BAB IV
IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan membahas implementasi dan pengujian terhadap sistem yang dibangun. Tahap ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman yang digunakan. Setelah implementasi dilakukan, selanjutnya akan dilakukan pengujian sistem dimana akan dilihat kekurangan-kekurangan pada sistem yang dibangun untuk pengembangan sistem selanjutnya. Pengujian yang dilakukan ada tiga, yaitu pengujian *black box*, pengujian akurasi, dan pengujian *beta*.

IV.1 Implementasi Sistem

Tahap implementasi sistem ini bertujuan untuk menerapkan modul perancangan yang telah dibuat pada bab sebelumnya ke sistem agar bisa digunakan atau dioperasikan. Tahap implementasi sistem ini dibagi menjadi dua, yaitu implementasi basis data dan implementasi antarmuka.

IV.1.1 Implementasi Basis Data

Implementasi ini merupakan tahap untuk pembuatan *database* yang sebelumnya telah dirancang dimana akan diimplementasikan secara langsung pada sistem yang dibangun. Implementasi *database* dalam SQL adalah sebagai berikut :

Tabel IV.1 Implementasi Basis Data

No	Fungsi	*Syntax*
1.	Pembuatan *database*	CREATE DATABASE dbaggregator
2.	Pembuatan tabel *categories*	CREATE TABLE `categories` (`category_id` int(11) NOT NULL AUTO_INCREMENT, `category` text, `parent_num` int(11) DEFAULT NULL, PRIMARY KEY (`category_id`)) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;
3.	Pembuatan tabel *links*	CREATE TABLE `links` (`link_id` int(11) NOT NULL AUTO_INCREMENT,

No	Fungsi	*Syntax*
		`site_id` int(11) DEFAULT NULL, `url` varchar(255) NOT NULL, `title` varchar(200) DEFAULT NULL, `description` varchar(255) DEFAULT NULL, `fulltxt` mediumtext, `indexdate` date DEFAULT NULL, `size` float DEFAULT NULL, `md5sum` varchar(32) DEFAULT NULL, `visible` int(11) DEFAULT '0', `level` int(11) DEFAULT NULL, PRIMARY KEY (`link_id`), KEY `site_id` (`site_id`)) ENGINE=InnoDB AUTO_INCREMENT=77 DEFAULT CHARSET=latin1;
4.	Pembuatan tabel *pending*	CREATE TABLE `pending` (`site_id` int(11) DEFAULT NULL, `temp_id` varchar(32) DEFAULT NULL, `level` int(11) DEFAULT NULL, `count` int(11) DEFAULT NULL, `num` int(11) DEFAULT NULL, KEY `site_id` (`site_id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
5.	Pembuatan tabel *sites*	CREATE TABLE `sites` (`site_id` int(11) NOT NULL AUTO_INCREMENT, `url` varchar(255) DEFAULT NULL, `title` varchar(255) DEFAULT NULL, `short_desc` text, `indexdate` date DEFAULT NULL, `spider_depth` int(11) DEFAULT '2', `required` text, `disallowed` text, `can_leave_domain` tinyint(1) DEFAULT NULL, PRIMARY KEY (`site_id`)) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;
6.	Pembuatan tabel *site category*	CREATE TABLE `site_category` (`site_id` int(11) DEFAULT NULL, `category_id` int(11) DEFAULT NULL, KEY `category_id` (`category_id`), KEY `site_id` (`site_id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
7.	Pembuatan tabel berita	CREATE TABLE `tbberita` (`DocId` int(11) NOT NULL AUTO_INCREMENT, `Judul` varchar(100) NOT NULL, `Berita` varchar(255) NOT NULL, `url` varchar(255) NOT NULL, `link_id` int(11) DEFAULT NULL, PRIMARY KEY (`DocId`), KEY `tbberita_ibfk_1` (`link_id`)) ENGINE=InnoDB AUTO_INCREMENT=262 DEFAULT CHARSET=latin1;
8.	Pembuatan tabel *cache*	CREATE TABLE `tbcentroid` (`idCentroid` int(11) NOT NULL AUTO_INCREMENT,

No	Fungsi	*Syntax*
		`Centroid` float NOT NULL, `Cluster` int(11) NOT NULL, `IdTerm` int(11) NOT NULL, `Term` varchar(30) NOT NULL, `ClusterId` int(11) DEFAULT NULL, PRIMARY KEY (`idCentroid`), KEY `ClusterId` (`ClusterId`)) ENGINE=InnoDB AUTO_INCREMENT=57825 DEFAULT CHARSET=latin1;
9.	Pembuatan tabel *cluster*	CREATE TABLE `tbcluster_doc` (`ClusterId` int(11) NOT NULL AUTO_INCREMENT, `DocId` int(11) NOT NULL, `Cluster` int(11) NOT NULL, PRIMARY KEY (`ClusterId`), KEY `DocId` (`DocId`)) ENGINE=InnoDB AUTO_INCREMENT=124 DEFAULT CHARSET=latin1;
10.	Pembuatan tabel *index*	CREATE TABLE `tbindex` (`Id` int(11) NOT NULL AUTO_INCREMENT, `Term` varchar(30) NOT NULL, `DocId` int(11) NOT NULL, `Count` int(11) NOT NULL, `Bobot` float NOT NULL, `Id_stem` int(11) DEFAULT NULL, PRIMARY KEY (`Id`), KEY `DocId` (`DocId`), KEY `tbindex_ibfk_2` (`Id_stem`)) ENGINE=InnoDB AUTO_INCREMENT=1943 DEFAULT CHARSET=latin1;
11.	Pembuatan tabel stem	CREATE TABLE `tbstem` (`Id_stem` int(11) NOT NULL AUTO_INCREMENT, `Term` varchar(30) NOT NULL, `Stem` varchar(30) NOT NULL, PRIMARY KEY (`Id_stem`)) ENGINE=InnoDB AUTO_INCREMENT=6037 DEFAULT CHARSET=latin1;
12.	Pembuatan tabel vektor	CREATE TABLE `tbvektor` (`DocId` int(11) NOT NULL, `Panjang` float NOT NULL, KEY `fk_DocId_vektor` (`DocId`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

IV.1.2 Implementasi Antarmuka

Implementasi antarmuka bertujuan untuk mengetahui setiap tampilan yang telah dibangun dan pengkodeannya ke dalam bentuk file program, sehingga akan diketahui apakah sistem yang telah dibuat benar-benar sesuai dengan yang direncanakan. Implementasi antarmuka dari sistem *news aggregator* ini dapat dilihat pada .

Tabel IV.2 Implementasi Antarmuka

No	Nama Antarmuka	Deskripsi	Nama *File*
1.	Home	Antarmuka awal apabila data inputan belum dimasukkan	admin.php ClassAdmin.php
2.	Situs	Antarmuka untuk menampilkan situs-situs yang telah dimasukkan ke sistem	
3.	Tambah Situs	Antarmuka untuk menambahkan data situs	
4.	Submenu Pilihan	Antarmuka untuk menampilkan sub menu pilihan dari setiap situs	
5.	Edit Situs	Antarmuka untuk mengedit data situs	
6.	*Crawling*	Antarmuka untuk mengatur proses *crawling*	
7.	Hapus Situs	Antarmuka untuk menghapus data situs	
8.	Kategori	Antarmuka untuk menampilkan data kategori	
9.	Tambah Kategori	Antarmuka untuk menambah data kategori	
10.	Edit Kategori	Antarmuka untuk mengedit data kategori	
11.	Hapus Kategori	Antarmuka untuk menghapus kategori	
12.	Proses *Crawling*	Antarmuka untuk menampilkan hasil proses *crawling*	crawling.php ClassIndexCrawling.php ClassIR.php messages.php
13.	Aggregator	Antarmuka untuk menampilkan kumpulan informasi berita	aggregator.php ClassAggregator.php
14.	Hasil Pencarian	Antarmuka untuk menampilkan hasil pencarian	
15.	Kategori Situs Berita	Antarmuka untuk menampilkan informasi berita berdasarkan kategori	

IV.2 Pengujian Sistem

Pengujian sistem merupakan tahapan untuk menemukan kesalahan-kesalahan dan kekurangan pada sistem yang dibangun sehingga bisa diketahui sistem tersebut telah memenuhi kriteria yang sesuai dengan tujuannya atau tidak.

Pengujian sistem tersebut akan dilakukan tiga macam, yaitu pengujian *blackbox* untuk mengetahui apakah setiap fungsional bekerja sesuai dengan yang diharapkan; pengujian akurasi yang digunakan untuk mengetahui akurasi dari hasil penerapan metode yang digunakan ke dalam sistem; dan pengujian *beta* yang digunakan untuk menguji fungsional sistem secara langsung ke pengguna sistem.

IV.2.1 Pengujian *Black box*

Pengujian *black box* merupakan pengujian terhadap fungsional dari sistem yang dibangun. Pengujian *black box* yang dilakukan pada sistem ini dibagi tiga bagian, yaitu skenario pengujian, kasus dan hasil pengujian, dan kesimpulan pengujian.

IV.2.1.1 Skenario Pengujian

Skenarion pengujian digunakan untuk menjelaskan pengujian terhadap suatu sistem. Skenario ini memaparkan urutan dan hal yang akan diuji yang dilakukan pada sistem *news aggregator*. Adapun skenario pengujian yang dilakukan pada sistem ini yang dapat dilihat pada Tabel IV.3.

Tabel IV.3 Skenario Pengujian

Komponen yang diuji	Poin Pengujian	Jenis Pengujian
Tambah Situs	Menambah data situs	*Black box*
Edit Situs	Mengedit data situs	*Black box*
Tambah Kategori	Menambah data kategori	*Black box*
Edit Kategori	Mengdit data kategori	*Black box*
Crawling Semua Situs Berita	Melakukan *crawling* semua situs	*Black box*
Crawling Per Situs Berita	Melakukan *crawling* per situs	*Black box*
Pencarian Informasi Berita	Melakukan pencarian informasi berita	*Black box*

IV.2.1.2 Kasus dan Hasil Pengujian

Kasus dan hasil pengujian ini membahas tentang pemaparan dari rencana pengujian yang telah disusun pada skenario pengujian. Pengujian *black box* ini hanya memperhatikan masukan dan keluaran dari sistem *news aggregator*.

Berikut ini pemaparan dari setiap poin pengujian yang terdapat pada skenario pengujian :

1. Pengujian Tambah Situs

   Pengujian tambah situs memaparkan hasil pengujian yang dilakukan oleh pengguna saat akan melakukan menambah situs baru ke sistem. Terdapat dua kondisi pada pengujian ini, yaitu ketika pengguna memasukkan data dengan benar dan ketika pengguna membiarkan *field* kosong.

Tabel IV.4 Pengujian Tambah Situs

Kasus dan Hasil Uji (data benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
URL : http://www.repunlika.co.id/berita/ Judul : Republika	Ketika menekan tombol Tambah, maka data yang ditambah akan tersimpan ke dalam *database*	Setelah mengisi data masukan dan menekan tombol Tambah, data tersimpan ke *database*	☑ Berhasil ☐ Tidak Berhasil
Kasus dan Hasil Uji (data kosong)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
URL : (kosong)	Ketika menekan tombol Tambah, sistem akan menampilkan pesan validasi kesalahan P05 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P05	☑ Berhasil ☐ Tidak Berhasil
Judul : (kosong)	Ketika menekan tombol Tambah, sistem akan menampilkan pesan validasi kesalahan P05 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P05	☑ Berhasil ☐ Tidak Berhasil
URL : (kosong) Judul : (kosong)	Ketika menekan tombol Tambah, sistem akan menampilkan pesan validasi kesalahan P05 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P05	☑ Berhasil ☐ Tidak Berhasil

2. Pengujian Edit Situs

Pengujian edit situs memaparkan hasil pengujian yang dilakukan oleh pengguna saat akan melakukan perubahan terhadap situs yang sudah ada di sistem. Terdapat dua kondisi pada pengujian ini, yaitu ketika pengguna memasukkan data dengan benar dan ketika pengguna membiarkan *field* kosong.

Tabel IV.5 Pengujian Edit Situs

Kasus dan Hasil Uji (data benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
URL : http://www.tribunnews.com/nasional/ Judul : Tribun Dept/Level : 1	Ketika menekan tombol Update, maka data yang diubah akan tersimpan ke dalam *database*	Setelah mengubah data masukan dan menekan tombol Update, data tersimpan ke *database*	☑ Berhasil ☐ Tidak Berhasil
Kasus dan Hasil Uji (data kosong)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
URL : (kosong)	Ketika menekan tombol Update, sistem akan menampilkan pesan validasi kesalahan P06 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P06	☑ Berhasil ☐ Tidak Berhasil
Judul : (kosong)	Ketika menekan tombol Update, sistem akan menampilkan pesan validasi kesalahan P06 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P06	☑ Berhasil ☐ Tidak Berhasil
Dept/Level : 0	Ketika menekan tombol Update, sistem akan menampilkan pesan validasi kesalahan P06 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P06	☑ Berhasil ☐ Tidak Berhasil

URL : (kosong) Judul : (kosong) Dept/Level : 0	Ketika menekan tombol Update, sistem akan menampilkan pesan validasi kesalahan P06 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P06	☑ Berhasil ☐ Tidak Berhasil

3. Pengujian Tambah Kategori

Pengujian tambah kategori memaparkan hasil pengujian yang dilakukan oleh pengguna saat akan melakukan menambah kategori baru ke sistem. Terdapat dua kondisi pada pengujian ini, yaitu ketika pengguna memasukkan data dengan benar dan ketika pengguna membiarkan *field* kosong.

Tabel IV.6 Pengujian Tambah Kategori

Kasus dan Hasil Uji (data benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Kategori : Detik	Ketika menekan tombol Tambah, maka data yang ditambah akan tersimpan ke dalam *database*	Setelah mengisi data masukan dan menekan tombol Tambah, data tersimpan ke *database*	☑ Berhasil ☐ Tidak Berhasil
Kasus dan Hasil Uji (data kosong)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Kategori : (kosong)	Ketika menekan tombol Tambah, sistem akan menampilkan pesan validasi kesalahan P07 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P07	☑ Berhasil ☐ Tidak Berhasil

4. Pengujian Edit Kategori

Pengujian edit situs memaparkan hasil pengujian yang dilakukan oleh pengguna saat akan melakukan perubahan terhadap kategori yang sudah ada di sistem. Terdapat dua kondisi pada pengujian ini, yaitu ketika pengguna memasukkan data dengan benar dan ketika pengguna membiarkan *field* kosong.

Tabel IV.7 Pengujian Edit Kategori

Kasus dan Hasil Uji (data benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Kategori : Detik	Ketika menekan tombol Update, maka data yang diubah akan tersimpan ke dalam *database*	Setelah mengubah data masukan dan menekan tombol Update, data tersimpan ke *database*	☑ Berhasil ☐ Tidak Berhasil
Kasus dan Hasil Uji (data kosong)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Kategori : (kosong)	Ketika menekan tombol Update, sistem akan menampilkan pesan validasi kesalahan P08 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P08	☑ Berhasil ☐ Tidak Berhasil

5. Pengujian *Crawling* Semua Situs Berita

Pengujian *crawling* semua situs berita memaparkan hasil pengujian yang dilakukan oleh pengguna saat akan melakukan pengambilan *link* informasi berita yang diambil dari situs-situs berita. Terdapat dua kondisi pada pengujian ini, yaitu ketika sistem terhubung dengan jaringan internet dan ketika sistem tidak terhubung dengan jaringan internet.

Tabel IV.8 Pengujian *Crawling* Semua Situs Berita

Kasus dan Hasil Uji (terhubung dengan jaringan internet)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak ada	Ketika menekan *link Crawling* Semua, maka sistem akan memulai mengambil *link-link* informasi berita yang diambil dari semua situs	Terdapat kumpulan *link* informasi berita	☑ Berhasil ☐ Tidak Berhasil
Kasus dan Hasil Uji (tidak terhubung dengan jaringan internet)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak ada	Ketika menekan *link Crawling* Semua, maka sistem akan menampilkan pesan validasi kesalahan P09 yang seperti pada	Sistem menampilkan pesan validasi kesalahan P09	☑ Berhasil ☐ Tidak Berhasil

	Error! Reference source not found.		

6. Pengujian *Crawling* Per Situs Berita

Pengujian *crawling* per situs berita memaparkan hasil pengujian yang dilakukan oleh pengguna saat akan melakukan pengambilan *link* informasi berita yang diambil dari satu situs berita. Terdapat dua kondisi pada pengujian ini, yaitu ketika sistem terhubung dengan jaringan internet dan ketika sistem tidak terhubung dengan jaringan internet.

Tabel IV.9 Pengujian *Crawling* Per Situs Berita

Kasus dan Hasil Uji (terhubung dengan jaringan internet)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak ada	Ketika menekan tombol Mulai *Crawling*, maka sistem akan memulai mengambil *link-link* informasi berita yang diambil dari situs berita yang di *crawling*	Terdapat kumpulan *link* informasi berita	☑ Berhasil ☐ Tidak Berhasil
Kasus dan Hasil Uji (tidak terhubung dengan jaringan internet)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak ada	Ketika menekan tombol Mulai *Crawling*, maka sistem akan menampilkan pesan validasi kesalahan P09 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P09	☑ Berhasil ☐ Tidak Berhasil

7. Pengujian Pencarian Informasi Berita

Pengujian pencarian informasi berita memaparkan hasil pengujian yang dilakukan oleh pengguna saat akan mencari informasi berita dari hasil *crawling* dengan memasukkan kata kunci. Terdapat dua kondisi pada pengujian ini, yaitu ketika sistem menemukan informasi berita berdasarkan kata kunci yang dimasukkan dan ketika sistem tidak menemukan informasi berita berdasarkan kata kunci yang dimasukkan.

Tabel IV.10 Pengujian Pencarian Informasi Berita

Kasus dan Hasil Uji (menemukan informasi berita)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Kata kunci : pendidikan	Ketika mengklik tombol Cari, maka sistem akan menampilkan informasi berita dari semua situs	Setelah memasukkan kata kunci dan menekan tombol Cari, sistem menampilkan informasi berita	☑ Berhasil ☐ Tidak Berhasil
Kasus dan Hasil Uji (tidak menemukan informasi berita)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Kata kunci : (kosong)	Ketika mengklik tombol Cari, maka sistem akan menampilkan pesan validasi kesalahan P10 yang seperti pada **Error! Reference source not found.**	Sistem menampilkan pesan validasi kesalahan P10	☑ Berhasil ☐ Tidak Berhasil

IV.2.1.3 Kesimpulan Pengujian *Black box*

Berdasarkan hasil pengujian yang dilakukan maka dapat disimpulkan bahwa sistem *news aggregator* yang dibangun sudah bekerja cukup baik dan sesuai dengan yang diharapkan, tetapi tidak menutup kemungkinan dapat terjadi kesalahan suatu saat pada saat sistem digunakan karena sistem ini masih berupa *prototype*, sehingga membutuhkan proses evaluasi yang lebih lanjut untuk mengetahui kekurangan dari sistem ini.

IV.2.2 Pengujian Akurasi

Pengujian akurasi *clustering* informasi berita dilakukan untuk mengetahui tingkat akurasi pengelompokkan informasi berita pendidikan yang dilakukan oleh algoritma K-Means pada sistem *news aggregator* dengan pengelompokkan yang dilakukan secara manual. Pengujian dilakukan dengan menggunakan 132 dokumen yang diambil dari hasil *crawling*, dimana dari hasil *crawling* tersebut ditemukan informasi berita yang dengan kata kunci "pendidikan" dan telah dilakukan proses *clustering* sbelemunya dengan *k* ditentukan secara dinamis yang dapat dilihat pada Tabel IV.11.

Tabel IV.11 Hasil Pengelompokkan informasi berita pendidikan

	Jumlah Informasi Pendidikan dalam 1 *Cluster*	Jumlah Informasi Pendidikan Bukan dalam 1 *Cluster*
News Aggregator	8	124
Manual	6	126

Adapun tabel *confusion matrix* dimana untuk membandingkan hasil prediksi atau yang dilakukan dengan sistem dengan yang dilakukan secara manual yang dapat dilihat pada Tabel IV.12.

Tabel IV.12 *Confusion Matrix*

Predicted Class / Actual Class		Manual	
		Cluster	Bukan *Cluster*
Sistem	*Cluster*	6	2
	Bukan *Cluster*	0	124

Setelah sistem melakukan *clustering*, lalu hitung nilai akurasinya berdasarkan pada Persamaan II-1.

$$\text{Akurasi} = \frac{6+124}{6+2+0+124} = 0.98$$

Berdasarkan pengujian akurasi, didapatkan hasil akurasi *clustering* berita dengan kata kunci "pendidikan" pada sistem *news aggregator* dengan menggunakan algoritma K-Means sebesar 98%. Kesimpulan yang diperoleh dari pengujian akurasi ini adalah bahwa algoritma K-Means dapat digunakan sebagai metode pengelompokkan untuk topik berita tertentu dari hasil *crawling* situs berita karena tingkat akurasinya yang besar, hanya saja kekurangan dari algoritma K-Means ini masih terdapat *outlier* atau dokumen yang seharusnya tidak dalam satu *cluster*.

IV.2.3 Pengujian Beta

Pengujian beta dilakukan secara langsung terhadap pengguna sistem dengan melakukan wawancara kepada pengguna terhadap sistem yang telah

dibangun. Wawancara dilakukan kepada 10 orang pengguna yang akan diajukan beberapa pertanyaan. Adapun pertanyaan-pertanyaan yang diajukan sebagai berikut :

a. Menurut anda apakah hasil pencarian yang dihasilkan oleh sistem *news aggregator* yang saya bangun sesuai dengan inputan kata kunci bila dibandingkan *news aggregator* yang pernah anda coba sebelumnya?
b. Apakah pengelompokkan informasi berita dari hasil pencarian sudah sesuai dengan topiknya (pendidikan)?

Berdasarkan dari wawancara yang dilakukan terhadap 10 pengguna tersebut, rata-rata dari 10 pengguna tersebut menjawab :

a. Sudah sesuai bila dibandingkan pencarian yang dilakukan di sistem *news aggregator* yang lain dengan inputan kata kunci yang sama.
b. Cukup sesuai, karena dari hasil pengelompokkan tersebut ada beberapa bukan topik pendidikan meskipun dari judul merujuk informasi berita pendidikan tapi dari isi beritanya ternyata bukan tentang pendidikan.

Berdasarkan hasil dari pengujian beta di atas, dapat disimpulkan bahwa sistem *news aggregator* ini bekerja dengan baik dalam melakukan proses menampilkan informasi berita tertentu yang telah dikelompokkan sebelumnya berdasarkan inputan kata kunci pengguna. Selain itu juga sistem ini juga mendapat masukan dala segi tampilan yang terlalu sedeharna, meski begitu dari fungsionalnya sistem ini bekerja dengan cukup baik.