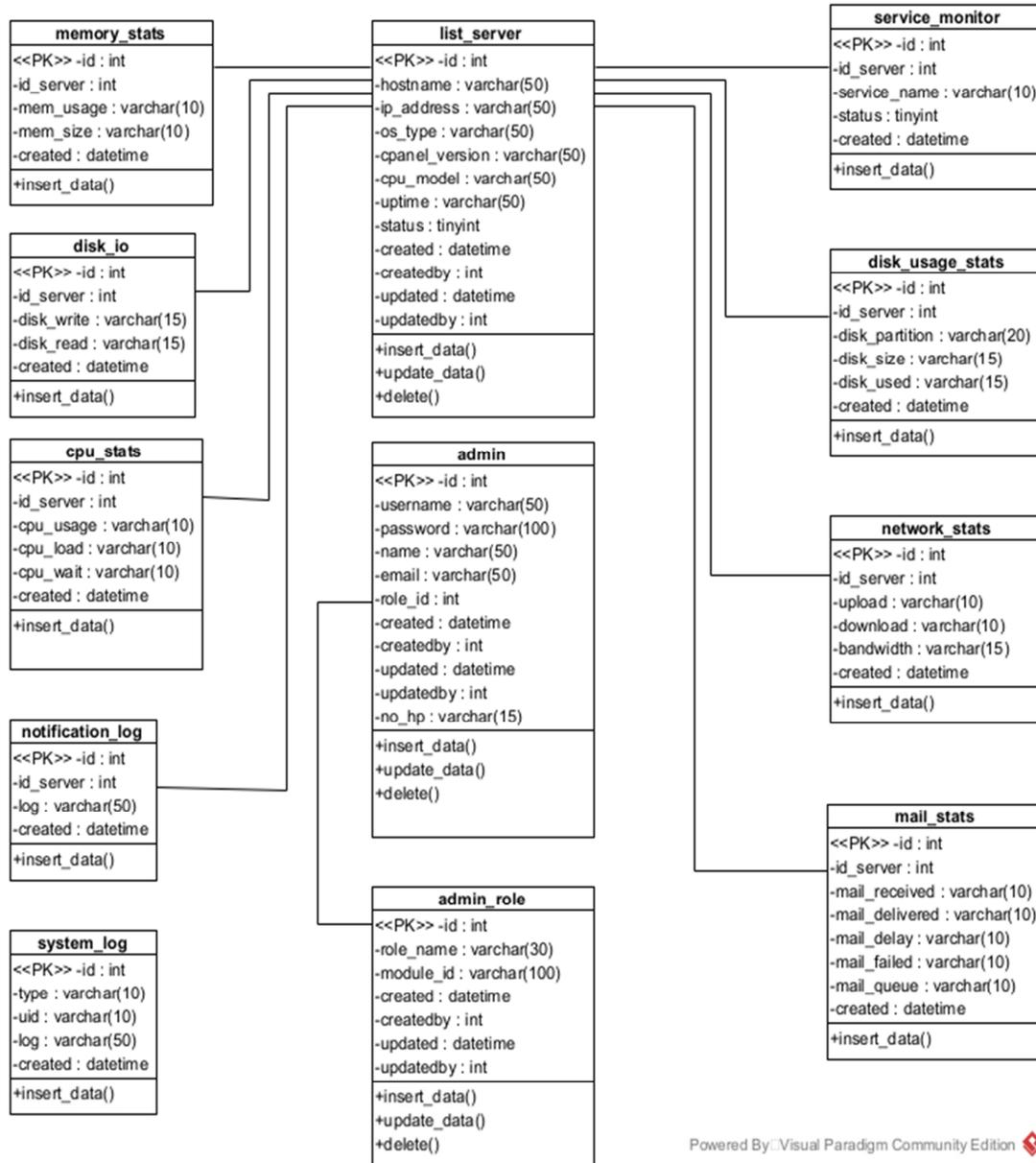
Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat [11]. Gambar 4 menampilkan Use case diagram yang merupakan gambaran aktor yang berinteraksi dalam sistem monitoring server hosting cpanel di PT Tujuh Ion Indonesia.



Gambar 3. Use Case Diagram

3. Class Diagram

Class diagram berfungsi menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem [11]. Gambar 4 menampilkan class diagram yang dibutuhkan pada sistem *monitoring* server hosting cpanel berbasis web menggunakan *script monitoring* pada PT. Tujuh Ion Indonesia.



Gambar 4. Class Diagram

3.3 Kondisi yang Menyebabkan Peringatan

Ada 6 komponen yang dapat memicu terjadinya peringatan. Ketika penggunaan *CPU* diatas 80%, atau beban rata-rata *CPU* atau *CPU load* lebih dari 20 maka peringatan akan terjadi. Ketika penggunaan *RAM* lebih dari 80% maka peringatan akan terjadi. Ketika jumlah email dalam antrian sistem lebih dari 100 email maka peringatan akan terjadi. Ketika *network IO* atau *network upload/download* lebih dari 100 Mbps maka peringatan akan terjadi. Ketika *disk IO* atau *disk read/write* lebih dari 200 Mbps atau penggunaan *disk* lebih dari 90% maka peringatan akan terjadi. Ketika service penting didalam server cpanel antara lain *Apache*, *Mysql*, *Exim* dan *Network* tidak aktif maka peringatan akan terjadi. Detail keadaan yang menyebabkan peringatan dijelaskan pada tabel 1.

Tabel 1. Keadaan yang Menyebabkan Peringatan

No	Komponen	Kondisi
1	CPU	CPU Usage > 80%
		CPU Load > 20
2	RAM	RAM Usage > 80%
3	Email	Email queue > 100
4	Network	Network Upload/Downloads > 100 Mbps
5	Disk	Disk Usage > 95%
		Disk IO > 200 Mbps
6	Application	Apache DOWN
		Mysql DOWN
		Exim DOWN
		Network DOWN

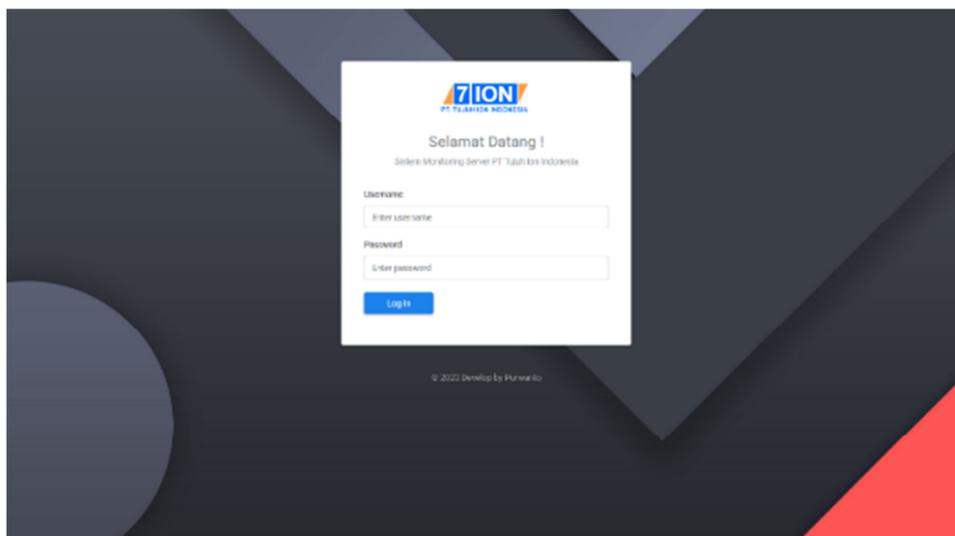
4. PENERAPAN DAN PENGUJIAN SISTEM

4.1 Penerapan Sistem

Penerapan sistem adalah fase penting dalam pengembangan sistem informasi. Fungsi utamanya adalah memungkinkan pengoperasian program aplikasi yang akan digunakan untuk mengelola suatu sistem. Dalam menjalankan proses implementasi, beberapa tindakan perlu diambil. Salah satunya adalah menyusun rencana implementasi (*implementation plan*), yang merupakan langkah awal dalam fase implementasi sistem. Rencana implementasi bertujuan untuk mengatur dan mengontrol biaya serta waktu yang diperlukan dalam membangun sistem tersebut.

1. Tampilan Halaman Login

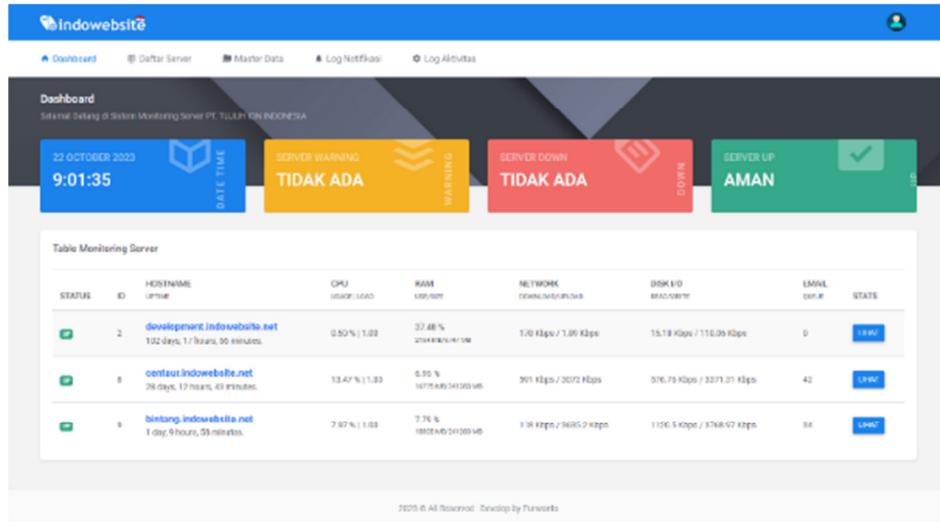
Gambar 5 menampilkan halaman login dari sistem monitoring. Halaman login menampilkan form username dan password serta tombol login untuk masuk ke sistem.



Gambar 5. Halaman Login

2. Tampilan Halaman *Dashboard*

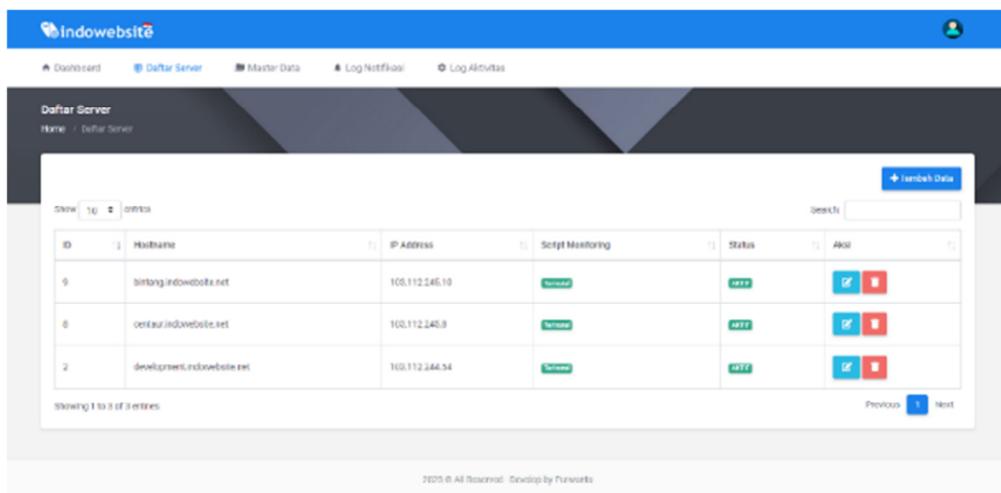
Gambar 6 menampilkan halaman dashboard yang merupakan halaman utama setelah berhasil login. Halaman dashboard menampilkan informasi penting mengenai status server dan juga tabel monitoring server.



Gambar 6. Halaman Dashboard

3. Tampilan Halaman Daftar Server

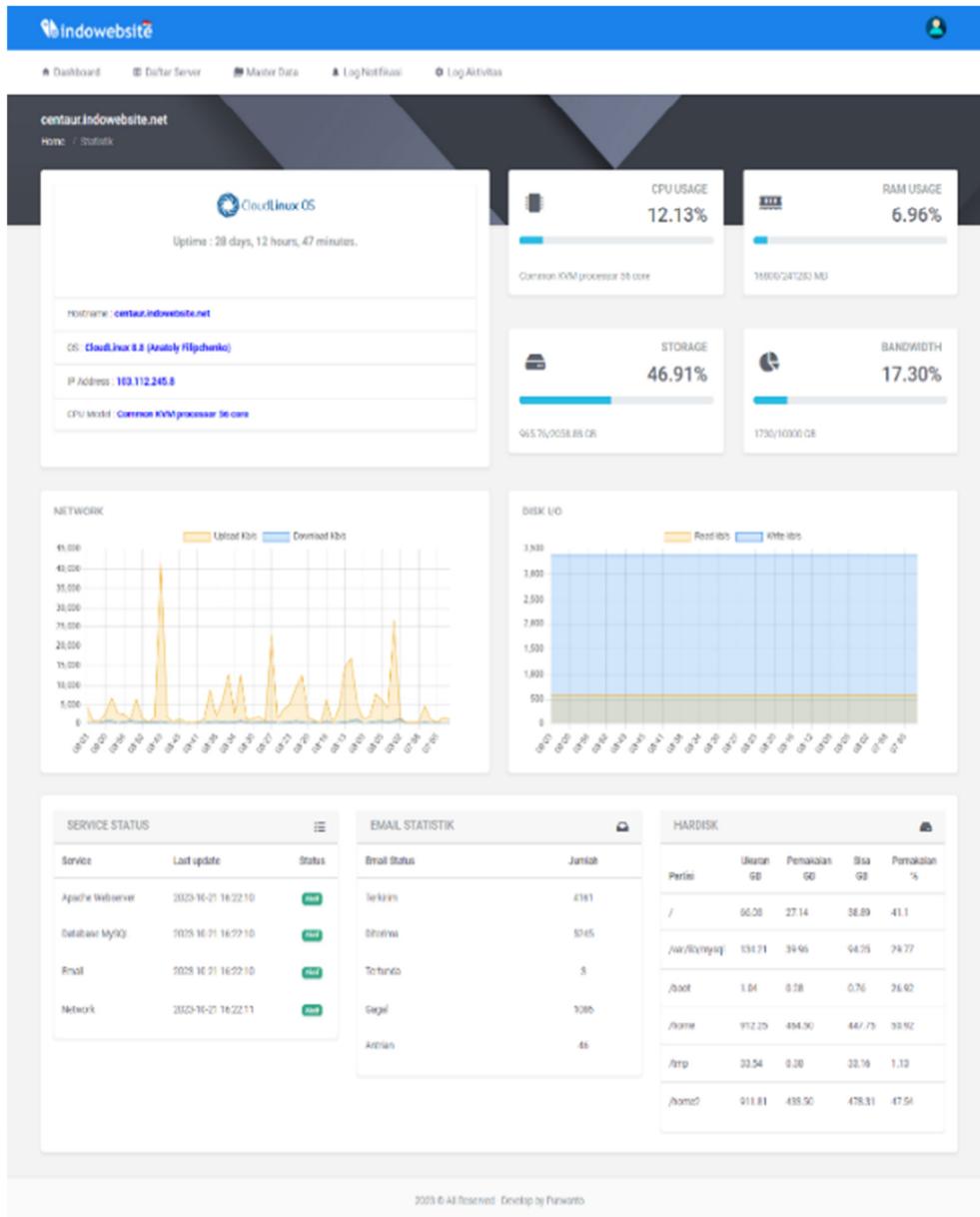
Gambar 7 menampilkan halaman daftar server. Halaman daftar server menampilkan tabel data server yang dimonitoring pada sistem monitoring. Pada halaman ini administrator juga dapat melakukan penambahan server, edit dan juga hapus.



Gambar 7. Halaman Daftar Server

4. Tampilan Halaman Statistik Server

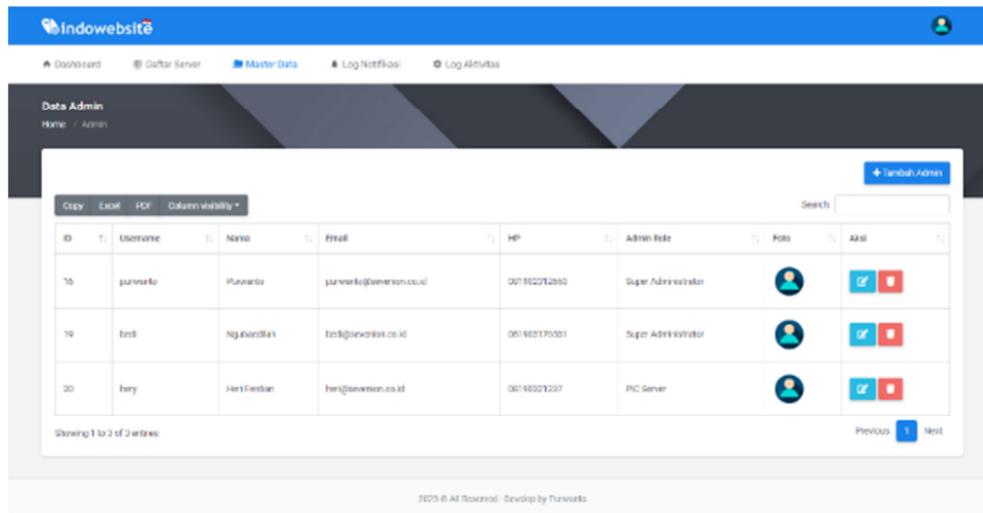
Gambar 8 menampilkan halaman statistik server. Halaman statistik server menampilkan statistik data monitoring server secara lengkap.



Gambar 8. Halaman Statistik Server

5. Tampilan Halaman Data Admin

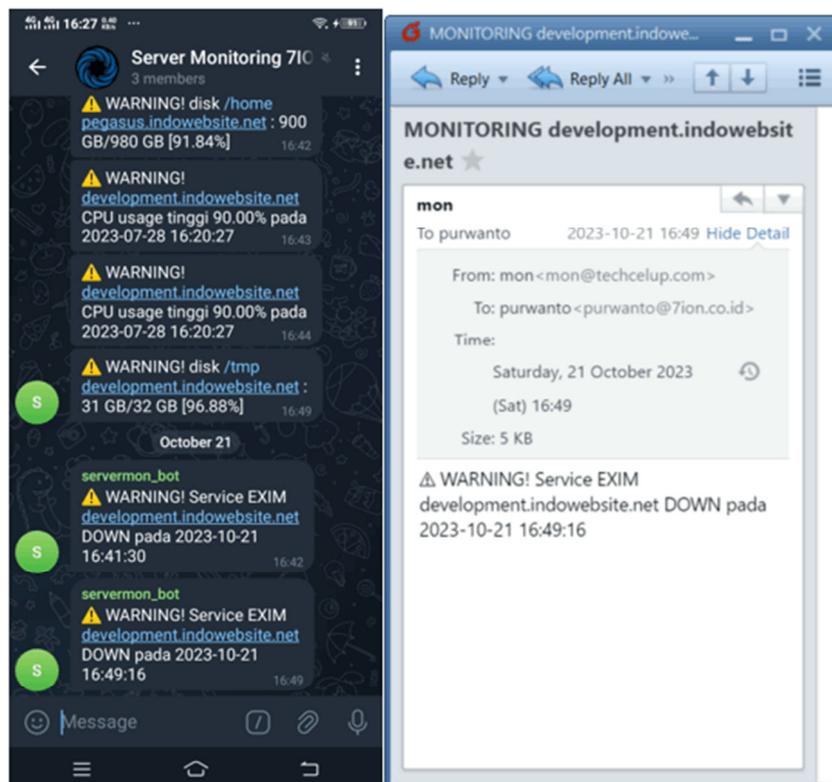
Gambar 9 menampilkan halaman data admin. Halaman data admin menampilkan tabel data admin yang terdaftar. Pada halaman ini juga administrator dapat melakukan penambahan admin baru, edit admin dan hapus admin.



Gambar 9. Halaman Data Admin

6. Notifikasi Telegram dan Email

Gambar 10 menampilkan tampilan notifikasi pada telegram dan email. Notifikasi terkirim ke telegram dan email yang berisi informasi hostname server dan kondisi yang menyebabkan trigger alert serta waktu terjadinya.



Gambar 10. Notifikasi Telegram dan Email

4.2 Pengujian Sistem

Pengujian sistem dilakukan dalam dua tahap yaitu pengujian monitoring pada komponen yang telah ditentukan dan pengujian notification alert ketika kondisi yang menyebabkan peringatan terjadi.

1. Pengujian *Monitoring*

Pengujian *monitoring* dilakukan untuk memastikan bahwa sistem *monitoring* dapat melakukan proses *monitoring*. Hasil pengujian monitoring dijelaskan pada tabel 2. Berdasarkan pengujian yang sudah dilakukan, data monitoring *CPU*, *memory*, *disk*, *network*, *email* dan aplikasi berhasil dibaca dan ditampilkan oleh aplikasi.

Tabel 2. Hasil Pengujian Monitoring

No	Komponen	Pengujian Data Monitoring	Status	Keterangan
1	CPU	CPU Usage	OK	Aplikasi mampu membaca data pemakaian CPU
		CPU load	OK	Aplikasi mampu membaca data beban atau load CPU
2	Memory	Memory Usage	OK	Aplikasi mampu membaca data pemakaian RAM/Memory
3	Disk	Disk Usage	OK	Aplikasi mampu membaca data pemakaian hardisk
		Disk IO	OK	Aplikasi mampu membaca data Disk IO atau kecepatan read/write hardisk
4	Network	Network Status	OK	Aplikasi mampu membaca data status jaringan
		Network IO	OK	Aplikasi mampu membaca data network IO atau kecepatan upload download
5	Email	Email Delivered	OK	Aplikasi mampu membaca data jumlah email terkirim
		Email Received	OK	Aplikasi mampu membaca data jumlah email diterima
		Email Delay	OK	Aplikasi mampu membaca data jumlah email tertunda
		Email Failed	OK	Aplikasi mampu membaca data jumlah email gagal
		Email Queue	OK	Aplikasi mampu membaca data jumlah email dalam antrian
6	Application	Apache	OK	Aplikasi mampu membaca status aplikasi webserver Apache atau HTTPD
		MySQL	OK	Aplikasi mampu membaca status database MySQL
		Exim	OK	Aplikasi mampu membaca status aplikasi email atau exim
		Network	OK	Aplikasi mampu membaca status service network

2. Pengujian *Notification Alert*

Pengujian *notification alert* dilakukan untuk memastikan bahwa pesan peringatan terkirim ke telegram dan email pada saat keadaan yang menyebabkan peringatan terjadi. Hasil pengujian *notification alert* dijelaskan pada tabel 3. Berdasarkan pengujian yang sudah dilakukan, *notification alert* untuk masing-masing *trigger condition* berhasil terkirim ke email dan *telegram* sesuai dengan yang diharapkan.

Tabel 3. Hasil Pengujian *Notification Alert*

No	Triger condition	Telegram	Email	Keterangan
1	CPU Usage > 80%	OK	OK	Notification alert terkirim ketika trigger terjadi
2	CPU Load > 20	OK	OK	Notification alert terkirim ketika trigger terjadi
3	RAM Usage > 80%	OK	OK	Notification alert terkirim ketika trigger terjadi
4	Email queue > 100	OK	OK	Notification alert terkirim ketika trigger terjadi
5	Network IO > 100 Mbps	OK	OK	Notification alert terkirim ketika trigger terjadi
6	Disk Usage > 95%	OK	OK	Notification alert terkirim ketika trigger terjadi
7	Disk IO > 200 Mbps	OK	OK	Notification alert terkirim ketika trigger terjadi
8	Apache DOWN	OK	OK	Notification alert terkirim ketika trigger terjadi
9	Mysql DOWN	OK	OK	Notification alert terkirim ketika trigger terjadi
10	Exim DOWN	OK	OK	Notification alert terkirim ketika trigger terjadi
11	Network DOWN	OK	OK	Notification alert terkirim ketika trigger terjadi

5. KESIMPULAN

Kesimpulan dari proses pembuatan dan pengujian sistem *monitoring* server hosting cpanel berbasis web menggunakan *script monitoring* pada PT. Tujuh Ion Indonesia adalah aplikasi *monitoring* server mampu melakukan *monitoring* terhadap komponen server hosting cpanel yang telah dipilih untuk dimonitor antara lain *CPU, Memory, Disk, Network, Email* dan aplikasi penting yang ada di server secara *realtime* dan dapat diakses melalui web *browser*. Dengan implementasi sistem berbasis web, memungkinkan pemantauan yang lebih mudah dan dapat diakses dari mana saja dengan banyak jenis *device* yang dapat terhubung dengan internet. Selain itu, pengujian *notification alert* juga berhasil dalam mengirimkan pesan ke telegram dan email ketika *trigger alert condition* terjadi. Dengan adanya sistem *monitoring* ini, diharapkan PT. Tujuh Ion Indonesia dapat meningkatkan respons terhadap masalah pada server hosting cpanel secara lebih efisien, mengurangi potensi *downtime* pada server dan meningkatkan kepuasan pelanggan dengan layanan yang lebih baik.

DAFTAR PUSTAKA

- [1] P. Gudda, *A Guide To Project Monitoring and Evaluation*, Bloomington: Authorhouse, 2011
- [2] A. Solichin, *Pemrograman Web Dengan PHP dan MySQL*, Jakarta: Budi Luhur, 2016
- [3] E. P. Prasetyo, J. D. Irawan dan A. FX, “Rancang Bangun Sistem Monitoring Server Visual Berbasis Web Menggunakan Script Monitoring pada Proxmox Virtual Environment,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, Vol. 6, No. 1, pp. 179-185, 2022.
- [4] J. Yanto dan M. Ruswanda, “Implementasi Sistem Monitoring Server Menggunakan Nagios,” *Jurnal JI-Tech*, Vol. XIII, No. 1, pp. 1-9, 2017
- [5] A. Michael, H. Hermawan dan H. I. Pratiwi, “Sistem Monitoring Server Dengan Menggunakan SNMP,” *Widyakala Journal*, Vol. VI, No. 2, pp. 163-166, 2019
- [6] D. Rahman, H. Amnur dan R. Indri, “Monitoring Server Dengan Prometheus dan Grafana serta Notifikasi Telegram,” *Jurnal Ilmiah Teknologi Sistem Informasi*, Vol. IV, No. 1, pp. 133-138, 2020
- [7] I. G. S. Widharma, “Perancangan Simulasi Sistem Pendaftaran Kursus Berbasis Web dengan Metode SDLC,” *Matrix - Jurnal Manajemen Teknologi dan Informatika*, Vol. VII, No. 2, pp. 38-41, 2017.
- [8] W. S. Dharmawan, D. Purwaningtiyas dan D. Risdiansyah, “Penerapan Metode SDLC Waterfall Dalam Perancangan Sistem Informasi Administrasi Keuangan Berbasis Desktop,” *Jurnal Khatulistiwa Informatika*, Vol. VI, No. 2, pp. 159-167, 2018
- [9] E. Ali, *Rekayasa Perangkat Lunak*, Yogyakarta: CV MFA, 2019
- [10] I. Budiman, S. Saori, R. N. Anwar, Fitriani dan M. Y. Pangestu, “Analisis Pengendalian Mutu di Bidang Industri Makanan (Studi Kasus: UMKM Mochi Kaswari Lampion Kota Sukabumi),” *Jurnal Inovasi Penelitian*, Vol. I, No. 10, pp. 2185-2190, 2021
- [11] R. A. Sukamto dan S. M, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Bandung: Informatika, 2016