

BAB II

TINJAUAN PUSTAKA

Seiring berkembangnya teknologi, *logging* menjadi satu hal yang penting dalam pembangunan aplikasi, terutama di industri. Dari masalah tersebut, mulai muncul berbagai teknologi untuk melakukan *logging* untuk berbagai bahasa pemrograman. Hal ini mendorong dilakukannya penelitian terkait pembangunan aplikasi pemantauan.

Penelitian pertama dilakukan oleh Endang Ray pada tahun 2015. Endang Ray membangun aplikasi untuk melakukan *monitoring* terhadap *server*. Aplikasi ini dibangun dengan basis *website* menggunakan bahasa pemrograman PHP dengan *framework* CodeIgniter. Penelitian ini bertujuan untuk melakukan pemantauan terhadap kondisi *server*. Adapun kondisi *server* yang dipantau meliputi pemeriksaan koneksi *server*, status ketersediaan port, pemakaian CPU dan *memory*, kondisi jaringan, *reboot host*, dan melakukan restart service. Pada penelitian ini, Endang Ray juga memberikan informasi anomali berupa *alerting* melalui notifikasi email [6].

Penelitian kedua dilakukan oleh Monika Davidekova dan Michal Gregus ml pada tahun 2016. Penelitian ini bertujuan untuk membangun arsitektur *logging* yang efektif. Penelitian ini diangkat dengan latar belakang penyimpanan data *log* menjadi sangat besar, sehingga perlu dioptimalisasi penggunaannya. Penelitian ini melakukan perbandingan terhadap dua bahasa pemrograman untuk melakukan *logging* yaitu Java dan C#. Selain itu, penelitian ini juga membahas faktor-faktor penting dalam *logging* [7].

Penelitian ketiga dilakukan oleh Nuraini Sabrina pada tahun 2022. Nuraini Sabrina membangun aplikasi untuk melakukan pemantauan data *log* terhadap transaksi di mesin ATM. Teknologi yang digunakan dalam penelitian ini adalah ElasticSearch, Logstash, dan Kibana. Penelitian ini dikembangkan untuk melakukan penelusuran saat terjadi kesalahan dan/atau kegagalan pada mesin ATM dalam satu *dashboard* [8].

Penelitian keempat dilakukan oleh Guntoro Yudhy Kusuma dan Unan Yusmaniar Oktiawati pada tahun 2022. Penelitian ini memanfaatkan teknologi *monitoring tools* yang sudah ada berupa Grafana dan Prometheus. Penelitian ini menggunakan OpenTelemetry untuk menyimpan data log, kemudian menghubungkannya ke Grafana dan Prometheus untuk visualisasi data. Tujuan dari penelitian ini adalah membangun aplikasi yang mampu memantau performa aplikasi, mendeteksi anomali, dan mengirimkan pemberitahuan [9].

Penelitian kelima dilakukan oleh Noviana Adelita Rahmawan pada tahun 2022 yang mengambil studi kasus PT Panasonic Manufacturing Indonesia (PMI). Dalam penelitian ini, Noviana merancang halaman *dashboard* untuk memantau performa *server* menggunakan Microsoft Excel. Dalam kasus Noviana, proses pengelolaan data *server* menggunakan Microsoft Excel, yang dihubungkan dengan *batch scripting* untuk mengetahui performa *server*. Adapun matriks yang diuji antara lain *CPU utilize*, *RAM utilize*, dan *disk usage*. Dalam penelitian ini, pengambilan data dilakukan secara berkala yaitu empat kali dalam satu hari (pukul 07:00 AM, 11:30 AM, 03:30 PM, dan 03:30 PM) [10].

Perbedaan utama penelitian penulis dengan penelitian lainnya adalah obyek pantau berupa *request log* dari API. Selain itu, penulis berfokus pada pembangunan aplikasi pemantauan dengan tampilan yang *human friendly*. Perbedaan detail penelitian penulis dengan penelitian lainnya dapat dilihat pada Tabel 2.1.

Tabel 2. 1 Perbandingan Penelitian Terdahulu

Judul	Tahun	Obyek Pantau	Teknologi dan/atau bahasa pemrograman	Tujuan	Sumber
Pengembangan Aplikasi Monitoring Server Berbasis Mobile Web dengan Sistem Notifikasi Email	2016	Server	CodeIgniter	Pemeriksaan koneksi <i>server</i> , status ketersediaan port, pemakaian CPU dan memory, pencatatan kondisi jaringan, <i>reboot host, restart service</i> , dan memberikan notifikasi email tentang keadaan jaringan	Endang Ray [6]
<i>Software Application Logging Aspect to Consider by Implementing Knowledge Management</i>	2016	Penyimpanan data log	Java, C#	Menemukan strategi logging yang efektif untuk meningkatkan <i>logging</i>	Monika Davidekova; Michal Gregus ml [7]
Pembangunan Sistem Monitoring Data Log Mesin ATM Menggunakan (ElasticSearch, Logstash, Kibana) dalam Satu Dashboard	2022	Log Transaksi ATM	ElasticSearch, Logstash, Kibana	Integrasi data log secara terpusat, mengetahui kegagalan pada sistem pembayaran ATM	Nuraini Sabrina [8]

Judul	Tahun	Obyek Pantau	Teknologi dan/atau bahasa pemrograman	Tujuan	Sumber
<i>Application Performance Monitoring System Design Using Opentelemetry and Grafana Stack</i>	2022	Performa aplikasi	Opentelemetry, Grafana, Prometheus	Memonitor performa aplikasi, mendeteksi anomali aplikasi, dan mengirimkan pemberitahuan.	Guntoro Yudhy Kusuma; Unan Yusmaniar Oktiawati [9]
Rancang Bangun Halaman <i>Dashboard Monitoring</i> Performa Server Menggunakan Microsoft Excel	2022	Performa server	Batch Script, Microsoft Excel	Melacak metrik kinerja atau performa dari server berupa CPU utilize,	Noviana Adelita Rahmawan [10]
Pembangunan Aplikasi Pemantauan Aktivitas Pengguna Terhadap API (<i>Application Programming Interface</i>) Berbasis Web (Studi Kasus PT XYZ)	2025	API	Django Rest Framework, Vue Js	Melakukan pemantauan aktivitas pengguna terhadap API, kondisi dan performa API dari suatu aplikasi	Abraham Jonathan Hortalanus*

Catatan: * (asterisk) = Penelitian yang dilakukan penulis