ANALISIS DATA WEB LOG WEBSITE e-GOVERNMENT PROVINSI BENGKULU UNTUK MENGETAHUI POLA AKSES PENGUNJUNG

[1]Diana, [2]Yovi Apridiansyah
[1,2] Prodi Teknik Informatika Fakultas Teknik Universitas Muhammadiyah Bengkulu
Jl. Bali Kota Bengkulu, telp (0736) 22765 / fax (0736) 26161
Email : diana@umb.ac.id, yoviapridiansyah@gmail.com

Abstract- A web log is a file that records entire accessed activity that is done towards web server. Web log analysis is used to get information that is stored in web server with designing web based application program. Web log data that used is the data in Bengkulu province official website's server. The results showed that the comparison of web log data before filtering using an application program is quite significant, from the amount of data 1.027 reduced to 212. That visitor's access pattern has been through a data cleaning process from irrelevant information such as picture file, multimedia files, request method except GET. The result of web log application program is consisted in these kinds of data, number of visitors, visited pages, status code, accessed file capability, referenced pages, and the browser used by the visitors. Therefore it can help system administrators and web desainer monitor website performance and quality for the sake of future development of e-government website

Key Word : *web log, user access pattern, e-government, Bengkulu*

Abstrak- Web Log adalah file yang mencatat seluruh aktivitas yang diakses yang dilakukan terhadap web server. Analisis web log digunakan untuk mendapatkan informasi yang disimpan di server web dengan merancang program aplikasi berbasis web. Data web log yang digunakan adalah data di server website resmi provinsi Bengkulu. Hasil penelitian menunjukkan bahwa perbandingan data *web log* sebelum penyaringan menggunakan program aplikasi cukup signifikan yaitu dari jumlah data 1,027 berkurang menjadi 212. Hasil ini menunjukkan bahwa lebih dari setengah data *web log* memiliki informasi yang tidak relevan, seperti *file* gambar, *file* multimedia, spasi, *request method* selain GET. Hasil program aplikasi *web log* adalah pola akses pengunjung *website,* diantaranya jumlah pengunjung, halaman yang paling banyak dikunjungi, kapasitas *file* yang diakses, kode status, halaman rujukan, dan *browser* yang digunakan oleh pengunjung. Oleh karena itu dapat membantu administrator sistem dan desainer web dalam memantau kinerja dan kualitas website demi pengembangan website e-government di masa mendatang

Kata Kunci : *web log,* pola akses pengunjung, *e-government,* Bengkulu

1 Pendahuluan

Teknologi informasi dan komunikasi (TIK) merupakan salah satu teknologi yang berkembang dengan sangat pesat. Berbagai keuntungan teknologi informasi khususnya internet banyak diterapkan dalam kehidupan manusia termasuk di bidang pemerintahan (*e-government*). Pemerintahan Provinsi Bengkulu telah menerapkan salah satu wujud nyata dari pengaplikasian *e-government* yang

umum dilaksanakan dan diatur pelaksanaannya di Indonesia yaitu pembuatan situs *web*. Dari *website e-government* tersebut, banyak data yang dapat kita analisis untuk perbaikan *website* itu sendiri. Data yang dapat kita analisis dari *website e-government* yaitu data *web log* yang terdapat pada *server website*.

Web log merupakan data mengenai interaksi masing-masing pengunjung pada setiap *session* [1]. Pengunjung suatu *website* akan berinteraksi melalui serangkaian permintaan. Interaksi ini dilakukan untuk mendapatkan informasi maupun layanan yang diinginkan oleh pengunjung *website*. Semakin banyak kunjungan yang dilakukan pada *website* semakin banyak juga data yang terekam pada *web log*. Ukuran data yang tersimpan pada *web log* tidak hanya dalam ukuran *megabyte*, tetapi juga dalam *terabyte* atau bahkan *petabyte*. Karena jumlah data yang besar dan pentingnya data *web log*, maka analisis tersebut perlu dilakukan sehingga informasi yang tersembunyi pada *web server* dapat digali.

Analisis terhadap *web log* dilakukan dengan merancang program aplikasi. Pada program aplikasi tersebut terdapat tahap *preprocessing* untuk menghilangkan informasi yang tidak relevan pada *web log*. *Preprocessing* data *web log* terdiri atas: *raw web log data*, *data cleaning*, *user identification*, *session identification*, dan *database of clean log* [2]. Hasil dari program analisis *web log* adalah pola akses pengunjung *website*. Pola akses pengunjung tersebut dapat digunakan untuk membantu para *administrator system* dan *desain web* dalam memantau kinerja dan kualitas *website* demi pengembangan *website e-government* dimasa yang akan datang.

II. Landasan Teori

A. *Website e-Government*

Evolusi *e-government* terdiri atas lima tingkatan, yaitu *emerging, enhanced, interactive, transactional*, dan *networked* [3]. *Website e-government* merupakan salah satu strategi di dalam melaksanakan pengembangan *e-government* secara sistematik melalui tahapan yang realistik dan terukur. *Website e-government* merupakan tingkat pertama dalam pengembangan *e-government* di Indonesia yang memiliki sasaran agar masyarakat Indonesia dapat dengan mudah memperoleh akses informasi dan layanan pemerintah daerah, serta ikut berpartisipasi di dalam pengembangan demokrasi di Indonesia [4].

B. *Web Log*

Web log adalah *file* yang mencatat semua akses yang dilakukan terhadap *web server*. Semakin banyak kunjungan yang dilakukan pada *website* semakin banyak juga data yang terekam pada *web log*. Data *web log* memiliki dua format, yaitu *common log format* dan

combined log format [5][6]. .Contoh *web log*:

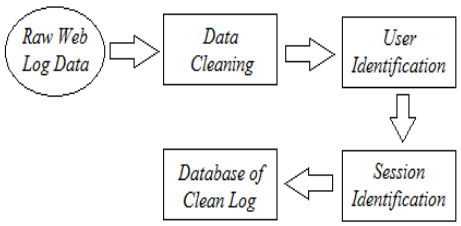
180.242.61.129 - - [11/May/2019:19:17:13 +0700] "GET /wp-includes/js/l10n.js?ver=20101110 HTTP/1.1" 200 308 "http://bengkuluprov.go.id/?page_id=64" "Mozilla/5.0 (Windows NT 5.1; rv:13.0) Gecko/20100101 Firefox/13.0"

Penjelasan struktur *web log*:

1 *180.242.61.129*: *IP address* atau *domain name*.
2 - : *Authuser* (*username* dan *password*), Tanda "-" menunjukkan bahwa autentifikasi *user* tidak digunakan pada saat *user* mengakses suatu direktori pada *web server*, kecuali jika *file* atau direktori yang diakses dilindungi oleh *user autentification*.
3 *[11/May/2019:19:17:13 +0700]*: Tanggal dan waktu akses pengunjung dalam format [*day/month/year:hour:minute:second zone*].
4 *"GET/wp-includes/js/l10n.js?ver=20101110 HTTP/1.1"*: *Request method* yang digunakan untuk mengetahui bagaimana sebuah *web browser* memproses sebuah *request* dan menerima sebuah *response* dari *web server*, misal *GET, POST* atau metode *HEAD* dari *Common Gateway Interface* (CGI).
5 *200*: Kode status saat *client* mengunjungi suatu halaman *website.* Misalnya, 200 adalah "OK" [7].
6 *308*: Kapasitas dokumen yang ditransfer (*Bytes*).
7 *http://bengkuluprov.go.id/?page_id=64*: Baris permintaan yang berasal dari *client*.
8 *"Mozilla/5.0 (Windows NT 5.1; rv:13.0) Gecko/20100101 Firefox/13.0"*: *browser* yang digunakan oleh *client*.

C. *Prepocessing*

Prepocessing digunakan untuk mengekstrak data yang berguna dari *web log* dalam menentukan pola akses pengunjung. Gambar 2 menunjukkan proses penyiapan data *web log* yang terdiri atas: *raw web log data, data cleaning, user identification, session identification,* dan *data base of clean log* [2].



Gambar 1 Proses penyiapan data w*eb log*

Keterangan:

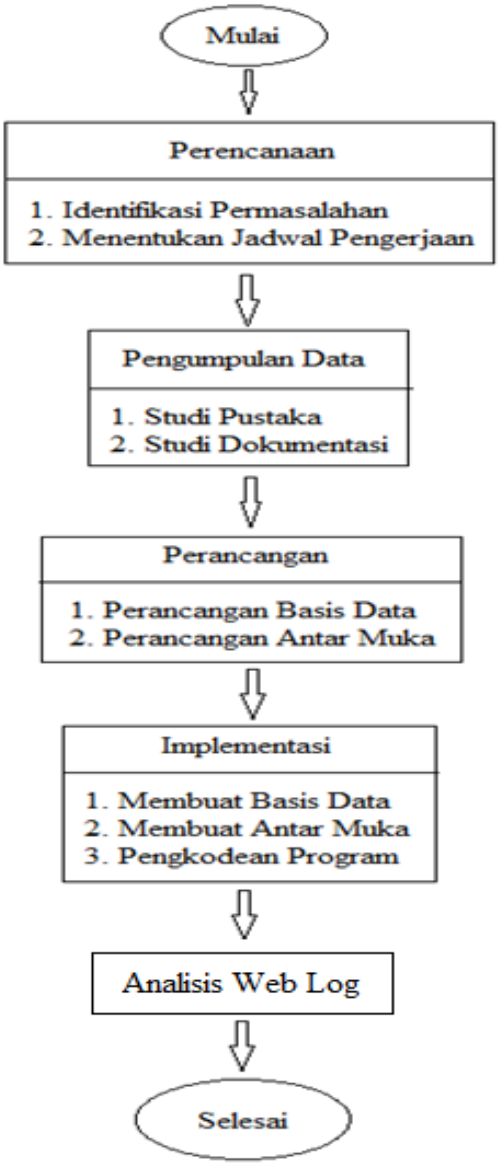
1 *Raw web log data* dilakukan dengan menyiapkan data *web log* asli yang terdapat pada *web server*.
2 *Data cleaning* dilakukan untuk menyaring informasi yang tidak relevan pada *web log* asli seperti: *request* terhadap *file* berekstensi *.jpg, .gif, .png, .ico, .css, .js,* GET/HTTP, navigasi yang dilakukan oleh *spider/robot/crawler*, kode 301, kode 404, kode 500. Informasi tersebut dihilangkan karena merupakan bagian dari suatu *request* terhadap sebuah halaman *web* [8][9].
3 *User identification* adalah proses identifikasi setiap *user* yang mengakses *website* dengan ketentuan: (i) Jika ada alamat *IP* yang baru, maka dianggap sebagai *user* baru, (ii) Jika ada alamat *IP* yang sama, tapi sistem operasi atau *browser* berbeda maka dianggap sebagai *user* baru.
4 *Session identification* dilakukan dengan mengelompokkan *request* dengan ketentuan: (i) Jika ada *user* baru, maka dianggap sebagai *session* baru, (ii) JIka halaman rujukan '-', maka dianggap sebagai *session* baru, dan (iii) Jika *request* melebihi waktu yang ditentukan yaitu 30 menit, maka dianggap sebagai *session* baru.
5 *Database of clean log*, dimana data *web log* siap digunakan untuk proses selanjutnya dalam menentukan pola akses pengunjung *web server*.

D. *Web Server*

Web server merupakan sistem perangkat keras dan piranti lunak yang terhubung pada *World Wide Web*. *Web server* memiliki fungsi menerima permintaan dari pengunjung *web* melalui *browser* dan mengirimkan hasilnya kembali dalam bentuk halaman *web* [10].

III. Metode Penelitian

Penelitian terbagi menjadi beberapa tahapan seperti Gambar 2 di bawah ini.



Gambar 2 Tahapan Penelitian

Berikut ini penjelasan masing-masing tahapan penelitian:

1. Perencanaan

   Pada tahap perencanaan, terdapat dua yaitu mengidentifikasi permasalahan dan menentukan jadwal pengerjaan. Identifikasi permasalahan merupakan tahap awal dalam penelitian ini, dimana penulis akan membuat latar belakan permasalahan, merumuskan permasalahan, menentukan ruang lingkup, mencari tujuan dan manfaat penelitian. Menentukan jadwal pengerjaan merupakan langkah selanjutnya yang dilakukan pada penelitian ini.

2. Pengumpulan Data

   Metode pengumpulan data pada penelitian ini terdiri dari studi pustaka yang dilakukan untuk mendapatkan pemahaman yang komprehensif tentang web log, bahasa pemrograman PHP, basis data MySQL dan studi dokumentasi yang dilakukan untuk mendapatkan data *web log website e-government* Provinsi Bengkulu.

3. Perancangan

   Pada tahap perancangan, penulis akan mulai merancang basis data sistem yang akan dikembangkan, kemudian merancangan tampilan antar muka sistem.

4. Implementasi

   Pada tahap ini dimulai membuat basis data menggunakan *MySQL*, kemudian tampilan aplikasi menggunakan *HTML dan CSS* lalu memulai pengkodean program menggunakan bahasa pemrograman PHP.

5. Analisis *Web Log*

Pada penelitian ini, data *web log* asli tidak bisa langsung digunakan untuk menganalisis *web log* karena data tersebut masih mengandung informasi yang tidak relevan. Pengembangan program aplikasi analisis *web log* menggunakan PHP digunakan untuk melakukan *preprocessing* data *web log* sehingga menghasilkan pola akses pengunjung *website*.

IV. Hasil dan Pembahasan

A. Analisis program aplikasi *web log*

Tahapan ini dilakukan dengan menganalisis data akses pada *web log* yang terdiri atas:

1. *Raw web log data*. Data *web log* yang digunakan pada penelitian ini adalah data *web log* pada *server website* provinsi Bengkulu tanggal 19 Juni 2019 dan merupakan *combined log format*.
2. *Data cleaning*, digunakan untuk menghilangkan data yang tidak relevan pada *web log*. Berikut adalah algoritma yang digunakan pada *data cleaning*.

Langkah 1: *Input* data *web log* (*MySQL*).

Langkah 2: Baca data *web log* yang tersimpan dalam *database MySQL*.

Langkah 3: Hapus data jika terdapat file gambar, file multimedia, spasi, *request method* selain GET.

Langkah 4: Ulangi langkah kedua dan ketiga untuk data selanjutnya sampai seluruh data *web log* selesai dibaca.

Data cleaning dilakukan terhadap *file* gambar, *request method* GET/POST, navigasi yang dilakukan oleh *spider/robot/crawler*, kode 301, kode 404, dan kode 500. Jika penelitian ini dibandingkan dengan penelitian sebelumnya, terdapat perbedaan pada proses *data cleaning* yaitu: (i) Tidak menghilangkan kode *client error* dan *server error*, karena kode kesalahan tersebut dapat digunakan untuk mengetahui *request* yang gagal diproses; (ii) Tidak menghilangkan *request method GET* karena tipe *GET* merepresentasikan halaman yang diakses atau diterima pengunjung.

3. *User identification.* Berikut adalah algoritma untuk menentukan proses identifikasi *user*.

Langkah 1: *Input* data *web log* (*MySQL*).

Langkah 2: Baca data *web log* yang tersimpan dalam *database MySQL*.

Langkah 3: Proses identifikasi user:
Jika ada alamat *IP* yang baru, maka dianggap sebagai *user* baru, atau
Jika ada alamat *IP* yang sama, tapi sistem operasi atau *browser* berbeda, maka dianggap sebagai *user* baru.

Langkah 4: Ulangi langkah kedua dan ketiga untuk data selanjutnya sampai seluruh data *web log* selesai dibaca.

4. *Session identification.* Berikut adalah algoritma untuk mengelompokkan *request.*

Langkah 1: *Input* data *web log* (*MySQL*).

Langkah 2: Baca data *web log* yang tersimpan dalam *database MySQL*.

Langkah 3: Proses identifikasi *session*:
Jika ada *user* baru, maka dianggap sebagai *session* baru, atau
Jika tidak terdapat halaman rujukan, maka dianggap sebagai *session* baru, atau
Jika *request* melebihi waktu yang ditentukan yaitu 30 menit, maka dianggap sebagai *session* baru.
Ulangi langkah kedua dan ketiga untuk data selanjutnya sampai seluruh data *web log* selesai dibaca.

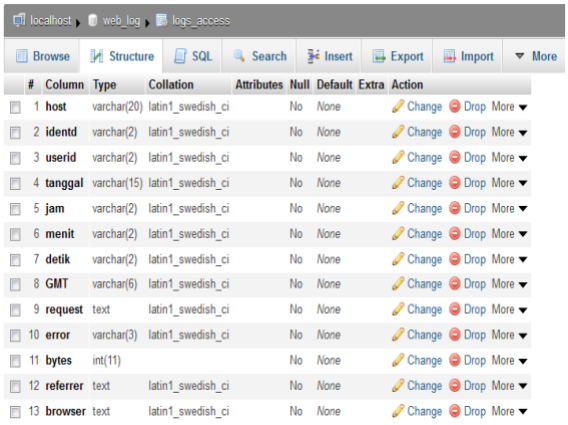
5. *Database of clean log.* Data *web log* siap digunakan untuk mendapatkan pola akses pengunjung menggunakan aplikasi *web log.*

B. Pembuatan aplikasi *web log*.

Pembuatan program aplikasi *web log* menggunakan bahasa pemrograman berbasis *web* (PHP) dan *MySQL* sebagai *database.*

Gambar 3 Tampilan halaman utama

Browse digunakan untuk memasukkan data *web log* (*.txt*) ke dalam program aplikasi, selanjutnya klik Proses untuk mengubah format *file web log (.txt)* menjadi format *.csv*. Hal ini dimaksudkan untuk memudahkan data *web log* dimasukkan ke dalam *database MySQL* (Gambar 4).



Gambar 4 Data *web log*

Untuk menghilangkan informasi yang tidak relevan pada data *web log* dengan klik menu *Data Cleaning* pada program aplikasi. Perbandingan data *web log* sebelum penyaringan menggunakan program aplikasi cukup signifikan yaitu dari jumlah data 1,027 berkurang menjadi 212. Hasil ini menunjukkan bahwa lebih dari setengah data *web log* memiliki informasi yang tidak relevan, seperti *file* gambar, *file* multimedia, spasi, *request method* selain GET.

Hasil program aplikasi dapat digunakan untuk mengetahui pola akses pengunjung

website, seperti jumlah pengunjung, banyaknya halaman yang dikunjungi, besarnya file yang diakses, kode status, halaman rujukan, dan *browser* yang digunakan oleh pengunjung.

V. Kesimpulan dan Saran

Dari hasil dan pembahasan yang telah dilakukan, diperoleh simpulan : Program aplikasi *web log* menghasilkan pola akses pengunjung sesuai dengan yang diharapkan dalam membantu *administrator web* dan *desainer web* memperbaiki kinerja dan kualitas *website e-government.*

Adapun saran yang dapat dilakukan untuk penelitian selanjutnya yaitu : Menambahkan menu untuk mengetahui jumlah *unique visitor*, waktu akses, dan perbaikan struktur *link* secara otomatis pada program aplikasi *web log*.

Referensi

[1] Diana. 2012. Evaluasi Website e-Government Menggunakan Analisis Kualitas dan Web Log [tesis]. Bogor : Program Pascasarjana, Institu Pertanian Bogor.

[2] Suneetha RK, Krishnamoorthi R. 2009b. Data Preprocessing and Easy Access Retrieval of Data through Data Ware House. Proceedings of the World Congress on Engineering and Computer Sience, San Francisco USA, 20-22 Okt 2009.

[3] The United Nations Department of Economic and Social Affairs. 2008. *Division for Public Administration and Development Management. United Nations E-Government Survey from E-Government to Connected Governance*, United Nations New York. Hlm 15-16.

[4] Inpres. 2003. Instruksi Presiden Republik Indonesia Nomor 3 Tahun 2003 tentang Kebijakan dan Strategi Nasional Pengembangan E-Government. Jakarta.

[5] World Wide Web. 1995 Status Code Definitions. http://www.w3.org/Protocols/ rfc2616/rfc2616-sec10.html.

[6] Nixon B. 2010. Pengembangan Program Penyaringan Data Web Log Untuk Analisis Pola Akses Pengunjung Web Server [tesis]. Jakarta: Program Pascasarjana, Universitas Indonesia.

[7] Khare R, Laurence S. 2000. HTTP Status Codes. http://www.ianan.org/ assignments/http-status-codes/http-status-codes.xml

[8] Suneetha RK, Krishnamoorthi R. 2009a. Identifying User Behavior by Analyzing Web Server Access Log File. *IJCSNS International Journal of Computer Science and Network Secutiry* 9:327-322. http ://paper.ijcsnc.org/07_book/200904/20090444.pdf

[9] Borghuis MGM. 2002. *A White Paper on the Filters to be Applied to Raw Usage Data Before Usage Analysis Can Start*. Ed ke-2. Usage Research Science Direct. Amsterdam.

[10] World Wide Web Consortium. 1999. Web Characterization Terminology & Definitions Sheet. http://www.w3.org/1999/05/WCA-terms