

BAB IV

ANALISA DAN PERANCANGAN

Sebelum masuk dalam tahap pembuatan aplikasi, maka terlebih dahulu perlu dilakukan analisa dan perancangan terhadap aplikasi yang akan dibuat. Tahap analisa merupakan tahapan yang sangat penting dalam pembuatan aplikasi, karena pada tahap ini dilakukan identifikasi permasalahan, kebutuhan sistem dan penyelesaian masalah. Sedangkan tahap perancangan merupakan tahap pembuatan rincian aplikasi dari hasil analisa menjadi bentuk perancangan yang dapat dimengerti oleh pengguna.

4.1 Analisa Kebutuhan Data

Data merupakan bahan mentah yang akan diolah untuk menghasilkan sebuah informasi. Beberapa data yang dibutuhkan pada pembuatan aplikasi *clustering* dokumen ini adalah sebagai berikut:

1. Data Koleksi Dokumen/Korpus

Data Koleksi dokumen adalah data inputan yang berisi kumpulan dokumen teks. Selanjutnya masing-masing dokumen yang ada dalam koleksi dokumen ini akan diproses ke dalam 4 tahapan *text mining* hingga memberikan *output* berupa dokumen dokumen teks yang telah terkluster/terkelompok.

2. Data Jumlah Kelompok

Data jumlah kelompok adalah data inputan yang berupa jumlah kelompok/kluster yang ingin dibentuk oleh pengguna terhadap koleksi dokumen.

Pada bab analisa dan perancangan ini, data koleksi dokumen yang digunakan adalah sebagai berikut :

Tabel 4.1 Data Koleksi Dokumen

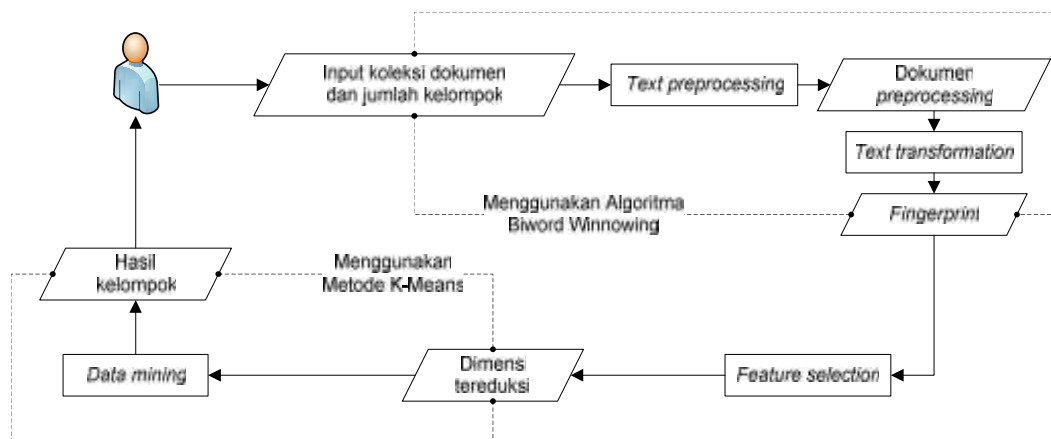
Dokumen	Isi Dokumen
1	Kata wahyu berasal dari bahasa Arab yakni Al-Wahyu, di mana Menurut pengertian bahasa kata Al-Wahyu mengandung arti suara, api, dan kecepatan.
2	Makna kata wahyu secara konstektual menunjukkan banyak makna. kata Al-Wahyu mengandung arti suara, api, dan

	kecepatan. Di samping itu ia pun mempunyai arti bisikan.
3	Jaringan komputer merupakan sejumlah komputer yang saling terhubung yang didesain AGAR dapat saling berbagi sumber daya dan bertukar “informasi”.
4	jaringan komputer merupakan sesuatu sangat penting, dengan adanya Jaringan KOMPUTER sejumlah komputer dapat saling terhubung hingga dapat saling berbagi ‘sumber daya’ dan bertukar informasi.

Dengan jumlah kelompok/kluster yang akan dibentuk adalah 2 kelompok/kluster.

4.2 Analisa Penyelesaian Masalah

Untuk memperjelas proses yang terjadi pada aplikasi *clustering* dokumen ini dapat digambarkan melalui gambar 4.1 sebagai berikut :



Gambar 4.1 Gambaran Umum Aplikasi *Clustering* Dokumen

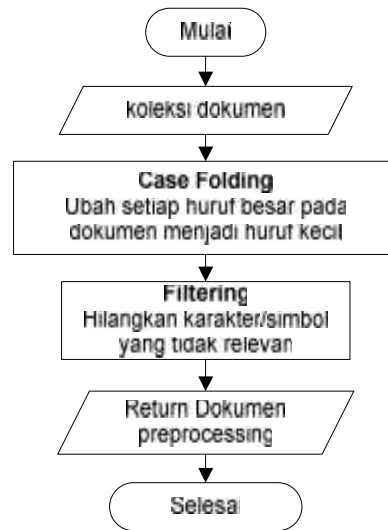
Gambar 4.1 diatas menggambarkan proses yang dilakukan oleh aplikasi *clustering* dokumen teks. Secara garis besar ada 4 tahapan yang akan dilakukan oleh aplikasi *clustering* ini, yaitu sebagai berikut :

- Text Preprocessing*
- Text transformation*
- Feature Selection*
- Data mining*

4.2.1 *Text Preprocessing*

Pada tahap ini akan dilakukan proses pengubahan huruf besar yang ada pada masing-masing dokumen menjadi huruf kecil, selanjutnya dilakukan proses

penghilangan karakter atau simbol yang tidak berguna bagi proses pengelompokkan. Untuk lebih jelasnya dapat dilihat pada *flowchat* berikut :



Gambar 4.2 Flowchart Text Preprocessing

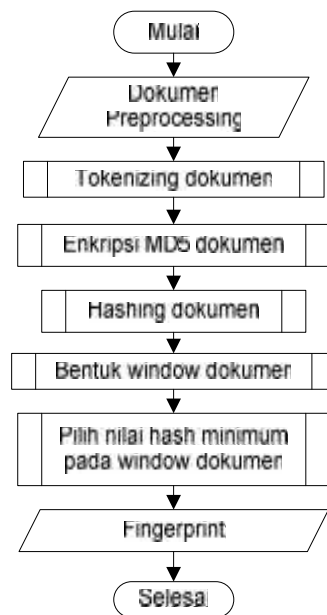
Berdasarkan gambar 4.2 di atas, proses *Preprocessing* dimulai pembacaan koleksi dokumen, lalu proses dilanjutkan mengubah setiap huruf besar ditemukan pada maka diganti dengan huruf kecil, dan jika simbol atau karakter tidak berguna ditemukan maka simbol atau karakter tersebut dibuang. Dokumen hasil *Preprocessing* akan disimpan dalam *array* dokumen *Preprocessing*. Hasil proses *Preprocessing* data koleksi dokumen pada tabel 4.1 adalah sebagai berikut :

Tabel 4.2 Dokumen Setelah Di-Preprocessing

Dokumen	Isi Dokumen
1	kata wahyu berasal dari bahasa arab yakni alwahyu di mana menurut pengertian bahasa kata alwahyu mengandung arti suara api dan kecepatan
2	makna kata wahyu secara konstektual menunjukkan banyak makna kata alwahyu mengandung arti suara api dan kecepatan di samping itu ia pun mempunyai arti bisikan
3	jaringan komputer merupakan sejumlah komputer yang saling terhubung yang didesain agar dapat saling berbagi sumber daya dan bertukar informasi
4	jaringan komputer merupakan sesuatu sangat penting dengan adanya jaringan komputer sejumlah komputer dapat saling terhubung hingga dapat saling berbagi sumber daya dan bertukar informasi

4.2.2 Text Transformation

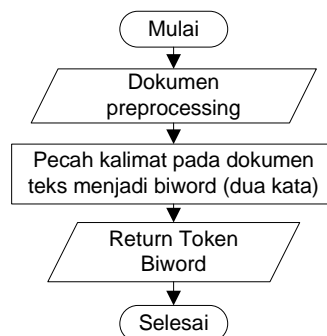
Setelah dokumen yang telah di-*Preprocessing* didapatkan, dilanjutkan dengan proses-proses yang ada pada tahap *Text transformation*. Ada beberapa proses yang terdapat dalam tahap *Text transformation*, yaitu *tokenizing*, enkripsi MD5, *hashing*, pembentukan window, dan pemilihan nilai *hash* minimum untuk dijadikan sebagai *fingerprint*. Untuk lebih jelasnya dapat dilihat pada *flowchart* berikut ini :



Gambar 4.3 Flowchart Text Transformation

4.2.2.1 Proses Tokenizing

Proses *tokenizing* adalah proses pemotongan kalimat yang terdapat pada dokumen menjadi 2 kata(*biword*). Proses *tokenizing* dapat dilihat pada *flowchart* berikut :



Gambar 4.4 Flowchart Proses Tokenizing

Berdasarkan gambar 4.4 di atas, proses *tokenizing* yaitu memecah setiap kalimat pada dokumen menjadi 2 kata. Hasil dari proses ini disimpan dalam *array* token *biword* agar dapat digunakan pada proses selanjutnya. Hasil proses *tokenizing* dengan menggunakan dokumen *Preprocessing* pada tabel 4.2 adalah sebagai berikut :

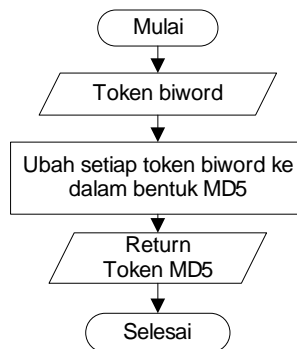
Tabel 4.3 Dokumen Hasil *Tokenizing*

Dokumen	Token Biword
1	[0] => kata wahyu [1] => wahyu berasal [2] => berasal dari [3] => dari bahasa [4] => bahasa arab [5] => arab yakni [6] => yakni alwahyu [7] => alwahyu di [8] => di mana [9] => mana menurut [10] => menurut pengertian [11] => pengertian bahasa [12] => bahasa kata [13] => kata alwahyu [14] => alwahyu mengandung [15] => mengandung arti [16] => arti suara [17] => suara api [18] => api dan [19] => dan kecepatan
2	[0] => makna kata [1] => kata wahyu [2] => wahyu secara [3] => secara konstektual [4] => konstektual menunjukkan [5] => menunjukkan banyak [6] => banyak makna [7] => makna kata [8] => kata alwahyu [9] => alwahyu mengandung [10] => mengandung arti [11] => arti suara [12] => suara api [13] => api dan [14] => dan kecepatan [15] => kecepatan di [16] => di samping [17] => samping itu [18] => itu ia [19] => ia pun [20] => pun mempunyai [21] => mempunyai arti [22] => arti bisikan

3	[0] => jaringan komputer [1] => komputer merupakan [2] => merupakan sejumlah [3] => sejumlah komputer [4] => komputer yang [5] => yang saling [6] => saling terhubung [7] => terhubung yang [8] => yang didesain [9] => didesain agar [10] => agar dapat [11] => dapat saling [12] => saling berbagi [13] => berbagi sumber [14] => sumber daya [15] => daya dan [16] => dan bertukar [17] => bertukar informasi
4	[0] => jaringan komputer [1] => komputer merupakan [2] => merupakan sesuatu [3] => sesuatu sangat [4] => sangat penting [5] => penting dengan [6] => dengan adanya [7] => adanya jaringan [8] => jaringan komputer [9] => komputer sejumlah [10] => sejumlah komputer [11] => komputer dapat [12] => dapat saling [13] => saling terhubung [14] => terhubung hingga [15] => hingga dapat [16] => dapat saling [17] => saling berbagi [18] => berbagi sumber [19] => sumber daya [20] => daya dan [21] => dan bertukar [22] => bertukar informasi

4.2.2.2 Proses Enkripsi MD5

Setelah token *biword* didapatkan, proses selanjutnya adalah melakukan proses enkripsi token *biword* ke dalam bentuk MD5. Hal ini dilakukan agar panjang string masing-masing token sama, yakni 32 karakter. Untuk lebih jelas, proses enkripsi MD5 dapat dilihat pada *flowchart* berikut ini :



Gambar 4.5 Flowchart Proses Enkripsi MD5

Proses enkripsi token *biword* ke dalam bentuk MD5 proses pengubahan setiap token *biword* menjadi bentuk MD5. Token *biword* yang telah dienkripsi disimpan dalam *array* token MD5. Hasil proses enkripsi token *biword* pada tabel 4.3 ke dalam bentuk MD5 adalah sebagai berikut :

Tabel 4.4 Dokumen Hasil Proses Enkripsi MD5

Dokumen	Token MD5
1	[0] => 539991ce7a6722cd49611851576fd4a4
	[1] => c12b422bd86b266de11b327c44324f10
	[2] => 8ac83c71daa0e91434db62a5b0333837
	[3] => 2b29a0b20a2bb49d5dd1fa825de40f84
	[4] => dad900b22110817c8de1dfed47685ce7
	[5] => 8c712d4b409982b3f63fbfe227758df9
	[6] => 161eac68c842b17c2e609f1a7051f1ce
	[7] => c910fc443c9cbae230b40310ca3189c4
	[8] => 37df9a1e1edda4f5e30dc4b3d137aa96
	[9] => 48d2928cb231ab27b54efe355664d957
	[10] => 1f755f65941a9fb67e60b112c768cd53
	[11] => b06c1e023fa1c80528a1bd64d011fe06
	[12] => c51821eecd37ff874d52ab471473b3e8
	[13] => 8e35f9fb2f9ddc4ff55383ee93d6862e
	[14] => 1b7124238a192ac57285b3c953887571
	[15] => 5a740554aae6f805d4c11609921ba943
	[16] => 9e6d5b6c2f6b4d5b483c6aca0d739a2a
	[17] => 7de1bcc6bf3613762db84a5f4e15f536
	[18] => 03917426abe5c472bc9c65d588d48857
	[19] => de7f1001596f96bfe78443d10eb7e48c

2	<pre> [0] => a78880c47bcef5e247abdfdccefdafa6e [1] => 539991ce7a6722cd49611851576fd4a4 [2] => 71f1eadf7bbf10f6ea571e71a451d1ab [3] => 7e8411fbfe0d4483c9fbb7be0718acc8 [4] => 07d4080ce3641f70f1e3e1e249ed3c68 [5] => 185ca6c5c3cc7df9420f21f5fc8f215b [6] => 906982bd1734230e884cb12614ce1f09 [7] => a78880c47bcef5e247abdfdccefdafa6e [8] => 8e35f9fb2f9ddc4ff55383ee93d6862e [9] => 1b7124238a192ac57285b3c953887571 [10] => 5a740554aae6f805d4c11609921ba943 [11] => 9e6d5b6c2f6b4d5b483c6aca0d739a2a [12] => 7de1bcc6bf3613762db84a5f4e15f536 [13] => 03917426abe5c472bc9c65d588d48857 [14] => de7f1001596f96bfe78443d10eb7e48c [15] => 0fb03acc047c2a5557f29ccf51c11a4c [16] => 78493e0340151e3ab24f452b6c7028fa [17] => 5818d6ba2bb49695bb3e49280baa153b [18] => eeea686eeac7dd0ae8d6d6f6d8ea7db2 [19] => 36551a766145c57b777d701846954a4e [20] => b53075d95191eed7ff6341580a0a5aa2 [21] => 1fa44b8b7d6f7b3344f08b2bda3bf9ec [22] => df066a163c7c73a095fb4997b70c8a2c </pre>
3	<pre> [0] => b2135317ebfed035ca53c2ba8d9a30ca [1] => 0d1cfa1654bb5b80c66fb995cedf10df [2] => 007c6fad66bca0e15a151adc640719cd [3] => 85e6252907ebbd7d7ef0726dab222b48 [4] => 84521cb47f184b29e4cda9e73c1ca947 [5] => 2ab2023c94387eb5559df2ae3de35ab1 [6] => a25fce33261303e7d2f6a4b222bd4b1d [7] => 75a0d33e0dee629c7790c777f35aba38 [8] => af8ce5d14e5ecc38aea2a72ebe3d1323 [9] => 06743dd9990dab244cc4e6457a65e68d [10] => 887d4b1b7dc1028e46a4d658f44d2059 [11] => 3267909a011dd5fa71f50840f10fb282 [12] => 48b6a4e7c67f068f02e6cbb97efdf1b7 [13] => a6ca8172ec9ec543b36bad9cb291186d [14] => 6579642e40c34a1a0f9e0e9b456b8c9b [15] => 6d6298cabf461c614e8e75a16e954036 [16] => 1779669b2b6d00c62d16654748ed2e28 [17] => 5c8ec4e0f4bb052127415cb91de9dc4b </pre>

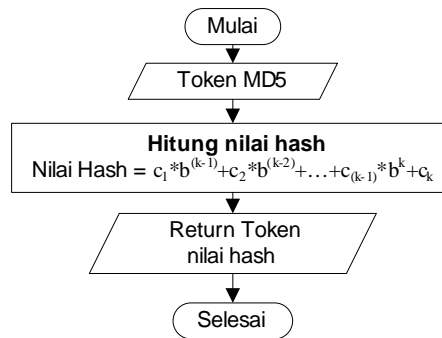

```

[0] => b2135317ebfed035ca53c2ba8d9a30ca
[1] => 0d1cfa1654bb5b80c66fb995cedf10df
[2] => 22876d5cc30a38806d6dde73d61635be
[3] => 8f70d785fc0e9e797ebcc633cd3992d9
[4] => efc8bd2212a87b70ea89b02ef038daa2
[5] => 8c5836290a16557126dbbce90aae0a7a
[6] => f2a61ad230e0e3ddaa3a724d5a2f425e
[7] => bfc18b8c79c0a34e8045b01b97f3e3b2
[8] => b2135317ebfed035ca53c2ba8d9a30ca
[9] => 3a554ae2ba81a0642718bb94bebf4b31
[10] => 85e6252907ebbd7d7ef0726dab222b48
[11] => 3c87f230c2ae583c17bb9f3dac39137e
[12] => 3267909a011dd5fa71f50840f10fb282
[13] => a25fce33261303e7d2f6a4b222bd4b1d
[14] => 7dc40efedaeff1403aac419af31c15a9
[15] => df927a2e0d853a54efbeb4908fc142e3
[16] => 3267909a011dd5fa71f50840f10fb282
[17] => 48b6a4e7c67f068f02e6cbb97efdf1b7
[18] => a6ca8172ec9ec543b36bad9cb291186d
[19] => 6579642e40c34a1a0f9e0e9b456b8c9b
[20] => 6d6298cabf461c614e8e75a16e954036
[21] => 1779669b2b6d00c62d16654748ed2e28
[22] => 5c8ec4e0f4bb052127415cb91de9dc4b

```

4.2.2.3 Proses Hashing

Proses *hashing* adalah proses perhitungan nilai *hash* token MD5. Proses perhitungan nilai *hash* dapat dilihat pada *flowchart* berikut :



Gambar 4.6 Flowchart Proses Hashing

Berdasarkan gambar 4.6, nilai hash dihitung dengan menggunakan persamaan 2.1, dimana c adalah nilai *ascii* karakter, b adalah nilai basis prima, dan k adalah banyak karakter pada token. Pada penelitian ini penulis menetapkan nilai $b=2$, nilai ini dipilih karena merupakan nilai basis prima terbaik untuk menghasilkan *fingerprint* dokumen yang berkualitas (Muhammad Ridho, 2013). Perhitungan nilai *hash* dengan menggunakan token MD5 pada tabel 4.4 dan persamaan 2.1 adalah sebagai berikut :

$$H_{(c_1 \dots c_k)} = c_1 * b^{(k-1)} + c_2 * b^{(k-2)} + \dots + c_{(k-1)} * b^k + c_k$$

$$539991ce7a6722cd49611851576fd4a4 = \text{ascii}(5) * 2^{(32-1)} + \text{ascii}(3) * 2^{(32-2)} + \text{ascii}(9) * 2^{(32-3)} + \dots + \text{ascii}(a) * 2^{(32-31)} + \text{ascii}(4) * 2^{(32-32)}$$

$$539991ce7a6722cd49611851576fd4a4 = 53 * 2^{31} + 51 * 2^{30} + 57 * 2^{29} + \dots + 97 * 2^1 + 52 * 2^0$$

$$539991ce7a6722cd49611851576fd4a4 = 231536545094$$

Dengan menggunakan cara perhitungan yang sama di dapatkan nilai *hash* selanjutnya sebagai berikut :

Tabel 4.5 Dokumen Hasil Perhitungan Nilai Hash

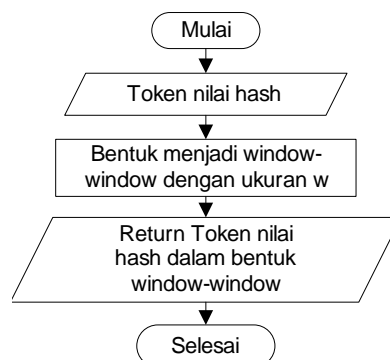
Dokumen	Token Nilai Hash
1	[0] => 231536545094
	[1] => 333369992522
	[2] => 310332283045
	[3] => 276213706012
	[4] => 402518953013
	[5] => 286932474181
	[6] => 240350746959
	[7] => 337110381774
	[8] => 268379466820
	[9] => 254735525285
	[10] => 276141258437
	[11] => 334553988234
	[12] => 327440999278
	[13] => 294065276801
	[14] => 266979401707
	[15] => 275678405319
	[16] => 304986008097
	[17] => 318355928400
	[18] => 216624451073
	[19] => 393307761339

2	<pre> [0] => 328664631125 [1] => 231536545094 [2] => 265017284408 [3] => 287056783858 [4] => 244550088424 [5] => 242671128896 [6] => 234935310825 [7] => 328664631125 [8] => 294065276801 [9] => 266979401707 [10] => 275678405319 [11] => 304986008097 [12] => 318355928400 [13] => 216624451073 [14] => 393307761339 [15] => 297342589447 [16] => 238414781245 [17] => 238397986260 [18] => 421734562670 [19] => 226928994273 [20] => 323800577648 [21] => 299032128001 [22] => 381967277051 </pre>
3	<pre> [0] => 318899477167 [1] => 287125347350 [2] => 230820125702 [3] => 259832555032 [4] => 236470423579 [5] => 291669771633 [6] => 341323065838 [7] => 261430266530 [8] => 396929009355 [9] => 223429476260 [10] => 254800349619 [11] => 222281078810 [12] => 261064518063 [13] => 360534464488 [14] => 232779288640 [15] => 283590873260 [16] => 224646586784 [17] => 299699652646 </pre>

4	<pre> [0] => 318899477167 [1] => 287125347350 [2] => 224591258005 [3] => 293624287489 [4] => 417934620536 [5] => 284174189955 [6] => 357566037655 [7] => 404951114522 [8] => 318899477167 [9] => 275547815583 [10] => 259832555032 [11] => 281622224983 [12] => 222281078810 [13] => 341323065838 [14] => 312577594983 [15] => 386621332053 [16] => 222281078810 [17] => 261064518063 [18] => 360534464488 [19] => 232779288640 [20] => 283590873260 [21] => 224646586784 [22] => 299699652646 </pre>
---	--

4.2.2.4 Pembentukan *Window*

Setelah nilai *hash* dihitung, proses selanjutnya yaitu melakukan pembentukan *window* dengan nilai *w*. Proses pembentukan *window* adalah proses memecah *array* token nilai *hash* menjadi kumpulan *array-array* baru sejumlah *w*. Untuk lebih jelas dapat dilihat pada *flowchart* proses pembentukan *window* berikut :



Gambar 4.7 Flowchart Proses Pembentukan *Window*

Pada penelitian ini penulis menggunakan nilai $w=8$, karena merupakan nilai yang dapat menghasilkan *fingerprint* yang berkualitas (Muhammad Ridho, 2013). Proses pembentukan *window* dengan menggunakan token nilai *hash* pada tabel 4.5 sebagai berikut :

Tabel 4.6 Hasil Proses Pembentukan Window

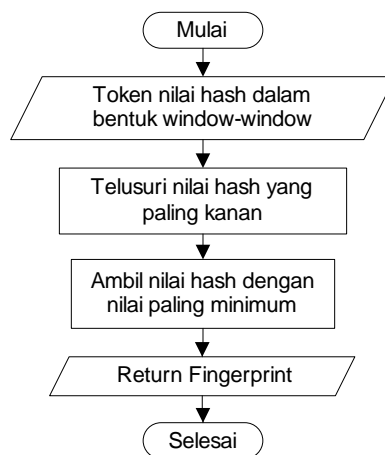
Dokumen	Token Nilai Hash Dalam Window
1	<p>[231536545094, 333369992522, 310332283045, 276213706012, 402518953013, 286932474181, 240350746959, 337110381774]</p> <p>[333369992522, 310332283045, 276213706012, 402518953013, 286932474181, 240350746959, 337110381774, 268379466820]</p> <p>[310332283045, 276213706012, 402518953013, 286932474181, 240350746959, 337110381774, 268379466820, 254735525285]</p> <p>[276213706012, 402518953013, 286932474181, 240350746959, 337110381774, 268379466820, 254735525285, 276141258437]</p> <p>[402518953013, 286932474181, 240350746959, 337110381774, 268379466820, 254735525285, 276141258437, 334553988234]</p> <p>[286932474181, 240350746959, 337110381774, 268379466820, 254735525285, 276141258437, 334553988234, 327440999278]</p> <p>[240350746959, 337110381774, 268379466820, 254735525285, 276141258437, 334553988234, 327440999278, 294065276801]</p> <p>[337110381774, 268379466820, 254735525285, 276141258437, 334553988234, 327440999278, 294065276801, 266979401707]</p> <p>[268379466820, 254735525285, 276141258437, 334553988234, 327440999278, 294065276801, 266979401707, 275678405319]</p> <p>[254735525285, 276141258437, 334553988234, 327440999278, 294065276801, 266979401707, 275678405319, 304986008097]</p> <p>[276141258437, 334553988234, 327440999278, 294065276801, 266979401707, 275678405319, 304986008097, 318355928400]</p> <p>[334553988234, 327440999278, 294065276801, 266979401707, 275678405319, 304986008097, 318355928400, 216624451073]</p> <p>[327440999278, 294065276801, 266979401707, 275678405319, 304986008097, 318355928400, 216624451073, 393307761339]</p>
2	<p>[328664631125, 231536545094, 265017284408, 287056783858, 244550088424, 242671128896, 234935310825, 328664631125]</p> <p>[231536545094, 265017284408, 287056783858, 244550088424, 242671128896, 234935310825, 328664631125, 294065276801]</p> <p>[265017284408, 287056783858, 244550088424, 242671128896, 234935310825, 328664631125, 294065276801, 266979401707]</p> <p>[287056783858, 244550088424, 242671128896, 234935310825, 328664631125, 294065276801, 266979401707, 275678405319]</p> <p>[244550088424, 242671128896, 234935310825, 328664631125, 294065276801, 266979401707, 275678405319, 304986008097]</p> <p>[242671128896, 234935310825, 328664631125, 294065276801, 266979401707, 275678405319, 304986008097, 318355928400]</p> <p>[234935310825, 328664631125, 294065276801, 266979401707, 275678405319, 304986008097, 318355928400, 216624451073]</p> <p>[328664631125, 294065276801, 266979401707, 275678405319, 304986008097, 318355928400, 216624451073, 393307761339]</p> <p>[294065276801, 266979401707, 275678405319, 304986008097, 318355928400, 216624451073, 393307761339, 297342589447]</p> <p>[266979401707, 275678405319, 304986008097, 318355928400, 216624451073, 393307761339, 297342589447, 238414781245]</p> <p>[275678405319, 304986008097, 318355928400, 216624451073, 393307761339, 297342589447, 238414781245, 238397986260]</p>

	<p>[304986008097,318355928400,216624451073,393307761339,297342589447,238414781245,238397986260,421734562670]</p> <p>[318355928400,216624451073,393307761339,297342589447,238414781245,238397986260,421734562670,226928994273]</p> <p>[216624451073,393307761339,297342589447,238414781245,238397986260,421734562670,226928994273,323800577648]</p> <p>[393307761339,297342589447,238414781245,238397986260,421734562670,226928994273,323800577648,299032128001]</p> <p>[297342589447,238414781245,238397986260,421734562670,226928994273,323800577648,299032128001,381967277051]</p>
3	<p>[318899477167,287125347350,230820125702,259832555032,236470423579,291669771633,341323065838,261430266530]</p> <p>[287125347350,230820125702,259832555032,236470423579,291669771633,341323065838,261430266530,396929009355]</p> <p>[230820125702,259832555032,236470423579,291669771633,341323065838,261430266530,396929009355,223429476260]</p> <p>[259832555032,236470423579,291669771633,341323065838,261430266530,396929009355,223429476260,254800349619]</p> <p>[236470423579,291669771633,341323065838,261430266530,396929009355,223429476260,254800349619,222281078810]</p> <p>[291669771633,341323065838,261430266530,396929009355,223429476260,254800349619,222281078810,261064518063]</p> <p>[341323065838,261430266530,396929009355,223429476260,254800349619,222281078810,261064518063,360534464488]</p> <p>[261430266530,396929009355,223429476260,254800349619,222281078810,261064518063,360534464488,232779288640]</p> <p>[396929009355,223429476260,254800349619,222281078810,261064518063,360534464488,232779288640,283590873260]</p> <p>[223429476260,254800349619,222281078810,261064518063,360534464488,232779288640,283590873260,224646586784]</p> <p>[254800349619,222281078810,261064518063,360534464488,232779288640,283590873260,224646586784,299699652646]</p>
4	<p>[318899477167,287125347350,224591258005,293624287489,417934620536,284174189955,357566037655,404951114522]</p> <p>[287125347350,224591258005,293624287489,417934620536,284174189955,357566037655,404951114522,318899477167]</p> <p>[224591258005,293624287489,417934620536,284174189955,357566037655,404951114522,318899477167,275547815583]</p> <p>[293624287489,417934620536,284174189955,357566037655,404951114522,318899477167,275547815583,259832555032]</p> <p>[417934620536,284174189955,357566037655,404951114522,318899477167,275547815583,259832555032,281622224983]</p> <p>[284174189955,357566037655,404951114522,318899477167,275547815583,259832555032,281622224983,222281078810]</p> <p>[357566037655,404951114522,318899477167,275547815583,259832555032,281622224983,222281078810,341323065838]</p> <p>[404951114522,318899477167,275547815583,259832555032,281622224983,222281078810,341323065838,312577594983]</p> <p>[318899477167,275547815583,259832555032,281622224983,222281078810,341323065838,312577594983,386621332053]</p> <p>[275547815583,259832555032,281622224983,222281078810,</p>

	341323065838,312577594983,386621332053,222281078810]
	[259832555032,281622224983,222281078810,341323065838,312577594983,386621332053,222281078810,261064518063]
	[281622224983,222281078810,341323065838,312577594983,386621332053,222281078810,261064518063,360534464488]
	[222281078810,341323065838,312577594983,386621332053,222281078810,261064518063,360534464488,232779288640]
	[341323065838,312577594983,386621332053,222281078810,261064518063,360534464488,232779288640,283590873260]
	[312577594983,386621332053,222281078810,261064518063,360534464488,232779288640,283590873260,224646586784]
	[386621332053,222281078810,261064518063,360534464488,232779288640,283590873260,224646586784,299699652646]

4.2.2.5 Pembentukan *Fingerprint*

Setelah *window* masing-masing dokumen terbentuk langkah selanjutnya yaitu melakukan pembentukan *fingerprint* dengan cara menelusuri nilai *hash* yang paling kanan dan memilih nilai *hash* terkecil. Proses pembentukan *fingerprint* dapat dilihat pada *flowchart* berikut :



Gambar 4.8 Flowchart Proses Pembentukan *Fingerprint*

Berdasarkan window yang telah dibentuk pada tabel 4.6, nilai yang di-**bold** adalah nilai minimum yang dipilih sebagai *fingerprint*. Sehingga didapat *fingerprint* dokumen sebagai berikut :

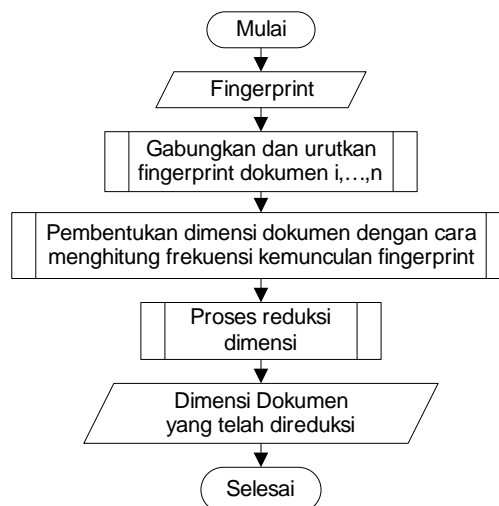
Tabel 4.7 Hasil Proses Pembentukan *Fingerprint*

Dokumen	<i>Fingerprint</i>
1	[0] => 231536545094
	[1] => 240350746959
	[2] => 254735525285
	[3] => 266979401707
	[4] => 216624451073

2	[0] => 231536545094 [1] => 234935310825 [2] => 216624451073 [3] => 226928994273
3	[0] => 230820125702 [1] => 223429476260 [2] => 222281078810
4	[0] => 224591258005 [1] => 259832555032 [2] => 222281078810

4.2.3 Feature Selection

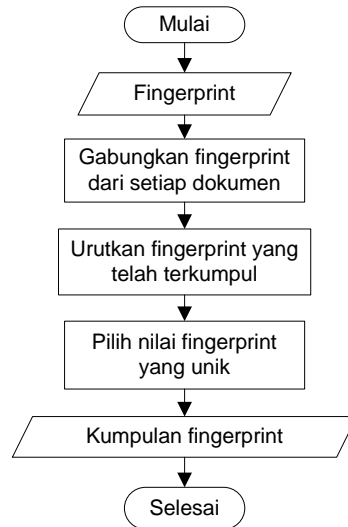
Setelah *fingerprint* dokumen didapatkan dalam tahap *Text transformation*, dilanjutkan dengan beberapa proses yang ada dalam tahapan *Feature Selection* untuk membentuk dimensi vector/dokumen dan mereduksinya. Beberapa proses yang ada dalam *Feature Selection* yaitu proses penggabungan dan pengurutan *fingerprint* setiap dokumen yang ada pada koleksi, pembentukan dimensi dan reduksi dimensi. Tahap *Feature Selection* dapat dilihat pada *flowchart* berikut :



Gambar 4.9 Flowchart Tahap Feature Selection

4.2.3.1 Penggabungan Dan Pengurutan *Fingerprint*

Proses pertama dalam tahapan *Feature Selection* adalah proses penggabungan dan pengurutan. Proses ini dilakukan dengan tujuan untuk membentuk *array* kumpulan *fingerprint* yang muncul pada seluruh dokumen yang ada pada koleksi. *Flowchart* proses penggabungan dan pengurutan *fingerprint* adalah sebagai berikut :



Gambar 4.10 Flowchart Proses Penggabungan dan Pengurutan *Fingerprint*

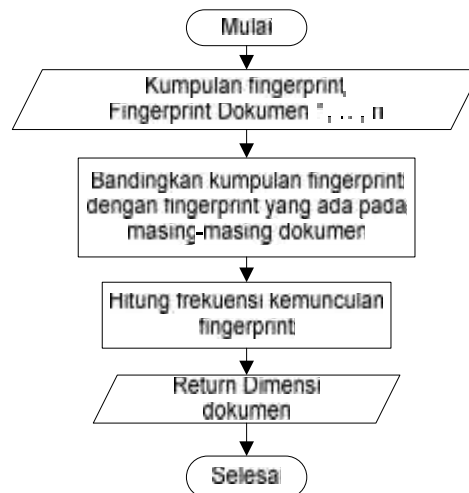
Hasil dari proses penggabungan dan pengurutan *fingerprint* dengan menggunakan *fingerprint* yang dihasilkan pada tabel 4.7 adalah sebagai berikut :

Tabel 4.8 Hasil Proses Penggabungan dan Pengurutan *Fingerprint*

No.	<i>Fingerprint</i>
1	216624451073
2	222281078810
3	223429476260
4	224591258005
5	226928994273
6	230820125702
7	231536545094
8	234935310825
9	240350746959
10	254735525285
11	259832555032
12	266979401707

4.2.3.2 Pembentukan Dimensi

Setelah daftar *fingerprint* didapatkan proses pembentukan dimensi dapat dimulai. Pembentukan dimensi dilakukan dengan cara membandingkan *array* daftar *fingerprint* dengan *array fingerprint* masing-masing dokumen, lalu menghitung frekuensi kemunculan *fingerprint* tersebut. Untuk lebih jelas, dapat dilihat pada *flowchart* berikut :



Gambar 4.11 Flowchart Proses Pembentukan Dimensi

Proses pembentukan dimensi dokumen dengan membandingkan kumpulan *fingerprint* pada tabel 4.8 dan *fingerprint* pada tabel 4.7 adalah sebagai berikut :

Tabel 4.9 Pembentukan Dimensi

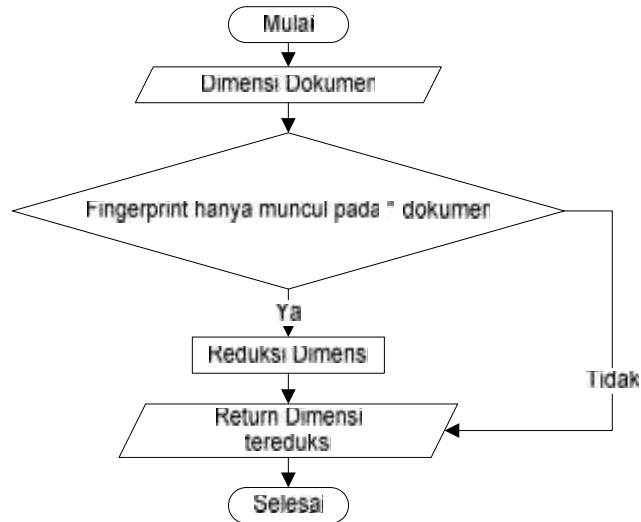
<i>Daftar Fingerprint</i>	<i>Daftar Biword</i>	Frekuensi Kemunculan Fingerprint Pada Dokumen			
		1	2	3	4
216624451073	Api dan	1	1	0	0
222281078810	Dapat saling	0	0	1	1
223429476260	Didesain agar	0	0	1	0
224591258005	Merupakan sesuatu	0	0	0	1
226928994273	Ia pun	0	1	0	0
230820125702	Merupakan sejumlah	0	0	1	0
231536545094	Kata wahyu	1	1	0	0
234935310825	Banyak makna	0	1	0	0
240350746959	Yakni alwahyu	1	0	0	0
254735525285	Mana menurut	1	0	0	0
259832555032	Sejumlah komputer	0	0	0	1
266979401707	Alwahyu mengandung	1	0	0	0

Berdasarkan tabel 4.9, didapat dimensi dokumen sebagai berikut :

- Dokumen 1 : 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1
- Dokumen 2 : 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0
- Dokumen 3 : 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0
- Dokumen 4 : 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0

4.2.3.3 Reduksi Dimensi

Proses reduksi dimensi dilakukan dengan tujuan untuk membuang dimensi-dimensi yang tidak relevan bagi proses pengelompokan nantinya. Proses reduksi dimensi dapat dilihat pada *flowchart* berikut :



Gambar 4.12 Flowchart Proses Reduksi Dimensi

Proses reduksi dimensi dokumen adalah proses pembuangan dimensi jika *fingerprint* hanya muncul pada 1 dokumen. Warna kuning pada tabel 4.9 adalah sebagai tanda bahwa dimensi-dimensi tersebut akan direduksi pada proses reduksi dimensi. Sehingga didapatkan dimensi dokumen yang telah direduksi sebagai berikut :

Tabel 4.10 Dimensi Yang Telah Direduksi

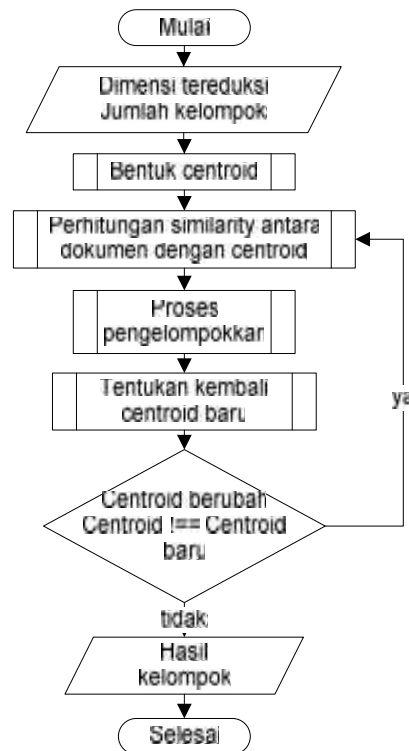
Token <i>Biword</i>	<i>Fingerprint</i>	Frekuensi Kemunculan <i>Fingerprint</i> Pada Dokumen			
		1	2	3	4
Api dan	216624451073	1	1	0	0
Dapat saling	222281078810	0	0	1	1
Kata wahyu	231536545094	1	1	0	0

Berdasarkan tabel 4.10, diperoleh dimensi masing-masing dokumen sebagai berikut :

- Dokumen 1 : 1, 0, 1
- Dokumen 2 : 1, 0, 1
- Dokumen 3 : 0, 1, 0
- Dokumen 4 : 0, 1, 0

4.2.4 Data mining

Pada tahap *Data mining*, dilakukan proses *clustering* itu sendiri dengan menggunakan metode k-means. Tahapan ini terdiri dari beberapa proses yang dapat dilihat pada *flowchart* berikut :



Gambar 4.13 Flowchart Tahap Data mining

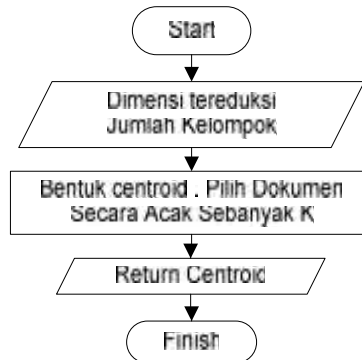
Berdasarkan gambar 4.13, proses yang terdapat dalam tahapan *Data mining* yaitu pembentukan *centroid*, perhitungan *similarity*, pengelompokkan, dan proses pembentukan *centroid* baru. Proses *clustering* berhenti jika *centroid* tidak lagi mengalami perubahan.

4.2.4.1 Pembentukan *Centroid*

Proses pertama yang ada dalam tahap *Data mining* adalah proses pembentukan *centroid* awal atau titik pusat kluster. Pada penelitian ini penulis menggunakan 2 cara pembentukan *centroid*. Pembentukan *centroid* cara pertama yaitu memilih secara acak dari dokumen ada, sedangkan cara kedua adalah membentuk setiap dimensi dokumen dengan cara menentukan nilainya secara acak berdasarkan *range* nilai dimensi terbesar dan terkecil dimensi dokumen.

a. Pembentukan *Centroid* Cara Pertama

Pada pembentukan *centroid* cara pertama, *centroid* dibentuk dengan cara memilih secara acak dari dokumen ada. *Flowchart* pembentukan *centroid* cara pertama adalah sebagai berikut :



Gambar 4.14 *Flowchart* Pembentukan *Centroid* Cara Pertama

Data yang digunakan pada proses ini adalah *array* dimensi dokumen yang direduksi dan jumlah kelompok. Pembentukan *centroid* cara I dengan menggunakan data dimensi dokumen pada tabel 4.10 dengan jumlah $k=2$ adalah sebagai berikut :

Tabel 4.11 Pembentukan *Centroid* Cara I

<i>Centroid</i>	Dimensi Dokumen Terpilih
1	1, 0, 1
2	0, 1, 0

b. Pembentukan *Centroid* Cara Kedua

Pada pembentukan *centroid* cara pertama, membentuk setiap dimensi dokumen dengan cara menentukan nilainya secara acak berdasarkan range nilai dimensi terbesar dan terkecil dimensi dokumen. Berikut *flowchart* proses pembentukan *centroid* cara kedua :



Gambar 4.15 Flowchart Pembentukan Centroid Cara Kedua

Pada pembentukan *centroid* cara kedua, proses diawali dengan menentukan nilai terkecil dan terbesar dari *array* dimensi dokumen tereduksi. Selanjutnya dilakukan proses pembentukan *centroid* sebanyak *k* dengan cara menentukan nilainya secara acak berdasarkan *range* nilai terkecil dan terbesar. Pembentukan *centroid* cara II dengan menggunakan data dimensi dokumen pada tabel 4.10 dengan jumlah $k=2$ adalah sebagai berikut :

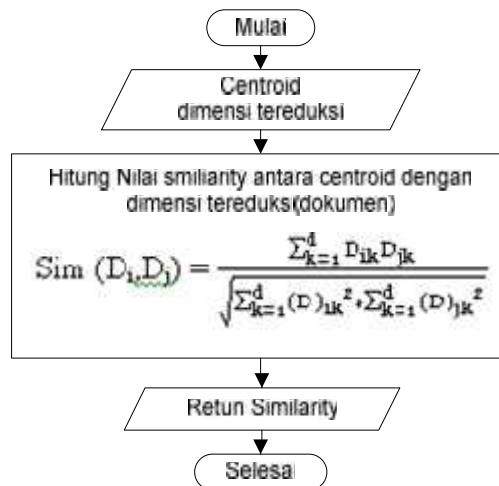
- Berdasarkan tabel 4.10, diperoleh :
 Nilai dimensi terbesar : 1, 1, 1
 Nilai dimensi terkecil : 0, 0, 0
- Sehingga *centroid* yang terbentuk yaitu sebagai berikut :

Tabel 4.12 Pembentukan Centroid Cara II

<i>Centroid</i>	Nilai Dimensi Centroid
1	0.79, 0.52, 0.09
2	0.15, 0.45, 0.82

4.2.4.2 Perhitungan *Similarity*

Pada proses ini dilakukan proses perhitungan *similarity* atau kemiripan antara *centroid* dengan dokumen. Proses perhitungan *similarity* antara *centroid* dengan dokumen dapat dilihat pada *flowchart* berikut :



Gambar 4.16 Flowchart Perhitungan Similarity

Perhitungan *similarity* dilakukan dengan menggunakan fungsi *cosine similarity* seperti pada persamaan 2.5, hasil dari *similarity* disimpan dalam *array similarity*. Perhitungan *similarity* antara *centroid* tabel 4.11 dengan dimensi dokumen tereduksi tabel 4.10 adalah sebagai berikut :

$$\text{Sim } (D_i, D_j) = \frac{\sum_k (D_i * D_j)}{\sum_k D_i^2 * \sum_k D_j^2}$$

$$\text{Sim } (C_1, D_1) = \frac{1*1 + 0*0 + (1*1)}{1^2 + 0^2 + 1^2 * (1^2 + 0^2 + 1)} = \frac{2}{2} = 1$$

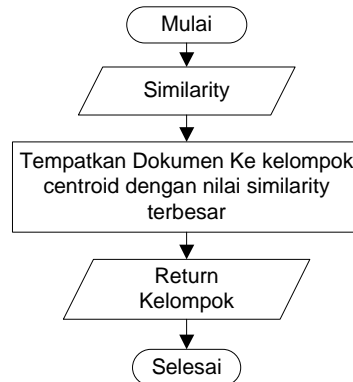
Dengan menggunakan cara yang sama, didapatkan hasil *similarity* antara *centroid* dan dokumen sebagai berikut :

Tabel 4.13 Hasil Perhitungan Similarity

Centroid	Similarity			
	Dokumen 1	Dokumen 2	Dokumen 3	Dokumen 4
1	1	1	0	0
2	0	0	1	1

4.2.4.3 Pengelompokkan

Setelah nilai *similarity* antara masing-masing *centroid* dengan dokumen-dokumen yang ada pada koleksi dokumen didapatkan, selanjutnya dilakukan pengelompokkan yaitu dengan cara menempatkan dokumen ke kelompok *centroid* yang memiliki nilai *similarity* terbesar dengan dokumen yang dimaksud. Proses pengelompokkan dapat dilihat pada *flowchart* sebagai berikut :



Gambar 4.17 Flowchart Proses Pengelompokkan

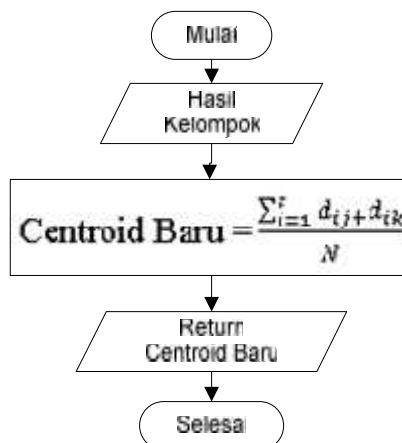
Jika melihat hasil *similarity* pada tabel 4.13, maka hasil kelompok yang di dapat adalah :

Tabel 4.14 Hasil Kelompok

Kelompok	Dokumen	Dimensi
1	1	1, 0, 1
	2	1, 0, 1
2	3	0, 1, 0
	4	0, 1, 0

4.2.4.4 Pembentukan *Centroid* Baru

Setelah dokumen telah masuk ke dalam kluster atau kelompoknya masing-masing, proses selanjutnya adalah pembentukan *centroid* baru. Pembentukan *centroid* baru dilakukan dengan cara menghitung rata-rata dimensi dokumen yang ada pada masing-masing kluster seperti pada persamaan 2.7. Untuk lebih jelas, proses pembentukan *centroid* baru dapat dilihat pada *flowchart* berikut :



Gambar 4.18 Flowchart Proses Pembentukan *Centroid* Baru

Centroid baru dibentuk berdasarkan jumlah rata-rata dimensi ke-i dokumen ke-j pada kluster t, dengan N adalah banyak dokumen pada kluster. Perhitungan *centroid* baru dengan menggunakan persamaan 2.7 dan data hasil kelompok tabel 4.14 adalah sebagai berikut :

$$\begin{aligned}
 - \text{Centroid baru I} &= \frac{\sum_{j=1}^t d_{ij} + d_{ik}}{N} = \frac{1+1}{2}, \frac{0+0}{2}, \frac{1+1}{2} = 1, 0, 1 \\
 - \text{Centroid Baru II} &= \frac{\sum_{j=1}^t d_{ij} + d_{ik}}{N} = \frac{0+0}{2}, \frac{1+1}{2}, \frac{0+0}{2} = 0, 1, 0
 \end{aligned}$$

Tabel 4.15 Hasil Pembentukan *Centroid* Baru

<i>Centroid</i>	Nilai Dimensi Baru <i>Centroid</i>
1	1, 0, 1
2	0, 1, 0

Setelah *centroid* baru terbentuk dilakukan pengecekan apakah *centroid* baru tabel 4.15 sama dengan *centroid* awal tabel 4.11. Jika tidak maka proses kembali kepada perhitungan *similarity*. Jika sama, maka proses pengelompokan dihentikan, sehingga didapatkan hasil kelompok akhir sebagai berikut :

Tabel 4.16 Hasil Kelompok Akhir

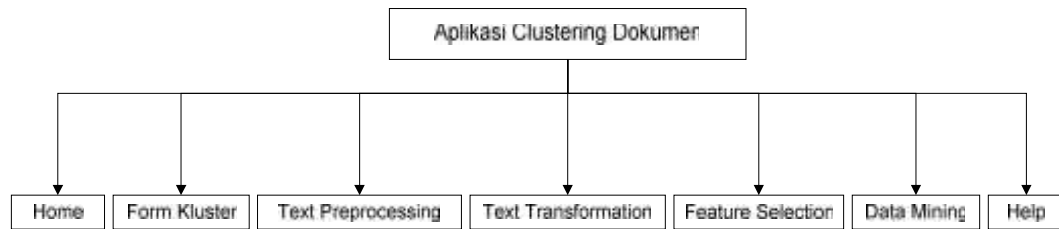
Kelompok	Dokumen	Dimensi
1	1	1, 0, 1
	2	1, 0, 1
2	3	0, 1, 0
	4	0, 1, 0

4.3 Perancangan Aplikasi

Pada tahap ini akan dibahas tentang perancangan aplikasi *clustering* dokumen teks berdasarkan tahapan analisa yang telah dibahas sebelumnya. Adapun perancangan yang akan dibuat adalah perancangan struktur menu dan perancangan *Interface* aplikasi.

4.3.1 Perancangan Struktur Menu

Perancangan struktur menu adalah tahap merancang menu-menu yang dapat digunakan pengguna untuk menjalankan aplikasi, sehingga dapat memudahkan pengguna dalam memilih proses yang akan dijalanannya. Untuk lebih jelasnya dapat dilihat pada gambar berikut :



Gambar 4.19 Rancangan Struktur Menu

4.3.2 Perancangan *Pseudo Code*

Perancangan *pseudo code* dibuat untuk mempermudah dalam memahami algoritma aplikasi yang akan dibangun dan mempermudah dalam mengimplementasikan kode program.

4.3.2.1 *Pseudo Code Text Preprocessing Dan Text Transformation*

Pada aplikasi ini, *Text Preprocessing* dan *Text Transformation* berada dalam satu pengulangan. Berikut *pseudo code* tahap *text preprocessing* dan *Text Transformation* aplikasi :

```

Function text_preprocessing_&_transformation (input $dok : String
                                             output $fingerprint :
                                             array)
Variable
    $hasil, $pieces, $space, $biword, $token_md5 : array
    $hash, $window, $min, $fingerprint, $w : array
    $folder, $handle, $file : boolean
    $idx_text, $i : integer
Algorithm
    $folder ← "C:/Dokumen Kluster"
    $handle ← opendir($folder)
    $idx_text ← 1
    While (false != ($file ← readdir($handle))) Do
        //Text Preprocessing
        $dok ← file_get_contents($folder.'\\'.$file)
        $hasil ← pembersihan_teks($dok)
        $pieces ← preg_split("/[\s]+/", $hasil)
        $space ← implode(" ", $pieces)

        //Text Transformation
        //Tokenizing
        $biword ← tokenizing($space,2)
  
```

```

//Enkripsi MD5
$token_md5 ← enkrip_md5($biword)

//Perhitungan Nilai Hash
For($i←0 to ($i<count($token_md5))
    $hash[$idx_text][$i] ← hitung_hash($token_md5[$i])
    $i++
End For

//Pembentukan Window
$window = array
For($i←0 to ($i < count($hash[$idx_text]))
    If(count(array_slice($hash[$idx_text], $i, 8)) >= 8)
        $window[] ← array_slice($hash[$idx_text], $i, 8)
    End If
    $i++
End For

//Pembentukan Fingerprint
Foreach($window as $w)
    $min[$idx_text][] ← min($w);
End Foreach

For($i←0 to ($i < count($min[$idx_text])))
    if($i==0)
        $fingerprint[$idx_text][] ← $min[$idx_text][$i]
    else($min[$idx_text][$i]!=$min[$idx_text][$i-1])
        $fingerprint [$idx_text][] ← $min[$idx_text][$i]
    End If
    End If
    $i++
End For
End While
return ($fingerprint)

```

4.3.2.2 Pseudo Code Feature Selection

Proses utama yang terdapat pada tahap ini adalah proses pembentukan dimensi dan proses reduksi dimensi. Berikut rancangan *pseudo code* tahap *feature selection* :

```

Function Feature Selection (input $fingerprint : array
                        output $dimensi_tereduksi : array)
Variable
    $i, $j : integer
    $all_fing, $sort_fing, dbsort_fing : array
Algorithm
//Pengumpulan Dan Pengurutan Fingerprint
$all_fing ← array_merge($fingerprint)

For($i←0 to ($i≤count($all_fing)-1)
    For($j←0 to ($j≤count($all_fing[$i])-1)
        $sort_fing[]←$all_fing[$i][$j]
    End For
End For

$dbsort_fing ← array_unique($sort_fing)
sort($dbsort_fing)

//Pembentukan Dimensi
$dimensi ← bentuk_dimensi($all_fing, $dbsort_fing)
//Reduksi Dimensi
$dimensi_tereduksi ← reduksi_dimensi($dimensi)
return ($dimensi_tereduksi)

```

4.3.2.3 Pseudo Code Data Mining

Tahap *data mining* adalah tahap *clustering* itu sendiri. Di dalam tahap ini dilakukan proses *clustering* dengan menggunakan metode *k-means*. Berikut rancangan *pseudo code* tahap *data mining* :

```

Function Data Mining(input $dimensi_tereduksi : array
                    $jml_kelompok : integer
                    output $hasil_kelompok : array)
Variable
    $jml_dimensi, $jml_dokumen, $ulang : integer
    $centroid, $centroid_baru, $similarity, $hasil_kelompok : array
    $changed : boolean
Algorithm
$changed ← true

//Bentuk Centroid Awal
$centroid ← buat_centroid($dimensi_tereduksi, $jml_kelompok)

```

```

$jml_dimensi ← count($dimensi_tereduksi[0])
$jml_dokumen ← count($dimensi_tereduksi)

While ($changed) Do
    //Perhitungan Similarity
    $similarity ← hitung_similarity($dimensi_tereduksi, $centroid)

    //Pengelompokkan
    $hasil_kelompok ← pengelompokan($similarity, $jml_kelompok, $jml_dokumen)

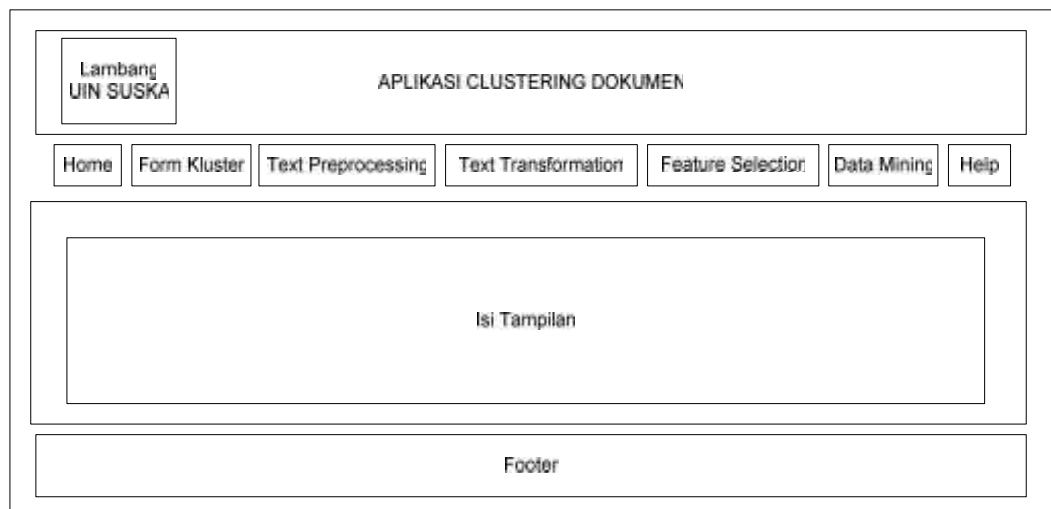
    //Pembuatan Centroid Baru
    $centroid_baru ← buat_centroid_baru($hasil_kelompok,
                                         $dimensi_tereduksi, $jml_dimensi)

    if($centroid == $centroid_baru)
        $changed ← false
        $centroid ← $centroid_baru
    End If
End While
return ($hasil_kelompok)

```

4.3.3 Perancangan *Interface*

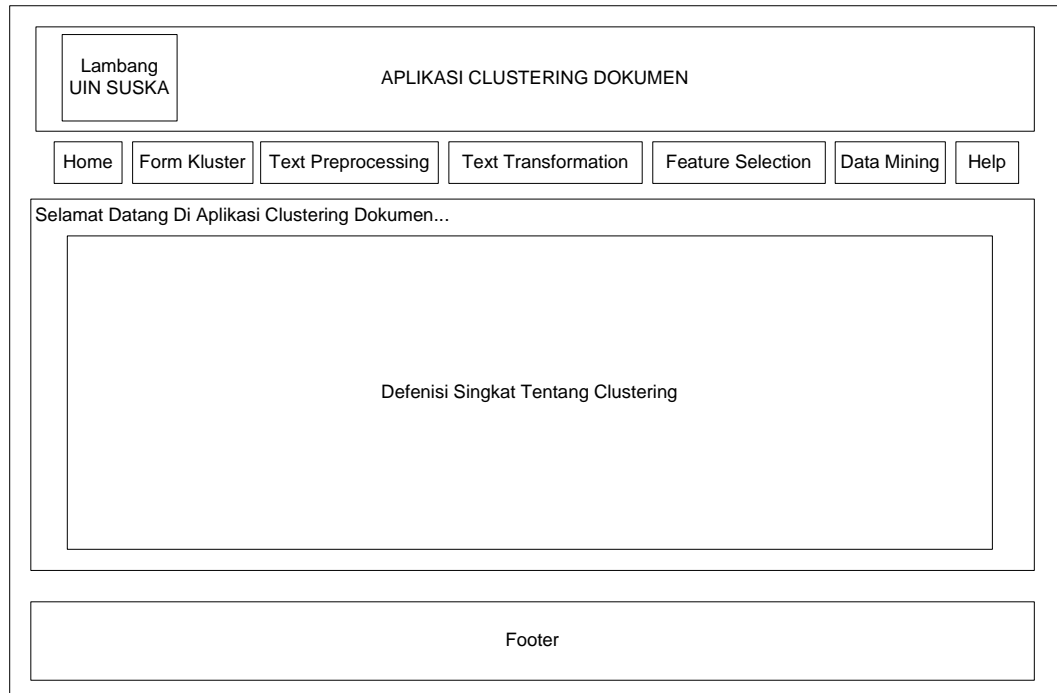
Interface adalah sarana pengembangan aplikasi yang digunakan agar terbangun komunikasi yang lebih mudah dan konsisten antara aplikasi dan pemakainya. Perancangan *Interface* menekankan pada tampilan yang menarik, akan tetapi mudah dipahami oleh penggunanya. Berikut adalah rancangan *Interface* aplikasi *clustering* dokumen yang akan dibangun :



Gambar 4.20 Rancangan *Interface* Aplikasi

4.3.3.1 Rancangan Menu Halaman Utama (*Home*)

Menu *Home* adalah halaman utama yang ditampilkan aplikasi kepada pengguna. Halaman ini berisi tentang definisi singkat mengenai *clustering* dan aplikasi *clustering* yang dibuat oleh penulis. Berikut rancangan menu *Home* :



Gambar 4.21 Rancangan *Interface* Halaman *Home*

4.3.3.2 Rancangan Menu *Form Kluster*

Halaman ini digunakan oleh pengguna untuk melakukan proses *clustering* dokumen teks. Pada halaman ini pengguna harus menginputkan jumlah kelompok yang ingin dibentuk setelah sebelumnya memastikan bahwa dokumen-dokumen yang ingin dikluster atau dikelompokkan telah berada pada direktori yang semestinya (C:/Dokumen Kluster). Berikut adalah rancangan tampilan menu *Form Kluster* :

Gambar 4.22 Rancangan *Interface* Menu *Form Kluster*

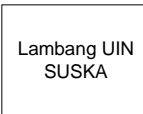
Ketika *button* Kelompokkan dijalankan, selanjutnya aplikasi akan menampilkan halaman yang memberikan informasi nama dokumen dan kelompoknya sebagai berikut :

Nama Dokumen	Kelompok Dokumen

Gambar 4.23 Rancangan *Interface* Tabel Informasi Kelompok Dokumen

4.3.3.3 Rancangan Menu *Text Preprocessing*

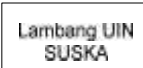
Menu *Text Preprocessing* merupakan halaman yang akan menampilkan informasi isi dokumen yang telah dilakukan proses *preprocessing*, yaitu proses *case folding*(pengubahan huruf besar ke huruf kecil) dan *filtering*(pembuangan karakter atau simbol yang tidak berguna). Untuk lebih jelas berikut rancangan *Interface* menu *Text Preprocessing* :

	APLIKASI CLUSTERING DOKUMEN						
<div style="display: flex; justify-content: space-around; margin-top: 5px;"> Home Form Kluster Text Preprocessing Text Transformation Feature Selection Data Mining Help </div>							
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Nama Dokumen</th> <th style="width: 50%;">Hasil Preprocessing</th> </tr> </thead> <tbody> <tr><td> </td><td> </td></tr> <tr><td> </td><td> </td></tr> </tbody> </table>		Nama Dokumen	Hasil Preprocessing				
Nama Dokumen	Hasil Preprocessing						
Footer							

Gambar 4.24 Rancangan *Interface* Menu *Text Preprocessing*

4.3.3.4 Rancangan Menu *Text Transformation*

Menu *Text Transformation* merupakan halaman yang akan menampilkan tabel informasi hasil proses *text transformation* terhadap dokumen, mulai dari *tokenizing* (pembentukan *biword*) hingga menghasilkan *fingerprint* dokumen. Untuk lebih jelas berikut rancangan *Interface* menu hasil *Text Transformation* :

	APLIKASI CLUSTERING DOKUMEN															
<div style="display: flex; justify-content: space-around; margin-top: 5px;"> Home Form Kluster Text Preprocessing Text Transformation Feature Selection Data Mining Help </div>																
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 20%;">Nama Dokumen</th> <th style="width: 15%;">Token Biword</th> <th style="width: 15%;">Token MD5</th> <th style="width: 15%;">Token Hash</th> <th style="width: 35%;">Fingerprint</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table>		Nama Dokumen	Token Biword	Token MD5	Token Hash	Fingerprint										
Nama Dokumen	Token Biword	Token MD5	Token Hash	Fingerprint												
Footer																

Gambar 4.25 Rancangan *Interface* Menu *Text Transformation*

4.3.3.5 Rancangan Menu *Feature Selection*

Menu *Feature Selection* merupakan halaman yang akan menampilkan informasi hasil proses *feature selection* terhadap dokumen, yaitu proses

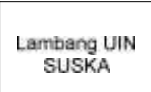
pembentukan dan reduksi dimensi. Untuk lebih jelas berikut rancangan *Interface* menu hasil *Feature Selection* :

		Dimensi Dokumen			
Token Biword	Fingerprint	Kelompok 1		Kelompok 2	
		Dok1	Dok2	Dok1	Dok2

Gambar 4.26 Rancangan *Interface* Menu *Feature Selection*

4.3.3.6 Rancangan Menu *Data Mining*

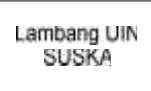
Menu *Data Mining* merupakan halaman yang menampilkan tabel informasi rekapitulasi hasil pengelompokan, yaitu informasi kelompok dan dokumen yang berada di dalam kelompok tersebut. Untuk lebih jelas berikut rancangan *Interface* menu hasil *Data mining* :

	APLIKASI CLUSTERING DOKUMEN								
<div style="display: flex; justify-content: space-around; margin-top: 5px;"> Home Form Kluster Text Preprocessing Text Transformation Feature Selection Data Mining Help </div>									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="padding: 5px;">Waktu Pengelompokkan</td> </tr> <tr> <td style="width: 50%; text-align: center; padding: 5px;">Kelompok Dokumen</td> <td style="width: 50%; text-align: center; padding: 5px;">Daftar Dokumen</td> </tr> <tr> <td style="height: 40px;"></td> <td></td> </tr> <tr> <td style="text-align: center; padding: 5px;">Total Kelompok .</td> <td style="text-align: center; padding: 5px;">Total Dokumen .</td> </tr> </table>		Waktu Pengelompokkan		Kelompok Dokumen	Daftar Dokumen			Total Kelompok .	Total Dokumen .
Waktu Pengelompokkan									
Kelompok Dokumen	Daftar Dokumen								
Total Kelompok .	Total Dokumen .								
<div style="border: 1px solid black; width: 100%; height: 20px; margin-top: 10px;"></div> Footer									

Gambar 4.27 Rancangan *Interface* Menu *Data Mining*

4.3.3.7 Rancangan Menu *Help*

Halaman ini adalah halaman yang menampilkan petunjuk penggunaan aplikasi *clustering* dokumen teks. Berikut rancangan *Interface* menu *help* :

	APLIKASI CLUSTERING DOKUMEN
<div style="display: flex; justify-content: space-around; margin-top: 5px;"> Home Form Kluster Text Preprocessing Text Transformation Feature Selection Data Mining Help </div>	
<div style="border: 1px solid black; width: 100%; height: 100px; margin-top: 10px;"></div> Tampilan Halaman Bantuan	
<div style="border: 1px solid black; width: 100%; height: 20px; margin-top: 10px;"></div> Footer	

Gambar 4.28 Rancangan *Interface* Menu *Help*