

PENGEMBANGAN APLIKASI *MONITORING BANDWIDTH USAGE* BERBASIS *WEBSITE* MENGGUNAKAN DJANGO

Ghanny Erlangga
Sekolah Vokasi
Universitas Gadjah Mada
Yogyakarta, Indonesia
ghanny.e@mail.ugm.ac.id

Unan Y Oktiawati
Sekolah Vokasi
Universitas Gadjah Mada
Yogyakarta, Indonesia
unan.yusmaniar@ugm.ac.id

Irving V Papatungan
Jurusan Informatika
Universitas Islam Indonesia
Yogyakarta, Indonesia
irving@uii.ac.id

Abstract—Stabilitas dalam sebuah jaringan merupakan salah satu faktor penting untuk menjaga kualitas konektivitas. Maka dari itu diperlukan *monitoring* secara berkala untuk mengetahui kondisi terkini secara cepat dan tepat suatu trafik jaringan. Selain itu, pemantauan secara realtime pada perangkat jaringan dapat memudahkan administrator apabila ada kendala dapat langsung diatasi dengan cepat. Belum lagi, sekarang ini banyak Internet Service Provider (ISP) sudah memiliki pelanggan baik dari pemerintahan, perusahaan, perhotelan, maupun Small Office Home Office (SOHO). Makalah ini menyajikan pengembangan sebuah antarmuka berbasis *website* yang dapat melakukan pemantauan penggunaan dan kecepatan *bandwidth* dari tiap pengguna. *Website* dikembangkan menggunakan framework Django, di mana *capture* data menggunakan protokol SNMP yang dimana data tersebut akan disimpan dalam sebuah *database timeseries InfluxDB*. Dari *website* yang dikembangkan administrator mudah memantau segala trafik jaringan dan *bandwidth usage* dari setiap user dan rata-rata penggunaan dalam rentang waktu tertentu maupun realtime. Data yang dihasilkan membentuk *time-series*. Selain itu, sistem ini akan memudahkan management *bandwidth* pada pelanggan yang dikehendaki.

Keywords—*Monitoring Bandwidth, Django, SNMP, InfluxDB*.

I. PENDAHULUAN

Pada era revolusi industri 4.0 ini, perkembangan teknologi yang digunakan manusia pada umumnya mengalami kemajuan terutama di industri bidang jaringan dan telekomunikasi. Akibatnya, pengguna internet baik jaringan kabel maupun nirkabel juga meningkat, terutama yang berjenis SOHO. Dengan kondisi tersebut, perusahaan penyedia layanan internet atau yang biasa disebut Internet Service Provider (ISP) perlu melakukan pemantauan terhadap aktifitas yang dilakukan setiap pelanggannya. Pemantauan yang dimaksud adalah pada penggunaan trafik jaringan agar dapat mengetahui utilitas throughput atau besaran penggunaan *bandwidth* secara aktual dari masing-masing pengguna.

Monitoring atau pemantauan jaringan merupakan proses penghimpunan data pada lalu lintas jaringan dan juga setiap perangkatnya. Nantinya data tersebut akan dianalisis untuk tujuan memaksimalkan setiap sumber daya yang ada pada topologi jaringan komputer. Pada penelitian berjudul “Implementasi Simple Network Management Protocol (SNMP) pada Aplikasi *Monitoring* Jaringan Berbasis *Website*” [1], dilakukan *monitoring* terhadap sebuah jaringan dengan menggunakan protokol SNMP untuk memperoleh data pada tiap perangkat jaringan. Setelah terkumpul pada

sebuah SNMP Manager, data tersebut dibuat grafik dengan menggunakan Round Robin *Database Tool* untuk ditampilkan ke dalam sebuah *website* sehingga akan memudahkan seorang administrator jaringan untuk memantau setiap keadaan dari perangkat jaringan hanya melalui satu interface. Data yang ditampilkan cukup akurat untuk mengatasi gangguan maupun perubahan pada sebuah perangkat.

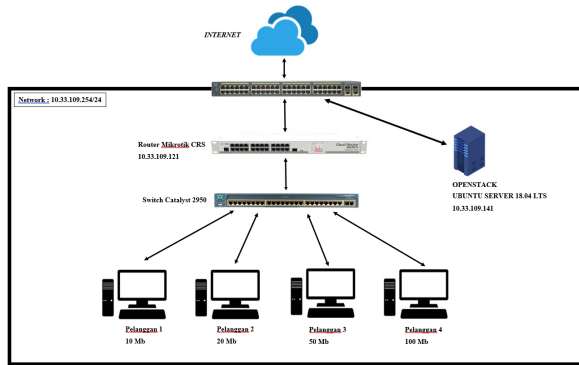
SNMP merupakan sebuah protokol pada jaringan komputer yang memiliki fungsi untuk memonitor dan mengatur sebuah jaringan komputer baik secara remote maupun dalam satu pusat kontrol saja [1]. Protokol SNMP bekerja pada layer 7 dari lapisan OSI. Dalam penggunaannya, protokol ini berkomunikasi menggunakan dua entitas yaitu SNMP Manager dan SNMP Agent. Pada penelitian berjudul “Sistem *Monitoring* Server dengan Menggunakan SNMP” [3] juga menggunakan SNMP protokol untuk melakukan *monitoring* pada beberapa server. Penggunaan protokol SNMP untuk *monitoring* dikarenakan protokol ini dapat manajemen jaringan serta dapat mengatur dan menyimpan segala informasi jaringan yang ada di dalamnya. Dengan begitu maka segala aktivitas di dalam server dapat dipantau dan dapat mendeteksi apabila terjadi permasalahan pada server yang digunakan. Cukup dengan mengaktifkan SNMP Agent pada perangkat router yang terletak pada core jaringan ISP, maka data penggunaan *bandwidth* setiap pengguna dapat terpantau melalui sebuah SNMP Manager.

Untuk mempermudah proses pemantauan *bandwidth usage* tersebut, hasil pemantauan perlu ditampilkan dalam wujud *website* secara real-time. Data *bandwidth usage* yang didapat melalui SNMP dari tiap pelanggan akan diolah pada sebuah sistem backend, disimpan pada sebuah *database timeseries* dan ditampilkan langsung ke *website*. Adanya *database timeseries*, maka data *bandwidth* dapat dilihat secara realtime per detik dan riwayat penggunaan *bandwidth* juga tersimpan. Sistem ini membantu ISP untuk melakukan *monitoring* jaringan dan meningkatkan kualitas trafik dari tiap penggunanya. Selain itu, sistem ini dapat membantu ISP melakukan manajemen penggunaan *bandwidth* berdasarkan kondisi yang dapat dilihat secara realtime, sehingga *bandwidth* yang tersedia dapat dimaksimalkan kepada setiap pelanggan yang memerlukan.

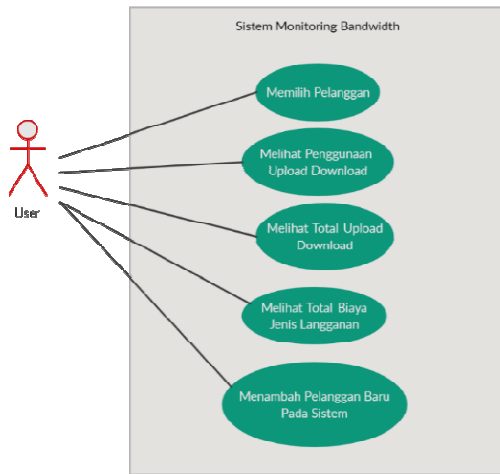
Dalam melakukan pemantauan utilitas pengguna, apabila terdapat pelanggan baru, maka perlu memasukkan Object Identifier (OID) dari pelanggan tersebut ke sistem *monitoring*. Hal ini menjadi sulit dilakukan ketika jumlah pelanggan baru cukup banyak, dan akan memakan waktu lama saat pencarian OID dari setiap pelanggan.

penggunaan *bandwidth* tiap pelanggan, *monitoring* penggunaan *bandwidth* realtime, *monitoring* penggunaan *bandwidth* rentang waktu, dan penambahan pelanggan baru ke sistem *monitoring*. Untuk topologi yang akan digunakan pada penelitian ini dapat dilihat pada Gambar 2. Sedangkan use case diagram atau gambaran fungsional dari sistem yang akan dibuat digambarkan dengan hubungan antara aktor dengan sistem dapat dilihat pada Gambar 3.

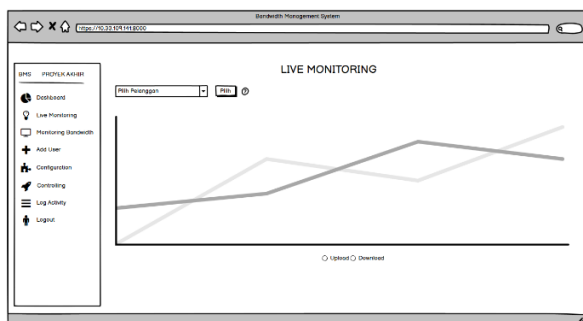
Untuk rancangan dari tampilan yang akan dibuat dalam aplikasi *website* dapat dilihat pada beberapa *wireframe* yang dapat dilihat pada Gambar 4 yaitu *wireframe live monitoring*, Gambar 5 yaitu *wireframe monitoring bandwidth*, dan Gambar 6 yaitu *wireframe add user*.



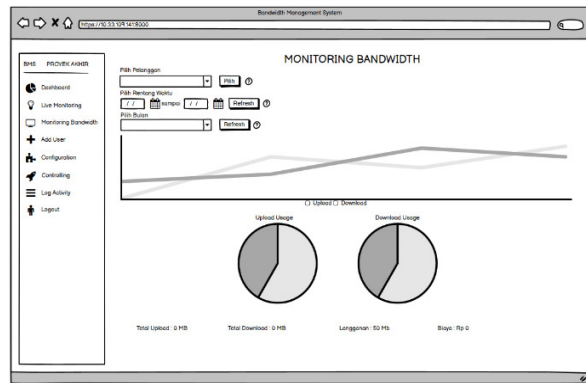
Gambar 2. Rancangan Topologi Jaringan



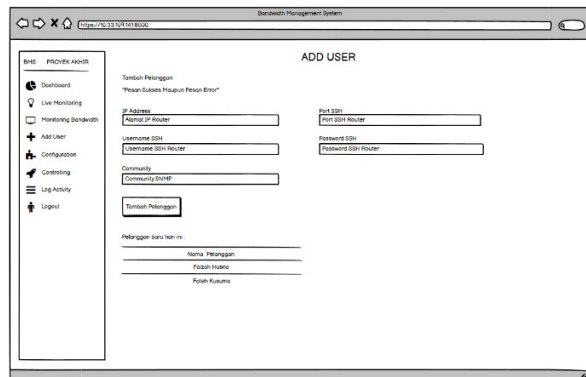
Gambar 3. Use Case Diagram



Gambar 4. Wireframe Live Monitoring



Gambar 5. Wireframe Monitoring Bandwidth



Gambar 6. Wireframe Add User

TABEL 1. PENGALAMATAN PERANGKAT

No	Perangkat	Alamat IP	Prefix
1	Mikrotik Cloud Router Switch	10.33.109.121	24
2	Cisco Switch 2950	-	-
3	Ubuntu Server OpenStack	10.33.109.141	24
4	PC Desktop Client	192.168.100.X	24

E. Tahap Implementasi

Pada tahap ini dilakukan implementasi topologi dengan memasang perangkat sesuai dengan rancangan yang telah dibuat sebelumnya. Juga untuk pengalamatan perangkat keras dapat dilihat pada Tabel 1.

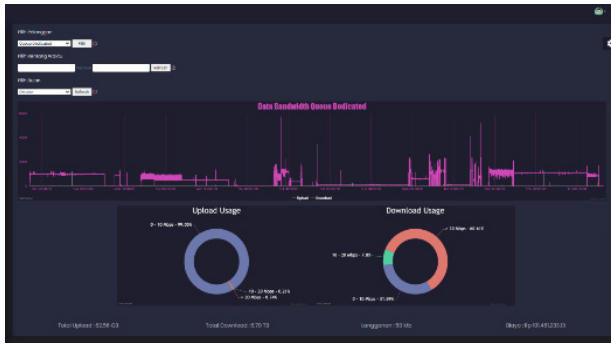
Selain itu, pada tahap ini juga dilakukan instalasi perangkat lunak yang diperlukan untuk pengembangan aplikasi seperti Visual Studio Code sebagai text editor pembuatan program yang diperlukan. Kemudian instalasi sistem operasi Ubuntu pada Openstack yang akan dipasang time-series database InfluxDB, plugin telegraf, serta database sistem mysql. Serta melakukan instalasi framework Django sebagai framework untuk pengembangan *website*.

F. Tahap Pengembangan Sistem Aplikasi

Pada tahap ini dilakukan penulisan kode program untuk membuat backend agar dapat mengakses data dari InfluxDB dan ditampilkan dalam bentuk grafik baik secara realtime maupun dalam jangka waktu tertentu. Serta membuat fungsi otomatis untuk menambahkan pelanggan baru pada *website*, agar pelanggan baru dapat dilakukan *monitoring* pada sistem *website*.



Gambar 7. Halaman *Live Monitoring*



Gambar 8. Halaman *Monitoring Bandwidth*

G. Tahap Pengujian Sistem Aplikasi

Pada tahap ini dilakukan pengujian sistem sesuai topologi yang telah dirancang sebelumnya. Setelah topologi diterapkan, dilakukan pengujian dengan metode *blackbox testing* oleh 15 responden menggunakan kuisioner sesuai dengan skenario yang telah ditentukan. Pengujian yang dilakukan oleh responden berupa pengujian fungsionalitas sistem *website* dan pengisian kuisioner *user accepting test*.

H. Tahap Analisis Hasil

Pada tahap ini dilakukan analisis hasil dari pengujian yang sebelumnya telah dilakukan oleh 15 responden baik secara daring maupun luring. Dari data responden yang menggunakan skala Likert akan disajikan hasil analisis berupa tabel yang berisikan isian dari setiap responden.

III. HASIL DAN PEMBAHASAN

A. Tampilan Antarmuka Sistem Aplikasi

Sistem aplikasi yang dikembangkan menggunakan tampilan template bernama “Django Black Dashboard”, sehingga secara default tampilan bertema gelap dengan warna dasar hitam dan warna sekunder adalah merah jambu. Tampilan antarmuka yang dibuat pada setiap fitur dibuat sesuai dengan *wireframe* yang telah dirancang sebelumnya seperti halaman *Live Monitoring*, *Monitoring Bandwidth*, dan *Add User*. Halaman *Live Monitoring* dapat dilihat pada Gambar 7 dan 8.

B. Hasil Uji Coba Konfigurasi Telegraf dan InfluxDB

Pengujian ini dilakukan dengan tujuan untuk mendapatkan nilai penggunaan *bandwidth* tiap pelanggan menggunakan protokol *SNMP*. Sehingga, dengan membuat file konfigurasi telegraf untuk setiap pelanggan, maka nantinya setiap *capture data bandwidth* yang ditangkap oleh

telegraf akan langsung diteruskan atau disimpan ke dalam *database InfluxDB*. File konfigurasi telegraf dapat dilihat pada Gambar 9 dan hasil pengujian file tersebut dapat dilihat pada Gambar 10.

```
root@ubuntu: /home/ubuntu
GNU nano 2.9.3 /etc/telegraf/telegraf.d/Queue Dedicated.conf
[[inputs.snmp]]
agents = ["10.33.109.121:161"]
version = 1
community = "public"
name = "Queue Dedicated"

[[inputs.snmp.field]]
name = "Upload"
oid = ".1.3.6.1.4.1.14988.1.1.2.1.1.9.1"
[[inputs.snmp.field]]
name = "Download"
oid = ".1.3.6.1.4.1.14988.1.1.2.1.1.8.1"
```

Gambar 9. File Konfigurasi Telegraf

```
root@ubuntu:/etc/telegraf/telegraf.d# telegraf --config Queue\ Dedicated.conf --test
2020-10-16T08:17:48Z I! Starting Telegraf 1.15.3
> Queue\ Dedicated,agent_host=10.33.109.121,host=ubuntu Download=7549267161444,Upload=63
437748981 1628322490000000
root@ubuntu:/etc/telegraf/telegraf.d# date
Fri Oct 16 15:17:49 WIB 2020
root@ubuntu:/etc/telegraf/telegraf.d#
```

Gambar 10. Hasil Pengujian Konfigurasi Telegraf

```
Connected to http://localhost:8086 version 1.8.2
InfluxDB shell version: 1.8.2
> use telegraf
Using database telegraf
> precision rfc3339
> select * from "Queue Dedicated" group by * order by desc limit 10
name: Queue Dedicated
tags: agent_host=10.33.109.121, host=ubuntu
time                Download          Upload
-----
2020-10-16T08:15:14Z 7549267159700    63287747155
2020-10-16T08:15:13Z 7549267159540    63287747103
2020-10-16T08:15:12Z 7549267159540    63287747103
2020-10-16T08:15:11Z 7549267159540    63287747051
2020-10-16T08:15:10Z 7549267159540    63287746999
2020-10-16T08:15:09Z 7549267159540    63287746999
2020-10-16T08:15:08Z 7549267159540    63287746999
2020-10-16T08:15:07Z 7549267159540    63287746999
2020-10-16T08:15:06Z 7549267159540    63287746999
2020-10-16T08:15:05Z 7549267159540    63287746999
> exit
root@ubuntu:/home/ubuntu# date
Fri Oct 16 15:15:18 WIB 2020
root@ubuntu:/home/ubuntu#
```

Gambar 11. Penyimpanan Data *Bandwidth*

Sedangkan untuk hasil pengujian *database InfluxDB* dalam menyimpan data *bandwidth* dapat dilihat pada Gambar 11.

C. Hasil Pengujian Sistem Aplikasi Bandwidth Monitoring

Pengujian fitur sistem aplikasi dilakukan untuk menguji fungsionalitas dari setiap fitur yang terdapat pada sistem aplikasi. Pengujian ini meliputi fitur *live monitoring*, *monitoring bandwidth*, dan *add user*. Hasil pengujian *blackbox testing* bisa dilihat pada Tabel 2.

TABEL 2. TABEL PENGUJIAN BLACKBOX TESTING

Kasus Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menampilkan data <i>monitoring</i> secara <i>realtime</i>	Dapat menampilkan data penggunaan <i>bandwidth</i> pelanggan terpilih secara <i>realtime</i> dan terupdate berkala.	Pengguna dapat memantau penggunaan <i>bandwidth</i> pelanggan secara <i>realtime</i> dan data ter-update tiap detik.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal
Menampilkan data	Dapat menampilkan	Pengguna dapat memantau	<input checked="" type="checkbox"/> Berhasil

<p><i>monitoring</i> berdasarkan waktu.</p>	<p>penggunaan <i>bandwidth</i> pelanggan dalam rentang waktu satu jam terakhir. Menampilkan total <i>upload</i>, <i>download</i>, dan biaya</p>	<p><i>bandwidth</i> pelanggan dalam rentang waktu satu jam terakhir beserta mendapatkan nilai total <i>upload</i>, <i>download</i>, dan biaya.</p>	<p><input type="checkbox"/> Gagal</p>
<p>Menambahakan pelanggan ke dalam sistem <i>monitoring</i></p>	<p>Sistem menambahkan pelanggan baru dan membuat konfigurasi telegraf, serta menyimpan ke MySQL.</p>	<p>Pengguna berhasil menambahkan pelanggan baru dan mendapat notifikasi bahwa berhasil</p>	<p><input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Gagal</p>

D. Hasil Pengujian User Acceptance Test

TABEL 3. HASIL KESELURUHAN KUESIONER

Pertanyaan	Skala					Total	Skor
	SS	S	K S	TS	STS		
Pertanyaan ke-1	14	1	0	0	0	74	98,6%
Pertanyaan ke-2	13	2	0	0	0	73	97,3%
Pertanyaan ke-3	13	2	0	0	0	73	97,3%
Pertanyaan ke-4	13	2	0	0	0	73	97,3%
Pertanyaan ke-5	13	2	0	0	0	73	97,3%
Pertanyaan ke-6	14	1	0	0	0	74	98,6%
Pertanyaan ke-7	13	2	0	0	0	73	97,3%
Pertanyaan ke-8	8	6	1	0	0	67	89,3 %
Pertanyaan ke-9	11	3	1	0	0	70	93,3 %
Pertanyaan ke-10	11	3	1	0	0	70	93,3 %
Pertanyaan ke-11	13	2	0	0	0	73	97,3%
Pertanyaan ke-12	11	4	0	0	0	71	94,6 %

Kuesioner pengujian dilakukan untuk mendapatkan data mengenai kemudahan penggunaan sistem aplikasi pada setiap fiturnya oleh pengguna. Kuesioner pengujian berisikan 12 pertanyaan mengenai berhasil atau tidaknya fitur sistem

aplikasi dan juga mengenai kemudahan penggunaan bagi pengguna. Kuesioner pengujian diisi oleh 15 orang responden yang memiliki pengetahuan tentang penggunaan *bandwidth* pada jaringan. Persentase jawaban yang didapatkan disimpulkan dengan interpretasi skor berdasarkan rumus yang dapat dilihat pada Tabel 3.

IV. KESIMPULAN

Berdasarkan data yang diperoleh serta analisis yang dilakukan dari penelitian Pengembangan Aplikasi *Monitoring Bandwidth Usage* Berbasis *Website* Menggunakan Django, dapat ditarik kesimpulan yaitu *monitoring bandwidth usage* tiap pelanggan pada Mikrotik dapat dilakukan dengan menggunakan perintah GET pada protokol SNMP, yaitu dilakukan GET menggunakan oid pada simple queue Mikrotik. Time series data pada InfluxDB tentang data penggunaan *bandwidth* tiap pelanggan dapat dikumpulkan secara real-time dengan menggunakan plugin telegraf. Berdasarkan uji user acceptance test yang dilakukan oleh responden didapatkan bahwa sistem aplikasi ini sudah cukup memudahkan untuk melakukan pemantauan *bandwidth* baik secara realtime maupun menggunakan rentang waktu tertentu.

DAFTAR PUSTAKA

- [1] Diana, & Maulana, F. (2016). Implementasi protokol SNMP untuk *monitoring* perangkat jaringan. Jurnal Informatika, 126-135
- [2] Fatiyah, A. C., Gumilang, S. F., & Winarsyah, D. (2019). Pengujian Fungsional dan Non Fungsional Aplikasi Web BorongAjaYuk. e-Proceeding of Engineering Vol.6, 8415-8421.
- [3] Michael, A., Hermawan, H., & Pratiwi, H. I. (2019). Sistem *Monitoring* Server Dengan Menggunakan SNMP. Widyakala Journal, 163-166.
- [4] Saputra, R. (2018). PERBANDINGAN INFLUXDB DAN PROMETHEUS UNTUK SISTEM NETWORK *MONITORING*. Bandung: ITB
- [5] Supendi, K. (2016). Using Docker for Django Web Development. Makalah IF2211 Strategi Algoritma.