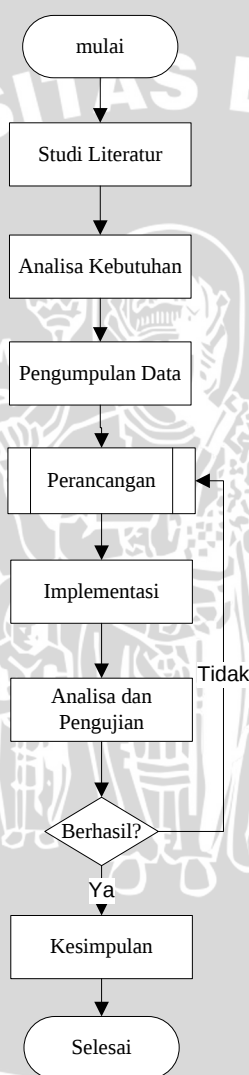


BAB III

METODE PENELITIAN DAN PERANCANGAN

Pada bab ini akan dijelaskan langkah – langkah yang terdiri dari studi literatur, metode pengambilan data, analisis kebutuhan, perancangan sistem, implementasi, pengujian dan analisis serta pengambilan keputusan. Gambar 3.1 merupakan diagram alir dari pelaksanaan penelitian secara keseluruhan



Gambar 3. 1 Diagram Alir Pelaksanaan Penelitian

3.1 Studi Literatur

Studi literatur mempelajari dasar teori yang digunakan untuk menunjang penelitian Penggalan Pola Sekuensial pada Data Akses Pengguna Website Menggunakan Algoritma *PrefixSpan*. Teori – teori tersebut meliputi :

- a. Log akses web
- b. Web Mining
- c. Web Usage Mining
- d. Pola Sekuensial (Algoritma Pola Sekuensial *PrefixSpan*)
- e. Pemrograman C#
- f. Database MySQL

3.2 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk menganalisis dan mendapatkan semua kebutuhan yang diperlukan dalam perancangan sistem Penggalan Pola Sekuensial pada Data Akses Pengguna Website Menggunakan Algoritma *PrefixSpan*. Tindakan yang dilakukan pada tahap analisis kebutuhan adalah mengidentifikasi perangkat keras, perangkat lunak, serta kebutuhan akan data yang digunakan.

3.2.1 Kebutuhan Perangkat Keras

- Laptop atau PC

3.2.2 Kebutuhan Perangkat Lunak

- Sistem Operasi Microsoft Windows 7
- Apache Web server
- Database Management System (DBMS) MySQL
- Visual Studio sebagai aplikasi untuk pembuatan sistem menggunakan bahasa pemrograman C#.

3.2.3 Data yang dibutuhkan adalah data akses pengguna website Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya yang tercatat dalam file *web access log*

3.3 Pengumpulan Data

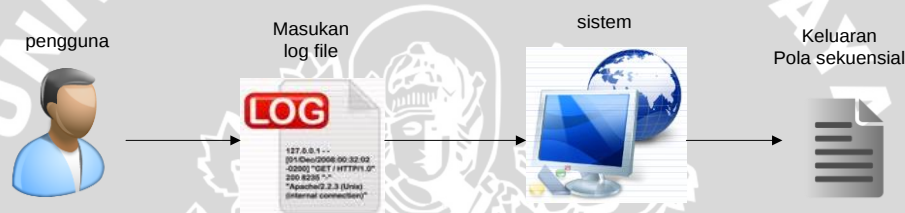
Penelitian ini membutuhkan data berupa data *web access log* PTIIK UB yang tersimpan dalam server. Berdasarkan jenis sumber data *web access log*

termasuk data sekunder, yaitu data yang diperoleh dari pihak lain. Data sekunder merupakan data yang dikumpulkan tetapi tidak dipersiapkan untuk kegiatan penelitian, namun data ini bisa digunakan untuk tujuan penelitian. Data diambil dari *web server* yang berada di Pengkajian dan Pengembangan Teknologi Informasi (PPTI) Universitas brawijaya.

3.4 Perancangan

Setelah pengambilan data dan semua kebutuhan sistem didapatkan maka selanjutnya dilakukan perancangan sistem. Rancangan yang disusun dapat diterapkan pada aplikasi sampai aplikasi berjalan.

3.4.1 Arsitektur Sistem



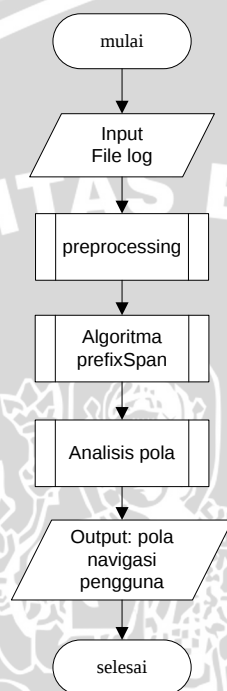
Gambar 3. 2 Arsitektur Sistem

Pengguna dapat menginputkan file *web access log*, sistem akan melakukan tahap *preprocessing* dimana tahap ini berupa pembersihan data, identifikasi pengguna, dan identifikasi sesi. Langkah selanjutnya sistem akan melakukan proses *pattern discovery* dimana inputan dari tahap ini adalah *sequence* dari id pengguna dan halaman apa saja yang sudah diakses pengguna. Pemrosesan pencarian pola dilakukan dengan menerapkan algoritma *PrefixSpan*. Keluaran yang dihasilkan dari proses *pattern discovery* ini berupa himpunan pola sekuensial yang berisi kumpulan dari pola yang didapat. Tahap *Pattern Analysis* merupakan proses untuk melakukan analisis terhadap hasil dari pola yang telah didapatkan dengan cara melakukan perhitungan *confidence* dan perhitungan *lift ratio*.

3.4.2 Perancangan Sistem

Proses perancangan sistem keseluruhan dapat dilihat pada gambar 3.3. Pada gambar 3.3 dapat dilihat bahwa masukan dari sistem adalah file *web acces log* PTIIK UB. File tersebut kemudian dilakukan proses *preprocessing* yang terdiri

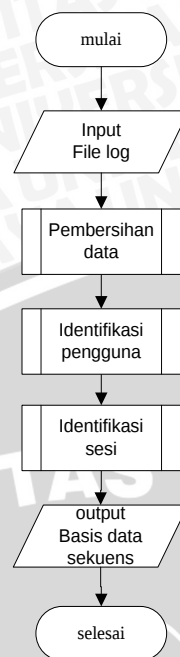
dari pembersihan data, identifikasi pengguna, dan identifikasi sesi. Hasil dari *preprocessing* akan dilakukan penggalian pola dengan menggunakan algoritma *PrefixSpan*. Hasil pola yang didapat akan dilakukan analisis pola sehingga dapat dihasilkan pola kebiasaan pengguna dalam mengakses *website* PTIIK.



Gambar 3. 3 Diagram Alir Sistem

3.4.2.1. Data Preprocessing

Tahap ini terdiri dari pembersihan data, identifikasi pengguna, dan identifikasi sesi. Untuk diagram alir dari *preprocessing* dapat dilihat pada gambar 3.4. Masukan proses ini adalah *web access log* dan keluaran yang dihasilkan adalah *sequence database* yang terdiri dari tabel *iduser*, *idsesi*, *idmapping*(substitusi dari halaman-halaman akses), dan waktu akses.



Gambar 3. 4 Data *Preprocessing*

- Pembersihan Data

Pada proses pembersihan data, data yang tidak relevan akan dihapus. Penghapusan dilakukan setelah adanya pemecahan terhadap baris log dan juga seleksi terhadap halaman akses yang diminta oleh pengguna, kemudian penghapusan dilakukan dengan cara memeriksa URL dan kode status http. Gambar 3.5 merupakan digram alir untuk proses pembersihan data.

Penjelasan gambar 3.5 :

- Masukan pada proses ini adalah *web acces log*
- File log akan dipecah berdasarkan baris pada file log.
- Baris yang memiliki format gambar, video, dan file CSS, file JS, Swf, zip. Format-format ini memiliki akhiran nama file seperti seperti “.gif”, “.jpeg”, “.jpg”, “.bmp”, “.png”, “.tiff”, “.mp4”, “.avi”, “.mkv”, “.pdf”, “.doc”, “.docx”, “.txt”, “.xls”, “.xml”, “.php” akan dihapus.
- Baris yang mencatat permintaan yang dilakukan robots (*crawler*) suatu mesin pencarian akan dihapus.
- Kode status HTTP yang gagal juga akan dihapus. Data dengan kode status http 200 yang akan digunakan. Kode status 200 menyatakan bahwa request

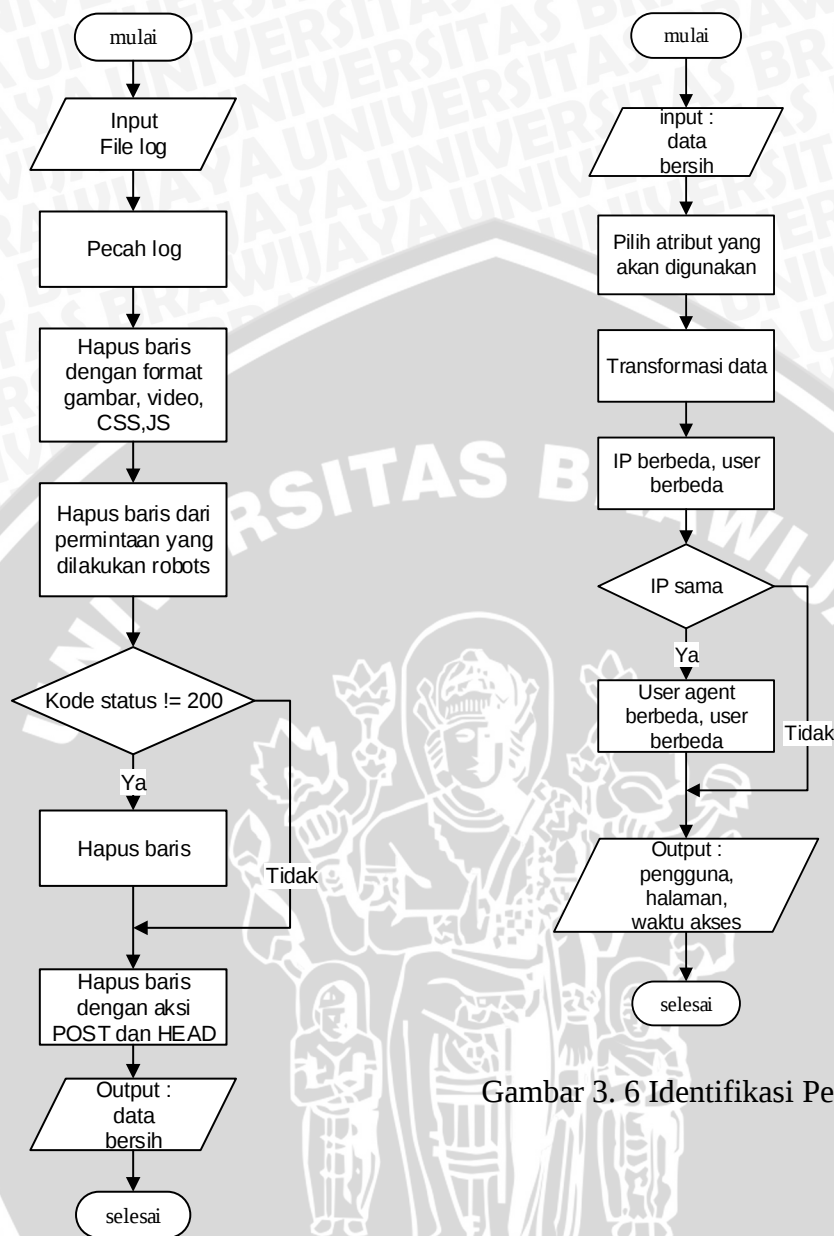
berhasil dilakukan dan server berhasil memproses permintaan lalu mengirimkan halaman yang diminta kepada pengguna.

- Catatan dengan aksi POST dan HEAD akan dihapus, yang diperhitungkan hanya GET.
- Keluaran dari proses ini adalah data bersih yang siap untuk dilakukan proses selanjutnya
- Identifikasi Pengguna

Gambar 3.6 merupakan digram alir untuk proses identifikasi pengguna.

Penjelasan diagram alir :

- Data log yang sudah dibersihkan akan dipilih beberapa atribut yang digunakan sebagai inputan identifikasi pengguna. Atribut-atribut yang digunakan adalah alamat IP dan *user agent* untuk identifikasi pengguna, waktu dan tanggal akses yang digunakan untuk identifikasi sesi, dan halaman yang diminta untuk mendapatkan halaman-halaman yang diakses pengguna.
- Halaman akses pengguna terlebih dulu akan disubstitusikan menjadi bentuk numerik dengan kode yang bersifat unik. Hal ini dilakukan agar lebih sederhana sehingga bisa mempermudah proses komputasi.
- IP yang berbeda menandakan pengguna yang berbeda.
- Jika memiliki alamat IP sama, tetapi *user agent* (tipe dari browser dan sistem operasi) berbeda maka dianggap pengguna yang berbeda.
- Proses ini akan menghasilkan pengguna yang mengakses *website* dan halaman apa yang diakses, dan juga tanggal dan waktu akses.
- Identifikasi Sesi
Identifikasi sesi digunakan untuk membagi akses halaman dari setiap pengguna pada satu waktu, dilakukan dengan cara menggunakan interval waktu (*threshold*) antara dua akses pengguna (*user*). *Threshold* yang digunakan adalah 30 menit (sesi standar PHP). Digram alir untuk proses identifikasi sesi dapat dilihat pada gambar 3.7



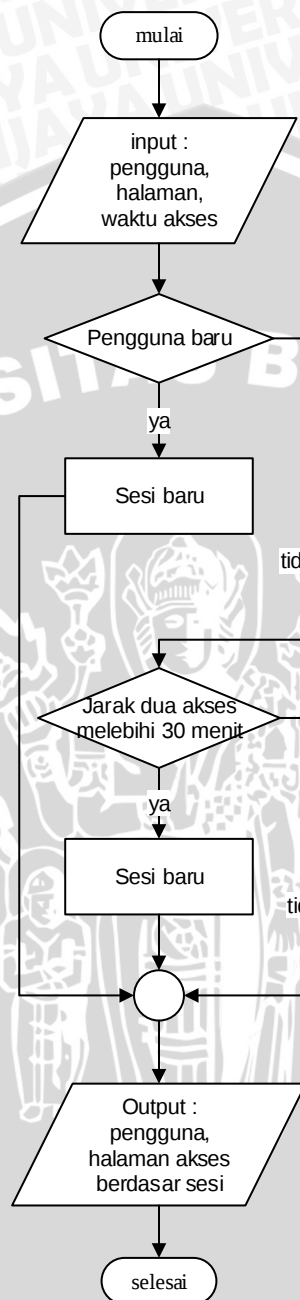
Gambar 3. 6 Identifikasi Pengguna

Gambar 3. 5 Pembersihan Data

Penjelasan diagram alir gambar 3.7

- Masukan proses ini adalah data hasil identifikasi pengguna
- Jika terdapat pengguna baru, maka ada sesi baru
- Jika jarak antara dua akses pengguna yang sama melebihi batas *threshold* maka pengguna tersebut dianggap memulai sesi baru. *Threshold* yang digunakan adalah 30 menit (sesi standar PHP).

- Keluaran yang dihasilkan adalah pengguna dan halaman yang diakses pengguna yang telah dibagi berdasarkan sesinya.



Gambar 3. 7 Identifikasi Sesi

3.4.2.2. Algoritma *PrefixSpan*

Hasil data yang sudah dilakukan *preprocessing* akan di implementasikan algoritma *PrefixSpan* untuk menghasilkan pola akses dari pengguna. Pada

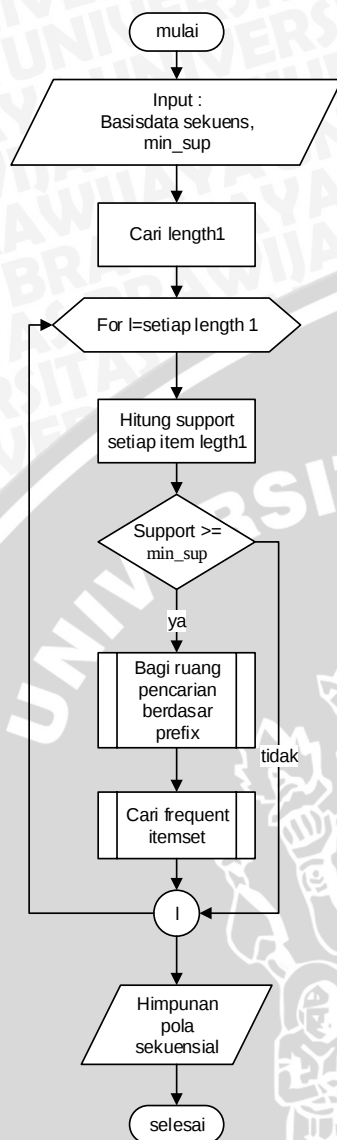
gambar 3.8 merupakan diagram alir proses algoritma *PrefixSpan* berikut adalah penjelasannya :

- Masukan dari proses ini adalah *sequence database* dan juga minimum *support* yang dimasukkan oleh pengguna.
- Mencari *length-1 sequential patterns* untuk dijadikan sebagai kandidat *prefix*.
- Scan *sequence database* satu kali untuk mendapatkan item yang sering muncul yang digunakan untuk menghitung *support*
- Jika setiap item dicari *length-1* yang memenuhi minimum *support*, dapat dijadikan *prefix* dan dilakukan proses selanjutnya.
- Membagi ruang pencarian. Himpunan lengkap dari pola sekuensial dapat dibagi menjadi beberapa bagian berdasarkan prefiksnya.
- Pencarian *frequent itemset* dari seluruh kandidat *prefix*.
- Keluaran yang dihasilkan adalah himpunan pola sekuensial.

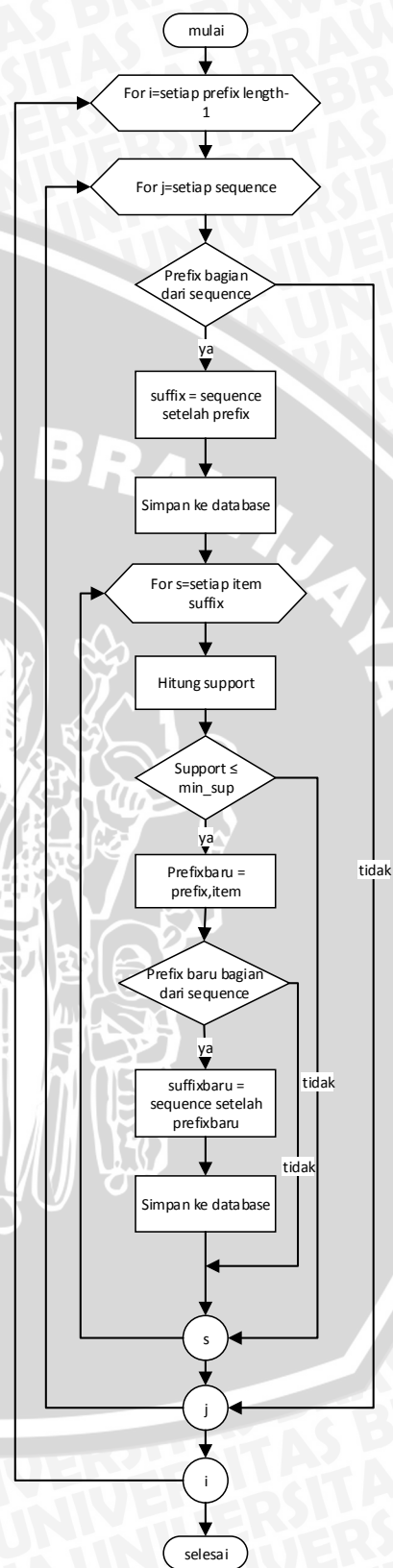
- **Pembagian Ruang Pencarian**

Gambar 3.9 merupakan diagram alir untuk proses pembagian ruang pencarian. Penjelasan diagram alir untuk gambar 3.9:

- Untuk setiap *prefix length-1* yang ada pada *sequence* maka akan didapatkan *suffix sequence*.
- *Suffix sequence* didapatkan dari *sequence* yang berada setelah *prefix*.
- Hasil *suffix sequence* kemudian akan disimpan di dalam database.
- Item-item *suffix sequence* yang disimpan di database kemudian di scan satu kali untuk mendapatkan nilai *support* dari item tersebut
- Untuk item yang memiliki *support* lebih dari minimum *support* akan menjadi *prefix* baru dengan aturan *prefix* baru = *prefix* lama,item
- Selanjutnya akan dicari *suffix sequence* yang baru. Iterasi berhenti ketika sudah tidak ada item yang memenuhi minimum *support*.



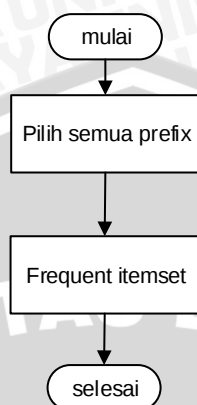
Gambar 3. 8 Algoritma *PrefixSpan*



Gambar 3. 9 Pembagian Ruang Pencarian

- Frequent Itemset

Gambar 3.10 merupakan diagram alir untuk proses pencarian *frequent itemset*



Gambar 3. 10 Pencarian *Frequent Itemset*

Penjelasan diagram alir:

- Semua *prefix* yang telah didapat pada proses-proses sebelumnya akan dijadikan *frequent itemset*. *Frequent itemset* inilah yang menjadi keluaran sebagai himpunan pola sekuensial

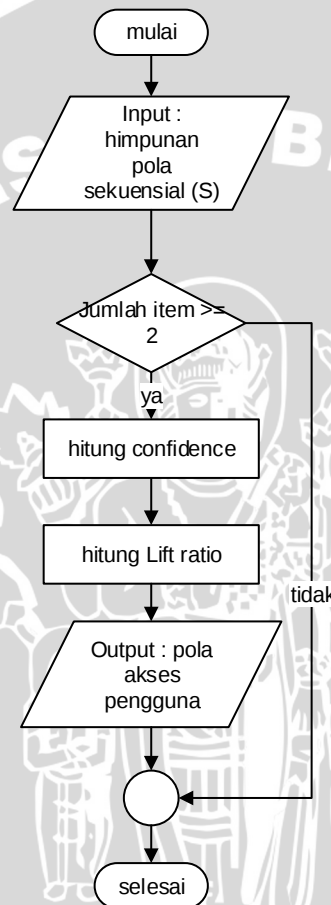
3.4.2.3. Analisis Pola

Proses analisis pola dilakukan dengan cara melakukan perhitungan *confidence* dan *lift ratio* terhadap himpunan pola yang dihasilkan oleh algoritma. Perhitungan dimulai dari pola yang memiliki dua item sampai pola yang memiliki item terpanjang.

Pada gambar 3.11 merupakan diagram alir untuk proses analisis pola menggunakan *confidence*.

- Masukan dari proses ini adalah himpunan pola sekuensial yang memiliki minimal dua item.
- Dilakukan perhitungan *confidence*. Untuk menghitung *confidence* digunakan persamaan (2-2).
- Kemudian dilakukan perhitungan *lift ratio* untuk mengetahui apakah *rule* yang dihasilkan bermanfaat atau tidak. Untuk menghitung *lift ratio* digunakan persamaan (2-4).

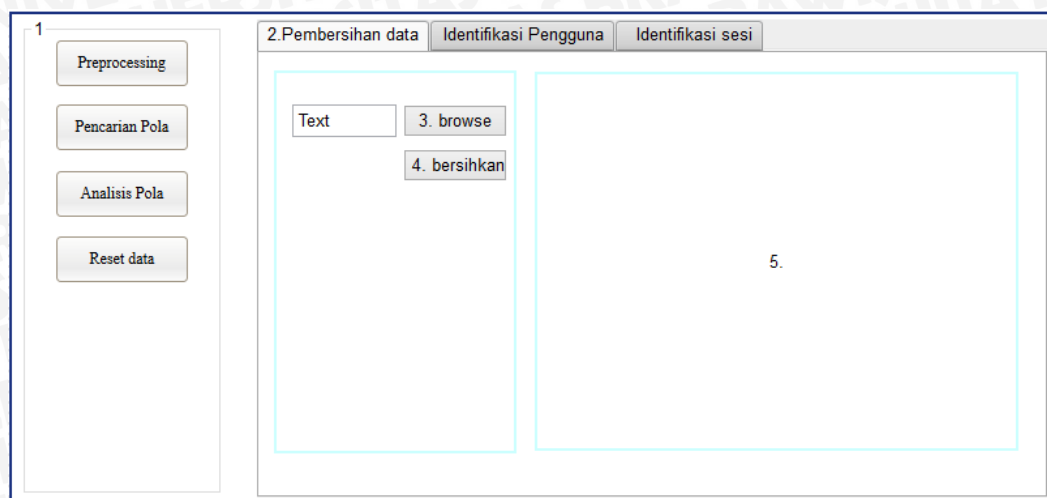
Keluran yang dihasilkan berupa pola pengguna mengunjungi *website* PTIIK UB yang bisa dimanfaatkan oleh *web designer* sebagai rekomendasi dalam menata ulang struktur *website*. Misalkan, ditemukan aturan dengan nilai *confidence* 70%, dimana A, B, dan C adalah halaman *website* yang bisa diakses. Jika pengguna mengunjungi halaman A dan B, maka terdapat 70% kemungkinan dia akan mengunjungi halaman C juga pada *session* yang sama.



Gambar 3. 11 Analisis Pola Menggunakan *Confidence* dan *lift ratio*

3.4.3 Perancangan Antarmuka

Proses implementasi akan membutuhkan sebuah aplikasi sebagai antarmuka dan media interaktif antara sistem dengan pengguna. Rancangan antarmuka dalam penelitian ini dibuat dengan menggunakan aplikasi pencil. Gambar 3.12 dan 3.13 adalah tampilan dari rancangan antarmuka yang nantinya akan di implementasikan pada aplikasi penggalan pola sekuensial.



Gambar 3. 12 Rancangan Tampilan *Preprocessing*

Gambar 3.12 merupakan rancangan antarmuka dari *preprocessing* yang terdiri dari proses pembersihan data, identifikasi pengguna, identifikasi sesi

1. Pada bagian satu berisi menu-menu yang akan ada dalam aplikasi seperti *preprocessing*, penggalian pola, analisis pola dan reset data.
2. Bagian dua merupakan tab menu *preprocessing* yang terdiri dari pembersihan data, identifikasi pengguna, dan identifikasi sesi.
3. Tombol “browse” berfungsi untuk memilih file *web acces log* yang akan digunakan sebagai inputan untuk proses selanjutnya.
4. Tombol “bersihkan” digunakan untuk melakukan proses pembersihan data pada file inputan.
5. Hasil dari proses pembersihan data akan ditampilkan dalam bentuk tabel dan diletakkan pada gambar bagian 5.

Tampilan proses pencarian pola pada gambar 3.13 terdapat dua proses, proses pertama adalah proses untuk mendapatkan *sequence database* / basis data transaksi berdasarkan halaman yang diinginkan oleh pengguna. Kemudian proses kedua digunakan untuk mencari pola sekeunsial dari *sequence database* yang sudah terbentuk. Dalam tampilan ini terdapat beberapa komponen antara lain :

1. Pada bagian nomor satu berisi menu-menu yang akan ada dalam aplikasi seperti *preprocessing*, penggalian pola, analisis pola dan reset data.

2. Bagian dua merupakan groupbox untuk proses pembuatan *sequence database* / basis data transaksi.
3. Tombol “proses” digunakan untuk memulai proses pembuatan *sequence database* berdasarkan masukan idmapping dari pengguna.
4. *Sequence database* yang dihasilkan akan ditampilkan pada bagian nomor 4.
5. Bagian lima merupakan groupbox proses pencarian pola.
6. Tombol “proses” digunakan untuk memulai proses dengan masukan minimum *support* dari pengguna. Minimum *support* berkisar antara nilai 0-100.
7. Hasil himpunan pola yang dihasilkan oleh proses ini akan ditmapilkan pada bagian tujuh.

Gambar 3. 13 Rancangan Antarmuka Proses Penggalian Pola

Tampilan proses analisis pola pada gambar 3.14 terdapat komponen sebagai berikut ini :

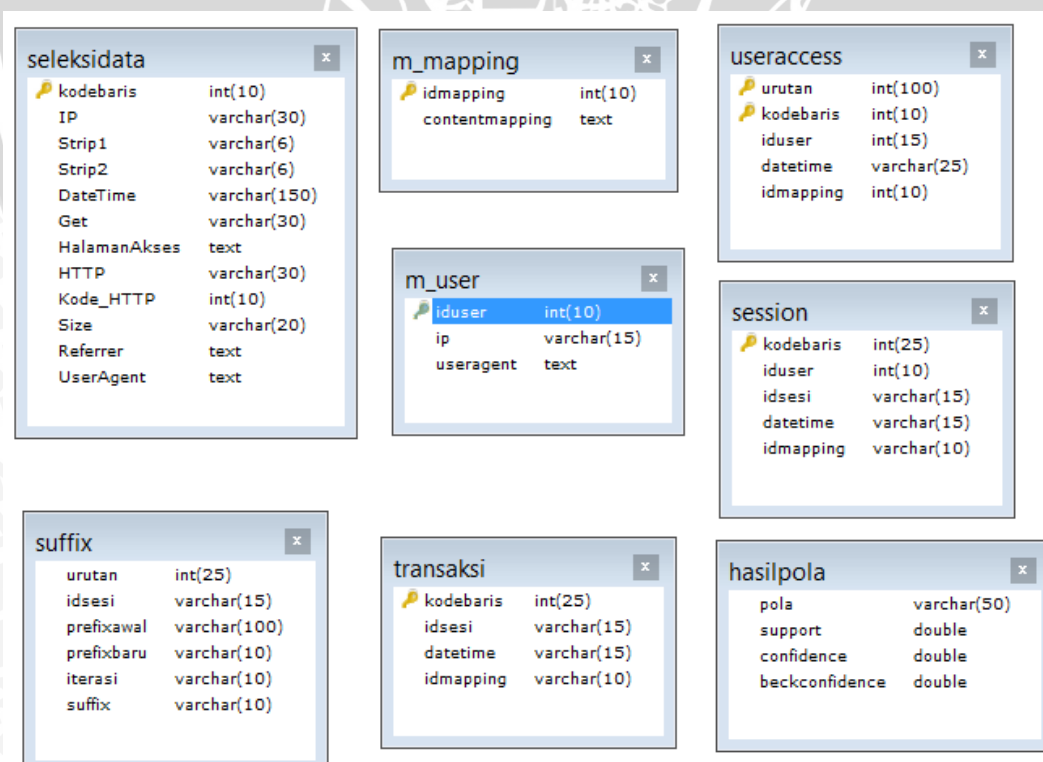
1. Pada bagian nomor satu berisi menu-menu yang akan ada dalam aplikasi seperti *preprocessing*, penggalian pola, analisis pola, dan reset data.
2. Tombol “proses” digunakan untuk memulai proses penghitungan *confidence* dengan masukan minimum *confidence* dari pengguna. Pola yang dianggap benar ada pola yang memiliki nilai *confidence* lebih dari 70%.
3. Hasil dari pola yang memenuhi minimum *confidence* akan ditampilkan pada bagian tiga.



Gambar 3. 14 Rancangan Antarmuka Proses Analisis Pola

3.4.4 Rancangan Basis Data

Pada Gambar 3.15 merupakan *physical diagram* dari rancangan basis data yang akan digunakan. Tidak terdapat relasi pada table-table yang ada di dalam basis data karena table-table tersebut hanya berfungsi untuk menyimpan data-data yang digunakan.



Gambar 3. 15 physical diagram

3.5 Implementasi Sistem

Pada tahap implementasi akan dilakukan penerapan desain dan perancangan yang telah dibuat. Sistem ini dibangun dengan bahasa C#, kemudian DBMS MySQL akan digunakan sebagai tempat penyimpanan semua data yang akan ataupun yang telah diproses. Implementasi sistem ini meliputi:

1. Penerapan *preprocessing* yang meliputi pembersihan data, identifikasi pengguna, dan identifikasi sesi
2. Penerapan algoritma *PrefixSpan* untuk mencari himpunan pola sekuensial
3. Penerapan analisis pola untuk mendapatkan pola navigasi pengguna *website* berdasarkan hasil dari *confidence*
4. Pembuatan antarmuka untuk *end-user*.

3.6 Perhitungan Manual

Penghitungan manual berfungsi untuk memberikan gambaran umum perancangan sistem yang akan dibangun. Contoh perhitungan manual proses *preprocessing* dan implementasi algoritma *PrefixSpan* pada data *web acces log* adalah sebagai berikut :

3.6.1. *Preprocessing*

Gambar 3.16 merupakan contoh *file web acces log*, data tersebut akan dilakukan proses *preprocessing*. Proses yang dilakukan adalah pembersihan data, identifikasi pengguna, dan identifikasi sesi.

3.6.1.1 Pembersihan Data

Data pertama kali akan dilakukan pemecahan yang kemudian akan diseleksi. Data hasil seleksi yang tidak sesuai akan dihapus. Dari data awal terdapat 3 data yang dihapus, data tersebut dihapus karena kode status HTTP yang dimiliki data pada baris tiga tidak sama dengan 200, pada baris tujuh dan delapan mengandung format gambar dan format dokumen pada URL dari halaman yang diminta pengguna. Sehingga dihasilkan data bersih sebanyak 22 baris. Data bersih dapat dilihat pada tabel 3.1

114.79.19.6 - - [18/Dec/2012:06:33:03 +0700] "GET /http://ptiik.ub.ac.id" 200 234 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.7 (KHTML, like Gecko) Chrome/16.0.912.63 Safari/535.7"

114.79.19.6 - - [18/Dec/2012:06:34:04 +0700] "GET /http://ptiik.ub.ac.id/content/read/organisasi-kemahasiswaan" 200 234 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.7 (KHTML, like Gecko) Chrome/16.0.912.63 Safari/535.7"

114.79.19.6 - - [18/Dec/2012:06:37:15 +0700] "GET /http://ptiik.ub.ac.id/content/pelayanan" 404 234 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.7 (KHTML, like Gecko) Chrome/16.0.912.63 Safari/535.7"

114.79.19.6 - - [18/Dec/2012:06:41:40 +0700] "GET /http://ptiik.ub.ac.id/content/read/informasi-bpp" 200 326 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.7 (KHTML, like Gecko) Chrome/16.0.912.63 Safari/535.7"

114.79.19.6 - - [18/Dec/2012:06:59:12 +0700] "GET /http://ptiik.ub.ac.id/content/read/struktur-organisasi" 200 234 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.7 (KHTML, like Gecko) Chrome/16.0.912.63 Safari/535.7"

127.0.0.1 - - [21/Jan/2013:22:59:12 +0700] "GET /http://ptiik.ub.ac.id" 200 234 "Mozilla/4.08 [en] (Win98; I;Nav)"

127.0.0.1 - - [21/Jan/2013:23:01:32 +0700] "GET /http://ptiik.ub.ac.id/monita/public/uploads/files/Skripsi-Final-last.jpg" 200 2326 "Mozilla/4.08 [en] (Win98; I;Nav)"

127.0.0.1 - - [21/Jan/2013:23:03:22 +0700] "GET /http://ptiik.ub.ac.id/monita/public/uploads/files/SP-1Form-pengajuan-TOPIK-Skripsi-baru.docx" 200 2326 "Mozilla/4.08 [en] (Win98; I;Nav)"

127.0.0.1 - - [21/Jan/2013:23:07:10 +0700] "GET /http://ptiik.ub.ac.id" 200 234 "Mozilla/4.08 [en] (Win98; I;Nav)"

127.0.0.1 - - [21/Jan/2013:23:10:42 +0700] "GET /http://ptiik.ub.ac.id/content/read/prestasi-mahasiswa" 200 2326 "Mozilla/4.08 [en] (Win98; I;Nav)"

127.0.0.1 - - [21/Jan/2013:23:23:21 +0700] "GET /http://ptiik.ub.ac.id/content/read/organisasi-kemahasiswaan" 200 234 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.7 (KHTML, like Gecko) Chrome/16.0.912.63 Safari/535.7"

127.0.0.1 - - [21/Jan/2013:23:25:02 +0700] "GET /http://ptiik.ub.ac.id/content/read/informasi-bpp" 200 326 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.7 (KHTML, like Gecko) Chrome/16.0.912.63 Safari/535.7"

127.0.0.1 - - [21/Jan/2013:23:25:12 +0700] "GET /http://ptiik.ub.ac.id/content/read/sekilas-ptiik" 200 2326 "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/535.7 (KHTML, like Gecko) Chrome/16.0.912.63 Safari/535.7"

127.123.456.78 - - [25/Mar/2013:03:04:41 +0700] "GET /http://ptiik.ub.ac.id" 200 3290 "Mozilla/3.04(Win95, I)"

127.123.456.78 - - [25/Mar/2013:03:05:31 +0700] "GET /http://ptiik.ub.ac.id/content/read/informasi-pendaftaran" 200 3290 "Mozilla/3.04(Win95, I)"

127.123.456.78 - - [25/Mar/2013:03:07:01 +0700] "GET /http://ptiik.ub.ac.id/content/read/informasi-pendaftaran" 200 3290 "Mozilla/3.04(Win95, I)"

127.123.456.78 - - [25/Mar/2013:03:10:46 +0700] "GET /http://ptiik.ub.ac.id/content/read/laboratorium" 200 236 "Mozilla/3.04(Win95, I)"

127.123.456.78 - - [25/Mar/2013:03:15:10 +0700] "GET /http://ptiik.ub.ac.id/content/read/prestasi-mahasiswa" 200 2326 "Mozilla/3.04(Win95, I)"

127.123.456.78 - - [25/Mar/2013:03:19:11 +0700] "GET /http://ptiik.ub.ac.id/content/read/ptiik-digital-library" 200 326 "Mozilla/3.04(Win95, I)"

172.123.456.78 - - [06/Feb/2013:12:01:01 +0700] "GET /http://ptiik.ub.ac.id" 200 3290 "Mozilla/3.01(X11, I, IRIX6.2, IP22)"

172.123.456.78 - - [06/Feb/2013:12:02:02 +0700] "GET http://ptiik.ub.ac.id/content/read/pusat-kajian" 200 290 "Mozilla/3.04(Win95, I)"

172.123.456.78 - - [06/Feb/2013:12:04:41 +0700] "GET /http://ptiik.ub.ac.id/content/read/sekilas-ptiik" 200 2326 "Mozilla/3.04(Win95, I)"

172.123.456.78 - - [06/Feb/2013:12:06:10 +0700] "GET /http://ptiik.ub.ac.id/content/read/laboratorium" 200 236 "Mozilla/3.04(Win95, I)"

172.123.456.78 - - [06/Feb/2013:12:09:01 +0700] "GET /http://ptiik.ub.ac.id/content/read/informasi-pendaftaran" 200 3290 "Mozilla/3.04(Win95, I)"

172.123.456.78 - - [06/Feb/2013:12:10:51 +0700] "GET /http://ptiik.ub.ac.id/content/read/prestasi-mahasiswa" 200 2326 "Mozilla/3.04(Win95, I)"

Gambar 3. 16 File Web Access Log

3.6.1.2 Identifikasi Pengguna

Data bersih yang didapatkan dari proses pembersihan data kemudian akan dipilih atribut-atribut yang akan digunakan. Tabel 3.2 merupakan hasil setelah dilakukan pemilihan atribut. Selanjutnya halaman akses akan disubstitusikan menjadi bentuk yang lebih sederhana agar memudahkan proses komputasi. Hasil substitusi dapat dilihat pada tabel 3.4. Proses substitusi diberlakukan aturan seperti tabel 3.3. Setelah substitusi selesai kemudian id pengguna akan dicari berdasarkan alamat IP dan *user agent* nya. Hasil dari proses identifikasi pengguna dapat dilihat pada tabel 3.5

3.6.1.3 Identifikasi sesi

Pada proses ini akan membagi akses pengguna berdasarkan waktu akses berdasarkan *threshold*. *Threshold* yang digunakan adalah 30 menit. Pada data yang digunakan tidak terdapat jarak akses pengguna yang melebihi *threshold*. Hasil dari proses identifikasi sesi adalah *sequence database* yang dapat dilihat pada tabel 3.6

Tabel 3. 1 Data Bersih Hasil *Preprocessing*

kode	IP	str	Str	Tanggal dan Waktu	Get	Halaman Akses	Status	Size	User Agent
1	114.79.19.6	-	-	[18/Dec/2012:06:33:03]	GET	/	200	234	AppleWebKit/535.7 (KHTML, like Gecko) C
2	114.79.19.6	-	-	[18/Dec/2012:06:34:04]	GET	/content/read/organisasi-ke	200	234	AppleWebKit/535.7 (KHTML, like Gecko) C
4	114.79.19.6	-	-	[18/Dec/2012:06:34:04]	GET	/content/read/informasi-bpt	200	236	AppleWebKit/535.7 (KHTML, like Gecko) C
5	114.79.19.6	-	-	[18/Dec/2012:06:59:12]	GET	/content/read/struktur-orga	200	234	AppleWebKit/535.7 (KHTML, like Gecko) C
6	127.0.0.1	-	-	[21/Jan/2013:22:59:12]	GET	/	200	234	Mozilla/4.08 [en] (Win98; I ;Nav)
9	127.0.0.1	-	-	[21/Jan/2013:23:07:10]	GET	/	200	234	Mozilla/4.08 [en] (Win98; I ;Nav)
10	127.0.0.1	-	-	[21/Jan/2013:23:10:42]	GET	/content/read/prestasi-mah	200	2326	Mozilla/4.08 [en] (Win98; I ;Nav)
11	127.0.0.1	-	-	[21/Jan/2013:23:23:21]	GET	/content/read/organisasi-ke	200	234	Mozilla/5.0 (Windows NT 6.1) AppleWebKit
12	127.0.0.1	-	-	[21/Jan/2013:23:25:02]	GET	/content/read/informasi-bpt	200	326	Mozilla/5.0 (Windows NT 6.1) AppleWebKit
13	127.0.0.1	-	-	[21/Jan/2013:23:25:12]	GET	/content/read/sekilas-ptiik	200	2326	Mozilla/5.0 (Windows NT 6.1) AppleWebKit
14	127.123.456.78	-	-	[25/Mar/2013:03:04:41]	GET	/	200	3290	Mozilla/3.04(Win95, I)
15	127.123.456.78	-	-	[25/Mar/2013:03:05:31]	GET	/content/read/informasi-per	200	3290	Mozilla/3.04(Win95, I)
16	127.123.456.78	-	-	[25/Mar/2013:03:07:01]	GET	/content/read/informasi-per	200	3290	Mozilla/3.04(Win95, I)
17	127.123.456.78	-	-	[25/Mar/2013:03:10:46]	GET	/content/read/laboratorium	200	236	Mozilla/3.04(Win95, I)
18	127.123.456.78	-	-	[25/Mar/2013:03:15:10]	GET	/content/read/prestasi-mah	200	2326	Mozilla/3.04(Win95, I)
19	127.123.456.78	-	-	[25/Mar/2013:03:19:11]	GET	/content/read/ptiik-digital-l	200	326	Mozilla/3.04(Win95, I)
20	172.123.456.78	-	-	[06/Feb/2013:12:01:01]	GET	/	200	3290	Mozilla/3.01(X11, I, IRIX6.2, IP22)
21	172.123.456.79	-	-	[06/Feb/2013:12:02:02]	GET	/content/read/pusat-kajian	200	290	Mozilla/3.01(X11, I, IRIX6.2, IP22)
22	172.123.456.80	-	-	[06/Feb/2013:12:04:41]	GET	/content/read/sekilas-ptiik	200	2326	Mozilla/3.01(X11, I, IRIX6.2, IP22)
23	172.123.456.81	-	-	[06/Feb/2013:12:06:10]	GET	/content/read/laboratorium	200	236	Mozilla/3.01(X11, I, IRIX6.2, IP22)
24	172.123.456.82	-	-	[06/Feb/2013:12:09:01]	GET	/content/read/informasi-per	200	3290	Mozilla/3.01(X11, I, IRIX6.2, IP22)
25	172.123.456.83	-	-	[06/Feb/2013:12:10:51]	GET	/content/read/prestasi-mah	200	2326	Mozilla/3.01(X11, I, IRIX6.2, IP22)

Tabel 3. 2 Atribut –atribut yang digunakan

IP	Tanggal dan Waktu	Halaman yang diminta	User Agent
114.79.19.6	18/Dec/2012:06:33:03	/	Mozilla/5.0 (Windows NT 6.1)
114.79.19.6	18/Dec/2012:06:34:04	/content/read/organisasi-kemahasiswaan	Mozilla/5.0 (Windows NT 6.1)
114.79.19.6	18/Dec/2012:06:41:40	/content/read/informasi-bpp	Mozilla/5.0 (Windows NT 6.1)
114.79.19.6	18/Dec/2012:06:59:12	/content/read/struktur-organisasi	Mozilla/5.0 (Windows NT 6.1)
127.0.0.1	21/Jan/2013:22:59:12	/	Mozilla/4.08 [en] (Win98; I;Nav)
127.0.0.1	21/Jan/2013:23:07:10	/	Mozilla/4.08 [en] (Win98; I;Nav)
127.0.0.1	21/Jan/2013:23:10:42	/content/read/prestasi-mahasiswa	Mozilla/4.08 [en] (Win98; I;Nav)
127.0.0.1	21/Jan/2013:23:23:21	/content/read/organisasi-kemahasiswaan	Mozilla/5.0 (Windows NT 6.1)
127.0.0.1	21/Jan/2013:23:25:02	/content/read/informasi-bpp	Mozilla/5.0 (Windows NT 6.1)
127.0.0.1	21/Jan/2013:23:25:12	content/read/sekilas-ptiik	Mozilla/5.0 (Windows NT 6.1)
127.123.456.78	25/Mar/2013:03:04:41	/	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:05:31	/content/read/informasi-pendaftaran	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:07:01	/content/read/informasi-pendaftaran	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:10:46	/content/read/laboratorium	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:15:10	/content/read/prestasi-mahasiswa	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:19:11	/content/read/ptiik-digital-library	Mozilla/3.04(Win95, I)
172.123.456.78	06/Feb/2013:12:01:01	/	Mozilla/4.08 [en] (Win98; I;Nav)
172.123.456.78	06/Feb/2013:12:02:02	/content/read/pusat-kajian	Mozilla/4.08 [en] (Win98; I;Nav)
172.123.456.78	06/Feb/2013:12:04:41	/content/read/sekilas-ptiik	Mozilla/4.08 [en] (Win98; I;Nav)
172.123.456.78	06/Feb/2013:12:06:10	/content/read/laboratorium	Mozilla/4.08 [en] (Win98; I;Nav)
172.123.456.78	06/Feb/2013:12:09:01	/content/read/informasi-pendaftaran	Mozilla/4.08 [en] (Win98; I;Nav)
172.123.456.78	06/Feb/2013:12:10:51	/content/read/prestasi-mahasiswa	Mozilla/4.08 [en] (Win98; I;Nav)

Tabel 3. 3 Mapping

Halaman akses	Mapping
/	1
/content/read/organisasi-kemahasiswaan	2
/content/read/informasi-bpp	3
/content/read/struktur-organisasi	4
/content/read/prestasi-mahasiswa	5
/content/read/sekilas-ptiik	6
/content/read/informasi-pendaftaran	7
/content/read/laboratorium	8
/content/read/ptiik-digital-library	9
/content/read/pusat-kajian	10

Tabel 3. 4 Hasil Substitusi

IP	Tanggal dan Waktu	Setelah Mapping	User agent
114.79.19.6	18/Dec/2012:06:33:03	1	Mozilla/5.0 (Windows NT 6.1)
114.79.19.6	18/Dec/2012:06:34:04	2	Mozilla/5.0 (Windows NT 6.1)
114.79.19.6	18/Dec/2012:06:41:40	3	Mozilla/5.0 (Windows NT 6.1)
114.79.19.6	18/Dec/2012:06:59:12	4	Mozilla/5.0 (Windows NT 6.1)
127.0.0.1	21/Jan/2013:22:59:12	1	Mozilla/4.08 [en] (Win98; I ;Nav)
127.0.0.1	21/Jan/2013:23:07:10	1	Mozilla/4.08 [en] (Win98; I ;Nav)
127.0.0.1	21/Jan/2013:23:10:42	5	Mozilla/4.08 [en] (Win98; I ;Nav)
127.0.0.1	21/Jan/2013:23:23:21	2	Mozilla/5.0 (Windows NT 6.1)
127.0.0.1	21/Jan/2013:23:25:02	3	Mozilla/5.0 (Windows NT 6.1)
127.0.0.1	21/Jan/2013:23:25:12	6	Mozilla/5.0 (Windows NT 6.1)
127.123.456.78	25/Mar/2013:03:04:41	1	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:05:31	7	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:07:01	7	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:10:46	8	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:15:10	5	Mozilla/3.04(Win95, I)
127.123.456.78	25/Mar/2013:03:19:11	9	Mozilla/3.04(Win95, I)
172.123.456.78	06/Feb/2013:12:01:01	1	Mozilla/4.08 [en] (Win98; I ;Nav)
172.123.456.78	06/Feb/2013:12:02:02	10	Mozilla/4.08 [en] (Win98; I ;Nav)
172.123.456.78	06/Feb/2013:12:04:41	6	Mozilla/4.08 [en] (Win98; I ;Nav)
172.123.456.78	06/Feb/2013:12:06:10	8	Mozilla/4.08 [en] (Win98; I ;Nav)
172.123.456.78	06/Feb/2013:12:09:01	7	Mozilla/4.08 [en] (Win98; I ;Nav)
172.123.456.78	06/Feb/2013:12:10:51	5	Mozilla/4.08 [en] (Win98; I ;Nav)

Tabel 3. 5 Hasil Akhir Proses Identifikasi Pengguna

ID user	IP	User agent	Mapping	Tanggal dan Waktu
user1	114.79.19.6	Mozilla/5.0 (Windows NT 6.1)	1	18/Dec/2012:06:33:03
	114.79.19.6	Mozilla/5.0 (Windows NT 6.1)	2	18/Dec/2012:06:34:04
	114.79.19.6	Mozilla/5.0 (Windows NT 6.1)	3	18/Dec/2012:06:41:40
	114.79.19.6	Mozilla/5.0 (Windows NT 6.1)	4	18/Dec/2012:06:59:12
user2	127.0.0.1	Mozilla/4.08 [en] (Win98; I ;Nav)	1	21/Jan/2013:22:59:12
	127.0.0.1	Mozilla/4.08 [en] (Win98; I ;Nav)	1	21/Jan/2013:23:07:10
	127.0.0.1	Mozilla/4.08 [en] (Win98; I ;Nav)	5	21/Jan/2013:23:10:42
user3	127.0.0.1	Mozilla/5.0 (Windows NT 6.1)	2	21/Jan/2013:23:23:21
	127.0.0.1	Mozilla/5.0 (Windows NT 6.1)	3	21/Jan/2013:23:25:02
	127.0.0.1	Mozilla/5.0 (Windows NT 6.1)	6	21/Jan/2013:23:25:12
user 4	127.123.456.78	Mozzila/3.04(Win95, I)	1	25/Mar/2013:03:04:41
	127.123.456.78	Mozzila/3.04(Win95, I)	7	25/Mar/2013:03:05:31
	127.123.456.78	Mozzila/3.04(Win95, I)	7	25/Mar/2013:03:07:01
	127.123.456.78	Mozzila/3.04(Win95, I)	8	25/Mar/2013:03:10:46
	127.123.456.78	Mozzila/3.04(Win95, I)	5	25/Mar/2013:03:15:10
	127.123.456.78	Mozzila/3.04(Win95, I)	9	25/Mar/2013:03:19:11
user5	172.123.456.78	06/Feb/2013:12:01:01	1	06/Feb/2013:12:01:01
	172.123.456.78	06/Feb/2013:12:02:02	10	06/Feb/2013:12:02:02
	172.123.456.78	06/Feb/2013:12:04:41	6	06/Feb/2013:12:04:41
	172.123.456.78	06/Feb/2013:12:06:10	8	06/Feb/2013:12:06:10
	172.123.456.78	06/Feb/2013:12:09:01	7	06/Feb/2013:12:09:01
	172.123.456.78	06/Feb/2013:12:10:51	5	06/Feb/2013:12:10:51

Tabel 3. 6 *Sequence Database*

id user	Id sesi	halaman yang diakses
User 1	Sesi01	1,2,3,4
User 2	Sesi02	1,1,5
User 3	Sesi03	2,3,6
User 4	Sesi04	1,7,7,8,5,9
User 5	Sesi05	1,10,6,8,7,5

3.6.2. Penerapan Algoritma *PrefixSpan*

Hasil identifikasi sesi pada tabel 3.6 merupakan masukan dari proses ini. Misal minimum *support* yang digunakan adalah 2

a. Langkah pertama mencari *length-1*

Tabel 3. 7 *length-1*

item	Support
1	4
2	2
3	2
4	1
5	2
6	2
7	2
8	2
9	1
10	1

Baris yang diberikan tanda merah tidak memenuhi minimum *support*, sehingga tidak digunakan sebagai *prefix* untuk proses selanjutnya.

b. *Prefix length -1*

Tabel 3. 8 *Prefix length -1*

<i>prefix</i>	Support
<1>	4
<2>	2
<3>	2
<5>	2
<6>	2
<7>	2
<8>	2

c. Proyeksi basis data

Tabel 3. 9 Hasil Suffix Berdasarkan *Prefix*nya

<i>Prefix</i>	Projected (suffix) sequences
<1>	<2,3,4> , <1,5> , <7,7,8,5,9> , <10,6,8,7,5>
<2>	<3,4> , <3,6>
<3>	<4> <6>
<5>	<9>
<6>	<8, 7, 5>
<7>	<7, 8, 5, 9> , <5>
<8>	<5,9> , <7,5>

- d. Mencari pola pada *prefix* <1>
- Dapatkan item yang ada pada suffix sequence *prefix* <1> dengan cara scan suffix sequence pada *prefix* <1>

Tabel 3. 10 Nilai Support item-item *prefix* <1>

items	Support
<1>	1
<2>	1
<3>	1
<4>	1
<5>	3
<6>	1
<7>	2
<8>	2
<9>	1
<10>	1

- Karena semua item yang memenuhi min sup memiliki *prefix* <1> maka item-item tersebut ditambahkan *prefix* <1> menjadi <1, 5>, <1, 7>, <1, 8>. Selanjutnya dilakukan scan terhadap *sequence database* untuk mencari length-2 pola sekuensial. maka didapatkan seperti pada tabel 3.11.

Tabel 3. 11 length-2 *prefix* <1>

Prefix	Support
<1, 5>	3
<1, 7>	2
<1, 8>	2

Prefix <1,5>, dengan *prefix* <1,5> suffix sequence yang dihasilkan adalah <9>. Jika dilakukan scan item, maka akan didapatkan item <9> dengan nilai support 1. Karena tidak memenuhi minimum support maka iterasi berhenti.

Prefix <1,7>, dengan *prefix* <1,7> didapatkan suffix sequence <7,8,5,9> , <5>. Dilakukan scan terhadap suffix sequence dan dihasilkan item <5> masih memenuhi minimum support. Kemudian dilanjutkan dengan *prefix* <1,7,5> yang menghasilkan suffix sequence <9>. Item <9> sudah tidak memenuhi minimum support.

Prefix <1,8>, dengan prefix <1,8> suffix sequence yang dihasilkan adalah <5,9> <7,5>. Jika dilakukan terhadap suffix sequence, maka akan didapatkan item <5> masih memenuhi minimum support. Kemudian dilanjutkan dengan prefix <1,8,5> yang menghasilkan suffix sequence <9>. Item <9> sudah tidak memenuhi minimum support karena hanya memiliki nilai support 1 Sehingga iterasi berhenti.

- Melakukan penggalan secara rekursif terhadap prefix <2> , <3>, <5>, <6>, <7>, <8>. Himpunan pola sekuensial yang dihasilkan dapat dilihat pada tabel 3.12

Tabel 3. 12 Hasil Himpunan Pola Sekuensial

prefix	Sequential Pattern
<1>	<1> , <1,5> , <1,7> , <1,7,5> <1,8> <1,8,5>
<2>	<2> , <2,3>
<3>	<3>
<5>	<5>
<6>	<6>
<7>	<7> , <7,5>
<8>	<8> , <8,5>

3.6.3. Penghitungan Confidence

Pada table 3.13 merupakan hasil dari perhitungan confidence terhadap pola sekuensial yang telah dihasilkan oleh algoritma PrefixSpan.

Proses perhitungan confidence dari rule yang dihasilkan

$$\text{rule } \langle 1 \rightarrow 5 \rangle = \frac{3}{4} \times 100 = 75$$

$$\text{rule } \langle 7 \rightarrow 5 \rangle = \frac{2}{2} \times 100 = 100$$

$$\text{rule } \langle 1 \rightarrow 7 \rangle = \frac{2}{4} \times 100 = 50$$

$$\text{rule } \langle 8 \rightarrow 5 \rangle = \frac{2}{2} \times 100 = 100$$

$$\text{rule } \langle 1 \rightarrow 8 \rangle = \frac{2}{4} \times 100 = 50$$

$$\text{rule } \langle 1,7 \rightarrow 5 \rangle = \frac{2}{2} \times 100 = 100$$

$$\text{rule } \langle 2 \rightarrow 3 \rangle = \frac{2}{2} \times 100 = 100$$

$$\text{rule } \langle 1,8 \rightarrow 5 \rangle = \frac{2}{2} \times 100 = 100$$

Perhitungan lift ratio dari rule yang dihasilkan

- Menghitung benchmark confidence

$$\text{rule } \langle 1 \rightarrow 5 \rangle = \frac{3}{5} \times 100 = 60$$

$$\text{rule } \langle 7 \rightarrow 5 \rangle = \frac{3}{5} \times 100 = 60$$

$$\text{rule } \langle 1 \rightarrow 7 \rangle = \frac{2}{5} \times 100 = 40$$

$$\text{rule } \langle 8 \rightarrow 5 \rangle = \frac{3}{5} \times 100 = 60$$

$$\text{rule } \langle 1 \rightarrow 8 \rangle = \frac{2}{5} \times 100 = 40$$

$$\text{rule } \langle 1,7 \rightarrow 5 \rangle = \frac{3}{5} \times 100 = 60$$

$$\text{rule } \langle 2 \rightarrow 3 \rangle = \frac{2}{5} \times 100 = 40$$

$$\text{rule } \langle 1,8 \rightarrow 5 \rangle = \frac{3}{5} \times 100 = 60$$

- Menghitung *lift ratio*

$$\text{rule } \langle 1 \rightarrow 5 \rangle = \frac{75}{60} = 1,25$$

$$\text{rule } \langle 7 \rightarrow 5 \rangle = \frac{100}{60} = 1,666666667$$

$$\text{rule } \langle 1 \rightarrow 7 \rangle = \frac{50}{40} = 1,25$$

$$\text{rule } \langle 8 \rightarrow 5 \rangle = \frac{100}{60} = 1,666666667$$

$$\text{rule } \langle 1 \rightarrow 8 \rangle = \frac{50}{40} = 1,25$$

$$\text{rule } \langle 1,7 \rightarrow 5 \rangle = \frac{100}{60} = 1,666666667$$

$$\text{rule } \langle 2 \rightarrow 3 \rangle = \frac{100}{40} = 2,5$$

$$\text{rule } \langle 1,8 \rightarrow 5 \rangle = \frac{100}{60} = 1,666666667$$

Tabel 3. 13 Hasil Aturan dengan nilai *Confidence* dan *Lift Ratio*

rule	Confidence	lift ratio
$\langle 1 \rightarrow 5 \rangle$	75	1,25
$\langle 1 \rightarrow 7 \rangle$	50	1,25
$\langle 1 \rightarrow 8 \rangle$	50	1,25
$\langle 2 \rightarrow 3 \rangle$	100	2,5
$\langle 7 \rightarrow 5 \rangle$	100	1,666666667
$\langle 8 \rightarrow 5 \rangle$	100	1,666666667
$\langle 1,7 \rightarrow 5 \rangle$	100	1,666666667
$\langle 1,8 \rightarrow 5 \rangle$	100	1,666666667

3.7 Perancangan Pengujian

Uji coba sistem penggalian pola sekuensial pada data akses pengguna *website* dibagi menjadi dua skenario, yaitu skenario satu dengan cara membandingkan hasil perhitungan manual dengan hasil sistem. Skenario dua dilakukan dengan cara memberikan nilai minimum *support* dan minimum *confidence* yang berbeda-beda terhadap sistem. Skenario uji coba digunakan untuk menguji validitas sistem aplikasi.

1. Skenario satu

Membandingkan hasil dari poin 3.6 perhitungan manual dengan hasil sistem. Apabila terdapat perbedaan antara hasil dari sistem dan perhitungan manual maka kemungkinan besar ada sesuatu yang kurang atau salah pada pemrograman aplikasi dan perlu dilakukan perbaikan dan pengujian ulang sistem sampai mendapatkan hasil yang sama dengan hasil manualisasi.

2. Skenario dua

Skenario dua dilakukan dengan cara memberikan nilai minimum *support* dan minimum *confidence* yang berbeda-beda terhadap sistem. Pada pengujian pengaruh minimum *support*, minimum *support* yang digunakan mulai dari 5,10,15,20,25..... sampai tidak ada pola dengan length-2 yang terbentuk. Kemudian untuk analisis aturan digunakan nilai minimum *confidence* 70%. Jika terdapat aturan yang memiliki nilai *confidence* diatas 70%, maka aturan tersebut dianggap benar. Sedangkan untuk *lift ratio* apabila didapatkan nilai *lift ratio* lebih dari satu maka menunjukkan bahwa ada manfaat dari *rule* yang terbentuk. Untuk rekap hasil pengaruh nilai *support* dan *confidence* nantinya akan dicatat seperti tabel 3.14

Pada pengujian pengaruh minimum *confidence*, *sequence database* yang digunakan sama seperti pada pengujian pengaruh minimum *support* tetapi nilai *support* yang dimasukkan tetap, yaitu 10%. Kemudian pada proses analisis pola akan diberi masukan nilai minimum *confidence* yang berbeda-beda. Minimum *confidence* yang digunakan adalah 70%, 80%, dan 90%.

Jika skenario satu dan skenario dua terpenuhi maka sistem dianggap valid, dan informasi yang dihasilkan dapat digunakan sebagai rekomendasi kepada *web designer* dalam menata ulang struktur *website* PTIIK.

Tabel 3. 14 Rekap Pengaruh Nilai Minimum *support*

No.	Minimum <i>support</i>	<i>Rule</i> yang terbentuk	Hasil <i>Confidence</i>	<i>Lift ratio</i>
1	20	<1,7→ 5>	100	
2		<1,8 → 5>	100	
3		<2→ 3>	100	
4				
5				
6				
N				