

**RANCANG BANGUN SISTEM TEMU KEMBALI INFORMASI  
ABSTRAK TUGAS AKHIR MAHASISWA PRODI TEKNIK  
INFORMATIKA UNSOED**

Oleh :  
Lasmedi Afuan

Prodi Teknik Informatika, Fakultas Sains dan Teknik,  
Universitas Jenderal Soedirman  
Jl. Mayjen Sungkono Blater Km 5. Purbalingga  
Email: lasmedi.afuan@unsoed.ac.id

**ABSTRAK**

Informasi merupakan hal yang sangat mudah didapatkan dan diakses. Tetapi terkadang informasi yang diperoleh tidak sesuai dengan apa yang diinginkan pengguna. Diperlukan sistem yang dapat membantu mencari informasi yang dibutuhkan secara efektif dan efisien. Sistem informasi ini sering kali disebut dengan istilah sistem temu kembali informasi (STKI). Pada penelitian ini membahas penerapan STKI untuk melakukan pencarian abstrak Tugas Akhir yang relevan sesuai dengan *query* yang dimasukan oleh pengguna. STKI Abstrak tugas akhir yang dikembangkan menggunakan metode *Vector Space Model* (VSM) dan metode pembobotan *tf-idf*, implementasi VSM dan metode *tf-idf* menggunakan bahasa pemrograman *server side* PHP dan Mysql sebagai DBMS untuk menyimpan informasi abstrak tugas akhir mahasiswa Prodi Teknik Informatika Unsoed.

**Kata kunci:** Sistem temu kembali informasi, *vector space model*, *tf-idf*, *php*, *mysql*

**A. PENDAHULUAN**

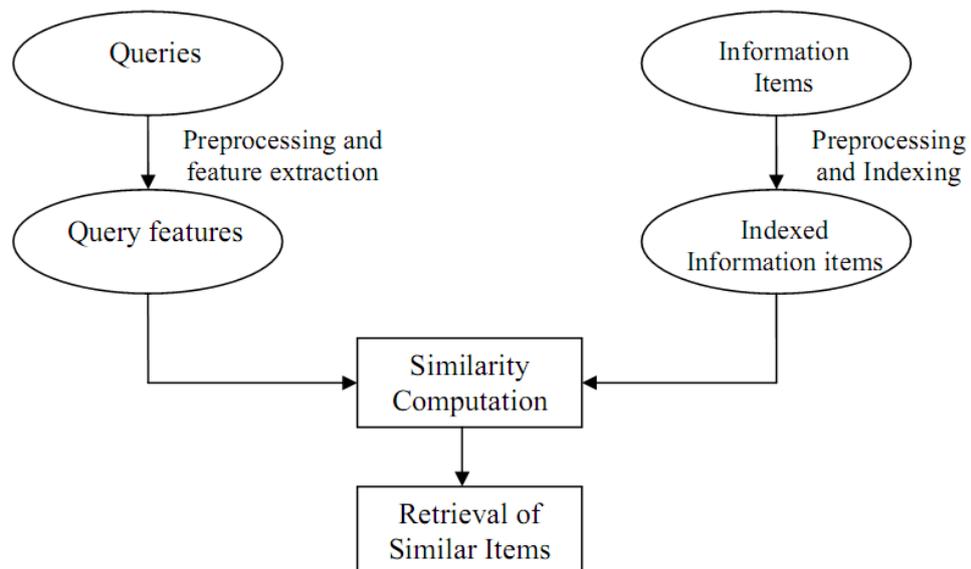
Teknologi informasi dan komunikasi pada era sekarang mengalami perkembangan pesat. Orang berlomba memanfaatkan TIK, TIK terutama *internet* telah digunakan sebagai alat untuk mengakses dan mendapatkan informasi (Afuan,2013). Permasalahan yang sering kali muncul dalam mengakses dan mendapatkan informasi adalah memilih informasi yang tepat sesuai dengan keinginan *user*. Untuk mengatasi masalah pencarian informasi, maka munculah sistem temu kembali informasi (STKI). STKI memungkinkan pengguna untuk mencari informasi yang tersimpan didalam dokumen secara efektif dan efisien. Efektif berarti user mendapatkan dokumen yang relevan dengan *query* yang diinputkan. Efisien berarti waktu pencarian yang sesingkat-singkatnya (Agusta, 2009).

Pada penelitian ini, penulis akan menjelaskan tahapan pengembangan Sistem temu kembali informasi yang digunakan untuk melakukan pencarian abstrak Tugas Akhir mahasiswa pada Program Studi Teknik Informatika Unsoed.

## B. TINJAUAN PUSTAKA

### 1. Sistem temu kembali Informasi

*Information Retrieval System (IR)* atau sistem temu kembali informasi merupakan proses menemukan data (umumnya dokumen) berupa teks yang sesuai dengan informasi yang dibutuhkan dari koleksi dokumen yang tersimpan didalam komputer (Manning, 2009). STKI menyediakan informasi mengenai subyek yang dibutuhkan. Data mencakup teks, tabel, gambar, ucapan, dan video. Informasi termasuk pengetahuan terkait yang dibutuhkan untuk mendukung penyelesaian masalah dan akuisisi pengetahuan (Cios, 2007). Sistem temu kembali informasi bertujuan untuk menghasilkan dokumen yang relevan dengan *query* yang dimasukan pengguna. Secara umum, STKI Abstrak tugas akhir mahasiswa TI Unsoed dibagi menjadi 2 tahapan yaitu tahap *preprocessing(indexing)* dan tahap *matching*. Arsitektur STKI dapat dilihat pada gambar 1



**Gambar 1.** Arsitektur STKI

Berdasarkan gambar 1, pada tahapan *preprocessing* terdiri atas proses sebagai berikut :

a. Tokenisasi

Yaitu mengubah dokumen menjadi kumpulan *term* dengan cara menghapus semua karakter dalam tanda baca yang terdapat pada dokumen abstrak TA dan mengubah kumpulan *term* menjadi huruf kecil semua (*lowercase*).

b. Stopword Removal

Proses penghapusan kata-kata yang sering ditampilkan dalam dokumen seperti: *pada, dengan, adalah, yaitu, yang, juga, dari, dia, kami, kamu, ini, itu, atau, dan, tersebut, keda*n sebagainya.

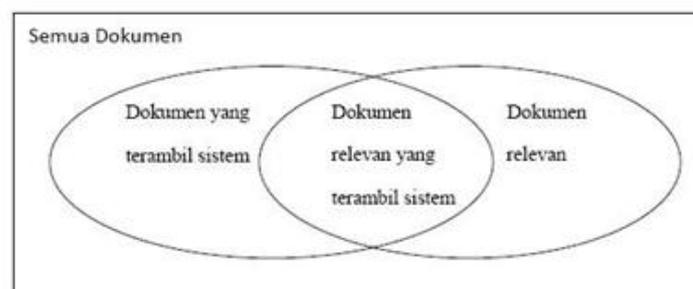
c. Stemming

Proses mengubah suatu kata bentukan menjadi kata dasar (*root word*).

d. Term Weighting (Pembobotan)

Proses pembobotan setiap *term* di dalam dokumen, proses pembobotan ini menggunakan metode *tf-idf*.

Tahapan *matching* (pencocokan) merupakan proses menemukan kembali informasi abstrak Tugas akhir yang relevan terhadap *query* yang diberikan. Tidak semua dokumen yang diambil (*retrieved*) oleh sistem merupakan dokumen abstrak yang sesuai dengan keinginan pengguna (*relevant*). Gambar 2 menunjukkan hubungan antara dokumen relevan, dokumen yang diambil oleh sistem, dan dokumen relevan yang diambil oleh system



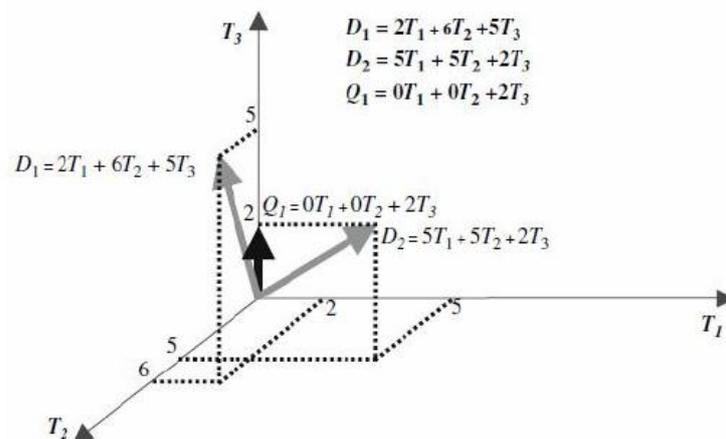
**Gambar 2.** *Matching* abstrak

## 2. Vector Space Model (VSM)

Kemiripan (*similarity*) antar dokumen didefinisikan berdasarkan representasi *bagof- words* dan dikonversi ke suatu model ruang vektor (*vector space model*, VSM). Model ini diperkenalkan oleh Salton (1983). Pada VSM, setiap dokumen di dalam database dan *query* pengguna direpresentasikan oleh suatu vektor multi-dimensi (Cios, 2007) dan (Poletti, 2004). Dimensi sesuai dengan jumlah *term* dalam dokumen yang terlibat. Pada model ini

- Vocabulary* merupakan kumpulan semua *term* berbeda yang tersisa dari dokumen setelah *preprocessing* dan mengandung *t term index*. Term-term ini membentuk suatu ruang vektor.
- Setiap *term* *i* di dalam dokumen atau *query* *j*, diberikan suatu bobot (*weight*) bernilai real  $w_{ij}$ .
- Dokumen dan *query* diekspresikan sebagai vektor *t* dimensi  $d_j = (w_1, w_2, \dots, w_{ij})$  dan terdapat *n* dokumen di dalam koleksi, yaitu  $j = 1, 2, \dots, n$ .

Contoh dari model ruang vektor tiga dimensi untuk dua dokumen  $D_1$  dan  $D_2$ , satu *query* pengguna  $Q_1$ , dan tiga term  $T_1$ ,  $T_2$  dan  $T_3$  diperlihatkan pada gambar 3



**Gambar 3.** Contoh VSM

Keberhasilan dari model VSM ini ditentukan oleh skema pembobotan terhadap suatu *term* baik untuk cakupan lokal maupun global, dan faktor normalisasi (Salton, 1983). Pembobotan lokal hanya berpedoman pada frekuensi munculnya

*term* dalam suatu dokumen dan tidak melihat frekuensi kemunculan *term* tersebut di dalam dokumen lainnya. Pembobotan global digunakan untuk memberikan tekanan terhadap *term* yang mengakibatkan perbedaan dan berdasarkan pada penyebaran dari *term* tertentu diseluruh dokumen. Bobot lokal suatu *term*  $i$  di dalam dokumen  $j$  ( $tf_{ij}$ ) dapat didefinisikan pada persamaan 1 sebagai berikut

$$tf_{ij} = \frac{f_{ij}}{\max_i(f_{ij})}$$

Bobot global dari suatu *term*  $i$  pada pendekatan *inverse document frequency* (*idf<sub>i</sub>*) dapat didefinisikan pada persamaan 2 sebagai berikut

$$idf_i = \log_2\left(\frac{n}{df_i}\right)$$

Bobot dari *term*  $i$  di dalam sistem IR ( $w_{ij}$ ) dihitung menggunakan ukuran *tf-idf* yang didefinisikan pada persamaan 3 (cios,2007) dan ( Lee,1997) sebagai berikut

$$w_{ij} = tf_{ij} \times idf_i$$

### 3. Similarity (ukuran kemiripan)

Salah satu ukuran kemiripan teks yang populer (Tata,2007) adalah *cosine similarity*. Ukuran ini menghitung nilai cosinus sudut antara dua vektor. Jika terdapat dua vektor dokumen  $d_j$  dan query  $q$ , serta  $t$  *term* diekstrak dari koleksi dokumen maka nilai cosinus antara  $d_j$  dan  $q$  didefinisikan pada persamaan 4 (Cios,1997)

$$similarity(\vec{d}_j, \vec{q}) = \frac{\vec{d}_j \cdot \vec{q}}{|\vec{d}_j| \cdot |\vec{q}|} = \frac{\sum_{i=1}^t (w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^t w_{ij}^2 \cdot \sum_{i=1}^t w_{iq}^2}}$$

### C. METODOLOGI PENELITIAN

Metode Penelitian yang digunakan dalam penelitian antara lain :

#### 1. Studi Pustaka

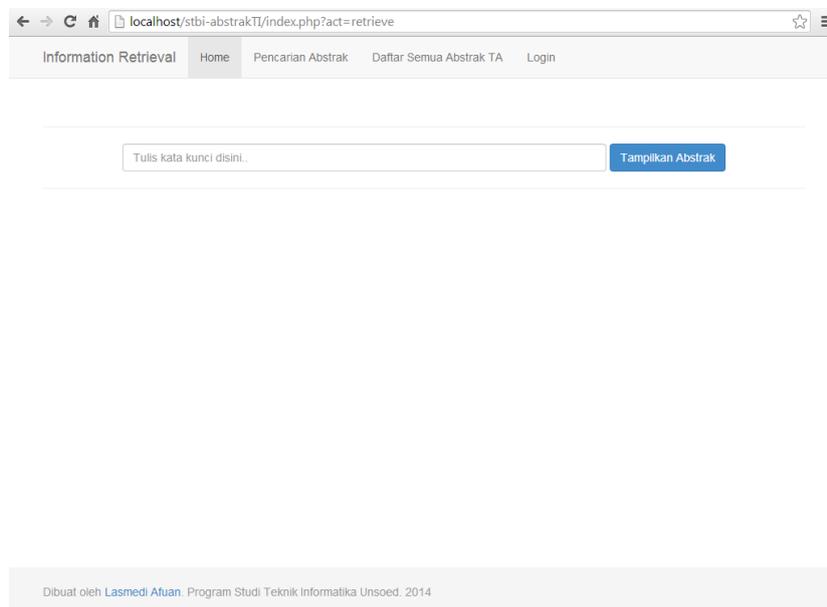
Studi pustaka dilakukan terkait dengan pengumpulan literatur, pustaka mengenai *information retrieval*, Metode Vector Space Model dan metode tf-idf.

#### 2. Ujicoba

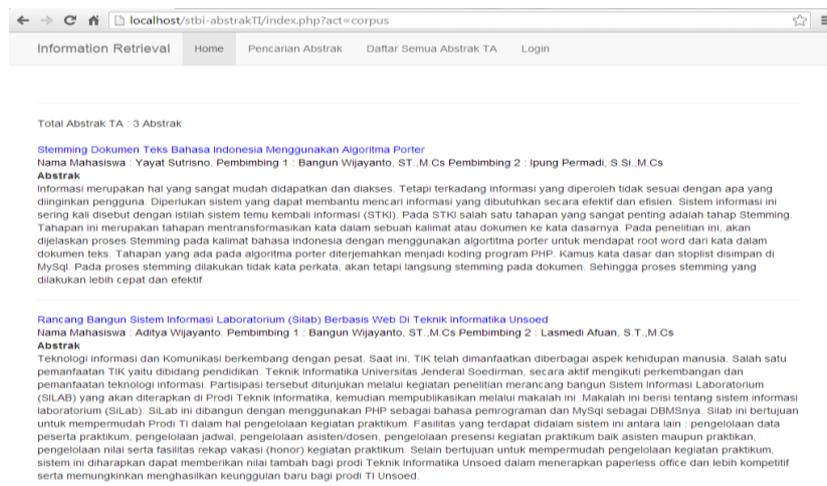
Tahapan ujicoba dilakukan mencoba STKI dengan menggunakan data abstrak tugas akhir mahasiswa pada Prodi Teknik Informatika.

### D. HASIL DAN PEMBAHASAN

Pada gambar 4, merupakan halaman awal dari aplikasi. Pada halaman awal ini terdapat 4 menu yaitu home, Pencarian abstrak, Daftar semua abstrak dan menu login. Pada halaman awal, menampilkan halaman pencarian abstrak, pengguna dapat mulai melakukan *retrieve* abstrak dengan mengisikan *keyword* pencarian pada inputan yang sudah disediakan.



**Gambar 4.** Halaman Awal STKI Abstrak



Gambar 5. Halaman daftar semua abstrak TA

Gambar 5 merupakan halaman yang menampilkan semua daftar abstrak mahasiswa yang ada di Program Studi Teknik Informatika. Pada halaman ini abstrak ditampilkan sebanyak 4 baris untuk setiap halamannya, jika pengguna ingin melihat daftar abstrak yang lainnya dapat dilakukan dengan mengklik halaman selanjutnya.

No	Term	Abstrak ID	Count	Nilai Bobot
1	informasi	1	5	0
2	merupakan	1	2	2.19722
3	hal	1	1	0.405465
4	sangat	1	2	0.81093
5	mudah	1	1	1.09861
6	didapatkan	1	1	1.09861
7	kses	1	1	1.09861
8	tetapi	1	2	0.81093
9	terkag	1	1	1.09861
10	diperoleh	1	1	1.09861
11	tidak	1	2	0.81093
12	sesuai	1	1	1.09861
13	apa	1	1	1.09861
14	ditinginkan	1	1	1.09861
15	perguruan	1	1	1.09861

Gambar 6. Halaman daftar index

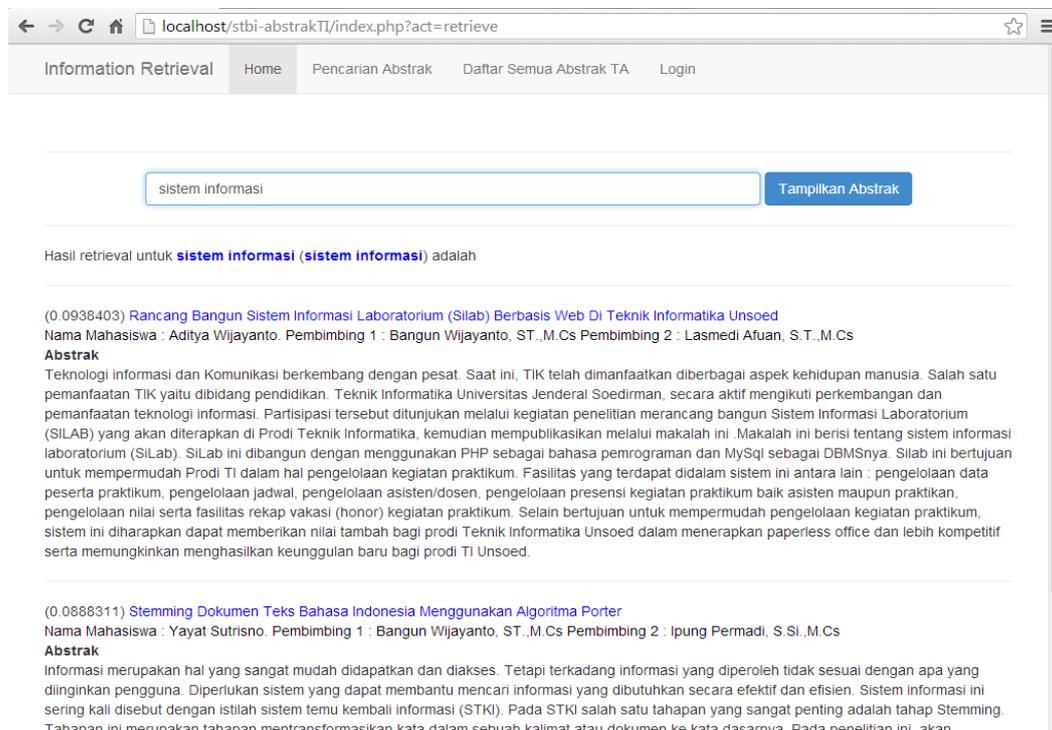
Gambar 6 merupakan halaman hasil perhitungan *index*, halaman ini merupakan penerapan dari persamaan 1,2 dan 3. Pada halaman ini menampilkan *term* frekuensi pada setiap abstrak, dan kemudian dilakukan penghitungan bobot dengan menggunakan persamaan yang sudah dijelaskan sebelumnya, gambar 6 merupakan bagian dari tahapan *preprocessing* dalam IR sistem.

## 1. Hasil Pengujian

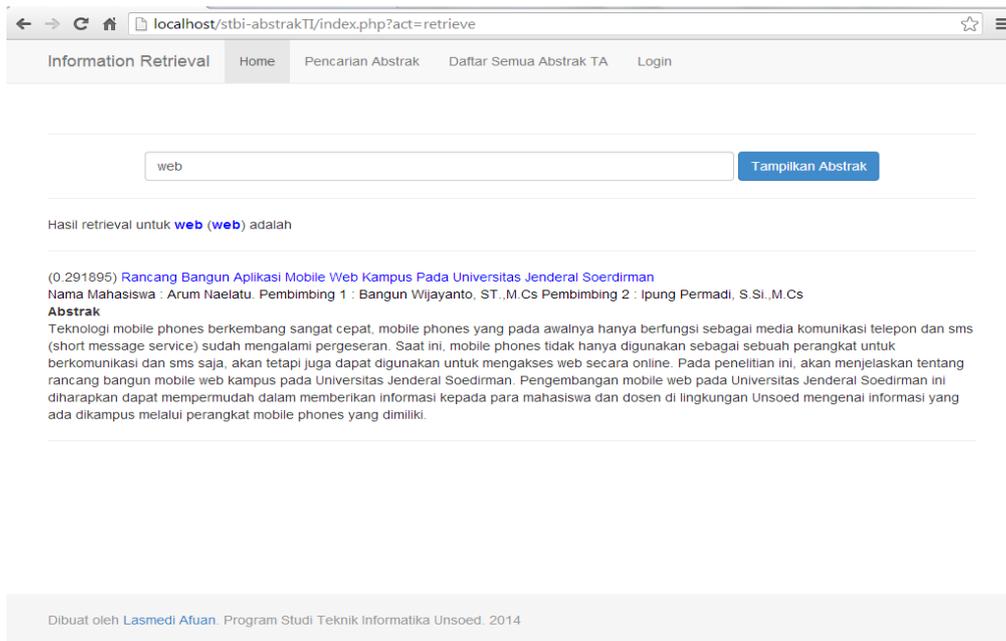
Pengujian sistem ini, dilakukan dengan menggunakan data abstrak tugas akhir mahasiswa di Program Studi Teknik Informatika, pada pengujian ini baru menggunakan 3 data abstrak tugas akhir mahasiswa.

## 2. Pengujian (dokumen teks)

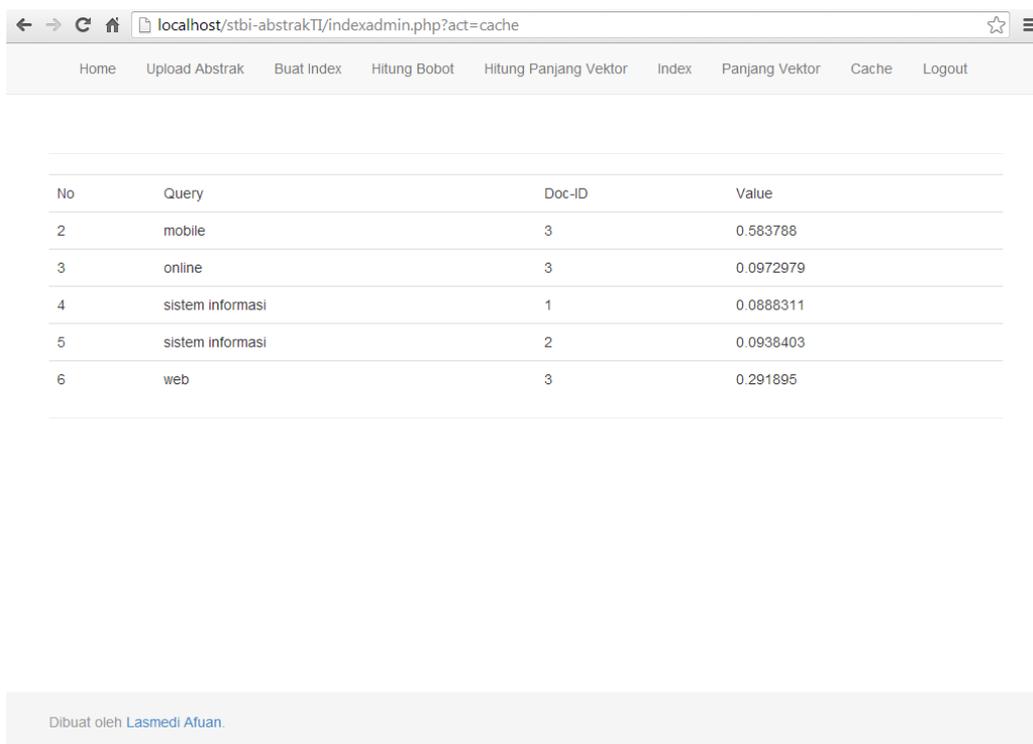
Gambar 6 dan gambar 7 merupakan halaman proses retrieval informasi, pada sistem dicoba memasukkan *keyword* sistem informasi, dan sistem menampilkan 2 abstrak yang terkait dengan sistem informasi. Sedangkan, ketika menggunakan keyword **web**, sistem meretrieve 1 abstrak.



**Gambar 7.** Halaman retrieval (*matching*)



**Gambar 8.** Halaman retrieval(*matching*)



**Gambar 8.** Cache

Gambar 8 merupakan cache *query* yang sudah pernah dimasukkan oleh pengguna, pada gambar 8 dapat dilihat keterkaitan query dengan abstrak yang

sudah dilakukan proses pembobotan, dan pada bagian kanan terdapat kolom value yang merupakan nilai similarity query dan dokumen abstrak.

## E. KESIMPULAN DAN SARAN

Setelah dilakukan implementasi, VSM dan metode tf-idf dapat digunakan dalam pengembangan sistem temu kembali informasi pada abstrak tugas akhir mahasiswa di Prodi Teknik Informatika. Hanya saja, pada sistem yang sudah dibuat, pada tahapan *stopword removal* perludibuat tabel atau file yang menampung data stopwords. Sebagai saran, sistem ini hendaknya dapat melakukan retrieve dokumen dalam berbagai format.

## DAFTAR PUSTAKA

- Agusta Ledy, 2009. Perbandingan Algoritma Stemming Porter Dengan Algoritma Nazief &Adriani Untuk Stemming Dokumen Teks Bahasa Indonesia. KNSI 2009, Bali .
- Afuan Lasmedi, 2013. Stemming Dokumen Teks Bahasa Indonesia Menggunakan Algoritma Porter. STI UAD 2013, Yogyakarta.
- Cios, Krzysztof J. Etc. 2007.*Data Mining A Knowledge Discovery Approach*, Springer
- Lee D.L. 1997. Document Ranking and the Vector-Space Model. *IEEE March-April 1997*.
- Manning D Christopher. 2009. *An Introduction to Information Retrieval*.Online Edition. Cambridge Up.
- Polettini, Nicola .2004. *The Vector Space Model in Information Retrieval – TermWeighting Problem*
- Tata, Sandeep, Patel M, Jignesh .2007. *Estimating he Selectivity of tf-idf based Cosine Similarity Predicates*, Sigmod Record December 2007 Vol 36 No. 4
- Salton, Gerard. 1983. *Introduction to Modern Information Retrieval*. McGraw Hill.